RL78 Family

RL78 Microcontroller (RL78 Protocol D) Serial Programming Guide

Introduction

This application note describes the specifications of the boot firmware in RL78 microcontrollers. If the firmware is used in a way that does not conform with the descriptions in this document, correct operation is not guaranteed.

Target Devices

RL78 family

* Please refer to the following site for target devices that support RL78 protocol D.

https://en-support.renesas.com/knowledgeBase/16979203

Contents

1. Terminology

The following terms used in this application note are defined below.

1. Boot firmware
2. Flash memory
3. Device or microcontroller
4. Programmer or host
5. Flash memory programming mode
6. Flash options

1.1 Boot Firmware

"Boot firmware" refers to a program that is embedded in the device in advance for the purpose of controlling rewriting of the flash memory.

1.2 Flash Memory

Flash memory consists of "code flash memory", from which programs can be executed, and "data flash memory" for storing data.

1.3 Device or Microcontroller

The RL78 microcontroller is called the "device" or "microcontroller" in this document.

1.4 Programmer or Host

The tool used to rewrite the on-chip flash memory of the device, such as a flash memory programmer, is called the "programmer" or "host".

1.5 Flash Memory Programming Mode

The state in which the programmer is connected to the device and the boot firmware is activated is called "flash memory programming mode".

For details, refer to section 4.2, Communication Establishment Phase.

1.6 Flash Options

The security flags, and flash shield window are collectively called the "flash options".

2. System Architecture

An RL78 microcontroller has embedded boot firmware to control rewriting of its flash memory. The firmware controls rewriting of the contents of on-chip flash memory through the transmission and reception of commands between the programmer and RL78 microcontroller via serial communications.

In flash memory programming mode, control commands sent from the programmer are received through the communications interface. Flash memory is rewritten according to the received commands and the results are sent to the programmer.

2.1 Block Diagram

Figure 2-1 is a block diagram of the boot firmware in an RL78 microcontroller.

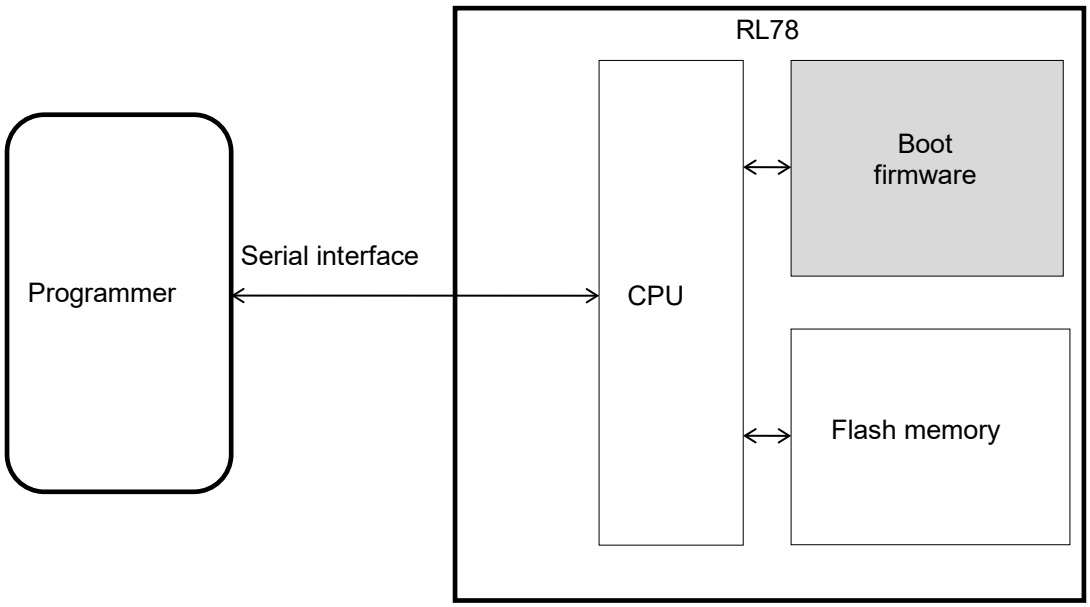


Figure 2-1 Block Diagram

3. Communication Modes

The boot firmware supports the following two communication modes.

1. Single-line UART communications
2. Dedicated UART communications

3.1 Single-Line UART Communications

The boot firmware supports single-line UART communications in which data are transmitted and received through a single line.

Figure 3-1 is a block diagram of single-line UART communications.

Table 3-1 lists the specifications of communications.

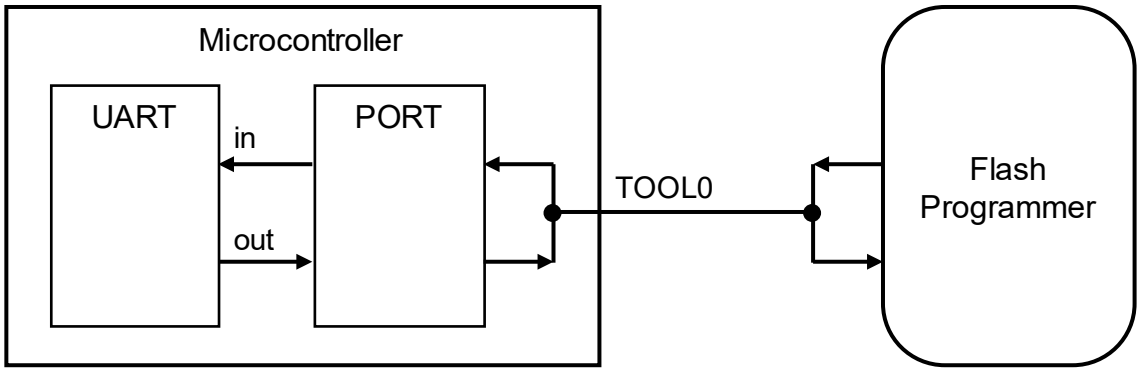


Figure 3-1 Single-Line UART Communications

Table 3-1 Specifications of Communications

Port	**TOOL0**	
Communication rates	115,200 bps 250,000 bps 500,000 bps 1,000,000 bps	Initial rate
Data length	8 bits	LSB-first transfer
Parity bit	No parity	
Stop bits	2 bits 1 bit	For transmission to the boot firmware For transmission from the boot firmware

3.2 Dedicated UART Communications

The boot firmware supports dedicated UART communications in which data are transmitted and received through two lines.

Figure 3-2 is a block diagram of dedicated UART communications.

Table 3-2 lists the specifications of communications.

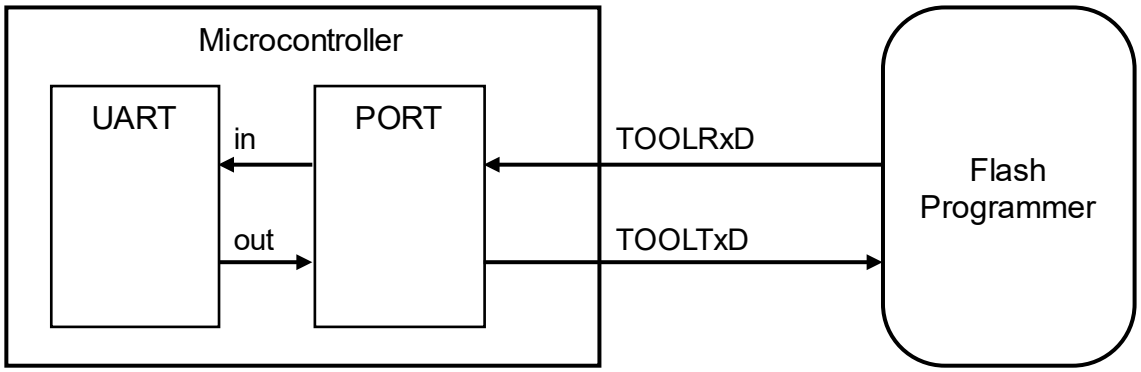


Figure 3-2 Dedicated UART Communications

Table 3-2 Specifications of Communications

Port	TOOLRxD (reception) TOOLTxD (transmission)
Communication rates	Same as for single-line UART communications
Data length	Same as for single-line UART communications
Parity bit	Same as for single-line UART communications
Stop bits	Same as for single-line UART communications

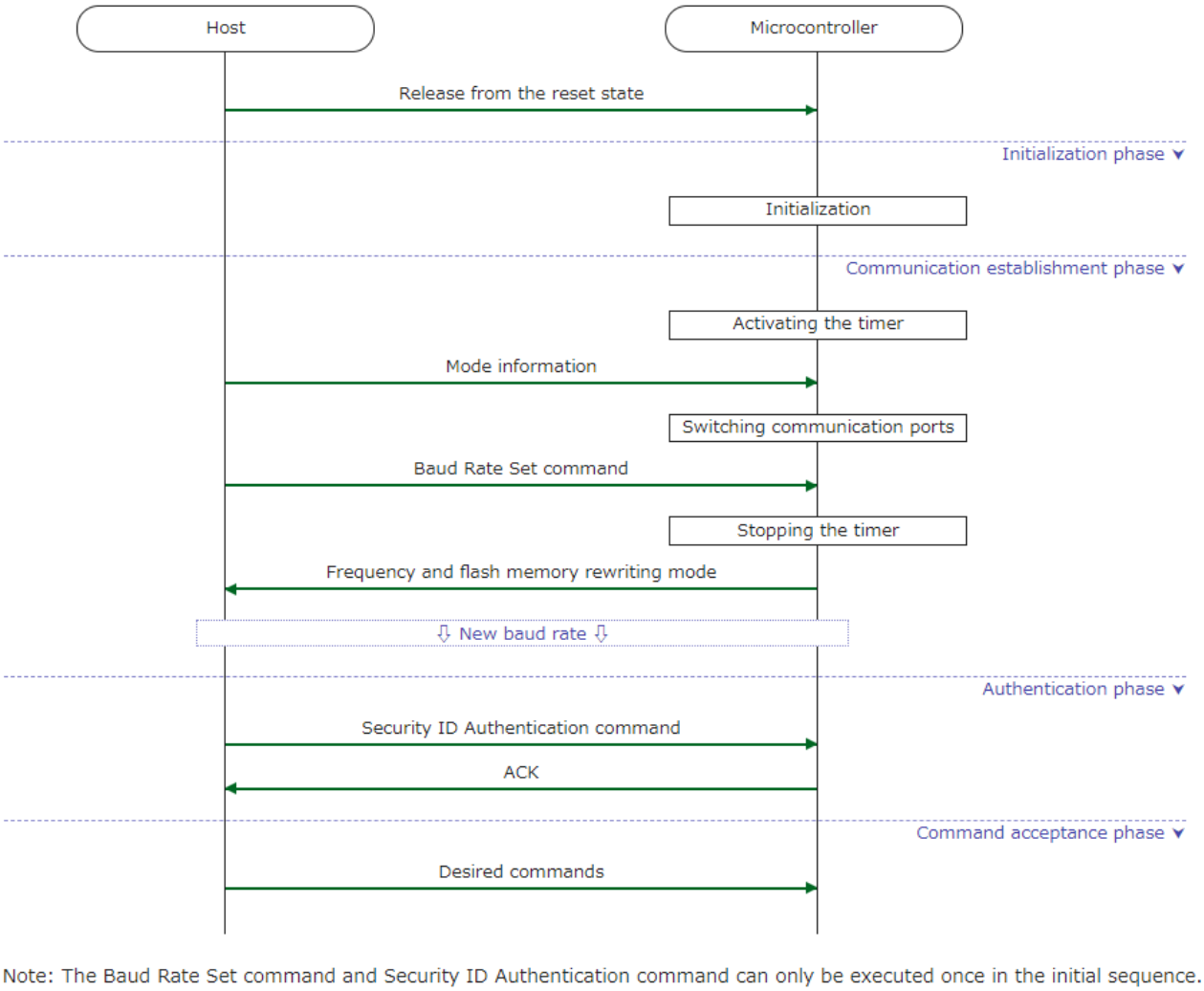
Note: Subsequent operation is not guaranteed if the communication cable is disconnected during communications.

4. General Procedures

The boot firmware is executed in the following order after release from the reset state. The order of this sequence is fixed.

1. Initialization phase
2. Communication establishment phase
3. Authentication phase
4. Command acceptance phase
5. Command packet reception

Figure 4-1 is a diagram of this sequence.



Note: The Baud Rate Set command and Security ID Authentication command can only be executed once in the initial sequence.

Figure 4-1 Sequence Diagram

4.1 Initialization Phase

The required peripheral functions of the device are initialized in this phase. After the initialization processing, execution proceeds to the communication establishment phase.

4.1.1 Procedure of Processing

After release from the reset state, the boot firmware initializes peripheral functions of the device.

- After initialization is successfully completed, execution proceeds to the communication establishment phase.

4.2 Communication Establishment Phase

The boot firmware sets up the communication pins and baud rate according to the specified communication mode.

After reception of the communication mode information (1 byte), the firmware only accepts the Baud Rate Set command.

After successful execution of the Baud Rate Set command, execution proceeds to either of the following phases.

- Authentication phase if the device is in flash memory programming mode and ID authentication is enabled
- Command acceptance phase if the device is in flash memory programming mode and ID authentication is disabled

Note: Communications are at 115.2 Kbps until the Baud Rate Set command is received.

4.2.1 Procedure of Processing

The boot firmware activates a timer and prepares for reception of the communication mode information.

- After waiting until the TOOL0 signal goes high, the firmware activates a timer and then prepares for reception.

After the preparation for reception, the firmware receives the communication mode information (1 byte) and analyzes the mode.

- If the received communication mode information is "3Ah", the device operates in single-line UART flash memory programming mode. (The communication pin is not switched.)
- If the received communication mode information is "00h", the device operates in dedicated UART flash memory programming mode. (The communication pins are switched to TOOLTxD and TOOLRxD.)
- If the received communication mode information is "3Ah" or "00h" and FLPEN (bit 5 at address 000C3h) = 0, execution enters an infinite loop.
- If the received communication mode information is neither of the above values, execution enters an infinite loop, after which, an internal reset will be generated due to a timer interrupt.

After successful completion of the mode analysis, the firmware receives a command packet and analyzes it.

After successful completion of the packet analysis, the Baud Rate Set command is executed.
(For details, refer to section 6.7, Baud Rate Set Command.)

- If the packet analysis is abnormally terminated, execution enters an infinite loop, after which, an internal reset will be generated.
- If the Baud Rate Set command is abnormally terminated, execution enters an infinite loop, after which, an internal reset will be generated.
- If the Baud Rate Set command is successfully completed and ID authentication is enabled, execution proceeds to the authentication phase.
- If the Baud Rate Set command is successfully completed and ID authentication is disabled, execution proceeds to the command acceptance phase.

Timing Charts of Communication Establishment

Figure 4-2 is a timing chart of dedicated UART communications.

Figure 4-3 is a timing chart of single-line UART communications.

Table 4-1 lists the specifications of UART communications.

Figure 4-4 is a timing chart when the mode information is incorrect.

Table 4-2 lists the specifications of communications when the mode information is incorrect.

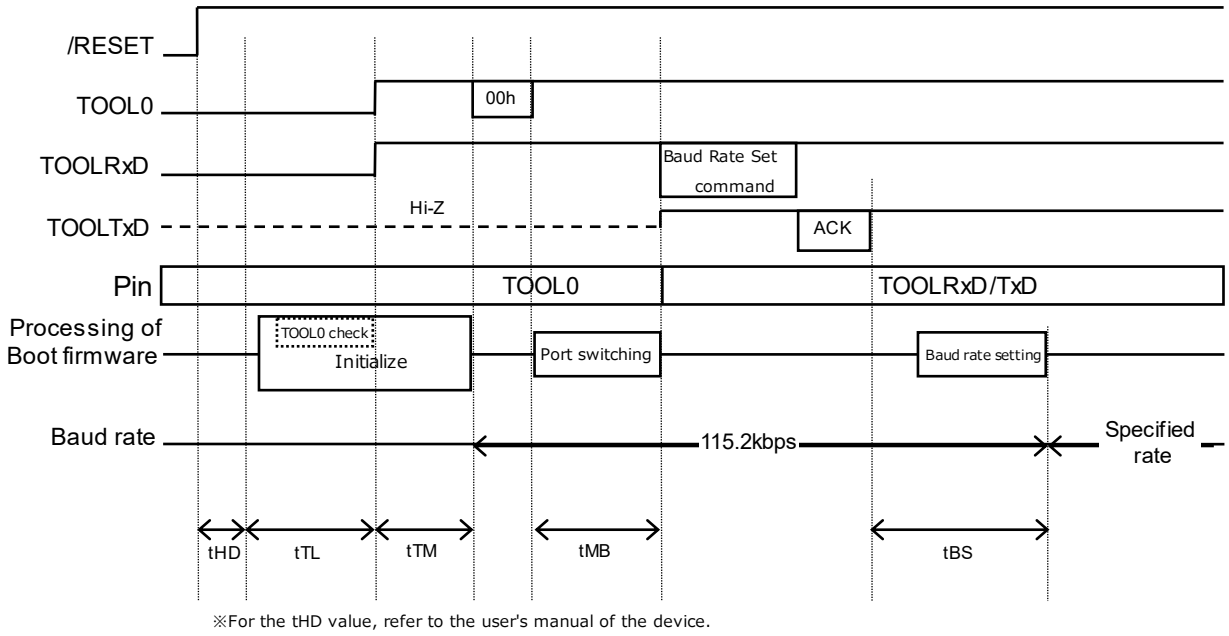


Figure 4-2 Timing Chart of Dedicated UART Communications

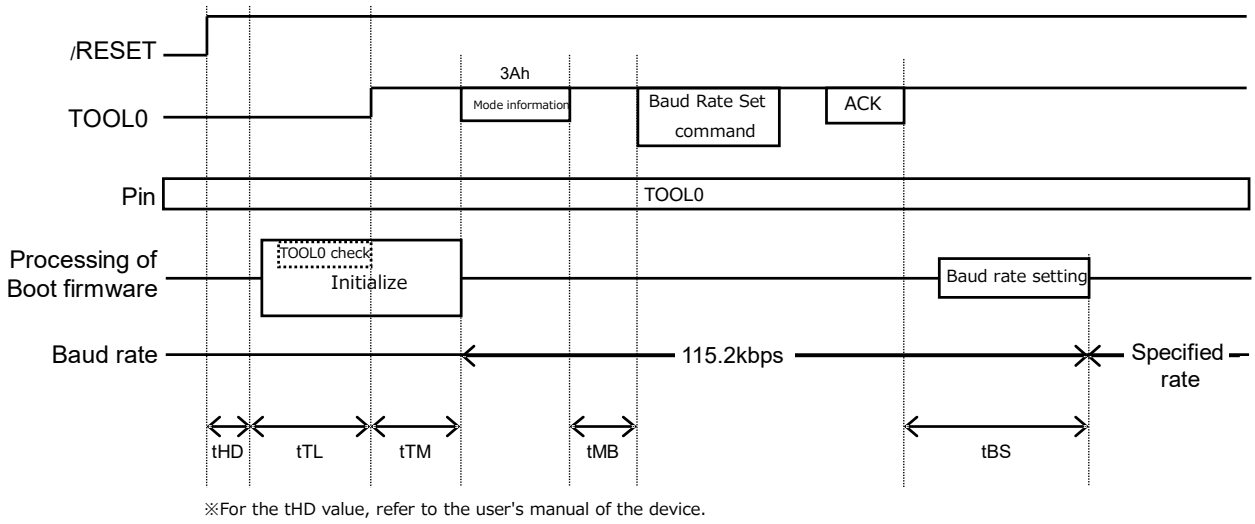


Figure 4-3 Timing Chart of Single-Line UART Communications

Table 4-1 Specifications of UART Communications

Parameter	Symbol	Min.	Typ.	Max.	Unit
After the end of tHD until the TOOL0 check (to see whether it is high or low) is possible	tTL	1	—	—	ms
After the end of the TOOL0 check until the transmission of the communication mode information is possible	tTM	1.2	—	—	ms
After the end of the transmission of the communication mode information until the transmission of the Baud Rate Set command is possible	tMB	10	—	—	us
Time after the end of ACK transmission for the Baud Rate Set command until the baud rate setting is completed	tBS	—	—	1	ms

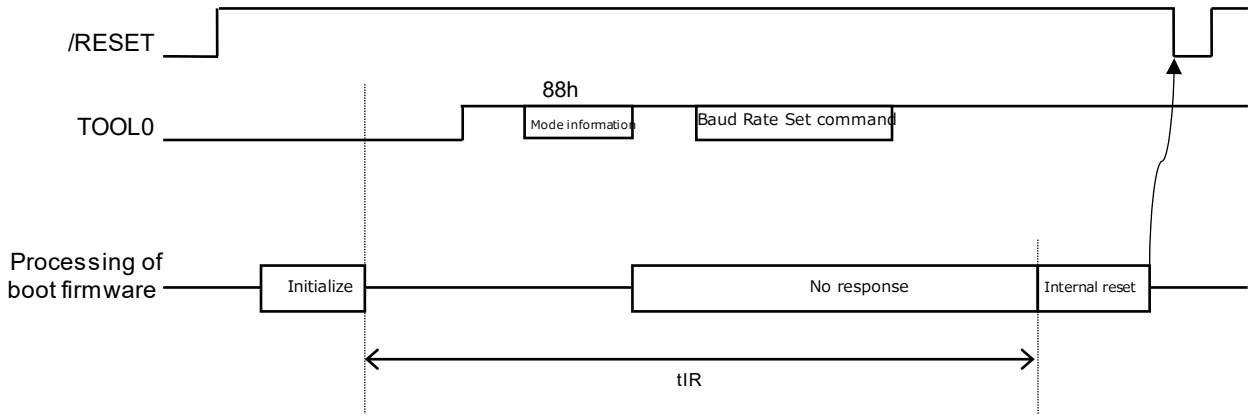


Figure 4-4 Timing Chart when the Mode Information is Incorrect

Table 4-2 Specifications of Communications when the Mode Information is Incorrect

Parameter	Symbol	Min.	Typ.	Max.	Unit
Time after the end of boot firmware initialization until the generation of an internal reset	tIR	—	—	100	ms
Time after TOOL0 high detection until the generation of an internal reset	tTR	—	—	100	ms

4.3 Authentication Phase

The programmer connection ID for serial programming is authenticated in this phase. Only the Security ID Authentication command and Silicon Signature command are accepted.

After successful execution of the Security ID Authentication command, execution proceeds to the command acceptance phase.

4.3.1 Procedure of Processing

Upon receiving a command packet, the boot firmware analyzes it.

After successful completion of the packet analysis, the Security ID Authentication command is executed.

(For details, refer to section 6.8, Security ID Authentication Command.)

- If the Security ID Authentication command is abnormally terminated, execution does not proceed to any other phase.
- If the Security ID Authentication command is successfully completed, execution proceeds to the command acceptance phase.
- After the end of the Silicon Signature command, execution does not proceed to any other phase.

4.4 Command Acceptance Phase

Commands other than the Baud Rate Set or Security ID Authentication command can be accepted in this phase.

Note: Use the Reset command and check the results to determine whether execution is currently in the command acceptance phase.

4.4.1 Procedure of Processing

Upon receiving a command packet, the boot firmware analyzes it.

After successful completion of the packet analysis, the received command is executed.

(For details, refer to the descriptions of individual commands.)

- After the end of command execution, execution does not proceed to any other phase.

4.5 Command Packet Reception

Upon receiving a command packet, the boot firmware analyzes it.

4.5.1 Procedure of Processing

- Reception of SOH is recognized as the start of a command packet.
  If the received value is not SOH, the firmware waits until SOH arrives.
- If the received command packet does not end with ETX, NACK is sent.
- If the value of SUM in the received command packet does not match the calculated checksum value, the checksum error code is sent.
- If the CMD in the received command packet is an undefined code, the command number error code is sent.
- If the command received in the communication establishment phase is not a Baud Rate Set command, the command number error code is sent.
- If the command received in the authentication phase is neither a Security ID Authentication command nor a Silicon Signature command, the command number error code is sent.
- If either of the following commands is received in the command acceptance phase, the command number error code is sent.
  — Baud Rate Set command
  — Security ID Authentication command
- If the value of LEN in the received command packet does not match that defined for the given command, NACK is sent.
- If any of the above errors has occurred in the communication establishment phase, execution enters an infinite loop with no further processing being done, after which, an internal reset will be generated due to a timer interrupt.

Note: The flash memory remains in the same state as before the command was received.

If any of the above errors has occurred in a phase other than the communication establishment phase, execution returns to the command wait state with no further processing being done.

Note: The flash memory remains in the same state as before the command was received.

5. Packet Format

5.1 Command Packet

The host sends command packets to the boot firmware in the following format.

Table 5-1 shows the format of a command packet.

Table 5-1 Format of a Command Packet

Symbol	Size	Value	Description
SOH	1 byte	01h	Start data of the packet
LEN	1 byte	—	Packet length (total length of the CMD and command information fields) [00h: 256 bytes]
CMD	1 byte	—	Command code
Command information	1 to 255 bytes	—	Command information Examples: For the Programming command: Address range for writing For the Block Erase command: Address of the block to be erased
SUM	1 byte	—	Checksum of the values from the LEN field to the last data in the command information Example: LEN + CMD + command information (1) + command information (2) + ... + command information (n) + SUM = 00h
ETX	1 byte	03h	End data of the packet

5.2 Data Packet

The host and boot firmware transfer data in the following format.

Table 5-2 shows the format of a data packet.

Table 5-2 Format of a Data Packet

Symbol	Size	Value	Description
STX	1 byte	02h	Start data of the packet
LEN	1 byte	—	Packet length (total length of the data field) [00h: 256 bytes]
Data	1 to 256 bytes	—	Data for transmission Examples: For the Programming command: Data to be written For transmission of the status: Status code
SUM	1 byte	—	Checksum of the values from the LEN field to the last data in the data field Example: LEN + data (1) + data (2) + ... + data (n) + SUM = 00h
ETX (ETB)	1 byte	03h (17h)	ETX: End data of the packet when there are no data packets to be sent after this packet ETB: End data of the packet when data are divided into multiple packets and remaining data packets are to be sent after this packet

5.3 CMD: Command Code

Table 5-3 is a list of the command codes (CMD).

Table 5-3 List of Command Codes (CMD)

Value	Name	Description
00h	Reset command	Returns ACK.
13h	Verify command	Verifies the data in the target area.
22h	Block Erase command	Erases the data in the target area.
32h	Block Blank Check command	Checks if the target area is blank.
40h	Programming command	Writes data to the target area.
41h	Secure Programming command	Writes data including security information to the target area.
9Ah	Baud Rate Set command	Specifies the baud rate and frequency for UART communications.
9Ch	Security ID Authentication command	Authenticates the programmer connection ID.
A0h	Security Set command	Specifies the security information.
A1h	Security Get command	Returns the security information.
A2h	Security Release command	Initializes the security information.
B0h	Checksum command	Calculates the checksum of the specified area.
C0h	Silicon Signature command	Returns the signature information.

5.4 STS: Status Code

Table 5-4 is a list of the status codes (STS).

Table 5-4 List of Status Codes (STS)

Value	Name	Description
04h	Command number error	The specified command code is undefined or the received command is not allowed in the current phase.
05h	Parameter error	A specified parameter was out of the allowable range.
06h	ACK	Response in case of successful operation
07h	Checksum error	The calculated checksum value does not match the SUM value in the packet.
0Fh	Verification error	A non-matching value was found during verification.
10h	Protection error	A sequencer error has occurred or the attempted processing has been prohibited by the Security Set command.
15h	NACK	The packet structure is illegal, etc.
1Ah	Erasure error	Erasure error
1Bh	Blank error or internal verification error	A blank error (a non-blank location has been found) or an internal verification error has occurred.
1Ch	Write error	Write error
23h	Frequency error	Generating a correct frequency for rewriting the flash memory from the specified parameters was not possible.
24h	ID authentication error	Failure in authentication of the programmer connection ID
25h	Security system error	Abnormal operation due to a malfunction of the secure driver software was found.

5.5 Note on Usage

Table 5-5 shows a note on usage.

Table 5-5 Note on Usage

(1)	After a power-on reset with bit 4 at address 000C4h being set to 0, a response may not be returned for commands accepted in the command acceptance phase.

6. Commands

6.1 Reset Command

The host uses this command to check the state of the boot firmware.

6.1.1 Sequence Diagram

Figure 6-1 is a diagram of the Reset command sequence.

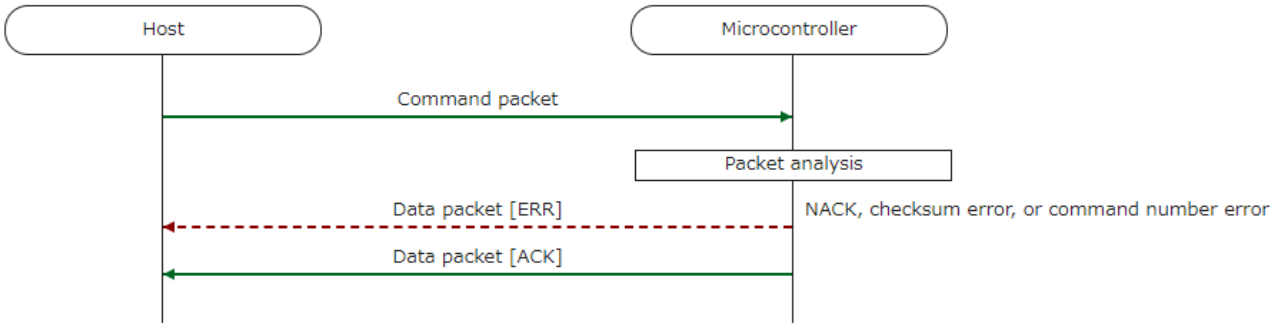


Figure 6-1 Reset Command

6.1.2 List of Packets to be Transmitted

Table 6-1 to Table 6-3 respectively show the command packet and two types of data packet (for ACK and ERR).

Table 6-1 Command Packet

SOH	(1 byte)	01h
LEN	(1 byte)	01h
CMD	(1 byte)	00h
SUM	(1 byte)	FFh
ETX	(1 byte)	03h

Table 6-2 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-3 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.1.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, ACK is sent.

- ACK is sent.

Note: The flash memory remains in the same state as before the command was received.

6.1.4 Status Information Returned from the Microcontroller

Table 6-4 lists the status information.

Table 6-4 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
Execution is in the command acceptance phase.	ACK

6.2 Verify Command

This command is used to compare the data in an area specified as an address range with the data received from the host. The code flash memory or data flash memory is specifiable but specifying a range that extends beyond the boundary of the code flash memory or data flash memory is not allowed. The addresses where the target area starts and ends must be the addresses where given blocks start and end.

Verification may produce different results before and after a reset depending on the setting of IDRDEN in the security option byte.

When a non-matching value is found, the verification error code is returned after the last verification of the specified range.

6.2.1 Sequence Diagram

Figure 6-2 is a diagram of the Verify command sequence.

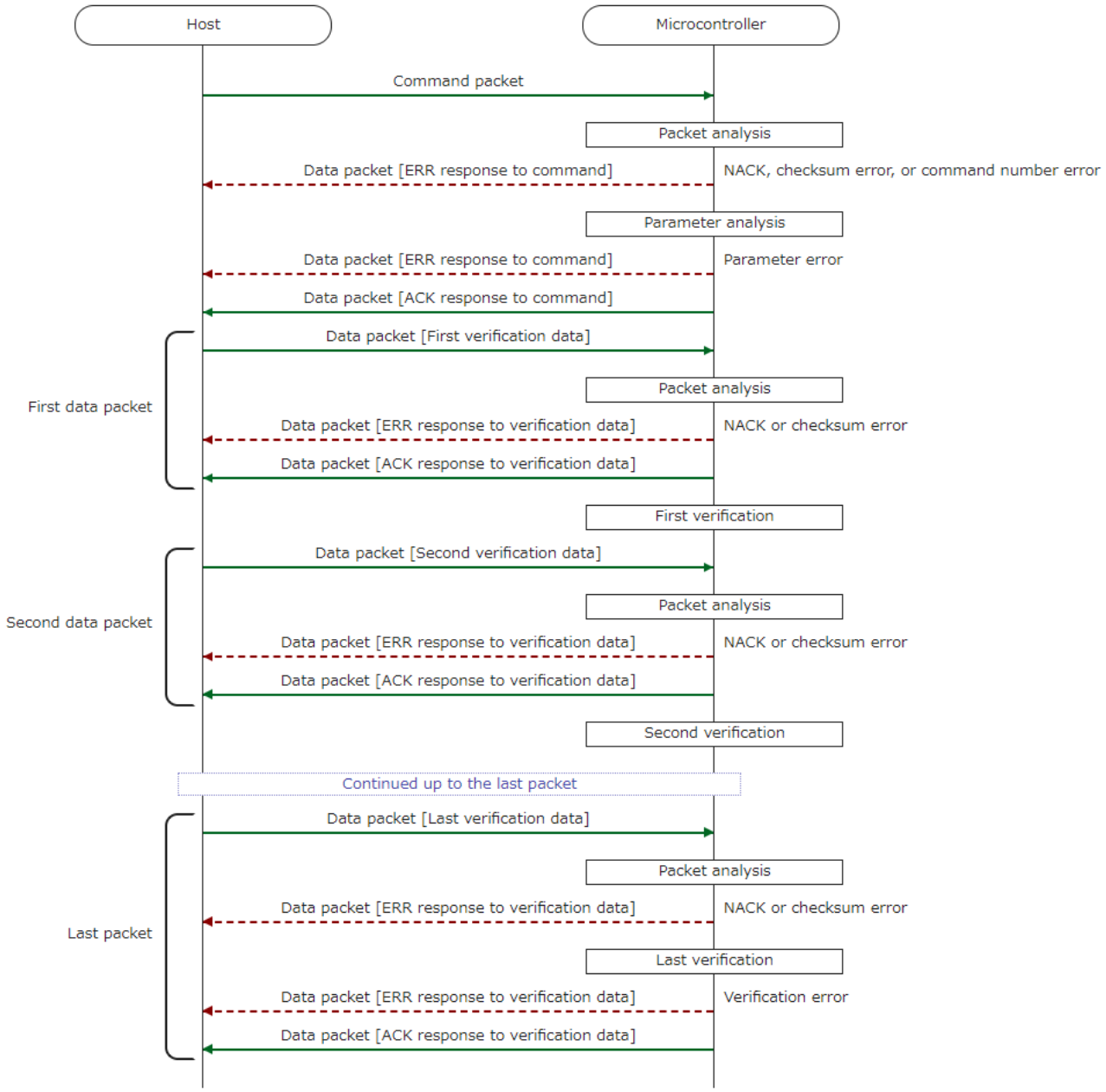


Figure 6-2 Verify Command

6.2.2 List of Packets to be Transmitted

Table 6-5 to Table 6-11 respectively show the command packet and six types of data packet (for verification data, last verification data, ACK response to command, ERR response to command, ACK response to verification data, and ERR response to verification data).

Table 6-5 Command Packet

SOH	(1 byte)	01h	
LEN	(1 byte)	07h	
CMD	(1 byte)	13h	
SAD	(3 bytes)	Start address	Order of transfer: Low → Middle → High Example: 23800h = 00h → 38h → 02h
EAD	(3 bytes)	End address	Order of transfer: Low → Middle → High Example: 3FFFFh = FFh → FFh → 03h
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-6 Data Packet [Verification Data]

STX	(1 byte)	02h
LEN	(1 byte)	00h (256 bytes)
DAT	(256 bytes)	Data to be verified
SUM	(1 byte)	Checksum value
ETB	(1 byte)	17h

Table 6-7 Data Packet [Last Verification Data]

STX	(1 byte)	02h
LEN	(1 byte)	00h (256 bytes)
DAT	(256 bytes)	Data to be verified
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-8 Data packet [ACK Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-9 Data Packet [ERR Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-10 Data Packet [ACK Response to Verification Data]

STX	(1 byte)	02h
LEN	(1 byte)	02h
STS (communication status)	(1 byte)	06h (ACK)
STS (verification status)	(1 byte)	06h (ACK)
SUM	(1 byte)	F2h
ETX	(1 byte)	03h

Table 6-11 Data Packet [ERR Response to Verification Data]

STX	(1 byte)	02h
LEN	(1 byte)	02h
STS (communication status)	(1 byte)	Status code
STS (verification status)	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.2.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, parameter analysis is executed.

- If SAD is greater than EAD, the parameter error code is sent.
- If SAD or EAD is neither in the code flash memory area nor in the data flash memory area, the parameter error code is sent.
- If the area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory, the parameter error code is sent.
- If either or both of SAD and EAD are not the correct addresses where a block starts or ends, the parameter error code is sent.
- If any of the above errors has occurred, execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

- If none of the above errors has occurred, ACK is sent.

After successful completion of the parameter analysis, a data packet is received and packet analysis is executed.

- Reception of STX is recognized as the start of a data packet.
  If the received value is not STX, the firmware waits until STX arrives.
- If the received data packet for the last verification data does not end with ETX, the STS (communication status code) is set to "NACK".
- If a received data packet that is not for the last verification data does not end with ETB, the STS (communication status code) is set to "NACK".
- If the value of SUM in the received data packet does not match the calculated checksum value, the STS (communication status code) is set to "checksum error".
- If the value of LEN in the received data packet does not match that defined for the given command, the STS (communication status code) is set to "NACK".
- If any of the above errors has occurred, the error code is sent and execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

If the received data packet is not for the last verification data, ACK is sent and verification is executed.

- ACK is sent and verification is executed.
- After the end of verification (regardless of whether the result of verification was successful or abnormal), the next data packet is received.

If the received data packet is for the last verification data, verification is executed but ACK is not sent.

- If a non-matching value is found anywhere in the specified area, the STS (verification status code) is set to "verification error", the error code is sent, and execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

- On successful completion of the verification, ACK is sent and execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

6.2.4 Status Information Returned from the Microcontroller

Table 6-12 lists the status information.

Table 6-12 Status Information

Condition	STS
The received command packet does not end with ETX.	NACK
The value of SUM in the received command packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received command packet does not match that defined for the given command.	NACK
SAD is greater than EAD.	Parameter error
SAD or EAD is neither in the code flash memory area nor in the data flash memory area.	Parameter error
The area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory.	Parameter error
SAD or EAD is not the correct address of the start or end of a block, respectively.	Parameter error
The received data packet does not end with ETX or ETB.	NACK
The total number of data items that have been received exceeds the size of the specified area.	NACK
The total number of data items that have been received including the last data packet (with ETX appended) does not reach the size of the specified area.	NACK
The value of LEN in the received data packet does not match that defined for the given command.	NACK
The value of SUM in a received data packet does not match the calculated checksum value.	Checksum error
A non-matching value has been found.	Verification error
Processing has been successfully completed.	ACK

6.2.5 Notes on Using this Command

Table 6-13 is a list of notes on using this command.

Table 6-13 Notes on Using this Command

(1)	Immediately after 0 is written to the IDRDEN bit of OCBT4 (bit 2 at address 000C4h), data including an ID can be successfully verified but verification after a reset will produce an error.
(2)	If the Verify command is executed immediately after the Secure Programming command while bit 4 at address 000C4h is not set to 0, verification may produce an error.

6.3 Block Erase Command

This command is used to erase the data in a single block specified by its address. The code flash memory or data flash memory is specifiable. The target address must be aligned with a block boundary.

6.3.1 Sequence Diagram

Figure 6-3 is a diagram of the Block Erase command sequence.

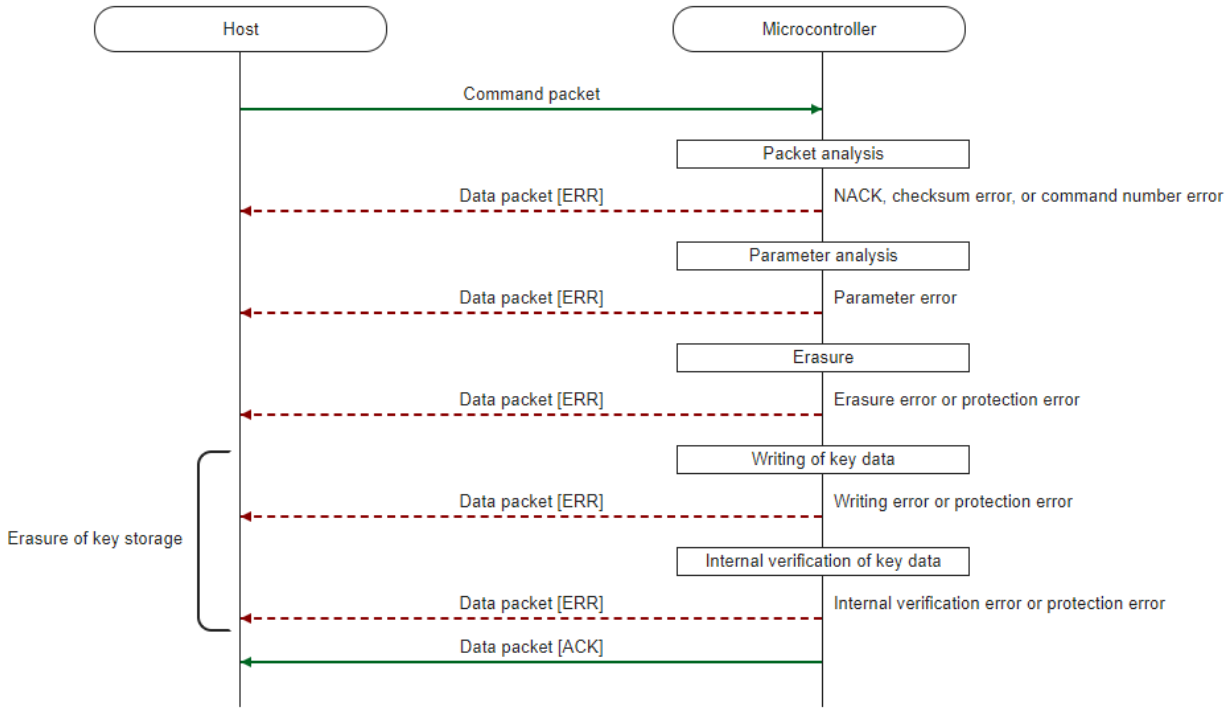


Figure 6-3 Block Erase Command

6.3.2 List of Packets to be Transmitted

Table 6-14 to Table 6-16 respectively show the command packet and two types of data packet (for ACK and ERR).

Table 6-14 Command Packet

SOH	(1 byte)	01h	
LEN	(1 byte)	04h	
CMD	(1 byte)	22h	
SAD	(3 bytes)	Start address	Order of transfer: Low → Middle → High Example: 23400h = 00h → 34h → 02h
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-15 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
EX	(1 byte)	03h

Table 6-16 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.3.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, parameter analysis is executed.

- If SAD is neither in the code flash memory area nor in the data flash memory area, the parameter error code is sent.
- If SAD is not aligned with a block boundary, the parameter error code is sent.
- If either of the above errors has occurred, execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

After successful completion of the parameter analysis, erasure is executed.

- If an erasure error (ERER) has occurred during the erasure processing, the erasure error code is sent.
- If a sequencer error (SEQER) has occurred during the erasure processing, the protection error code is sent.
- If either of the above errors has occurred, execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either an erasure error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after an erasure error or a protection error.

- After successful completion of the erasure, ACK is sent and execution returns to the command wait state if bit 0 at address 000C4h is set to 0 or the target area for erasure does not include the last 2 Kbytes of the data flash memory.

Note: The specified area of the flash memory is set to the erased state.

After successful completion of the erasure, if bit 0 at address 000C4h is set to 1 and the target area for erasure includes the last 2 Kbytes of the data flash memory, key data are written to part of the last 1 Kbyte of the data flash memory.

- If a write error has occurred (WRER has been set) during the processing for writing, the write error code is sent.
- If a sequencer error has occurred (SEQER has been set) during the processing for writing, the protection error code is sent.
- If either of the above errors has occurred, execution returns to the command wait state.

Notes: 1. The key data written to the flash memory becomes undefined.
2. If another command is executed immediately after either a write error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after a write error or a protection error.

After successful completion of the processing for writing to the data flash memory, internal verification is executed.

- If an internal verification error has occurred (IVER has been set) during internal verification, the internal verification error code is sent.
- If a sequencer error has occurred (SEQER has been set) during internal verification, the protection error code is sent.
- If either of the above errors has occurred, execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either an internal verification error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after an internal verification error or a protection error.

- On successful completion of internal verification, ACK is sent and execution returns to the command wait state.

Note: The specified area of the flash memory is set to the erased state.

6.3.4 Status Information Returned from the Microcontroller

Table 6-17 lists the status information.

Table 6-17 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
SAD is neither in the code flash memory area nor in the data flash memory area.	Parameter error
SAD is not aligned with a block boundary.	Parameter error
A sequencer error has occurred during the erasure processing.	Protection error
An erasure error has occurred during the erasure processing.	Erasure error
A write error has occurred during the processing for writing.	Write error
An internal verification error has occurred during internal verification.	Internal verification error
Processing has been successfully completed.	ACK

6.4 Block Blank Check Command

This command is used to perform blank checking of an area specified as an address range. The code flash memory or data flash memory is specifiable but specifying a range that extends beyond the boundary of the code flash memory or data flash memory is not allowed. The addresses of the target area must be specified as boundaries in block units.

6.4.1 Sequence Diagram

Figure 6-4 is a diagram of the Block Blank Check command sequence.

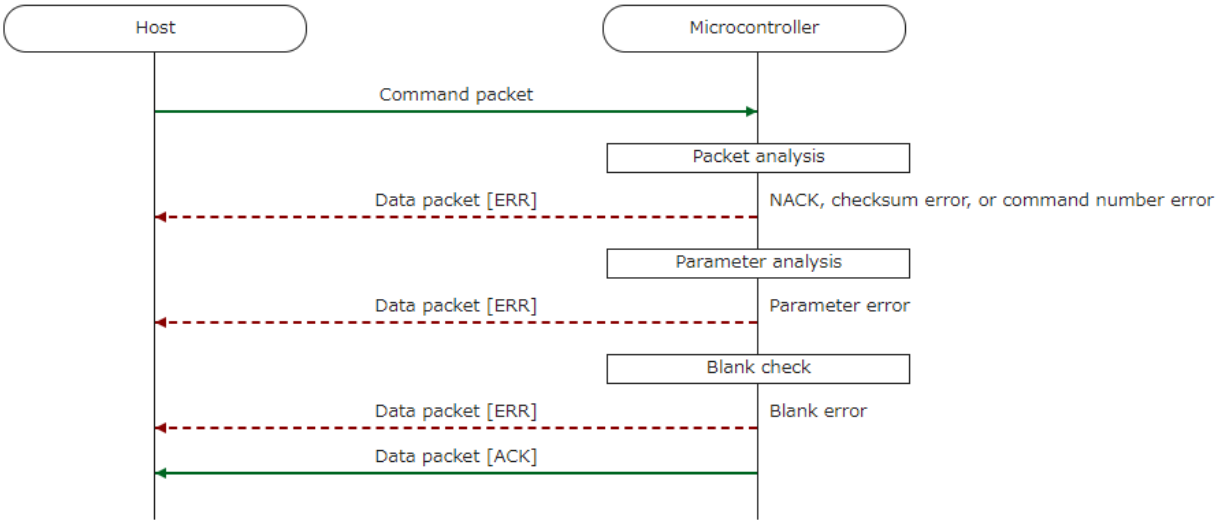


Figure 6-4 Block Blank Check Command

6.4.2 List of Packets to be Transmitted

Table 6-18 to Table 6-20 respectively show the command packet and two types of data packet (for ACK and ERR).

Table 6-18 Command Packet

SOH	(1 byte)	01h	
LEN	(1 byte)	08h	
CMD	(1 byte)	32h	
SAD	(3 bytes)	Start address	Order of transfer: Low → Middle → High Example: 23400h = 00h → 34h → 02h
EAD	(3 bytes)	End address	Order of transfer: Low → Middle → High Example: 3FFFFh = FFh → FFh → 03h
TAR	(1 byte)	Target area	00h: Area specified by SAD and EAD 01h: Area specified by SAD and EAD and the following areas BTFLG: Boot flag BTPR: Boot block cluster protection SEPR: Sector erase protection WRPR: Write protection TEPR: Test mode protection FSW: Flash shield window
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-19 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-20 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.4.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, parameter analysis is executed.

- If SAD is greater than EAD, the parameter error code is sent.
- If SAD or EAD is neither in the code flash memory area nor in the data flash memory area, the parameter error code is sent.
- If the area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory, the parameter error code is sent.
- If SAD or EAD is not aligned with a block boundary, the parameter error code is sent.
- If the specified TAR value is not allowed for this command, the parameter error code is sent.
- If any of the above errors has occurred, execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

After successful completion of the parameter analysis, blank checking is executed.

- If a blank error has occurred during blank checking for the area specified by SAD and EAD, the blank error code is sent and execution returns to the command wait state.

Notes: 1. The flash memory remains in the same state as before the command was received.
2. If another command is executed immediately after a blank error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after a blank error.

- On successful completion of blank checking, ACK is sent and execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

6.4.4 Status Information Returned from the Microcontroller

Table 6-21 lists the status information.

Table 6-21 Status Information

Condition	STS
The received command packet does not end with ETX.	NACK
The value of SUM in the received command packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received command packet does not match that defined for the given command.	NACK
SAD is greater than EAD.	Parameter error
SAD or EAD is neither in the code flash memory area nor in the data flash memory area.	Parameter error
The area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory.	Parameter error
SAD or EAD is not aligned with a block boundary.	Parameter error
The specified TAR value is not allowed for this command.	Parameter error
A blank error has occurred during the blank checking of the specified area.	Blank error
Processing has been successfully completed.	ACK

6.5 Programming Command

This command is used to write the data received from the host to an area specified as an address range. The code flash memory or data flash memory is specifiable but specifying a range that extends beyond the boundary of the code flash memory or data flash memory is not allowed. The addresses where the target area starts and ends must be the addresses where given blocks start and end.

6.5.1 Sequence Diagram

Figure 6-5 is a diagram of the Programming command sequence.

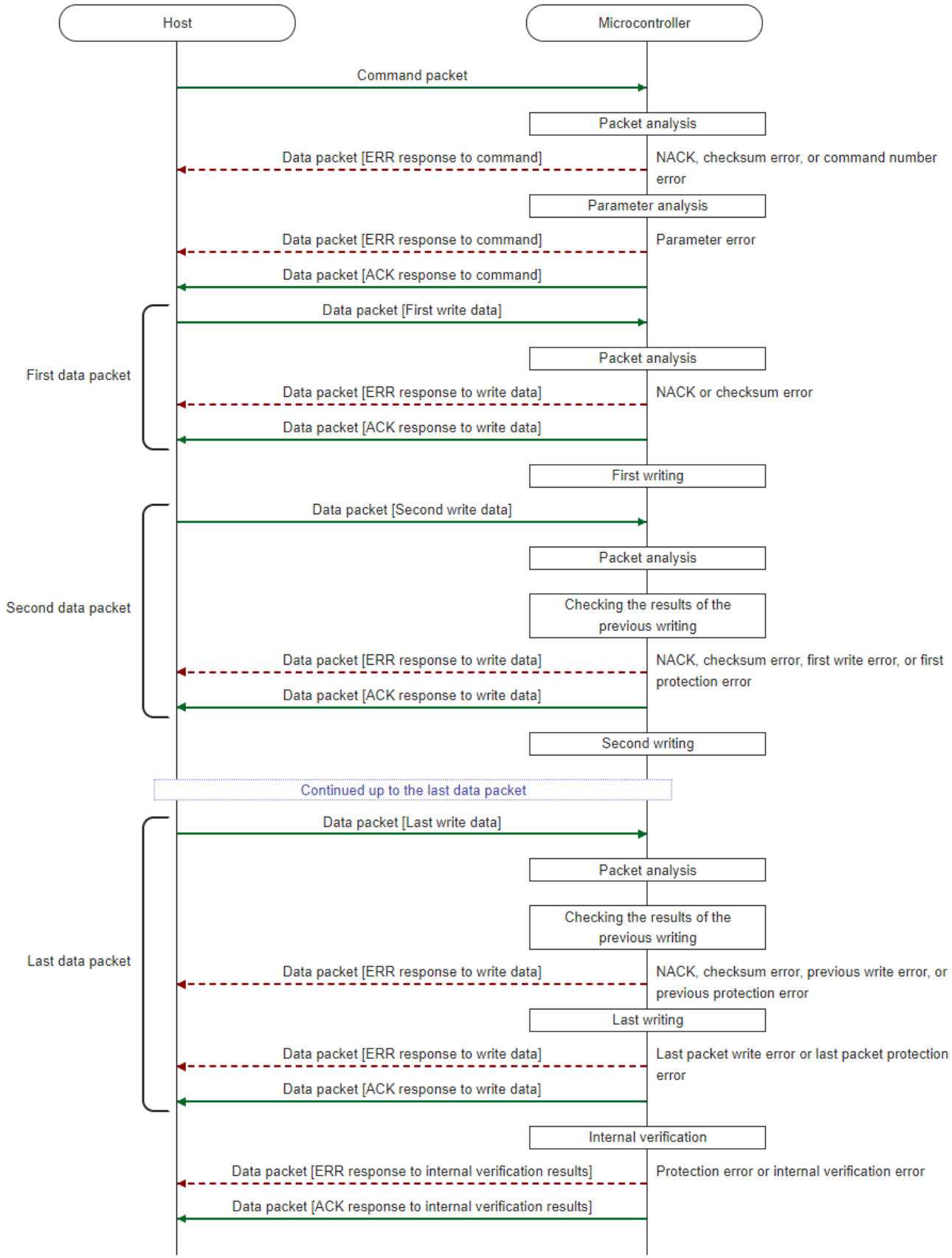


Figure 6-5 Programming Command

6.5.2 List of Packets to be Transmitted

Table 6-22 to Table 6-30 respectively show the command packet and eight types of data packet (for write data, last write data, ACK response to command, ERR response to command, ACK response to write data, ERR response to write data, ACK response to internal verification results, and ERR response to internal verification results).

Table 6-22 Command Packet

SOH	(1 byte)	01h	
LEN	(1 byte)	07h	
CMD	(1 byte)	40h	
SAD	(3 bytes)	Start address	Order of transfer: Low → Middle → High Example: 23400h = 00h → 34h → 02h
EAD	(3 bytes)	End address	Order of transfer: Low → Middle → High Example: 3FFFFh = FFh → FFh → 03h
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-23 Data Packet [Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	00h (256 bytes)
DAT	(256 bytes)	Data to be written
SUM	(1 byte)	Checksum value
ETB	(1 byte)	17h

Table 6-24 Data Packet [Last Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	00h (256 bytes)
DAT	(256 bytes)	Data to be written
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-25 Data Packet [ACK Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-26 Data Packet [ERR Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-27 Data Packet [ACK Response to Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	02h
STS (communication status)	(1 byte)	06h (ACK)
STS (writing status)	(1 byte)	06h (ACK)
SUM	(1 byte)	F2h
ETX	(1 byte)	03h

Table 6-28 Data Packet [ERR Response to Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	02h
STS (communication status)	(1 byte)	Status code
STS (writing status)	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-29 Data Packet [ACK Response to Internal Verification Results]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS (internal verification status)	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-30 Data Packet [ERR Response to Internal Verification Results]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS (internal verification status)	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.5.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, parameter analysis is executed.

- If SAD is greater than EAD, the parameter error code is sent.
- If SAD or EAD is neither in the code flash memory area nor in the data flash memory area, the parameter error code is sent.
- If the area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory, the parameter error code is sent.
- The addresses where the target area starts and ends must be the addresses where given blocks start and end.
- If any of the above errors has occurred, execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

- If none of the above errors has occurred, ACK is sent.

After successful completion of the parameter analysis, a data packet is received and packet analysis is executed.

- Reception of STX is recognized as the start of a data packet.
  If the received value is not STX, the firmware waits until STX arrives.
- If the received data packet for the last of the data to be written does not end with ETX, the STS (communication status code) is set to "NACK".
- If a received data packet that is not for the last of the data to be written does not end with ETB, the STS (communication status code) is set to "NACK".
- If the value of SUM in the received data packet does not match the calculated checksum value, the STS (communication status code) is set to "checksum error".
- If the value of LEN in the received data packet does not match that defined for the given command, the STS (communication status code) is set to "NACK".
- If any of the above errors has occurred for the first data to be written, the error code is sent and execution returns to the command wait state with no further processing being done.

Note: The flash memory remains in the same state as before the command was received.

- For the second or subsequent data to be written, the results of the previous data writing are checked regardless of whether any of the above errors has occurred.

If the received data packet is for the second or subsequent data to be written, the results of the previous writing are checked.

- If a write error has occurred during the processing for writing of the previous data packet, the STS (writing status code) is set to "write error".
- If a sequencer error has occurred during the processing for writing of the previous data packet, the STS (writing status code) is set to "protection error".
- If either of the above errors has occurred, the error code is sent and execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either a write error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
   If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after a write error or a protection error.

If the received data packet is not for the last of the data to be written, ACK is sent and processing for writing is executed.

- ACK is sent and processing for writing is executed.
- After the end of the processing for writing, the next data packet is received.

If the received data packet is for the last of the data to be written, processing for writing is executed but ACK is not sent.

- Processing for writing is executed but ACK is not sent.
- If a write error has occurred (WRER has been set) during the processing for writing, the STS (writing status code) is set to "write error".
- If a sequencer error has occurred (SEQER has been set) during the processing for writing, the STS (writing status code) is set to "protection error".
- If either of the above errors has occurred, the error code is sent and execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either a write error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after a write error or a protection error.

- After successful completion of the processing for writing, ACK is sent and internal verification is executed.

Note: The specified area of the flash memory is set to the written state.

- If an internal verification error has occurred (IVER has been set) during the internal verification processing, the STS (internal verification status code) is set to "internal verification error".
- If a sequencer error has occurred (SEQER has been set) during the internal verification processing, the STS (internal verification status code) is set to "protection error".
- If either of the above errors has occurred, the error code is sent and execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either an internal verification error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after an internal verification error or a protection error.

- On successful completion of internal verification, ACK is sent and execution returns to the command wait state.

Note: The specified area of the flash memory is set to the written state.

6.5.4 Status Information Returned from the Microcontroller

Table 6-31 lists the status information.

Table 6-31 Status Information

Condition	STS
The received command packet does not end with ETX.	NACK
The value of SUM in the received command packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received command packet does not match that defined for the given command.	NACK
SAD is greater than EAD.	Parameter error
SAD or EAD is neither in the code flash memory area nor in the data flash memory area.	Parameter error
The area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory.	Parameter error
SAD or EAD is not the correct address of the start or end of a block, respectively.	Parameter error
The received data packet does not end with ETX or ETB.	NACK
The total number of data items that have been received exceeds the size of the specified area.	NACK
The total number of data items that have been received including the last data packet does not reach the size of the specified area.	NACK
The value of LEN in the received data packet does not match that defined for the given command.	NACK
The value of SUM in a received data packet does not match the calculated checksum value.	Checksum error
A sequencer error has occurred during the processing for writing.	Protection error
A write error has occurred during the processing for writing.	Write error
An internal verification error has occurred during internal verification.	Internal verification error
Processing has been successfully completed.	ACK

6.5.5 Note on Using this Command

Table 6-32 shows the note on using this command.

Table 6-32 Note on Using this Command

(1)	If 0 is written to bit 4 at address 000C4h and a reset is then applied, acceptance of a command in the command acceptance phase may cause the generation of a reset.

6.6 Secure Programming Command

This command is used to write the data received from the host to an area specified as an address range. The command simultaneously registers secure data.

To use secure driver software, codes corresponding to secure data 1 and 2 should be prepared in advance. Specifying a range that extends beyond the boundary of the code flash memory or data flash memory is not allowed. The addresses where the target area starts and ends must be the addresses where given blocks start and end.

6.6.1 Sequence Diagram

Figure 6-6 is a diagram of the Secure Programming command sequence.

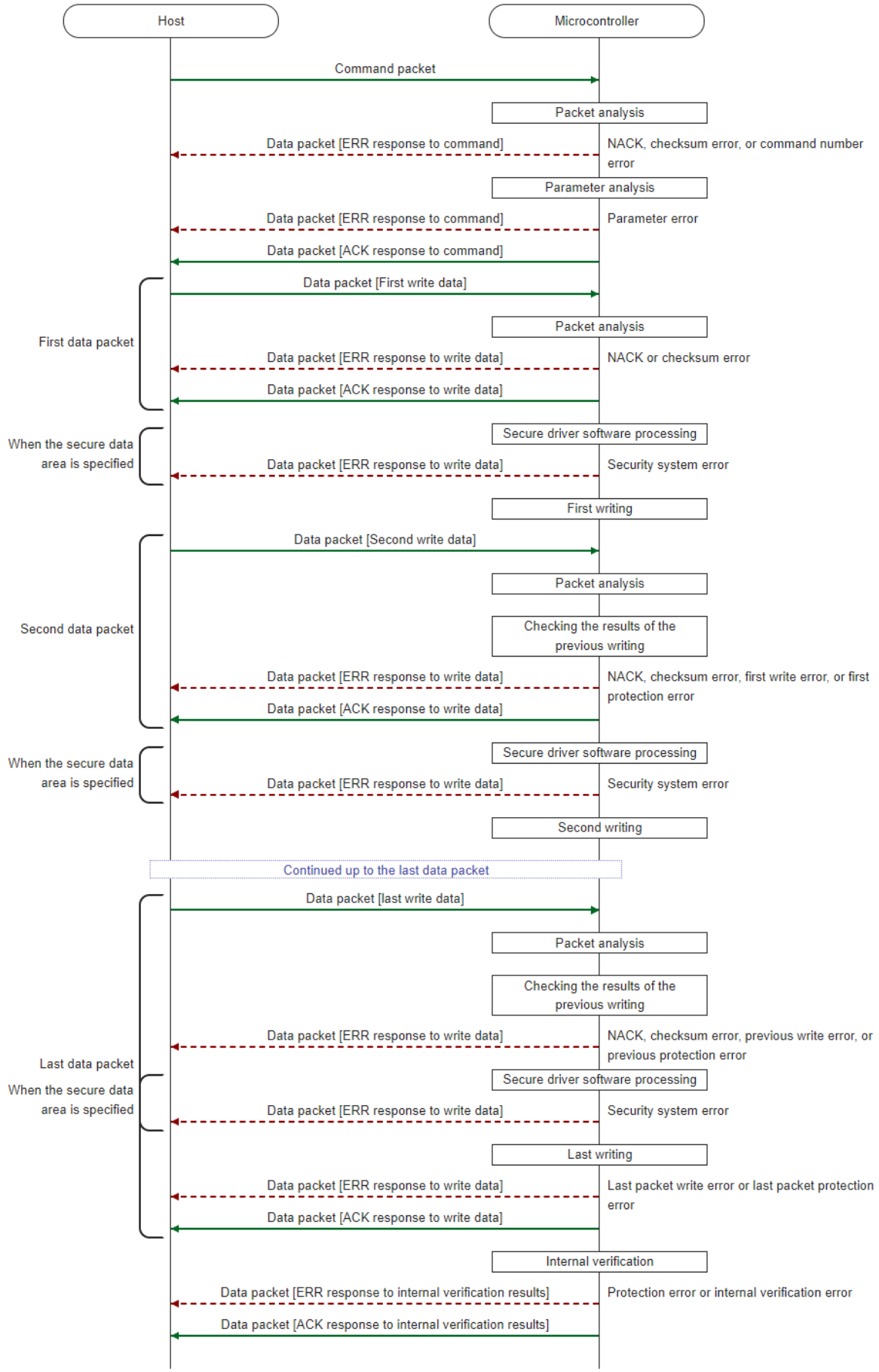


Figure 6-6 Secure Programming Command

6.6.2 List of Packets to be Transmitted

Table 6-33 to Table 6-41 respectively show the command packet and eight types of data packet (for write data, last write data, ACK response to command, ERR response to command, ACK response to write data, ERR response to write data, ACK response to internal verification results, and ERR response to internal verification results).

Table 6-33 Command Packet

<table>
<tr><td>SOH</td><td>(1 byte)</td><td>01h</td><td></td></tr>
<tr><td>LEN</td><td>(1 byte)</td><td>27h</td><td></td></tr>
<tr><td>CMD</td><td>(1 byte)</td><td>41h</td><td></td></tr>
<tr><td>SAD</td><td>(3 bytes)</td><td>Start address</td><td>Order of transfer: Low →→ Middle → High
Example: 23400h = 00h → 34h → 02h</td></tr>
<tr><td>EAD</td><td>(3 bytes)</td><td>End address</td><td>Order of transfer: Low → Middle → High
Example: 3FFFFh = FFh → FFh → 03h</td></tr>
<tr><td>PWM</td><td>(16 bytes)</td><td>Code for secure data 1</td><td>Example: When the code for secure data 1 is "0123456789ABCDEFF0F1F2F3F4F5F6F7", the programmer should transfer data in the order from PWM0 to PWM15 — that is, 01h, 23h, ..., F6h, and F7h.
Order of PWM transfer:
<table>
<tr><td>1st</td><td>2nd</td><td>3rd</td><td>4th</td><td>5th</td><td>6th</td><td>7th</td><td>8th</td></tr>
<tr><td>01</td><td>23</td><td>45</td><td>67</td><td>89</td><td>AB</td><td>CD</td><td>EF</td></tr>
<tr><td>9th</td><td>10th</td><td>11th</td><td>12th</td><td>13th</td><td>14th</td><td>15th</td><td>16th</td></tr>
<tr><td>F0</td><td>F1</td><td>F2</td><td>F3</td><td>F4</td><td>F5</td><td>F6</td><td>F7</td></tr>
</table></td></tr>
<tr><td>PWK</td><td>(16 bytes)</td><td>Code for secure data 2</td><td>Example: When the code for secure data 2 is "0123456789ABCDEFF0F1F2F3F4F5F6F7", the programmer should transfer data in the order from PWM0 to PWM15 — that is, 01h, 23h, ..., F6h, and F7h.
Order of PWM transfer:
<table>
<tr><td>1st</td><td>2nd</td><td>3rd</td><td>4th</td><td>5th</td><td>6th</td><td>7th</td><td>8th</td></tr>
<tr><td>01</td><td>23</td><td>45</td><td>67</td><td>89</td><td>AB</td><td>CD</td><td>EF</td></tr>
<tr><td>9th</td><td>10th</td><td>11th</td><td>12th</td><td>13th</td><td>14th</td><td>15th</td><td>16th</td></tr>
<tr><td>F0</td><td>F1</td><td>F2</td><td>F3</td><td>F4</td><td>F5</td><td>F6</td><td>F7</td></tr>
</table></td></tr>
<tr><td>SUM</td><td>(1 byte)</td><td>Checksum value</td><td></td></tr>
<tr><td>ETX</td><td>(1 byte)</td><td>03h</td><td></td></tr>
</table>

Table 6-34 Data Packet [Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	00h (256 bytes)
DAT	(256 bytes)	Data to be written
SUM	(1 byte)	Checksum value
ETB	(1 byte)	17h

Table 6-35 Data Packet [Last Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	00h (256 bytes)
DAT	(256 bytes)	Data to be written
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-36 Data Packet [ACK Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-37 Data Packet [ERR Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-38 Data Packet [ACK Response to Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	02h
STS (communication status)	(1 byte)	06h (ACK)
STS (writing status)	(1 byte)	06h (ACK)
SUM	(1 byte)	F2h
ETX	(1 byte)	03h

Table 6-39 Data Packet [ERR Response to Write Data]

STX	(1 byte)	02h
LEN	(1 byte)	02h
STS (communication status)	(1 byte)	Status code
STS (writing status)	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-40 Data Packet [ACK Response to Internal Verification Results]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS (internal verification status)	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-41 Data Packet [ERR Response to Internal Verification Results]

STX	(1 byte)	02h
LEN	(1 byte)	02h
STS (internal verification status)	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.6.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, parameter analysis is executed.

- If SAD is greater than EAD, the parameter error code is sent.
- If SAD or EAD is neither in the code flash memory area nor in the data flash memory area, the parameter error code is sent.
- If the area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory, the parameter error code is sent.
- If either or both of SAD and EAD are not the correct addresses where a block starts or ends, the parameter error code is sent.
- If any of the above errors has occurred, execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

- If none of the above errors has occurred, ACK is sent.

After successful completion of the parameter analysis, a data packet is received and packet analysis is executed.

- Reception of STX is recognized as the start of a data packet.
  If the received value is not STX, the firmware waits until STX arrives.
- If the received data packet for the last of the data to be written does not end with ETX, the STS (communication status code) is set to "NACK".
- If a received data packet that is not for the last of the data to be written does not end with ETB, the STS (communication status code) is set to "NACK".
- If the value of SUM in the received data packet does not match the calculated checksum value, the STS (communication status code) is set to "checksum error".
- If the value of LEN in the received data packet does not match that defined for the given command, the STS (communication status code) is set to "NACK".
- If any of the above errors has occurred for the first data to be written, the error code is sent and execution returns to the command wait state with no further processing being done.

Note: The flash memory remains in the same state as before the command was received.

- For the second or subsequent data to be written, the results of the previous data writing are checked regardless of whether any of the above errors has occurred.

If the received data packet is for the second or subsequent data to be written, the results of the previous writing are checked.

- If a security system error (an error due to the secure driver software) has occurred before the processing for writing of the previous data packet, the STS (writing status code) is set to "security system error".
- If a write error has occurred (WRER has been set) during the processing for writing of the previous data packet, the STS (writing status code) is set to "write error".
- If a sequencer error has occurred (SEQER has been set) during the processing for writing of the previous data packet, the STS (writing status code) is set to "protection error".
- If any of the above errors has occurred, the error code is sent and execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either a write error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
   If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after a write error or a protection error.

If the received data packet is not for the last of the data to be written, ACK is sent and processing for writing is executed.

- ACK is sent and processing for writing is executed.
- The secure driver software executes the required processing. After the end of the processing, the next data packet is received.

If the received data packet is for the last of the data to be written, the secure driver software executes processing but ACK is not sent.

- Processing for writing is executed but ACK is not sent.
- If a security system error (an error due to the secure driver software) has occurred before the processing for writing, the STS (writing status code) is set to "security system error".
- If a write error has occurred (WRER has been set) during the processing for writing, the STS (writing status code) is set to "write error".
- If a sequencer error has occurred (SEQER has been set) during the processing for writing, the STS (writing status code) is set to "protection error".
- If any of the above errors has occurred, the error code is sent and execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either a write error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after a write error or a protection error.

- After successful completion of the processing for writing, ACK is sent and internal verification is executed.

Note: The specified area of the flash memory is set to the written state.

- If an internal verification error has occurred (IVER has been set) during the internal verification processing, the STS (internal verification status code) is set to "internal verification error".
- If a sequencer error has occurred (SEQER has been set) during the internal verification processing, the STS (internal verification status code) is set to "protection error".
- If either of the above errors has occurred, the error code is sent and execution returns to the command wait state.

Notes: 1. The state of the specified area in the flash memory becomes undefined.
2. If another command is executed immediately after either an internal verification error or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after an internal verification error or a protection error.

- On successful completion of internal verification, ACK is sent and execution returns to the command wait state.

Note: The specified area of the flash memory is set to the written state.

6.6.4 Status Information Returned from the Microcontroller

Table 6-42 lists the status information.

Table 6-42 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received command packet does not match that defined for the given command.	NACK
SAD is greater than EAD.	Parameter error
SAD or EAD is neither in the code flash memory area nor in the data flash memory area.	Parameter error
The area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory.	Parameter error
SAD or EAD is not the correct address of the start or end of a block, respectively.	Parameter error
The received data packet does not end with ETX or ETB.	NACK
The total number of data items that have been received exceeds the size of the specified area.	NACK
The total number of data items that have been received including the last data packet (with ETX appended) does not reach the size of the specified area.	NACK
The value of LEN in the received data packet does not match that defined for the given command.	NACK
The value of SUM in the received data packet does not match the calculated checksum value.	Checksum error
Abnormal operation due to a malfunction of the secure driver software was found.	Security system error
A sequencer error has occurred (SEQER has been set) during the processing for writing.	Protection error
A write error has occurred (WRER has been set) during the processing for writing.	Write error
An internal verification error has occurred (IVER has been set) during internal verification.	Internal verification error
Processing has been successfully completed.	ACK

6.6.5 Timing Chart

Figure 6-7 is a timing chart and Table 6-43 lists the specifications of communications.

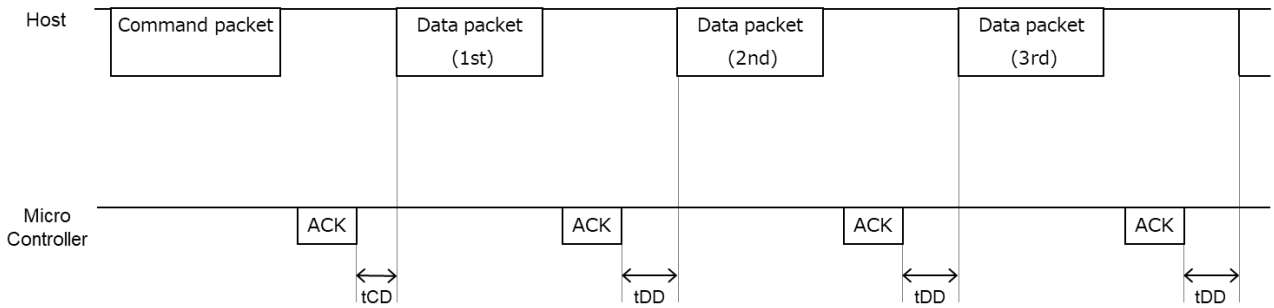


Figure 6-7 Timing Chart

Table 6-43 Specifications of Communications

Parameter	Symbol	Min.	Typ.	Max.	Unit
Time after the end of ACK reception for the command packet and until the transmission of the data packet is allowed	tCD	30	—	—	us
Time after the end of ACK reception for the data packet and until the next transmission of the data packet is allowed	tDD	300	—	—	us

6.6.6 Notes on Using this Command

Table 6-44 is a list of notes on using this command.

Table 6-44 Notes on Using this Command

(1)	If this command is executed while bit 4 at address 000C4h is set to 1 and bit 0 at address 000C4h is set to 0, a security system error may occur.
(2)	If 0 is written to bit 4 at address 000C4h and a reset is then applied, acceptance of a command in the command acceptance phase may cause the generation of a reset.
(3)	If the Verify command is executed immediately after the Secure Programming command while bit 4 at address 000C4h is not set to 0, verification may produce an error.

6.7 Baud Rate Set Command

This command is used to receive the communication baud rate and VDD voltage settings from the host and set up the operating frequency of the device and communication baud rate.

If this command includes an error, the boot firmware becomes unresponsive and waits for a device reset to be applied due to the time limit of the 100-ms timer.

6.7.1 Sequence Diagram

Figure 6-8 is a diagram of the Baud Rate Set command sequence.

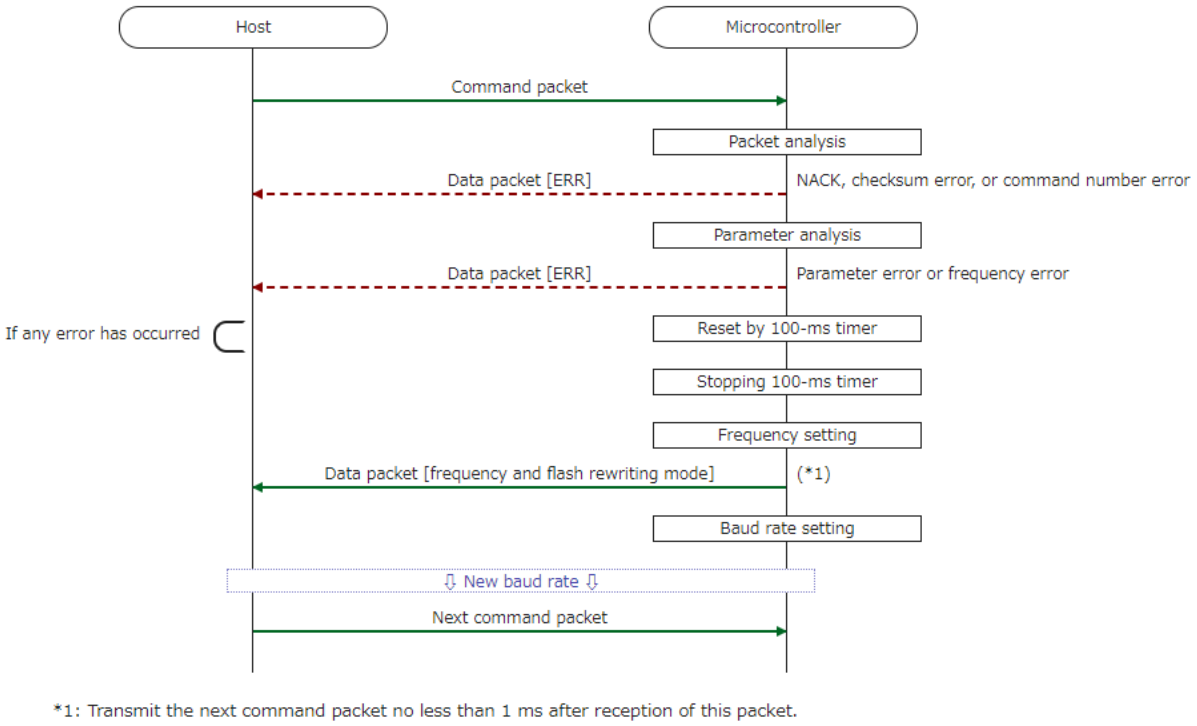


Figure 6-8 Baud Rate Set Command

6.7.2 List of Packets to be Transmitted

Table 6-45 to Table 6-47 respectively show the command packet and two types of data packet (for frequency and flash memory rewriting mode and for ERR).

Table 6-45 Command Packet

SOH	(1 byte)	01h	
LEN	(1 byte)	03h	
CMD	(1 byte)	9Ah	
BRT	(1 byte)	Communication baud rate	00h: 115.2 Kbps 01h: 250 Kbps 02h: 500 Kbps 03h: 1 Mbps Note: A value not listed here generates a parameter error.
VDD	(1 byte)	VDD voltage to be applied (in 100-mV units) (Truncate the digits after the decimal point.)	Example: 1.89 (V) → 18.9 (100 mV) → 18 → 12h Note: A value equivalent to less than 2.7 V generates a parameter error.
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-46 Data Packet [Frequency and Flash Memory Rewriting Mode]

STX	(1 byte)	02h	
LEN	(1 byte)	03h	
STS	(1 byte)	06h (ACK)	
FRQ	(1 byte)	CPU operating frequency (MHz) (Truncate the digits after the decimal point.)	Example: 30.9 (MHz) → 30 → 1Eh
FPM	(1 byte)	Flash memory rewriting mode	00h: Full-speed mode 01h: Wide-voltage mode
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-47 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.7.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, parameter analysis is executed.

- If the specified BRT value (communication baud rate) exceeds the allowable range, the parameter error code is sent.
- If the specified VDD value (VDD voltage to be applied) exceeds the allowable range, the parameter error code is sent.
- If generating a correct frequency for rewriting the flash memory from the specified parameters is not possible, the frequency error code is sent.
- If any of the above errors has occurred, execution enters an infinite loop with no further processing being done, after which, an internal reset will be generated.

Note: The flash memory remains in the same state as before the command was received.

After successful completion of the parameter analysis, the timer is stopped, the frequency is set up, and a data packet is returned.

- After the timer is stopped and the frequency is set up, the CPU operating frequency and the flash memory rewriting mode are sent.

Notes: 1. The flash memory remains in the same state as before the command was received.
2. Before sending the next command packet, wait for no less than 1 ms after this response.

After the data packet is returned, the communication baud rate is set up.

- In the flash memory programming mode, execution proceeds to the authentication phase.

6.7.4 Status Information Returned from the Microcontroller

Table 6-48 lists the status information.

Table 6-48 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
The value of BRT in the received packet is not allowed for this command.	Parameter error
The value of VDD in the received packet is not allowed for this command.	Parameter error
Generating a correct frequency for rewriting the flash memory from the specified parameters was not possible.	Frequency error
Processing has been successfully completed.	ACK

6.7.5 Timing Chart

Figure 6-9 is a timing chart and Table 6-49 lists the specifications of UART communications.

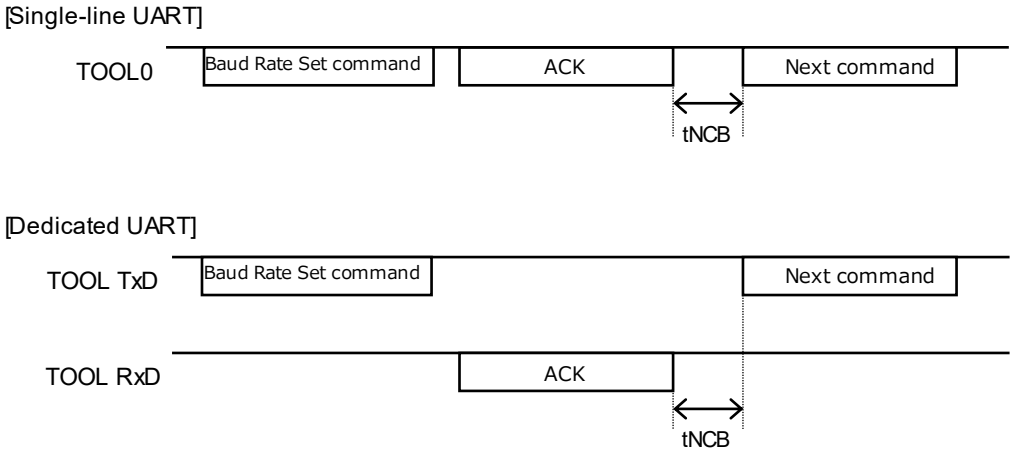


Figure 6-9 Timing Chart

Table 6-49 Specifications of UART Communications

Parameter	Symbol	Min.	Typ.	Max.	Unit
Time after the end of ACK reception for the Baud Rate Set command and until the transmission of the next command packet is allowed	tNCB	1	—	—	ms

6.7.6 List of Parameters

Table 6-50 is a list of parameters (operating frequency and flash memory rewriting mode) to be returned in response to this command and the conditions for returning parameters.

Table 6-50 List of Parameters

Condition			Parameter	
VDD	**Flash Operating Mode**	**HOCO Frequency ***	**CPU Operating Frequency**	**Flash Memory Rewriting Mode**
2.7 V or higher	HS (high-speed main) mode	40 MHz	40 MHz	Full-speed mode
		32 MHz	32 MHz	

Note: * This frequency is selected according to an option byte setting (bits 4 and 3 at address 000C2h).

6.8 Security ID Authentication Command

This command is used to authenticate the programmer connection ID that is received from the programmer.

6.8.1 Sequence Diagram

Figure 6-10 is a diagram of the Security ID Authentication command sequence.

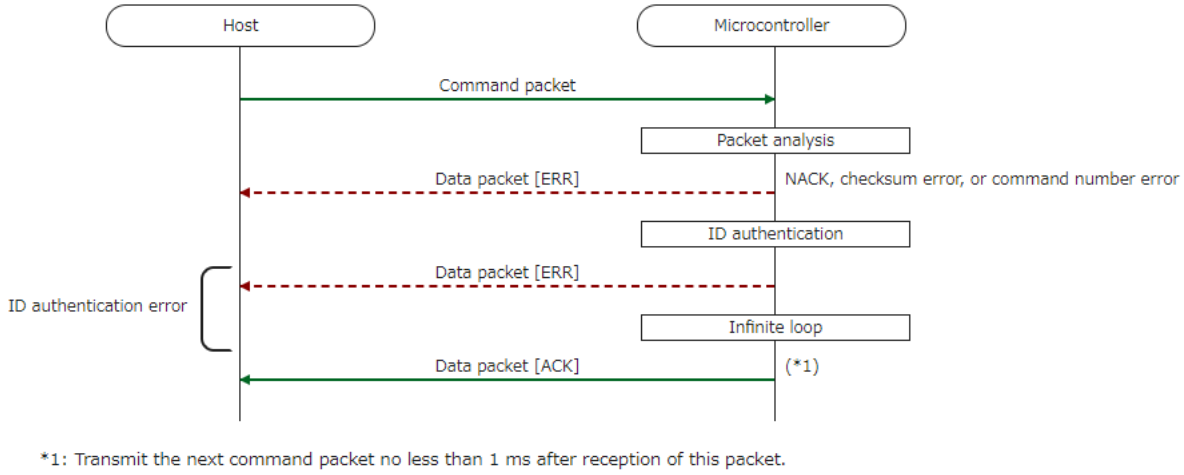


Figure 6-10 Security ID Authentication Command

6.8.2 List of Packets to be Transmitted

Table 6-51 to Table 6-53 respectively show the command packet and two types of data packet (for ACK and ERR).

Table 6-51 Command Packet

<table>
<tr><td>SOH</td><td>(1 byte)</td><td colspan="2">01h</td></tr>
<tr><td>LEN</td><td>(1 byte)</td><td colspan="2">11h</td></tr>
<tr><td>CMD</td><td>(1 byte)</td><td colspan="2">9Ch</td></tr>
<tr><td>IDC</td><td>(16 bytes)</td><td>ID code</td><td>When the security ID code is stored in the flash memory as shown below, the IDC should be sent in the order 01h, 23h, …, 00h, and 11h.

Correspondence between IDC values and storage addresses:
<table><tr><td>D6h</td><td>D7h</td><td>D8h</td><td>D9h</td><td>DAh</td><td>DBh</td><td>DCh</td><td>DDh</td></tr><tr><td>01</td><td>23</td><td>45</td><td>67</td><td>89</td><td>AB</td><td>CD</td><td>EF</td></tr><tr><td>DEh</td><td>DFh</td><td>E0h</td><td>E1h</td><td>E2h</td><td>E3h</td><td>E4h</td><td>E5h</td></tr><tr><td>F0</td><td>F1</td><td>F2</td><td>F3</td><td>F4</td><td>F5</td><td>F6</td><td>F7</td></tr></table>
Order of IDC transmission:
<table><tr><td>1st</td><td>2nd</td><td>3rd</td><td>4th</td><td>5th</td><td>6th</td><td>7th</td><td>8th</td></tr><tr><td>01</td><td>23</td><td>45</td><td>67</td><td>89</td><td>AB</td><td>CD</td><td>EF</td></tr><tr><td>9th</td><td>10th</td><td>11th</td><td>12th</td><td>13th</td><td>14th</td><td>15th</td><td>16th</td></tr><tr><td>F0</td><td>F1</td><td>F2</td><td>F3</td><td>F4</td><td>F5</td><td>F6</td><td>F7</td></tr></table></td></tr>
<tr><td>SUM</td><td>(1 byte)</td><td colspan="2">Checksum value</td></tr>
<tr><td>ETX</td><td>(1 byte)</td><td colspan="2">03h</td></tr>
</table>

Table 6-52 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-53 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.8.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, the received ID is authenticated.

- In case of failure in ID authentication, the ID authentication error code is sent and execution enters an infinite loop.

Note: The flash memory remains in the same state as before the command was received.

- In case of pass in ID authentication, ACK is sent and execution proceeds to the command acceptance phase.

Note: The flash memory remains in the same state as before the command was received.

6.8.4 Status Information Returned from the Microcontroller

Table 6-54 lists the status information.

Table 6-54 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
Failure in ID authentication	ID authentication error
Processing has been successfully completed.	ACK

6.8.5 Timing Chart

Figure 6-11 is a timing chart and Table 6-55 lists the specifications of communications.

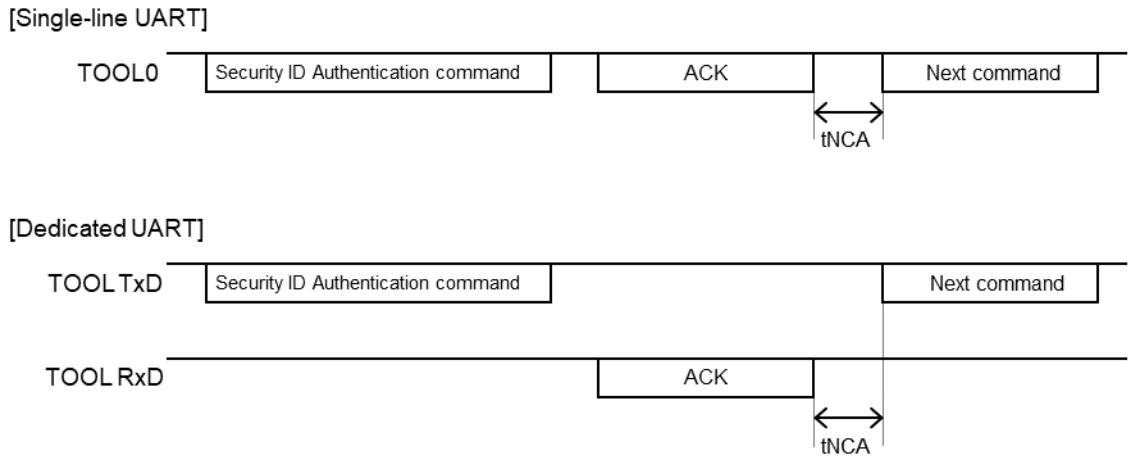


Figure 6-11 Timing Chart

Table 6-55 Specifications of Communications

Parameter	Symbol	Min.	Typ.	Max.	Unit
Time after the end of ACK reception for the Security ID Authentication command and until the transmission of the next command packet is allowed	tNCA	1	—	—	ms

6.9 Security Set Command

This command is used to make security settings (set flash options in the extra area) according to the information received from the host.

6.9.1 Sequence Diagram

Figure 6-12 is a diagram of the Security Set command sequence.

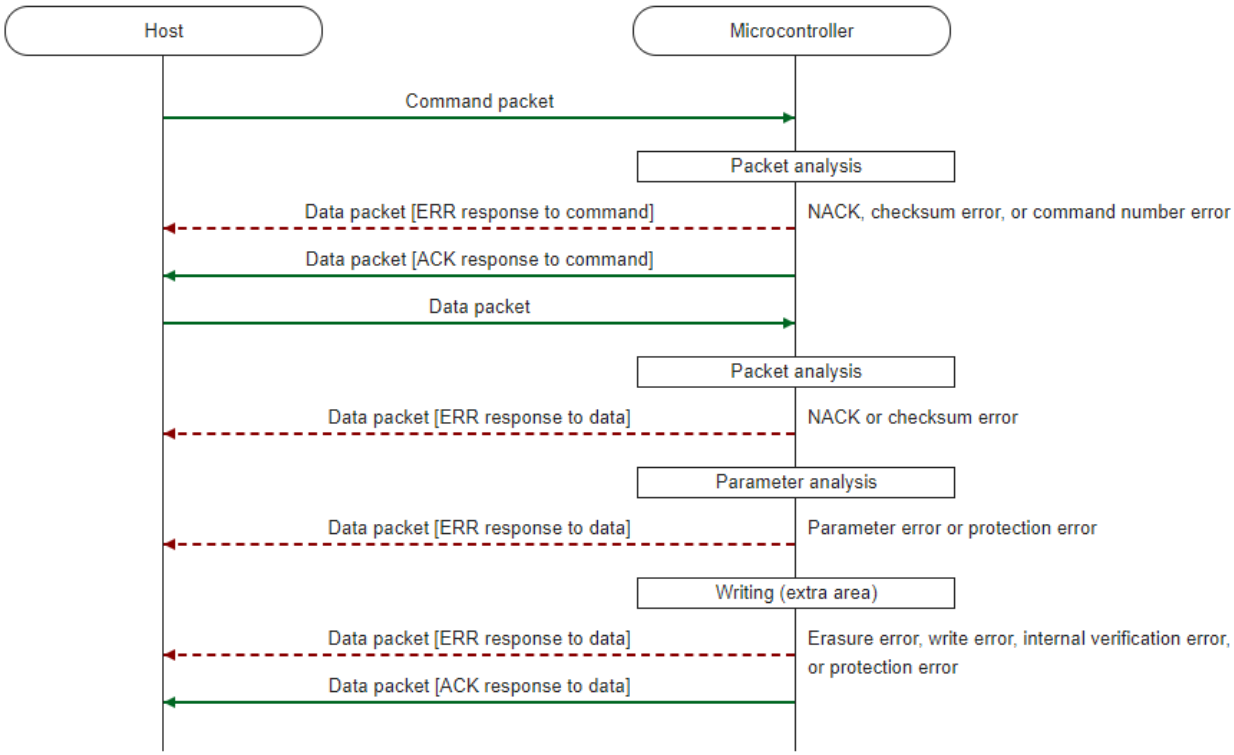


Figure 6-12 Security Set Command

6.9.2 List of Packets to be Transmitted

Table 6-56 to Table 6-61 respectively show the command packet and five types of data packet (for ACK response to command, ERR response to command, security data, ACK response to data, and ERR response to data).

Table 6-56 Command Packet

STX	(1 byte)	01h
LEN	(1 byte)	01h
CMD	(1 byte)	A0h
SUM	(1 byte)	5Fh
ETX	(1 byte)	03h

Table 6-57 Data Packet [ACK Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-58 Data Packet [ERR Response to Command]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

Table 6-59 Data Packet

SOH	(1 byte)	02h	
LEN	(1 byte)	08h	
FLG	(1 byte)	Security flag	[Bit 0] 1 [Bit 1] BTPR (boot block cluster protection) 0: Rewriting of boot cluster 0 is disabled. 1: Rewriting of boot cluster 0 is enabled. [Bit 2] SEPR (sector erase protection) 0: Block erasure is disabled. 1: Block erasure is enabled. [Bit 3] 1 [Bit 4] WRPR (write protection) 0: Writing is disabled. 1: Writing is enabled. [Bit 5] 1 [Bit 6] 1 [Bit 7] TEPR (test mode protection) 0: Test mode is disabled. 1: Test mode is enabled.
RSV	(1 byte)	Reserved	Not used. Any value from 00h to FFh can be specified.

SSL	(1 byte)	FSW start block (low)	SSL [bits 7 to 0] FSW start block (value of FSWS in the extra area) Example: Transferred parameters: SSL = 02h, SSH = 00h FSW start block = block 2 (002h is written to FSWS.)
SSH	(1 byte)	FSW start block (high)	SSH [bits 1 and 0] FSW start block (value of FSWS in the extra area) SSH [bits 7 to 2] All 0 Example: Transferred parameters: SSL = 02h, SSH = 00h FSW start block = block 2 (002h is written to FSWS.)
SEL	(1 byte)	FSW end block (low)	SEL [bits 7 to 0] FSW end block (value of FSWE – 1 in the extra area) Example: Transferred parameters: SEL = 40h, SEH = 01h FSW end block = block 320 (141h is written to FSWE.)
SEH	(1 byte)	FSW end block (high)	SEH [bits 1 and 0] FSW end block (value of FSWE – 1 in the extra area) SEH [bits 7 to 2] All 0 Example: Transferred parameters: SEL = 40h, SEH = 01h FSW end block = block 320 (141h is written to FSWE.)
RSV	(1 byte)	Reserved	Not used. Any value from 00h to FFh can be specified.
RSV	(1 byte)	Reserved	Not used. Any value from 00h to FFh can be specified.
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-60 Data Packet [ACK Response to Data]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-61 Data Packet [ERR Response to Data]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.9.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

- ACK is sent if an error has not occurred.

After successful completion of the packet analysis, a data packet is received and packet analysis is executed.

- Reception of STX is recognized as the start of a data packet.
  If the received value is not STX, the firmware waits until STX arrives.
- If the received data packet does not end with ETX, the STS is set to "NACK".
- If the value of SUM in the received data packet does not match the calculated checksum value, the STS is set to "checksum error".
- If either of the above errors has occurred, execution returns to the command wait state with no further processing being done.

After successful completion of the packet analysis, parameter analysis is executed.

- When FSW start block > FSW end block, the parameter error code is sent.
- When the FSW end block is outside the code flash memory area, the parameter error code is sent.
- If modifying SEPR, WRPR, or BTPR from 0 to 1 is attempted, the protection error code is sent.
- If any of the above error has occurred, execution returns to the command wait state with no further processing being done.

Note: The flash memory remains in the same state as before the command was received.

After successful completion of the parameter analysis, the flash options are rewritten.

- If an erasure error has occurred (ERER has been set in FSASTL) during the rewriting processing, the erasure error code is sent.
- If a write error has occurred (WRER has been set in FSASTL) during the rewriting processing, the write error code is sent.
- If an internal verification error has occurred (IVER has been set in FSASTL) during the rewriting processing, the internal verification error code is sent.
- If a sequencer error has occurred (SEQER or ESEQER has been set in FSASTL) during the rewriting processing, the protection error code is sent.
- If any of the above errors has occurred, execution returns to the command wait state.

Notes: 1. If an erasure error or a write error has occurred, the state of the flash options becomes undefined.
2. If a protection error has occurred, the flash memory remains in the same state as before the command was received.
3. If another command is executed immediately after an erasure error, a write error, or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
   If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after an erasure error, a write error, or a protection error.

- On successful completion of the rewriting processing, ACK is sent and execution returns to the command wait state.

Note: The flash option area of the flash memory is set to the written state.

6.9.4 Status Information Returned from the Microcontroller

Table 6-62 lists the status information.

Table 6-62 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
Prohibited setting of the last block number in the boot area was attempted.	Parameter error
Prohibited FSW start or end block setting was attempted.	Parameter error
Prohibited security flag setting was attempted.	Protection error
An erasure error has occurred.	Erasure error
A write error has occurred.	Write error
An internal verification error has occurred.	Internal verification error
A sequencer error has occurred.	Protection error
Processing has been successfully completed.	ACK

6.9.5 Notes on Using this Command

Table 6-63 is a list of notes on using this command.

Table 6-63 Notes on Using this Command

(1)	The security settings made by this command take effect as soon as the command execution is completed. (There is no need to reset the device to make the settings effective after this command.)
(2)	When SEPR or BTPR has been set to 0, the Security Release command cannot be executed. In this case, no security settings including those for these flags can be erased.
(3)	If WRPR is set to 0 while bit 0 at address 000C4h is set to 1 and the last block or the block before last in the data flash memory is in the erased state, the value of WRPR cannot be restored to 1.

6.10 Security Get Command

This command is used to send the current settings of the security flags to the host.

6.10.1 Sequence Diagram

Figure 6-13 is a diagram of the Security Get command sequence.

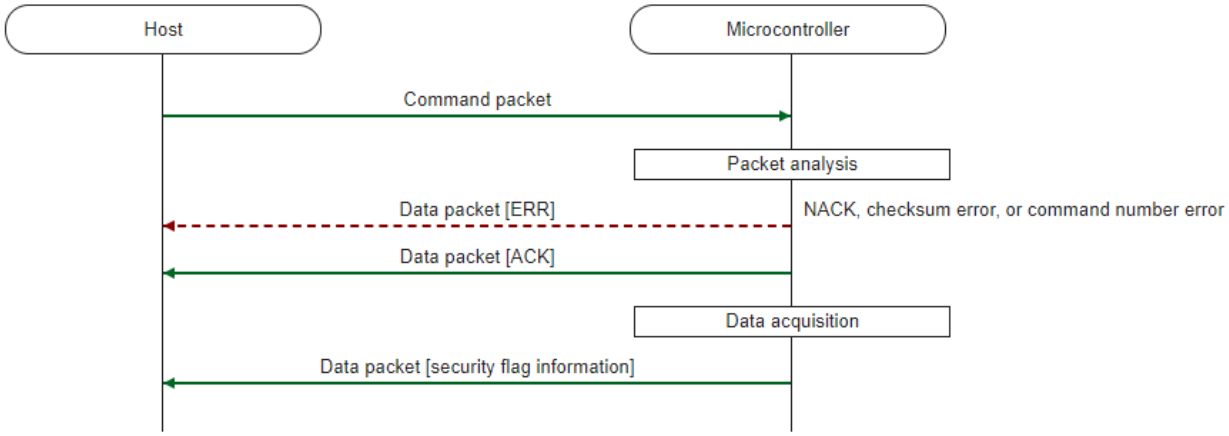


Figure 6-13 Security Get Command

6.10.2 List of Packets to be Transmitted

Table 6-64 to Table 6-67 respectively show the command packet and three types of data packet (for security flag information, ACK, and ERR).

Table 6-64 Command Packet

SOH	(1 byte)	01h
LEN	(1 byte)	01h
CMD	(1 byte)	A1h
SUM	(1 byte)	5Eh
ETX	(1 byte)	03h

Table 6-65 Data Packet [Security Flag Information]

STX	(1 byte)	02h	
LEN	(1 byte)	08h	
FLG	(1 byte)	Security flag	[Bit 0] BTFLG (boot flag) 0: Booting is from boot cluster 1. 1: Booting is from boot cluster 0. [Bit 1] BTPR (Boot block cluster protection) 0: Rewriting of boot cluster 0 is disabled. 1: Rewriting of boot cluster 0 is enabled. [Bit 2] SEPR (sector erase protection) 0: Block erasure is disabled. 1: Block erasure is enabled. [Bit 3] 1 [Bit 4] WRPR (write protection) 0: Writing is disabled. 1: Writing is enabled. [Bit 5] 1 [Bit 6] 1 [Bit 7] TEPR (test mode protection) 0: Test mode is disabled. 1: Test mode is enabled.
BLB	(1 byte)	Last block number in boot area	Number of the last block in the boot area
SWS	(2 bytes)	FSW start block	[Bits 9 to 0] FSW start block (value of FSWS in the extra area) [Bits 15 to 10] All 0 Order of transfer: Low → High Example: Transferred parameters = 02h, 00h FSW start block = block 2 (FSWS value is 002h.)
SWE	(2 bytes)	FSW end block	[Bits 9 to 0] FSW end block (value of FSWE – 1 in the extra area) [Bits 15 to 10] All 0 Order of transfer: Low → High Example: Transferred parameters = 40h, 01h FSW end block = block 320 (FSWE value is 141h.)
RSV	(1 byte)	Reserved	Not used. "FFh"
RSV	(1 byte)	Reserved	Not used. "FFh"
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-66 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-67 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.10.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, ACK is sent in response to the command packet.

- ACK is sent in response to the command packet.

After ACK is sent, the security flag information is read.

- The information read from the security flag is sent and execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

6.10.4 Status Information Returned from the Microcontroller

Table 6-68 lists the status information.

Table 6-68 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
Processing has been successfully completed.	ACK

6.11 Security Release Command

This command is used to erase the currently set security information (flash options in the extra area). Erasure of the flash option settings is only possible when the code flash memory and data flash memory areas are blank.

The Security Release command can only be executed when both of the following conditions are satisfied.

- Neither the "block erasure protection" nor "boot cluster 0 protection" is set.
- The code flash memory and data flash memory are blank.

6.11.1 Sequence Diagram

Figure 6-14 is a diagram of the Security Release command sequence.

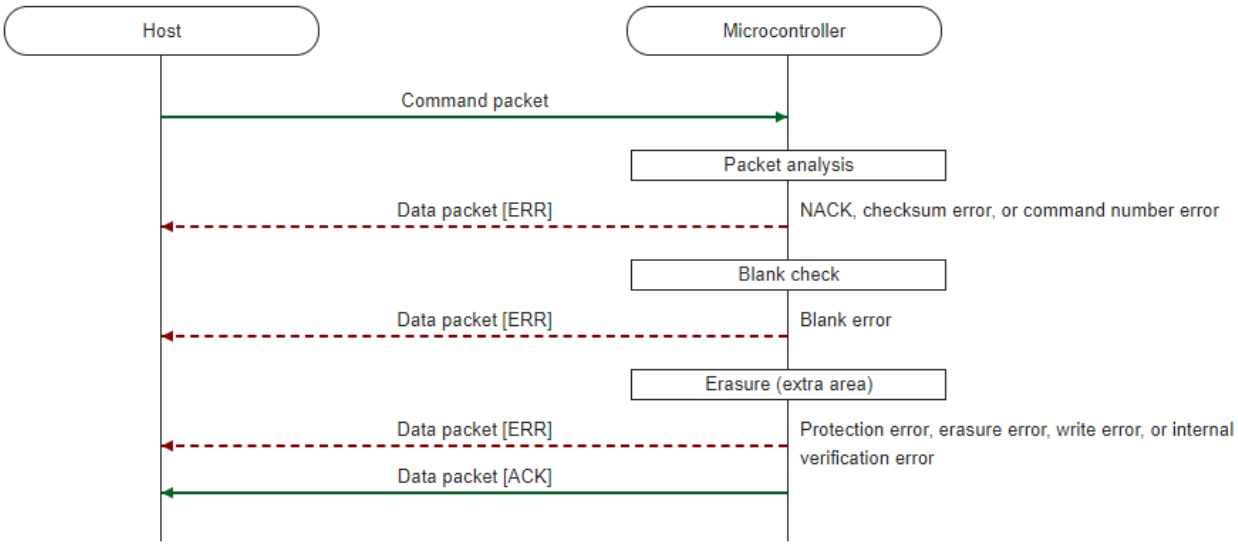


Figure 6-14 Security Release Command

6.11.2 List of Packets to be Transmitted

Table 6-69 to Table 6-71 respectively show the command packet and two types of data packet (for ACK and ERR).

Table 6-69 Command Packet

SOH	(1 byte)	01h
LEN	(1 byte)	01h
CMD	(1 byte)	A2h
SUM	(1 byte)	5Dh
ETX	(1 byte)	03h

Table 6-70 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-71 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.11.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, blank checking is executed.

- If the code flash memory or data flash memory has a non-blank area, the blank error code is sent.

Notes: 1. The flash memory remains in the same state as before the command was received.
2. If another command is executed immediately after a blank error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after a blank error.

After successful completion of blank checking, the flash options are erased.

- If an erasure error has occurred (ERER has been set in FSASTL) during the erasure processing, the erasure error code is sent.
- If an internal verification error has occurred (IVER has been set in FSASTL) during the erasure processing, the internal verification error code is sent.
- If a write error has occurred (WRER has been set in FSASTL) during the erasure processing, the write error code is sent.
- If a sequencer error has occurred (SEQER or ESEQER has been set in FSASTL) during the erasure processing, the protection error code is sent.
- If any of the above errors has occurred, execution returns to the command wait state.

Notes: 1. If an erasure error, a write error, or an internal verification error has occurred, the state of the flash options becomes undefined.
2. If a protection error has occurred, the flash options remain in the same state as before the command was received.
3. If another command is executed immediately after an erasure error, a write error, an internal verification error, or a protection error and if the command execution is unsuccessful, an incorrect error code may be returned.
If the return of an incorrect error code may produce any problem, reset the device before executing another command immediately after an erasure error, a write error, an internal verification error, or a protection error.

- On successful completion of the erasure, ACK is sent and execution returns to the command wait state.

Note: The flash options of the flash memory are set to the erased state.

6.11.4 Status Information Returned from the Microcontroller

Table 6-72 lists the status information.

Table 6-72 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
The code flash memory or data flash memory has a non-blank area.	Blank error
An erasure error has occurred.	Erasure error
A write error has occurred.	Write error
An internal verification error has occurred.	Internal verification error
A sequencer error has occurred.	Protection error
Processing has been successfully completed.	ACK

6.11.5 Notes on Using this Command

Table 6-73 is a list of notes on using this command.

Table 6-73 Notes on Using this Command

(1)	The security settings erased by this command become ineffective as soon as the command execution is completed. (There is no need to reset the device to make the security ineffective after this command.)
(2)	To erase the code flash memory and data flash memory before executing the Security Release command, use the procedure shown in the flowchart in section 7.5, Erasing Security Information.

6.12 Checksum Command

This command is used to return the checksum value for an area specified as an address range. The code flash memory or data flash memory is specifiable but specifying a range that extends beyond the boundary of the code flash memory or data flash memory is not allowed. The addresses where the target area starts and ends must be the addresses where given blocks start and end.

The result of checksum calculation may differ from the expected value depending on the setting of the security option byte (IDRDEN).

Calculation of checksum:

The initial value of the checksum is set to 0000h, and the data in the specified area are read in byte units and subtracted from the current checksum value in order. Generation of borrowing during the calculation is ignored.

6.12.1 Sequence Diagram

Figure 6-15 is a diagram of the Checksum command sequence.

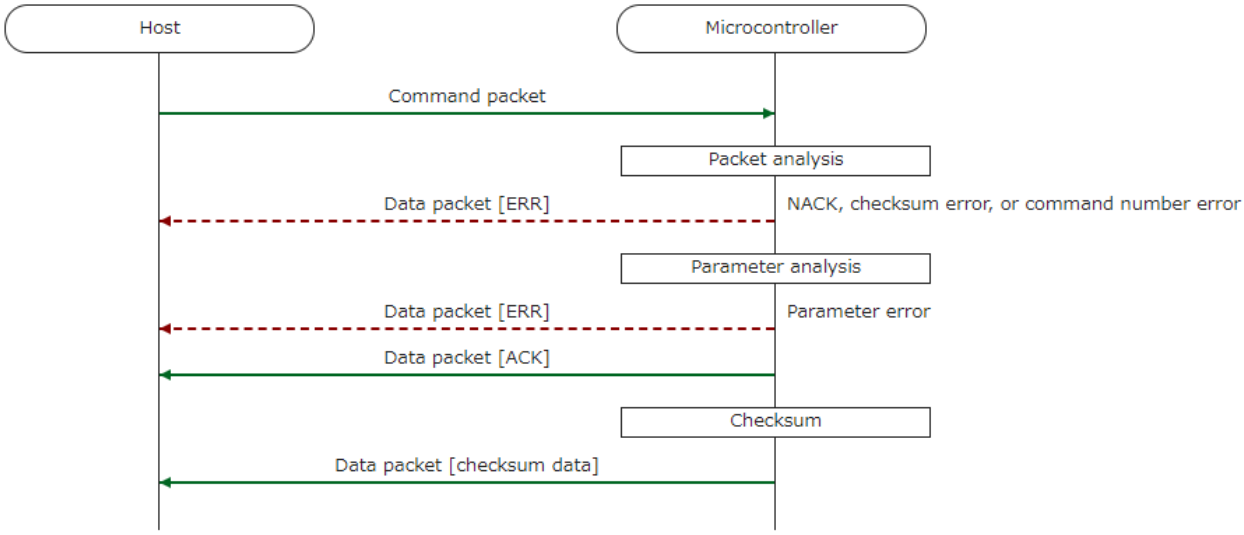


Figure 6-15 Checksum Command

6.12.2 List of Packets to be Transmitted

Table 6-74 to Table 6-77 respectively show the command packet and three types of data packet (for checksum data, ACK, and ERR).

Table 6-74 Command Packet

SOH	(1 byte)	01h	
LEN	(1 byte)	07h	
CMD	(1 byte)	B0h	
SAD	(3 bytes)	Start address	Order of transfer: Low → Middle → High Example: 23400h = 00h → 34h → 02h
EAD	(3 bytes)	End address	Order of transfer: Low → Middle → High Example: 3FFFFh = FFh → FFh → 03h
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-75 Data Packet [Checksum Data]

STX	(1 byte)	02h	
LEN	(1 byte)	02h	
DAT	(2 bytes)	Checksum data	Order of transfer: Low → High Example: 0123h = 23h → 01h
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-76 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-77 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.12.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

After successful completion of the packet analysis, parameter analysis is executed.

- If SAD is greater than EAD, the parameter error code is sent.
- If SAD or EAD is neither in the code flash memory area nor in the data flash memory area, the parameter error code is sent.
- If the area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory, the parameter error code is sent.
- If either or both of SAD and EAD are not the correct addresses where a block starts or ends, the parameter error code is sent.
- If any of the above errors has occurred, execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

- If none of the above errors has occurred, ACK is sent.

After successful completion of the parameter analysis, the checksum is calculated.

- The calculated checksum value is sent and execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

6.12.4 Status Information Returned from the Microcontroller

Table 6-78 lists the status information.

Table 6-78 Status Information

Condition	STS
The received command packet does not end with ETX.	NACK
The value of SUM in the received command packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received command packet does not match that defined for the given command.	NACK
SAD is greater than EAD.	Parameter error
SAD or EAD is neither in the code flash memory area nor in the data flash memory area.	Parameter error
The area specified by SAD and EAD extends beyond the boundary of the code flash memory or data flash memory.	Parameter error
SAD or EAD is not the correct address of the start or end of a block, respectively.	Parameter error
Checksum calculation has started.	ACK

6.13 Silicon Signature Command

This command is used to have the device send its signature information to the host.

6.13.1 Sequence Diagram

Figure 6-16 is a diagram of the Silicon Signature command sequence.

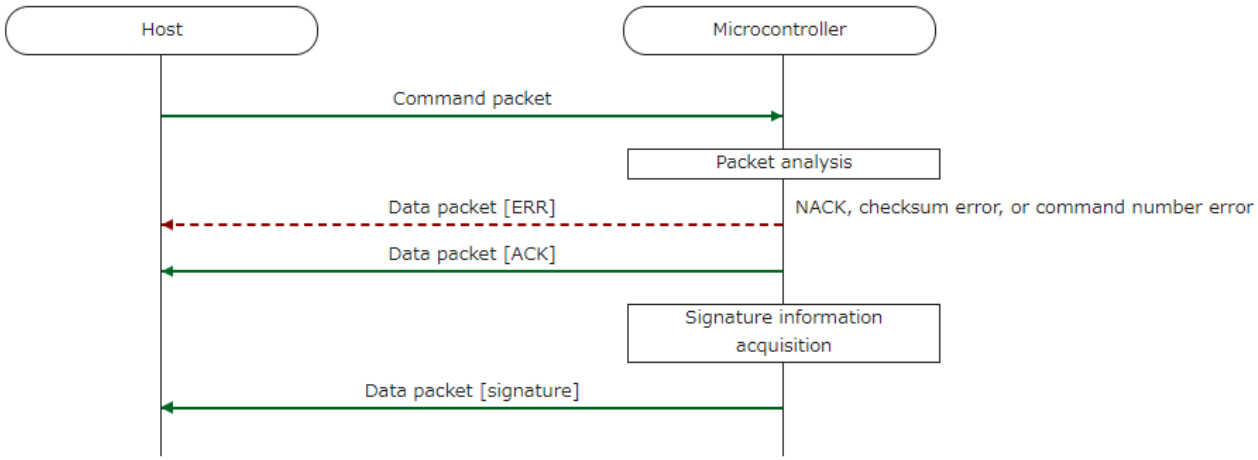


Figure 6-16 Silicon Signature Command

6.13.2 List of Packets to be Transmitted

Table 6-79 to Table 6-82 respectively show the command packet and three types of data packet (for signature, ACK, and ERR).

Table 6-79 Command Packet

SOH	(1 byte)	01h
LEN	(1 byte)	01h
COM	(1 byte)	C0h
SUM	(1 byte)	3Fh
ETX	(1 byte)	03h

Table 6-80 Data Packet [Signature]

STX	(1 byte)	02h	
LEN	(1 byte)	16h	
DVC	(3 bytes)	Device function code	10000Bh: RL78/F2x Order of transfer: High → Middle → Low Example: RL78/F2x = 10h → 00h → 0Bh
DEV	(10 bytes)	Device name (ASCII code)	Example: R7F100GAJ = 52h, 37h, 46h, 31h, 30h, 30h, 47h, 41h, 4Ah, 20h
CFE	(3 bytes)	Last address of code flash memory area	Order of transfer: Low → Middle → High Example: F0FFFh = FFh → 0Fh → 0Fh
DFE	(3 bytes)	Last address of data flash memory area	Order of transfer: Low → Middle → High Example: F4FFFh = FFh → 4Fh → 0Fh If the device does not incorporate data flash memory, these values are all fixed to 00H. Example: Not incorporated = 00h → 00h → 00h
FWV	(3 bytes)	Boot firmware version	Example: V1.23 = 1h, 2h, 3h
SUM	(1 byte)	Checksum value	
ETX	(1 byte)	03h	

Table 6-81 Data Packet [ACK]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	06h (ACK)
SUM	(1 byte)	F9h
ETX	(1 byte)	03h

Table 6-82 Data Packet [ERR]

STX	(1 byte)	02h
LEN	(1 byte)	01h
STS	(1 byte)	Status code
SUM	(1 byte)	Checksum value
ETX	(1 byte)	03h

6.13.3 Procedure of Processing

Upon reception of the command packet, packet analysis is executed.

- If the packet analysis has been successfully completed, ACK is sent.

After ACK is sent, the signature information is sent.

- Signature information is sent and execution returns to the command wait state.

Note: The flash memory remains in the same state as before the command was received.

6.13.4 Status Information Returned from the Microcontroller

Table 6-83 lists the status information.

Table 6-83 Status Information

Condition	STS
The received packet does not end with ETX.	NACK
The value of SUM in the received packet does not match the calculated checksum value.	Checksum error
Execution of the given command is not allowed in the current phase.	Command number error
The value of LEN in the received packet does not match that defined for the given command.	NACK
Processing has been successfully completed.	ACK

7. Flowcharts

7.1 Initial Communications

Figure 7-1 is a flowchart of initial communications.

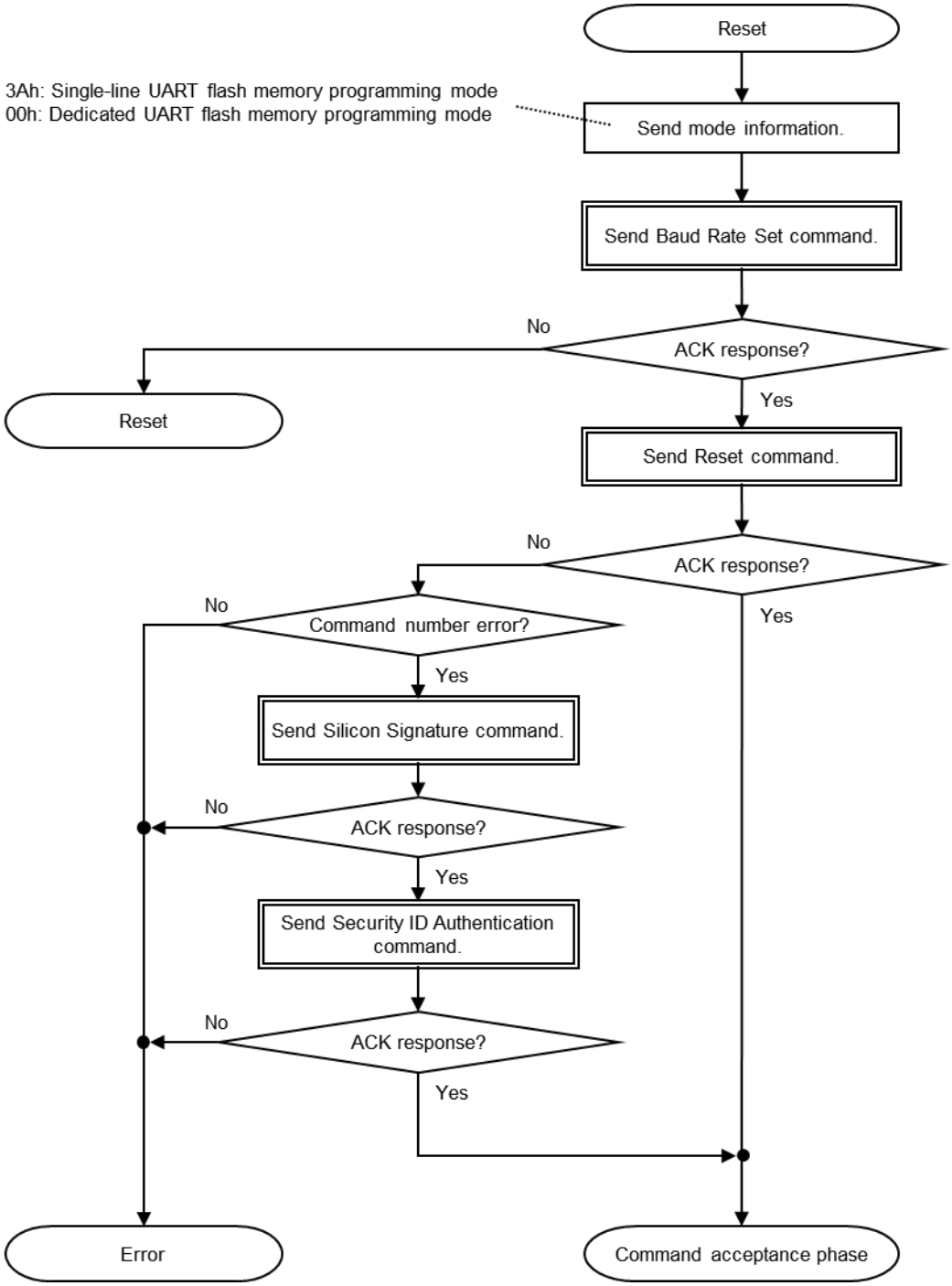


Figure 7-1 Initial Communications

7.2 Acquiring Signature Information

Figure 7-2 is a flowchart of the acquisition of signature information.

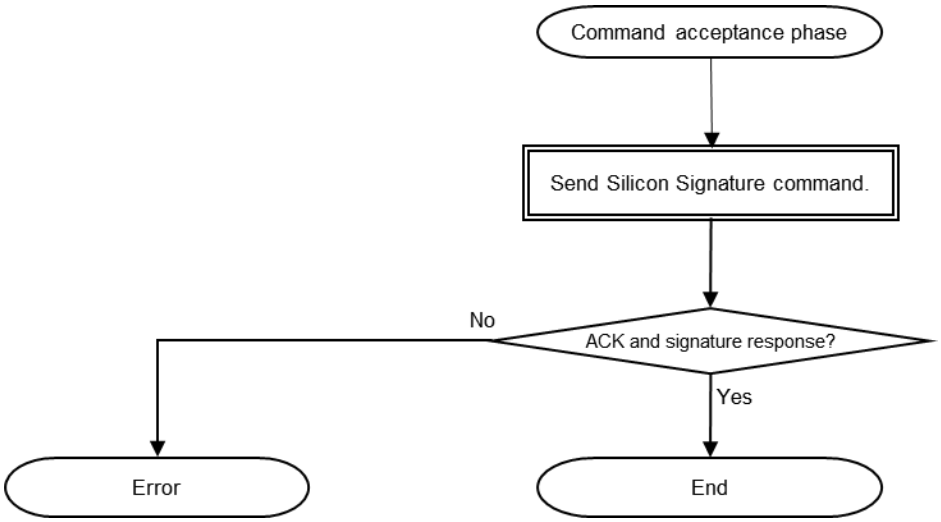


Figure 7-2 Acquiring Signature Information

7.3 Rewriting Code Flash Memory or Data Flash Memory

Figure 7-3 is a flowchart of rewriting the code flash memory or data flash memory.

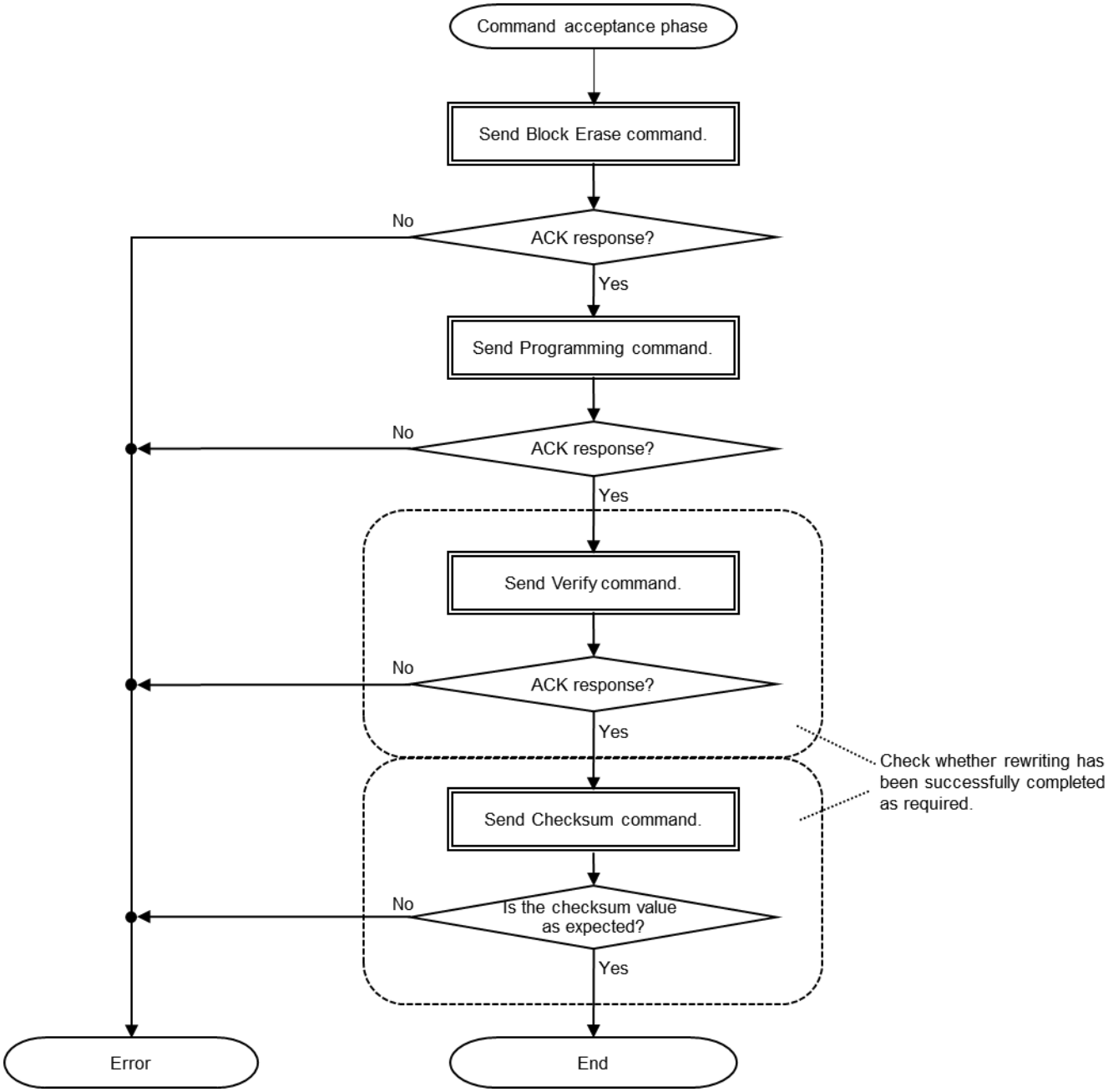


Figure 7-3 Rewriting Code Flash Memory or Data Flash Memory

7.4 Rewriting Security Flags and Flash Shield Window Settings

Figure 7-4 is a flowchart of rewriting security flag and flash shield window (FSW) settings.

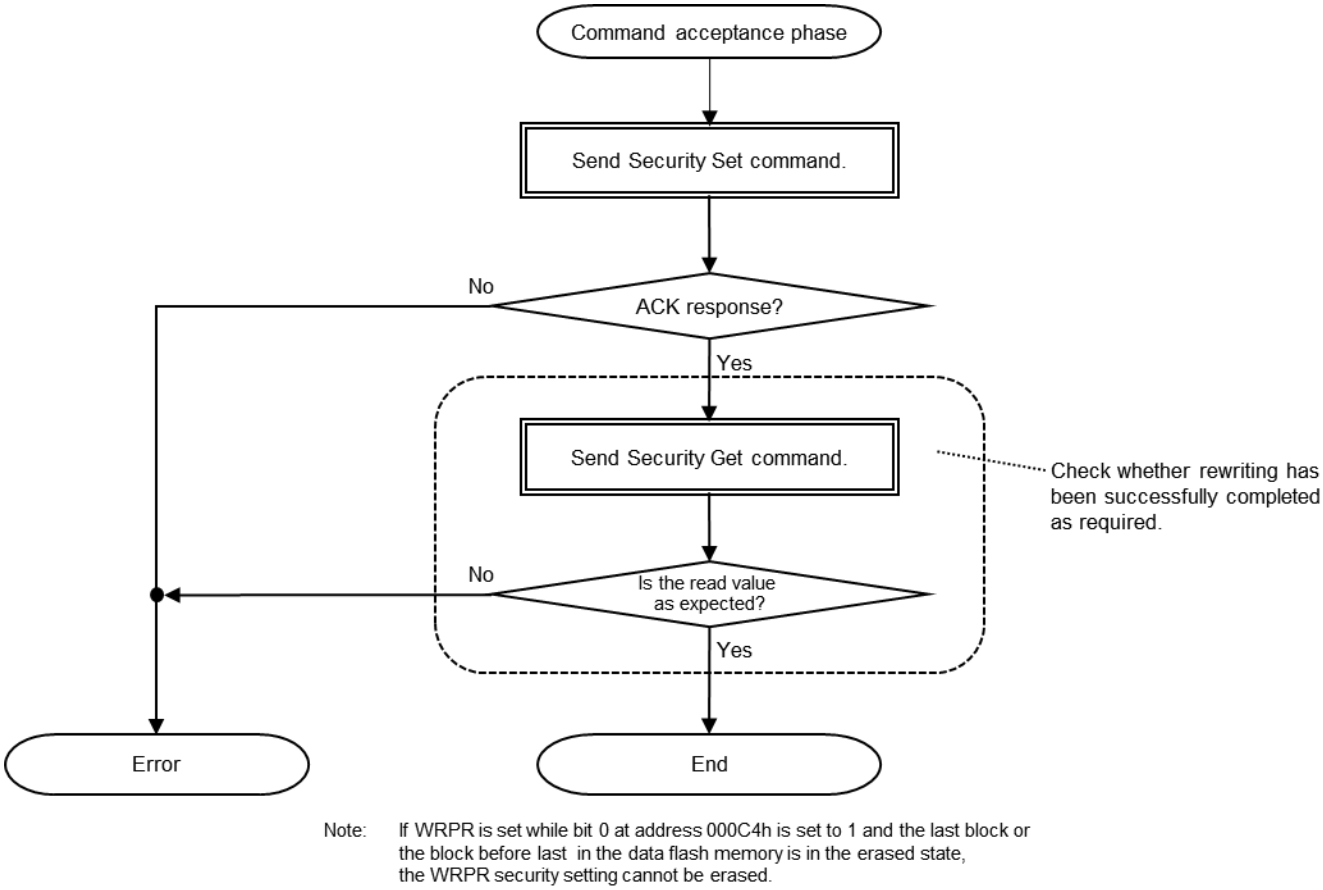


Figure 7-4 Rewriting Security Flags and FSW Settings

7.5 Erasing Security Information

Figure 7-5 is a flowchart of erasing the security information.

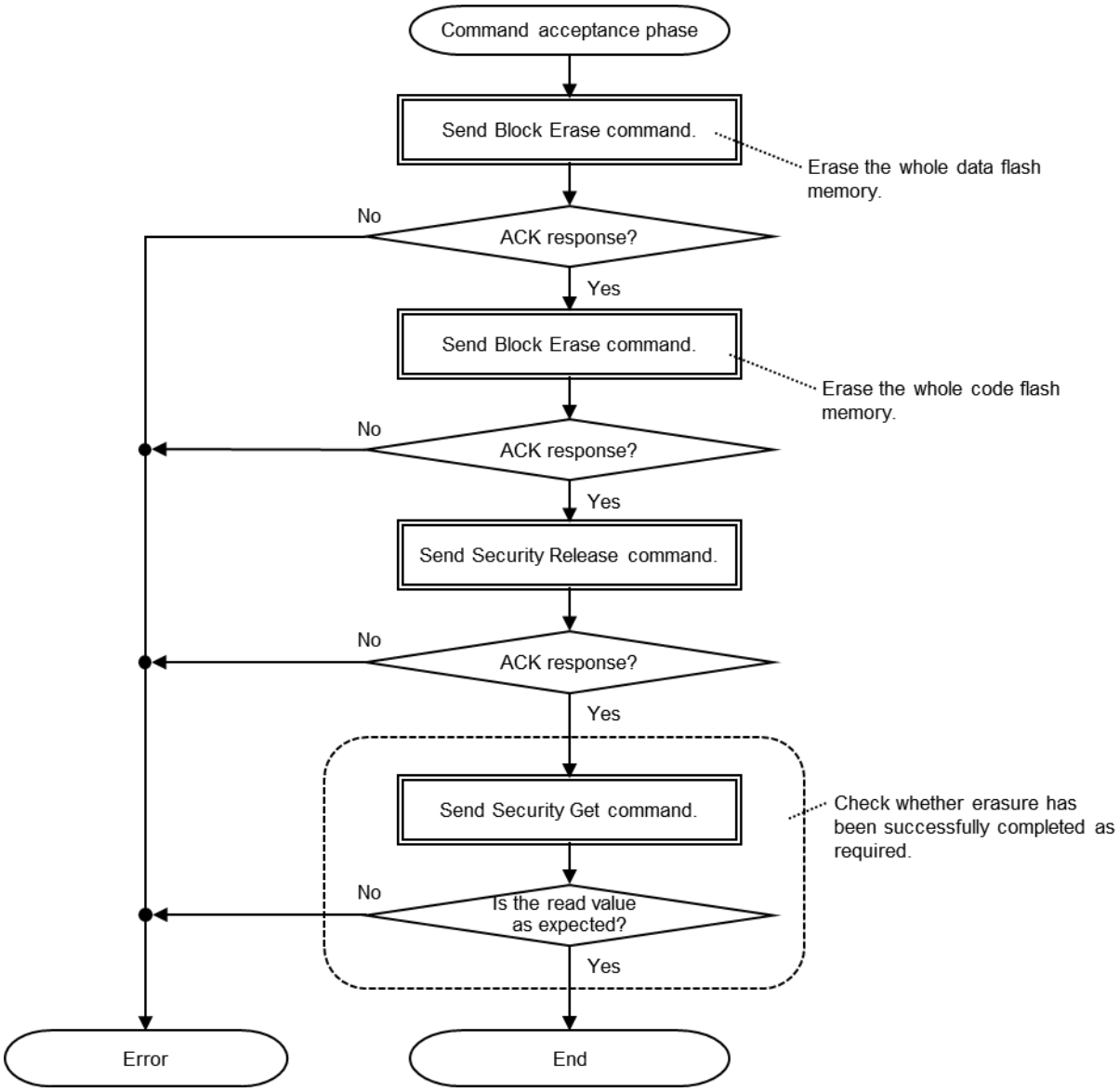


Figure 7-5 Erasing Security Information

7.6 Cancelling Command Processing

For a command involving data packet transmission from the programmer, the command processing can be cancelled by the intentional transmission of an abnormal data packet, which returns execution to the command acceptance phase.

Figure 7-6 is a flowchart of cancelling command processing.

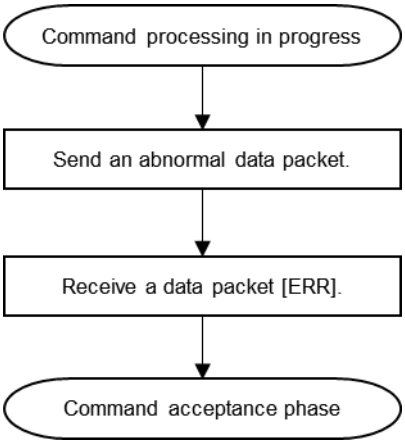


Figure 7-6 Cancelling Command Processing

Example of an Abnormal Data Packet:

Command	Timing of Cancellation	Example of an Abnormal Data Packet
Verify command	Transmission of verification data	
Programming command	Transmission of data	

STX	(1 byte)	02h
LEN	(1 byte)	01h
DAT	(1 byte)	00h
SUM	(1 byte)	FFh
EXT or ETB	(1 byte)	FFh (abnormal value)

7.7 Timeout

The following figure is a flowchart of judging the timeout of a response and gives rough standards of timeout times.

Figure 7-7 is a flowchart of judging a timeout.

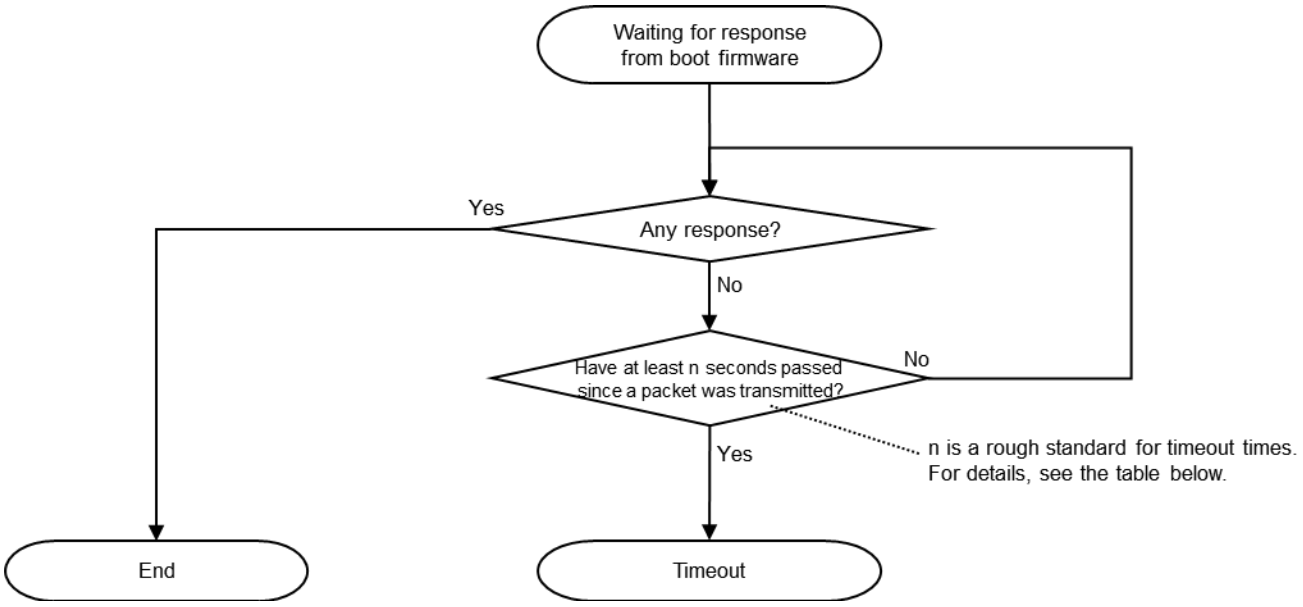


Figure 7-7 Timeout

Rough Standards of Timeout Times:

Command	Timing	Rough Standards of Timeout Times (ms)
Checksum command	ACK response	1000
	Checksum data response	(12 ÷ CPU operating frequency (MHz)) × specified range (in 256-byte units) Example: When the CPU operating frequency is 2 MHz and an area of 128 Kbytes is specified (12 ÷ 2) × 512 = 3072 (ms)
Other commands	All packet responses	1000

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jun.30.22	—	First edition issued
2.00	Aug.10.22	12	Changed and corrected typos (Table 4-1 Specifications of UART Communications)
		13	Changed (Table 4-2 Specifications of Communications when the Mode Information is Incorrect)
		49	Added Timing Chart to Secure Programming command (Figure 6-7, Table 6-43)
		53	Changed (Table 6-49 Specifications of UART Communications)
		58	Changed (Table 6-55 Specifications of Communications)

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.