NITGEN®

RS-232C / LVCMOS serial protocol for Stand-Alone Fingerprint Recognition Device

Protocol Guide

Volume1. Overview & Commands

(Supported device: FIM40 Ver. 1.21 or higher,
FIM50 Ver 1.21 or higher,
FIM60 Ver 2.01 or higher)

Version 2.09

© Copyright 2015 NITGEN Co., Ltd.

ALL RIGHTS RESERVED

Serial Number:

Specifications can be changed without notice.

Revision Information

Date	Version	Description
2010-05-03	2.0	Release
2010-12-10	2.01	Change Command CMD_DELETE_ALL_FP Add New Command CMD_CTL_IO CMD_CFG_IO Add New Chapter Appendix F
2011-03-21	2.02	Add New Model – FIM5130 New GPIO feature is added – Finger Status Change Command CMD_AUTO_IDENTIFY
2011-04-06	2.03	Add Latent Option – SI_USING_LATENT
2011-05-19	2.04	Change Command CMD_AUTO_IDENTIFY_RESULT
2011-06-27	2.05	Change Description CMD_CHG_EMULMODE (configure only) CMD_CHG_LENGTH_OF_USERID (configure only) SI_LENGTH_OF_USER_ID (read only) SI_EMULATION_MODE (read only)
2011-08-01	2.06	Change Command CMD_GET_TEMPLATE
2011-09-06	2.07	Change Command CMD_GET_DEVICE_INFO (Supported device list) CMD_SET_DEFAULT_SYSINFO (GPIO's default values) CMD_SET_CAPTURE_PERIOD (The unit of tick) CMD_ADD_FP (Description) Change Description FIM40/50 mode → none emulation mode Change Appendix B

		Update commands for each emulation mode. Change Appendix D Description of SI_CAPTURE_TIMEOUT is changed Change Appendix E Typographical and logical errors are fixed
2012-02-24	2.08	Document Renewal Protocol Document is divided into 3 docs Protocol Overview is added Example and Example Condition is added RS232 UART Waveform is added Design alternation
2015-02-23	2.09	Add New Model – FIM60

Contents

1. GENERAL DESCRIPTION 8

2. PACKET DESCRIPTION 9

■ PACKET STRUCTURE 9

■ ERROR CODE 10

■ HOW TO MAKE THE HEADER CHECKSUM & THE DATA CHECKSUM 10

3 COMMAND DESCRIPTION 11

■ INITIALIZATION 11

CMD_REQUEST_CONNECTION 11

CMD_SET_BAUDRATE 12

CMD_GET_FIRMWARE_VERSION2 14

CMD_GET_DEVICE_INFO 15

■ MATCHING 16

CMD_VERIFY_FP 16

CMD_IDENTIFY_FP 19

CMD_IDENTIFY_RID_FP 22

CMD_INSTANT_MATCHING 25

CMD_GET_TEMPLATE 27

CMD_CANCEL 29

CMD_INSTANT_VERIFY 30

CMD_INSTANT_IDENTIFY 32

CMD_AUTO_IDENTIFY 35

CMD_AUTO_IDENTIFY_RESULT 37

■ DATABASE MANAGEMENT 39

CMD_DELETE_FP 39

CMD_DELETE_ALL_FP 41

CMD_SET_MASTER 42

CMD_LEAVE_MASTER_MODE 44

CMD_SET_MASTER_PASSWORD *45*
CMD_READ_USER_DATA *47*
CMD_WRITE_USER_DATA *49*
CMD_ERASE_USER_DATA_BLOCK *51*
CMD_DELETE_MASTER_PASSWORD *52*
CMD_ENTER_MASTER_MODE2 ..... *53*
CMD_GET_FP_LIST2 ..... *56*
CMD_GET_MASTER_LIST2 ..... *58*
CMD_READ_LOG_DATA2 ..... *60*
CMD_REGISTER_FP *62*
CMD_CHANGE_FP *66*
CMD_ADD_FP *69*
CMD_GET_FP *72*
CMD_DELETE_ALL_LOG *76*
CMD_REGISTER_MULTI_FP *77*
■ CONFIGURATION 80
CMD_SET_OPP_OPTION *80*
CMD_GET_OPP_OPTION *82*
CMD_SET_SECURITY_LEVEL *84*
CMD_GET_SECURITY_LEVEL *86*
CMD_SET_CAPTURE_OPTION *88*
CMD_GET_CAPTURE_OPTION *90*
CMD_SET_LOG_OPTION *92*
CMD_GET_LOG_OPTION *93*
CMD_SET_CAPTURE_PERIOD *94*
CMD_GET_CAPTURE_PERIOD *95*
CMD_SET_SYSINFO *96*
CMD_GET_SYSINFO *98*
CMD_SAVE_SYSINFO *100*
CMD_CHG_NUM_OF_TEMP *101*
CMD_SET_DEFAULT_SYSINFO *102*
CMD_CHG_EMULMODE *103*
CMD_CHG_LENGTH_OF_USERID *105*

■ SYSTEM MANAGEMENT 107
CMD_STATUS_CHECK *107*
CMD_GET_FP_IMAGE2 ..... *109*
CMD_UPGRADE_FIRMWARE2 ..... *111*
CMD_SET_TIME *113*
CMD_GET_TIME *115*
CMD_CTL_IO *117*
CMD_GET_IMAGE_QUALITY *120*
CMD_CFG_IO *122*

APPENDIX A. RS232 UART WAVEFORM 124

APPENDIX B. SUPPORT INFORMATION 124

1. GENERAL DESCRIPTION

NITGEN FIM modules are independent fingerprint identification devices processing commands through UART interface. They receive command from host, run function and return result of command. Host can control FIM modules and check the result by using command.

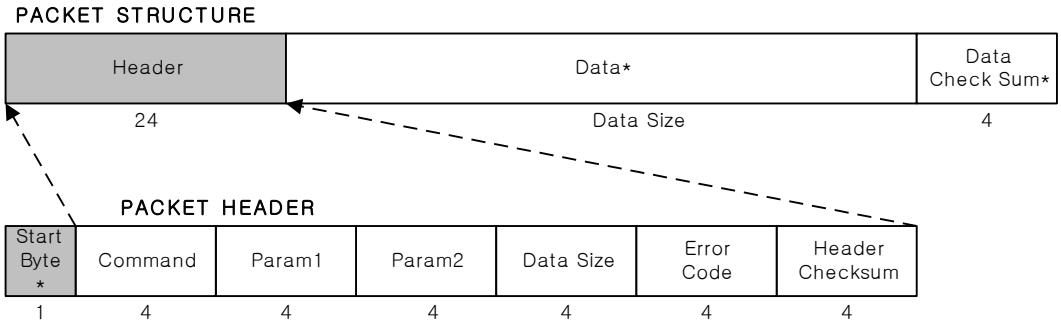
Command is sent and received by packet structure. This packet consists of communication start byte, command code, parameters, data size, error code and auxiliary data. In packet, checksum is added for checking packet arrival.

This document describes packet structure and command code for FIM modules.

2. PACKET DESCRIPTION

■ Packet Structure

The following figure shows the organization of a packet. The packet consists of start byte, header, data (optional), and data checksum (optional). Data and data checksum block is sent only if needed.



* If data size is zero, then data and data check sum is not used.
* Start byte: 0x7E

The packet structure is using big-endian order.
UART interface is configured as 1 start bit, 1 stop bit, 8-bit data and no parity.

Please refer 'Appendix A. RS232 UART waveform' to check sample waveform of CMD_REQUEST_CONNECTION.

The maximum size of a packet is 64Kbyte.

Size (Start Byte) + Size (Header) + Size (Data) + Size (Data Checksum) ≤ 65,536

If data size is so large that the host/device cannot carry data in a single packet, the host/device divides data into small data blocks and sends them over several subsequent packets. And the packet index has the value from 0 to 255. The maximum data size that can be sent is calculated as the following.

Max Data block = 256 x 65,507 = 16,769,792 [byte]

Packet index is transferred by the parameter of the header. The following explains the format of packet index.

(Packet index (0~N) << 8) + (Max Packet Index N)

For example, if single packet is sent, packet index is 0x0000.
If two packets are sent, the first packet index is 0x0001, and the second packet index is 0x0101.
If three packets are sent, packet indexes are sequentially 0x0002, 0x0102, and 0x0202.

Warning: The total data size of multiple packets is dependent on the target devices.
The multi-packet is executed after last packet is transferred.

■ Error Code

If the host sends the command packet, the device returns the acknowledge packet with the packet error code. If Error code is not “ERR_NONE”, the previously sent command packet is ignored in the device. The host needs to check the returned error code, and then retry or does something.

ERROR CODE LIST		
ERR_NONE	The command packet successfully executed	0x0
ERR_CHECKSUM_ERROR	There exists checksum error in header or data block.	0x2
ERR_INVALID_CMD	The command sent to the device is invalid.	0x5

■ How to Make the Header Checksum & the Data Checksum

Checksum data can be calculated by adding all byte data.
For example, in order to create the header checksum, 20 bytes from command to error code should be added.

3 COMMAND DESCRIPTION

■ INITIALIZATION

CMD_REQUEST_CONNECTION

Overview

Description	This command is very easy and simple command to inspect device operation and get number of registered users.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution		

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x01	Command	0x01
Param1	0	Param1	RESULT_SUCCEEDED
Param2	0	Param2	User Count
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x1	0x0	0x0	0x0	0x0	0x1
Acknowledge	0x7E	0x1	0x1	0x0	0x0	0x0	0x2

Example Condition

User Count = 0

Result = Succeeded

CMD_SET_BAUDRATE

Overview

Description	This command configures baud-rate of current UART channel and saves the value. This command is only supported in FIM3x or FIM30 emulation mode of FIM4x/5x products. Otherwise SI_CHANNEL0_BAUDRATE and SI_CHANNEL1_BAUDRATE of CMD_SET_SYSINFO can be used to configure baud-rate.	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	If succeeded, the device returns acknowledge packet and changes baud-rate to new speed. New baud-rate is applied from the next packet.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x02	Command	0x02
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_CANCELED
Param2	Baud-rate (0 ~ 4) 0 – 115,200 bps 1 – 57,600 bps 2 – 38,400 bps 3 – 19,200 bps 4 – 9,600 bps 5 – 14,400 bps	Param2	Baud rate (0 ~ 4) 0 – 115,200 bps 1 – 57,600 bps 2 – 38,400 bps 3 – 19,200 bps 4 – 9,600 bps 5 – 14,,400 bps
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x2	0x0	0x2	0x0	0x0	0x4
Acknowledge	0x7E	0x2	0x1	0x2	0x0	0x0	0x5

Example Condition

Baud-rate = 38,400 bps

Result = Succeeded

CMD_GET_FIRMWARE_VERSION2

Overview

Description	This command requests the current version of firmware of FIM module. The version information is BCD code and has the following format - **Param2 of ACK**: 0x0000ABCD - **Version**: AB.CD	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution		

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x04	Command	0x04
Param1	0	Param1	RESULT_SUCCEEDED RESULT_CANCELED
Param2	0	Param2	Version Information
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x4	0x0	0x0	0x0	0x0	0x4
Acknowledge	0x7E	0x4	0x1	0x123	0x0	0x0	0x29

Example Condition

Version = 1.23

Result = Succeeded

CMD_GET_DEVICE_INFO

Overview

Description	This command returns the type of FIM module. The type information has the following format Param2 of ACK: 0x0000ABCD Device Type: FIM ABCD i.e. FIM4060 = 0x00004060 FIM5060 = 0x00005060 FIM6060 = 0x00006060	
Availability	Products	FIM4x/5x/6x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Legacy devices are described by using different method. Please refer ComProtocol V2.07 or older to check the legacy devices.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x05	Command	0x05
Param1	0	Param1	RESULT_SUCCEEDED RESULT_CANCELED
Param2	0	Param2	Device Type
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x5	0x0	0x0	0x0	0x0	0x5
Acknowledge	0x7E	0x5	0x1	0x5110	0x0	0x0	0x67

Example Condition

Device Type = FIM5110

Result = Succeeded

MATCHING

CMD_VERIFY_FP

Overview

Description	This command is used to verify a user with user's ID Verification means find a user with ID, what we called 1:1 Matching. There are two ways to verify a user: Fingerprint Matching and Password Matching. Different data structures are used for each method. - **Fingerprint Matching**: Only ID should be sent as a data. * ID_INFO structure is used - **Password Matching**: ID and Password should be sent as a data. * ID_INFO and PASSWORD_INFO structures are used Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document. FPID and Password are made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, PFID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, PFID and Password values should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	The size of ID_INFO structure can be changed if emulation mode is turned off. ID length can be read by SI_LENGTH_OF_USER_ID of CMD_GET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x11	Command	0x11
Param1[1)]	0 – FP verification 1 – Password verification	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_INVALID_ID

			RESULT_INVALID_PARAM RESULT_NOT_IN_TIME RESULT_CANCELED RESULT_EXTRACT_FAIL
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Command Param1 = 0) Template Index Number **ELSE** 0 **ELSE** 0
Data Size	**IF** FP verification Size of (a fraction of FPID) **ELSE IF** password Size of (a fraction of FPID + password) **ELSE** 0	Data Size	0
Error Code	X	Error Code	Error Code
Data	**IF** (Param1 == 0) A fraction of FPID **ELSE IF** (Param1 == 1) A fraction of FPID + password **ELSE** -	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x11	0x1	0x0	0x1B	0x0	0x2D
	Data	0x31 32 33 34 00 00 00 00 00 00 00 35 36 37 38 00 00 00 00 00 00 00 00 00 00 00 00				D-Chksum	0x1A4
Acknowledge	0x7E	0x11	0x1	0x0	0x0	0x0	0x12

Example Condition

Matching method: Password Matching

LENGTH_OF_FPID = 11

LENGTH_OF_PASSWD = 16

ID = 1234

Password = 5678

Result = Succeeded

CMD_IDENTIFY_FP

Overview

Description	This command is used to identify a user by fingerprint input. Identification means find a user without ID, what we called 1:N Matching. If device find a user ID having matched fingerprint, it returns the ID. According to the Param1 of the command packet, auxiliary information such as Template Index or User Type can also be returned. If multi-template is registered in certain ID, Template Index shows which template is matched in the ID. (Range: 0 ~ 9) User type means privilege of the registered user: Normal User (0) and Master User (1). FPID is made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, FPID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, FPID value should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution		

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x12	Command	0x12
Param1	0 – User ID request only 1 – User ID and Template Index request 2 – User ID and User Type request	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_IN_TIME RESULT_IDENTIFY_TIMEOUT RESULT_CANCELED RESULT_EXTRACT_FAIL
Param2	0	Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED)

			IF (Command Param1 = 0x00) Size of FPID (various between devices) **ELSE IF** (Command Param1 = 0x01) Size of (FPID + Template Index) **ELSE IF** (Command Param1 = 0x02) Size of (FPID + User Type) **ELSE** 0 **ELSE** 0
Error Code	0	Error Code	Error Code
Data	-	Data	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Command Param1 = 0) FPID **ELSE IF** (Command Param1 = 1) (FPID + Template Index) **ELSE IF** (Command Param1 = 2) (FPID + User Type) **ELSE** 0 **ELSE** 0

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x12	0x1	0x0	0x0	0x0	0x13
Acknowledge	0x7E	0x12	0x1	0x0	0xC	0x0	0x1F
	Data	0x31 32 33 34 00 00 00 00 00 00 00 03				D-Chksum	0xCD

Example Condition

Matching method: User ID and Template Index request

LENGTH_OF_FPID = 11

Template Index = 4th template (0x3)

Matched ID = 1234

Result = Succeeded

CMD_IDENTIFY_RID_FP

Overview

Description	This command is used to identify a user by fingerprint input with limited length of ID. Reduced ID narrows the range of search and provides more fast response time. If device find a user ID having matched fingerprint, it returns the ID. According to the Param1 of the command packet, auxiliary information such as Template Index or User Type can also be returned. If multi-template is registered in certain ID, Template Index shows which template is matched in the ID. (Range: 0 ~ 9) User type means privilege of the registered user: Normal User (0) and Master User (1). FPID is made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, FPID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, FPID value should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20
Caution	Reduced ID should be composed with '*' character (0x2A) at unknown digit to complete ID For instance, if you want to identify with ID starting with "12" and ID required 4 digits, you should compose reduced ID as "12**".	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x13	Command	0x13
Param1	0 – User ID request only 1 – User ID and Template Index request 2 – User ID and User Type request	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_IN_TIME RESULT_IDENTIFY_TIMEOUT RESULT_CANCELED
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)

Data Size	Size of FPID	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Command Param1 == 0x00) Size of FPID **ELSE IF**(Command Param1 == 0x01) Size of (FPID + Template Index) **ELSE IF**(Command Param1 == 0x02) Size of (FPID + User Type) **ELSE** 0 **ELSE** 0
Error Code	0	Error Code	Error Code
Data	A fraction of FPID	Data	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Command Param1 == 0x00) FPID **ELSE IF** (Command Param1 == 0x01) (FPID + Template Index) **ELSE IF** (Command Param1 == 0x02) (FPID + User Type) **ELSE** 0 **ELSE** 0

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x13	0x0	0x0	0xB	0x0	0x1E
	Data	0x31 32 2A 2A 00 00 00 00 00 00 00				D-Chksum	0xB7
Acknowledge	0x7E	0x13	0x1	0x0	0xB	0x0	0x1F
	Data	0x31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xCA

Example Condition

Matching method: User ID request only

LENGTH_OF_FPID = 11

Reduced ID = 12**

Matched ID = 1234

Result = Succeeded

CMD_INSTANT_MATCHING

Overview

Description	This command is used to verify fingerprint with the template data sent from the host. Template data can be obtained by using CMD_GET_TEMPLATE. The template of FIM4x/5x and FIM2x include format information of template in itself as a header. Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Template format of each emulation mode is different. NONE = FIM20 NONE ≠ FIM30 FIM20 ≠ FIM30 Therefore, template data made in FIM30 mode cannot be used to verify in NONE and FIM20 mode Multi Template mode is not working in NONE and FIM20 mode and only Nitgen format is supported in FIM30 mode. Aux Info should be sent in FIM30 mode; however, different kind of value is applied by template mode.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x15	Command	0x15
Param1	**(Template Mode << 8) + Aux Info** **Template Mode** 0 – single template 1 – Multi Template (FIM30 emulation mode only) **Aux Info (FIM30 emulation mode only)** **IF** (Template Mode == 0) 0 – Nitgen template 1 – FDA01 compatible style 2 – ISO 19794-2 template 3 – ANSI 378 template	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_IN_TIME RESULT_INVALID_PARAM RESULT_TOO_LARGE_DATA RESULT_CANCELED RESULT_EXTRACT_FAIL

	ELSE IF (Template Mode == 1) 1~10 – the number of templates (NITGEN Format Only)		
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	0
Data Size	Size of (A fraction of TEMPLATE_INFO)	Data Size	0
Error Code	0	Error Code	Error Code
Data	A fraction of TEMPLATE_INFO	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x15	0x0	0x0	0x194	0x0	0xAA
	Data	404 bytes template data (Header 4 byte + Template 400 bytes)				D-Chksum	-
Acknowledge	0x7E	0x15	0x1	0x0	0x0	0x0	0x16

Example Condition

Emulation mode: NONE

Inserted template format: Nitgen format

Result = Succeeded

CMD_GET_TEMPLATE

Overview

Description	This command returns template data of fingerprint. According to Param1 of command packet, various template formats are supported. - **Default (Nitgen format)**: fixed size. 400bytes (FIM30 mode) or 404bytes (NONE, FIM20 mode). - **FDA01 Format (Old Nitgen format)**: fixed size. 400bytes. This format is only supported in FIM30 mode. - **ISO 19794-2 Format**: variable size. Template data is composed under the ISO 19794-2 standard. - **ANSI 378 Format**: variable size. Template data is composed under the ANSI 378 standard. According to Param2 of command packet, overturned template data is supported. Template format is referring to TEMPLATE_INFO structure. Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Template format of each emulation mode is different. NONE = FIM20 NONE ≠ FIM30 FIM20 ≠ FIM30 Therefore, template data made in FIM30 mode cannot be used to verify in NONE and FIM20 mode	

Packet Composition

COMMAND PACKET		**ACKNOWLEDGEMENT PACKET**	
Command	0x16	Command	0x16
Param1	0 – Default 1 – FDA01 Format (FIM30 emulation mode only) 2 – ISO 19794-2 Format 3 – ANSI 378 Format	Param1	RESULT_SUCCEEDED RESULT_FAILELD RESULT_NOT_IN_TIME RESULT_CANCELED RESULT_EXTRACT_FAIL
Param2	0 – Normal template 1– 180° rotated template	Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Size of (A fraction of TEPLATE_INFO)

			ELSE 0
Error Code	0	Error Code	Error Code
Data	-	Data	**IF** (Param1 == RESULT_SUCCEEDED) A fraction of TEMPLATE_INFO **ELSE** -

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x16	0x0	0x1	0x0	0x0	0x17
Acknowledge	0x7E	0x16	0x1	0x0	0x194	0x0	0xAC
	Data	404 bytes overturned template data (Header 4 byte + Template 400 bytes)				D-Chksum	-

Example Condition

Template Format: Default (Nitgen Format)

Rotation: Overturned (180° rotated)

Result = Succeeded

CMD_CANCEL

Overview

Description	This command is used to stop the currently executing command. After cancellation is done, acknowledge packet of executed command is returned with RESULT_CANCELED. Otherwise, in idle status, acknowledge packet of CMD_CANCEL is returned with RESULT_IDLE_STATUS.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	This command stops all executing commands immediately.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x17	Command	0x17
Param1	0	Param1	RESULT_IDLE_STATUS
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error Code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x12	0x0	0x0	0x0	0x0	0x12
Command	0x7E	0x17	0x0	0x0	0x0	0x0	0x17
Acknowledge	0x7E	0x12	0xD	0x0	0x0	0x0	0x1F

Example Condition

Currently executing command: CMD_IDENTIFY_FP (0x12), ID request only

Result = Canceled

CMD_INSTANT_VERIFY

Overview

Description	This command verifies the template data sent from the host by using ID. Template data can be obtained by using CMD_GET_TEMPLATE. Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document. FPID is made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, FPID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, FPID value should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20
Caution	Template data obtained from FIM30 mode cannot be used.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x18	Command	0x18
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_IN_TIME RESULT_INVALID_PARAM RESULT_INVALID_ID RESULT_TOO_LARGE_DATA RESULT_CANCELED
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) Template Index Number **ELSE** 0
Data Size	Size of (A fraction of FPID + Template)	Data Size	0

Error Code	0	Error Code	Error Code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x18	0x0	0x0	0x19F	0x0	0xAD
	Data	0x31 32 33 34 00 00 00 00 00 00 00 + 404 bytes template data (Header 4 byte + Template 400 bytes)				D-Chksum	-
Acknowledge	0x7E	0x18	0x1	0x0	0x0	0x0	0x19

Example Condition

Emulation mode: NONE

ID: 1234

Inserted template format: Nitgen format

Result = Succeeded

CMD_INSTANT_IDENTIFY

Overview

Description	This command identifies the template data sent from the host by fingerprint input. Template data can be obtained by using CMD_GET_TEMPLATE. Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document. FPID is made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, FPID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, FPID value should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20
Caution	Template data obtained from FIM30 mode cannot be used.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x19	Command	0x19
Param1	0 – User ID request only 1 – User ID and Template Index request 2 – User ID and User Type request	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_IN_TIME RESULT_IDENTIFY_TIMEOUT RESULT_INVALID_PARAM RESULT_TOO_LARGE_DATA RESULT_CANCELED
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	0
Data Size	Size (A fraction of Template)	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Command Param1 = 0) Size of FPID (various between devices)

			ELSE IF (Command Param1 = 1) Size of (FPID + Template Index) **ELSE IF** (Command Param1 = 0x02) Size of (FPID + User Type) **ELSE** 0 **ELSE** 0
Error Code	0	Error Code	Error Code
Data	A fraction of Template	Data	**IF** (Param1 == Succeeded) **IF** (Command Param1 = 0) FPID **ELSE IF** (Command Param1 = 1) (FPID + Template Index) **ELSE IF** (Command Param1 = 0x02) (FPID + User Type) **ELSE** 0 **ELSE** 0

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x19	0x1	0x0	0x194	0x0	0xAF
	Data	404 bytes template data (Header 4 byte + Template 400 bytes)				D-Chksum	-
Acknowledge	0x7E	0x19	0x1	0x0	0xC	0x0	0x26
	Data	0x31 32 33 34 00 00 00 00 00 00 00 01				D-Chksum	0xCB

Example Condition

Matching method: User ID and Template Index request

LENGTH_OF_FPID = 11

Template Index = 2nd template (0x1)

Matched ID = 1234

Result = Succeeded

CMD_AUTO_IDENTIFY

Overview

Description	Auto-identify mode is enabled and disabled by using this command. If finger is placed on the sensor, in auto-identify mode, module returns acknowledge packet of CMD_AUTO_IDENTIFY_RESULT command. This command provides various modes as follow. - **Normal Mode**: Automatically capture and identify when finger is placed on the sensor. Identification result is returned in this mode. - **Finger Checking Mode**: Acknowledge packet is returned immediately without capturing when finger is place on the sensor. - **Get Template Mode**: Template data (Nitgen format) is returned without matching when finger is placed on the sensor. - **Get Template in Failure Mode**: Automatically capture and identify when finger is placed on the sensor. Identification result is returned if authentication is succeeded; otherwise, template data (Nitgen format) is returned.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Module ignores almost commands except CMD_CANCEL, CMD_AUTO_IDENTIFY and CMD_STATUS_CHECK if auto-identify mode is enabled.	

Packet Composition

COMMAND PACKET		**ACKNOWLEDGEMENT PACKET**[1)]	
Command	0x1A	Command	0x1A
Param1	Selection of Operation 0 – Stop Auto Identification 1 – Start as Normal Mode 2 – Start as Finger Checking Mode 3 – Start as Get Template Mode 4 – Start as Get Template in Failure Mode	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error Code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x1A	0x1	0x0	0x0	0x0	0x1B
Acknowledge	0x7E	0x1A	0x1	0x0	0x0	0x0	0x1B
Acknowledge	0x7E	0x1B	0x1	0x2	0xB	0x0	0x29
	Data	0x31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xCA
Command	0x7E	0x1A	0x0	0x0	0x0	0x0	0x1A
Acknowledge	0x7E	0x1A	0x1	0x0	0x0	0x0	0x1B

Example Condition

Auto-identify mode: Start as Normal Mode → Auto-identify result (0x1B) → Stop Auto Identification

LENGTH_OF_FPID = 11

Template Index = 3rd template (0x2)

Matched ID = 1234

Result = Succeeded

CMD_AUTO_IDENTIFY_RESULT

Overview

Description	This command is used for only acknowledge packet. In auto-identify mode, module returns the identification result through this command. FPID is made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, FPID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, FPID value should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Only acknowledge packet is available.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command		Command	0x1B
Param1		Param1	RESULT_SUCCEEDED RESULT_FAILED
Param2		Param2	**IF** (Param1 == RESULT_SUCCEEDED) Template Index Number **ELSE** 0
Data Size		Data Size	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Normal Mode) Size of FPID (various between devices) **ELSE IF** (Get Template Mode) Size of (A fraction of TEPLATE_INFO) **ELSE** **IF** (Get Template Mode in Failure) Size of (A fraction of TEPLATE_INFO) **ELSE** 0

Error Code		Error Code	Error Code
Data		Data	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Normal Mode) (various between devices) **ELSE IF** (Get Template Mode) (A fraction of TEPLATE_INFO) **ELSE** **IF** (Get Template Mode in Failure) (A fraction of TEPLATE_INFO) **ELSE** 0

Example

Please refer the example of CMD_AUTO_IDENTIFY (0x1A)

DATABASE MANAGEMENT

CMD_DELETE_FP

Overview

Description	This command is used to delete a user. If user is deleted successfully, currently remained user count is returned in Param2 of acknowledge packet.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x22	Command	0x22
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA REAULT_INVALID_ID RESULT_NOT_MASTER_MODE
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) Registered User Count **ELSE** 0
Data Size	Size (A fraction of FPID)	Data Size	0
Error Code	0	Error Code	Error Code
Data	A fraction of FPID	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x22	0x0	0x0	0xB	0x0	0x2D
	Data	0x31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xCA
Acknowledge	0x7E	0x22	0x1	0x2	0x0	0x0	0x25

Example Condition

LENGTH_OF_FPID = 11

Deleted ID = 1234

Remained User Count = 2

Result = Succeeded

CMD_DELETE_ALL_FP

Overview

Description	This command is used to delete whole user data at once. This command provides various modes as follow. - **Delete all users**: Delete whole user data. - **Delete all users except masters**: Delete whole normal-privilege user data. - **Delete all masters**: Delete whole master-privilege user data. - **Format FP area**: Format the memory area for storing user data.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x23	Command	0x23
Param1	0 – Delete all users 1 – Delete all normal users 2 – Delete all master users 3 – Format FP Area	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE RESULT_CANCELED
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error Code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x23	0x0	0x0	0x0	0x0	0x23
Acknowledge	0x7E	0x23	0x1	0x0	0x0	0x0	0x24

Example Condition

Deletion mode = Delete all users

Result = Succeeded

CMD_SET_MASTER

Overview

Description	This command changes the privilege of a user. In order to set normal-privilege, master-privilege user ID should be sent. Reversely, normal-privilege user ID should be sent to set master-privilege. Master count is returned if succeeded.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x24	Command	0x24
Param1	0 – Clear Master Flag 1 – Set Master Flag	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_INVALID_PARAM REAULT_INVALID_ID RESULT_NOT_MASTER_MODE RESULT_EXCEEDED_MASTER_CNT
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	Master Count
Data Size	Size of (A fraction of FPID)	Data Size	0
Error Code	0	Error Code	Error Code
Data	A fraction of FPID	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x24	0x1	0x0	0xB	0x0	0x30
	Data	0x31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xCA
Acknowledge	0x7E	0x24	0x1	0x2	0x0	0x0	0x27

Example Condition

Mode = Set Master Flag

LENGTH_OF_FPID = 11

Modified ID = 1234

Registered Master Count = 2

Result = Succeeded

CMD_LEAVE_MASTER_MODE

Overview

Description	FIM module has two operating mode: normal privilege mode and master privilege mode. Host is able to request authentication commands such as verification, identification and so on in normal privilege mode; however, master privilege mode is essentially required to request certain commands such as changing system option and managing users. This command is used to leave master mode. In order to enter master mode, CMD_ENTER_MASTER_MODE2 should be used.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		**ACKNOWLEDGEMENT PACKET**	
Command	0x26	Command	0x26
Param1	0	Param1	RESULT_SUCCEEDED RESULT_NOT_MASTER_MODE
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x26	0x0	0x0	0x0	0x0	0x26
Acknowledge	0x7E	0x26	0x1	0x0	0x0	0x0	0x27

Example Condition

Current privilege mode = Master privilege mode

Changed privilege mode = Normal privilege mode

Result = Succeeded

CMD_SET_MASTER_PASSWORD

Overview

Description	FIM module provides a board password. Board password can be another method to enter master privilege mode. This command configures board password. Board password is empty in default system.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. If board password is forgotten and master-privilege user is not registered in the module, there is no way to enter master mode. Board password should be managed carefully.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x27	Command	0x27
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_MASTER_MODE
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	0
Data Size	Size of (A fraction of Password)	Data Size	0
Error Code	0	Error Code	Error code
Data	A fraction of Password	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x27	0x0	0x0	0x10	0x0	0x37
	Data	0x35 36 37 38 00 00 00 00 00 00 00 00 00 00 00 00				D-Chksum	0xDA
Acknowledge	0x7E	0x27	0x1	0x0	0x0	0x0	0x28

Example Condition

LENGTH_OF_PASSWD = 16

Board password = 5678

Result = Succeeded

CMD_READ_USER_DATA

Overview

Description	FIM module provides a custom data block in non-volatile memory, 64Kbytes. Host is able to read/write any data in this area. This command is used to read data in custom data block. In order to read data from custom data block, address (Param1 of command packet) and data length (Param2 od command packet) should be sent. - **Range of address**: 0 ~ 65535 - **Range of data length**: 1 ~ 65535	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. (Address + data length) should be lower than 65536.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x2B	Command	0x2B
Param1	Address	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	Data length (byte)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) Data length (byte) **ELSE** 0
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Data length (byte) **ELSE** 0
Error Code	0	Error Code	Error code
Data	-	Data	**IF** (Param1 == RESULT_SUCCEEDED) User data **ELSE**

			-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x2B	0x20	0x10	0x0	0x0	0x5B
Acknowledge	0x7E	0x2B	0x1	0x10	0x10	0x0	0x4C
	Data	0x00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF				D-Chksum	0x9E7

Example Condition

Start address = 0x20 (32)

Data length = 0x10 (16)

Data = 0x00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF

Result = Succeeded

CMD_WRITE_USER_DATA

Overview

Description	FIM module provides a custom data block in non-volatile memory, 64Kbytes. Host is able to read/write any data in this area. This command is used to write data in custom data block. In order to write data from custom data block, address (Param1 of command packet) and data length (Param2 od command packet) should be sent. - **Range of address**: 0 ~ 65535 - **Range of data length**: 1 ~ 65535	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. (Address + data length) should be lower than 65536.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x2C	Command	0x2C
Param1	Address	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	Data length (byte)	Param2	0
Data Size	Data length	Data Size	0
Error Code	0	Error Code	Error code
Data	User data	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x2C	0x20	0x10	0x10	0x0	0x6C
	Data	0x00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF				D-Chksum	0x9E7
Acknowledge	0x7E	0x2C	0x1	0x0	0x0	0x0	0x2D

Example Condition

Start address = 0x20 (32)

Data length = 0x10 (16)

Data = 0x00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF

Result = Succeeded

CMD_ERASE_USER_DATA_BLOCK

Overview

Description	FIM module provides a custom data block in non-volatile memory, 64Kbytes. This command is used to delete whole data in custom data block. The data in custom area is initialized to 0xFF	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x2D	Command	0x2D
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x2D	0x0	0x0	0x0	0x0	0x2D
Acknowledge	0x7E	0x2D	0x1	0x0	0x0	0x0	0x2E

Example Condition

Result = Succeeded

CMD_DELETE_MASTER_PASSWORD

Overview

Description	FIM module provides a board password. Board password can be another method to enter master privilege mode. This command deletes board password.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x2E	Command	0x2E
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x2E	0x0	0x0	0x0	0x0	0x2E
Acknowledge	0x7E	0x2E	0x1	0x0	0x0	0x0	0x2F

Example Condition

Result = Succeeded

CMD_ENTER_MASTER_MODE2

Overview

Description	FIM module has two operating mode: normal privilege mode and master privilege mode. Host is able to request authentication commands such as verification, identification and so on in normal privilege mode; however, master privilege mode is essentially required to request certain commands such as changing system option and managing users. This command is used to enter master mode. If one of the following is satisfied, device requires master authentication. - Master-privilege user is existed in the device. - Board password is not null string. Otherwise, host is able to enter master mode without authentication by setting Param1 of command packet to 3. This command provides various methods as follow. - **Master FP verification**: verify a master with ID. - **Master password verification**: verify a master with ID and password. - **Board password verification**: verify a master with device password - **Instant FP verification**: verify a master with ID and template. In order to leave master mode, CMD_LEAVE_MASTER_MODE should be used.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x2F	Command	0x2F
Param1	**Master authentication type** 0 – Master FP verification 1 – Master password verification 2 – Board password verification 3 – Null 4 – Instant FP verification	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_IN_TIME RESULT_INVALID_PARAM RESULT_INVALID_ID RESULT_CANCELED

			RESULT_EXTRACT_FAIL
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	**Master authentication type** 0 – Master FP verification 1 – Master password verification 2 – FDA board password verification 3 – Null 4 – Instant FP verification
Data Size	**IF** (Param1 == 0) Size of (A fraction of FPID) **ELSE IF** (Param1 == 1) Size of (A fraction of FPID + Password) **ELSE IF** (Param1 == 2) Size of (A fraction of password) **ELSE IF** (Param1 == 4) Size of (A fraction of FPID + Template) **ELSE** 0	Data Size	0
Error Code	X	Error Code	Error code
Data	**IF** (Param1 == 0) A fraction of FPID **ELSE IF** (Param1 == 1) A fraction of FPID + Password **ELSE IF** (Param1 == 2) A fraction of Password **ELSE IF** (Param1 == 4) FPID + Template **ELSE** -	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x2F	0x0	0x0	0xB	0x0	0x3A
	Data	0x31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xCA
Acknowledge	0x7E	0x2F	0x1	0x0	0x0	0x0	0x30

Example Condition

Master authentication type = Master FP verification

LENGTH_OF_FPID = 11

Matched ID = 1234

Result = Succeeded

CMD_GET_FP_LIST2

Overview

Description	This command is used to get user list including masters. The user list block may be different according to devices. Please check device information by using CMD_GET_DEVICE_INFO before using this command. This command provides two list types as follow. - **User count & ID list**: FP_LIST_TYPE0 structure is used in this type. This structure consists of the number of user, the size of FPID and user list. - **User count only**: FP_LIST_TYPE1 structure is used in this type. This structure contains the number of user only. Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x30	Command	0x30
Param1	**List data selection** 0 = User count & ID list 1 = User count only	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE RESULT_INVALID_PARAM RESULT_CANCELED
Param2	Packet Index (0~N)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) (Packet Index (0~N) << 8) + (Max Packet Index N) **ELSE** -
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Size of (a fraction of FP list block) **ELSE** 0

Error Code	0	Error Code	Error code
Data	-	Data	A fraction of FP list block

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x30	0x0	0x0	0x0	0x0	0x30
Acknowledge	0x7E	0x30	0x1	0x0	0xF	0x0	0x40
	Data	0x00 01 00 0B 31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xD6

Example Condition

List Type = User count & ID list

User Count = 1 normal user

LENGTH_OF_FPID = 11

Registered ID = 1234

Result = Succeeded

CMD_GET_MASTER_LIST2

Overview

Description	This command is used to get master list only. The master list block may be different according to devices. Please check device information by using CMD_GET_DEVICE_INFO before using this command. This command provides two list types as follow. - **Master count & ID list**: FP_LIST_TYPE0 structure is used in this type. This structure consists of the number of master, the size of FPID and master list. - **Master count only**: FP_LIST_TYPE1 structure is used in this type. This structure contains the number of master only. Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x31	Command	0x31
Param1	**List data selection** 0 = Master count & ID list 1 = Master count only	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE RESULT_INVALID_PARAM RESULT_CANCELED
Param2	Packet index (0~N)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) (Packet Index (0~N) << 8) + (Max Packet Index N) **ELSE** -
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Size of (a fraction of master list block) **ELSE** 0
Error Code	0	Error Code	Error code
Data	-	Data	A fraction of master list block

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x31	0x0	0x0	0x0	0x0	0x31
Acknowledge	0x7E	0x31	0x1	0x0	0x4	0x0	0x40
	Data	0x00 00 00 0B				D-Chksum	0x0B

Example Condition

List Type = Master count & ID list

User Count = 1 normal user, no master user

LENGTH_OF_FPID = 11

Result = Succeeded

CMD_READ_LOG_DATA2

Overview

Description	This command is used to get log data. The log data block may be different according to emulation mode. Please check emulation mode information by using CMD_GET_SYSINFO before using this command. LOG_DATA_BLOCK structure is used in this command. This command provides various modes as follow. - **Param2 previous log**: Log is returned located in certain location. This location is decided by the value of Param2. - **Oldest unread log**: Oldest unread log is returned. - **Last written log**: Last written log is returned. - **All logs**: Whole logs of module are returned. - **From oldest unread to last**: Whole unread logs are returned. Data structures and related information can be found in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. Log is not stored in default system. Log option should be enabled to store log data by using SI_USING_LOG of CMD_SET_SYSINFO.	

Packet Composition

COMMAND PACKET		**ACKNOWLEDGEMENT PACKET**	
Command	0x32	Command	0x32
Param1	Log request mode 0 = Param2 previous log 1 = oldest unread log 2 = last written log 3 = All log 4 = from oldest unread to last	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE RESULT_INVALID_PARAM RESULT_CANCELED
Param2	**IF** (Param1 == 0) Nth log **ELSE IF** (Param1 == 3)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) (Packet Index (0~N) << 8) + (Max Packet Index N)

	Index(0~N) **ELSE IF** (Param1 == 4) Index (0~N) **ELSE** 0		**ELSE** -
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Size of a fraction of Log data block **ELSE** 0
Error Code	0	Error Code	Error code
Data	-	Data	**IF** (Param1 == RESULT_SUCCEEDED) Size of a fraction of Log data block **ELSE** 0

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x32	0x3	0x0	0x0	0x0	0x35
Acknowledge	0x7E	0x32	0x1	0x0	0x24	0x0	0x57
	Data	0x00 01 00 20 + one log for none emulation mode				D-Chksum	-

Example Condition

Read Mode = All logs

Emulation Mode = NONE

Log Count = 1

Log Size = 32(0x20)

Result = Succeeded

CMD_REGISTER_FP

Overview

Description	This command is used to register user data. User data made in this command consists of ID, password (optional) and two templates. In order to control template count for one ID, please use CMD_REGISTER_MULTI_FP. Param1 of command packet is used to decide user's privilege. And Param2 of command packet means the current step of the registration. The duty of each step is described as follow. - **Param2 = 0x00**: 1st step value. Create a new user with specified ID and password. And captures a 1st finger for 1st template. - **Param2 = 0x10**: 1st step value. Create a new user with automatically created ID. The range of ID is 0000(start) to 9999(end). Password is not applied. And captures a 1st finger for 1st template. - **Param2 = 0x01**: 2nd step value. Captures a 1st finger for 2nd template then compare with 1st template. The user data is stored into the module if 1st template and 2nd template is matched then registration is done. However, if template number of each user is configured as 4 templates in FIM20 mode, the user data is not stored and wait 3rd step packet. - **Param2 = 0x02**: 2nd step value. Captures a 1st finger for 2nd template. 1st template and 2nd template can be different in this mode. If template number of each user is configured as 4 templates in FIM20 mode, the user data is not stored and wait 3rd step packet. Otherwise, the user data is stored into the module then registration is done. - **Param2 = 0x03**: 3rd step value. Captures a 2nd finger for 3rd template. - **Param2 = 0x04**: 4th step value. Captures a 2nd finger for 4th template then compare with 3rd template. The user data is stored into the module if 3rd template and 4th template is matched then registration is done. - **Param2 = 0x05**: 4th step value. Captures a 2nd finger for 4th template. 3rd template and 4th template can be different in this mode. Registration is done after capturing. The registration can be done after executing certain Param2 value of command packet such as 0x1, 0x2, 0x4 and 0x5. Up to 4 fingers can be registered by using 'different finger' in Param2 of command packet. FPID and Password are made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, PFID is composed as below.

	- 0x31 32 33 34 35 36 37 38 39 30 00 In other words, PFID and Password values should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. User data is not stored if registration is not completed.	

Packet Composition

COMMAND PACKET		**ACKNOWLEDGEMENT PACKET**	
Command	0x33	Command	0x33
Param1	**Privilege** 0 – User 1 – Master	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_IN_TIME RESULT_INVALID_PARAM RESULT_USED_ID RESULT_DB_IS FULL RESULT_NOT_MASTER_MODE RESULT_ANOTHER_FINGER RESULT_CANCELED RESULT_EXTRACT_FAIL RESULT_INVALID_SEQUENCE
Param2	**Packet Index** 0x00 – Extract 1st Template from sensor with ID and password 0x10 – Extract 1st Template from sensor with auto-generated ID 0x01 – Extract 2nd Template from sensor & Save 0x02 – Extract 2nd Template from sensor & Save with different finger 0x03 – Extract 3rd Template from sensor	Param2	**IF** (Param1 == RESULT_SUCCEEDED) && (((Packet Index == 0x01 or 0x02) && (2 templates mode)) || ((Packet Index == 0x04 or 0x05) && (4 templates mode))) Registered User Count **ELSE** 0

	(FIM20 emulation mode Only) 0x04 – Extract 4th Template form sensor & save (FIM20 emulation mode Only) 0x05 – Extract 4th Template from sensor & save with different finger (FIM20 emulation mode Only)		
Data Size	**IF** (Packet index == 0) Size of (FPID + Password) **ELSE** 0	Data Size	0
Error Code	0	Error Code	Error Code
Data	**IF** (Packet index == 0) FPID + password **ELSE** 0	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x33	0x1	0x0	0x1B	0x0	0x4F
	Data	0x31 32 33 34 00 00 00 00 00 00 00 35 36 37 38 00 00 00 00 00 00 00 00 00 00 00 00				D-Chksum	0x1A4
Acknowledge	0x7E	0x33	0x1	0x0	0x0	0x0	0x34
Command	0x7E	0x33	0x1	0x1	0x0	0x0	0x35
Acknowledge	0x7E	0x33	0x1	0x2	0x0	0x0	0x36

Example Condition

Privilege = Master

Registration Type = Using specified ID and Password

Template Number = 2

LENGTH_OF_FPID = 11

LENGTH_OF_PASSWD = 16

Registered ID = 1234

Registered Password = 5678

Total User Count after Registration = 2

Result = Succeeded

CMD_CHANGE_FP

Overview

Description	This command is used to change each user data. Privilege, password and user security level can be modified through this command. The type of modification is described as follow. - **Change Privilege & Save**: The privilege is decided by Param2 of command packet. '0' means normal privilege and '1' means master privilege. - **Change Password & Save**: The password is changed or newly registered by sending ID and password. - **Change User Security Level & Save**: Verify security level is following global value in default system. The global values can be obtained by using SI_VERIFY_SECURITY_LEVEL of CMD_GET_SYSINFO. However, personal verify security level can be applied for certain user. The range of user security level is 1 ~ 9. FPID is made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, FPID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, FPID value should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x34	Command	0x34
Param1	1 – Change Privilege & Save 2 – Change Password & Save 3 – Change User security Level & Save	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_IN_TIME RESULT_INVALID_PARAM RESULT_INVALID_ID

			RESULT_NOT_MASTER_MODE RESULT_CANCELED RESULT_EXTRACT_FAIL
Param2	**IF** (Param1 == 0x01) 0 – set to normal privilege 1 – set to master privilege **ELSE IF** (Param1 == 0x03) Security Level **ELSE** 0	Param2	0
Data Size	**IF** (Param1 == 0x01) Size of (FPID) **ELSE IF** (Param1 == 0x02) Size of (FPID + Password) **ELSE IF** (Param1 == 0x03) Size of(FPID) **ELSE** 0	Data Size	0
Error Code	0	Error Code	Error Code
Data	**IF** (Param1 == 0x01) FPID **ELSE IF**(Param1 == 0x02) FPID + password **ELSE IF**(Param1 == 0x03) FPID **ELSE** 0	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x34	0x1	0x1	0xB	0x0	0x41
	Data	0x31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xCA
Acknowledge	0x7E	0x34	0x1	0x0	0x0	0x0	0x35

Example Condition

Modification Mode = Change Privilege & Save (Normal → Master)

LENGTH_OF_FPID = 11

Matched ID = 1234

Result = Succeeded

CMD_ADD_FP

Overview

Description	This command is used to register user DB data sent from host. User DB data is obtained by using CMD_GET_FP. Various kinds of DB structures are used according to emulation mode and template format. The structure is decided by Param1 of command packet and emulation mode of the module described as follow. - **Param1 = 0x01**: This value is used for supporting compatibility with old structure format. For this reason, FIM_OLD_DB_FIM30 in FIM30 mode and FIM_OLD_DB2 in NONE/FIM20 mode structures are registered through this value. These structures contain 2 templates. - **Param1 = 0x02**: This value is used for supporting compatibility with old structure format. FIM_OLD_DB4 structure is registered through this value. The module should be configured as 4 template mode in FIM20 emulation mode. This structure contains 4 templates. - **Param1 = 0x11**: FIM_DB_FIM30 in FIM30 mode and FIM_DB2 in NONE/FIM20 mode structures are registered through this value. These structures contain 2 templates and various formats are supported such as Nitgen, ISO and ANSI. The template format is classified by DB header. - **Param1 = 0x12**: FIM_DB4 structure in NONE/FIM20 mode is registered through this value. This structure contains 4 templates and various template formats are supported such as Nitgen, ISO and ANSI. The template format is classified by DB header. The module should be configured as 4 template mode in FIM20 emulation mode. - **Param1 = 0x20**: FIM_MT_DB structure in NONE mode is registered though this value. This structure contains multiple templates (1 ~ 10) and various template formats are supported such as Nitgen, ISO and ANSI. The template format is classified by DB header. In case of Nitgen format, the size of structure is fixed. Otherwise, variable size is applied because template size is decided by minutiae information of the finger.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. FIM3x products are not able to accept DB data made in FIM41/51 products.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x35	Command	0x35
Param1	**DB structure version** **For FIM30 emulation mode** 0x01 – FIM_OLD_DB_FIM30 structure 0x11 – FIM_DB_FIM30 structure (NITGEN, ISO 19794-2 and ANSI 378) **For none emulation mode and FIM20 emulation mode** 0x01 – FIM_OLD_DB2 structure 0x02 – FIM_OLD_DB4 structure 0x11 – FIM_DB2 structure (NITGEN, ISO 19794-2 and ANSI 378) 0x12 – FIM_DB4 structure (NITGEN, ISO 19794-2 and ANSI 378) **For none emulation mode only** 0x20 – FIM_MT_DB structure (NITGEN, ISO 19794-2 and ANSI 378)	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_INVALID_PARAM RESULT_USED_ID RESULT_DB_IS_FULL RESULT_NOT_MASTER_MODE RESULT_CANCELED RESULT_WRONG_TEMP_MODE
Param2	(Packet index (0~N) << 8) + (Max Packet Index N)	Param2	0
Data Size	Size of (a fraction of DB structure)	Data Size	0
Error Code	0	Error Code	Error Code
Data	A fraction of DB structure	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x35	0x20	0x0	0x1D8	0x0	0x12E
	Data	FIM_MT_DB structure data (472 bytes)				D-Chksum	-
Acknowledge	0x7E	0x35	0x1	0x0	0x0	0x0	0x36

Example Condition

Emulation Mode = NONE

DB Structure Version = FIM_MT_DB

Number of Template in the DB = 1

Template Format = Nitgen format

LENGTH_OF_FPID = 11

Result = Succeeded

CMD_GET_FP

Overview

Description	This command is used to obtain user DB data from the module. User DB data is uploaded by using CMD_ADD_FP. Various kinds of DB structures are used according to emulation mode and template format. The structure is decided by Param1 of command packet and emulation mode of the module described as follow. - **Param2 = 0x01**: This value is used for supporting compatibility with old structure format. For this reason, FIM_OLD_DB_FIM30 in FIM30 mode and FIM_OLD_DB2 in NONE/FIM20 mode structures are obtained through this value. These structures contain 2 templates. - **Param2 = 0x02**: This value is used for supporting compatibility with old structure format. FIM_OLD_DB4 structure is obtained through this value. This structure contains 4 templates. - **Param2 = 0x11, 0x13 and 0x15**: FIM_DB_FIM30 in FIM30 mode and FIM_DB2 in NONE/FIM20 mode structures are obtained through this value. These structures contain 2 templates and various formats are supported such as Nitgen, ISO and ANSI. - **Param2 = 0x12, 0x14 and 0x16**: FIM_DB4 structure in NONE/FIM20 mode is obtained through this value. This structure contains 4 templates and various template formats are supported such as Nitgen, ISO and ANSI. - **Param2 = 0x20, 0x21 and 0x22**: FIM_MT_DB structure in NONE mode is obtained though this value. This structure contains multiple templates (1 ~ 10) and various template formats are supported such as Nitgen, ISO and ANSI. In case of Nitgen format, the size of structure is fixed. Otherwise, variable size is applied because template size is decided by minutiae information of the finger. There are two methods to get user DB data as follow. - **Param1 = 0**: Get DB with ID. Module returns the matched DB to ID. - **Param1 = 1**: Get DB with sequential order. '1' is used in param1 of command packet to get first DB. Then, '2' should be used to get whole DB until error is returned.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x36	Command	0x36
Param1	**Get operation** 0 – FPID DB 1 – First DB 2 – Next DB	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_INVALID_PARAM RESULT_INVALID_ID RESULT_NOT_MASTER_MODE RESULT_CANCELED
Param2	**DB structure version** **For FIM30 Emulation mode** 0x01 – FIM_OLD_DB_FIM30 Structure 0x11 – FIM_DB_FIM30 structure (Template: NITGEN format) 0x13 – FIM_DB_FIM30 structure (Template: ISO 19794-2 format) 0x15 – FIM_DB_FIM30 structure (Template: ANSI 378 format) **For none emulation mode and** **FIM20 Emulation mode** 0x01 – FIM_OLD_DB2 structure 0x02 – FIM_OLD_DB4 structure 0x11 – FIM_DB2 structure (Template:-NITGEN format) 0x12 – FIM_DB4 structure (Template:-NITGEN format) 0x13 – FIM_DB2 structure (Template: ISO 19794-2 format) 0x14 – FIM_DB4 structure (Template: ISO 19794-2 format) 0x15 – FIM_DB2 structure	Param2	0

	(Template: ANSI 378 format) 0x16 – FIM_DB4 structure (Template: ANSI 378 format) **For none emulation mode only** 0x20 – FIM_MT_DB structure (Template: NIGEN format) 0x21 – FIM_MT_DB structure (Template: ISO 19794-2 format) 0x22 – FIM_MT_DB structure (Template: ANSI 378 format)		
Data Size	**IF** (Param1 == 0) Size of FPID **ELSE** 0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Size of DB structure **ELSE** 0
Error Code	0	Error Code	Error Code
Data	**IF** (Param1 == 0) FPID **ELSE** -	Data	**IF** (Param1 == RESULT_SUCCEEDED) DB structure **ELSE** 0

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x36	0x0	0x20	0xB	0x0	0x61
	Data	0x31 32 33 34 00 00 00 00 00 00 00				D-Chksum	0xCA
Acknowledge	0x7E	0x36	0x1	0x0	0x1D8	0x0	0x110
	Data	FIM_MT_DB structure data (472 bytes)				D-Chksum	-

Example Condition

Get Operation = FPID DB

Emulation Mode = NONE

LENGTH_OF_FPID = 11

FPID = 1234

DB Structure Version = FIM_MT_DB

Number of Template in the DB = 1

Template Format = Nitgen format

Result = Succeeded

CMD_DELETE_ALL_LOG

Overview

Description	This command deletes all logs.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. Deleted logs cannot be recovered.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x37	Command	0x37
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE RESULT_CANCELED
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error Code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x37	0x0	0x0	0x0	0x0	0x37
Acknowledge	0x7E	0x37	0x1	0x0	0x0	0x0	0x38

Example Condition

Result = Succeeded

CMD_REGISTER_MULTI_FP

Overview

Description

This command is used to register user data.

User data made in this command consists of ID, password (optional) and multiple templates.

Up to 10 templates can be stored in one ID.

Param1 of command packet is used to decide user's privilege.

And Param2 of command packet means the current step of the registration with finger index.

The value of Param2 consists of Finger Index and Capture Mode as below.

Param2 of command packet							
31 28	27 24	23 20	19 16	15 12	11 8	7 4	3 0
-	-	-	-	-	-	FP Index	C. Mode

- **Finger Index**: The range of this value is'0' to '9'. This value should be used from '0' sequentially.
- **Capture Mode**: Capture Mode consists of 5 modes as follow.
  * **Capture Mode = 0**: 1st step value. Create a new user with specified ID and password when Finger Index = 0. Otherwise, this value is used to continue the registration of additional finger. And captures a finger for 1st template
  * **Capture Mode = 1**: 1st step value. Create a new user with automatically created ID when Finger Index = 0. Otherwise, this value is used to continue the registration of additional finger. The range of ID is 0000(start) to 9999(end). Password is not applied. And captures a finger for 1st template.
  * **Capture Mode = 2**: 2nd step value. Captures a finger for 2nd template then compare with 1st template. If both templates are matched, the user data is not stored and wait next packet.
  * **Capture Mode = 3**: 2nd step value. Captures a finger for 2nd template then compare with 1st template. If both templates are matched, the user data is stored into the module then registration is done.
  * **Capture Mode = 4**: 3rd step value. Stores the user data into the module then registration is done if host sent a packet with 'Capture Mode = 2' before.

The registration can be done after executing certain Capture Mode value of command packet such as '3' and '4'.

FPID and Password are made with NULL byte in a last position.

For example, firstly, in case of ID = 1234, FPID is composed as below.

- 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care)

Secondly, in case of ID = 1234567890, PFID is composed as below.

	- 0x31 32 33 34 35 36 37 38 39 30 00 In other words, PFID and Password values should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE
Caution	Master privilege is essentially required to perform this operation. User data is not stored if registration is not completed.	

Packet Composition

COMMAND PACKET		**ACKNOWLEDGEMENT PACKET**	
Command	0x38	Command	0x38
Param1	**Privilege** 0 – User 1 – Master	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_NOT_IN_TIME RESULT_INVALID_PARAM RESULT_USED_ID RESULT_DB_IS FULL RESULT_NOT_MASTER_MODE RESULT_ANOTHER_FINGER RESULT_CANCELED RESULT_EXTRACT_FAIL RESULT_INVALID_SEQUENCE
Param2	**Packet Index** bit[31..8] = Not Used bit[7..4] = Finger Index bit[3..0] = Capture Mode	Param2	**IF(**(Param1 == RESULT_SUCCEEDED) && (Capture Mode == 0x03 or 0x04)) Registered User count **ELSE** 0
Data Size	**IF** ((Finger Index == 0) && (Capture Mode == 0)) Size of (FPID + Password) **ELSE** 0	Data Size	0

Error Code	0	Error Code	Error Code
Data	**IF** ((Finger Index == 0) && (Capture Mode == 0)) FPID + password **ELSE** 0	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x38	0x1	0x0	0x1B	0x0	0x54
	Data	0x31 32 33 34 00 00 00 00 00 00 00 35 36 37 38 00 00 00 00 00 00 00 00 00 00 00 00				D-Chksum	0x1A4
Acknowledge	0x7E	0x38	0x1	0x0	0x0	0x0	0x39
Command	0x7E	0x38	0x1	0x2	0x0	0x0	0x3B
Acknowledge	0x7E	0x38	0x1	0x0	0x0	0x0	0x39
Command	0x7E	0x38	0x1	0x10	0x0	0x0	0x49
Acknowledge	0x7E	0x38	0x1	0x0	0x0	0x0	0x39
Command	0x7E	0x38	0x1	0x13	0x0	0x0	0x4C
Acknowledge	0x7E	0x38	0x1	0x2	0x0	0x0	0x3B

Example Condition

Privilege = Master

Registration Type = Using specified ID and Password

LENGTH_OF_FPID = 11

LENGTH_OF_PASSWD = 16

Registered ID = 1234

Registered Password = 5678

Registered Template Count = 2

Total User Count after Registration = 2

Result = Succeeded

CONFIGURATION

CMD_SET_OPP_OPTION

Overview

Description	This command is used to set fingerprint sensor's setting. The configuration is stored into the non-volatile memory directly. Default values are defined as follow. - Gain = 2 - Brightness = 45 - Contrast = 20	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_IMAGE_BRIGHTNESS, SI_IMAGE_GAIN and SI_IMAGE_CONTRAST of CMD_SET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x40	Command	0x40
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	Option value bit[23..16] = gain bit[15..8] = brightness bit[7..0] = contrast	Param2	Option value bit[23..16] = gain bit[15..8] = brightness bit[7..0] = contrast
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x40	0x0	0x22D14	0x0	0x0	0x83
Acknowledge	0x7E	0x40	0x1	0x22D14	0x0	0x0	0x84

Example Condition

Gain = 2

Brightness = 45

Contrast = 20

Result = Succeeded

CMD_GET_OPP_OPTION

Overview

Description	This command is used to get fingerprint sensor's setting. Default values are defined as follow. - Gain = 2 - Brightness = 45 - Contrast = 20	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_IMAGE_BRIGHTNESS, SI_IMAGE_GAIN and SI_IMAGE_CONTRAST of CMD_GET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x41	Command	0x41
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	**IF** (Param1 == RESULT_SUCCEEDED) Option value bit[23..16] = gain bit[15..8] = brightness bit[7..0] = contrast **ELSE** -
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x41	0x0	0x0	0x0	0x0	0x41
Acknowledge	0x7E	0x41	0x1	0x22D14	0x0	0x0	0x85

Example Condition

Gain = 2

Brightness = 45

Contrast = 20

Result = Succeeded

CMD_SET_SECURITY_LEVEL

Overview

Description	This command is used to set security level. The configuration is stored into non-volatile memory directly. Default values are defined as follow. - Verification Security Level = 5 - Identification Security Level = 8	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_VERIFY_SECURITY_LEVEL and SI_IDENTIFY_SECURITY_LEVEL of CMD_SET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x42	Command	0x42
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	Bit[31..16] = Verification Security Level Bit[15..0] = Identification Security Level	Param2	bit[31..16] = Verification Security Level bit[15..0] = Identification Security Level
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x42	0x0	0x50008	0x0	0x0	0x4F
Acknowledge	0x7E	0x42	0x1	0x50008	0x0	0x0	0x50

Example Condition

Verification Security Level = 5

Identification Security Level = 8

Result = Succeeded

CMD_GET_SECURITY_LEVEL

Overview

Description	This command is used to get security level. Default values are defined as follow. - Verification Security Level = 5 - Identification Security Level = 8	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_VERIFY_SECURITY_LEVEL and SI_IDENTIFY_SECURITY_LEVEL of CMD_GET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x43	Command	0x43
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	**IF** (Param1 == RESULT_SUCCEEDED) bit[31..16] = Verification Security Level bit[15..0] = Identification Security Level **ELSE** -
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x43	0x0	0x0	0x0	0x0	0x43
Acknowledge	0x7E	0x43	0x1	0x50008	0x0	0x0	0x51

Example Condition

Verification Security Level = 5

Identification Security Level = 8

Result = Succeeded

CMD_SET_CAPTURE_OPTION

Overview

Description	This command is used to set capturing option. The configuration is stored into the non-volatile memory directly. Latent and Adaptive Capture functions are configured by this command. - **Latent**: This function is used to prevent false acceptance of latent fingerprints. - **Adaptive Capture**: This function is used to improve the recognition ratio for dry/wet fingerprint. The module automatically judges the captured finger image is good to register/identify or not and modifies internal setting; however, it takes more processing time. This time varies on Adaptive Count. - **Adaptive Count**: This value means the max number of capturing count in Adaptive capture. The range of this value is '0 ~ 255'. Default values are defined as follow. - Latent = FALSE (Disabled). - Adaptive Capture = FALSE (Disabled). - Adaptive Count = 5.	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. If Capture Option is Latent & Adaptive Capture, both values are modified to Param2 of command packet at once. This command is operating same as SI_USING_LATENT, SI_ADAPIVE_CAPTURE and SI_NUM_OF_ADAPTIVE_CAP of CMD_SET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x44	Command	0x44
Param1	**Capture Option** 0 – Latent & Adaptive Capture 1 – Latent 2 – Adaptive Capture 8 – Adaptive Count	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	**IF** (Param1 == 0, 1 or 2) Disable = 0 / Enable = 1 **Else IF** (Param1 == 8) Adaptive Count **ELSE** -	Param2	**IF** (Command Param1 == 0, 1 or 2) Disable = 0 / Enable = 1 **ELSE IF** (Command Param1 == 8) Adaptive Count **ELSE** -
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x44	0x0	0x1	0x0	0x0	0x45
Acknowledge	0x7E	0x44	0x1	0x1	0x0	0x0	0x46

Example Condition

Capture Option = Latent & Adaptive Capture (Disable → Enable)

Result = Succeeded

CMD_GET_CAPTURE_OPTION

Overview

Description	This command is used to get capturing option. Latent and Adaptive Capture functions are obtained by this command. - **Latent**: This function is used to prevent false acceptance of latent fingerprints. - **Adaptive Capture**: This function is used to improve the recognition ratio for dry/wet fingerprint. The module automatically judges the captured finger image is good to register/identify or not and modifies internal setting; however, it takes more processing time. This time varies on Adaptive Count. - **Adaptive Count**: This value means the max number of capturing count in Adaptive capture. The range of this value is '0 ~ 255'. Default values are defined as follow. - Latent = FALSE (Disabled). - Adaptive Capture = FALSE (Disabled). - Adaptive Count = 5.	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_USING_LATENT, SI_ADAPIVE_CAPTURE and SI_NUM_OF_ADAPTIVE_CAP of CMD_GET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x45	Command	0x45
Param1	**Capture Option** 0 – Reserved 1 – Latent 2 – Adaptive Capture 8 – Adaptive Count	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	**IF** (Param1 == RESULT_SUCCEEDED) **IF** (Command Param1 == 0, 1 or 2) Disable = 0 / Enable = 1 **ELSE IF** (Command Param1 == 8) Adaptive Count **ELSE** - **ELSE** -
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x45	0x8	0x0	0x0	0x0	0x4D
Acknowledge	0x7E	0x45	0x1	0x5	0x0	0x0	0x4B

Example Condition

Capture Option = Adaptive Count

Adaptive Count = 5

Result = Succeeded

CMD_SET_LOG_OPTION

Overview

Description	This command is used to set log option. The configuration is stored into the non-volatile memory directly. Default value is FALSE (Disabled)	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_USING_LOG of CMD_SET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x48	Command	0x48
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	Disable = 0 / Enable = 1	Param2	Disable = 0 / Enable = 1
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x48	0x0	0x1	0x0	0x0	0x49
Acknowledge	0x7E	0x48	0x1	0x1	0x0	0x0	0x4A

Example Condition

Log Option = Enable

Result = Succeeded

CMD_GET_LOG_OPTION

Overview

Description	This command is used to set log option. Default value is FALSE (Disabled)	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_USING_LOG of CMD_GET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x49	Command	0x49
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	**IF** (Param1 == RESULT_SUCCEEDED) Disable = 0 / Enable = 1 **ELSE** -
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x49	0x0	0x0	0x0	0x0	0x49
Acknowledge	0x7E	0x49	0x1	0x1	0x0	0x0	0x4B

Example Condition

Log Option = Enable

Result = Succeeded

CMD_SET_CAPTURE_PERIOD

Overview

Description	This command is used to set capture timeout. The configuration is stored into the non-volatile memory directly. The module tries to capture until tick count reaches to a capture timeout. The range of this value is '1 ~ 255'. Default value is '5'. It means 5 seconds; 1 second tick count is used.	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_CAPTURE_TIMEOUT of CMD_SET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x4A	Command	0x4A
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	Capture Timeout	Param2	Capture Timeout
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x4A	0x0	0x5	0x0	0x0	0x4F
Acknowledge	0x7E	0x4A	0x1	0x5	0x0	0x0	0x50

Example Condition

Capture Timeout = 5 seconds

Result = Succeeded

CMD_GET_CAPTURE_PERIOD

Overview

Description	This command is used to get capture timeout. The module tries to capture until tick count reaches to a capture timeout. The range of this value is '1 ~ 255'. Default value is '5'. It means 5 seconds; 1 second tick count is used.	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM30
Caution	Master privilege is essentially required to perform this operation. This command is operating same as SI_CAPTURE_TIMEOUT of CMD_GET_SYSINFO.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x4B	Command	0x4B
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	**IF** (Param1 == RESULT_SUCCEEDED) Capture period (1 ~ 255) **ELSE** -
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x4B	0x0	0x0	0x0	0x0	0x4B
Acknowledge	0x7E	0x4B	0x1	0x5	0x0	0x0	0x51

Example Condition

Capture Timeout = 5 seconds

Result = Succeeded

CMD_SET_SYSINFO

Overview

Description	This command is used to set option values of the module. Almost all options are changed by using this command; however, changed values are not stored into the non-volatile memory. Therefore, not saved options will be disappeared if module is power-off without CMD_SAVE_SYSINFO. SI_Type, SI_Value and SI_INFO are described in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. CMD_SAVE_SYSINFO should be executed to preserve changed options.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x4C	Command	0x4C
Param1	SI_Type	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE RESULT_DB_ISNOT_EMPTY
Param2	SI_Value	Param2	0
Data Size	0	Data Size	**IF** (Param1 == Succeeded) Size of (SI_INFO) **ELSE** 0
Error Code	0	Error Code	Error code
Data	-	Data	**IF** (Param1 == Succeeded) SI_INFO **ELSE** -

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x4C	0x20	0x2D	0x0	0x0	0x99
Acknowledge	0x7E	0x4C	0x1	0x0	0x8	0x0	0x55
	Data	0x00 00 00 20 00 00 00 2D				D-Chksum	0x4D

Example Condition

SI_Type = SI_IMAGE_BRIGHTNESS

SI_Value = 45

Result = Succeeded

CMD_GET_SYSINFO

Overview

Description	This command is used to get option values of the module. Almost all options are obtained by using this command. SI_Type, SI_Value and SI_INFO are described in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x4D	Command	0x4D
Param1	SI_Type	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_NOT_MASTER_MODE
Param2	0	Param2	SI_Value
Data Size	0	Data Size	**IF** (Param1 == Succeeded) Size of (SI_INFO) **ELSE** 0
Error Code	0	Error Code	Error code
Data	-	Data	**IF** (Param1 == Succeeded) SI_INFO **ELSE** -

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x4D	0x30	0x0	0x0	0x0	0x7D
Acknowledge	0x7E	0x4D	0x1	0x5	0x8	0x0	0x5B
	Data	0x00 00 00 30 00 00 00 05				D-Chksum	0x35

Example Condition

SI_Type = SI_VERIFY_SECURITY_LEVEL

SI_Value = 5

Result = Succeeded

CMD_SAVE_SYSINFO

Overview

Description	This command is used to save option values of the module. The options changed by CMD_SET_SYSINFO are stored into the non-volatile memory through this command. Stored values will be recovered when boot-up.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. This command should be executed to preserve changed options.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x4E	Command	0x4E
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x4E	0x0	0x0	0x0	0x0	0x4E
Acknowledge	0x7E	0x4E	0x1	0x0	0x0	0x0	0x4F

Example Condition

Result = Succeeded

CMD_CHG_NUM_OF_TEMP

Overview

Description	This command is used to change template mode in FIM20 emulation mode. The configuration is stored into the non-volatile memory directly to prevent conflict of DB. The range of this value is '2' or '4'	
Availability	Products	FIM4x/5x
	Emulation Mode	FIM20
Caution	Master privilege is essentially required to perform this operation. This command should be executed when user is not registered. Otherwise, error is returned.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x4F	Command	0x4F
Param1	Number of Template (2 or 4)	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE RESULT_INVALID_PARAM RESULT_DB_ISNOT_EMPTY
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x4F	0x2	0x0	0x0	0x0	0x51
Acknowledge	0x7E	0x4F	0x1	0x0	0x0	0x0	0x50

Example Condition

Number of Template = 2

Result = Succeeded

CMD_SET_DEFAULT_SYSINFO

Overview

Description	This command changes all options listed in SI_Type to default value except followings → SI_NUM_OF_TEMP, SI_CHANNEL0_BAUDRATE and SI_CHANNEL1_BAUDRATE. Also, programmable GPIO configuration in NONE emulation mode is changed to all disabled (0xFF). Default values are described in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. This command should be executed to preserve changed options.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x50	Command	0x50
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x50	0x0	0x0	0x0	0x0	0x50
Acknowledge	0x7E	0x50	0x1	0x0	0x0	0x0	0x51

Example Condition

Result = Succeeded

CMD_CHG_EMULMODE

Overview

Description	This command changes emulation mode of the module. Length of user ID (SI_LENGTH_OF_USER_ID) can also be changed because fixed ID length is used in FIM20/FIM30 mode. The configuration is stored into the non-volatile memory directly to prevent conflict of DB. Default value is NONE mode.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Master privilege is essentially required to perform this operation. This command should be executed when user is not registered. Otherwise, error is returned.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x51	Command	0x51
Param1	0x01 – FIM20 emulation mode 0x02 – FIM30 emulation mode 0xFF – None emulation mode	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	0x01 – FIM20 emulation mode 0x02 – FIM30 emulation mode 0xFF – None emulation mode
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x51	0x1	0x0	0x0	0x0	0x52
Acknowledge	0x7E	0x51	0x1	0x1	0x0	0x0	0x53

Example Condition

Emulation Mode = FIM20 mode.

Result = Succeeded

CMD_CHG_LENGTH_OF_USERID

Overview

Description	This command changes ID length of the module. The configuration is stored into the non-volatile memory directly to prevent conflict of DB. Default value is '10' and LENGTH_OF_FPID is '11' because FPID is made with NULL byte in a last position. For example, firstly, in case of ID = 1234, FPID is composed as below. - 0x31 32 33 34 00 xx xx xx xx xx xx (xx = Do not care) Secondly, in case of ID = 1234567890, FPID is composed as below. - 0x31 32 33 34 35 36 37 38 39 30 00 In other words, FPID values should be finished with NULL byte.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE
Caution	Master privilege is essentially required to perform this operation. This command should be executed when user is not registered. Otherwise, error is returned. Whole commands using ID as a data such as CMD_VERIFY_FP and CMD_ADD_FP should be modified to accommodate changed ID length.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x52	Command	0x52
Param1	Length of User ID (4~15)	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_MASTER_MODE
Param2	0	Param2	Length of User ID
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x52	0xA	0x0	0x0	0x0	0x5C
Acknowledge	0x7E	0x52	0x1	0xA	0x0	0x0	0x5D

Example Condition

ID Length = 10

Result = Succeeded

SYSTEM MANAGEMENT

CMD_STATUS_CHECK

Overview

Description	This command is used to check the status of the module. The status of acknowledge packet is described as below. - Idle (0): The module is in idle condition. - Busy (1): The module is processing another command. - DB Uploading (3): The module is uploading user DB data is boot-up sequence. - Auto-identify mode (4): The module is in auto-identify mode.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution		

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x62	Command	0x62
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED
Param2	0	Param2	**STATUS** 0 – Idle 1 – Busy 3 – DB Uploading 4 – Auto-identify mode
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x62	0x0	0x0	0x0	0x0	0x62
Acknowledge	0x7E	0x62	0x1	0x4	0x0	0x0	0x67

Example Condition

The status of the module = Auto-identify mode

Result = Succeeded

CMD_GET_FP_IMAGE2

Overview

Description	This command is used to get a fingerprint image as raw format. Image data cannot be transferred at once; multiple packets should be used. The size of each data is different as follow. - **NONE/FIM20 mode**: Data Size = 3720 bytes - **FIM30 mode**: Data Size = 512 bytes	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Total image size depends on the sensor type. Transferred data should be combined sequentially.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x63	Command	0x63
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_NOT_IN_TIME RESULT_CANCELED
Param2	Packet index (0~N)	Param2	**IF** (Param1 == RESULT_SUCCEEDED) (Packet Index (0~N) << 8) + (Max Packet Index N) **ELSE** 0
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Size of (a fraction of image data block) **ELSE** 0
Error Code	0	Error Code	Error code
Data	-	Data	**IF** (Param1 == RESULT_SUCCEEDED) A fraction of image data block **ELSE** -

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x63	0x0	0x0	0x0	0x0	0x63
Acknowledge	0x7E	0x63	0x1	0x14	0xE88	0x0	0x10E
	Data	3720 bytes image data				D-Chksum	-
Command	0x7E	0x63	0x0	0x1	0x0	0x0	0x64
Acknowledge	0x7E	0x63	0x1	0x114	0xE88	0x0	0x10F
	Data	3720 bytes image data				D-Chksum	-
Transmitting continuously							
Command	0x7E	0x63	0x0	0x13	0x0	0x0	0x76
Acknowledge	0x7E	0x63	0x1	0x1314	0xE88	0x0	0x121
	Data	3720 bytes image data				D-Chksum	-
Command	0x7E	0x63	0x0	0x14	0x0	0x0	0x77
Acknowledge	0x7E	0x63	0x1	0x1414	0xE10	0x0	0xAA
	Data	3600 bytes remained image data				D-Chksum	-

Example Condition

Module Name = FIM4060

Image Size = 78,000 bytes (260 x 300)

Result = Succeeded

CMD_UPGRADE_FIRMWARE2

Overview

Description	This command is used to update firmware. Firmware data cannot be transferred at once; multiple packets should be used. The sequence of update is described as follow. - Collects firmware data to volatile memory sent from host until last packet is arrived. - Verifies firmware data - Copies firmware data from volatile memory to non-volatile memory. - In order to complete firmware updating, the module should be rebooted. The maximum size of each data can be defined by host. The range of the value is '1 ~ 32768'; however, we recommends 3072 bytes used in EvTools for each packet because the stability of 3072 bytes is proved by testing for a long time. FIRMWARE_DATA_BLOCK structure is used in this command and the structure is described in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	Transferred data should be combined sequentially. This operation should be processed very carefully. The module will not work if wrong data is written. And this problem cannot be solved by user.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x64	Command	0x64
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_CANCELED
Param2	(Packet Index (0~N) << 8) + (Max Packet Index N)	Param2	Param2 of command packet
Data Size	Size of (a fraction of Firmware data)	Data Size	0
Error Code	0	Error Code	Error code
Data	Firmware data	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x64	0x0	0x97	0xC04	0x0	0x10B
	Data	F/W Size (4 bytes) + F/W Data (3072 bytes)				D-Chksum	-
Acknowledge	0x7E	0x64	0x1	0x97	0x0	0x0	0xFC
Command	0x7E	0x64	0x0	0x197	0xC04	0x0	0x10C
	Data	F/W Size (4 bytes) + F/W Data (3072 bytes)				D-Chksum	-
Acknowledge	0x7E	0x64	0x1	0x197	0x0	0x0	0xFD
Transmitting Continuously							
Command	0x7E	0x64	0x0	0x9697	0xC04	0x0	0x1A1
	Data	F/W Size (4 bytes) + F/W Data (3072 bytes)				D-Chksum	-
Acknowledge	0x7E	0x64	0x1	0x9697	0x0	0x0	0x192
Command	0x7E	0x64	0x0	0x9797	0xC04	0x0	0x1A2
	Data	F/W Size (4 bytes) + F/W Data (3072 bytes)				D-Chksum	-
Acknowledge	0x7E	0x64	0x1	0x9797	0x0	0x0	0x193

Example Condition

Source Firmware = FIM5060(V1_20)_B1.bin

Transferred Data Size = 467,552 bytes (= (4 + 3072) * 152)

Transferred Firmware Size = 466,944 bytes (= 3072 * 152)

F/W Size (Exact firmware size) = 466,172 bytes

Garbage data size = 772 bytes (= Transferred Firmware Size – F/W Size)

Result = Succeeded

* Garbage data is not processed in the firmware.

CMD_SET_TIME

Overview

Description	This command is used to set time to real-time clock (RTC) of the module. All time data is written as BCD code. For instance, 15th November 2011 12(H):34(M):56(S) is written as follow; 0x20 11 11 15 12 34 56 xx (xx: do not care). TIME_INFO structure is used in this command and the structure is described in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	In internal RTC (default) products, time is not preserved in power-off condition. External RTC products should be used to preserve time without power-supply.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x65	Command	0x65
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_DATASIZE RESULT_INVALID_DATA RESULT_CANCELED
Param2	0	Param2	0
Data Size	Size of TIME_INFO	Data Size	0
Error Code	0	Error Code	Error code
Data	TIME_INFO	Data	-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x65	0x0	0x0	0x8	0x0	0x6D
	Data	0x20 11 11 15 12 34 56 00				D-Chksum	0xF3
Acknowledge	0x7E	0x65	0x1	0x0	0x0	0x0	0x66

Example Condition

Source Time = 15th November 2011 12(H):34(M):56(S)

Result = Succeeded

CMD_GET_TIME

Overview

Description	This command is used to get time from real-time clock (RTC) of the module. All time data is written as BCD code. For instance, 15th November 2011 12(H):34(M):56(S) is written as follow; 0x20 11 11 15 12 34 56 xx (xx: do not care). TIME_INFO structure is used in this command and the structure is described in the ComProtocol Vol2. ErrorCode & Structures document.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution	In internal RTC (default) products, time is not preserved in power-off condition. External RTC products should be used to preserve time without power-supply.	

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x66	Command	0x66
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_CANCELED
Param2	0	Param2	0
Data Size	0	Data Size	**IF** (Param1 == RESULT_SUCCEEDED) Size of TIME_INFO **ELSE** 0
Error Code	0	Error Code	Error code
Data	-	Data	**IF** (Param1 == RESULT_SUCCEEDED) TIME_INFO **ELSE** -

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x66	0x0	0x0	0x0	0x0	0x66
Acknowledge	0x7E	0x66	0x1	0x0	0x8	0x0	0x6F
	Data	0x20 11 11 15 12 34 56 00				D-Chksum	0xF3

Example Condition

Source Time = 15th November 2011 12(H):34(M):56(S)

Result = Succeeded

CMD_CTL_IO

Overview

<table>
<tr><td>Description</td><td>
This command is used to control GPIO pins.
Functionalities and number of pins depend on the device and emulation mode described below.
<table>
<tr><th>Products</th><th>Emul-Mode</th><th>Description</th></tr>
<tr><td rowspan="3">FIM4x & FIM50/51/52 & FIM60</td><td>NONE</td><td>Support 8 programmable GPIOs (General Purpose Input/Output)</td></tr>
<tr><td>FIM20</td><td rowspan="2">Support Relay0 on GPIO0, Relay1 on GPIO1. Remained GPIOs are all disabled and cannot be configured</td></tr>
<tr><td>FIM30</td></tr>
<tr><td rowspan="3">FIM5360</td><td>NONE</td><td>Support 5 programmable GPIOs (General Purpose Input/Output)</td></tr>
<tr><td>FIM20</td><td>Support Relay0 on GPIO0, Relay1 on GPIO1. Remained GPIOs are all disabled and cannot be configured</td></tr>
<tr><td>FIM30</td><td>Support GPIOs functionalities same as FIM3x products.
GPIO0 – GPIO_OUT_SUCCESS_HIGH
GPIO1 – GPIO_OUT_FAIL_HIGH
GPIO2 – GPIO_IN_ENROLL_LOW
GPIO3 – GPIO_IN_DELETE_LOW
GPIO4 – GPIO_IN_IDENTIFY_LOW</td></tr>
</table>
First of all, in case of programmable GPIOs, this command is only used for GPIO_IN_NORMAL and GPIO_OUT_NORMAL mode. In order to write a signal level to certain GPIO, the GPIO should be configured as GPIO_OUT_NORMAL. On the other hands, GPIO should be configured as GPIO_IN_NORMAL to read a signal level from the GPIO. Sensor LED of optical sensor is also adjusted irrespective of GPIO configuration.
Secondly, in case of Relay0 and 1, only GPIO0 and 1 are under control. Host is able to read/write signal level of the GPIO0 and 1. Otherwise, remained GPIOs are not working. Sensor LED of optical sensor is also adjusted irrespective of GPIO configuration.
Lastly, 5 GPIOs of FIM5360 have intrinsic functionalities in FIM30 mode. The GPIOs are not controlled by this command.
Please refer the ComProtocol Vol2. ErrorCode & Structures document for the detail of programmable GPIOs.
</td></tr>
<tr><td rowspan="2">Availability</td><td>Products</td><td>FIM4x/5x</td></tr>
<tr><td>Emulation Mode</td><td>NONE/FIM20/FIM30</td></tr>
<tr><td>Caution</td><td colspan="2">Default values of all programmable GPIOs are disabled.</td></tr>
</table>

	Programmable GPIOs are configured by CMD_CFG_IO.

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x67	Command	0x67
Param1	**Control Target** **For programmable GPIOs** 0x00 ~ 0x07 – Selection of GPIOs 0x10 – Sensor LED (Write only) **For Relay 0 and 1 (Write only)** 0x01 – Sensor LED 0x40 – Relay Channel 0 0x41 – Relay Channel 1	Param1	RESULT_SUCCEEDED RESULT_FAILED RESULT_INVALID_PARAM RESULT_CANCELED
Param2	**Control Value** **For programmable GPIOs** (Direction << 8) + (Value) **Direction** 0 – Write 1 – Read **Value** 0 – Low/Off 1 – High/On **For Relay 0 and 1** 0 – Low/Off 1 – High/On	Param2	**For programmable GPIOs** **IF** (Direction == 1) (Direction << 8) + (Status of GPIO) **ELSE** 0 **For relay 0 and 1** 0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x67	0x3	0x10	0x0	0x0	0x7A
Acknowledge	0x7E	0x67	0x1	0x11	0x0	0x0	0x79

Example Condition

Emulation Mode = NONE

Control Target = GPIO3

Direction = Read

Value = Do not care

Status of GPIO = High

Result = Succeeded

CMD_GET_IMAGE_QUALITY

Overview

Description	This command is used to fingerprint quality after using following commands - CMD_VERIFY_FP - CMD_IDENTIFY_FP - CMD_IDENTIFY_RID_FP - CMD_INSTANT_MATCHING - CMD_GET_TEMPLATE - CMD_AUTO_IDENTIFY_RESULT - CMD_ENTER_MASTER_MODE2 - CMD_REGISTER_FP - CMD_REGISTER_MULTI_FP. - CMD_GET_FP_IMAGE2 - GPIO_IN_ENROLL_HIGH of programmable GPIO - GPIO_IN_ENROLL_LOW of programmable GPIO - GPIO_IN_DELETE_HIGH of programmable GPIO - GPIO_IN_DELETE_LOW of programmable GPIO - GPIO_IN_IDENTIFY_HIGH of programmable GPIO - GPIO_IN_IDENTIFY_LOW of programmable GPIO Otherwise, the value of fingerprint quality is invalid. The range of this value is '0 ~ 100'.	
Availability	Products	FIM4x/5x
	Emulation Mode	NONE/FIM20/FIM30
Caution		

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x68	Command	0x68
Param1	0	Param1	RESULT_SUCCEEDED RESULT_FAILED
Param2	0	Param2	Quality Value
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code
	-		-

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x68	0x0	0x0	0x0	0x0	0x68
Acknowledge	0x7E	0x68	0x1	0x54	0x0	0x0	0x12C

Example Condition

Quality Value = 84

Result = Succeeded

CMD_CFG_IO

Overview

<table>
<tr><td>Description</td><td colspan="2">This command is used to configure programmable GPIO pins.
Whole programmable GPIOs are configured to the following values.
<table>
<tr><th>Param2 of command packet</th><th>Function</th></tr>
<tr><td>0x00</td><td>GPIO_IN_NORMAL</td></tr>
<tr><td>0x01</td><td>GPIO_IN_ENROLL_HIGH</td></tr>
<tr><td>0x02</td><td>GPIO_IN_ENROLL_LOW</td></tr>
<tr><td>0x03</td><td>GPIO_IN_DELETE_HIGH</td></tr>
<tr><td>0x04</td><td>GPIO_IN_DELETE_LOW</td></tr>
<tr><td>0x05</td><td>GPIO_IN_IDENTIFY_HIGH</td></tr>
<tr><td>0x06</td><td>GPIO_IN_IDENTIFY_LOW</td></tr>
<tr><td>0x80</td><td>GPIO_OUT_NORMAL</td></tr>
<tr><td>0x81</td><td>GPIO_OUT_SUCCESS_HIGH</td></tr>
<tr><td>0x82</td><td>GPIO_OUT_SUCCESS_LOW</td></tr>
<tr><td>0x83</td><td>GPIO_OUT_FAIL_HIGH</td></tr>
<tr><td>0x84</td><td>GPIO_OUT_FAIL_LOW</td></tr>
<tr><td>0x85</td><td>GPIO_OUT_CAPTURING</td></tr>
<tr><td>0x86</td><td>GPIO_OUT_FP_STATUS</td></tr>
<tr><td>0xFF</td><td>GPIO_DISABLE</td></tr>
</table>
Please refer the ComProtocol Vol2. ErrorCode & Structures document for the detail of programmable GPIOs.
Default values of all programmable GPIOs are disabled.
GPIO_IN_NORMAL and GPIO_OUT_NORMAL are controlled by CMD_CTL_IO.</td></tr>
<tr><td rowspan="2">Availability</td><td>Products</td><td>FIM4x/5x</td></tr>
<tr><td>Emulation Mode</td><td>NONE</td></tr>
<tr><td>Caution</td><td colspan="2">Programmable GPIOs are working under LVCMOS (3.3V) level.
The configuration is initialized by CMD_SET_DEFAULT_SYSINFO.
CMD_SAVE_SYSINFO should be executed to preserve changed options.</td></tr>
</table>

Packet Composition

COMMAND PACKET		ACKNOWLEDGEMENT PACKET	
Command	0x69	Command	0x69
Param1	Selection of GPIO (0~7)	Param1	RESULT_SUCCEEDED RESULT_FAILED
Param2	Value	Param2	0
Data Size	0	Data Size	0
Error Code	0	Error Code	Error code

Example

Packet Type	StartByte	Command	Param1	Param2	DataSize	ErrorCode	H-Chksum
Command	0x7E	0x69	0x4	0x80	0x0	0x0	0xED
Acknowledge	0x7E	0x69	0x1	0x0	0x0	0x0	0x6A

Example Condition

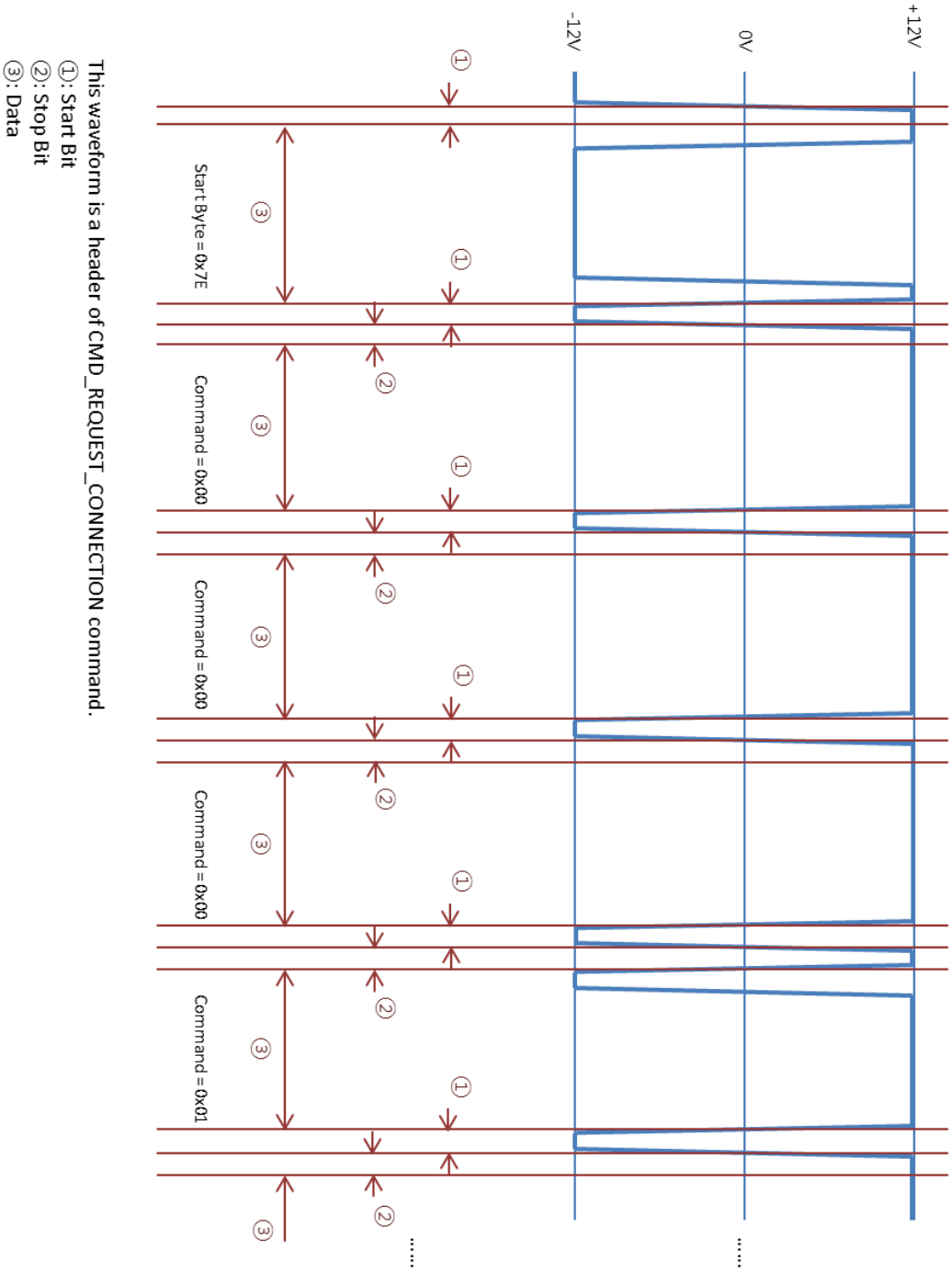
Selection of GPIO = 4

Value = GPIO_OUT_NORMAL

Result = Succeeded

Appendix A. RS232 UART waveform

CMD_REQUEST_CONNECTION packet is composed described below.



Appendix B. Support Information

NITGEN&COMPANY CO., LTD.

6, Wiryeseong-daero, Songpa-gu, Seoul, 138-827

Korea

Tel: +82-2-6488-3232

Fax: +82-2-6488-3096

E-mail: Customer@nitgen.com