SONY

CXD5603GF

User's Manual

ver 0.19

Sony Semiconductor Solutions Corporation
IoT Solutions Business Division

History of Revisions

Revised ID	Details of revisions made	Preparation date	Approval date	Remarks
0.01	Established first edition.	2015/11/13	2015/11/13	Provisional release
0.02	Added @GUSE and @GTIM.	2015/12/2	2015/12/2	
0.03	Added @ABPT, @ABUP, @BUP, @GPOE, @GSP and @GSW.	2015/12/21	2015/12/21	
	Added @GNS.	2016/2/18	2016/2/18	
0.04	Added @GPOS, @GSOP, @SLP and @WUP.	2016/5/20	2016/5/20	
0.05	Added @BUPC, @FER, @GALG, @GALS, @GEMG, @GEMS, @GPPS, @GTCX, @LALG, @LALS, @LEMG, @LEMS and @VER.	2016/10/14	2016/10/14	
0.06	Modified @SLP, @WUP.	2016/11/8	2016/12/1	
	Changed the default value of @GUSE.	2016/11/8	2016/12/1	
	Added @GPTC.	2016/11/10	2016/12/1	
0.07	Corrected some errors	2016/12/12	2016/12/12	
0.08	Added the commands with binary data to command sequence section.	2017/5/9	2017/5/9	
0.09	Added the arguments to @GSOP and @SLP.	2017/6/30	2017/6/30	
	Added the operation status description.	2017/6/30	2017/6/30	
0.10	Corrected some errors.	2017/7/18	2017/7/18	
0.11	Added @CSBR.	2017/8/21	2017/10/12	
	Added the explanation of I^2C interface.	2017/10/12	2017/10/12	
0.12	Added some description of @GPTC.	2017/10/13	2017/11/28	
	Corrected the erroneous description.	2017/11/14	2017/11/28	
	Added Section 6.	2017/11/28	2017/11/28	
0.13	Added the explanation of firmware update.	2017/12/26	2018/2/5	
	Added the explanation of flash-less boot.	2018/1/5	2018/2/5	
0.14	Modified the error description in I^2C interface (removed the description of “configuration data”) .	2018/7/13	2018/8/21	
	Added @GTE, @GTR, @GTS.	2018/8/15	2018/8/21	
	Modified the error description I^2C firmware update.	2018/8/21	2018/8/21	
0.15	Added Galileo, BeiDou and QZSS L1-s to the argument of @GNS.	2018/10/15	2018/12/26	
	Modified NMEA specification for NMEA 4.1 support.	2018/10/25	2018/12/26	
	Add Galileo and BeiDou to NMEA talker ID. Changed	2018/10/25	2018/12/26	

	QZSS talker ID from QZ to GQ.			
	Added @GSUC and delete @GUSE.	2018/10/26	2018/12/26	
	Added the new operation mode to Low power mode (the case of positioning period is under 30s).	2018/12/18	2019/1/29	
0.16	Modified the argument description in @GNS command	2019/2/12	2019/2/12	
	Removed the description of indoor detection of Low power mode.	2019/2/12	2019/2/12	
	The packet table of “Update command sending” in I^2C firmware update was wrong and modified.	2019/2/12	2019/2/12	
0.17	Added the unit of the return value of @GTR command.	2019/3/20	2019/6/10	
	Added the description of the transferred data size limitation for I^2C firmware update.	2019/6/10	2019/6/10	
	Modified the error description of the line code in UART firmware update flash-less boot sequence.	2019/6/14	2019/6/14	
	Added “Driving mode” and removed “reserved” from @GSUC command.	2019/6/14	2019/6/14	
0.18	Modified the error description of “Trail mode” bit allocation in @GSUC.	2019/6/18	2019/6/18	
	Removed the restriction of issuing @GSUC and @GUSE.	2019/6/19	2019/6/19	
0.19	Added @QALG, @QALS, @QEMG, @QEMS.	2019/9/26	2019/9/26	

Table of Contents

1 What this document contains

This document sets forth the CXD5603GF host controller command specifications.

It describes the command protocol used when the CXD5603GF is controlled from the host controller, and the specifications of these commands.

2 Host controller interface specifications

The CXD5603GF is connected to the host controller using UART. This section describes the communication specification.

2.1 UART interface

The settings of the UART interface are listed below.

- Baud rate : 115,200 bps
- Data length : 8 bits
- Parity : None
- Stop bit : 1 bit
- Flow control : None

2.2 I²C interface

The CXD5603GF is operated as I²C slave device. The settings of the I²C interface are listed below.

- I²C clock frequency : 400kHz
- Address length : 7 bits
- Slave address : 0x24

When connecting to I²C, one additional GPIO pin in the CXD5603GF is required to input host controller as “data output request” in addition to SDA and SCL. When output data is available in the CXD5603GF, the CXD5603GF outputs “H” from data output request GPIO. When output data is not available in the CXD5603GF, the CXD5603GF outputs “L” from data output request GPIO. For handling the data from the CXD5603GF, data output request GPIO should be connected to host controller’s interrupt port.

When the host controller updates the firmware via I²C interface, the host controller have to send 256bytes data per packet at least.

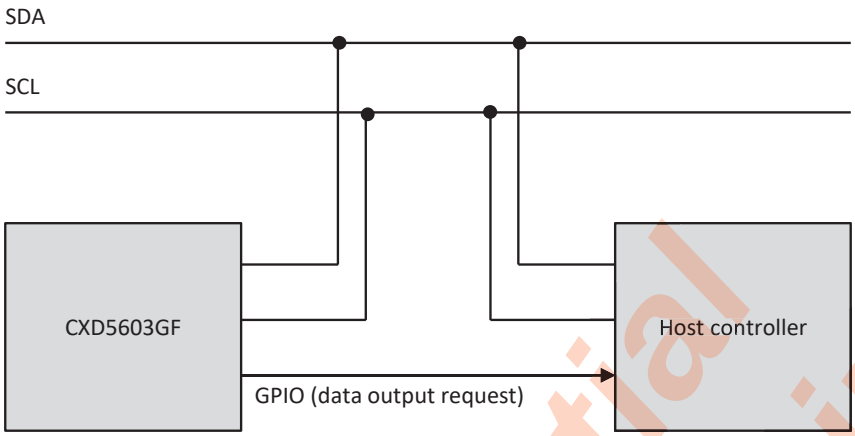


Fig.1 The connection with host controller using I²C

2.2.1 The structure of data from host controller

The data that is sent from host controller is either:

- Command, or
- Data associated with command

They are called “data attribute”.

When sending data with I²C, the entire data is being divided into the packet(s) which size is 28 bytes or less. The host controller must wait over 2ms every time sending one packets. One packet shall include one “data attribute”.

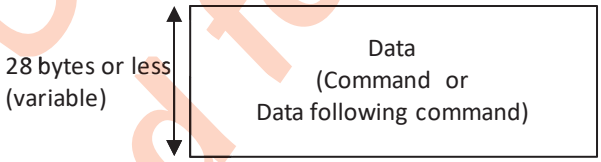


Fig.2 The structure of data from host controller

2.2.2 The structure of data from CXD5603GF

The data that is sent from CXD5603GF is either:

- Command responses
- Data following commands
- NMEA sentences

They are called “data attribute”.

The data length that is sent in one I²C transaction is 74 bytes (fixed value). It includes more than one packet of any attributes. Below is the structure of 74 bytes data from CXD5603GF.

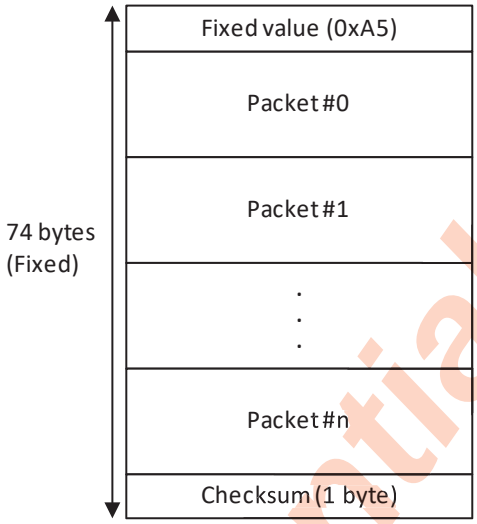


Fig.3 The structure of data from CXD5603GF

Each data chunk is followed by the preamble data (0xA5). The packet type is described at the top of each packet. Data size is determined by the packet type. When the packet type is 0x0F, data size field is inserted to accommodate variable data size.

Table 1. The type of packets

Packet type	Contents	Data size
0x00-0x04	Reserved	
0x0F	Command response, data following commands, NMEA sentence	Variable (described in data size field)

The structures of each packet types are below:

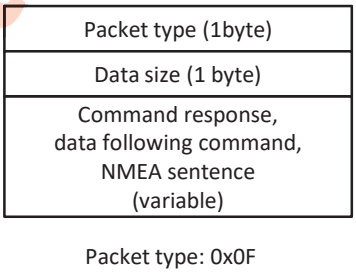


Fig.4 The structure of the packet of each packet type

Any type, any number of packets is included in one transaction data (74 bytes) . Command (specified at “Command specification” section) / data following command (specified at “Command specification” section) /

NMEA sentence (specified at “NMEA sentences” section) are stored in the packet of packet type=0x0F. These data are divided if needed and sent by more than one transaction.

If the total data size which is transmitted from CXD5603GF is less than 74bytes, Dummy data is added to 74 bytes of remaining area.

The host must retrieve data in a timely manner or the data may be lost. Please design the host controller’s system with consideration for enough bandwidth. When host controller receives the sensor data with high frequency, please do the transmission and reception alternately.

Checksum for whole of the data is added at the end of data. Checksum is the lower 1 byte of the 1’s complement of the sum for whole of the data (74 bytes, excluding checksum itself)

3 Command specifications

This section describes the protocol for the commands used between the host controller and CXD5603GF.

3.1 Command format

Commands are described in ASCII code.

The format for the control commands transmitted to the CXD5603GF is given below.

```
@xxx <arg 1> <arg2> … <CR><LF>
```

On receipt of a command from the host controller, the CXD5603GF transmits the command reply message in accordance with the result yielded by executing the command.

When communication is successful: `[xxx] Done<CR><LF>` (where "xxx" is the command name)

When communication has failed: `[xxx] Err n<CR><LF>` (where "xxx" is the command name, and n is the error code)

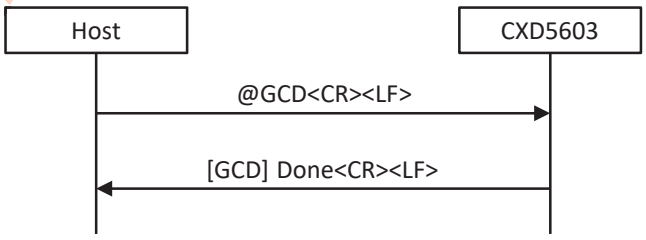
Take steps to prevent another command from being issued before the reply message (Done or Err) indicating the command completion has been returned from the CXD5603GF.

The period between the completion of sending the commands and the sending command responses varies according to commands and situations, but it may reach 5s at the worst case. The host controller should judge as a timeout after a lapse of 5s.

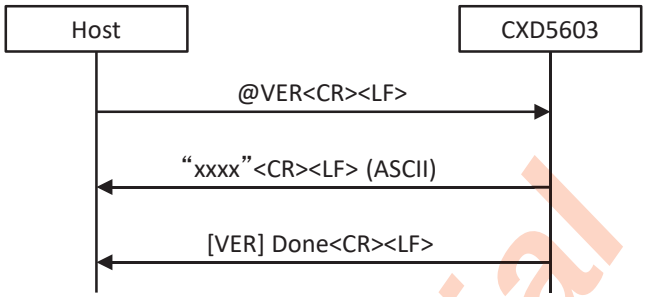
3.2 Command sequence

This section describes the sequence of the commands between the host controller and CXD5603GF for each type of command.

3.2.1 Commands not entailing an exchange of data

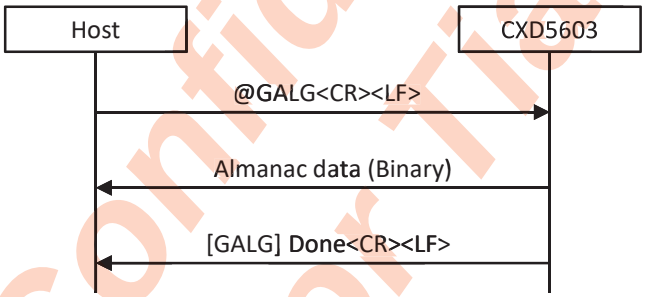


3.2.2 Commands that entail data (ASCII data) transmission from the CXD5603GF



The data to be transmitted is ASCII format data which is terminated using a line break code (<CR><LF>). The data length and data contents differ from one command to another so refer to the command specifications.

3.2.3 Commands that entail data (binary data) transmission from the CXD5603GF



A header and footer that describe the data length, checksum, etc. are inserted into the binary data transmitted from the CXD5603GF before and after the data.

The binary data format is shown below.

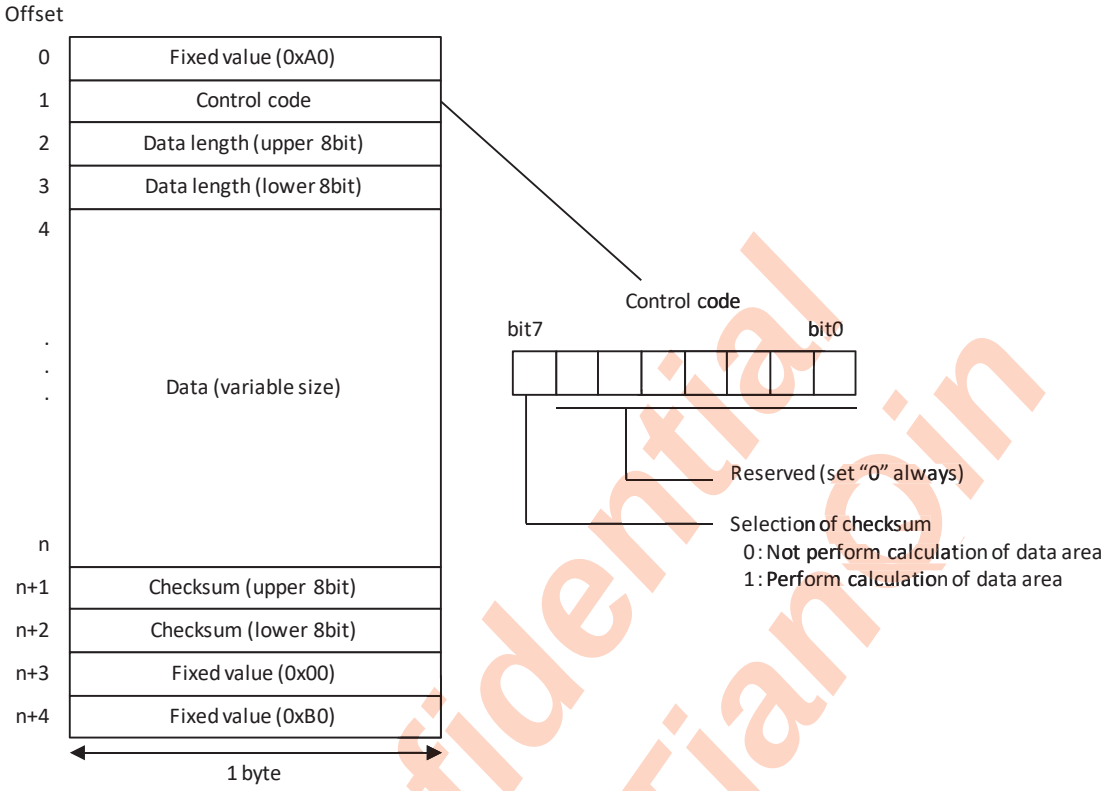
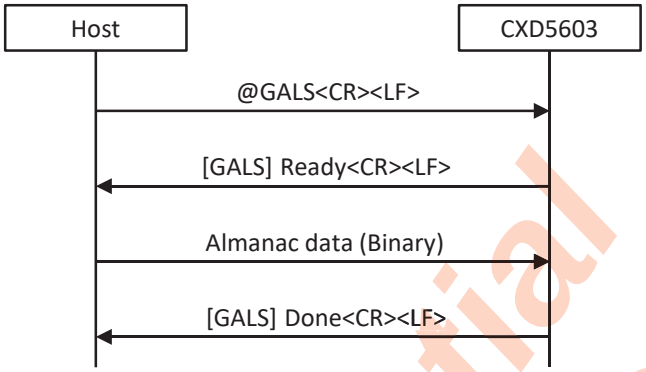


Fig.5 Binary data format

The size of the data excluding the header and footer is described in the "data length" area of the header.

If the data checksum is to be calculated, first the data checksum is described in the "checksum" area of the footer, and then "1" is described for bit 7 of the control code. When "0" is set for bit 7 of the control code, the data in the "checksum" area of the footer is invalid. The data checksum is the lower 16 bits of the sum of the header and the data area in 8 bits unit.

3.2.4 Commands that entail the injection of data (binary data) into the CXD5603GF



The same data format as for the binary data which is transmitted from the CXD5603GF is used. As with the data transmitted from the CXD5603GF, insert a header and footer before and after the data, and set the appropriate values in the fields.

3.3 Command specifications

This section describes the specifications of each command in turn.

3.3.1 @ABPT: Automatic backup data save ON/OFF

This command is used to setting the interval of the automatic backup data save function. The unit is “minute” and the value from 1 to 1,440 min can be set (default value: 60min) .

This command must be issued at Idle state and the automatic backup data save function disabled.

Please take account to the life of the flash memory when using this function.

Format: **@ABPT <arg 1><CR><LF>**

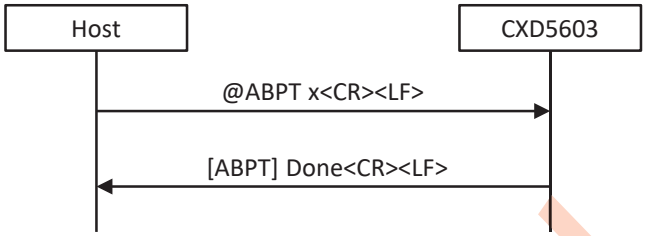
Argument:

Field	Description
arg 1	Set the interval of the automatic backup data save. The unit is minute (default value is 60min) .

Response:

Sentence	Description
“[ABPT] Done”	This indicates that the command has been executed successfully.
“[ABPT] Err n”	This indicates that an error has occurred.

Sequence:



3.3.2 @ABUP: Automatic backup data save ON/OFF

This command is used to control the automatic backup data save function.

When "1" is specified for the argument, the backup data contents are saved in the flash memory automatically at the first fix (This save is not executed if the time specified by @ABPT has not elapsed since the last save). Then the backup data contents are saved in the flash memory automatically with specified interval set by @ABPT beginning at the first fix.

For information about the backup data, see "@BUP".

This command must be issued at Idle state. When this command is issued at Exec state, error is returned. When the automatic backup data save is executing, the sentence may be output erratically sometimes.

Please take account to the life of the flash memory when using this function.

Format: **@ABUP <arg 1><CR><LF>**

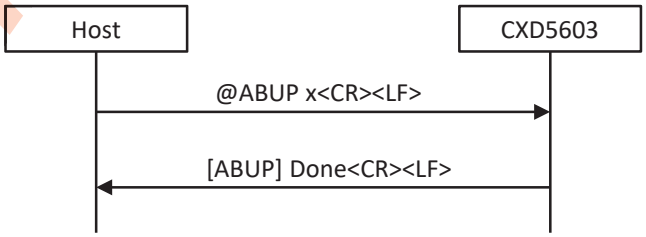
Argument:

Field	Description
arg 1	Control automatic backup data save function. 0: OFF (default value). 1: ON.

Response:

Sentence	Description
"[ABUP] Done"	This indicates that the command has been executed successfully.
"[ABUP] Err n"	This indicates that an error has occurred.

Sequence:



3.3.3 @BSSL: Output sentence select

This command is used to select the NMEA sentence to be output.

The sentences are assigned to each of the bits of the argument. “1” is set for the bits of the sentences which are to be output, and “0” is set for the bits of the sentences whose output is not required. Arguments can be specified in decimal or hexadecimal notation. With hexadecimal notation, add ‘0x’ in front of the numeral.

Format: **@BSSL <arg 1><CR><LF>**

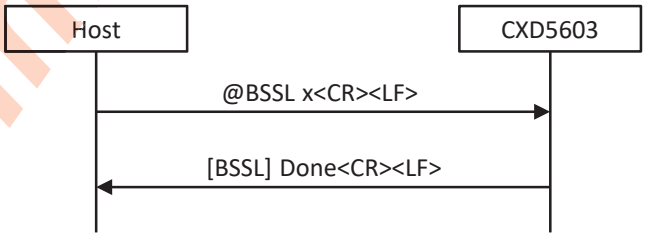
Argument:

Field	Description
arg 1	Output NMEA sentence bit0 : GGA bit1 : GLL bit2 : GSA bit3 : GSV bit4 : GNS bit5 : RMC bit6 : VTG bit7 : ZDA bit8 : Reserved bit9 : Reserved bit10 : Reserved bit11 : Reserved bit12 : Reserved bit13 : Reserved bit14 : Reserved bit15 : Reserved bit16 : Reserved bit17 : Reserved (Default value: 0x000000EF)

Response:

Sentence	Description
“[BSSL] Done”	This indicates that the command has been executed successfully.
“[BSSL] Err n”	This indicates that an error has occurred.

Sequence:



Examples of commands:

```
@BSSL 5<CR><LF>        // Output of GSA, GGA sentences only permitted
```

```
@BSSL 12<CR><LF>      // Output of GSV, GSA sentences only permitted
@BSSL 0x7<CR><LF>     // Output of GSA, GLL, GGA sentences only permitted
```

3.3.4 @BUP: Backup data save

This command is used to save the backup data. The backup data contents are saved in the flash memory.

The backup data saved in the flash memory is automatically restored at boot-up from power OFF.

The receiver position, ephemeris, almanac, TCXO offset and other information required for hot start are included in the backup data, and by saving the backup data in the flash memory using this command, hot start can be initiated when the system is booted from power OFF. (The time must be injected.)

This command must be issued at Idle state. When this command is issued at Exec state, error is returned. For information about the operation status, see **Error! Reference source not found.**.

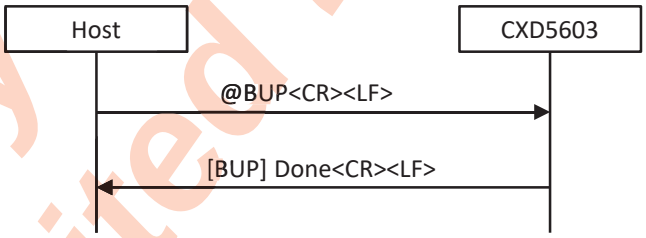
Format: **@BUP<CR><LF>**

Argument: None

Response:

Sentence	Description
“[BUP] Done”	This indicates that the command has been executed successfully.
“[BUP] Err n”	This indicates that an error has occurred.

Sequence:



3.3.5 @BUPC: Backup data clear

This command is used to clear the backup data saved in the flash memory by @BUP.

This command must be issued at Idle state and the automatic backup data save function disabled.

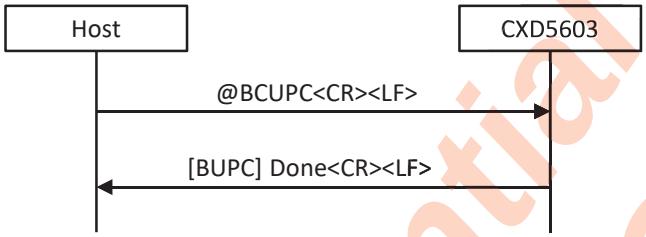
Format: **@BUPC<CR><LF>**

Argument: None

Response:

Sentence	Description
“[BUPC] Done”	This indicates that the command has been executed successfully.
“[BUPC] Err n”	This indicates that an error has occurred.

Sequence:



3.3.6 @CSBR: UART0 baud rate setting

This command is used to set the UART0 baud rate of the CXD5603GF.

When the command is executed successfully, UART0 is changed to the baud rate specified by the argument. Therefore, ensure that the Done response is received at the original baud rate. When the command has failed, the original baud rate is not changed. In the default status, the baud rate is set to 115200 bps.

Format: **`@CSBR <arg 1><CR><LF>`**

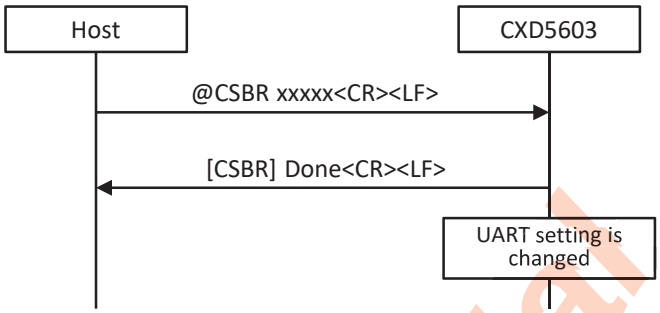
Argument:

Field	Description
arg 1	The baud rate is specified using an integer. The unit used is bps. Specify one of the following as the baud rate. 4800, 9600, 14400, 19200, 38400, 57600, 115200, 230400, 460800 (Default value: 115200)

Response:

Sentence	Description
“[CSBR] Done”	This indicates that the command has been executed successfully.
“[CSBR] Err n”	This indicates that an error has occurred.

Sequence:



3.3.7 @FER: Flash memory erase

This command is used to erase the program on the flash memory. The CXD5603GF re-start after this command issued in special mode.

This command must be issued at “Idle” mode.

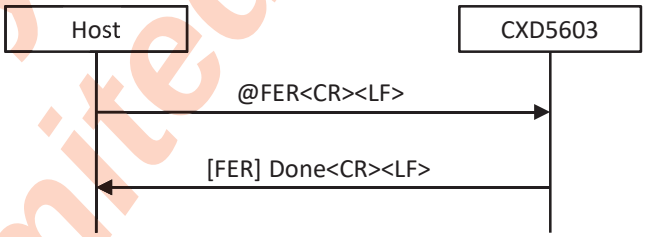
Format: **@FER<CR><LF>**

Argument: None

Response:

Sentence	Description
“[FER] Done”	This indicates that the command has been executed successfully.
“[FER] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.8 @GALG: GPS almanac data acquisition

This command is used to acquire the GPS almanac data received by CXD5603GF. When the command is received, the CXD5603GF transmits the GPS almanac data (binary data) to the host controller.

The GPS almanac data size is 2048 bytes. In addition, the data which is actually transferred has the header and footer added.

This command must be issued at “Idle” mode.

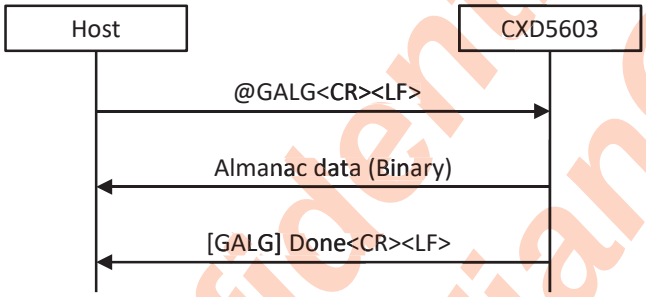
Format: **@GALG<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GALG] Done"	This indicates that the command has been executed successfully.
"[GALG] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.9 @GALS: GPS almanac data injection

This command is used to inject the GPS almanac data into the CXD5603GF. Transmit the GPS almanac data (binary data) following the Ready response from the CXD5603GF.

The GPS almanac data size is 2048 bytes. In addition, the data which is actually required has the header and footer added.

This command must be issued at "Idle" mode.

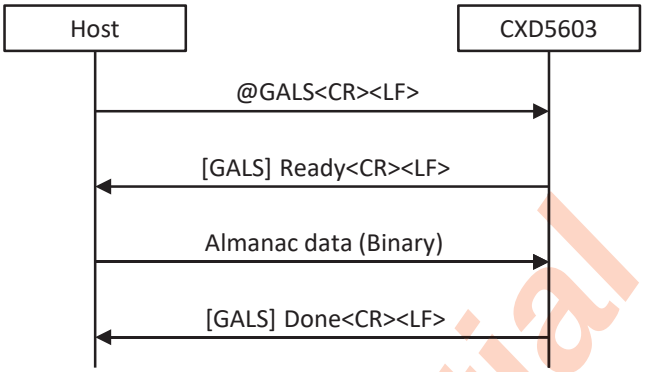
Format: **@GALS<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GALS] Done"	This indicates that the command has been executed successfully.
"[GALS] Ready"	This indicates that the preparations for receiving the almanac data have been completed.
"[GALS] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.10 @GCD: Cold start

This command is used to start the positioning with cold start.

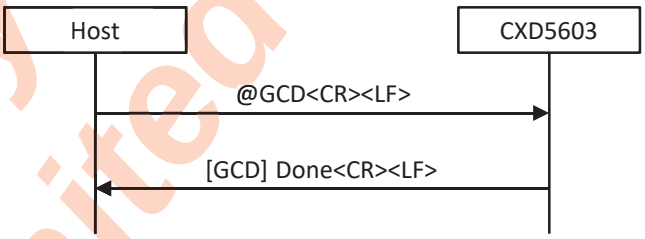
Format: **@GCD<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GCD] Done"	This indicates that the command has been executed successfully.
"[GCD] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.11 @GEMG: GPS ephemeris data acquisition

This command is used to acquire the GPS ephemeris data which has been received by the CXD5603GF. When the command is received, the CXD5603GF transmits the GPS ephemeris data (binary data) to the host controller. The GPS ephemeris data size is 3072 bytes. In addition, the data which is actually transferred has the header and footer added.

This command must be issued at "Idle" mode.

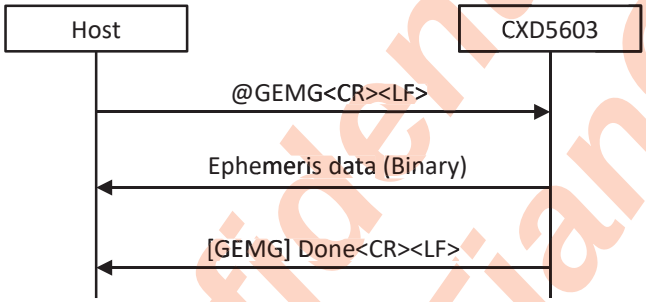
Format: **@GEMG<CR><LF>**

Argument: None

Response:

Sentence	Description
“[GEMG] Done”	This indicates that the command has been executed successfully.
“[GEMG] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.12 @GEMS: GPS ephemeris data injection

This command is used to inject the GPS ephemeris data into the CXD5603GF. Transmit the GPS ephemeris data (binary data) following the Ready response from the CXD5603GF.

The GPS ephemeris data size is 3072 bytes. In addition, the data which is actually required has the header and footer added.

This command must be issued at “Idle” mode.

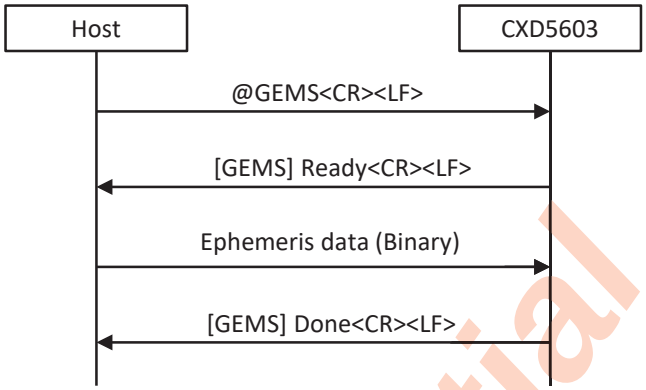
Format: **@GEMS<CR><LF>**

Argument: None

Response:

Sentence	Description
“[GEMS] Done”	This indicates that the command has been executed successfully.
“[GEMS] Ready”	This indicates that the preparations for receiving the almanac data have been completed.
“[GEMS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.13 @GNS: Positioning-use satellite setting

This command is used to select the satellite systems to be used for positioning.

The satellite systems are assigned to the bits of the argument. “1” is set for the bits of the systems which are to be used and “0” is set for the bits of the systems which are not be used. Arguments can be specified in decimal or hexadecimal notation. With hexadecimal notation, add “0x” in front of the numeral.

Only one satellite system can be selected from GLONASS, BeiDou and Galileo. If over 2 satellite systems are selected from these satellite systems, error will return (-EINVAL) .

This command must be issued at “Idle” mode.

Format: **`@GNS <arg 1><CR><LF>`**

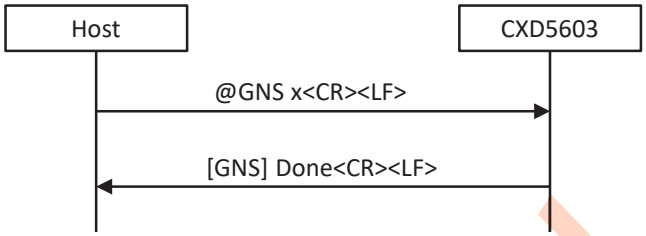
Argument:

Field	Description
arg 1	The satellite systems used for positioning are set on a bit by bit basis (0: system not used, 1: system used). bit 0 : GPS bit 1 : GLONASS bit 2 : SBAS bit 3 : QZSS L1-CA bit 5 : QZSS L1-S bit 6 : BeiDou bit 7 : Galileo (Default value: 0x01)

Response:

Sentence	Description
“[GNS] Done”	This indicates that the command has been executed successfully.
“[GNS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



Examples of commands:

```
@GNS 3<CR><LF>          // Use the GPS and GLONASS systems.
@GNS 0x7<CR><LF>        // Use the GPS, GLONASS and SBAS systems.
```

3.3.14 @GPOE: Receiver position setting (ellipsoidal coordinates)

This command is used to set the approximate position of the receiver in the CXD5603GF. The receiver position is set using ellipsoidal coordinates (latitude, longitude). The north latitude and east longitude directions are "+" values so when specifying the receiver position using a south latitude and west longitude, add a "-" (minus) sign in front to the values.

The receiver position, current time and TCXO offset value are required in order to initiate a hot start so the receiver position must have been set in the CXD5603GF prior to hot start using this command. (This is not necessary if the position is backed up in the flash memory.)

Format: **`@GPOE <arg 1> <arg 2> <arg 3> <arg 4> <arg 5> <arg 6><CR><LF>`**

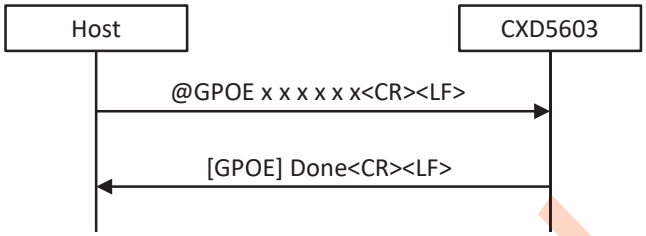
Argument:

Field	Description
arg 1	This specifies the latitude (degrees) of the receiver using an integer.
arg 2	This specifies the latitude (minutes) of the receiver using an integer.
arg 3	This specifies the latitude (seconds) of the receiver using an integer.
arg 4	This specifies the longitude (degrees) of the receiver using an integer.
arg 5	This specifies the longitude (minutes) of the receiver using an integer.
arg 6	This specifies the longitude (seconds) of the receiver using an integer.

Response:

Sentence	Description
"[GPOE] Done"	This indicates that the command has been executed successfully.
"[GPOE] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



Examples of commands:

```
@GPOE 35 37 09 139 43 51<CR><LF>  // North latitude 35°37’09”, east longitude 139°43’51”
@GPOE 33 07 19 -117 19 18<CR><LF> // North latitude 33°07’ 19”, west longitude 117°19’18”
```

3.3.15 @GPOS: Receiver position setting (ellipsoidal coordinates)

This command is used to set the approximate position of the receiver in the CXD5603GF. This command supports higher-accuracy position than @GPOE.

The receiver position is set using ellipsoidal coordinates (latitude, longitude) and altitude. The north latitude and east longitude directions are “+” values so when specifying the receiver position using a south latitude and west longitude, add a “-” (minus) sign in front to the values.

The receiver position, current time and TCXO offset value are required in order to initiate a hot start so the receiver position must have been set in the CXD5603GF prior to hot start using this command. (This is not necessary if the position is backed up in the flash memory.)

Format: **`@GPOS <arg 1> <arg 2> <arg 3><CR><LF>`**

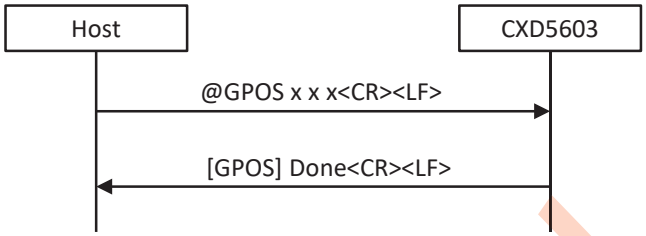
Argument:

Field	Description
arg 1	This specifies the latitude (degrees) * 10^6 of the receiver using an integer. e.g. 43.123456 degrees north: set “43123456”.
arg 2	This specifies the longitude (degrees) *10^6 of the receiver using an integer. e.g.139.789000 degrees east: set “139789000”.
arg 3	This specifies the altitude * 10 of the receiver using an integer. e.g. 102.0m : set “1020”.

Response:

Sentence	Description
“[GPOS] Done”	This indicates that the command has been executed successfully.
“[GPOS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



Examples of commands:

```
@GPOS 35123456 139987650 0<CR><LF>     // North latitude 35.123456 degree,
                                       // east longitude 139.987650 degree,
                                       // altitude 0m
@GPOS 33070710 -117121310 -15<CR><LF> // North latitude 33.070710 degree,
                                       // west longitude 117.121310 degree,
                                       // altitude -1.5m
```

3.3.16 @GPPS: 1PPS output setting

This command is used to control 1PPS output.

When 1PPS output is enabled, timing pulse is output in 1 sec period from 1PPS output port after clock information being received from GNSS. When 1PPS output is disabled, timing pulse is not output from 1PPS output port.

Format: **@GPPS <arg 1><CR><LF>**

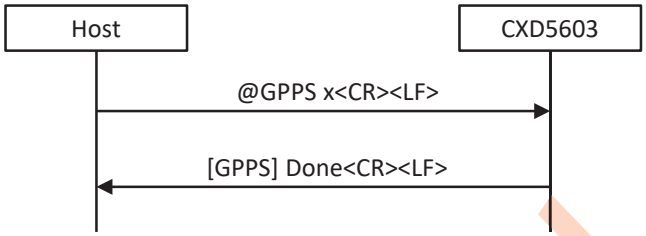
Argument:

Field	Description
arg 1	1PPS output control 0 : Disable 1PPS output (default value) 1 : Enable 1PPS output

Response:

Sentence	Description
“[GPPS] Done”	This indicates that the command has been executed successfully.
“[GPPS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.17 @GPTC: TCXO offset acquisition

This command is used to acquire the TCXO offset value measured by the CXD5603GF. When the command is received, the CXD5603GF transmits the TCXO offset value (ASCII data) . The unit of the TCXO offset value is "Hz" and the sign (+ or -) is added at the top.

The value converted by GPS L1 frequency is acquired. When getting TCXO frequency offset, this value must be multiplied by (-1 * Nominal frequency of TCXO) / 1575420000.

When the TCXO offset has not been calculated, the text "INVALID" returns.

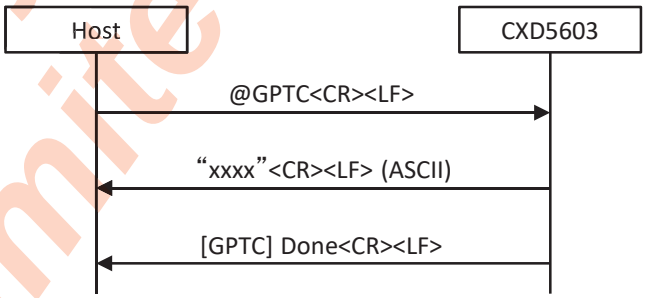
Format: **@GPTC<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GPTC] Done"	This indicates that the command has been executed successfully.
"[GPTC] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.18 @GSOP: Operation mode setting

This command is used to set the operation mode of the CXD5603GF. The operation mode and positioning cycle can be specified.

The sleep time can be specified but only when the Normal mode has been specified. The positioning operation is

performed during the remaining time of the positioning cycle after operation has transferred to the Sleep state for the time specified with each specified positioning cycle. When the fix is not valid, some satellites are tracked and the operation time (equal to the positioning cycle minus the sleep time) is less than 1 minute, the CXD5603GF doesn't transit to the Sleep state in this usage. If the sleep time must be kept certainly, set the parameters so that the operation time may be 1 minute or more.

When the Low power mode is used, set the positioning cycle to the value over 1sec.

Format: **@GSOP <arg 1> <arg 2> <arg 3><CR><LF>**

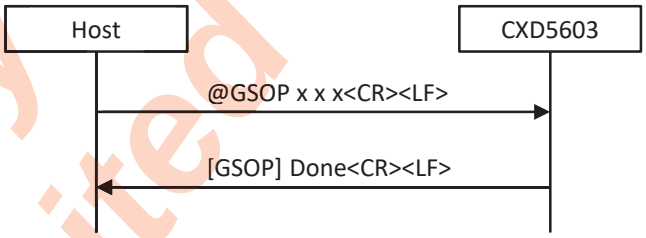
Argument:

Field	Description
arg 1	This specifies the operation mode of the receiver. 1 : Normal (default value) 2 : Low Power
arg 2	This specifies the positioning cycle [ms] using an integer. (Default value: 1000)
arg 3	This specifies the sleep time [ms] in the Normal mode using an integer. When "0" is specified, the sleep operation is not performed, and positioning is executed continuously. In modes other than Normal, this is an invalid parameter. (Default value: 0)

Response:

Sentence	Description
"[GSOP] Done"	This indicates that the command has been executed successfully.
"[GSOP] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



Examples of commands:

```
@GSOP 1 3000 0<CR><LF>          // Normal mode, positioning cycle of 3 seconds (no sleep
                                // operation)
@GSOP 1 10000 5000<CR><LF>      // Normal mode, positioning cycle of 10 seconds (sleep
                                // time of 5 seconds)
                                // (The pattern of a sleep time of 5 seconds and the
                                // positioning operation of 5 seconds is repeated.)
@GSOP 2 30000 0<CR><LF>         // Low power mode, positioning cycle of 30 seconds
```

3.3.19 @GSP: Hot start for position accuracy

This command is used to start positioning using a hot start. The position accuracy is prioritized until first fix. TTFF is about 1s longer than @GSR. There is no difference with @GSR after first fix.

When the conditions for the hot start have not been met, positioning is started automatically using a warm start or cold start.

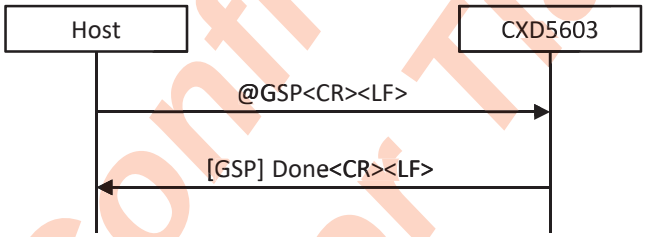
Format: **@GSP<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GSP] Done"	This indicates that the command has been executed successfully.
"[GSP] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.20 @GSR: Hot start for TTFF

This command is used to start positioning using a hot start. The TTFF is prioritized until first fix. TTFF is about 1s shorter than @GSP but the position accuracy is somewhat worse than @GSP. There is no difference with @GSP after first fix.

When the conditions for the hot start have not been met, positioning is started automatically using a warm start or cold start.

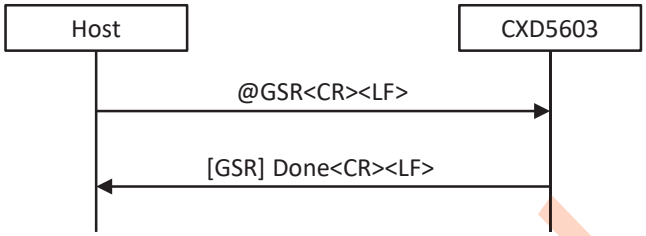
Format: **@GSR<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GSR] Done"	This indicates that the command has been executed successfully.
"[GSR] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.21 @GSTP: Positioning stop

This command is used to stop the positioning. The CXD5603GF transfers to the Idle state.

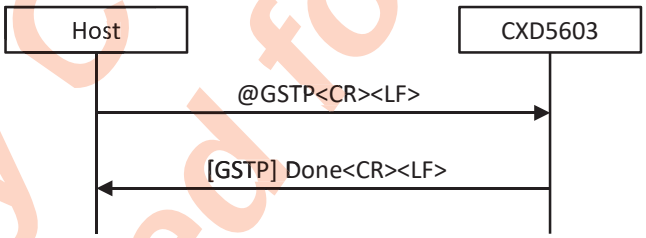
Format: **@GSTP<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GSTP] Done"	This indicates that the command has been executed successfully.
"[GSTP] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.22 @GSUC: Positioning algorithm setting

This command is used to select the GNSS positioning algorithm for the special use case.

When enabling some algorithm, set the appropriate bits of argument 1 and set "1 (enable)" to argument 2. When disabling some algorithm, set the appropriate bits of argument 1 and set "0 (disable)" to argument 2.

Format: **@GSUC <arg 1> <arg 2><CR><LF>**

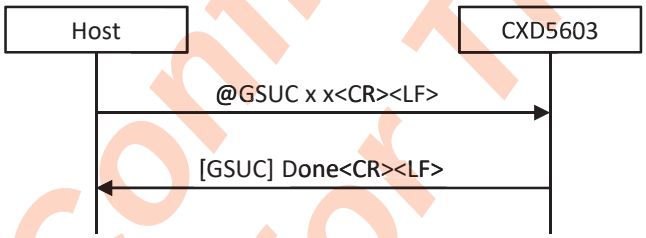
Argument:

Field	Description
arg 1	Select the algorithm to enable or disable. The set algorithms can be enabled/disabled by setting of argument 2. The setting of the other algorithm does not change. GNSS positioning algorithm are set on a bit by bit basis (0: not used, 1: used). bit 7 : Swimming mode bit 14 : Trail mode bit 15 : Driving mode
arg 2	Select enabling or disabling of the algorithm selected by argument 1. 0 : Disable 1 : Enable

Response:

Sentence	Description
“[GSUC] Done”	This indicates that the command has been executed successfully.
“[GSUC] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.23 @GSW: Warm start

This command is used to start positioning using a warm start. When the conditions for the warm start have not been met, positioning is started automatically using a cold start.

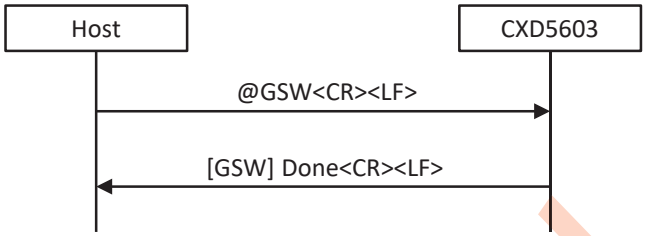
Format: **@GSW<CR><LF>**

Argument: None

Response:

Sentence	Description
“[GSW] Done”	This indicates that the command has been executed successfully.
“[GSW] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.24 @GTCX: TCXO offset setting

This command is used to set the TCXO offset value of the receiver in the CXD5603GF. The TCXO offset value of the receiver is set in Hz. The "+" or "-" direction can be specified by adding a sign to the argument.

The receiver position, current time and TCXO offset value are required in order to initiate a hot start so the time must have been set in the CXD5603GF prior to hot start using this command (This is not necessary if the time is backed up on the flash memory).

Format: **`@GTCX <arg 1><CR><LF>`**

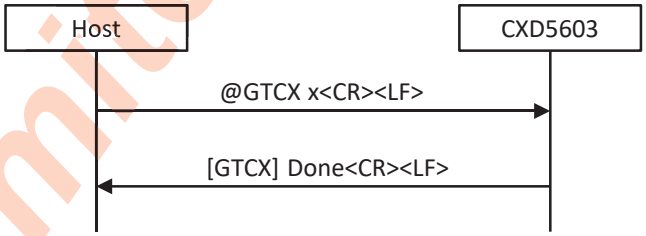
Argument:

Field	Description
arg 1	The TCXO offset value (Hz) is set using an integer. (Default value: 0)

Response:

Sentence	Description
"[GTCX] Done"	This indicates that the command has been executed successfully.
"[GTCX] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



Examples of commands:

```
@GTCX -250<CR><LF>     // TCXO offset: -250Hz
@GTCX 100<CR><LF>      // TCXO offset: +100Hz
```

3.3.25 @GTE: GPS test end

This command is used to end the GPS test. When the test is ended using the command, the CXD5603GF returns to the state in which normal commands can be received.

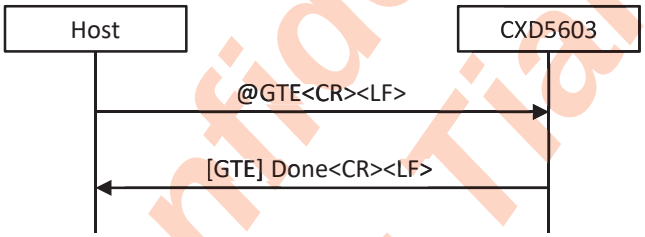
Format: **@GTE<CR><LF>**

Argument: None

Response:

Sentence	Description
"[GTE] Done"	This indicates that the command has been executed successfully.
"[GTE] Err n"	This indicates that an error has occurred. "n" is where the error code is entered.

Sequence:



3.3.26 @GTIM: Time setting

This command is used to set the time of the receiver in the CXD5603GF. The UTC time standard is used for the receiver time which employs the format of year, month, day, hours, minutes and seconds.

The receiver position, current time and TCXO offset value are required in order to initiate a hot start so the time must have been set in the CXD5603GF prior to hot start using this command.

Format: **@GTIM <arg 1> <arg 2> <arg 3> <arg 4> <arg 5> <arg 6><CR><LF>**

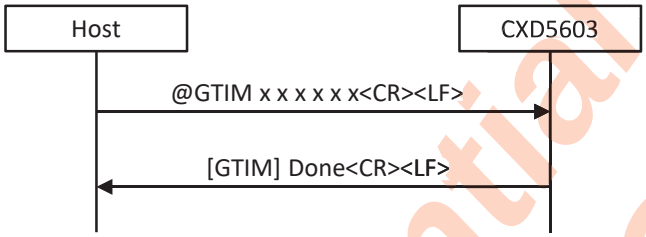
Argument:

Field	Description
arg 1	This specifies the UTC time (year) using an integer.
arg 2	This specifies the UTC time (month) using an integer.
arg 3	This specifies the UTC time (day) using an integer.
arg 4	This specifies the UTC time (hour) using an integer.
arg 5	This specifies the UTC time (minutes) using an integer.
arg 6	This specifies the UTC time (seconds) using an integer.

Response:

Sentence	Description
“[GTIM] Done”	This indicates that the command has been executed successfully.
“[GTIM] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



Examples of commands:

```
@GTIM 2013 02 01 13 30 30<CR><LF>      // 2013/2/1 13:30:30
@GTIM 2013 07 10 00 00 00<CR><LF>      // 2013/7/10 00:00:00”
```

3.3.27 @GTR: GPS test result output

This command is used to output the GPS test results. Wait one second after the @GTS command is issued, and then issue the command.

The CN level [dBHz] and Doppler frequency [Hz] are returned as the test results.

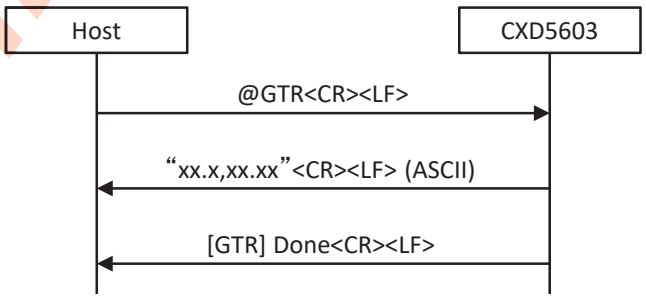
Format: **@GTR<CR><LF>**

Argument: None

Response:

Sentence	Description
“[GTR] Done”	This indicates that the command has been executed successfully.
“[GTR] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.28 @GTS: GPS test start

This command is used to start the GPS test. The test results are output by issuing the @GTR command after a wait of one second after the @GTS command has been issued.

This command can be issued only in the Idle state. When it is issued, no subsequent commands except for the @GTR and @GTE commands are accepted.

Format: **@GTS <arg 1> <arg 2> <arg 3> <arg 4><CR><LF>**

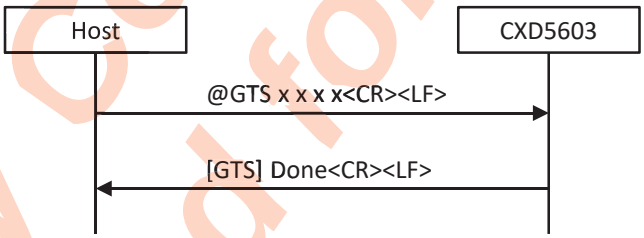
Argument:

Field	Description
arg 1	This specifies the number of the satellite used for the test.
arg 2	Reserved. Always specify “0” for this.
arg 3	Reserved. Always specify “0” for this.
arg 4	Reserved. Always specify “0” for this.

Response:

Sentence	Description
“[GTS] Done”	This indicates that the command has been executed successfully.
“[GTS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



Examples of commands:

```
@GTS 1 0 0 0<CR><LF> // The test is started using satellite no.1.
```

3.3.29 @GUSE: Positioning algorithm setting

This command is used to select the GNSS positioning algorithm for the special use case.

In normal use case, select the default algorithm by setting “0x00” for the argument. When the special algorithm should be used, set the appropriate bits.

Format: **@GUSE <arg 1><CR><LF>**

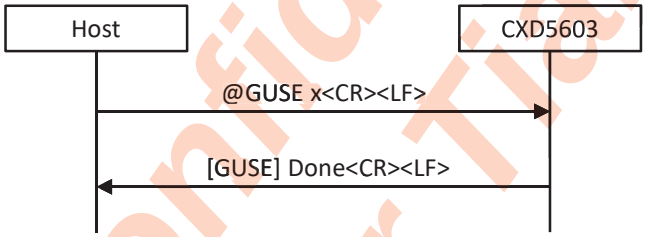
Argument:

Field	Description
arg 1	GNSS positioning algorithm are set on a bit by bit basis (0: not used, 1: used). bit 0 : Special algorithm for swinging the receiver by the hand. bit 1 : reserved (always specify ”0” for this) bit 2 : reserved (always specify ”0” for this) bit 3 : reserved (always specify ”0” for this) bit 4 : reserved (always specify ”0” for this) bit 5 : reserved (always specify ”0” for this) bit 6 : reserved (always specify ”0” for this) bit 7 : reserved (always specify ”0” for this) (Default value: 0x01)

Response:

Sentence	Description
“[GUSE] Done”	This indicates that the command has been executed successfully.
“[GUSE] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.30 @LALG: GLONASS almanac data acquisition

This command is used to acquire the GLONASS almanac data received by CXD5603GF. When the command is received, the CXD5603GF transmits the GLONASS almanac data (binary data) to the host controller.

The GLONASS almanac data size is 576 bytes. In addition, the data which is actually transferred has the header and footer added.

This command must be issued at “Idle” mode.

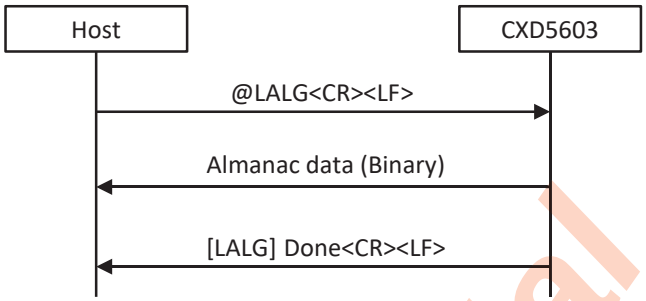
Format: **@LALG<CR><LF>**

Argument: None

Response:

Sentence	Description
“[LALG] Done”	This indicates that the command has been executed successfully.
“[LALG] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.31 @LALS: GLONASS almanac data injection

This command is used to inject the GLONASS almanac data into the CXD5603GF. Transmit the GLONASS almanac data (binary data) following the Ready response from the CXD5603GF.

The GLONASS almanac data size is 576 bytes. In addition, the data which is actually required has the header and footer added.

This command must be issued at “Idle” mode.

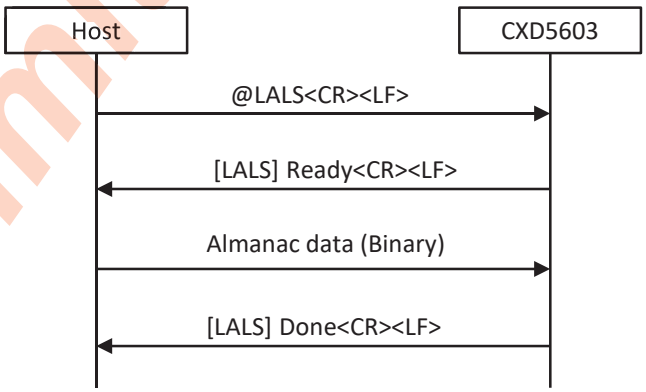
Format: **@LALS<CR><LF>**

Argument: None

Response:

Sentence	Description
“[LALS] Done”	This indicates that the command has been executed successfully.
“[LALS] Ready”	This indicates that the preparations for receiving the almanac data have been completed.
“[LALS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.32 @LEMG: GLONASS ephemeris data acquisition

This command is used to acquire the GLONASS ephemeris data which has been received by CXD5603GF.

When the command is received, the CXD5603GF transmits the GLONASS ephemeris data (binary data) to the host controller.

The GLONASS ephemeris data size is 1152 bytes. In addition, the data which is actually transferred has the header and footer added.

This command must be issued at “Idle” mode.

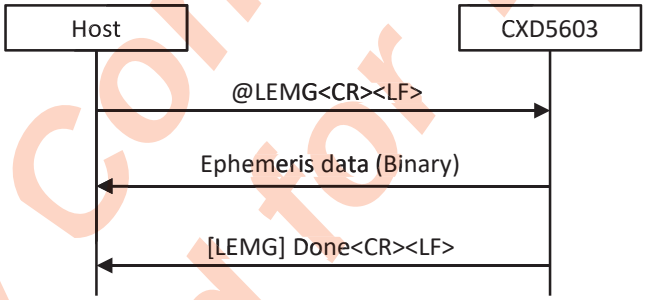
Format: **@LEMG<CR><LF>**

Argument: None

Response:

Sentence	Description
“[LEMG] Done”	This indicates that the command has been executed successfully.
“[LEMG] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.33 @LEMS: GLONASS ephemeris data injection

This command is used to inject the GLONASS ephemeris data into the CXD5603GF. Transmit the GLONASS ephemeris data (binary data) following the Ready response from the CXD5603GF.

The GLONASS ephemeris data size is 1152 bytes. In addition, the data which is actually required has the header and footer added.

This command must be issued at “Idle” mode.

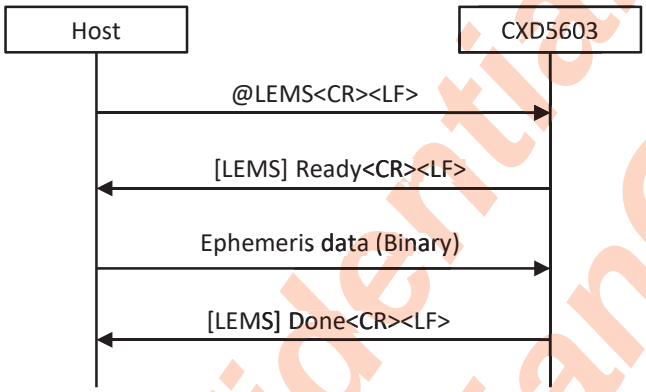
Format: **@LEMS<CR><LF>**

Argument: None

Response:

Sentence	Description
“[LEMS] Done”	This indicates that the command has been executed successfully.
“[LEMS] Ready”	This indicates that the preparations for receiving the almanac data have been completed.
“[LEMS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.34 @QALG: QZSS almanac data acquisition

This command is used to acquire the QZSS almanac data received by CXD5603GF. When the command is received, the CXD5603GF transmits the QZSS almanac data (binary data) to the host controller.

The QZSS almanac data size is 672 bytes. In addition, the data which is actually transferred has the header and footer added.

This command must be issued at “Idle” mode.

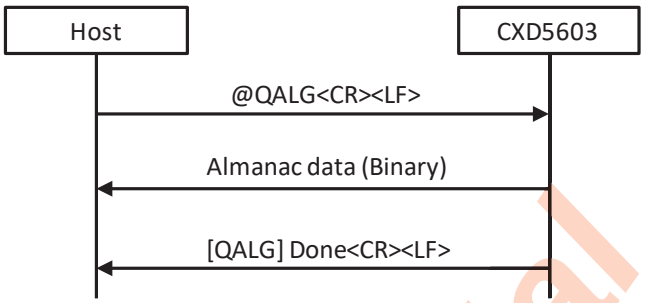
Format: **@QALG<CR><LF>**

Argument: None

Response:

Sentence	Description
“[QALG] Done”	This indicates that the command has been executed successfully.
“[QALG] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.35 @QALS: QZSS almanac data injection

This command is used to inject the QZSS almanac data into the CXD5603GF. Transmit the QZSS almanac data (binary data) following the Ready response from the CXD5603GF.

The QZSS almanac data size is 672 bytes. In addition, the data which is actually required has the header and footer added.

This command must be issued at “Idle” mode.

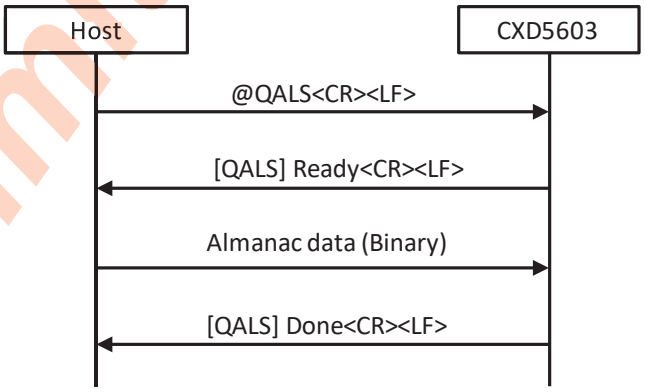
Format: **@QALS<CR><LF>**

Argument: None

Response:

Sentence	Description
“[QALS] Done”	This indicates that the command has been executed successfully.
“[QALS] Ready”	This indicates that the preparations for receiving the almanac data have been completed.
“[QALS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.36 @QEMG: QZSS ephemeris data acquisition

This command is used to acquire the QZSS ephemeris data which has been received by CXD5603GF. When the command is received, the CXD5603GF transmits the QZSS ephemeris data (binary data) to the host controller. The QZSS ephemeris data size is 960 bytes. In addition, the data which is actually transferred has the header and footer added.

This command must be issued at “Idle” mode.

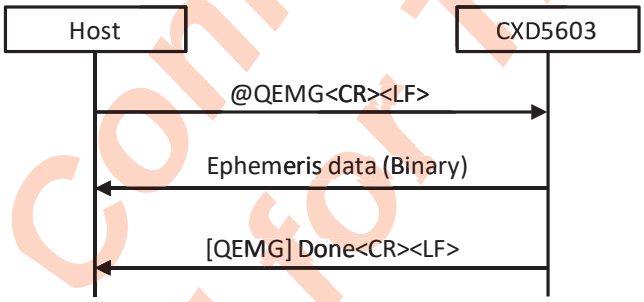
Format: **@QEMG<CR><LF>**

Argument: None

Response:

Sentence	Description
“[QEMG] Done”	This indicates that the command has been executed successfully.
“[QEMG] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.37 @QEMS: QZSS ephemeris data injection

This command is used to inject the QZSS ephemeris data into the CXD5603GF. Transmit the QZSS ephemeris data (binary data) following the Ready response from the CXD5603GF.

The QZSS ephemeris data size is 960 bytes. In addition, the data which is actually required has the header and footer added.

This command must be issued at “Idle” mode.

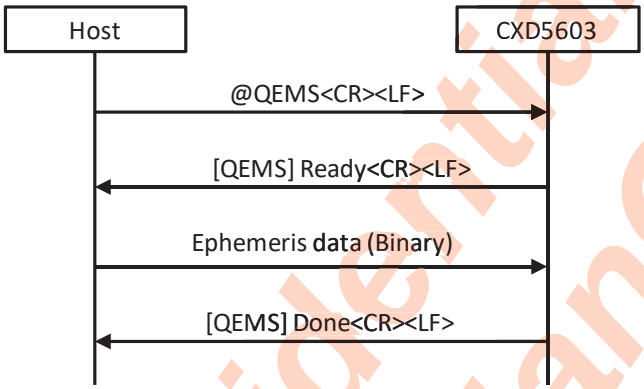
Format: **@QEMS<CR><LF>**

Argument: None

Response:

Sentence	Description
“[QEMS] Done”	This indicates that the command has been executed successfully.
“[QEMS] Ready”	This indicates that the preparations for receiving the almanac data have been completed.
“[QEMS] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.38 @SLP: Sleep

This command is used to transfer operation to the Sleep state. It specifies transfer to each sleep state using an argument. The status at sleeping differs according to Sleep states.

State	Main RAM	Backup RAM	RTC	After wake up
Sleep 0	Retained	Retained	Operation	Re-start with previous setting
Sleep 1	OFF	Retained	Operation	Reboot
Sleep 2	OFF	OFF	Operation	Reboot

This command must be issued at Idle state. When this command is issued at Exec state, error is returned.

Format: **@SLP <arg 1><CR><LF>**

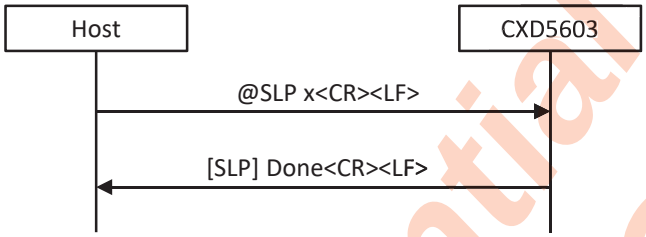
Argument:

Field	Description
arg 1	This selects whether to transfer to the Sleep state or Deep Sleep state. 0: Transfer to Sleep 0. 1: Transfer to Sleep 1. 2: Transfer to Sleep 2.

Response:

Sentence	Description
“[SLP] Done”	This indicates that the command has been executed successfully.
“[SLP] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.39 @VER: Firmware revision number acquisition

This command is used to acquire the revision number of the firmware.

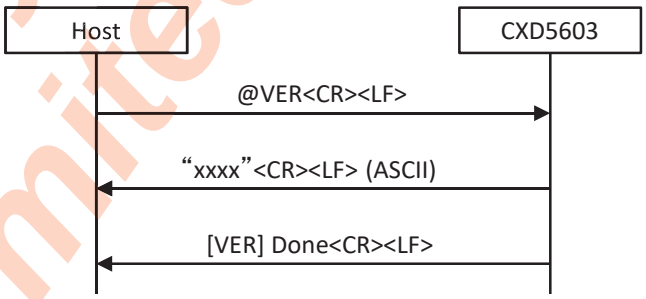
Format: **@VER<CR><LF>**

Argument: None

Response:

Sentence	Description
“[VER] Done”	This indicates that the command has been executed successfully.
“[VER] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



3.3.40 @WUP: Wake-up

This command is used to transfer to the Idle state from the Sleep state.

When this command has been issued in the Sleep state, the command reply message is not output until the transfer to the Idle state is completed. Repeatedly issue this command until the command reply message is

output.

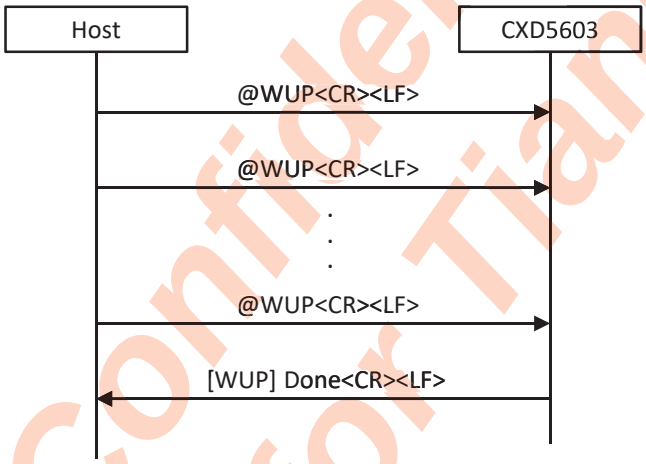
Format: **@WUP<CR><LF>**

Argument: None

Response:

Sentence	Description
“[WUP] Done”	This indicates that the command has been executed successfully.
“[WUP] Err n”	This indicates that an error has occurred. “n” is where the error code is entered.

Sequence:



4 NMEA sentence specifications

This section describes the specifications of NMEA sentences. CXD5603GF outputs NMEA0183 compliant sentences, IMES sentences and proprietary sentences whose talker ID is “$PS”.

4.1 NMEA 0183 compliant sentences

CXD5603GF outputs below sentences of NMEA0183 (ver 4.10) compliant sentences.

4.1.1 GGA : Global Positioning System Fix Data

Format:`$--GGA,hhmmss.ss,llll.ll,a,yyyyy.yy,a,x,xx,x.x,x.x,M,x.x,M,x.x,xxxx*hh`**<CR><LF>**

Fields:

Field	Format	Description
Header	$	
Talker ID	--	GP
Sentence ID	GGA	
UTC of position	hhmmss.ss	hh [hour] mm [min] ss.ss [sec]
Latitude	llll.ll	dd [degree] mm.mmmm [min]
Latitude – N/S	a	N : North latitude, S : South latitude
Longitude	yyyyy.yy	ddd [degree] mm.mmmm [min]
Longitude – E/W	a	E : East longitude, W : West longitude
Quality indicator	x	0 : Fix not available 1 : Fix valid 2 : Fix valid, Differential GPS 6 : Dead reckoning
Number of satellites in use	xx	
HDOP	x.x	
Altitude (mean-sea-level), meters	x.x,M	[m]
Geoidal separation, meters	x.x,M	[m]
Age of DGPS data	x.x	NULL
Differential reference station ID	xxxx	NULL
Checksum	*hh	
Termination	<CR><LF>	

4.1.2 GLL : Geographic Position – Latitude / Longitude

Format : `$--GLL,llll.ll,a,yyyyy.yy,a,hhmmss.ss,A,a*hh<CR><LF>`

Fields:

Field	Format	Description
Header	$	
Talker ID	--	GP : Using only GPS for positioning GL : Using only GLONASS for positioning GA : Using only Galileo for positioning BD : Using only BeiDou for positioning GQ : Using only QZS for positioning GN : Using combined satellite systems for positioning
Sentence ID	GLL	
Latitude	llll.ll	dd [degree] mm.mmmm [min]
Latitude – N/S	a	N : North latitude, S : South latitude
Longitude	yyyyy.yy	ddd [degree] mm.mmmm [min]
Longitude – E/W	a	E : East longitude, W : West longitude
UTC of position	hhmmss.ss	hh [hour] mm [min] ss.ss [sec]
Status	A	A : Data valid, V : Data not valid
Mode Indicator	a	Positioning system Mode Indicator : A : Autonomous mode D : Differential mode E : Dead reckoning mode N : Data not valid
Checksum	*hh	
Termination	<CR><LF>	

4.1.3 GNS: GNSS Fix Data

Format:

`$--GNS,hhmmss.ss,llll.ll,a,yyyyy.yy,a,c--c,xx,x.x,x.x,M,x.x,M,x.x,xxxx,a*hh<CR><LF>`

Fields:

Field	Format	Description
Header	$	
Talker ID	--	GP : Using only GPS for positioning GL : Using only GLONASS for positioning GA : Using only Galileo for positioning BD : Using only BeiDou for positioning GQ : Using only QZS for positioning GN : Using combined satellite systems for positioning
Sentence ID	GNS	
UTC of position	hhmmss.ss	hh [hour] mm [min] ss.ss [sec]
Latitude	llll.ll	dd [degree] mm.mmmm [min]
Latitude – N/S	a	N : North latitude, S : South latitude
Longitude	yyyyy.yy	ddd [degree] mm.mmmm [min]
Longitude – E/W	a	E : East longitude, W : West longitude
Mode indicator	c--c	Positioning system Mode Indicator (1st character : GPS, 2nd character : GLONASS) A : Autonomous mode D : Differential mode E : Dead reckoning mode N : Data not valid
Number of satellites in use	xx	
HDOP	x.x	
Altitude (mean-sea-level)	x.x,M	[m]
Geoidal separation, meters	x.x,M	[m]
Age of DGPS data	x.x	
Differential reference station ID	xxxx	NULL
Navigation status	a	
Checksum	*hh	
Termination	<CR><LF>	

4.1.4 GSA: GNSS DOP and Active Satellites

When the combined satellite systems are used for positioning, the sentences from each satellite system are output one by one (Talker ID of each sentences are “GN”).

Format:$--GSA,a,x,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,x.x,x.x,x.x,h*hh**<CR><LF>**

Fields:

Field	Format	Description
Header	$	
Talker ID	--	GP : Using only GPS for positioning GL : Using only GLONASS for positioning GA : Using only Galileo for positioning BD : Using only BeiDou for positioning GQ : Using only QZS for positioning GN : Using combined satellite systems for positioning
Sentence ID	GSA	
2D / 3D Mode	a	A : Automatically switch 2D / 3D
Mode	x	1 : Fix not available, 2 : 2D, 3 : 3D
Used satellite #1	xx	
. . .		
Used satellite #12	xx	
PDOP	x.x	
HDOP	x.x	
VDOP	x.x	
GNSS system ID	h	1 : GPS 2 : GLONASS 3 : Galileo 4 : BeiDou 5 : QZSS
Checksum	*hh	
Termination	<CR><LF>	

4.1.5 GSV: GNSS Satellites In View

Format : `$--GSV,x,x,xx,xx,xx,xxx,xx,.....,xx,xx,xxx,xx,h*hh<CR><LF>`

Fields:

Field		Format	Description
Header		$	
Talker ID		--	GP : GPS satellites in view GL : GLONASS satellites in view GA : Galileo satellites in view BD : BeiDou satellites in view GQ : QZS satellites in view
Sentence ID		GSV	
Total number of sentences		x	
Sentence number		x	
Total number of satellites in view		xx	
SV1	Satellite ID	xx	
	Elevation	xx	[degree]
	Azimuth	xxx	[degree]
	SNR (C/N)	xx	[dB-Hz] (NULL at no acquisition)
SV2	Satellite ID	xx	
	Elevation	xx	[degree]
	Azimuth	xxx	[degree]
	SNR (C/N)	xx	[dB-Hz] (NULL at no acquisition)
SV3	Satellite ID	xx	
	Elevation	xx	[degree]
	Azimuth	xxx	[degree]
	SNR (C/N)	xx	[dB-Hz] (NULL at no acquisition)
SV4	Satellite ID	xx	
	Elevation	xx	[degree]
	Azimuth	xxx	[degree]
	SNR (C/N)	xx	[dB-Hz] (NULL at no acquisition)
Signal ID		h	
Checksum		*hh	
Termination		<CR><LF>	

4.1.6 RMC: Recommended Minimum Specific GNSS Data

Format : `$--RMC,hhmmss.ss,A,llll.ll,a,yyyyy.yy,a,x.x,x.x,xxxxxx,x.x,a,a,a*hh<CR><LF>`

Fields:

Field	Format	Description
Header	$	
Talker ID	--	GP : Using only GPS for positioning GL : Using only GLONASS for positioning GA : Using only Galileo for positioning BD : Using only BeiDou for positioning GQ : Using only QZS for positioning GN : Using combined satellite systems for positioning
Sentence ID	RMC	
UTC of position fix	hhmmss.ss	hh [hour] mm [min] ss.ss [sec]
Status	A	A : Data valid, V : Data not valid
Latitude	llll.ll	dd [degree] mm.mmmm [min]
Latitude – N/S	a	N : North latitude, S : South latitude
Longitude	yyyyy.yy	ddd [degree] mm.mmmm [min]
Longitude – E/W	a	E : East longitude, W : West longitude
Speed over ground	x.x	[knot]
Course over ground	x.x	[degree]
Date	xxxxxx	dd [day] mm [month] yy [year]
Magnetic variation	x.x	[degree]
Magnetic variation – E/W	a	E : East, W : West
Mode Indicator	a	A : Autonomous mode D : Differential mode E : Dead reckoning mode N : Data not valid
Navigation status	a	
Checksum	*hh	
Termination	<CR><LF>	

4.1.7 VTG: Course Over Ground & Ground Speed

Format : `$--VTG,x.x,T,x.x,M,x.x,N,x.x,K,a*hh<CR><LF>`

Fields:

Field	Format	Description
Header	$	
Talker ID	--	GP : Using only GPS for positioning GL : Using only GLONASS for positioning GA : Using only Galileo for positioning BD : Using only BeiDou for positioning GQ : Using only QZS for positioning GN : Using combined satellite systems for positioning
Sentence ID	VTG	
Course over ground - True	x.x,T	[degrees]
Course over ground - Magnetic	x.x,M	NULL
Speed over ground	x.x,N	[knot]
Speed over ground	x.x,K	[km/h]
Mode Indicator	a	A : Autonomous mode D : Differential mode E : Dead reckoning mode N : Data not valid
Checksum	*hh	
Termination	<CR><LF>	

4.1.8 ZDA: Time & Date

Format : `$--ZDA,hhmmss.ss,xx,xx,xxxx,xx,xx*hh<CR><LF>`

Fields :

Field	Format	Description
Header	$	
Talker ID	--	GP : Using only GPS for positioning GL : Using only GLONASS for positioning GA : Using only Galileo for positioning BD : Using only BeiDou for positioning GQ : Using only QZS for positioning
Sentence ID	ZDA	
UTC	hhmmss.ss	hh [hour] mm [min] ss.ss [sec]
Day	xx	
Month	xx	
Year	xxxx	
Local zone hours	xx	NULL
Local zone minutes	xx	NULL
Checksum	*hh	
Termination	<CR><LF>	

4.2 Satellite ID

Below values are stored in satellite ID of GSA and GSV sentences.

Satellite system	Talker ID	Satellite ID
GPS	GP	1～32
SBAS	GP	33～64
GLONASS	GL	65～88
Galileo	GA	1～36
BeiDou	BD	1～30
QZSS (L1 C/A)	GQ	1～10
QZSS (L1S)	GQ	55～63

5 CXD5603GF operation states

The operation of the CXD5603GF has five states, and the transitions between these states are shown below.

5.1 Operation states

The operation status of the CXD5603GF has five states, and the CXD5603GF transits between these states as shown in Fig.6.

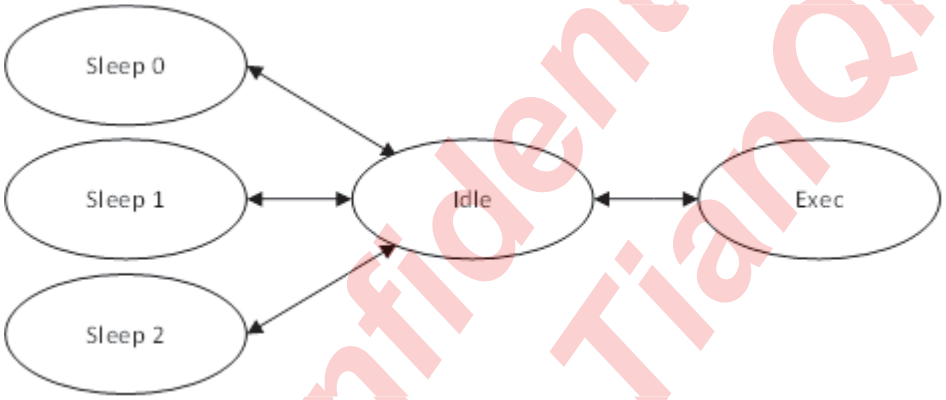


Fig.6 Transition between the CXD5603GF states

Each of the states is defined below.

Sleep 0

In this state, the power is supplied only to CPU, RAM, the backup RAM and real-time clock. CPU is in WFI state. The program and data in RAM and clock are retained. After wakeup, the program re-starts with previous status before entered sleep.

Sleep 1

In this state, the power is supplied only to the backup RAM and real-time clock. The real-time clock continues to operate, and the values in the backup RAM (where the receiver position, ephemeris, almanac, TCXO offset value, etc. are stored) are retained so the conditions required for hot start are retained. After wakeup, the program is rebooted.

Sleep 2

In this state, the power of all the blocks except for the real-time clock has been turned off so the power consumption is the lowest. The real-time clock continues to operate so the time is retained. After wakeup, the program is rebooted.

Idle

In this state, the power of all the blocks is supplied, and the GPS operation is stopped.

Exec
In this state, the power of all the blocks is supplied, and the GPS positioning operation is underway. Some blocks may be turned off depending on the conditions of positioning operation and satellite signal.

Transitions from one state to another can be initiated by issuing commands from the host controller.
When GPS has started positioning and the Low Power mode has been selected as the operation mode, the receiver state is being selected automatically to minimize the power consumption.

5.2 Operation modes

There are three operation modes in the positioning, and they can be specified using the @GSOP command. These operation modes can be switched during operation.

Normal
In this mode, all the GPS-related circuits are activated, and the positioning operation is performed continuously. In this mode, the GPS circuits and positioning processing are operating so the power consumption is the highest but the performance is also the highest.
The Sleep time can be specified only when the Normal mode has been selected. Operation transfers to the Sleep mode only for the specified time with each positioning cycle, and the positioning operation is performed continuously for the remaining time of the positioning cycle.

Low Power
In this mode, the positioning operation is performed at a low level of power consumption. Once the satellites are picked up and positioning starts, some of the GPS circuits are set to OFF, and operation is performed intermittently at a low level of power consumption. If the positioning has failed or the number of satellites has decreased, some of the GPS circuits are set to ON, and the positioning operation is performed continuously.

5.2.1 Normal mode

The CXD5603GF works continuously with all GPS circuits activated and outputs NMEA sentences with the specified period by @GSOP as shown in Fig.7. NMEA sentence is output immediately after the first fix, then NMEA sentence is output with the specified period again from that point.

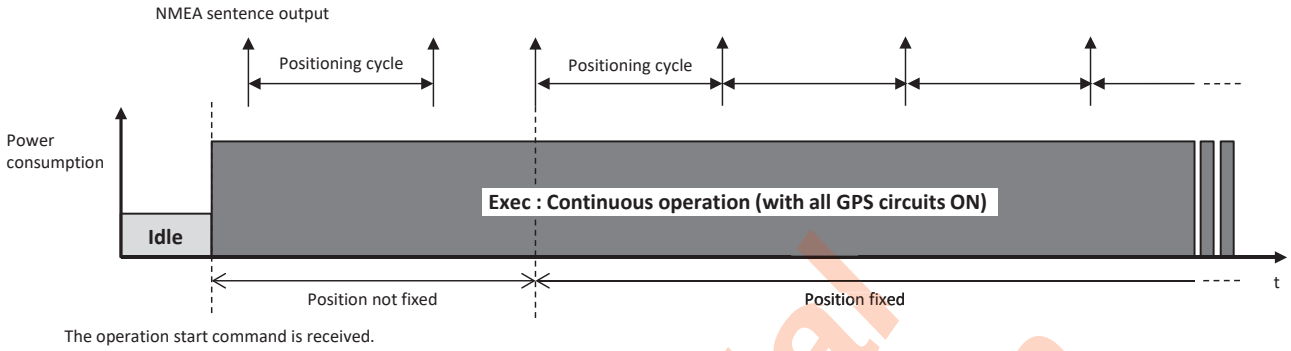


Fig.7 The operation sequence of Normal mode

When the Normal mode is selected and the other than "0" is set to sleep time of @GSOP, the CXD5603GF works intermittently. The CXD5603GF enters the Sleep state in specified time periodically and works at the Exec state in the rest of the time.

When the time subtracting sleep time from positioning cycle (that is operating time) is under 60sec, the CXD5603GF continues to work at the Exec state until position fixed as shown in Fig.8. When position is not fixed in the middle of operation, the CXD5603GF also continues to work at the Exec state until position fixed.

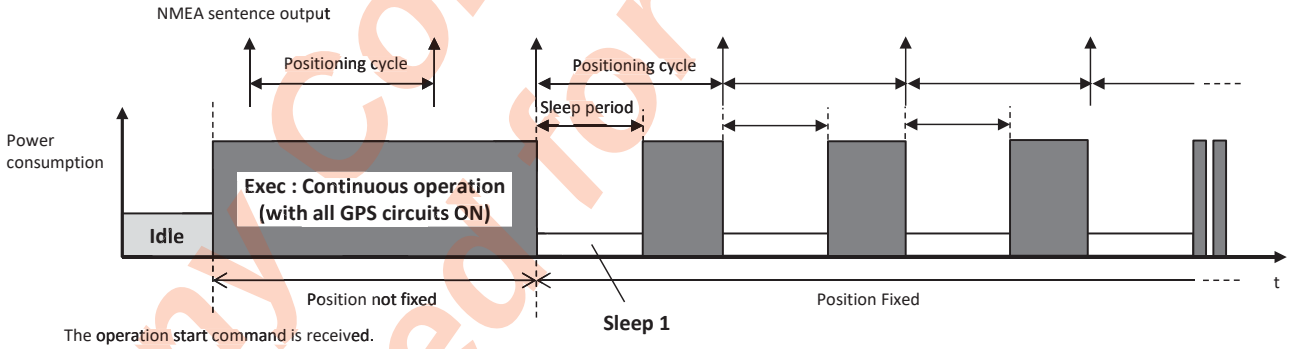


Fig.8 The operation sequence of Normal mode with sleep (operating time is under 60sec)

On the other hand, when the operating time is equal or more than 60sec, the CXD5603GF works by alternating between Sleep and Exec in the specified period as shown in Fig.9.

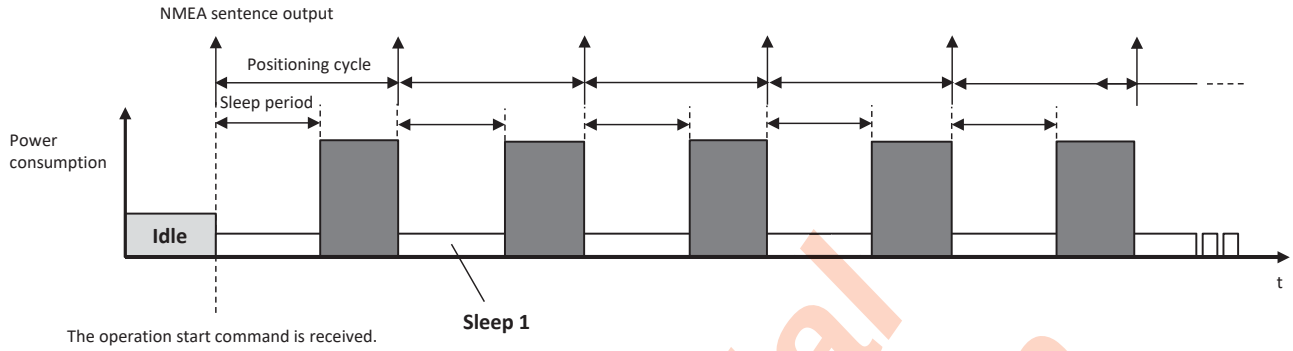


Fig.9 The operation sequence of Normal mode with sleep (operating time is equal or more than 60sec)

5.2.2 Low power mode

The CXD5603GF works intermittently and achieves low power consumption.

At the beginning, the CXD5603GF works at the Exec state continuously for acquisition the satellites and positioning. The CXD5603GF works with some GPS circuit activated / not- activated depending on the conditions of positioning and receiving signals in this period. When the position is fixed and the condition of receiving signals reaches a certain level, the CXD5603GF transits to the intermittent operation and achieves low power consumption.

The actual operation of intermittent operation varies according to the positioning cycle. When the positioning cycle is under 30sec, the CXD5603GF works by alternating between Exec (500ms) and Sleep (500ms) with a period of 1sec as shown in **Error! Reference source not found.**. Some GPS circuits are not activated during the Exec state of this intermittent operation. The interval of NMEA sentence output is the positioning cycle period as specified by @GSOP.

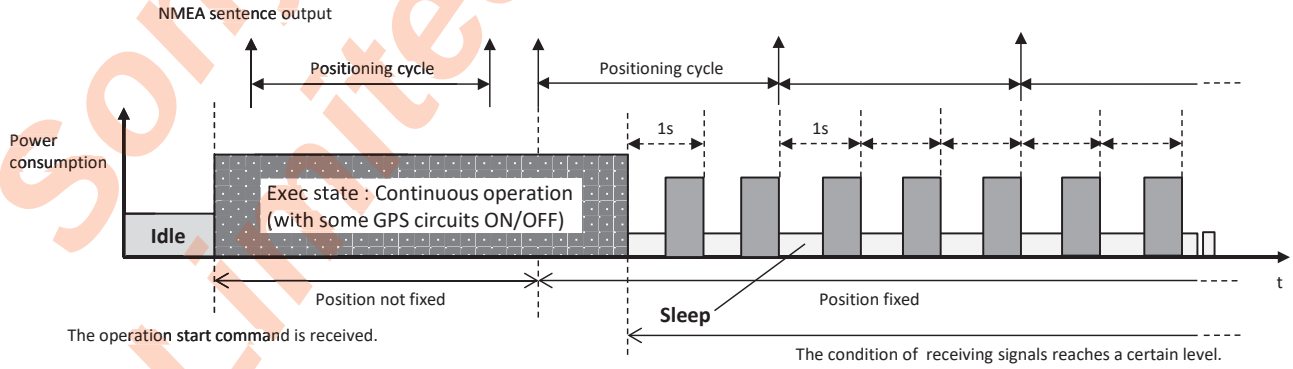


Fig.10 The operation sequence of Low Power mode (positioning cycle is under 30sec)

When the positioning cycle is equal or more than 30sec, the CXD5603GF works by alternating between Sleep (the positioning cycle – 15sec) and Exec (15sec) with a period specified by @GSOP as shown in **Error!**

Reference source not found.. When the condition of receiving signals reaches a certain level for the first time, the CXD5603GF transits to the intermittent operation by alternating between Sleep (500ms) and Exec (500ms) until the next period of positioning cycle

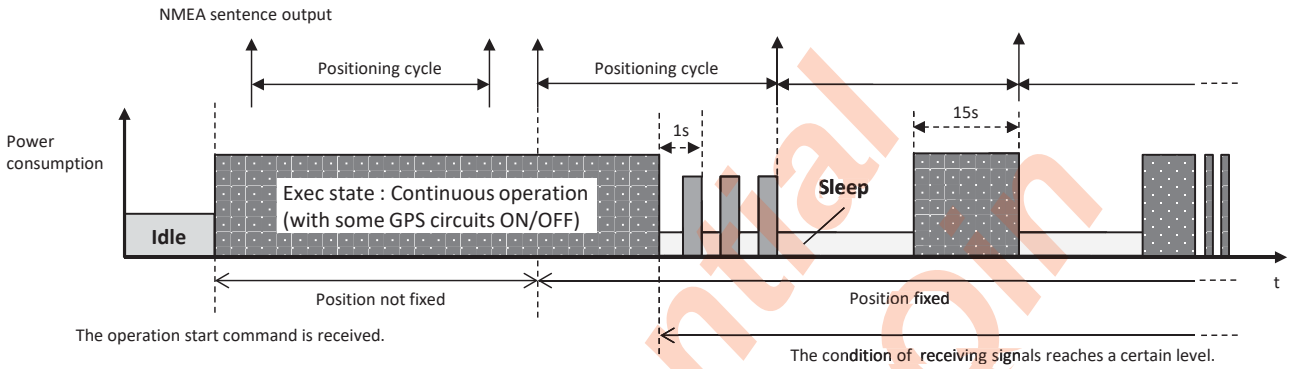


Fig.11 The operation sequence of Low Power mode (positioning cycle is equal or more than 30sec)

When position is not fixed in the middle of the intermittent operation, the CXD5603GF works at the Exec state and tries positioning a certain period of time to avoid failing positioning forever.

In the case of the positioning cycle being under 30sec, when the position if fixed and the condition of receiving signals reaches a certain level, the CXD5603GF transits to the intermittent operation as shown in **Error! Reference source not found.**.

In the case of the positioning cycle being equal or more than 30sec, when the position is fixed and the condition of receiving signals reaches a certain level, the CXD5603GF transits to the intermittent operation with changing Sleep (500ms) / Exec (500ms) until next period of positioning cycle and transit to the usual intermittent operation after that as shown in **Error! Reference source not found.**.

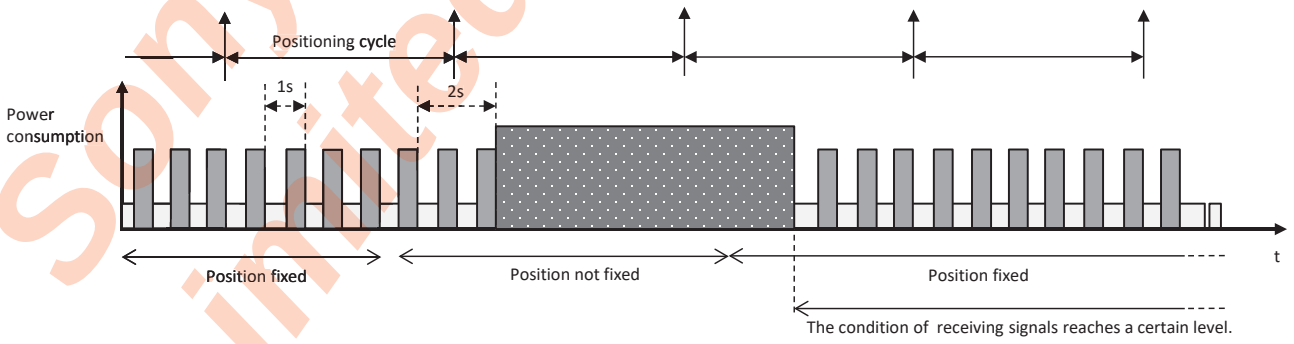


Fig.12 The case of position not fixed in the middle of Low Power mode (positioning cycle is under 30sec)

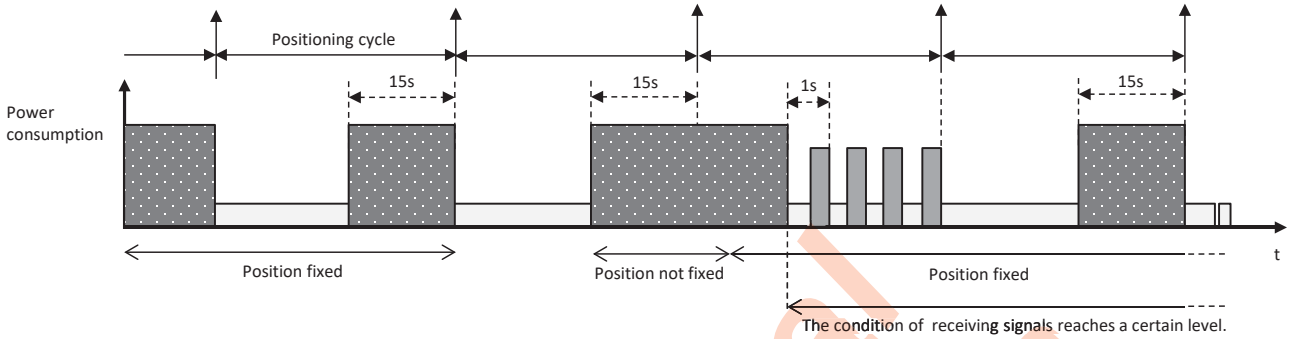


Fig.13 The case of position not fixed in the middle of Low Power mode (positioning cycle is equal or more than 30sec)

Only in the case of the positioning cycle being under 30sec, when the condition of receiving signals becomes bad, the CXD5603GF works at the Exec state a certain period of time even if the position is fixed. The CXD5603GF continues to work at the Exec state until the condition of receiving signals reaches a certain level.

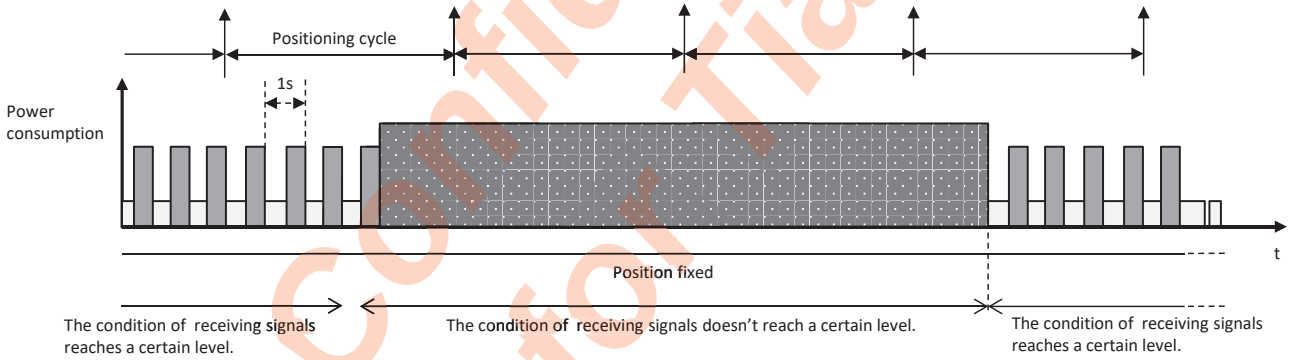


Fig.14 The case of the condition of receiving signals becomes bad (positioning cycle is under 30sec)

6 Firmware updating function

The firmware stored in the external flash memory of the CXD5603GF can be rewritten by issuing commands from the host controller. The data rewriting method is described below.

The sequence is different in UART and I^2C.

6.1 Firmware update sequence with UART

The firmware is updated according to the following sequence.

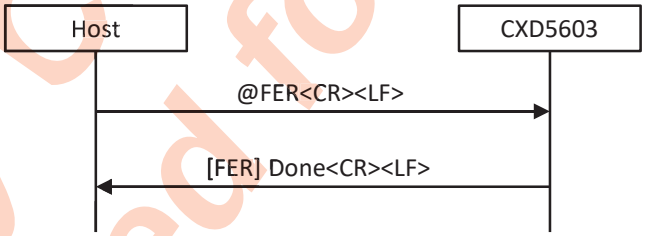
① The flash memory is erased.

② The firmware data is injected.

Each step is described below in detail.

6.1.1 The flash memory is erased.

The flash memory is erased before the firmware is downloaded. Use the normal control command (@FER) to erase the flash memory (The command protocol is the same as that for normal control commands) .



After completing @FER, reset start is initiated for the CXD5603GF automatically. When the reset start is initiated with the program on the flash memory is erased, the CXD5603GF is booted in the special mode and sends below message (command prompt) .

```
H>
```

In this mode, the CXD5603GF receives only the special commands for the firmware update.

6.1.2 The firmware data is injected.

When the CXD5603GF is booted in the special mode, the host controller can inject the firmware using the special

commands.

The firmware consists of below 4 files.

- <filename>.ebin1
- <filename>.ebin2
- <filename>.ebin3
- <filename>.ebin4

There are 3 commands in the special mode, “LOADH”, “LOADC” and “EXEC”. These command must be sent in ASCII character terminated by <CR> after recognized the CXD5603GF sent the command prompt (“H>”) . The CXD5603GF sends the execution result in ASCII character terminated by <CR> after receiving and executing these commands.

When inject the firmware file after “LOADH” and “LOADC”, there is no need to add header and footer. The firmware file can be sent as it is. The firmware must be injected 10ms after “LOADH” and “LOADC” sent.

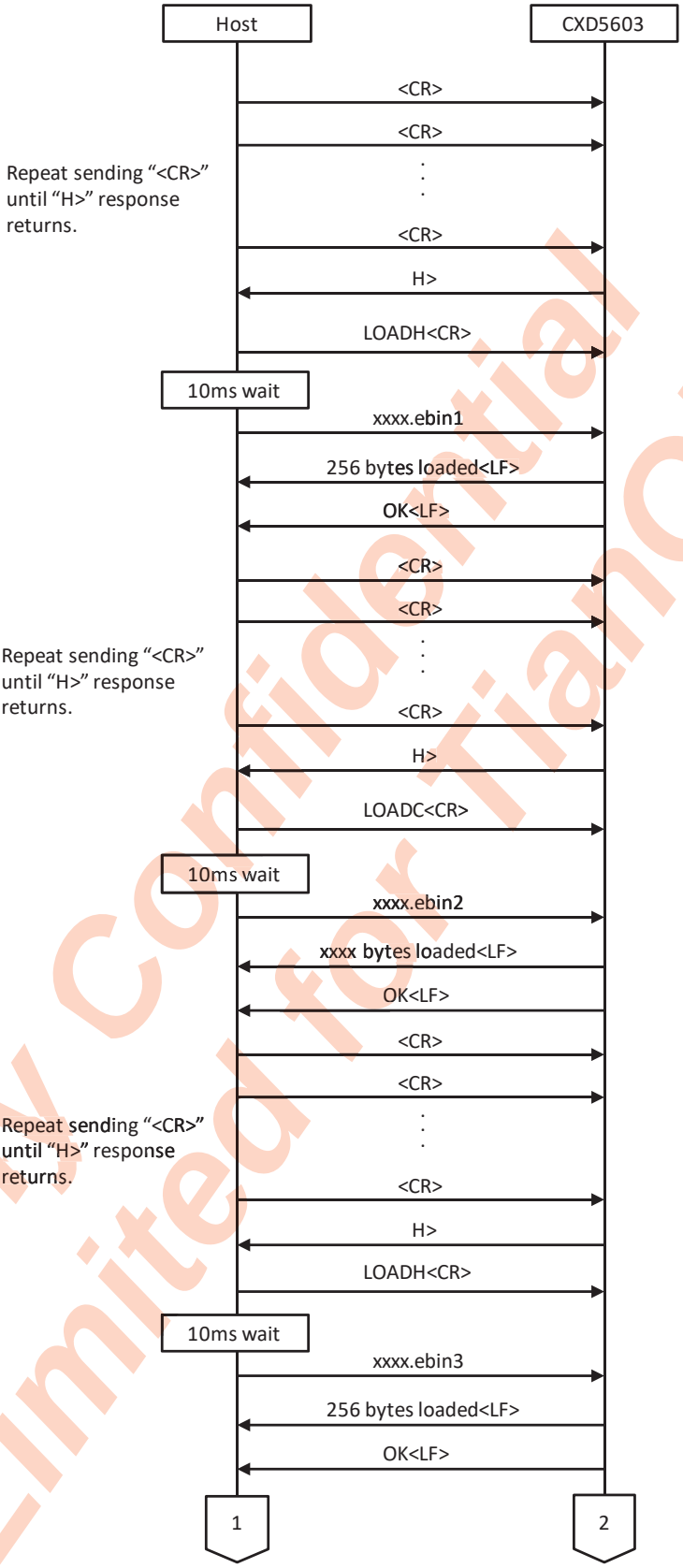


Fig.15 The sequence of the firmware injection (1/2)

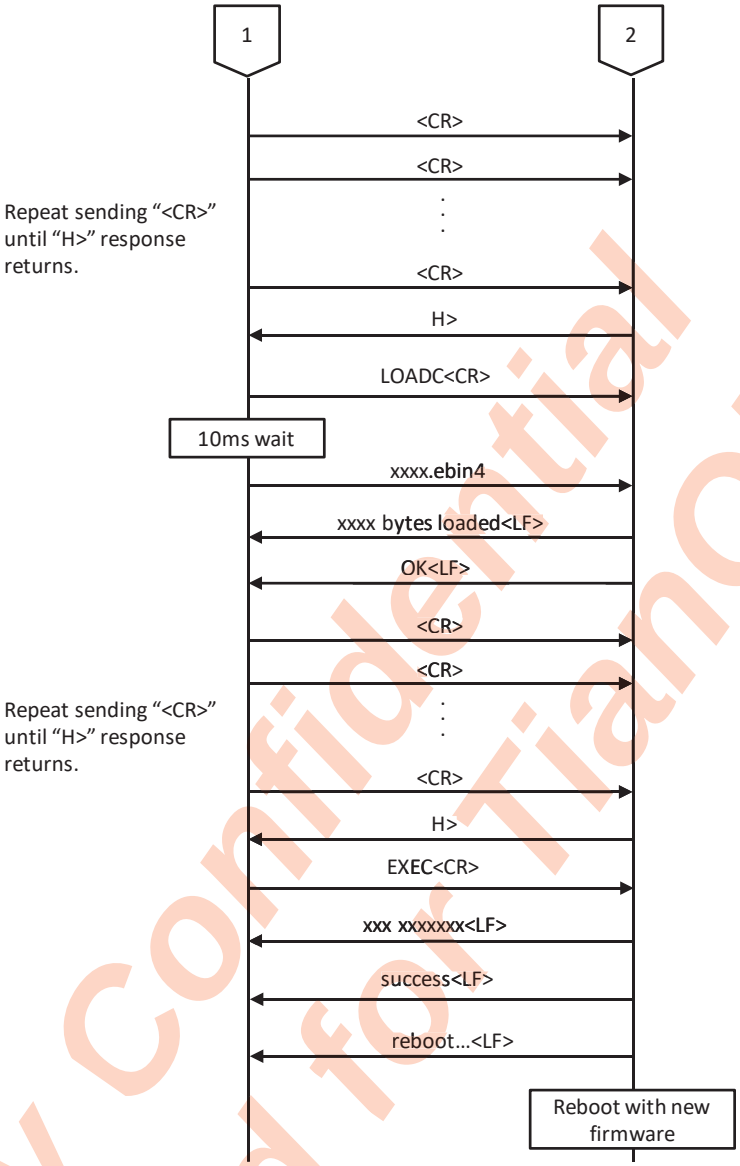


Fig.16 The sequence of the firmware injection (2/2)

When the firmware injection is completed normally, the string "reboot<CR>" is sent and the CXD5603GF reboot with injected firmware.

If other than "OK<CR>" returned for LOADH/LOADC or other than "success<CR>" returned for EXEC, start over from 6.1.2.

6.2 Firmware update sequence with I²C

The firmware is updated according to the following sequence.

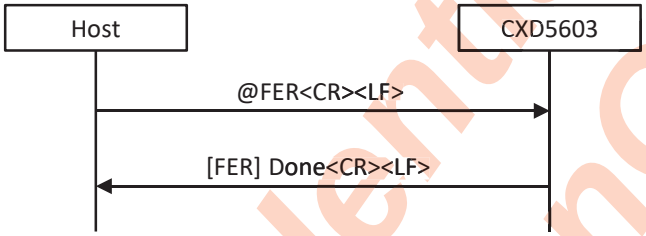
① The flash memory is erased.

② The firmware data is injected.

Each step is described below in detail.

6.2.1 The flash memory is erased.

The flash memory is erased before the firmware is downloaded. Use the normal control command (@FER) to erase the flash memory (The command protocol is the same as that for normal control commands) .



After completing @FER, reset start is initiated for the CXD5603GF automatically. When the reset start is initiated with the program on the flash memory is erased, the CXD5603GF is booted in the special mode and it receives just the special commands for the firmware update.

6.2.2 The special commands for the firmware update

The special commands are used for the firmware injection. Note that the format of these commands are completely different from the normal control commands.

Buffer status check

The CXD5603GF has 4 buffers for receiving the firmware (buffer 0 – 3) . It is necessary to check which buffer is available before sending the firmware data. This command is used for this purpose.

If the CXD5603GF returns “buffer status check is not available”, repeat sending this command and wait for the status changing to “buffer status check is available”.

Write packet (1 byte) :

Offset [bytes]	Length [bytes]	Description
0	1	Command type : 0x03 (Buffer status check)

Read packet (5 bytes) :

Offset [bytes]	Length [bytes]	Description
0	1	0 : Buffer status check is available Other than 0 : Buffer status check is not available
1	4	The status of each buffer (0 : Not available, 1 : Available) bit 0 : Buffer 0 bit 1 : Buffer 1 bit 2 : Buffer 2 bit 3 : Buffer 3

Update command sending

This command is used for sending the update commands. The update commands are used for sending the firmware data and writing sent firmware to the flash memory.

Data send commands (LOADH/LOADC1/LOADC2) are used for sending the firmware data. Use different command depending on the firmware data.

Firmware write command (EXEC) is used for writing the sent firmware to the flash memory.

Update command sending command uses buffer 0 or buffer 1 and it is necessary to check buffer 0 or buffer 1 is available by buffer status check command before sending this commands. Buffer 0 and buffer 1 must be used one after the other. Another buffer from the buffer just used before must be used in whole of sequence.

The transferred data cannot be divided into smaller size than 256 bytes except the last fractional data. So, the host controller must be able to send 263 bytes data (transferred data: 256 bytes, the other data: 7 bytes) in 1 packet.

Write packet (variable length, maximum 263 bytes) :

Offset [bytes]	Length [bytes]	Description
0	1	Command type : 0xA0 (Update command sending, using buffer 0) 0xA1 (Update command sending, using buffer 1)
2	1	Lower byte of (transfer data length+3) If there is no data to send (the case of “EXEC”), set 0x03.
1	1	Upper byte of (transfer data length+3) If there is no data to send (the case of “EXEC”), set 0x00.
3	1	Set 0x42 always.
4	1	Sequence number. Increment every time update command is sent. Initial value is 0x00. When it overflows, set 0x00 again. This value must not be initialized until the entire of the firmware update sequence finished.
5	1	Update command type (see firmware update sequence for detail) : 0x02 : LOADH 0x04 : LOADC1 0x05 : LOADC2 0x06 : EXEC
6	0～256	Transferred data (firmware data) . 256 bytes data are sent per command. The fractional part is sent at last command. When update command type is “EXEC”, no data is transferred (0 byte) .
Last byte	1	Set 0x00 always.

Command execution status check

This command is used for checking the status of command execution.

This command uses buffer 3 and it is necessary to check buffer 3 is available by buffer status check command before sending this command.

If the CXD5603GF returns “command execution status check is not available”, repeat sending this command and wait for the status changing to “command execution status check is available”.

Write packet (1 byte) :

Offset [bytes]	Length [bytes]	Description
0	1	Command type : 0xA3 (Command execution status check)
1	1	Set 0x08 always.
2	1	Set 0x00 always.

Read packet (9 bytes) :

Offset [bytes]	Length [bytes]	Description
0	1	0 : Command execution status check is available Other than 0 : Command execution status check is not available
1	1	Fixed value (0x49) .
2	1	Fixed value (0x32) .
3	1	Sequence number of received command.
4	1	Update command type.
5	2	Buffer number.
7	2	The status of command execution : 0x0400 : Normal end Other than 0x0400 : Abnormal end

Command execution result check

This command is used for getting the message that shows the result of command execution. The form of the message is ASCII string.

This command uses buffer 2 and it is necessary to check buffer 2 is available by buffer status check command before sending this command.

If the CXD5603GF returns “command execution result check is not available”, repeat sending this command and wait for the status changing to “command execution result check is available”.

Write packet (1 byte) :

Offset [bytes]	Length [bytes]	Description
0	1	Command type : 0xA2 (Command execution result check)
1	1	Set 0x20 always.
2	1	Set 0x00 always.

Read packet (33 bytes) :

Offset [bytes]	Length [bytes]	Description
0	1	0 : Command execution result check is available Other than 0 : Command execution result check is not available
1	1	Fixed value (0x49) .
2	1	Fixed value (0x32) .
3	1	Sequence number of received command.
4	1	Update command type.
5	1	Buffer number.
7	2	Not used.
9	1	Size of the command execution result message.
10	variable	Command execution result message (ASCII string) .
To 32	Variable	Not used.

The command execution result messages are ASCII strings. Every message is terminated by the linefeed code “<CR>”. When other than normal end message returns, it is necessary to reset the CXD5603GF and restart the firmware update procedure from the beginning.

The command execution result messages are listed below.

Message	Description	Executed oprration	備考
“Header OK<CR>”	Data transfer succeeded.	Writing ebin1or ebin3	
“Header NG<CR>”	Data transfer not succeeded.	Writing ebin1 or ebin3	
“****** bytes loaded<CR>”	Data transfer succeeded.	Writing ebin 2 or ebin 4	Received data size is stored in “******”*
“Code NG<CR>”	Data transfer not succeeded.	Writing ebin 2 or ebin 4	
“Exec 0x****<CR>”	Flash programming succeeded.	EXEC	“XECgris internal information. Don in sngr

6.2.3 The firmware data is injected.

When the CXD5603GF is booted in the special mode, the host controller can inject the firmware using the special

commands.

The firmware consists of below 4 files.

- <filename>.ebin1
- <filename>.ebin2
- <filename>.ebin3
- <filename>.ebin4

The setting of I^2C interface are listed below. These setting cannot be changed.

- Clock frequency : 400kHz
- Address length : 7 bits
- Slave address : 0x24

The firmware update sequence is shown as below. Don't spare over 10s between each command, or timeout occurs.

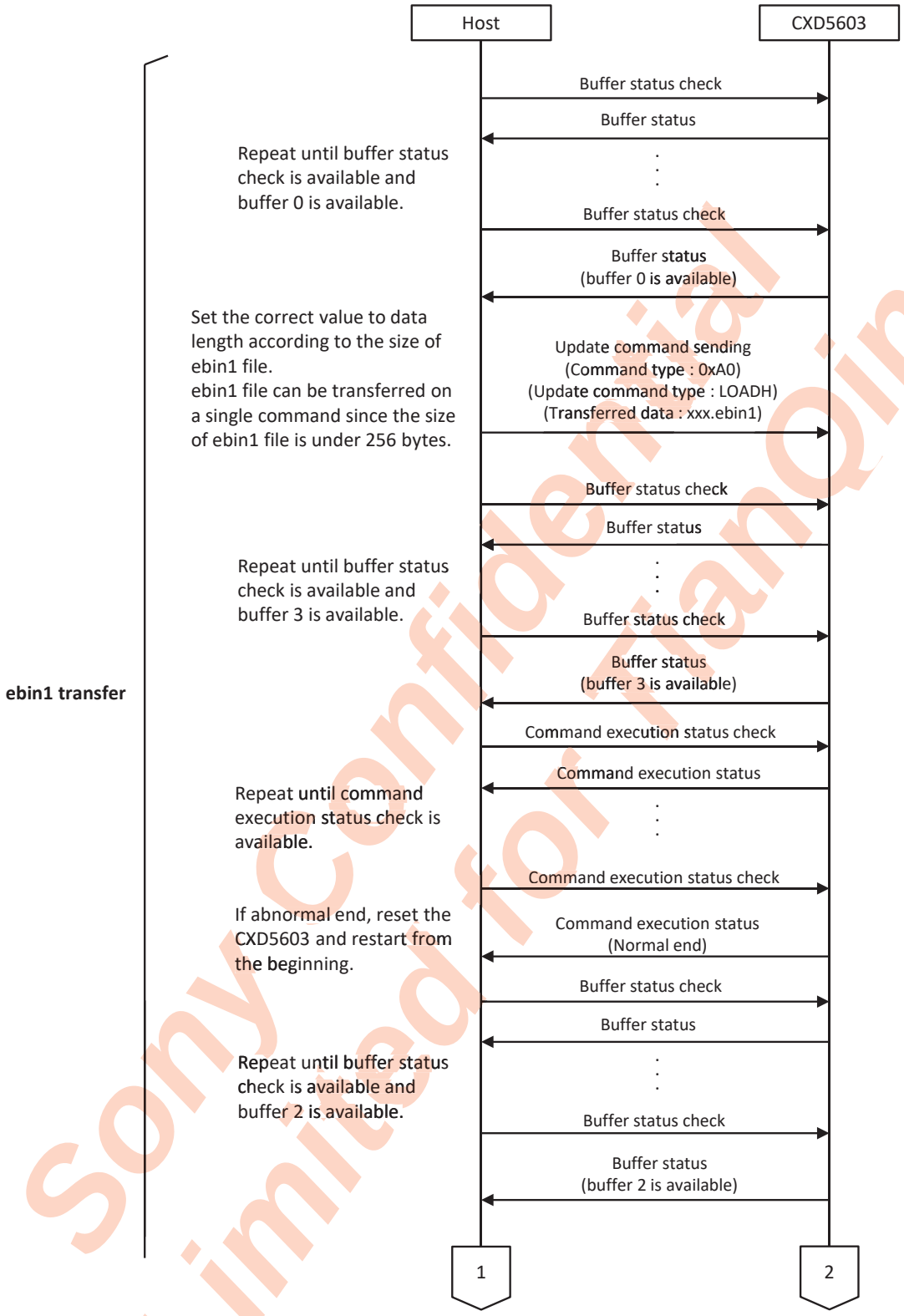


Fig.17 The sequence of the firmware injection (1/5)

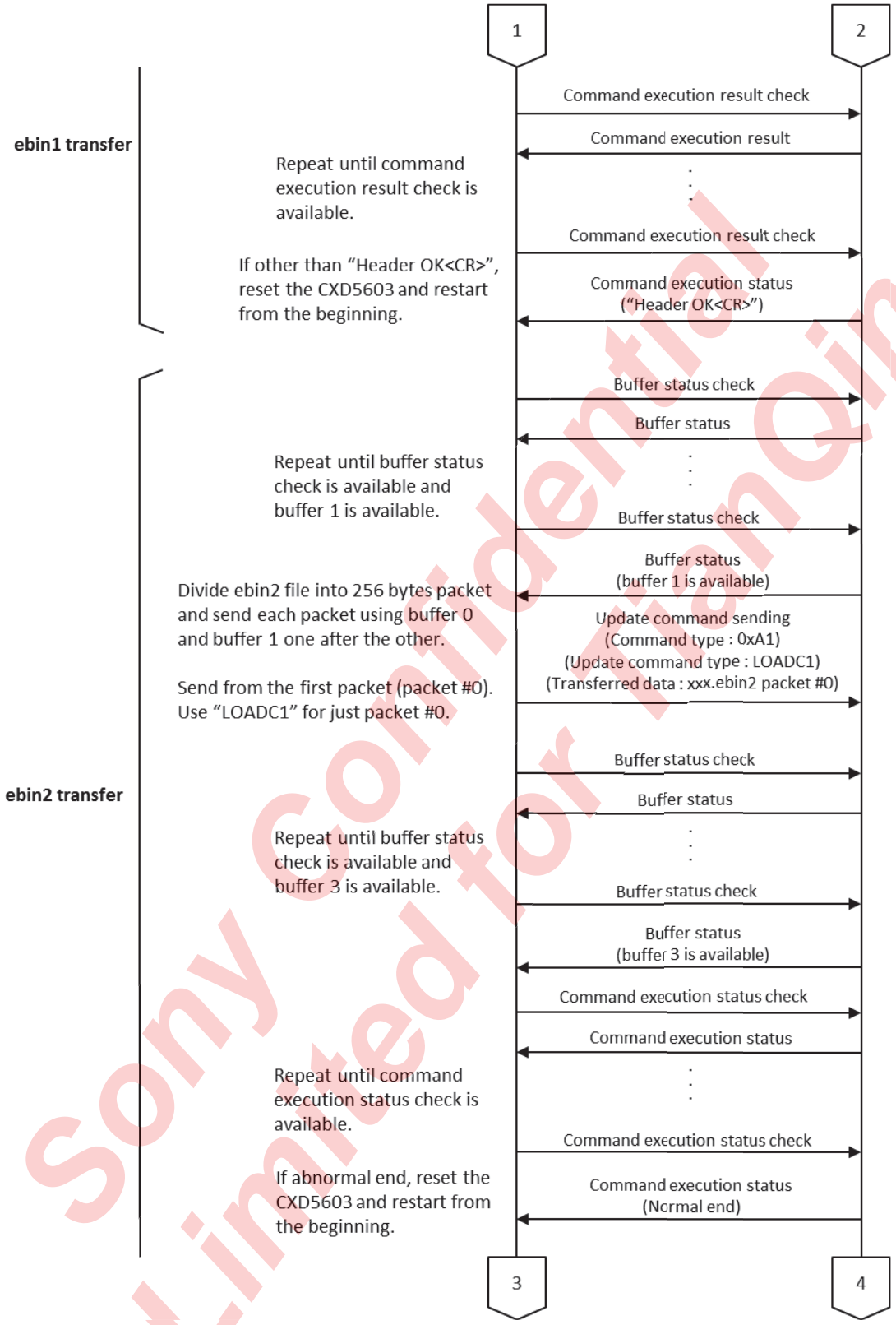


Fig.18 The sequence of the firmware injection (2/5)

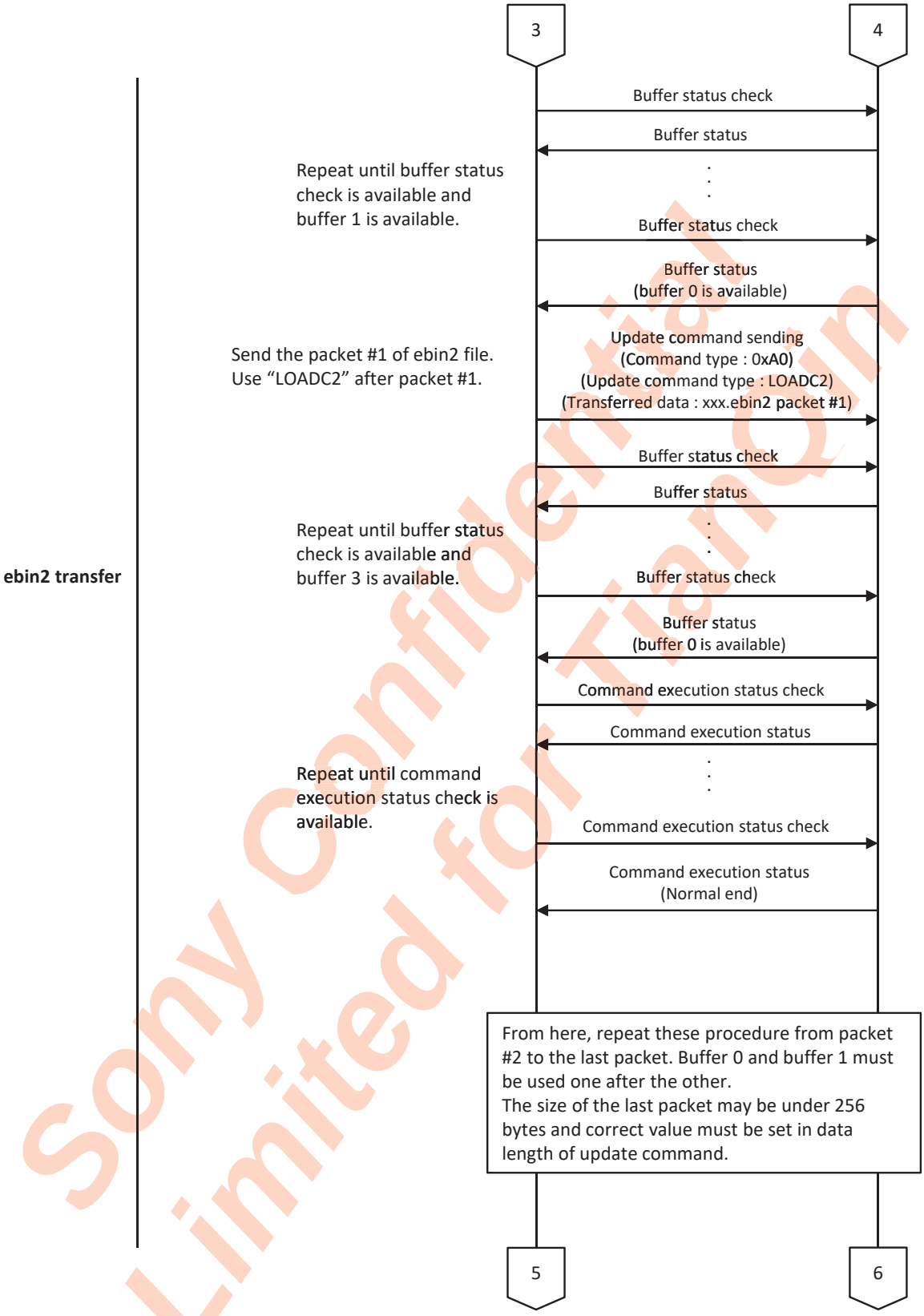


Fig.19 The sequence of the firmware injection (3/5)

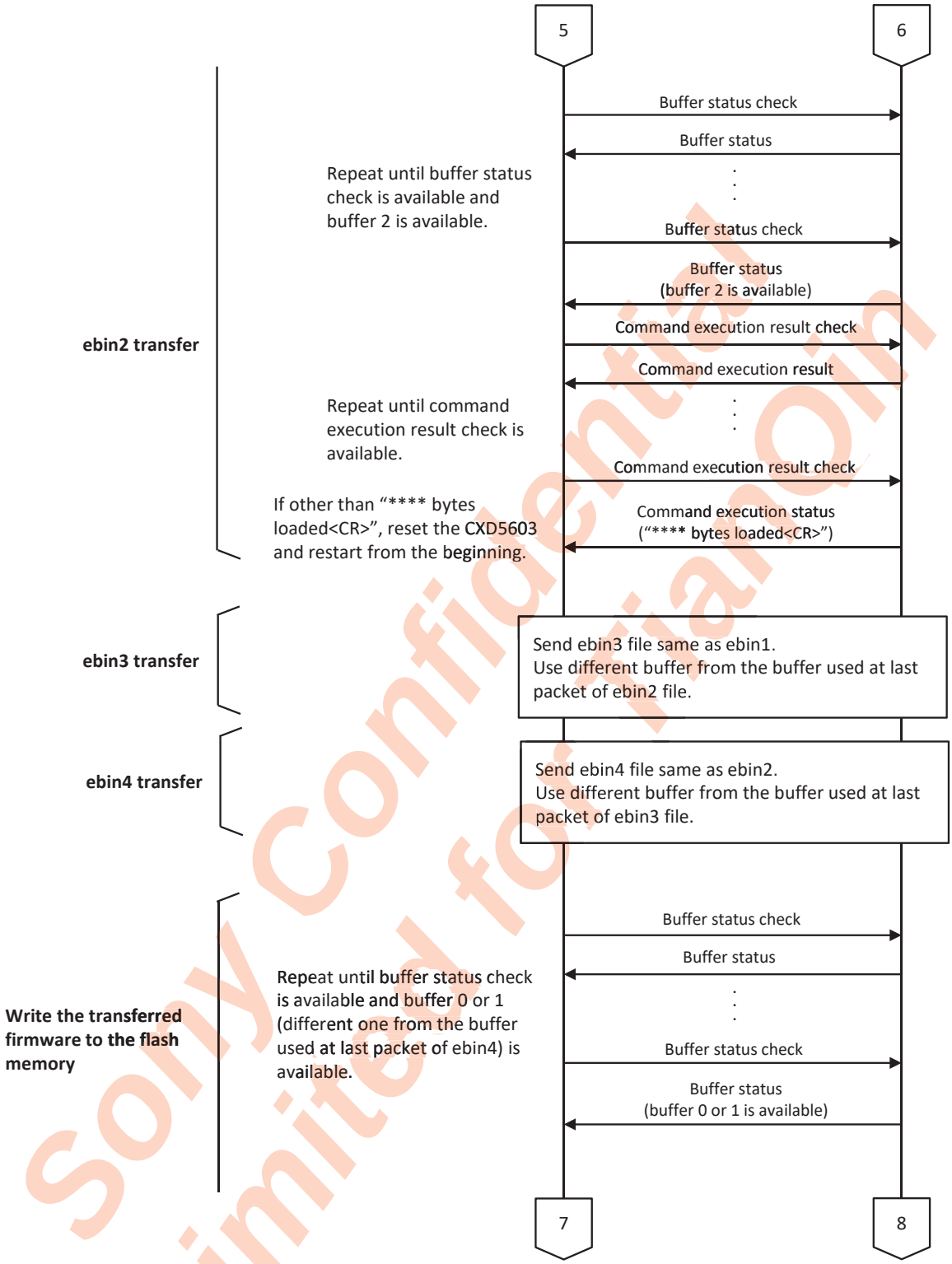


Fig.20 The sequence of the firmware injection (4/5)

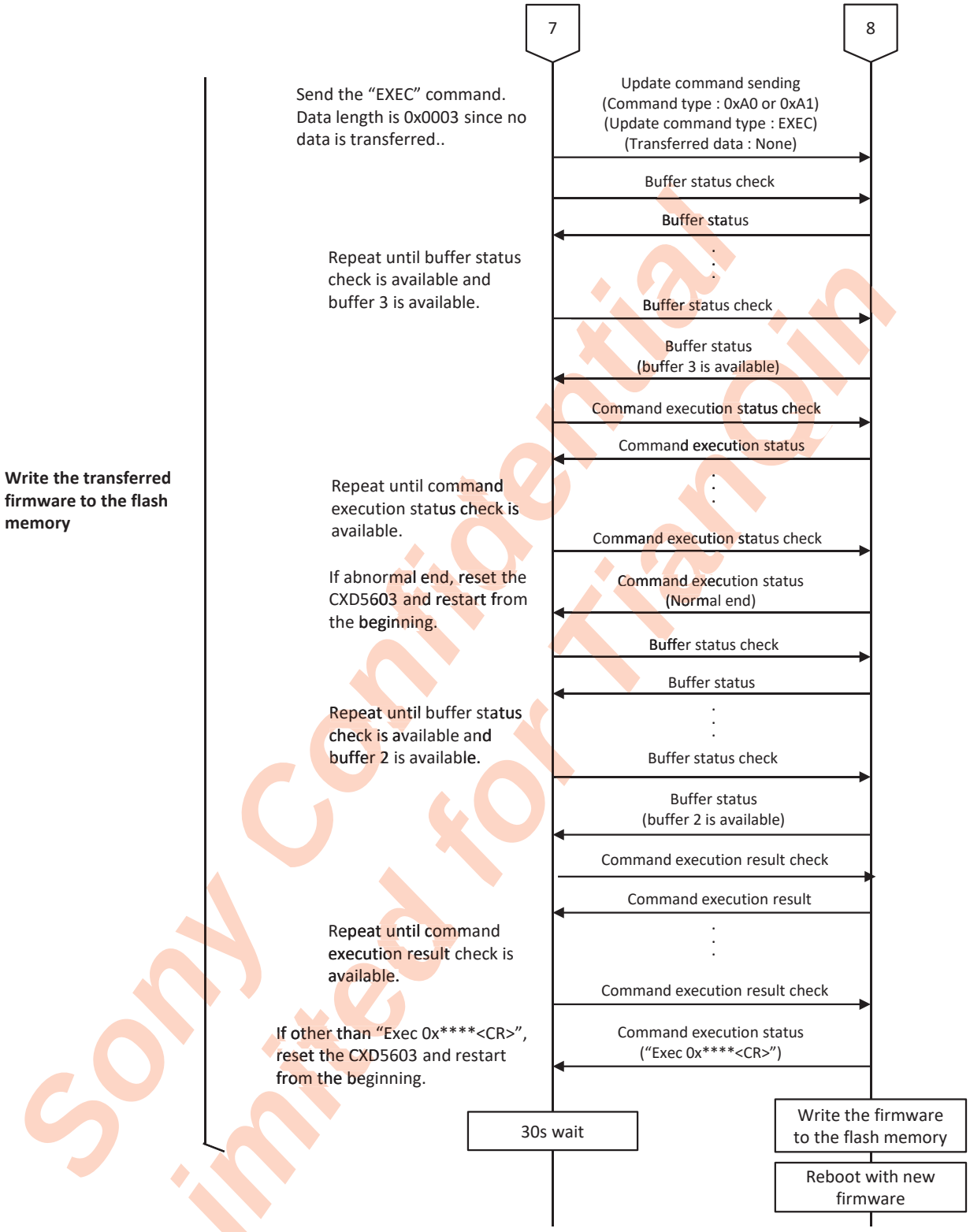


Fig.21 The sequence of the firmware injection (5/5)

6.3 Restoration of the CXD5603GF in an unusual state

The CXD5603GF boots in special mode by reset the CXD5603GF with setting BOOT_REC pin to “H” and it is possible to write the firmware again. If the invalid firmware is written in the flash memory, recover with this method.

7 Flash-less boot function

The CXD5603GF can work without external flash memory by the host controller injecting the firmware at booting. This operation is called flash-less boot function.

7.1 The Constraints in the flash-less boot

There are some constraints in the functionality when the flash-less boot is used. The constraints are listed below.

Functionality	Normal use	Flash-less boot
Host interface	UART or I^2C	UART only
@ABUP command	Available	Not available (automatic backup function is always disabled)
@BUP command	Available	Not available
@SLP command	Sleep 0, Sleep 1, Sleep 2 can be selected.	Only Sleep 0 can be selected.
Normal mode	Sleep period can be specified.	Sleep period cannot be specified (always “0”) .
Low Power mode	Available	Not available

7.2 The setting of UART interface

The UART setting for UART is listed below.

- Baud rate : 2M bps (115,200 bps at beginning)
- Data length : 8 bits
- Parity : None
- Stop bit : 2 bits (1 bit when baud rate is 115,200 bps)
- Flow control : Hardware (CTS/RTS)

When the firmware injection finished, UART setting will change as written in 2.1.

7.3 The boot sequence

When the CXD5603GF boots without external flash memory, the loader and the firmware can be injected using the special commands.

The loader consists of below 2 files.

- cxd5603_loader*******.ebin1
- cxd5603_loader*******.ebin2

The firmware consists of below 4 files.

- cxd5603_fl_<version number>.ebin1
- cxd5603_fl_<version number>.ebin2
- cxd5603_fl_<version number>.ebin3
- cxd5603_fl_<version number>.ebin4

This means 6 files must be injected from the host controller when using flash-less boot function.
The baud rate of UART must be 115,200 bps at injecting the loader and 2M bps at injecting the firmware.
There are 3 commands in the special mode, “LOADH”, “LOADC” and “EXEC”. These command must be sent in ASCII character terminated by <CR> after recognized the CXD5603GF sent the command prompt (“H>”) . The CXD5603GF sends the execution result in ASCII character terminated by <CR> after receiving and executing these commands.

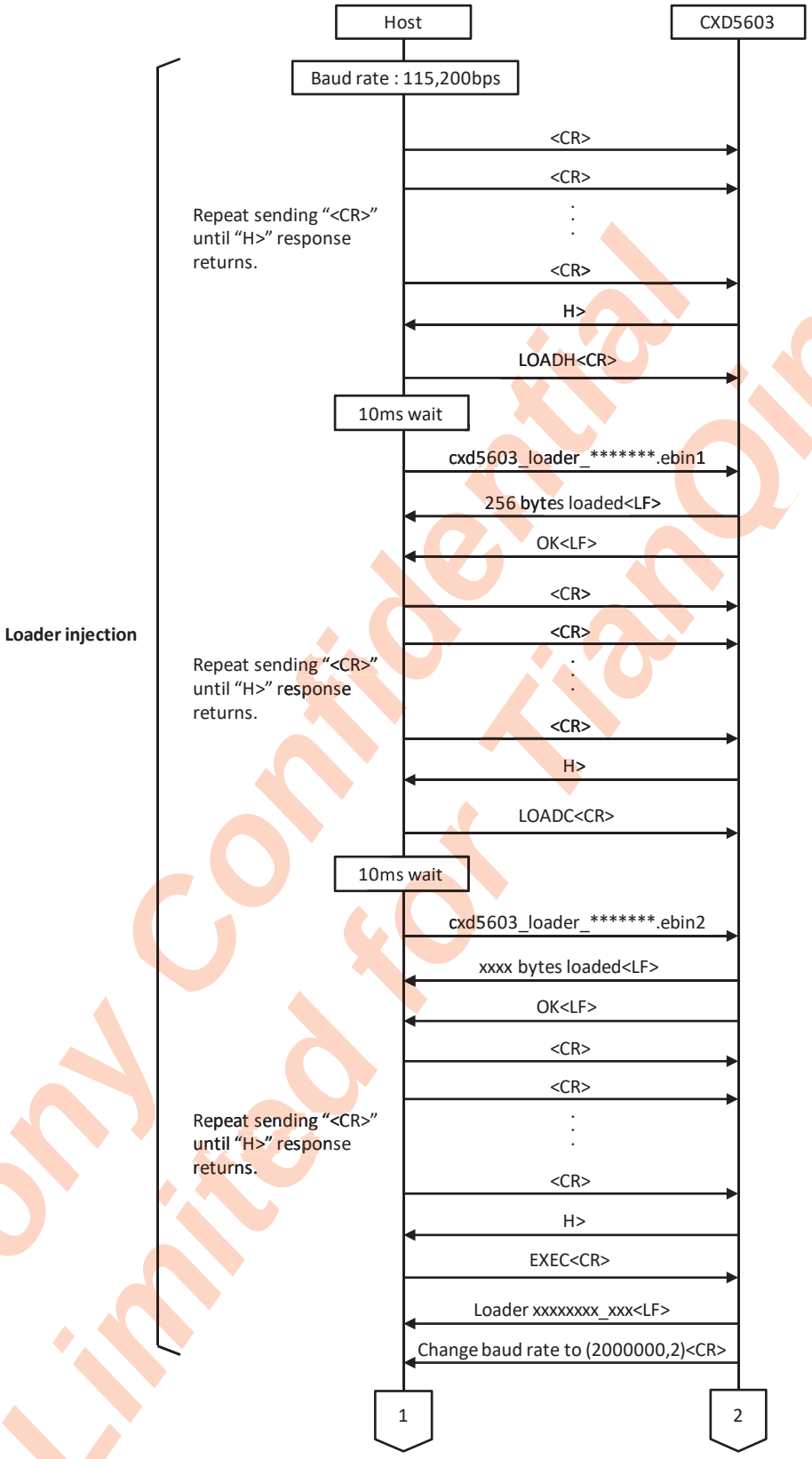


Fig.22 The sequence of the flash-less boot (1/3)

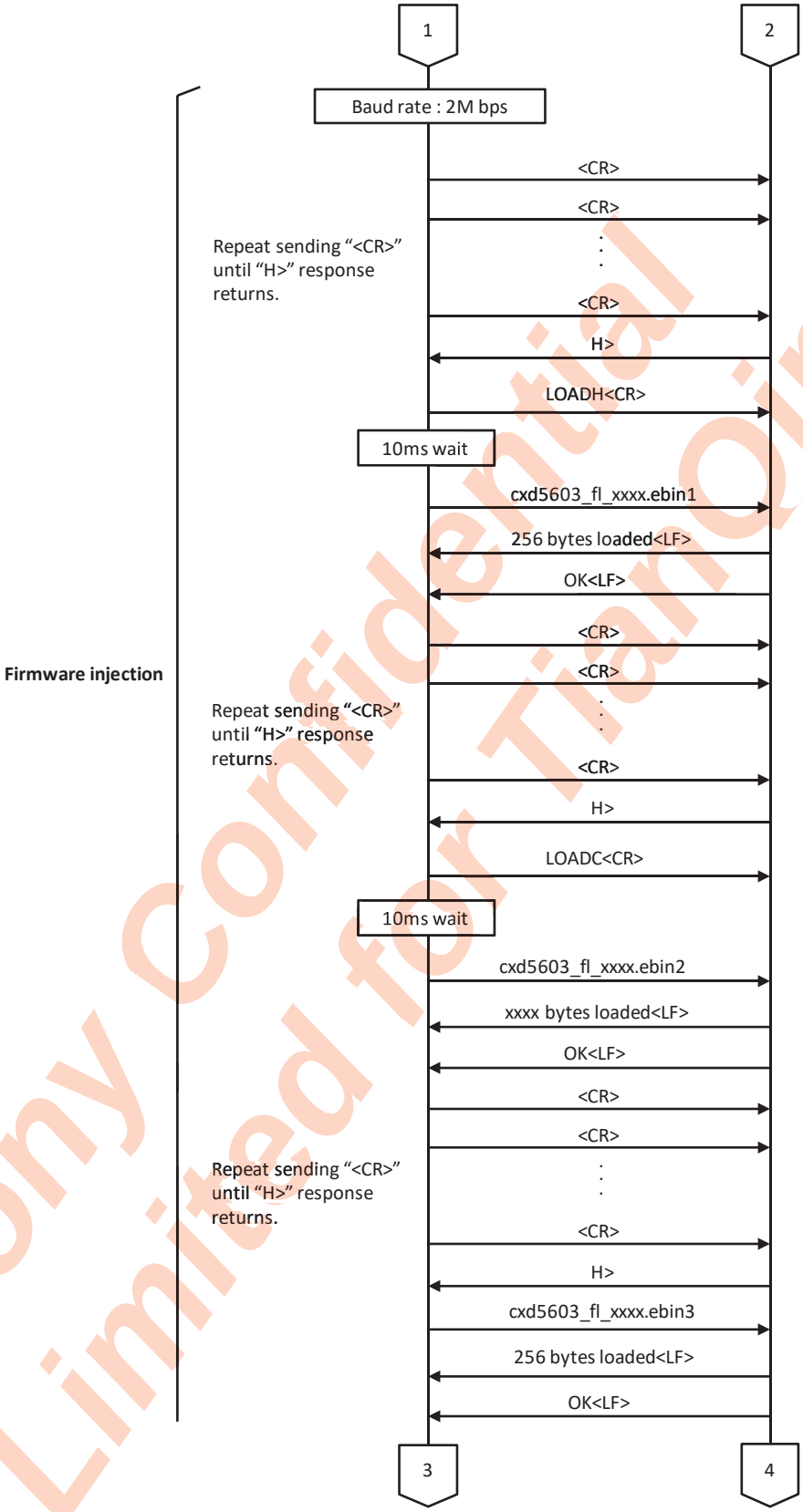


Fig.23 The sequence of the flash-less boot (2/3)

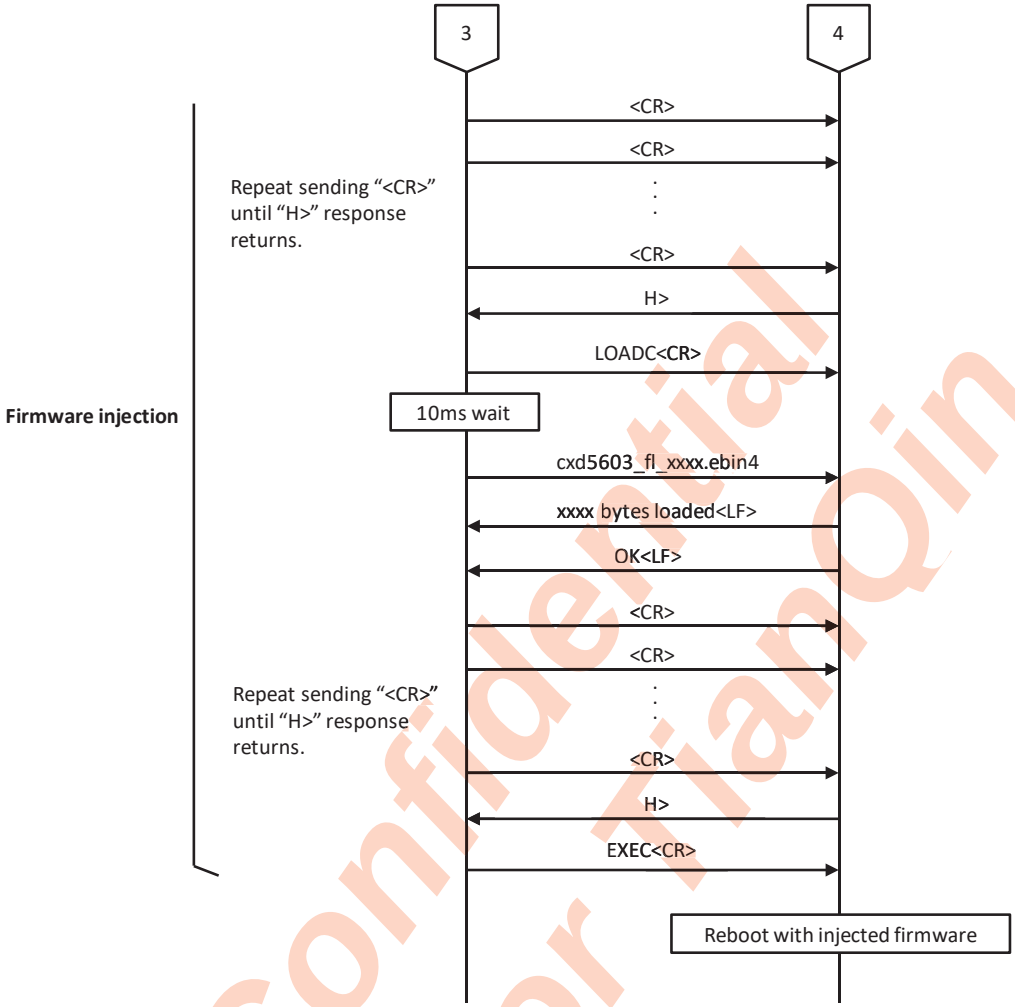


Fig.24 The sequence of the flash-less boot (3/3)

7.4 The handle at error occurred

When the CXD5603GF didn't return the expected response for the special commands ("LOADH", "LOADC" and "EXEC") , restart the sequence from the beginning of loader injection (when the error occurred in the middle of loader injection) or firmware injection (when the error occurred in the middle of firmware injection) . No reset is needed for restart.

8 Other operating specifications

8.1 GPS week number rollover

The GPS Week Number count began at the midnight of 05 January 1980 / 06 January 1980. Since then, the GPS week number has been incrementing steadily by one each week. Only the bottom 10 bits of the week number is transmitted. Hence, the GPS Week number field is given as a modulo 1024. This means that at the completion of week 1023, the GPS week number rolled over to 0 for the first time on midnight GPS Time of the evening of 21 August 1999 / morning of 22 August 1999. Note that this corresponded to 23:59:47 UTC on 21 August 1999. The next rollover will happen in 2019.

The CXD5603GF solves the rollover issue by establishing its own origin date, week number, and rollover count. The origin week number acts as a reference rollover week number since all received week numbers must be at least as large as the CXD5603GF reference rollover week number. This reference rollover week number is hard-coded into the CXD5603GF firmware (e.g., Sunday, 1st 2009 – Week number 1513).

When the CXD5603GF fixes the position and time, it updates the stored origin date, week number and rollover count (i.e., establish a new origin date to be used with later fixes). Updated information can be stored in the flash memory by the @BUP command, and these are used from the next boot up. In addition, the @GTIM command can be used to inject the time and date, which causes the CXD5603GF to update the CXD5603GF stored origin date and week number.

9 Error codes

When the CXD5603GF responds with an error reply to a command issued by the host, an error code indicating the nature of the error is transmitted with the reply. This is a negative value or “0” which is a POSIX standard subset. The error codes are listed in the table below.

Value	Definition	Significance
0	0	Command processing successful
-1	-EPERM	Internal error
-2	-ENOENT	A command which is not supported has been input.
-3	-ESRCH	The internal communication cancel process has failed.
-4	-EINTR	Internal error
-5	-EIO	Flash ROM access or DMA processing has failed
-6	-ENXIO	Internal error
-7	-E2BIG	The injection data is smaller than the requested size.
-8	-ENOEXEC	Internal error
-9	-EBADF	Internal error
-11	-EAGAIN	Power-on has failed.
-12	-ENOMEM	Memory allocation has failed.
-13	-EACCES	Power control has failed.
-16	-EBUSY	Processing was not requested in the correct status.
-17	-EEXIST	Internal error
-19	-ENODEV	Internal error
-22	-EINVAL	The argument is outside the specified range.
-28	-ENOSPC	Internal error
-35	-ENOMSG	The message data type is incorrect.
-36	-EIDRM	Internal error
-46	-ENOLCK	Internal error
-47	-ECANCELED	Internal error
-48	-ENOTSUP	UART/I^{2}C control has failed.
-54	-EBADRQC	The command argument is not correct.
-62	-ETIME	Processing failed due to a timeout.
-71	-EPROTO	The data injection content is not correct.
-79	-EOVERFLOW	Internal error
-132	-ENOBUFS	Internal error
-143	-ESHUTDOWN	Internal error
-145	-ETIMEDOUT	The command failed due to a timeout.
-151	-ESTALE	Internal error

10 Requests and checkpoints regarding the use of this document

The ratings, etc. given in this document are subject to change without notice due to improvements, etc.

No authorization is given by this document for any approval to implement the industrial property rights relating to the document contents or for any guarantees with respect to any other rights.

In cases where examples of circuits have been provided in this document, these circuits are provided merely to show typical application examples for the purpose of reference for use. As such, Sony disclaims all liability for any damages caused by the use of these circuits.