R8C/Mx Series, LAxA Group

Standard Serial I/O Mode Serial Protocol Specification

R01AN0216EJ0110
Rev. 1.10
Feb. 15, 2013

Abstract

This document describes the R8C/Mx Series, LAxA Group serial protocol specification. The boot program stored in the boot ROM area prior to MCU shipment can control the flash memory by communicating with a serial programmer. Standard serial I/O mode 2 or standard serial I/O mode 3 can be selected for communication.

Introduction

This specification defines the following:

- Boot program
- Initial settings
- Control commands
- Timings

Applicable products: R8C/Mx Series, LAxA Group

Applicable boot program version: 1.00 or later

1. Boot Program

The program in the boot ROM area operates when the MODE pin is set to low and reset is deasserted. This is called a boot program. To enter boot mode, hold the MODE pin low for 30 ms or more before and after reset is deasserted (refer to Figure 2.1).

1.1 Operating Environment

(1) Standard Serial I/O Mode 2
Serial interface: Connecting with a serial programmer in clock asynchronous serial I/O mode
Supply voltage: 2.7 V ≤ VCC ≤ 5.5 V
CPU clock and count source for communication: Typ. 18.432 MHz generated by the high-speed on-chip oscillator

(2) Standard Serial I/O Mode 3
Serial interface: Connecting with a serial programmer in special clock asynchronous serial I/O mode
Supply voltage: 2.7 V ≤ VCC ≤ 5.5 V
CPU clock and count source for communication: Typ. 20 MHz generated by the high-speed on-chip oscillator

Communication may not be available due to frequency fluctuations of the high-speed on-chip oscillator, the board wire layout, or other usage conditions. Refer to the User's Manual: Hardware or Renesas Electronics website for high-speed on-chip oscillator electrical characteristics. Refer to the User's Manual: Hardware for erase and program voltage electrical characteristics.

Careful evaluation on the user system is recommended when communicating or reprogramming.

1.2 Boot Program Content

(1) Initial settings
(2) Initial communication with a serial programmer
(3) Control commands
Flash control commands (program, erase, read)
Various setting commands (such as communication speed setting, status read)

1.3 Communication with a Serial Programmer

Standard serial I/O mode 2 or standard serial I/O mode 3 can be selected for communication with a serial programmer. Standard serial I/O mode 2 is an asynchronous communication format. Standard serial I/O mode 3 is an asynchronous half duplex communication format. Figure 1.1 shows the communication formats.

The transfer data format is as follows:

Start bit: 1 bit
Transfer data: 8 bits
Parity bit: Not used
Stop bit: Transmission: 2 bits, Reception: 1 bit

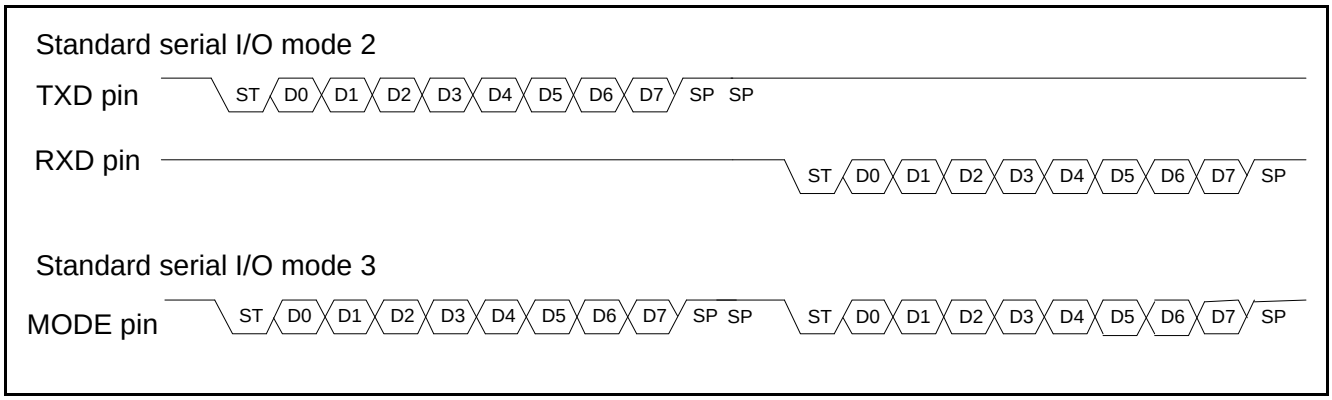


Figure 1.1 Communication Formats

1.4 Assigned Pins

(1) MODE pin

Standard serial I/O mode 2 or standard serial I/O mode 3 is selected according to the MODE pin level after reset is deasserted. This pin also functions as TXD or RXD when standard serial I/O mode 3 is selected. When using standard serial I/O mode 3, pull up the MODE pin with approximately 5 kΩ.

(2) TxD and RxD pins

These pins are the transmit and receive pins in standard serial I/O mode 2. They are not used in standard serial I/O mode 3.

(3) RESET pin

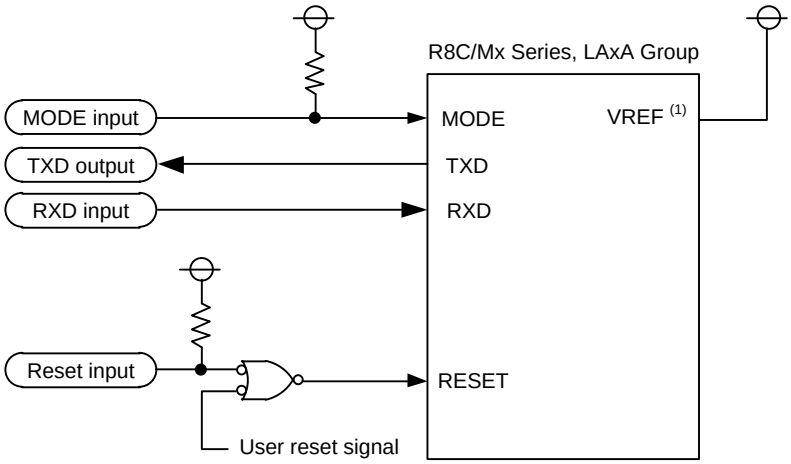
This pin controls a reset via a serial programmer.

(4) Vcc and Vss pins

When outputting a high or low signal from a serial programmer, use the voltage according to the MCU's high or low input voltage levels.

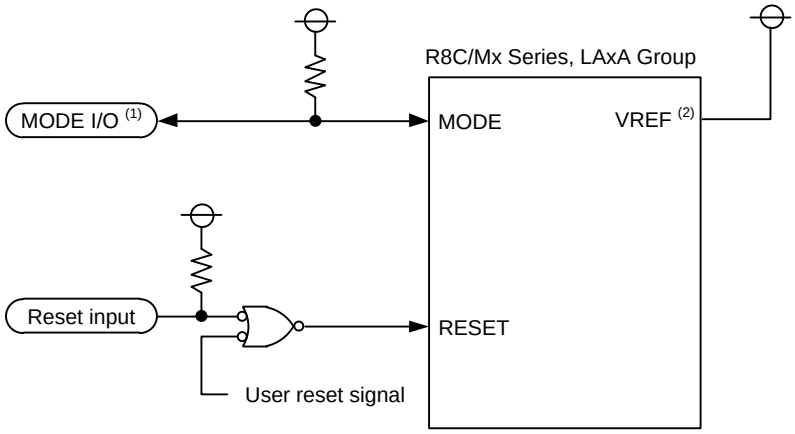
(5) VREF pin (only for R8C/LAxA Group)

Apply the reference voltage to the VREF pin as shown in the A/D Converter Characteristics in the Electrical Characteristics chapter in the User's Manual: Hardware or connect the VREF pin to VCC.



Notes:
1. As shown in the A/D Converter Characteristics in the Electrical Characteristics chapter in the User's Manual: Hardware, apply the reference voltage to the VREF pin or connect the VREF pin to VCC only for the R8C/LAxA Group.
2. When operating with the on-chip oscillator clock, it is not necessary to connect an oscillation circuit.
3. Refer to Standard Serial I/O Mode in Flash Memory chapter in the User's Manual: Hardware for connection.

Figure 1.2 Connection Example in Standard Serial I/O Mode 2



Notes:
1. N-channel open-drain output
2. As shown in the A/D Converter Characteristics in the Electrical Characteristics chapter in the User's Manual: Hardware, apply the reference voltage to the VREF pin or connect the VREF pin to VCC only for the R8C/LAxA Group.
3. When operating with the on-chip oscillator clock, it is not necessary to connect an oscillation circuit.
4. Refer to Standard Serial I/O Mode in Flash Memory chapter in the User's Manual: Hardware for connection.

Figure 1.3 Connection Example in Standard Serial I/O Mode 3

2. Initial Settings

As an initial operation of boot program, perform the following operations sequentially.

(1) Decide a communication format
(2) Adjust the bit rate

2.1 Deciding Communication Format

Communication format is selected from either standard serial I/O mode 2 or standard serial I/O mode 3. The MODE pin level is checked twice – Typ. 200 ms and Typ. 230 ms after the reset is deasserted. Then, the MCU enters one of the modes described below. Fix the MODE pin level within 100 ms after reset is deasserted, and between 210 and 220 ms after reset is deasserted. The timing diagram is shown in Figure 2.1 Timing to Decide Communication Format. Refer to the User's Manual: Hardware for tw (por1).

	After Typ. 200 ms	After Typ. 230 ms
Standard serial I/O mode 2 (typ. 18.432 MHz)	Low	Low
Standard serial I/O mode 3 (typ. 20 MHz)	High	High

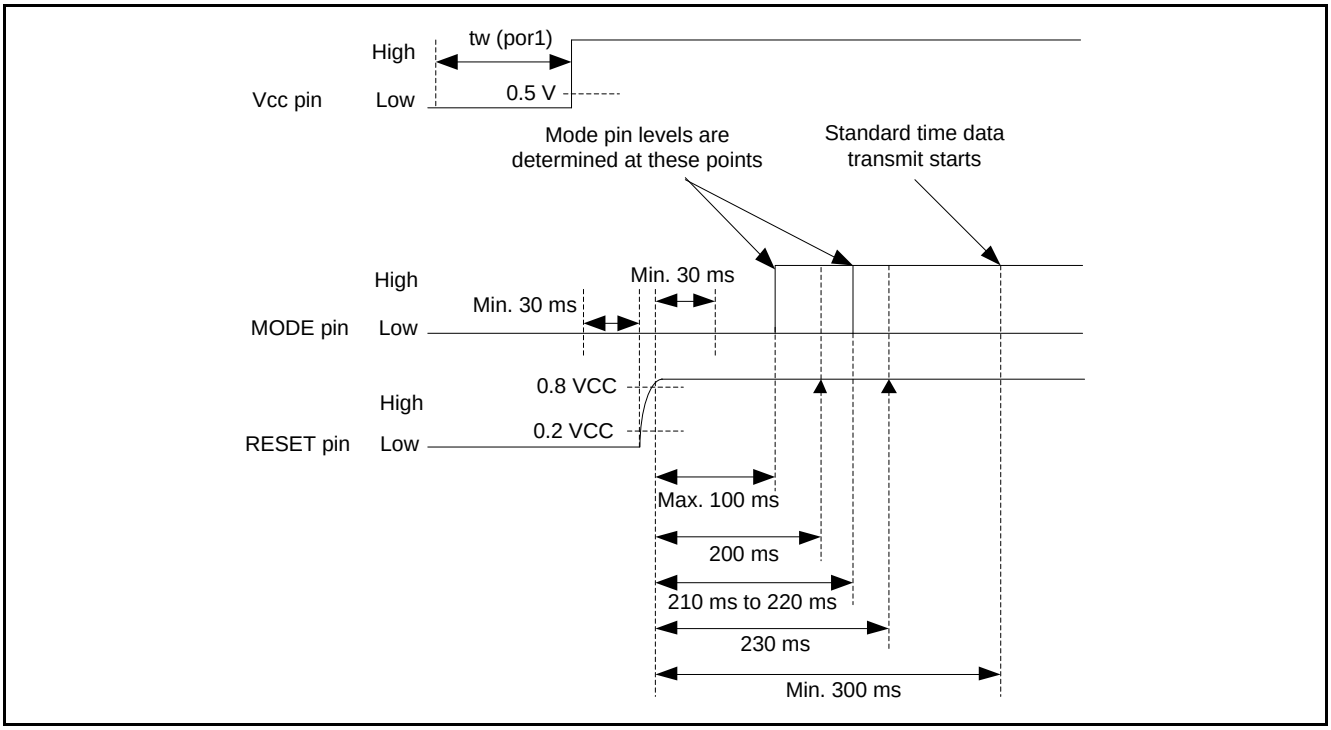


Figure 2.1 Timing to Decide Communication Format

2.2 Adjusting Bit Rate

The bit rate is adjusted to 9600 bps by receiving the standard time data (00h) 16 times at a bit rate of 9600 bps and the bite rate 9600 command (B0h) from the serial programmer. When the bit rate 9600 command (B0h) is received successfully, the MCU returns the command. Figure 2.2 shows the adjustment procedure of bit rate.

Transmit the standard time data at least 300 ms after reset is deasserted.

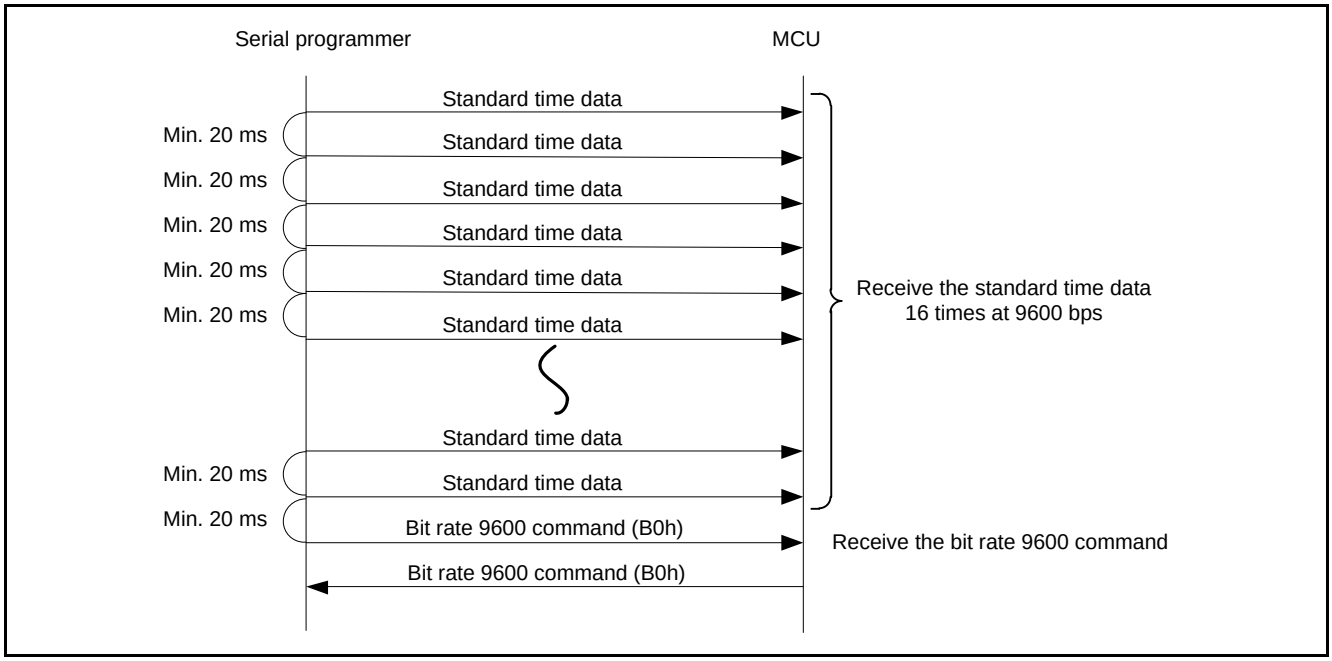


Figure 2.2 Bit Rate Adjustment Procedure

3. Command Specification

3.1 Control Commands

Control commands are listed below.

Control Command	1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte or More	ID Unchecked
Page read	FFH	Middle-order address	High-order address	**Data**	**Data**	**Data**	**Up to last data**	Acknowledgment disabled
Page program	41H	Middle-order address	High-order address	Data	Data	Data	Up to last data	Acknowledgment disabled
Unit program	49H	Low-order address	Middle-order address	High-order address	Size	Data	Up to last data	Acknowledgment disabled
Block erase	20H	Middle-order address	High-order address	D0H				Acknowledgment disabled
Erase all unlocked blocks	A7H	D0H						Acknowledgment disabled
Read status register	70H	**SRD**	**SRD1**					Acknowledgment enabled
Clear status register	50H							Acknowledgment disabled
All block blank check	26H	D0H						Acknowledgment disabled
Blank check	F7H	Start address (middle-order)	Start address (high-order)	End address (middle-order)	End address (high-order)	**Data**	**Up to last data**	Acknowledgment disabled
Verify check	F9H	Start address (middle-order)	Start address (high-order)	End address (middle-order)	End address (high-order)	**Determine code (low-order)**	**Determine code (high-order)**	Acknowledgment disabled
ID data check function	F5H	Low-order address	Middle-order address	High-order address	ID size	ID1	Up to ID7	Acknowledgment enabled
Version information output function	FBH	**Version**	**Version**	**Version**	**Version**	**Version**	**Up to last version**	Acknowledgment enabled
Bit rate 9600	B0H	**B0H**						Acknowledgment enabled
Bit rate 19200	B1H	**B1H**						Acknowledgment enabled
Bit rate 38400	B2H	**B2H**						Acknowledgment enabled
Bit rate 57600	B3H	**B3H**						Acknowledgment enabled
Bit rate 115200	B4H	**B4H**						Acknowledgment enabled
Bit rate setting	B5H	Data	**Data**					Acknowledgment enabled
Standard serial I/O mode 3 bit rate setting	B7H	Data	**B7H**					Acknowledgment enabled
Standard time data	00H							Acknowledgment enabled
Boot end command	01H	D0H	**01H**					Acknowledgment enabled

Notes: 1. Shaded areas show transmission from the MCU to the programmer. Non-shaded areas show transmission from the programmer to the MCU.
2. SRD: Status register data. SRD1: Status register data 1.
3. Blank products can acknowledge all commands.
4. Standard time data is transferred 16 times during initial communication.
5. The number of received data is not checked and the timeout error is not processed in the boot program. When transmitting a command, make sure there is no excess or shortage of data.

4. Commands

4.1 Page Read

4.1.1 Operation

This command is used to read the specified user ROM area in the flash memory in 256-byte units. The read area is specified by the high-order address (A16 to A23) and middle-order address (A8 to A15). The targets are 256 bytes from addresses xxxx00 to xxxxFFh.

4.1.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 256th Byte
	Command	**Address**		**Data**	**Up to last data**
Programmer to MCU	FFh	Middle-order address	High-order address		
MCU to Programmer				Data 0	Up to Data 255

Note: 1. Data 0 is the data which is stored in address xxxx00. Data 255 is the data which is stored in address xxxxFFh.

4.1.3 Procedure

(1) Transmit the page read command FFh to the first byte.
(2) Transmit the middle-order address to the second byte and the high-order address to the third byte.
(3) After the fourth byte, receive the data sequentially from the data which is stored in address xxxx00.

4.2 Page Program

4.2.1 Operation

This command is used to program the data to the specified user ROM area in the flash memory in 256-byte units. The program area is specified by the high-order address (A16 to A23) and middle-order address (A8 to A15). The targets are 256 bytes from xxxx00 to xxxxFFh.

This command programs blocks even if they are protected by the lock bits.

4.2.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 256th Byte
	Command	Address		Data	Up to last data
Programmer to MCU	41h	Middle-order address	High-order address	Data 0	Up to Data 255
MCU to Programmer					

Note: 1. Data 0 is the data whose destination address is xxxx00. Data 255 is the data whose destination data address is xxxxFFh.

4.2.3 Procedure

(1) Transmit the page program command 41h to the first byte.
(2) Transmit the middle-order address to the second byte and the high-order address to the third byte.
(3) After the fourth byte, transmit the program data sequentially from the data whose destination address is xxxx00.

When the program data is less than 256 bytes, transmit FFh for the shortage. When the program data is 257 bytes or more, any data after the 257th byte is considered a command. If an error occurs during programming, the SR4 bit becomes 1 (Error).

After executing this command, use the read status register command to confirm that program error does not occur.

4.3 Unit Program

4.3.1 Operation

This command is used to program the data to the specified user ROM area in the flash memory by the specified size. The starting address of the program area is specified by the high-order address (A16 to A23), middle-order address (A8 to A15), and low-order address (A0 to A7). The specified amount of program data is programmed from the start address.

This command programs blocks even if they are protected by the lock bits.

4.3.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	Up to N Byte
	Command	Address			No. of Pcs	Data	Up to last data
Programmer to MCU	49h	Low-order address	Middle-order address	High-order address	Size	Data (1)	Up to Data (N)
MCU to Programmer							

Notes: 1. Data (1) is data to be written to the start address and Data (N) is data to be written to (start address + N-1)
2. Make sure the “Size” does not change the value in the high-order address (e.g. When the start address is 1FFF0h, the maximum “Size” is 0Fh).

4.3.3 Procedure

(1) Transmit the unit program command 49h to the first byte.
(2) Transmit the low-order address to the second byte, the middle-order address to the third byte, and the high-order address to the fourth byte.
(3) Transmit the program size (01h to FFh) to the fifth byte.
(4) Transmit the specified amount of program data sequentially to the sixth byte or later.

If an error occurs during programming, the SR4 bit becomes 1 (Error).

After executing this command, use the read status register command to confirm that program error does not occur.

4.4 Block Erase

4.4.1 Operation

This command is used to erase the specified block in the flash memory. The block area is specified by the eight high-order bits (A16 to A23) and eight middle-order bits (A8 to A15) at any address of the block to be erased.

This command erases blocks even if they are protected by the lock bits.

4.4.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 256th Byte
	Command	Block Address			
Programmer to MCU	20h	Middle-order address	High-order address	D0h	
MCU to Programmer					

4.4.3 Procedure

(1) Transmit the block erase command 20h to the first byte.
(2) Transmit the middle-order address to the second byte and the high-order address to the third byte.
(3) Transmit the confirmation command D0h to the fourth byte.

After the confirmation command D0h is received, erasing starts on the specified block. Erasing sets the flash content to FFh. If an error occurs during erasing, the SR5 bit becomes 1 (Error).

After executing this command, use the read status register command to confirm that erase error does not occur.

4.5 Erase All Unlocked Blocks

4.5.1 Operation

This command is used to erase the entire user ROM area (data flash area and program ROM) in the flash memory.

This command erases blocks even if they are protected by the lock bits.

4.5.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 256th Byte
	Command				
Programmer to MCU	A7h	D0h			
MCU to Programmer					

4.5.3 Procedure

(1) Transmit the erase all unlocked block command A7h to the first byte.
(2) Transmit the confirmation command D0h to the second byte.

After receiving the confirmation command D0h, erasing starts on all blocks. Erasing sets the flash content to FFh. If an error occurs during erasing, the SR5 bit becomes 1 (Error).

After executing this command, use the read status register command to confirm that erase error does not occur.

4.6 Read Status Register

4.6.1 Operation

This command is used to confirm the operating status of the flash memory.

4.6.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 256th Byte
	Command	SRD			
Programmer to MCU	70h				
MCU to Programmer		SRD output	SRD1 output		

4.6.3 Procedure

(1) Transmit the read status register command 70h to the first byte.
(2) Receive SRD to the second byte.
(3) Receive SRD1 to the third byte.

4.6.4 SRD Register

Bit in SRD	Status Name	Definition	
		1	0
SR7 (bit 7)	Sequencer status	Ready	Busy
SR6 (bit 6)	Reserved		
SR5 (bit 5)	Erase status/Blank check status	Error	Completed normally
SR4 (bit 4)	Program status	Error	Completed normally
SR3 (bit 3)	Reserved		
SR2 (bit 2)	Reserved		
SR1 (bit 1)	Reserved		
SR0 (bit 0)	Reserved		

(1) Sequencer status

The sequencer status shows the operating status of the flash memory. This bit becomes 0 during auto-programming or auto-erase. It becomes 1 when auto-programming or auto-erase is completed.

(2) Erase status

The erase status shows the erase operating status. If an error occurs, this bit becomes 1. It becomes 0 when the clear status register command is executed.

(3) Blank check status

Blank check status shows whether the flash memory is blank or not. This bit becomes 1 when data other than blank data FFh is read. This bit becomes 0 after the clear status register command is executed.

(4) Program status

The program status shows the programming status. If an error occurs, this bit becomes 1. It becomes 0 when the clear status register command is executed.

(5) Reserved

The read value is undefined.

4.6.5 SRD1 Register

Bit in SRD1	Status Name	Definition	
		1	0
SR15 (bit 7)	Reserved		
SR14 (bit 6)	Reserved		
SR13 (bit 5)	Reserved		
SR12 (bit 4)	Reserved		
SR11 (bit 3)	ID check bits	00: Unchecked 01: Mismatched 10: Reserved 11: Matched	
SR10 (bit 2)			
SR9 (bit 1)	Reserved		
SR8 (bit 0)	Reserved		

(1) ID check bits

These bits show the ID check result.

(2) Reserved

The read value is undefined.

4.7 Clear Status Register

4.7.1 Operation

This command is used to initialize the status register. Initialize the status register when there is an error with the read status command.

4.7.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 256th Byte
	Command				
Programmer to MCU	50h				
MCU to Programmer					

4.7.3 Procedure

(1) Transmit the clear status register command 50h to the first byte.

After executing this command, use the read status register command to confirm that clear status error does not occur.

4.8 All Block Blank Check

4.8.1 Operation

This command is used to check whether the entire user ROM area (data flash and program ROM) in the flash memory is blank or not.

4.8.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 256th Byte
	Command				
Programmer to MCU	26h	D0h			
MCU to Programmer					

4.8.3 Procedure

(1) Transmit the all block blank check command 26h to the first byte.
(2) Transmit the confirmation command D0h to the second byte.

After receiving the confirmation command D0h, blank check for all blocks starts. Blank check determines whether the flash memory content is FFh or not. If an error occurs, SR5 becomes 1 (Error).

After executing this command, use the read status register command to confirmation that blank error does not occur.

4.9 Blank Check

4.9.1 Operation

This command is used to check whether the specified area in the flash memory is blank or not. The result is shown as a 4-byte code, which includes last address (3 bytes) of determine area and determine code (1 byte). This code is returned to the programmer.

4.9.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte	8th Byte	9th Byte
	Command	**Address**							**Determine code**
Programmer to MCU	F7h	Start address (middle-order)	Start address (high-order)	End address (middle-order)	End address (high-order)				
MCU to Programmer						Las address (low-order)	Last address (middle-order)	Last address (high-order)	Determine code

Note: 1. Specify the area by start address and end address. The low-order address of the start address is 00h, and the low-order address of the end address is FFh. Specify the area where the flash memory is allocated continuously for the start address and end address.

4.9.3 Procedure

(1) Transmit the blank check command F7h to the first byte.
(2) Transmit the middle-order address of the start address to the second byte and its high-order address to the third byte. Transmit the middle-order address of the end address to the fourth byte and its high-order address to the fifth byte.
(3) Receive the low-order address of the last address at the sixth byte, its middle-order address at the seventh byte, and its high-order address at the eighth byte.
(4) The determine code is received to the ninth byte.

4.9.4 Blank Check Determination Results

Determination results are determined by a reply code. The table below shows the reply code definitions (4 bytes).

Order of Reply	Item	Content	Definition	
			Normal (blank)	**Error**
1	Last address (low-order)	Last address of the area where blank check was performed	End address of the specified area	Address within the specified area
2	Last address (middle-order)			
3	Last address (high-order)			
4	Determine code	Memory content which was read by blank check	FFh	Other than FFh

4.10 Verify Check

4.10.1 Operation

This command is used to generate the verification determine code (2 bytes) from the data programmed in the specified area of the flash memory. The verification determine code (2 bytes) is obtained by adding the data in the specified area byte-by-byte, and calculating the one's complement on the lower 2 bytes of the added value.

4.10.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	7th Byte
	Command	**Address**				**Determine code**	
Programmer to MCU	F9h	Start address (middle-order)	Start address (high-order)	End address (middle-order)	End address (high-order)		
MCU to Programmer						Low-order	High-order

Note: 1. Specify the area by start address and end address. The low-order address of the start address is 00h, and the low-order address of the end address is FFh. Specify the area where the flash memory is allocated continuously for the start address and end address.

4.10.3 Procedure

(1) Transmit the verify check command F9h to the first byte.

(2) Transmit the middle-order address of the start address to the second byte and its high-order address to the third byte. Transmit the middle-order address of the end address to the fourth byte and its high-order address to the fifth byte.

(3) Receive the low-order determine code to the sixth byte and the high-order determine code to the seventh byte.

4.11 ID Data Check Function

4.11.1 Operation

This command is used to check if each ID stored in the flash memory matches an ID transmitted from the serial programmer. Some commands cannot be acknowledged if the ID check function finds a mismatch.

4.11.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	5th Byte	6th Byte	Up to 12th Byte
	Command	**Address**			**ID Size**	**ID**	**ID**
Programmer to MCU	F5h	DFh	FFh	00h	07h	ID1	UP to ID7
MCU to Programmer							

Notes: 1. Address means the address where ID1 is stored.

4.11.3 Procedure

(1) Transmit the ID data check function command F5h to the first byte.
(2)Transmit the low-order address where ID1 is stored to the second byte, the middle-order address to the third byte, and the high-order address to the fourth byte.
(3) Transmit the number of IDs (07h) to the fifth byte.
(4) Transmit the IDs sequentially to the sixth byte or later.

After transmission, the results are reflected in SR10 and SR11. If the transmitted address is not the ID address, or the ID size is not 7, it is determined as a mismatch even if the IDs are matched.

After executing this command, use the read status register command to confirm that the status is 11 (Matched).

4.12 Version Information Output Function

4.12.1 Operation

This command is used to confirm the boot program version.

4.12.2 Packet

	1st Byte	2nd Byte	3rd Byte	4th Byte	Up to 9th Byte
	Command				
Programmer to MCU	FBh				
MCU to Programmer		V	E	R	X

Note: 1. Version information consists of an eight-character ASCII code written as “VER. X. XX” (X’s are Roman numerals). Version information is received starting with “V”

4.12.3 Procedure

(1) Transmit the version information output function command FBh to the first byte.

(2) Receive the version information from the second byte to the ninth byte in ASCII characters.

4.13 Bit Rate 9600

4.13.1 Operation

This command is used to change the bit rate to typ. 9600 bps.

4.13.2 Packet

	1st Byte	2nd Byte
	Command	
Programmer to MCU	B0h	
MCU to Programmer		B0h

4.13.3 Procedure

(1) Transmit the bit rate 9600 command B0h to the first byte.
(2) Receive the confirmation command B0h to the second byte.
(3) The boot program sets the bit rate to typ. 9600 bps after transmitting the confirmation command.

4.14 Bit Rate 19200

4.14.1 Operation

This command is used to change the bit rate to typ. 19200 bps.

4.14.2 Packet

	1st Byte	2nd Byte
	Command	
Programmer to MCU	B1h	
MCU to Programmer		B1h

4.14.3 Procedure

(1) Transmit the bit rate 19200 command B1h to the first byte.
(2) Receive the confirmation command B1h to the second byte.
(3) The boot program sets the bit rate to typ. 19200 bps after transmitting the confirmation command.

4.15 Bit Rate 38400

4.15.1 Operation

This command is used to change the bit rate to typ. 38400 bps.

4.15.2 Packet

	1st Byte	2nd Byte
	Command	
Programmer to MCU	B2h	
MCU to Programmer		B2h

4.15.3 Procedure

(1) Transmit the bit rate 38400 command B2h to the first byte.
(2) Receive the confirmation command B2h to the second byte.
(3) The boot program sets the bit rate to typ. 38400 bps after transmitting the confirmation command.

4.16 Bit Rate 57600

4.16.1 Operation

This command is used to change the bit rate to typ. 57600 bps.

4.16.2 Packet

	1st Byte	2nd Byte
	Command	
Programmer to MCU	B3h	
MCU to Programmer		B3h

4.16.3 Procedure

(1) Transmit the bit rate 57600 command B3h to the first byte.
(2) Receive the confirmation command B3h to the second byte.
(3) The boot program sets the bit rate to typ. 57600 bps after transmitting the confirmation command.

4.17 Bit Rate 115200

4.17.1 Operation

This command is used to change the bit rate to typ. 115200 bps.

4.17.2 Packet

	1st Byte	2nd Byte
	Command	
Programmer to MCU	B4h	
MCU to Programmer		B4h

4.17.3 Procedure

(1) Transmit the bit rate 115200 command B4h to the first byte.
(2) Receive the confirmation command B4h to the second byte.
(3) The boot program sets the bit rate to typ. 115200 bps after transmitting the confirmation command.

4.18 Bit Rate Setting

4.18.1 Operation

This command is used to transmit the parameter data to set the bit rate in the boot program.

4.18.2 Packet

	1st Byte	2nd Byte	3rd Byte
	Command	Setting value	
Programmer to MCU	B5h	Data	
MCU to Programmer			Data

Note: 1. Set the data to 00h or 01h.

4.18.3 Procedure

(1) Transmit the bit rate setting command B5h to the first byte.
(2) Transmit the parameter data to the second byte.
(3) Receive the confirmation command (data transmitted to the second byte) to the third byte.
(4) The boot program sets the bit rate according to the received data after transmitting the confirmation command.

4.18.4 Communication Bit Rate

The bit rates set by the boot program are shown below.

Data	Bit Rate (bps)	
	Standard serial I/O mode 2	Standard serial I/O mode 3
01h	230400	250000
00h	460800	500000

4.19 Standard Serial I/O Mode 3 Bit Rate Setting

4.19.1 Operation

This command is used to transmit the parameter data to set the bit rate in standard serial I/O mode 3 in the boot program.

4.19.2 Packet

	1st Byte	2nd Byte	3rd Byte
	Command	**Setting value**	
Programmer to MCU	B7h	Data	
MCU to Programmer			B7h

Note: 1. Refer to the table below for settable data.

4.19.3 Procedure

(1) Transmit the standard serial I/O mode 3 bit rate setting command B7h to the first byte.
(2) Transmit the parameter data to the second byte.
(3) Receive the confirmation command B7h to the third byte.
(4) The boot program sets the bit rate according to the received data after transmitting the confirmation command.

4.19.4 Communication Bit Rate

The following table shows bit rates set in the boot program when the high-speed on-chip oscillator frequency is typ. 40 MHz.

Data	Bit Rate (bps)
09h to FFh	Reserved
08h	500000
07h	250000
00h to 06h	Reserved

4.20 Boot End Command

4.20.1 Operation

This command is sent after reprogramming is completed or before power-off.

4.20.2 Packet

	1st Byte	2nd Byte	3rd Byte
	Command		
Programmer to MCU	01h	D0h	
MCU to Programmer			01h

4.20.3 Procedure

(1) Transmit the boot end command 01h to the first byte.
(2) Transmit the confirmation command D0h to the second byte.
(3) Receive the confirmation command 01h to the third byte.
(4) Reprogramming is completed, or turn off the power supply.

Execute the boot end command when reprogramming is completed or before turning off the power supply. After executing the boot end command, the boot program enters an endless loop process and does not accept any commands.

5. Timing

5.1 Data Transmit Interval (Between Bytes)

When transmitting a standard time data, set the transmit interval to min. 20 ms. This transmit interval is required in the boot program for a receive process time.

5.2 Switching from Transmit to Receive in Standard Serial I/O Mode 3

Switch from transmit to receive within 500 μs.

5.2.1 Read Status Register Command in Programming and Erasing

By executing the read status register command after transmitting page program, unit program, block erase and erase all unlocked blocks commands, the boot program transmits the values in the SRD after programming or erasing. For details on electrical characteristics of program time and erase time, refer to the User's Manual: Hardware.

Website and Support

Renesas Electronics website
http://www.renesas.com

Inquiries
http://www.renesas.com/contact/

REVISION HISTORY	R8C/Mx Series, LAxA Group Standard Serial I/O Mode Serial Protocol Specification

Rev.	Date	Description	
		Page	Summary
1.00	Sep. 01, 2010	—	First edition issued
1.10	Feb. 15, 2013	11	4.3.2, Packet, note (2) added
		—	Fixed typos

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins
   Handle unused pins in accord 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 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.
5. Differences between Products
   Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.
   — The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of 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. 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 should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended 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; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor 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 damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may 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 should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700, Fax: +44-1628-651-804

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 2.2