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	33MHz
Connectivity	I²C, IrDA, SCI, SmartCard
Peripherals	DMA, POR, PWM, WDT
Number of I/O	84
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 10x10b; D/A 2x8b
Oscillator Type	Internal
Operating Temperature	-20°C ~ 75°C (TA)
Mounting Type	Surface Mount
Package / Case	128-BFQFP
Supplier Device Package	128-QFP (14x20)
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/d12363vf33v

Section 22 Clock Pulse Generator

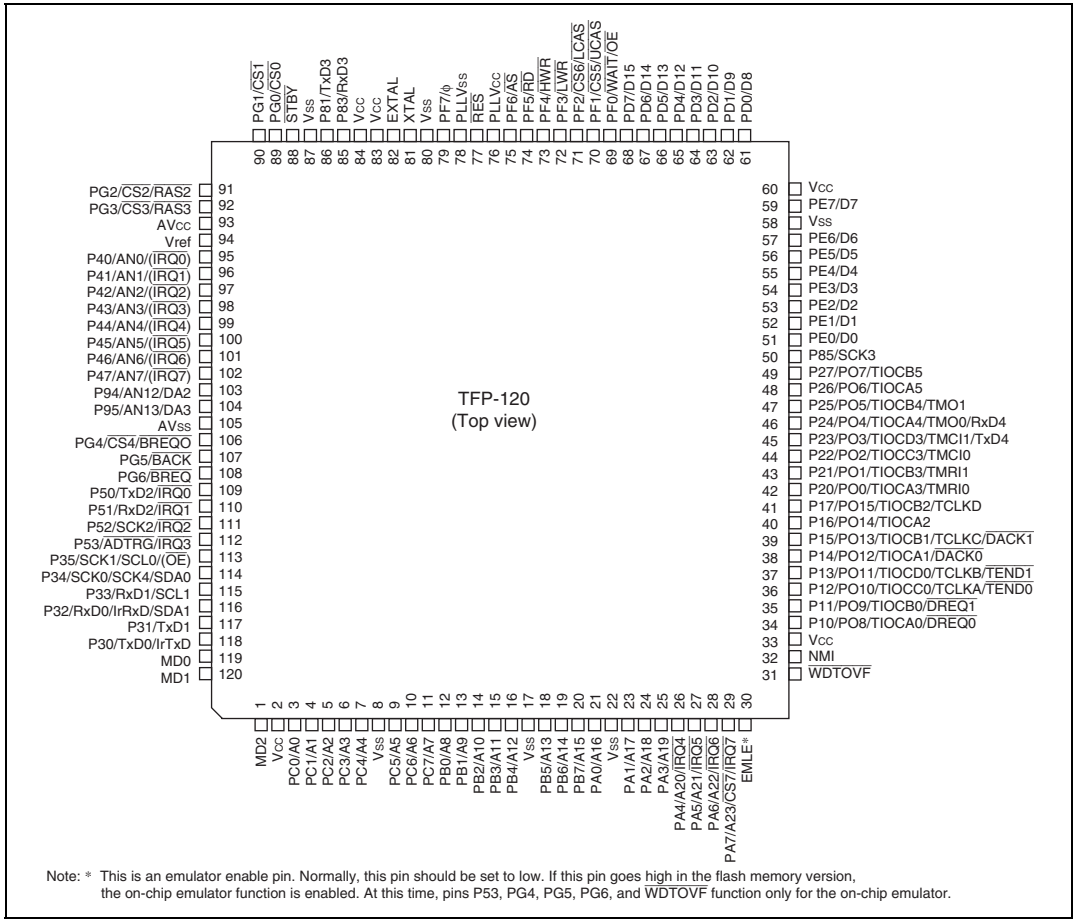
Section 23 Power-Down Modes

Section 25 Electrical Characteristics

1.3 Pin Description

1.3.1 Pin Arrangement

Figures 1.3 to 1.5 show the pin arrangements of this LSI.



Note: * This is an emulator enable pin. Normally, this pin should be set to low. If this pin goes high in the flash memory version, the on-chip emulator function is enabled. At this time, pins P53, PG4, PG5, PG6, and WDTOVF function only for the on-chip emulator.

Figure 1.3 Pin Arrangement of H8S/2367F, H8S/2365, and H8S/2363

2.1.2 Differences from H8/300 CPU

In comparison to the H8/300 CPU, the H8S/2000 CPU has the following enhancements.

- More general registers and control registers
  - Eight 16-bit extended registers, and one 8-bit and two 32-bit control registers, have been added.
- Expanded address space
  - Normal mode supports the same 64-kbyte address space as the H8/300 CPU.
  - Advanced mode supports a maximum 16-Mbyte address space.
- Enhanced addressing
  - The addressing modes have been enhanced to make effective use of the 16-Mbyte address space.
- Enhanced instructions
  - Addressing modes of bit-manipulation instructions have been enhanced.
  - Signed multiply and divide instructions have been added.
  - Two-bit shift and two-bit rotate instructions have been added.
  - Instructions for saving and restoring multiple registers have been added.
  - A test and set instruction has been added.
- Higher speed
  - Basic instructions are executed twice as fast.

2.1.3 Differences from H8/300H CPU

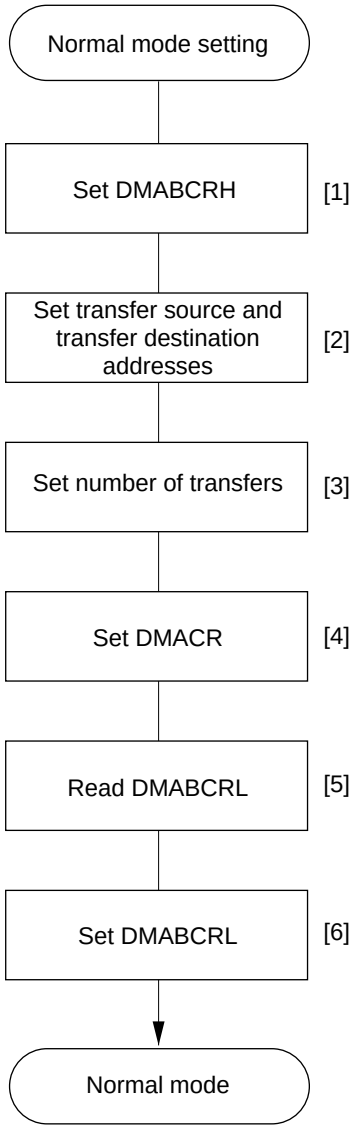
In comparison to the H8/300H CPU, the H8S/2000 CPU has the following enhancements.

- Additional control register
  - One 8-bit control register has been added.
- Enhanced instructions
  - Addressing modes of bit-manipulation instructions have been enhanced.
  - Two-bit shift and two-bit rotate instructions have been added.
  - Instructions for saving and restoring multiple registers have been added.
  - A test and set instruction has been added.
- Higher speed
  - Basic instructions are executed twice as fast.

5.3.6 IRQ Pin Select Register (ITSR)

ITSR selects input pins $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$.

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	–	All 0	R/W	Reserved The write value should always be 0.
7	ITS7	0	R/W	Selects $\overline{\text{IRQ7}}$ input pin. 0: PA7 1: P47
6	ITS6	0	R/W	Selects $\overline{\text{IRQ6}}$ input pin. 0: PA6 1: P46
5	ITS5	0	R/W	Selects $\overline{\text{IRQ5}}$ input pin. 0: PA5 1: P45
4	ITS4	0	R/W	Selects $\overline{\text{IRQ4}}$ input pin. 0: PA4 1: P44
3	ITS3	0	R/W	Selects $\overline{\text{IRQ3}}$ input pin. 0: P53 1: P43
2	ITS2	0	R/W	Selects $\overline{\text{IRQ2}}$ input pin. 0: P52 1: P42
1	ITS1	0	R/W	Selects $\overline{\text{IRQ1}}$ input pin. 0: P51 1: P41
0	ITS0	0	R/W	Selects $\overline{\text{IRQ0}}$ input pin. 0: P50 1: P40



[1] Set each bit in DMABCRH.
- Set the FAE bit to 1 to select full address mode.
- Specify enabling or disabling of internal interrupt clearing with the DTA bit.

[2] Set the transfer source address in MARA, and the transfer destination address in MARB.

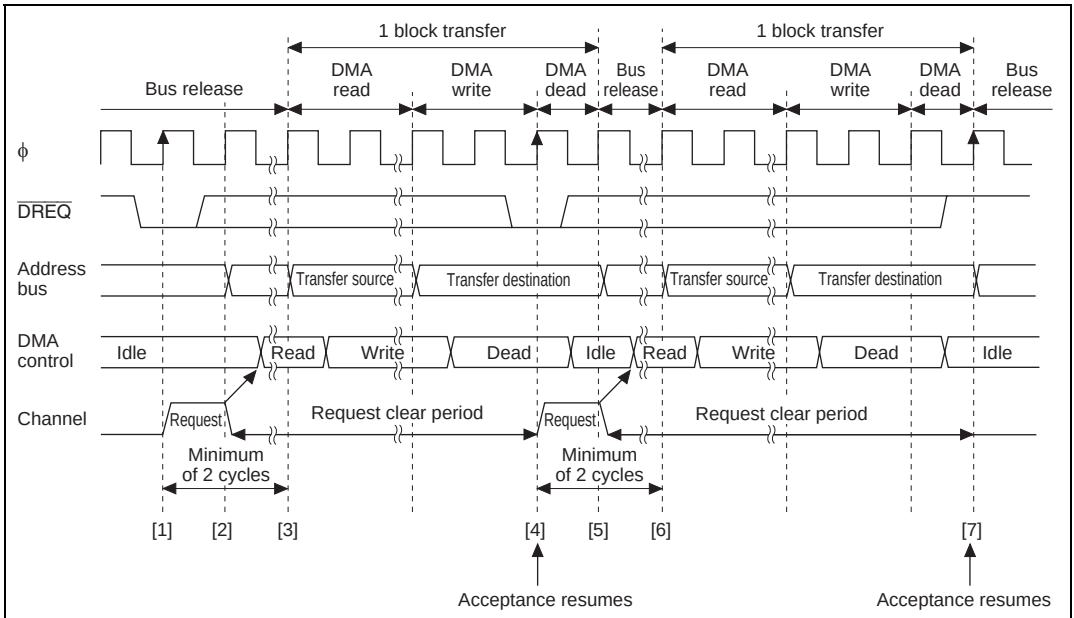
[3] Set the number of transfers in ETCRA.

[4] Set each bit in DMACRA and DMACRB.
- Set the transfer data size with the DTSZ bit.
- Specify whether MARA is to be incremented, decremented, or fixed, with the SAID and SAIDE bits.
- Clear the BLKE bit to 0 to select normal mode.
- Specify whether MARB is to be incremented, decremented, or fixed, with the DAID and DAIDE bits.
- Select the activation source with bits DTF3 to DTF0.

[5] Read DTE = 0 and DTME = 0 in DMABCRL.

[6] Set each bit in DMABCRL.
- Specify enabling or disabling of transfer end interrupts with the DTIE bit.
- Set both the DTME bit and the DTE bit to 1 to enable transfer.

Figure 7.12 Example of Normal Mode Setting Procedure



[1] Acceptance after transfer enabling; the $\overline{\text{DREQ}}$ pin low level is sampled on the rising edge of φ, and the request is held.
[2] [5] The request is cleared at the next bus break, and activation is started in the DMAC.
[3] [6] The DMA cycle is started.
[4] [7] Acceptance is resumed after the dead cycle is completed.
(As in [1], the $\overline{\text{DREQ}}$ pin low level is sampled on the rising edge of φ, and the request is held.)

Note: In write data buffer mode, bus breaks from [2] to [7] may be hidden, and not visible.

Figure 7.25 Example of $\overline{\text{DREQ}}$ Pin Low Level Activated Block Transfer Mode Transfer

$\overline{\text{DREQ}}$ pin sampling is performed every cycle, with the rising edge of the next φ cycle after the end of the DMABCR write cycle for setting the transfer enabled state as the starting point.

When the $\overline{\text{DREQ}}$ pin low level is sampled while acceptance by means of the $\overline{\text{DREQ}}$ pin is possible, the request is held in the DMAC. Then, when activation is initiated in the DMAC, the request is cleared. After the end of the dead cycle, acceptance resumes, $\overline{\text{DREQ}}$ pin low level sampling is performed again, and this operation is repeated until the transfer ends.

7.5.10 DMA Transfer (Single Address Mode) Bus Cycles

Single Address Mode (Read): Figure 7.26 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and byte-size single address mode transfer (read) is performed from external 8-bit, 2-state access space to an external device.

The timing for setting the TEND flag depends on the value of the GM bit in SMR. The TEND flag generation timing is shown in figure 14.27.

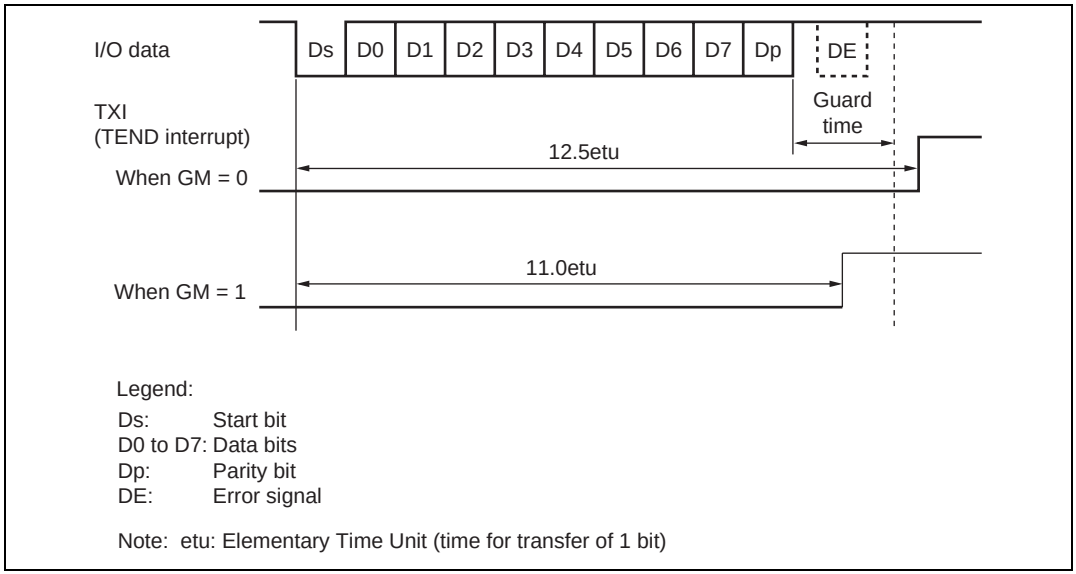


Figure 14.27 TEND Flag Generation Timing in Transmission Operation

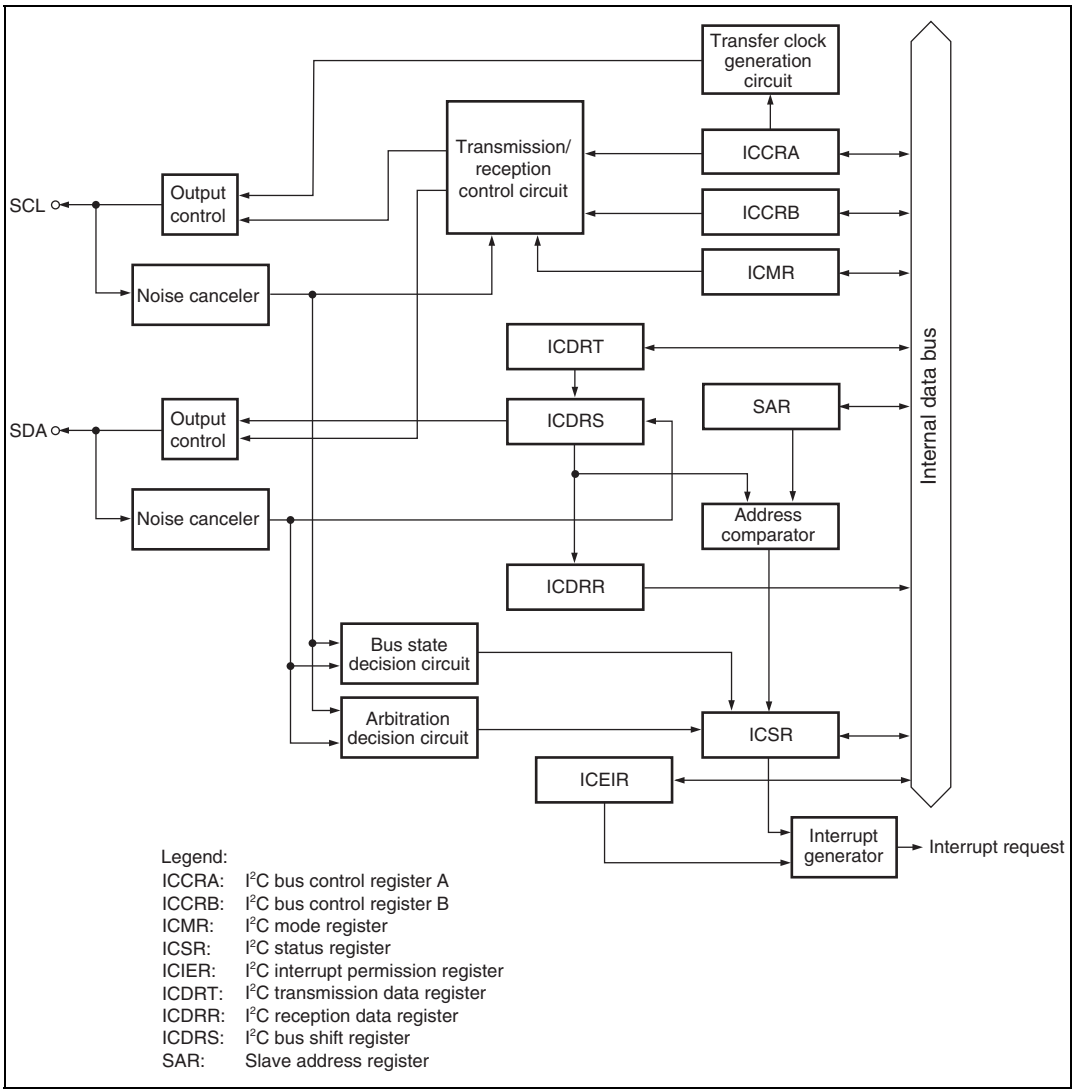


Figure 15.1 Block Diagram of I²C Bus Interface2

3. Clear RDRF after reading ICDRR every time RDRF is set. If 8th receive clock pulse falls while RDRF is 1, SCL is fixed low until ICDRR is read. The change of the acknowledge before reading ICDRR, to be returned to the master device, is reflected to the next transmit frame.
4. The last byte data is read by reading ICDRR.

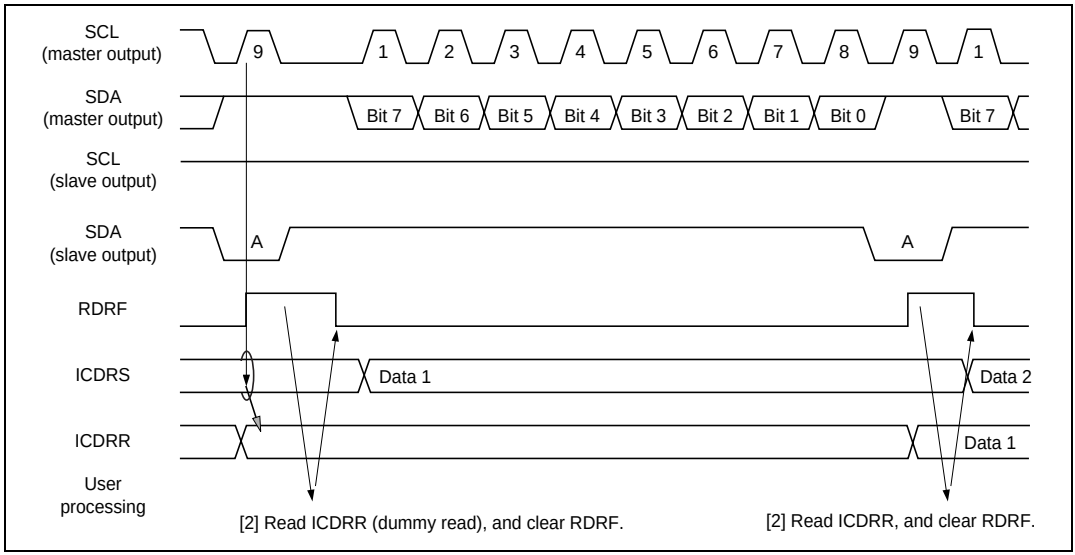
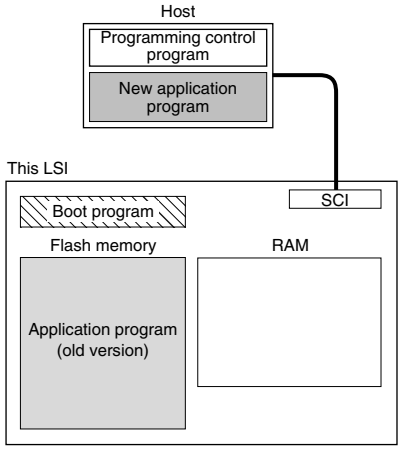
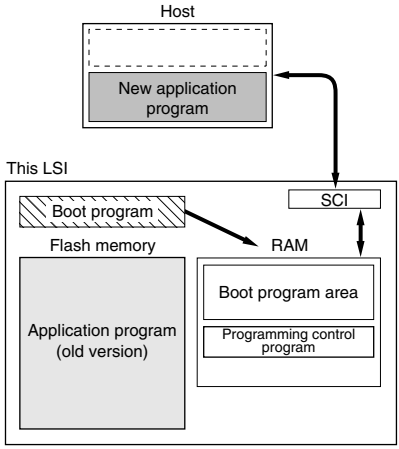


Figure 15.11 Slave Receive Mode Operation Timing 1

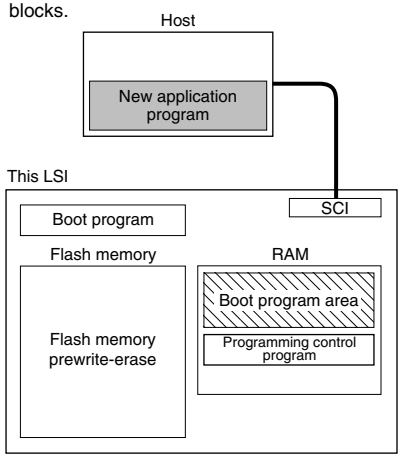
1. Initial state
   The old program version or data remains written in the flash memory. The user should prepare the programming control program and new application program beforehand in the host.



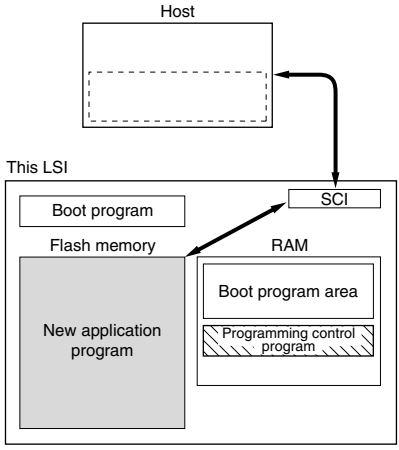
2. Programming control program transfer
   When boot mode is entered, the boot program in the chip (originally incorporated in the chip) is started and the programming control program in the host is transferred to RAM via SCI communication. The boot program required for flash memory erasing is automatically transferred to the RAM boot program area.



3. Flash memory initialization
   The erase program in the boot program area (in RAM) is executed, and the flash memory is initialized (to H'FF). In boot mode, entire flash memory erasure is performed, without regard to blocks.



4. Writing new application program
   The programming control program transferred from the host to RAM is executed, and the new application program in the host is written into the flash memory.



Program execution state

Figure 19.3 Boot Mode

19.11 Usage Notes

Precautions concerning the use of on-board programming mode, and programmer mode are summarized below.

1. Use the specified voltages and timing for programming and erasing.

   Applied voltages in excess of the rating can permanently damage the device. Use a PROM programmer that supports the Renesas microcomputer device type with 512-kbyte on-chip flash memory (FZTAT512V3A).

   Do not select the HN27C4096 setting for the PROM programmer, and only use the specified socket adapter.
2. Reset the flash memory before turning on/off the power.

   When applying or disconnecting Vcc power, fix the $\overline{\text{RES}}$ pin low and place the flash memory in the hardware protection state. The power-on and power-off timing requirements should also be satisfied in the event of a power failure and subsequent recovery.
3. Use the recommended algorithm when programming and erasing flash memory.

   The recommended algorithm enables programming and erasing to be carried out without subjecting the device to voltage stress or sacrificing program data reliability. When setting the P or E bit in FLMCR1, the watchdog timer should be set beforehand as a precaution against program runaway, etc.
4. Do not set or clear the SWE bit during execution of a program in flash memory.

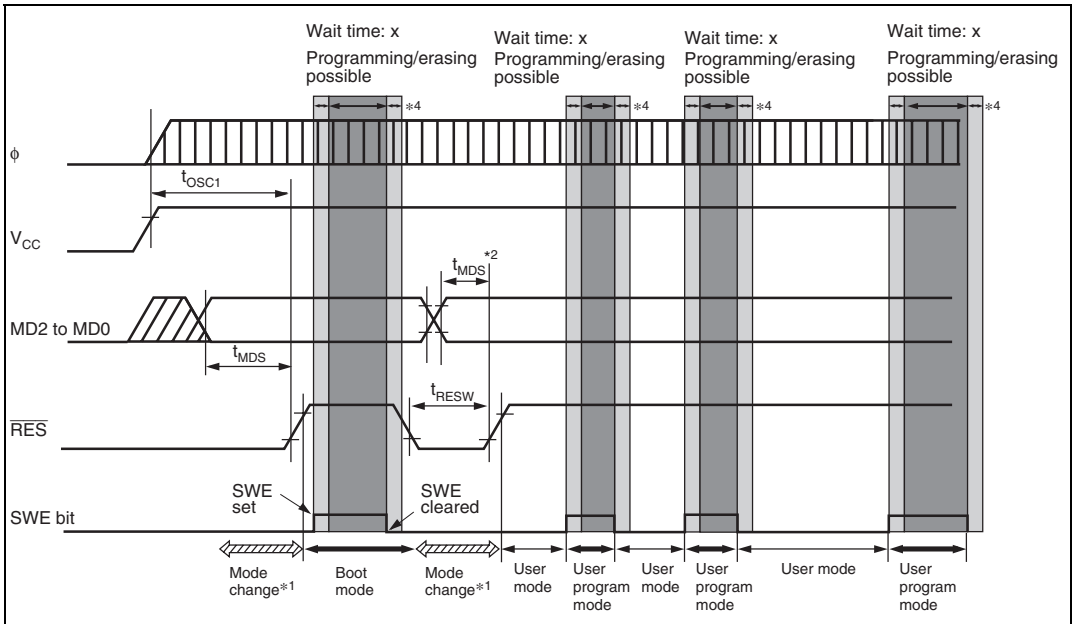
   Wait for at least 100 μs after clearing the SWE bit before executing a program or reading data in flash memory.

   When the SWE bit is set, data in flash memory can be rewritten. When the SWE bit is set to 1, data in flash memory can be read only in program-verify/erase-verify mode. Access flash memory only for verify operations (verification during programming/erasing). Also, do not clear the SWE bit during programming, erasing, or verifying.
5. Do not use interrupts while flash memory is being programmed or erased.

   All interrupt requests, including NMI, should be disabled during programming/erasing the flash memory to give priority to program/erase operations.
6. Do not perform additional programming. Erase the memory before reprogramming.

   In on-board programming, perform only one programming operation on a 128-byte programming unit block. In programmer mode, too, perform only one programming operation on a 128-byte programming unit block. Programming should be carried out with the entire programming unit block erased.
7. Before programming, check that the chip is correctly mounted in the PROM programmer.

   Overcurrent damage to the device can result if the index marks on the PROM programmer socket, socket adapter, and chip are not correctly aligned.



Period during which flash memory access is prohibited
(x: Wait time after setting SWE bit)∗3

Period during which flash memory can be programmed
(Execution of program in flash memory prohibited, and data reads other than verify operations prohibited)

Notes: 1. When entering boot mode or making a transition from boot mode to another mode, mode switching must be carried out by means of $\overline{RES}$ input. The state of ports with multiplexed address functions and bus control output pins ($\overline{AS}$, $\overline{RD}$, $\overline{HWR}$, $\overline{LWR}$) will change during this switchover interval (the interval during which the $\overline{RES}$ pin input is low), and therefore these pins should not be used as output signals during this time.
2. When making a transition from boot mode to another mode, a mode programming setup time t_{MDS} (min) of 200 ns is necessary with respect to $\overline{RES}$ clearance timing.
3. See section 25.2.5, Flash Memory Characteristics.
4. Wait time: 100 µs

Figure 19.10 Mode Transition Timing
(Example: Boot Mode → User Mode ↔ User Program Mode)

- Flash Program Code Select Register (FPCS)

  FPCS selects the on-chip programming program to be downloaded.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PPVS	0	R/W	Program Pulse Verify Selects the programming program. 0: On-chip programming program is not selected [Clear condition] When transfer is completed 1: On-chip programming program is selected

- Flash Erase Code Select Register (FECS)

  FECS selects download of the on-chip erasing program.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	EPVB	0	R/W	Erase Pulse Verify Block Selects the erasing program. 0: On-chip erasing program is not selected [Clear condition] When transfer is completed 1: On-chip erasing program is selected

(4) Inquiry and Selection States

The boot program returns information from the flash memory in response to the host's inquiry commands and sets the device code, clock mode, and bit rate in response to the host's selection command.

Inquiry and selection commands are listed below.

Table 20.11 Inquiry and Selection Commands

Command	Command Name	Description
H'20	Supported Device Inquiry	Inquiry regarding device codes
H'10	Device Selection	Selection of device code
H'21	Clock Mode Inquiry	Inquiry regarding numbers of clock modes and values of each mode
H'11	Clock Mode Selection	Indication of the selected clock mode
H'22	Multiplication Ratio Inquiry	Inquiry regarding the number of frequency-multiplied clock types, the number of multiplication ratios, and the values of each multiple
H'23	Operating Clock Frequency Inquiry	Inquiry regarding the maximum and minimum values of the main clock and peripheral clocks
H'24	User Boot MAT Information Inquiry	Inquiry regarding the number of user boot MATs and the start and last addresses of each MAT
H'25	User MAT Information Inquiry	Inquiry regarding the a number of user MATs and the start and last addresses of each MAT
H'26	Block for Erasing Information Inquiry	Inquiry regarding the number of blocks and the start and last addresses of each block
H'27	Programming Unit Inquiry	Inquiry regarding the unit of programming data
H'3F	New Bit Rate Selection	Selection of new bit rate
H'40	Transition to Programming/Erasing State	Erasing of user MAT and user boot MAT, and entry to programming/erasing state
H'4F	Boot Program Status Inquiry	Inquiry into the operated status of the boot program

The selection commands, which are device selection (H'10), clock mode selection (H'11), and new bit rate selection (H'3F), should be sent from the host in that order. These commands will certainly be needed. When two or more selection commands are sent at once, the last command will be valid.

Item	Symbol	Min	Max	Unit	Test Conditions
Address read data access time 1	t_{AA1}	—	$1.0 \times t_{cyc} - 20$	ns	Figures 25.6 to 25.19
Address read data access time 2	t_{AA2}	—	$1.5 \times t_{cyc} - 20$	ns	
Address read data access time 3	t_{AA3}	—	$2.0 \times t_{cyc} - 20$	ns	
Address read data access time 4	t_{AA4}	—	$2.5 \times t_{cyc} - 20$	ns	
Address read data access time 5	t_{AA5}	—	$3.0 \times t_{cyc} - 20$	ns	

Instruction	1	2	3	4	5	6	7	8	9
BGE d:16	R:W 2nd	1 state of internal operation	R:W EA						
BLT d:16	R:W 2nd	1 state of internal operation	R:W EA						
BGT d:16	R:W 2nd	1 state of internal operation	R:W EA						
BLE d:16	R:W 2nd	1 state of internal operation	R:W EA						
BCLR #xx:3,Rd	R:W NEXT								
BCLR #xx:3,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA					
BCLR #xx:3,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA					
BCLR #xx:3,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA				
BCLR #xx:3,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA			
BCLR Rn,Rd	R:W NEXT								
BCLR Rn,@ERd	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA					
BCLR Rn,@aa:8	R:W 2nd	R:B:M EA	R:W:M NEXT	W:B EA					
BCLR Rn,@aa:16	R:W 2nd	R:W 3rd	R:B:M EA	R:W:M NEXT	W:B EA				
BCLR Rn,@aa:32	R:W 2nd	R:W 3rd	R:W 4th	R:B:M EA	R:W:M NEXT	W:B EA			
BIAND #xx:3,Rd	R:W NEXT								
BIAND #xx:3,@ERd	R:W 2nd	R:B EA	R:W NEXT						
BIAND #xx:3,@aa:8	R:W 2nd	R:B EA	R:W NEXT						

Instruction	1	2	3	4	5	6	7	8	9
SHAR.B #2,Rd	R:W NEXT								
SHAR.W Rd	R:W NEXT								
SHAR.W #2,Rd	R:W NEXT								
SHAR.L ERd	R:W NEXT								
SHAR.L #2,ERd	R:W NEXT								
SHLL.B Rd	R:W NEXT								
SHLL.B #2,Rd	R:W NEXT								
SHLL.W Rd	R:W NEXT								
SHLL.W #2,Rd	R:W NEXT								
SHLL.L ERd	R:W NEXT								
SHLL.L #2,ERd	R:W NEXT								
SHLR.B Rd	R:W NEXT								
SHLR.B #2,Rd	R:W NEXT								
SHLR.W Rd	R:W NEXT								
SHLR.W #2,Rd	R:W NEXT								
SHLR.L ERd	R:W NEXT								
SHLR.L #2,ERd	R:W NEXT								
SLEEP	R:W NEXT	Internal operation: M							