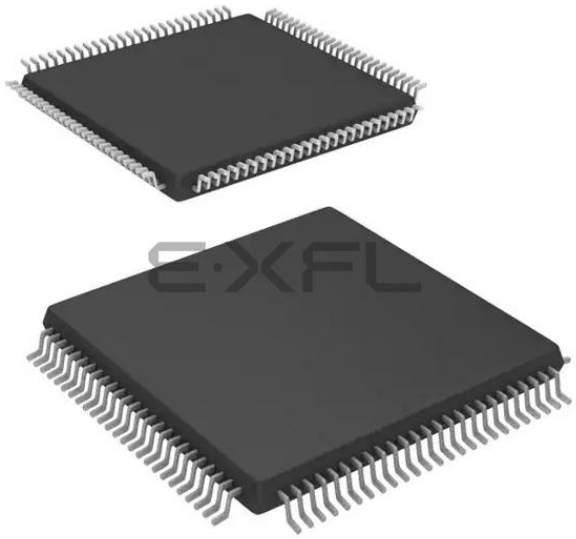
Renesas Electronics America Inc - D12312SVTE25V Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	H8S/2000
Core Size	16-Bit
Speed	25MHz
Connectivity	SCI, SmartCard
Peripherals	POR, PWM, WDT
Number of I/O	70
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/d12312svte25v

Table 1.1 Overview

Item	Specification
CPU	• General-register machine — Sixteen 16-bit general registers (also usable as sixteen 8-bit registers or eight 32-bit registers) • High-speed operation suitable for realtime control — Maximum clock rate: 25 MHz — High-speed arithmetic operations 8/16/32-bit register-register add/subtract: 40 ns (at 25-MHz operation) 16 × 16-bit register-register multiply: 800 ns (at 25-MHz operation) 32 ÷ 16-bit register-register divide: 800 ns (at 25-MHz operation) • Instruction set suitable for high-speed operation — Sixty-five basic instructions — 8/16/32-bit move/arithmetic and logic instructions — Unsigned/signed multiply and divide instructions — Powerful bit-manipulation instructions • CPU operating mode — Advanced mode: 16-Mbyte address space
Bus controller	• Address space divided into 8 areas, with bus specifications settable independently for each area • Chip select output possible for each area • Choice of 8-bit or 16-bit access space for each area • 2-state or 3-state access space can be designated for each area • Number of program wait states can be set for each area • Burst ROM directly connectable • External bus release function
Data transfer controller (DTC)	• Can be activated by internal interrupt or software • Multiple transfers or multiple types of transfer possible for one activation source • Transfer possible in repeat mode, block transfer mode, etc. • Request can be sent to CPU for interrupt that activated DTC
16-bit timer-pulse unit (TPU)	• 6-channel 16-bit timer • Pulse I/O processing capability for up to 16 pins • Automatic 2-phase encoder count capability

Table 5.4 Interrupt Sources, Vector Addresses, and Interrupt Priorities

Interrupt Source	Origin of Interrupt Source	Vector Number	Vector Address*	IPR	Priority	DTC Activation
Power-on reset		0	H'0000	—	High	—
Reserved		1	H'0004			
Reserved for system use		2	H'0008			
		3	H'000C			
		4	H'0010			
Trace		5	H'0014			
Reserved for system use		6	H'0018			
NMI	External pin	7	H'001C			
Trap instruction (4 sources)		8	H'0020			
		9	H'0024			
		10	H'0028			
		11	H'002C			
Reserved for system use		12	H'0030			
		13	H'0034			
		14	H'0038			
		15	H'003C			
IRQ0	External pin	16	H'0040	IPRA6 to IPRA4		○
IRQ1		17	H'0044	IPRA2 to IPRA0		○
IRQ2		18	H'0048	IPRB6 to IPRB4		○
IRQ3		19	H'004C			○
IRQ4		20	H'0050	IPRB2 to IPRB0		○
IRQ5		21	H'0054			○
IRQ6		22	H'0058	IPRC6 to IPRC4		○
IRQ7		23	H'005C		Low	○

5.4.2 Interrupt Control Mode 0

Enabling and disabling of IRQ interrupts and on-chip supporting module interrupts can be set by means of the I bit in the CPU's CCR. Interrupts are enabled when the I bit is cleared to 0, and disabled when set to 1.

Figure 5.5 shows a flowchart of the interrupt acceptance operation in this case.

[1] If an interrupt source occurs when the corresponding interrupt enable bit is set to 1, an interrupt request is sent to the interrupt controller.

[2] The I bit is then referenced. If the I bit is cleared to 0, the interrupt request is accepted. If the I bit is set to 1, only an NMI interrupt is accepted, and other interrupt requests are held pending.

[3] Interrupt requests are sent to the interrupt controller, the highest-ranked interrupt according to the priority system is accepted, and other interrupt requests are held pending.

[4] When an interrupt request is accepted, interrupt exception handling starts after execution of the current instruction has been completed.

[5] The PC and CCR are saved to the stack area by interrupt exception handling. The PC saved on the stack shows the address of the first instruction to be executed after returning from the interrupt handling routine.

[6] Next, the I bit in CCR is set to 1. This masks all interrupts except NMI.

[7] A vector address is generated for the accepted interrupt, and execution of the interrupt handling routine starts at the address indicated by the contents of that vector address.

5.6 DTC Activation by Interrupt

5.6.1 Overview

The DTC can be activated by an interrupt. In this case, the following options are available.

1. Interrupt request to CPU
2. Activation request to DTC
3. Selection of a number of the above

For details of interrupt requests that can be used with to activate the DTC, see section 7, Data Transfer Controller.

5.6.2 Block Diagram

Figure 5.9 shows a block diagram of the DTC and interrupt controller.

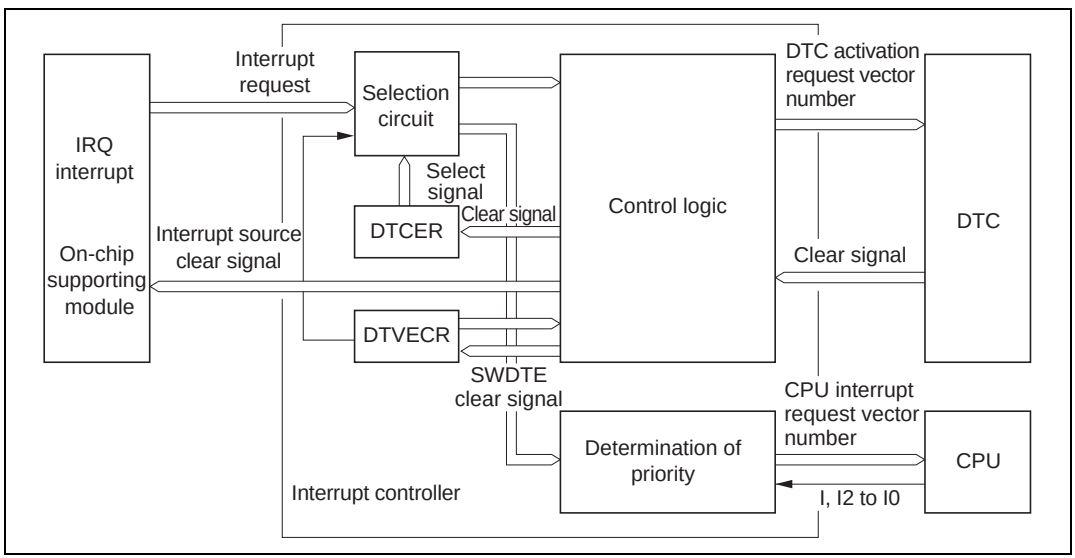


Figure 5.9 Interrupt Control for DTC

Port B Data Register (PBDR)

Bit	:	7	6	5	4	3	2	1	0
		PB7DR	PB6DR	PB5DR	PB4DR	PB3DR	PB2DR	PB1DR	PB0DR
Initial value	:	0	0	0	0	0	0	0	0
R/W	:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PBDR is an 8-bit readable/writable register that stores output data for the port B pins (PB7 to PB0). PBDR is initialized to H'00 by a reset, and in hardware standby mode. It retains its prior state in software standby mode.

Port B Register (PORTB)

Bit	:	7	6	5	4	3	2	1	0
		PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
Initial value	:	—*	—*	—*	—*	—*	—*	—*	—*
R/W	:	R	R	R	R	R	R	R	R

Note: * Determined by state of pins PB7 to PB0.

PORTB is an 8-bit read-only register that shows the pin states. It cannot be written to. Writing of output data for the port B pins (PB7 to PB0) must always be performed on PBDR.

If a port B read is performed while PBDDR bits are set to 1, the PBDR values are read. If a port B read is performed while PBDDR bits are cleared to 0, the pin states are read.

After a reset and in hardware standby mode, PORTB contents are determined by the pin states, as PBDDR and PBDR are initialized. PORTB retains its prior state in software standby mode.

Port Function Control Register 2 (PFCR2)

Bit	:	7	6	5	4	3	2	1	0
		—	—	CS167E	CS25E	ASOD	—	—	—
Initial value	:	0	0	1	1	0	0	0	0
R/W	:	R/W	R/W	R/W	R/W	R/W	R	R	R

PFCR2 is an 8-bit readable/writable register that performs I/O port control. PFCR2 is initialized to H'30 by a reset, and in hardware standby mode.

Bits 7 and 6—Reserved: Only 0 should be written to these bits.

Bit 5—CS167 Enable (CS167E): Enables or disables $\overline{CS1}$, $\overline{CS6}$, and $\overline{CS7}$ output. For details, see section 8.12, Port G.

Bit 4—CS25 Enable (CS25E): Enables or disables $\overline{CS2}$, $\overline{CS3}$, $\overline{CS4}$, and $\overline{CS5}$ output. Change the CS25E setting only when the DDR bits are cleared to 0. This bit is valid in modes 4 to 6.

Bit 4 CS25E	Description
0	$\overline{CS2}$, $\overline{CS3}$, $\overline{CS4}$, and $\overline{CS5}$ output disabled (can be used as I/O ports)
1	$\overline{CS2}$, $\overline{CS3}$, $\overline{CS4}$, and $\overline{CS5}$ output enabled (Initial value)

Bit 3—AS Output Disable (ASOD): Enables or disables $\overline{AS}$ output. This bit is valid in modes 4 to 6.

Bit 3 ASOD	Description
0	PF6 is used as $\overline{AS}$ output pin (Initial value)
1	PF6 is designated as I/O port, and does not function as $\overline{AS}$ output pin

Bits 2 to 0—Reserved: When read, these bits are always read as 0.

- Phase counting mode 2

  Figure 9.30 shows an example of phase counting mode 2 operation, and table 9.10 summarizes the TCNT up/down-count conditions.

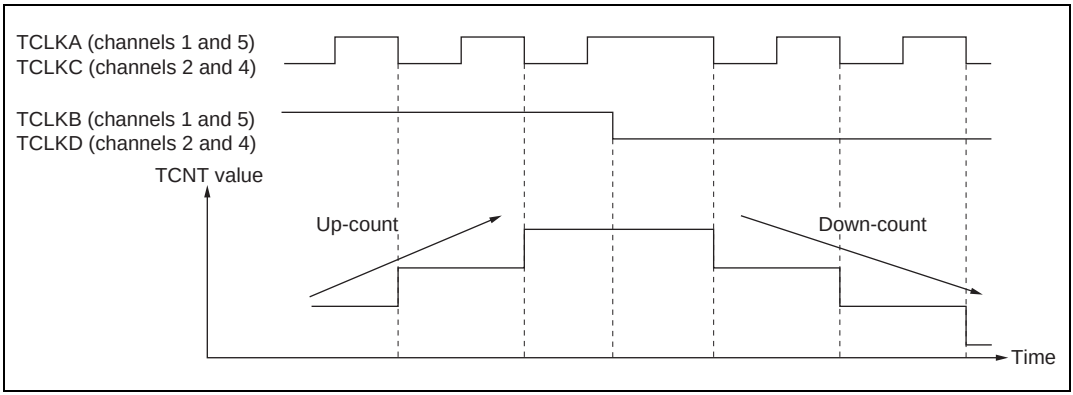


Figure 9.30 Example of Phase Counting Mode 2 Operation

Table 9.10 Up/Down-Count Conditions in Phase Counting Mode 2

TCLKA (Channels 1 and 5) TCLKC (Channels 2 and 4)	TCLKB (Channels 1 and 5) TCLKD (Channels 2 and 4)	Operation
High level	↑	Don't care
Low level	↓	
↑	Low level	
↓	High level	Up-count
High level	↓	Don't care
Low level	↑	
↑	High level	
↓	Low level	Down-count

Legend:

↑ : Rising edge

↓ : Falling edge

Table 9.13 lists the TPU interrupt sources.

Table 9.13 TPU Interrupts

Channel	Interrupt Source	Description	DTC Activation	Priority
0	TGI0A	TGR0A input capture/compare match	Possible	High
	TGI0B	TGR0B input capture/compare match	Possible	
	TGI0C	TGR0C input capture/compare match	Possible	
	TGI0D	TGR0D input capture/compare match	Possible	
	TCI0V	TCNT0 overflow	Not possible	
1	TGI1A	TGR1A input capture/compare match	Possible	
	TGI1B	TGR1B input capture/compare match	Possible	
	TCI1V	TCNT1 overflow	Not possible	
	TCI1U	TCNT1 underflow	Not possible	
2	TGI2A	TGR2A input capture/compare match	Possible	
	TGI2B	TGR2B input capture/compare match	Possible	
	TCI2V	TCNT2 overflow	Not possible	
	TCI2U	TCNT2 underflow	Not possible	
3	TGI3A	TGR3A input capture/compare match	Possible	
	TGI3B	TGR3B input capture/compare match	Possible	
	TGI3C	TGR3C input capture/compare match	Possible	
	TGI3D	TGR3D input capture/compare match	Possible	
	TCI3V	TCNT3 overflow	Not possible	
4	TGI4A	TGR4A input capture/compare match	Possible	
	TGI4B	TGR4B input capture/compare match	Possible	
	TCI4V	TCNT4 overflow	Not possible	
	TCI4U	TCNT4 underflow	Not possible	
5	TGI5A	TGR5A input capture/compare match	Possible	
	TGI5B	TGR5B input capture/compare match	Possible	
	TCI5V	TCNT5 overflow	Not possible	
	TCI5U	TCNT5 underflow	Not possible	Low

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

Contention between Buffer Register Write and Input Capture: If the input capture signal is generated in the T_2 state of a buffer write cycle, the buffer operation takes precedence and the write to the buffer register is not performed.

Figure 9.55 shows the timing in this case.

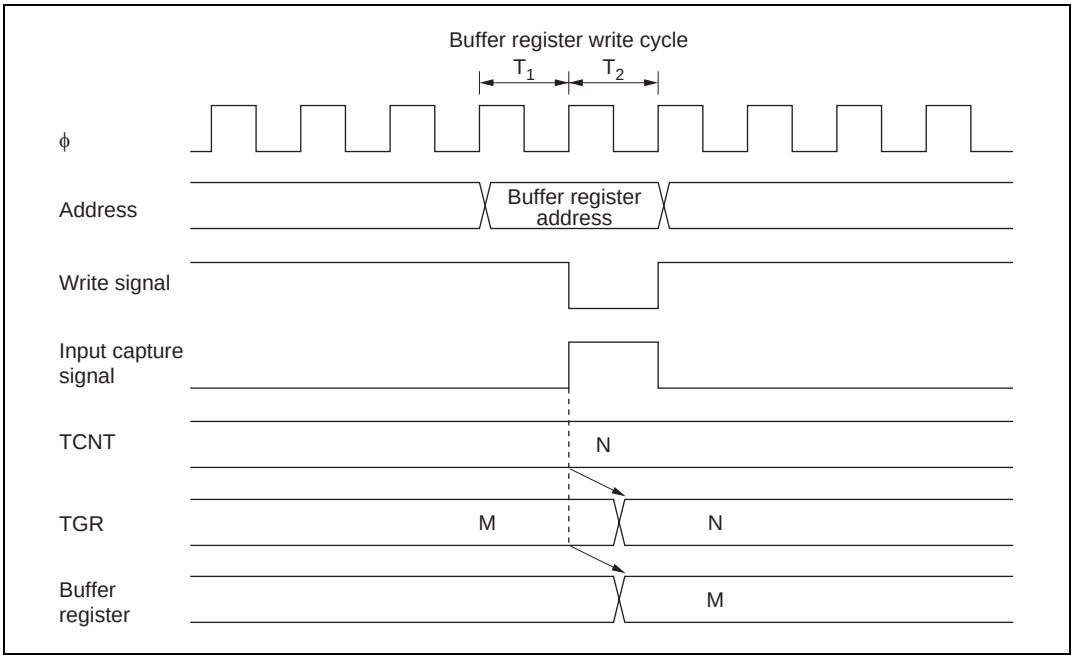


Figure 9.55 Contention between Buffer Register Write and Input Capture

Bit 2—Transmit End (TEND): Indicates that there is no valid data in TDR when the last bit of the transmit character is sent, and transmission has been ended.

The TEND flag is read-only and cannot be modified.

Bit 2 TEND	Description	
0	[Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the DTC is activated by a TXI interrupt and writes data to TDR	
1	[Setting conditions] • When the TE bit in SCR is 0 • When TDRE = 1 at transmission of the last bit of a 1-byte serial transmit character	(Initial value)

Bit 1—Multiprocessor Bit (MPB): When reception is performed using multiprocessor format in asynchronous mode, MPB stores the multiprocessor bit in the receive data.

MPB is a read-only bit, and cannot be modified.

Bit 1 MPB	Description	
0	[Clearing condition] When data with a 0 multiprocessor bit is received	(Initial value)*
1	[Setting condition] When data with a 1 multiprocessor bit is received	

Note: * Retains its previous state when the RE bit in SCR is cleared to 0 with multiprocessor format.

Bit 0—Multiprocessor Bit Transfer (MPBT): When transmission is performed using multiprocessor format in asynchronous mode, MPBT stores the multiprocessor bit to be added to the transmit data.

The MPBT bit setting is invalid when multiprocessor format is not used, when not transmitting, and in synchronous mode.

Bit 0 MPBT	Description	
0	Data with a 0 multiprocessor bit is transmitted	(Initial value)
1	Data with a 1 multiprocessor bit is transmitted	

Bit 3—Smart Card Data Transfer Direction (SDIR): Selects the serial/parallel conversion format.

This bit is valid when 8-bit data is used as the transmit/receive format.

Bit 3 SDIR	Description	
0	TDR contents are transmitted LSB-first	(Initial value)
	Receive data is stored in RDR LSB-first	
1	TDR contents are transmitted MSB-first	
	Receive data is stored in RDR MSB-first	

Bit 2—Smart Card Data Invert (SINV): Specifies inversion of the data logic level. The SINV bit does not affect the logic level of the parity bit(s): parity bit inversion requires inversion of the O/$\overline{E}$ bit in SMR.

Bit 2 SINV	Description	
0	TDR contents are transmitted without modification	(Initial value)
	Receive data is stored in RDR without modification	
1	TDR contents are inverted before being transmitted	
	Receive data is stored in RDR in inverted form	

Bit 1—Reserved: This bit cannot be modified and is always read as 1.

Bit 0—Smart Card Interface Mode Select (SMIF): When the smart card interface operates as a normal SCI, 0 should be written to this bit.

Bit 0 SMIF	Description	
0	Operates as normal SCI (smart card interface function disabled)	(Initial value)
1	Smart card interface function enabled	

12.4 SCI Interrupts

The SCI has four interrupt sources: the transmit-end interrupt (TEI) request, receive-error interrupt (ERI) request, receive-data-full interrupt (RXI) request, and transmit-data-empty interrupt (TXI) request. Table 12.12 shows the interrupt sources and their relative priorities. Individual interrupt sources can be enabled or disabled with the TIE, RIE, and TEIE bits in the SCR. Each kind of interrupt request is sent to the interrupt controller independently.

When the TDRE flag in SSR is set to 1, a TXI interrupt request is generated. When the TEND flag in SSR is set to 1, a TEI interrupt request is generated. A TXI interrupt can activate the DTC to perform data transfer. The TDRE flag is cleared to 0 automatically when data transfer is performed by the DTC. The DTC cannot be activated by a TEI interrupt request.

When the RDRF flag in SSR is set to 1, an RXI interrupt request is generated. When the ORER, PER, or FER flag in SSR is set to 1, an ERI interrupt request is generated. An RXI interrupt can activate the DTC to perform data transfer. The RDRF flag is cleared to 0 automatically when data transfer is performed by the DTC. The DTC cannot be activated by an ERI interrupt request.

Table 12.12 SCI Interrupt Sources

Channel	Interrupt Source	Description	DTC Activation	Priority*
0	ERI	Interrupt due to receive error (ORER, FER, or PER)	Not possible	High
	RXI	Interrupt due to receive data full state (RDRF)	Possible	
	TXI	Interrupt due to transmit data empty state (TDRE)	Possible	
	TEI	Interrupt due to transmission end (TEND)	Not possible	
1	ERI	Interrupt due to receive error (ORER, FER, or PER)	Not possible	
	RXI	Interrupt due to receive data full state (RDRF)	Possible	
	TXI	Interrupt due to transmit data empty state (TDRE)	Possible	
	TEI	Interrupt due to transmission end (TEND)	Not possible	Low

Note: * This table shows the initial state immediate after a reset. Relative priorities among channels can be changed by the interrupt controller.

With the above processing, interrupt handling or data transfer by the DTC is possible.

If reception ends and the RDRF flag is set to 1 while the RIE bit is set to 1 and interrupt requests are enabled, a receive data full interrupt (RXI) request will be generated. If an error occurs in reception and either the ORER flag or the PER flag is set to 1, a transmit/receive-error interrupt (ERI) request will be generated.

If the DTC is activated by an RXI request, the receive data in which the error occurred is skipped, and only the number of bytes of receive data set in the DTC are transferred.

For details, see Interrupt Operation and Data Transfer Operation by DTC below.

If a parity error occurs during reception and the PER is set to 1, the received data is still transferred to RDR, and therefore this data can be read.

Note: For details of operation in block transfer mode, see section 12.3.2, Operation in Asynchronous Mode.

Mode Switching Operation: When switching from receive mode to transmit mode, first confirm that the receive operation has been completed, then start from initialization, clearing RE bit to 0 and setting TE bit to 1. The RDRF flag or the PER and ORER flags can be used to check that the receive operation has been completed.

When switching from transmit mode to receive mode, first confirm that the transmit operation has been completed, then start from initialization, clearing TE bit to 0 and setting RE bit to 1. The TEND flag can be used to check that the transmit operation has been completed.

Fixing Clock Output: When the GSM bit in SMR is set to 1, the clock output can be fixed with bits CKE1 and CKE0 in SCR. At this time, the minimum clock pulse width can be made the specified width.

Figure 13.8 shows the timing for fixing the clock output. In this example, GSM is set to 1, CKE1 is cleared to 0, and the CKE0 bit is controlled.

- Programmer mode

  Flash memory can be programmed/erased in programmer mode, using a PROM programmer, as well as in on-board programming mode.

Note: * Flash memory emulation by RAM is not supported in the H8S/2314 F-ZTAT.

17.4.2 Overview

Block Diagram

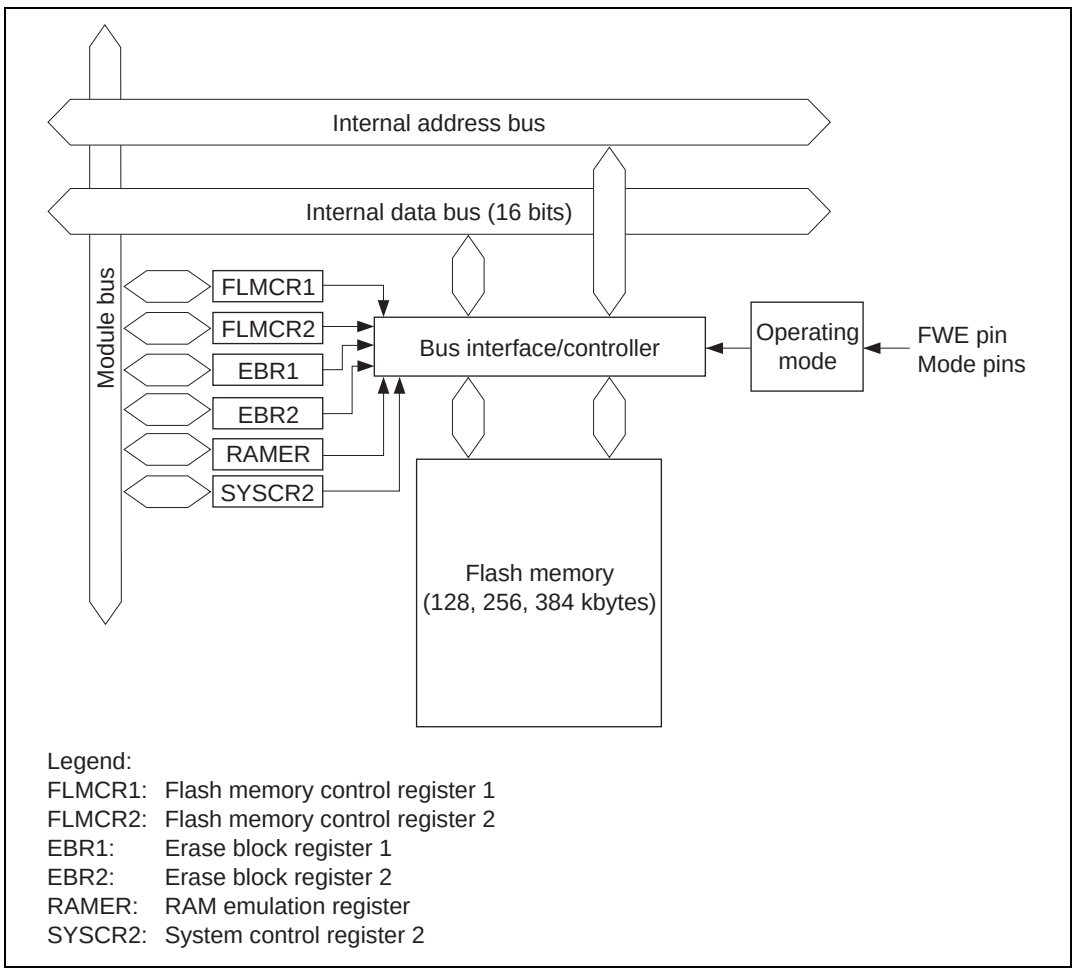


Figure 17.2 Block Diagram of Flash Memory

17.5 Register Descriptions

17.5.1 Flash Memory Control Register 1 (FLMCR1)

Bit	:	7	6	5	4	3	2	1	0
		FWE	SWE	ESU	PSU	EV	PV	E	P
Initial value	:	1/0	0	0	0	0	0	0	0
R/W	:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

FLMCR1 is an 8-bit register used for flash memory operating mode control. Program-verify mode or erase-verify mode is entered by setting SWE to 1 when FWE = 1, then setting the EV or PV bit. Program mode is entered by setting SWE to 1 when FWE = 1, then setting the PSU bit, and finally setting the P bit. Erase mode is entered by setting SWE to 1 when FWE = 1, then setting the ESU bit, and finally setting the E bit. FLMCR1 is initialized by a reset, and in hardware standby mode and software standby mode. Its initial value is H'80 when a high level is input to the FWE pin, and H'00 when a low level is input. When on-chip flash memory is disabled, a read will return H'00, and writes are invalid.

Writes to the SWE bit in FLMCR1 are enabled only when FWE = 1; writes to bits ESU, PSU, EV, and PV only when FWE = 1 and SWE = 1; writes to the E bit only when FWE = 1, SWE = 1, and ESU = 1; and writes to the P bit only when FWE = 1, SWE = 1, and PSU = 1.

Bit 7—Flash Write Enable Bit (FWE): Sets hardware protection against flash memory programming/erasing.

Bit 7 FWE	Description
0	When a low level is input to the FWE pin (hardware-protected state)
1	When a high level is input to the FWE pin

Bit 6—Software Write Enable Bit (SWE): Enables or disables flash memory programming and erasing. This bit should be set when setting FLMCR1 bits 5 to 0, EBR1 bits 7 to 0, and EBR2 bits 3 to 0*.

When SWE = 1, the flash memory can only be read in program-verify or erase-verify mode.

Note: * EBR2 bits 5 to 0 should be set in the H8S/2315 F-ZTAT and H8S/2314 F-ZTAT. Bits 1 and 0 should be set in the H8S/2317 F-ZTAT.

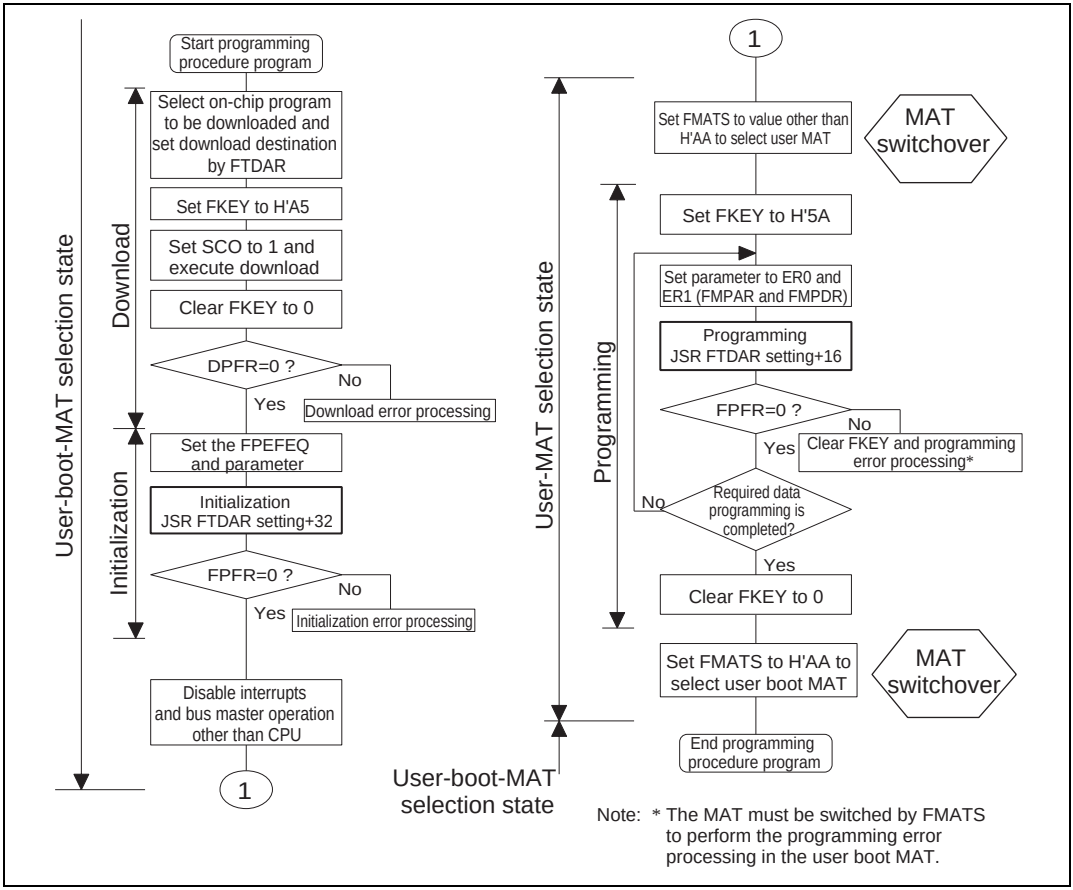


Figure 17.73 Procedure for Programming User MAT in User Boot Mode

The difference between the programming procedures in user program mode and user boot mode is whether the MAT is switched or not as shown in figure 17.73.

In user boot mode, the user boot MAT can be seen in the flash memory space with the user MAT hidden in the background. The user MAT and user boot MAT are switched only while the user MAT is being programmed. Because the user boot MAT is hidden while the user MAT is being programmed, the procedure program must be located in an area other than flash memory. After programming finishes, switch the MATs again to return to the first state.

MAT switchover is enabled by writing a specific value to FMATS. However note that while the MATs are being switched, the LSI is in an unstable state, e.g. access to a MAT is not allowed until MAT switching is completely finished, and if an interrupt occurs, from which MAT the interrupt

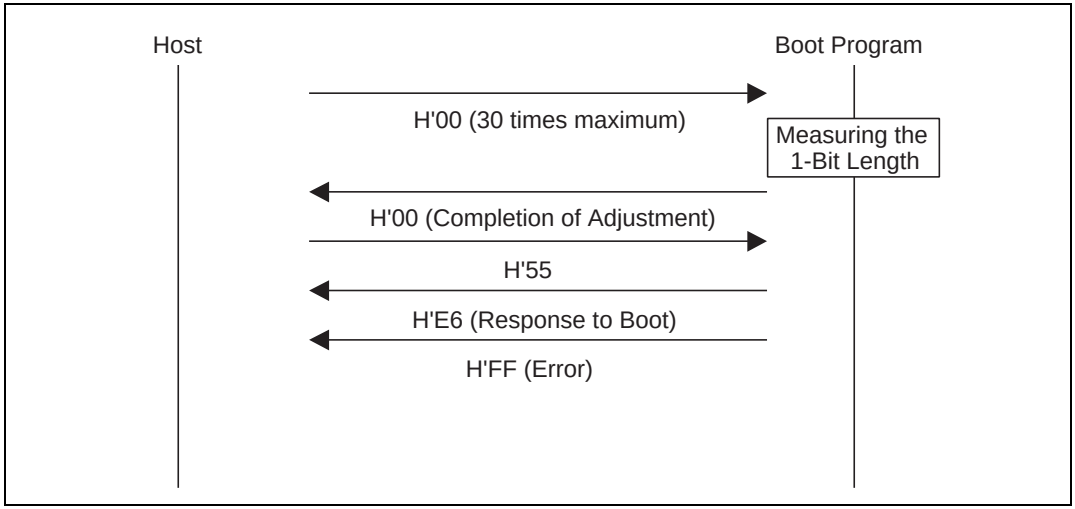


Figure 17.83 Bit-Rate-Adjustment Sequence

Communications Protocol

After adjustment of the bit rate, the protocol for communications between the host and the boot program is as shown below.

(1) One-byte commands and one-byte responses

These commands and responses are comprised of a single byte. These are consists of the inquiries and the ACK for successful completion.

(2) n-byte commands or n-byte responses

These commands and responses are comprised of n bytes of data. These are selections and responses to inquiries.

The amount of programming data is not included under this heading because it is determined in another command.

(3) Error response

The error response is a response to inquiries. It consists of an error response and an error code and comes two bytes.

(4) Programming of n bytes

The size is not specified in commands. The size of n is indicated in response to the programming unit inquiry.

(5) Memory read response

This response consists of four bytes of data.

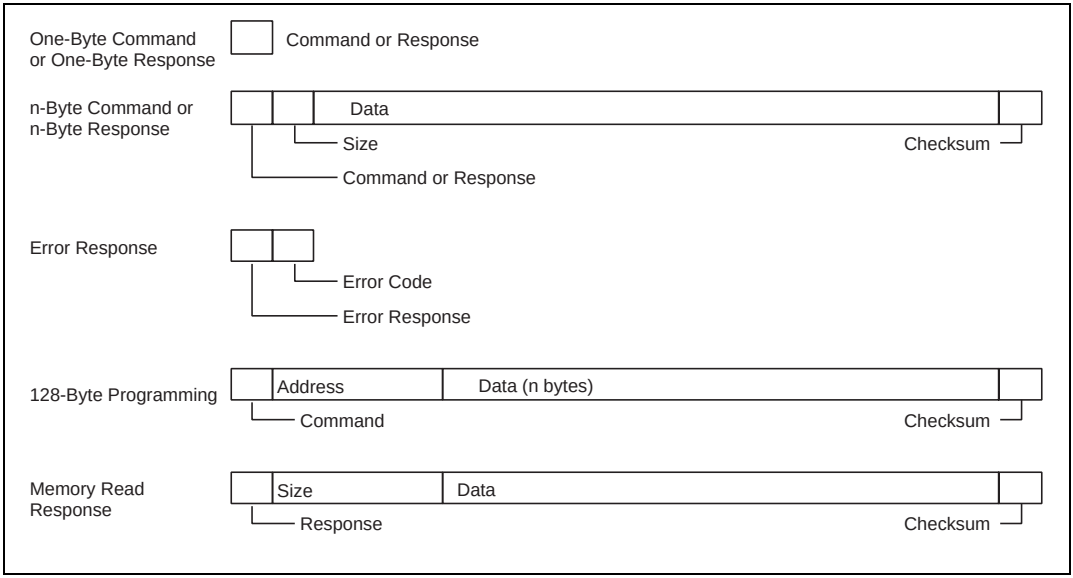


Figure 17.84 Communication Protocol Format

- Command (1 byte): Commands including inquiries, selection, programming, erasing, and checking
- Response (1 byte): Response to an inquiry
- Size (1 byte): The amount of data for transmission excluding the command, amount of data, and checksum
- Checksum (1 byte): The checksum is calculated so that the total of all values from the command byte to the SUM byte becomes H'00
- Data (n bytes): Detailed data of a command or response
- Error Response (1 byte): Error response to a command
- Error Code (1 byte): Type of the error
- Address (4 bytes): Address for programming
- Data (n bytes): Data to be programmed (the size is indicated in the response to the programming unit inquiry.)
- Size (4 bytes): Four-byte response to a memory read

Table 20.22 Permissible Output Currents

Condition B: V_{CC} = 3.0 V to 3.6 V, AV_{CC} = 3.0 to 3.6 V, V_{ref} = 3.0 V to AV_{CC},
V_{SS} = AV_{SS} = 0 V, T_a = –20 to +75°C (regular specifications),
T_a = –40 to +85°C (wide-range specifications)

Item		Symbol	Min	Typ	Max	Unit
Permissible output low current (per pin)	All output pins	I_{OL}	—	—	2.0	mA
Permissible output low current (total)	Total of all output pins	$\sum I_{OL}$	—	—	80	mA
Permissible output high current (per pin)	All output pins	$-I_{OH}$	—	—	2.0	mA
Permissible output high current (total)	Total of all output pins	$\sum -I_{OH}$	—	—	40	mA

Note: To protect chip reliability, do not exceed the output current values in table 20.22.