RX651 Group

SH7044 to RX651 Microcontroller Migration Guide

Introduction

This application note describes points requiring special attention, points of difference, etc., that need to be borne in mind when replacing the SH7044 with the RX651 in a user system. For detailed information on each function, refer to the latest version of the User's Manual: Hardware.

Target Device

RX651

Contents

1. CPU Architecture

1.1 Registers

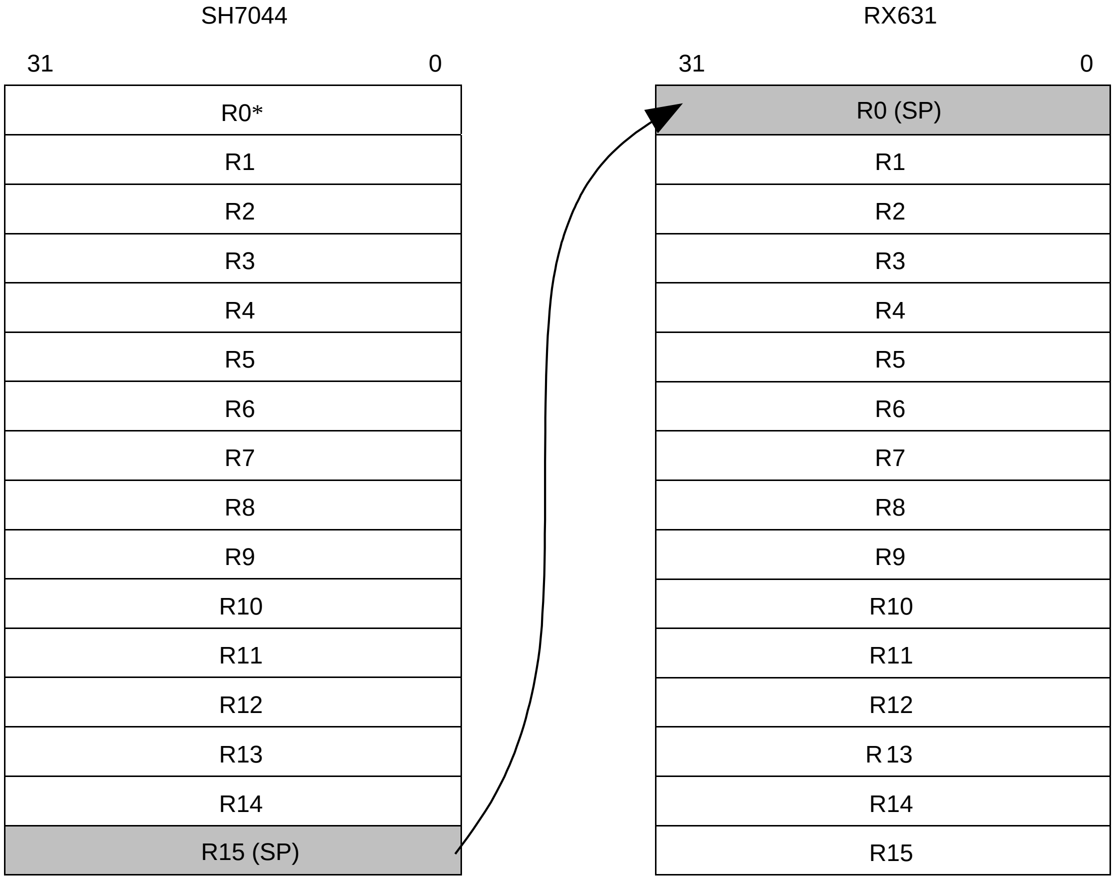
The points of difference between the registers of the SH7044 and the RX651 are described below.

1.1.1 General-Purpose Registers

The SH7044 and RX651 each have 16 32-bit general-purpose registers. They differ in that the register used as the stack pointer (SP) is different.

- SH7044: R15
- RX651: R0

Figure 1.1 shows the differences between general-purpose registers. On the SH7044, R0 is also used as an index register.



Note: * Used as the index register in the indexed register indirect and indexed GBR indirect addressing modes. R0 may be fixed as the source or destination register, depending on the instruction.

Figure 1.1 Differences Between General-Purpose Registers

1.1.2 Control Registers

The SH7044 and RX651 control registers have differences as shown in the differences in the control registers in Figure 1.2.

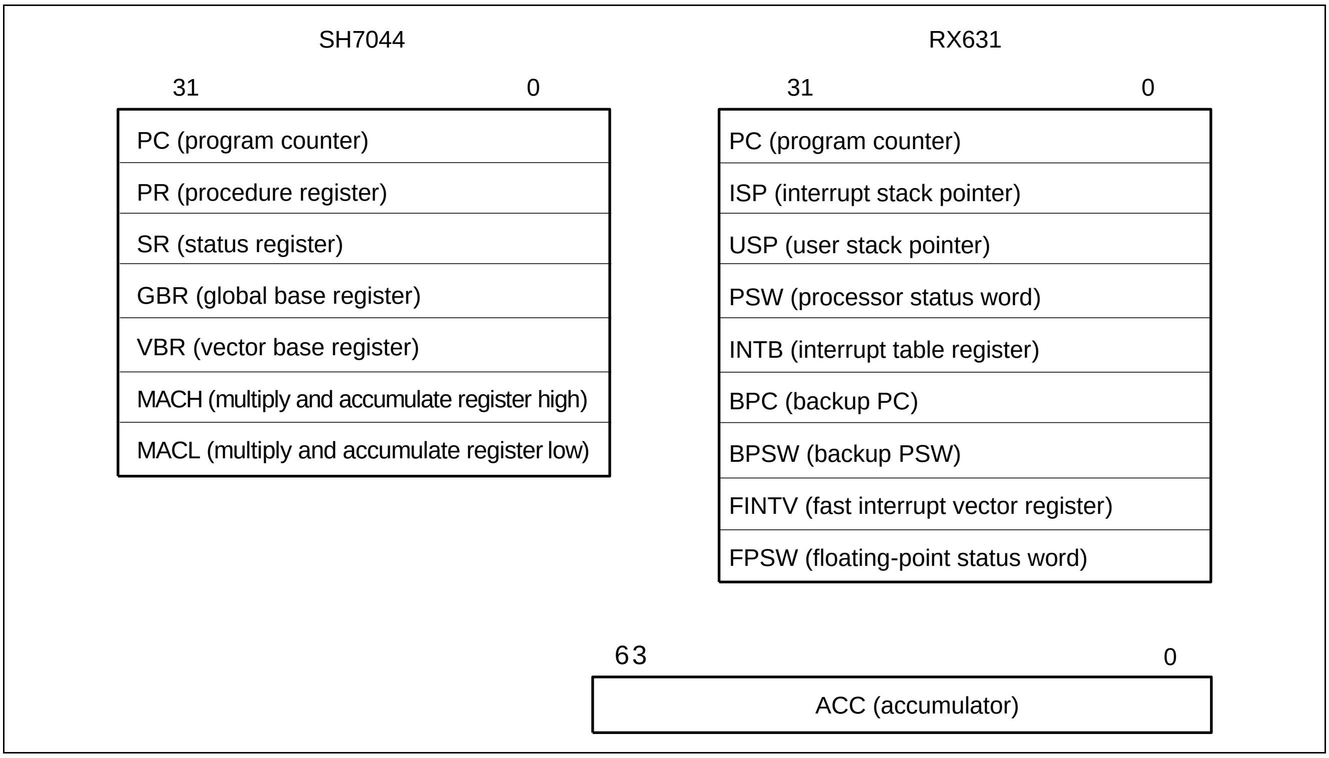

SH7044
31
0
PC (program counter)
PR (procedure register)
SR (status register)
GBR (global base register)
VBR (vector base register)
MACH (multiply and accumulate register high)
MACL (multiply and accumulate register low)
RX631
31
0
PC (program counter)
ISP (interrupt stack pointer)
USP (user stack pointer)
PSW (processor status word)
INTB (interrupt table register)
BPC (backup PC)
BPSW (backup PSW)
FINTV (fast interrupt vector register)
FPSW (floating-point status word)
63
0
ACC (accumulator)

Figure 1.2 Differences Between Control Registers

The RX651 has no registers corresponding to PR and GBR on the SH7044. The ACC register on the RX651 corresponds to MACH and MACL on the SH7044. An outline of the control registers that are implemented on the RX651 but not on the SH7044 is presented below.

Table 1.1 Control Registers available on just RX651

Register name	Explanation
Interrupt Stack Pointer (ISP) User Stack Pointer (USP)	The RX651 has two types of stack pointers. Whether the stack pointer operates as the ISP or USP depends on the value of the stack pointer select bit (U) in the processor status word (PSW).
Interrupt Table Register (INTB)*1	Specify the start address of the variable vector table.
Backup PC/Backup PSW（BPC/BPSW）	RX651 has normal interrupt and fast interrupt. With fast interrupts, the contents of the PC and PSW are saved to the dedicated registers (BPC and BPSW), so it is possible to shorten the processing time for saving the registers. BPC and BPSW do not support multiple interrupts.
Fast Interrupt Vector Register（FINTV）	A register that specifies the jump destination when a fast interrupt occurs.
Floating-Point Status Word（FPSW）	RX651 Register that shows various statuses of the operation result (Results of floating-point operations) of the on-chip FPU.
Exception Table Register（EXTB）	Specify the start address of the variable vector table.

Note:1 The functionality of this register is equivalent to that of VBR on the SH7044.

- Differences between status registers

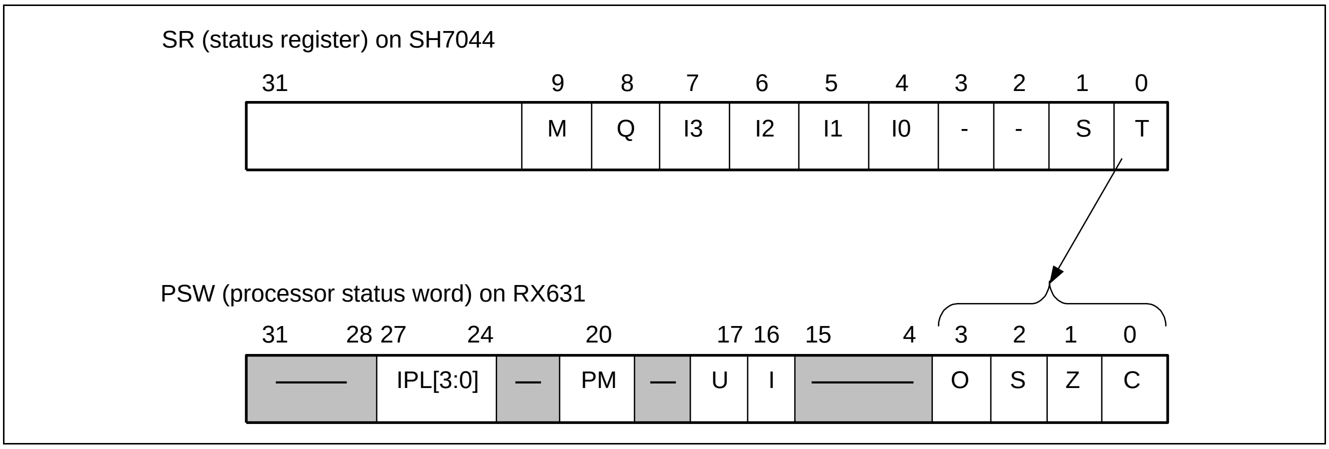


Figure 1.3 Differences Between SR (SH7044) and PSW (RX651)

Table 1.2 Differences Between SR (SH7044) and PSW (RX651)

SR Bit Name	PSW Bit Name	Description
T	C Z S O	The calculation result (true/false, carry/borrow, etc.) indicated by the T bit on the SH7044 is shown by four flags (C, Z, S, and O) on the RX651. C: Carry flag (0/1 = No carry has occurred./A carry has occurred.) Z: Zero flag S: Sign flag O: Overflow flag
S	—	Controls the functionality that prevents overflows during ALU arithmetic operations performed by the DSP unit of the SH7044. On the RX651 there is no bit corresponding to the S bit, and the occurrence of an overflow during a floating-point operation is reported by the FPSW flag. It is also possible to perform exception handling when an overflow occurs.
I0, I1, I2, I3	IPL[3:0]	These are the interrupt mask bits. Both the SH7044 and the RX651 support level settings from 0 (lowest) to 15 (highest). Only interrupts with a priority level higher than this setting are accepted.
Q	—	The Q bit is used by the DIV0U, DIV0S, and DIV1 instructions on the SH7044. There is no corresponding bit on the RX651.
M	—	The M bit is used by the DIV0U, DIV0S, and DIV1 instructions on the SH7044. There is no corresponding bit on the RX651.
—	I	Interrupt enable bit 0: Interrupts are disabled. 1: Interrupts are enabled. This bit is used to enable interrupt requests on the RX651. The initial state is 0, so it is necessary to set this bit to 1 in order to accept interrupts. Also, this bit is cleared to 0 when an exception is accepted, and no interrupts are accepted while its value remains 0. Note that the interrupt status flag of the interrupt controller is reset when an interrupt request occurs, regardless of the setting of this bit.
—	U	This bit specifies the stack pointer used by the RX651. 0: Interrupt stack pointer (ISP) 1: User stack pointer (USP) This bit is cleared to 0 when an exception is accepted.
—	PM	This bit specifies the processor mode of the RX651. 0: Supervisor mode 1: User mode This bit is cleared to 0 when an exception is accepted.

1.2 Option-Setting Memory(OFSM)

The RX651 has an option setting memory with registers that determine the MCU state after reset.

For the setting method of the option setting memory, refer to the User’s Manual: Hardware.

1.2.1 Outline of Option-Setting Memory

Option setting memory (OFSM) is a general term for the registers shown below (For detailed specifications, see the User’s Manual: Hardware).

b31 b0

Address	...	Register Description
FF7F FFE8h to FF7F FFEFh	UB code A	Codes necessary when using user boot mode. (Do not overwrite these codes when using USB boot mode.)
FF7F FFF0h to FF7F FFF7h	UB code B	
FF7F FFF8h to FF7F FFFBh	Endian select register B (MDEB) (user boot mode)	Register for selecting the endian setting of the CPU.
	—	—
FFFF FF80h to FFFF FF83h	Endian select register S (MDES) (single-chip mode)	Register for selecting the endian setting of the CPU.
	—	—
FFFF FF88h to FFFF FF8Bh	Option function select register 1 (OFS1)	The OFS1 register is used to make the following two settings: • Voltage monitor 0 reset is enabled/ disabled after a reset. • HOCO oscillation is enabled/ disabled after a reset.
FFFF FF8Ch to FFFF FF8Fh	Option function select register 0 (OFS0)	The OFS0 register is used to make settings for the independent watchdog timer (IWDT) and watchdog timer (WDT).
	...	—

Figure 1.4 Option-Setting Memory Area

1.2.2 Endian Setting

The SH7044 is fixed in big-endian mode. On the RX651, instructions are fixed in little-endian, and the data order is selectable between little-endian and big-endian. The endian setting is specified by means of the endian select bits (MDE[2:0]) in the MDES and MDEB registers in the option-setting memory.

When switching from the SH7044 to the RX651, it is possible to use big-endian order by specifying big-endian in the option settings of the genuine Renesas compiler. This allows migration without the need to be conscious of endianness in the user program.

The endian setting can be switched for each CS area in the external address space. However, instruction code cannot be allocated to an external space with an endian setting that differs from that of the chip. When allocating instruction code to an external space, ensure that an area with the same endian setting as the chip is used. (For details, see the User's Manual: Hardware.)

In actuality, Figure 1.5 Endian specification by compiler option :Endian Setting Example, is generated automatically according to the compiler option setting.*

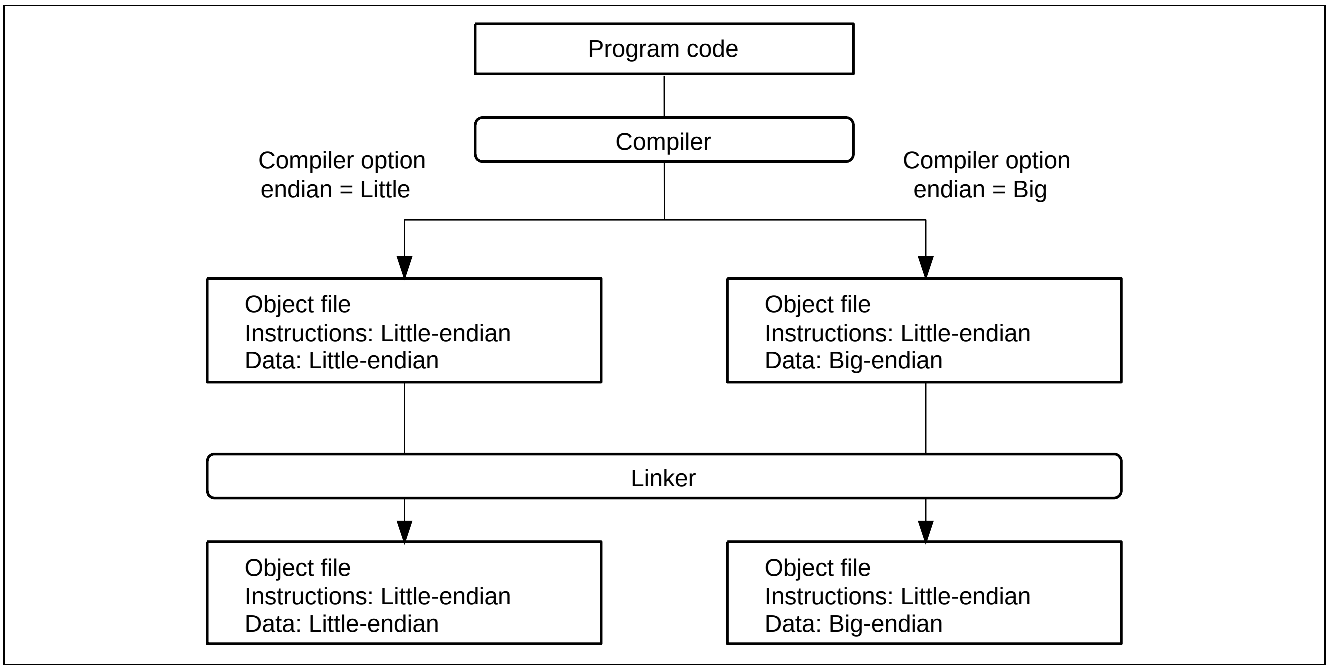


Figure 1.5 Specifying Endianness by Compiler Option

1.3 Reset Function

1.3.1 Reset Sources

SH7044 and RX651 reset sources are shown in table 1.3 Reset Sources.

Table 1.3 Reset Sources

	SH7044	RX651
Reset type	• Power-on reset (pin reset) • Manual reset (pin reset)	• RES# pin reset • Power-on reset (internal reset) • Voltage monitor 0 reset • Voltage monitor 1 reset • Voltage monitor 2 reset • Deep software standby reset • Independent watchdog timer reset • Watchdog timer reset • Software reset

(1) Reset vector configuration

The SH7044 has separate vectors for power-on resets and for manual resets (PC and SP).*

The RX651 has a single reset vector for multiple reset sources. The reset source is identified in reset status registers 0 to 2 during reset processing, and processing for the corresponding source is performed.

(2)Stack pointer

On the SH7044, it is necessary to specify the end address (+1) of the stack area in the reset vector. There is no stack pointer setting area in the vector table on the RX651, so the stack pointer is set in ISP and USP.

Note:1 See 1.7.4, Vector Configuration, for details of the vector tables.

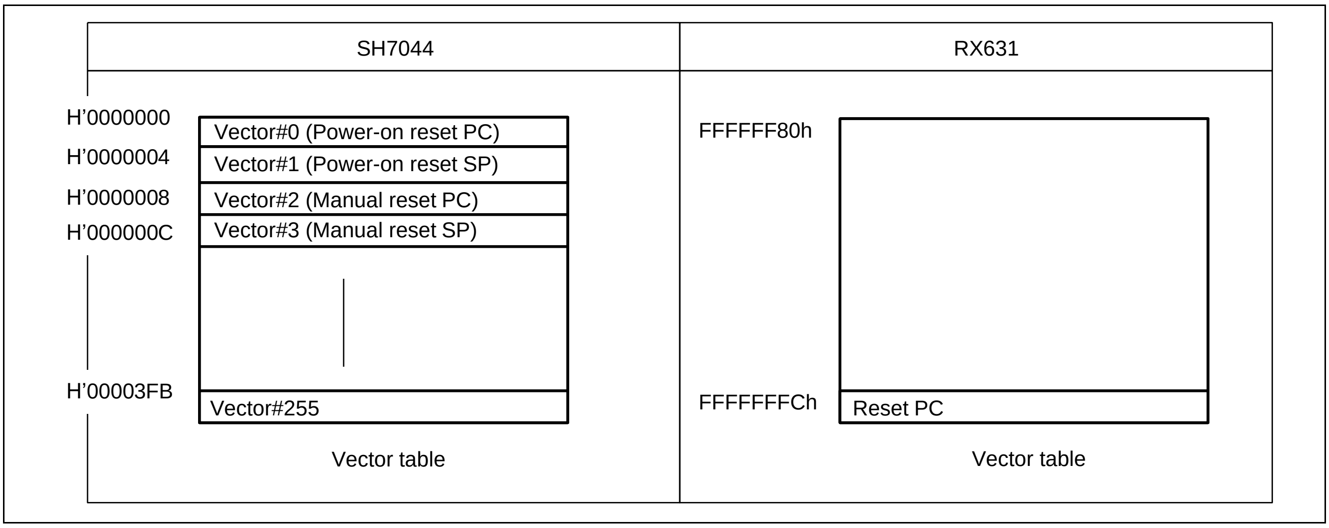


Figure 1.6 Reset Vectors on SH7044 and RX651

1.3.2 Reset Sources and Initialization Scope

The initialization scope of the reset sources differs between the SH7044 and the RX651. Table 1.4 lists the reset types and their initialization scope on the SH7044, and table 1.5 lists the reset types and their initialization scope on theRX651. (For details, see the User's Manual: Hardware.)

Table 1.4 SH7044 Reset Sources and Initialization Scope

Item	Power-On Reset	Manual Reset
CPU	○	○
On-chip peripheral modules	○	—

○: Reset —: No reset

Table 1.5 RX651 Reset Sources and Initialization Scope

	Reset Sources								
Reset Target	Res# Pin Reset	Power-On Reset	Voltage Monitor 0 Reset	Independent Watchdog Timer Reset	Watchdog Timer Reset	Voltage Monitor 1 Reset	Voltage Monitor 2 Reset	Deep Software Standby Reset	Software Reset
Power-on reset detection flag	○	—	—	—	—	—	—	—	—
Cold start/warm start determination flag	—	○	—	—	—	—	—	—	—
Voltage monitor 0 reset detection flag	○	○	—	—	—	—	—	—	—
Independent watchdog timer reset detection flag	○	○	○	—	—	—	—	○	—
Independent watchdog timer registers	○	○	○	—	—	—	—	○	—
Watchdog timer reset detection flag	○	○	○	○	—	—	—	○	—
Watchdog timer registers	○	○	○	○	—	—	—	○	—
Voltage monitor 1 reset detection flag	○	○	○	○	○	—	—	—	—
Voltage monitor function 1 registers	○	○	○	○	○	—	—	*1	—
Voltage monitor 2 reset detection flag	○	○	○	○	○	○	—	—	—
Voltage monitor function 2 registers	○	○	○	○	○	○	—	*2	—
Deep software standby reset detection flag	○	○	○	○	○	○	○	—	—
Software reset detection flag	○	○	○	○	○	○	○	○	—
Realtime clock registers	—	—	—	—	—	—	—	—	—
High-speed on-chip oscillator–related registers	○	○	○	○	○	○	○	—	○
Main clock oscillator–related registers	○	○	○	○	○	○	○	—	○
Pin states	○	○	○	○	○	○	○	—	○
Low power consumption–related registers	○	○	○	○	○	○	○	—	○
Registers other than the above, CPU, and internal state	○	○	○	○	○	○	○	○	○

○: Reset —: No change

Notes: 1. Only LVD1CR1 and LVD1SR are initialized.
2. Only LVD1CR2 and LVD2SR are initialized.

1.4 Clock Settings

1.4.1 Clock Sources

The clock sources and clock generation circuits of the SH7044 and RX651 are listed below.

Table 1.6 List of SH7044 and RX651 Clock Sources

SH7044	RX651
Oscillator (EXTAL and XTAL) + PLL circuit	• Main clock oscillator (EXTAL and XTAL) + PLL circuit • Subclock oscillator (XCIN and XCOUT) • High-speed on-chip oscillator (HOCO) • Low-speed on-chip oscillator (LOCO) • IWDT-dedicated on-chip oscillator

Note: In the description below, the high-speed on-chip oscillator is referred to as the HOCO and the low-speed on-chip oscillator as the LOCO.

1.4.2 Clock Generation Circuit

On the SH7044 clock control is not performed in software. Each peripheral device operations in synchronization with the system clock (ϕ) or a clock generated by the prescaler. On the RX651 a large variety of clocks operate under software control.

On the RX651 the LOCO operates as the clock source after a reset. The operation of necessary clock sources and PLL circuits other than the LOCO is started during system initialization, and various clocks are selected, such as the system clock and bus clocks. When making changes to clock-related settings, it is necessary to consider the register setting sequence and the oscillation and clock oscillation stabilization time.

See the following application note for details of the clock setting procedure.

RX63N Group, RX651 Group Initial Setting (R01AN3034EJ0211)

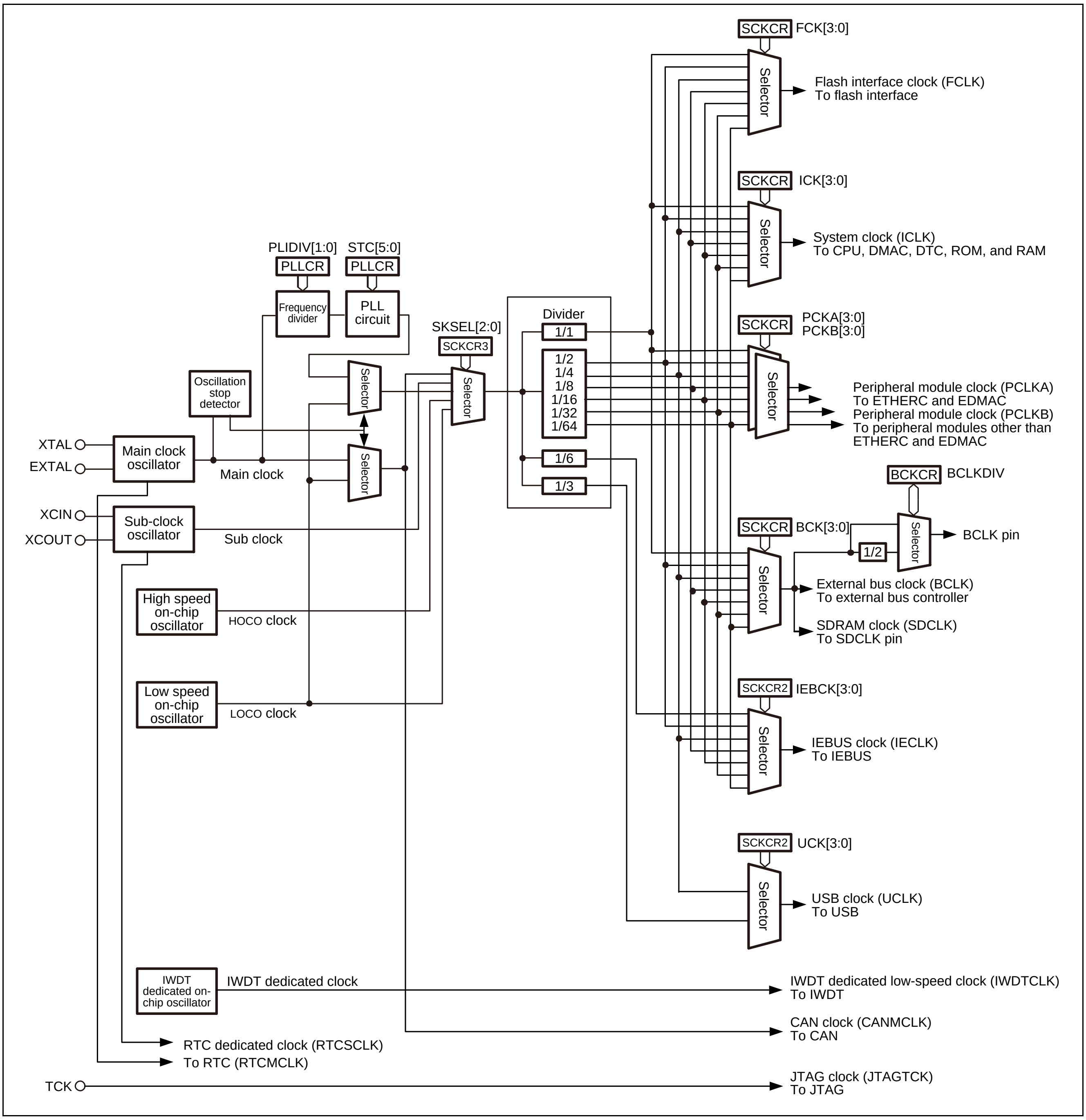


Figure 1.7 RX651 Clock Generation Circuit

1.5 Operation Modes

1.5.1 Comparison of Operation Modes

The table below shows a comparison of the operation modes of the SH7044 and RX651. For details of each operation mode, see the User’s Manual: Hardware.

Table 1.7 Comparison of Operation Modes

SH7044 Operation Mode	RX651 Operation Mode	Description
MCU mode 0 MCU mode 1	On-chip ROM disabled extended mode	An operation mode in which the on-chip ROM is disabled and the external address space is enabled. The external bus width differs from that of mode 0 and mode 1 on the SH7044.
MCU mode 2	On-chip ROM enabled extended mode	An operation mode in which the on-chip ROM is enabled and the external address space is enabled
Single-chip mode	Single-chip mode	An operation mode in which the on-chip ROM is enabled and the external address space is disabled
Boot mode	Boot mode (SCI interface)	An operation mode in which the flash memory modifying program (boot program), which is stored in a dedicated area internal to the microcontroller, is run. The on-chip flash memory can be programmed by a device external to the microcontroller by using the asynchronous serial interface (SCI1).
User program mode	Functionality equivalent to the SH7044 can be implemented in ordinary operation mode.	An operation mode that is only transitioned to when the setting value of the FWP pin changes and in which the on-chip flash memory is programmed by a programming/erase control program that has been prepared ahead of time by the user. It is possible to implement equivalent functionality in ordinary operation mode on the RX651, so it is not necessary to change the pin states.
PROM mode	Program rewriting of onboard RX is possible by emulator etc.	PROM mode can be programmed into the on-chip ROM using a general purpose PROM writer.
Writer mode	Program rewriting of onboard RX is possible by emulator etc.	The writer mode supports flash memory read mode, automatic write mode, automatic erase mode, and status read mode.
—	Boot mode (USB interface) / (FINE interface)	A mode in which the flash memory rewrite program (boot program) stored in the dedicated area inside the MCU operates. Using USB or fine, it is possible to rewrite the on-chip flash memory from the MCU external.

1.5.2 Comparison of Memory

The figure below shows a comparison of memory maps in on-chip ROM enabled mode (on-chip ROM enabled extended mode on the RX651).

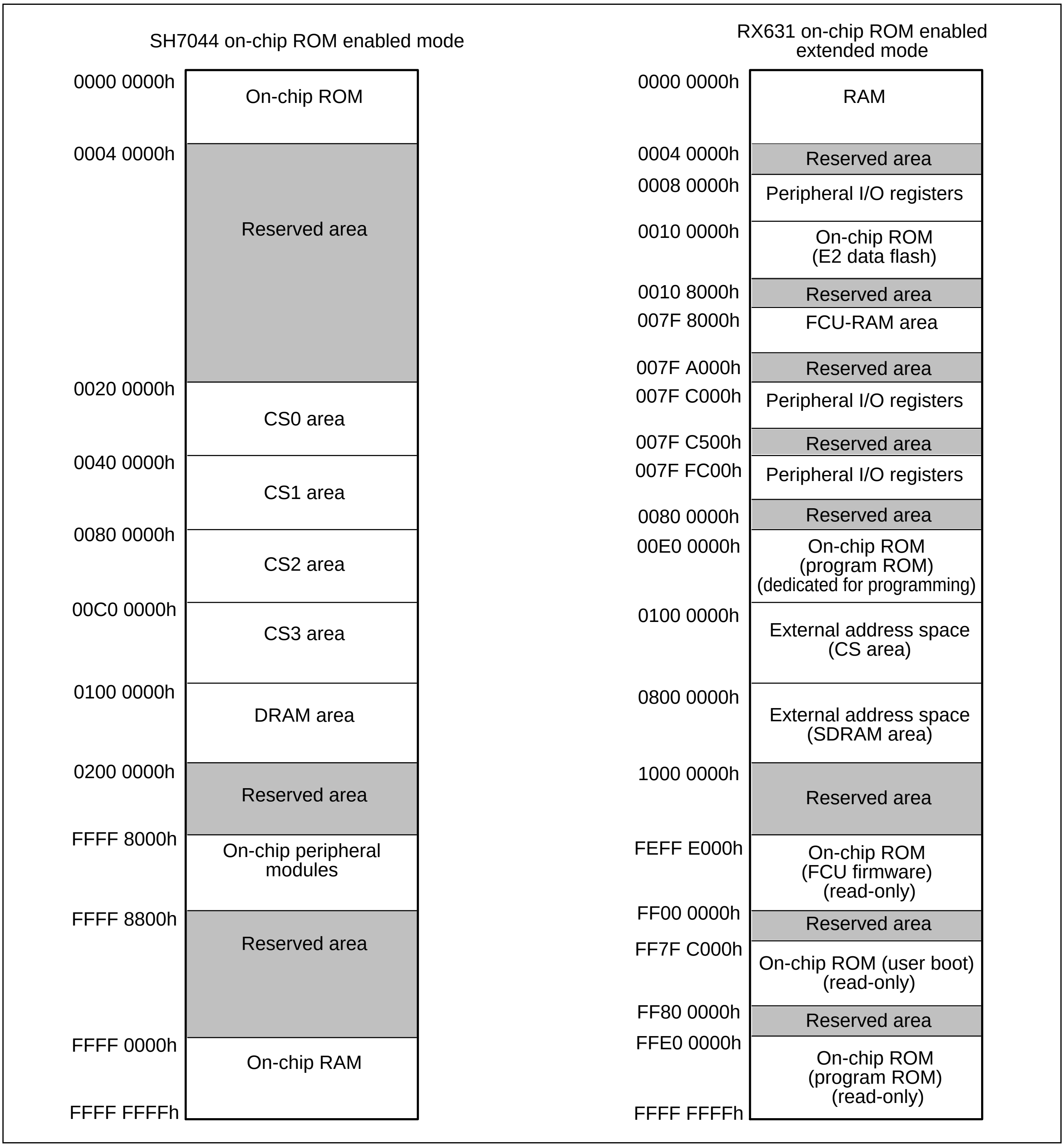


Figure 1.8 SH7044 and RX651 Memory Map Comparison (On-Chip ROM Enabled Mode)

The figure below shows a comparison of memory maps in single-chip mode.

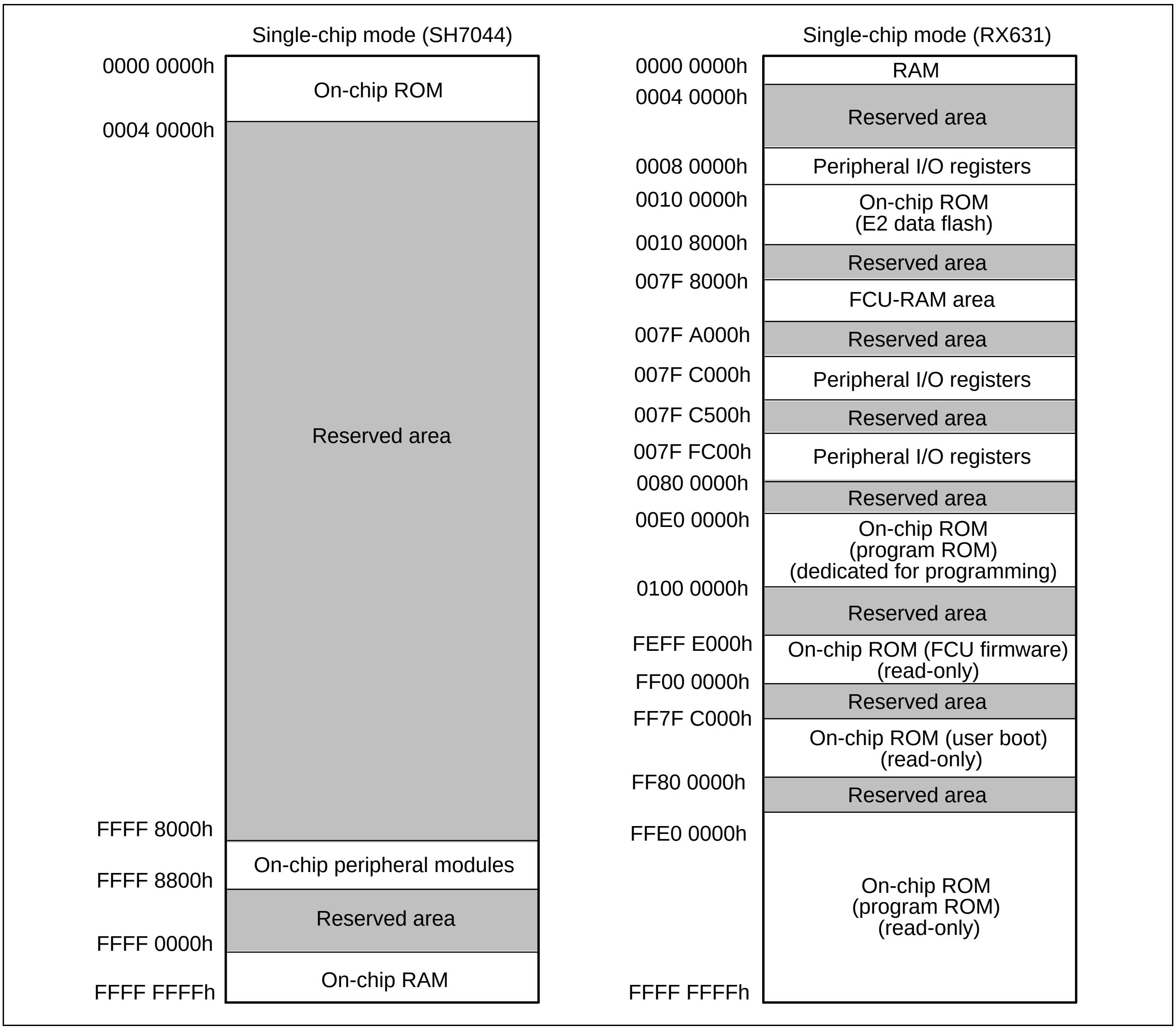


Figure 1.9 SH7044 and RX651 Memory Map Comparison (Single-Chip Mode)

The figure below shows a comparison of memory maps in on-chip ROM disabled mode.

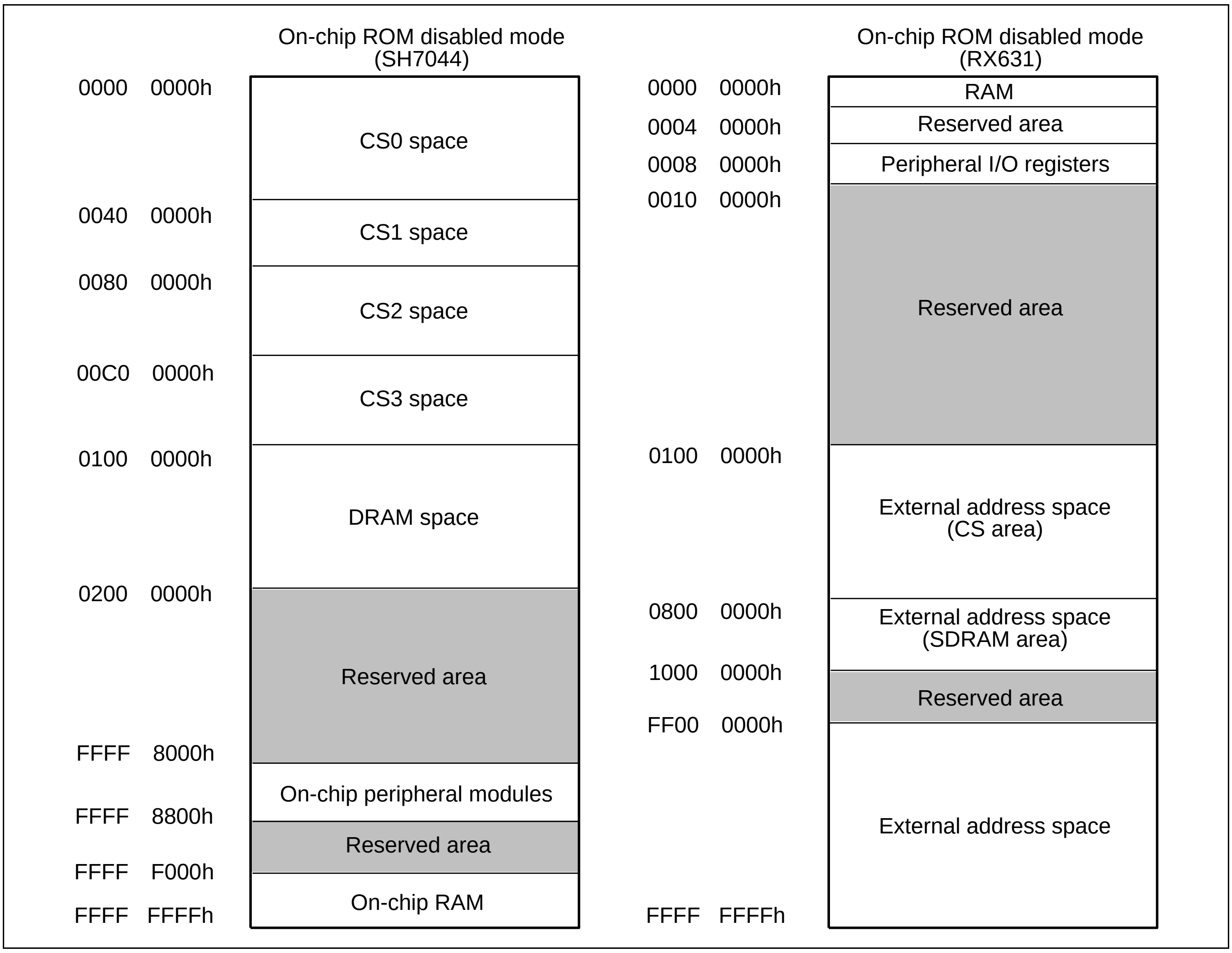


Figure 1.10 SH7044 and RX651 Memory Map Comparison (On-Chip ROM Disabled Mode)

- On the RX651 the RAM is allocated to addresses adjacent to 0000 0000h and ROM (for reading data) to addresses adjacent to FFFF FFFFh. Also, the RX651 has on-chip data flash for storing data.
- On the RX651 the peripheral IO registers are allocated within the address range from 0008 0000h to 000F FFFFh, and only the flash-related registers and peripheral clock notification register are allocated within the address range from 007F C000h to 007F FFFFh.
- On the RX651 the external address space is allocated within the address range from 0100 0000h to 0FFF FFFFh and configured as seven CS spaces of 16 MB each and a 128 MB SDRAM space.

1.5.3 Operation Mode Settings

Whereas on the SH7044 operation mode settings are made only with the MD1, MD0, and FWP pins, on the RX651 operation mode settings can be made by means of the MD pin and UB pin when a reset is canceled, or by software after a reset is canceled.

Table 1.8:RX651 terminal setting and operation mode lists the operation modes that are determined by pin settings, and table 1.9: RX651 SYSCR0 register setting and operation mode lists the operation modes that are set in software after a reset is canceled.

Table 1.8 Pin Settings and Operation Modes on RX651

Pin		
MD	UB	Mode Name
1	—	Single-chip mode
0	0	Boot mode
	1	Boot mode (USB interface) *1
0 → 1*1	0	Boot mode (FINE interface)

Notes: 1. After resetting the MD terminal to 0, switch between 20 and 100 msec.

Table 1.9 SYSCR0 Register Settings and Operation Modes on RX651

SYSCR0 Register*2		
ROME Bit*1	EXBE Bit	Mode Name
0 (On-chip ROM disabled) *1	0 (external bus disabled)	Single-chip mode
1 (On-chip ROM enabled)*2	0 (external bus disabled)*2	
0 (On-chip ROM disabled) *1	1 (external bus enabled)	On-chip ROM disabled extended mode
1 (On-chip ROM enabled)	1 (external bus enabled)	On-chip ROM enabled extended mode

Notes: 1. Once the ROME bit is set to 0, it cannot be reverted to 1.

2. After the STSCR0 register is reset, ROME = 1 and EXBE = 0.

1.6 Processor Modes

The RX CPU supports two processing modes: supervisor mode and user mode. These processor modes enable hierarchical CPU resource protection.

This makes it possible, when replacing the SH7044 with the RX651, to replace the software by operating in supervisor mode only, without using user mode. In other words, software can be replaced without the need to be conscious of the processor mode.

Table 1.10 Processor Modes

Processor Modes	Transition Conditions	Outline
Supervisor mode	• Reset cancellation • Exception occurrence (PSW.PM bit cleared to 0)	All CPU resources are accessible, and all instructions can be executed (no limitations). This is the mode in which the OS and other system programs ordinarily operate.
User mode	• PSW.PM bit set to 1 In this case, first set to 1 the PSW.PM bit saved to the stack, then execute the RTE instruction. Alternately, first set to 1 the PSW.PM bit saved to BPSW, then execute the RTFI instruction.	Write access to some CPU resources, such as some bits in PSW and to BPC and BPSW, is restricted, and privileged instructions cannot be used. This is the mode in which user programs such as application programs ordinarily operate.

Transitioning from supervisor mode to user mode

```
    MVFC      PSW,R1          ; The RTE instruction is used to simulate return from an exception.
    OR        #00100000h,R1   ;
    PUSH.L    R1              ;
    MVFC      PC,R1           ;
    ADD       #10,R1          ;
    PUSH.L    R1              ;
    RTE
    NOP
    NOP
```

Transitioning from user mode to supervisor mode

Operation transitions to supervisor mode when exception handling occurs. Operation then transitions again to user mode after the return from exception handling.

Another way to cause a transition to supervisor mode is to use an instruction that generates an unconditional trap, such as the INT instruction or BRK instruction.

1.7 Exception Handling

The points of difference regarding exception handling in general on the SH7044 and RX651, including interrupts, are described below.

1.7.1 Types of Exception Handling

A comparative listing of exception sources on the SH7044 and RX651 is shown below.

Table 1.11 Exception Sources on SH7044 and RX651

SH7044	RX651	Main Points of Difference
Power-on reset	Reset	On the SH7044 there are separate vectors for power-on resets and manual resets. On the RX651 there is a single reset vector. The reset source is identified in reset status registers 0 to 2 during reset interrupt handling, and appropriate processing is performed.
Manual reset		
Address error	Access exception	On the SH7044 this exception occurs when an attempt is made to access an access-prohibited area or an address to which access is prohibited. On the RX651 this exception occurs when a memory protection error occurs. On the SH7044 the next instruction is saved to PC when this exception occurs. On the RX651 the instruction that generated this exception is saved to PC.
CPU address error	—	
DMAC / DTC address error	—	
Interrupt (NMI)	Non-maskable interrupt	None
Interrupt (external/internal)	Interrupt (external/internal)	The RX651 also supports fast interrupts (level 15)
TRAP instruction (TRAPA instruction)	Unconditional trap (INT, BRK instruction)	The SH7044 has 32 sources, but the RX651 has 16 sources with dedicated vectors and up to 256 sources when sources also used for interrupts are included.
General illegal instruction	Undefined instruction exception	In SH7044, if an undefined code other than the delayed branch instruction (delay slot) is decoded, a general unjust instruction occurs. When an undefined code placed in a delayed branch instruction (delay slot) or an instruction that rewrites a PC is decoded, a slot injustice instruction occurs. RX651 raises an undefined instruction exception when it detects the execution of an undefined instruction.
Illegal slot instruction		Initiated when an instruction to rewrite an undefined code or PC placed in a deferred branch instruction (delay slot) is decoded.
—	Privileged instruction	There are no exceptions equivalent to privileged instruction exceptions and floating-point exceptions in SH7044. In SH7044, the PC of the following instruction is evacuated when this exception occurs, and the PC of exception generation instruction is evacuated in RX651.
—	Floating-point exception	

1.7.2 Exception Handling Priority

The comparative priority of exception sources on the SH7044 and RX651 is shown below.

Table 1.12 Exception Event Priority

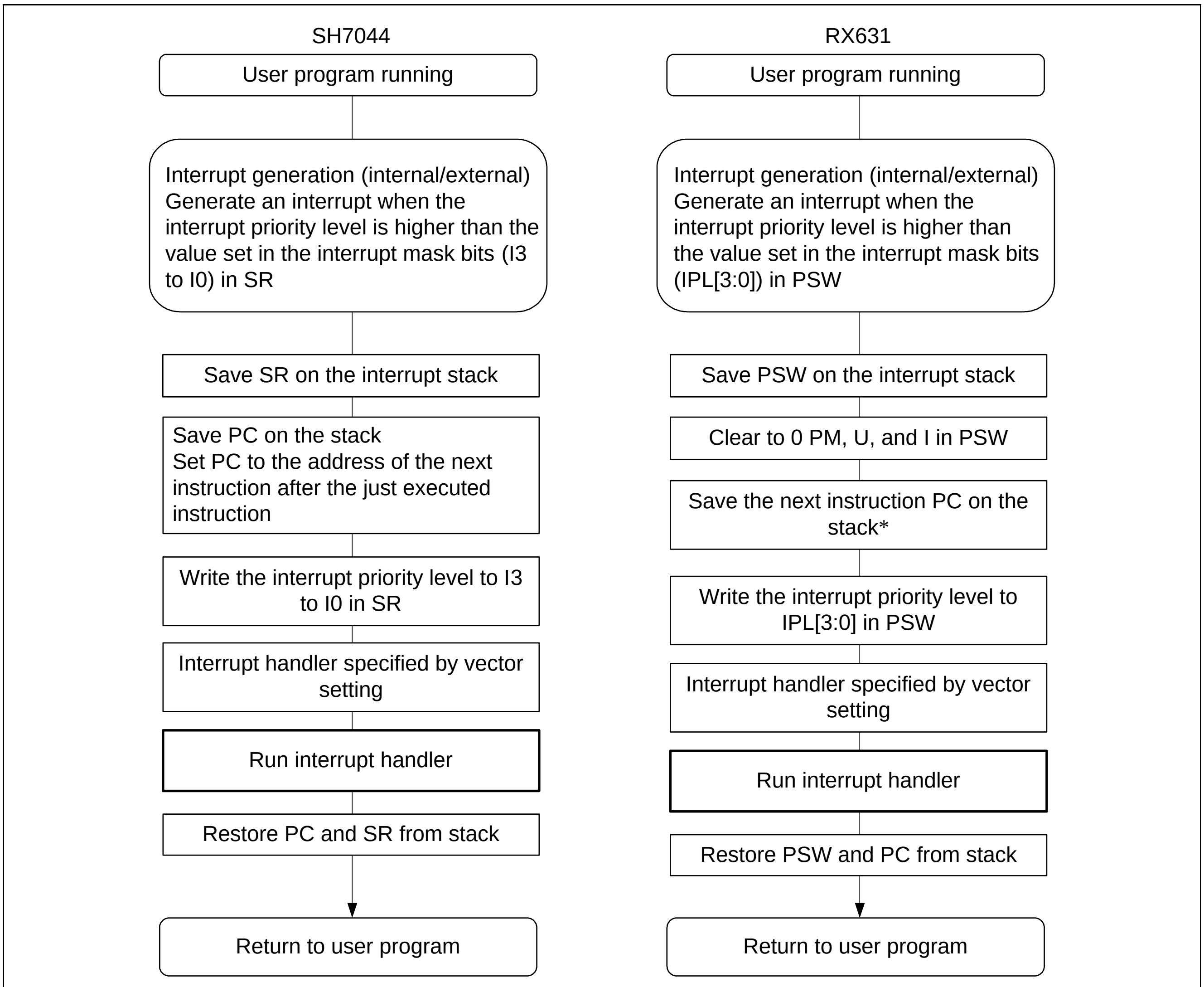
Priority	SH7044	RX651	Remarks
High	Power-on reset	Reset	
	Manual reset	Non-maskable interrupt	
	Address error exception	Interrupt (external/internal)	
	Interrupt (NMI)	Instruction access exception	
	Interrupt (external/internal)	Undefined instruction exception, privileged instruction exception	
	TRAP instruction	Unconditional trap	
	General illegal instruction exception	Operand access exception	
Low	Illegal slot instruction exception	Floating-point exception	

Note: Among interrupts, the priority is determined by the interrupt controller.

On the SH7044 address errors have higher priority than interrupts (internal or external), but on the RX651 both instruction access exceptions and operand access exceptions have lower priority than interrupts.

1.7.3 Basic Processing Sequence of Exception Handling

The basic processing sequence interrupt exception handling on the SH7044 and RX651 is shown below.


SH7044
User program running
Interrupt generation (internal/external) Generate an interrupt when the interrupt priority level is higher than the value set in the interrupt mask bits (I3 to I0) in SR
Save SR on the interrupt stack
Save PC on the stack
Set PC to the address of the next instruction after the just executed instruction
Write the interrupt priority level to I3 to I0 in SR
Interrupt handler specified by vector setting
Run interrupt handler
Restore PC and SR from stack
Return to user program
RX631
User program running
Interrupt generation (internal/external) Generate an interrupt when the interrupt priority level is higher than the value set in the interrupt mask bits (IPL[3:0]) in PSW
Save PSW on the interrupt stack
Clear to 0 PM, U, and I in PSW
Save the next instruction PC on the stack*
Write the interrupt priority level to IPL[3:0] in PSW
Interrupt handler specified by vector setting
Run interrupt handler
Restore PSW and PC from stack
Return to user program

Note: * When the instruction being executed is RMPA, SCMPU, SMOVB, SMOVF, SMOVU, SSTR, SUNTIL, or SWHILE, the PC of the instruction being executed is saved on the stack.

Figure 1.11 Interrupt (Internal/External) Processing Sequence

1.7.4 Vector Configuration

Both the SH7044 and RX651 have a relocatable vector configuration, which allows the vector table to be reallocated. On the SH7044 the vector base register (VBR) specifies the start of the vector table. (Note that VBR is initialized to 0 after a reset, so it is not possible to change the reset vector.)

The RX651's INTB (interrupt table register) points to the beginning of the interrupt vector table, and EXTB (exception table register) points to the beginning of the exception vector table.

The interrupt vector table has been assigned an interrupt and an unconditional trap that can be repositioned.

The exception vector table has been assigned a system exception.

The RX651 reset is a fixed vector.

Also, the fast interrupt vector is set in the FINTV register.

Figure 1.12 Vector table settings shows the difference between vector tables.

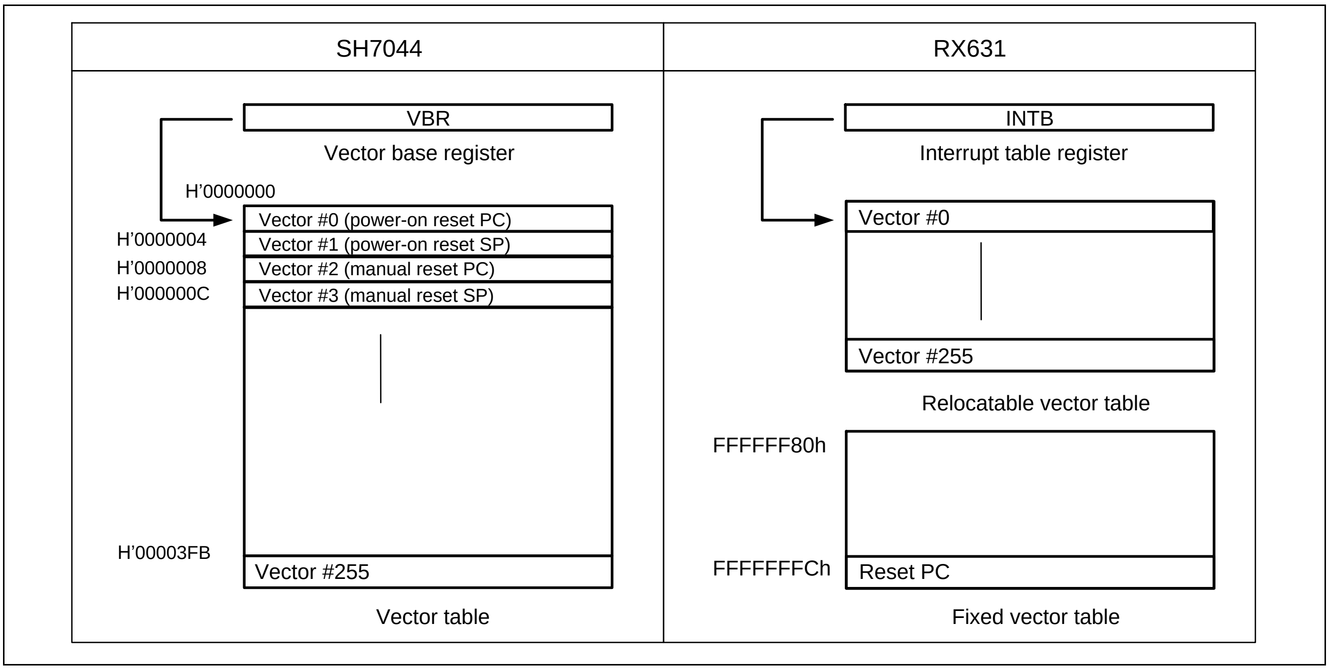


Figure 1.12 Vector Table Settings

1.7.5 Interrupt Masking by SR (SH7044) and PSW (RX651)

On the RX651 the I bits in control register PSW are used to set the interrupt mask level. The I bits indicate which interrupts are enabled and which are disabled.

Table 1.13 Interrupt-Related Bits in SR and PSW

SH7044	RX651	
SR Register	**PSW Register**	**Description**
I0, I1, I2, I3	IPL[3:0]	CPU interrupt mask level (priority level) Setting value: 0 to Fh (levels 0 to 15) When an interrupt request occurs, this level setting is compared with the priority level set for the individual interrupt source, and the interrupt is enabled if its level setting is higher than the mask level.
—	I	Interrupt enable bit 0: Interrupts are disabled. 1: Interrupts are enabled. When an interrupt occurs, the interrupt status flag in the interrupt controller is set to 1. After a system reset, this bit is set to 1, enabling acceptance of interrupts. When an exception is accepted, this bit is cleared to 0 and no interrupts are accepted while its value remains 0.

1.8 Interrupt Handling

This chapter describes the differences in interrupt processing, focusing on the interrupt controller.

1.8.1 Interrupt Controller

Table 1.14 Comparison of interrupt controller specifications lists the differences in the interrupt controller specifications.

Table 1.14 Comparison of Interrupt Controller Specifications

Item		SH7044	RX651
Interrupts	Peripheral function interrupts	• Interrupts from peripheral modules • Interrupt detection: Edge/level*1	• Interrupts from peripheral modules • Interrupt detection: Edge/level*1 • Group interrupt function • Software configurable interrupt B function • Software configurable interrupt A function
	External pin interrupts	• IRQ0 to IRQ7 pins • Sources: 8 • Interrupt detection: Low level or falling edge can be specified for each source.	• IRQ0 to IRQ15 pins • Sources: 16 • Interrupt detection: Low level, falling edge, rising edge, or both edges can be specified for each source. • Digital filter function support
	Software interrupts	None	Supported
	Interrupt priority	A level from 0 to Fh can be specified for each source by a register setting.	The priority level is set by the interrupt factor priority register r (IPRr) (r = 000 to 255).
	Fast interrupt function	None	Supported
	DTC and DMAC control	Activation supported*2	Activation supported
	EXDMAC control	None	EXDMAC bootable with software configurable interrupts
	User break interrupt	Supported	None It can be supported by the debugger function of the emulator
Non-maskable interrupts	NMI pin interrupts	• Interrupt detection method (selection of falling or rising edge) • NMI input level read bit provided	• Interrupt detection method (selection of falling or rising edge) • Digital filter function
	Other sources (Other than exception handling)	• • None	• Interrupt at oscillation stop detection • WDT underflow or refresh error • IWDT underflow or refresh error • Voltage monitor 1 interrupt Voltage monitor 2 interrupt • RAM error interrupt

Notes: 1. The detection method is fixed for fixed-connection peripheral modules.

2. On the SH7044 activation source setting is performed on the DTC or DMAC.

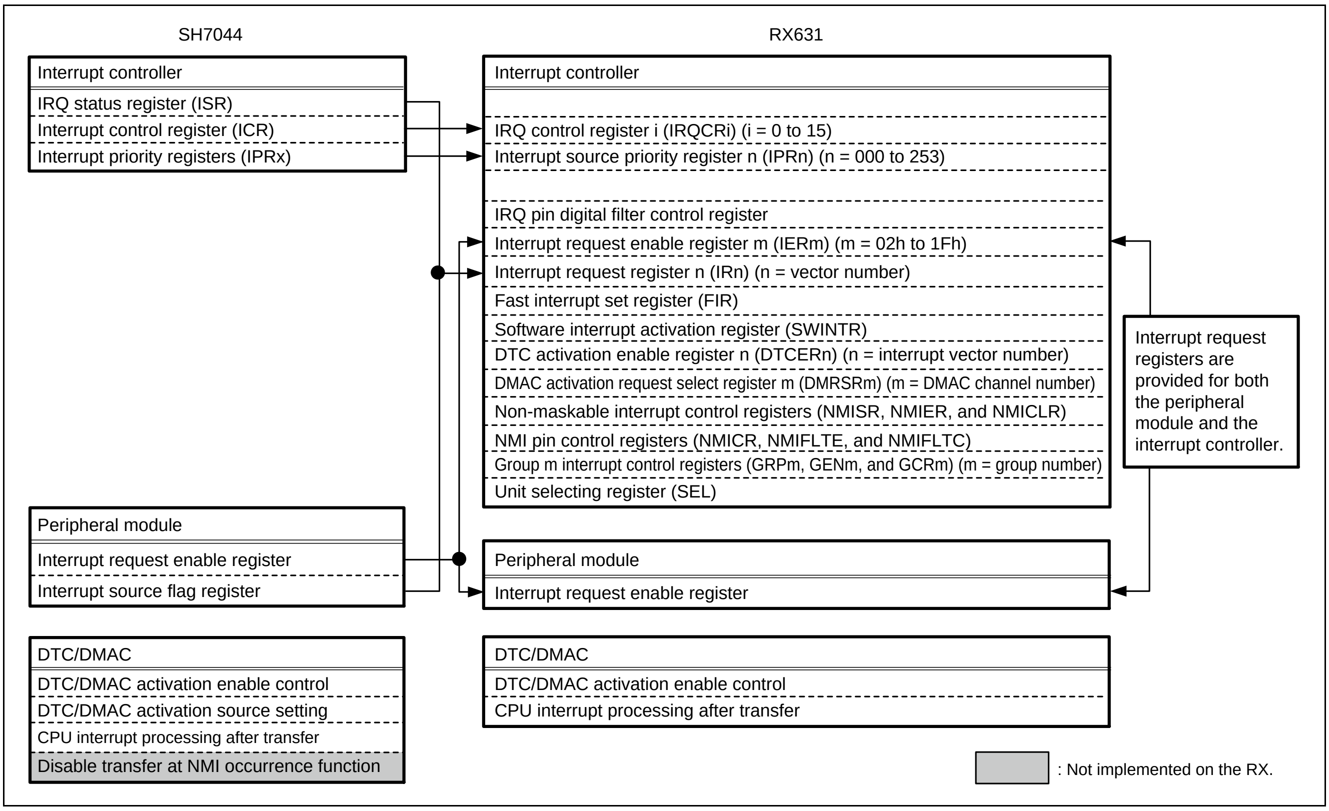

SH7044
RX631
Interrupt controller
IRQ status register (ISR)
Interrupt control register (ICR)
Interrupt priority registers (IPRx)
Interrupt controller
IRQ control register i (IRQCRi) (i = 0 to 15)
Interrupt source priority register n (IPRn) (n = 000 to 253)
IRQ pin digital filter control register
Interrupt request enable register m (IERm) (m = 02h to 1Fh)
Interrupt request register n (IRn) (n = vector number)
Fast interrupt set register (FIR)
Software interrupt activation register (SWINTR)
DTC activation enable register n (DTCERn) (n = interrupt vector number)
DMAC activation request select register m (DMRSRm) (m = DMAC channel number)
Non-maskable interrupt control registers (NMISR, NMIER, and NMICLR)
NMI pin control registers (NMICR, NMIFLTE, and NMIFLTC)
Group m interrupt control registers (GRPm, GENm, and GCRm) (m = group number)
Unit selecting register (SEL)
Interrupt request registers are provided for both the peripheral module and the interrupt controller.
Peripheral module
Interrupt request enable register
Interrupt source flag register
Peripheral module
Interrupt request enable register
DTC/DMAC
DTC/DMAC activation enable control
DTC/DMAC activation source setting
CPU interrupt processing after transfer
Disable transfer at NMI occurrence function
DTC/DMAC
DTC/DMAC activation enable control
CPU interrupt processing after transfer
: Not implemented on the RX.

Figure 1.13 Differences Between Interrupt Controller Registers

Figure 1.13 shows the differences between the interrupt controllers of the SH7044 and RX651.

The interrupt controller of the SH7044 controls IRQ interrupt flags, while peripheral module interrupt flags are controlled by the peripheral modules.

On the RX651 the interrupt controller controls all interrupt status flags, for both IRQs and peripheral modules.* In addition, the interrupt controller controls the activation source settings for the DTC and DMAC. The disable transfer at NMI occurrence function of the DTC and DMAC on the SH7044 is not implemented on the RX651.

Note: * The interrupt controller contains an interrupt request register for each interrupt source, but there are also interrupt enable bits implemented in the peripheral modules. (For details, see the User’s Manual: Hardware.)

1.8.2 Interrupt Flag Management

When a peripheral module of the SH7044 generates an interrupt by edge detection, the corresponding interrupt flag (interrupt source flag) in the interrupt handler is cleared (the flag is cleared and a dummy read is performed). This is done because the interrupt will be generated once again if the flag is not cleared by the handler. On the RX651 the interrupt flags (interrupt status flags) are managed internally by the interrupt controller. The interrupt controller has a function whereby when it sends an interrupt request to the CPU or DTC/DMAC and receives a response indicating that it was accepted, it automatically clears the corresponding interrupt status flag. It is therefore not necessary to clear the flag and do a dummy read as on the SH7044. Note that in the case of interrupts generated by level detection the source flags reside in the peripheral modules, so they do need to be cleared. For details, see the User’s Manual: Hardware.

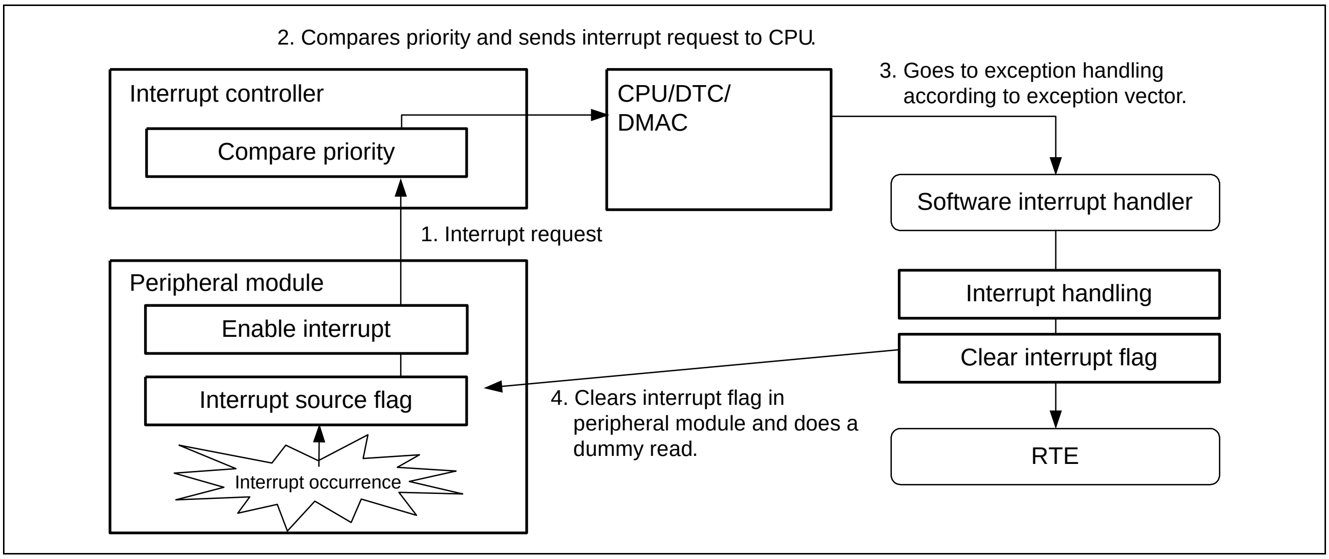

2. Compares priority and sends interrupt request to CPU.
Interrupt controller
Compare priority
CPU/DTC/ DMAC
3. Goes to exception handling according to exception vector.
Software interrupt handler
1. Interrupt request
Peripheral module
Enable interrupt
Interrupt source flag
Interrupt occurrence
Interrupt handling
Clear interrupt flag
4. Clears interrupt flag in peripheral module and does a dummy read.
RTE

Figure 1.14 SH7044 Peripheral Module Interrupt (Edge Detection)

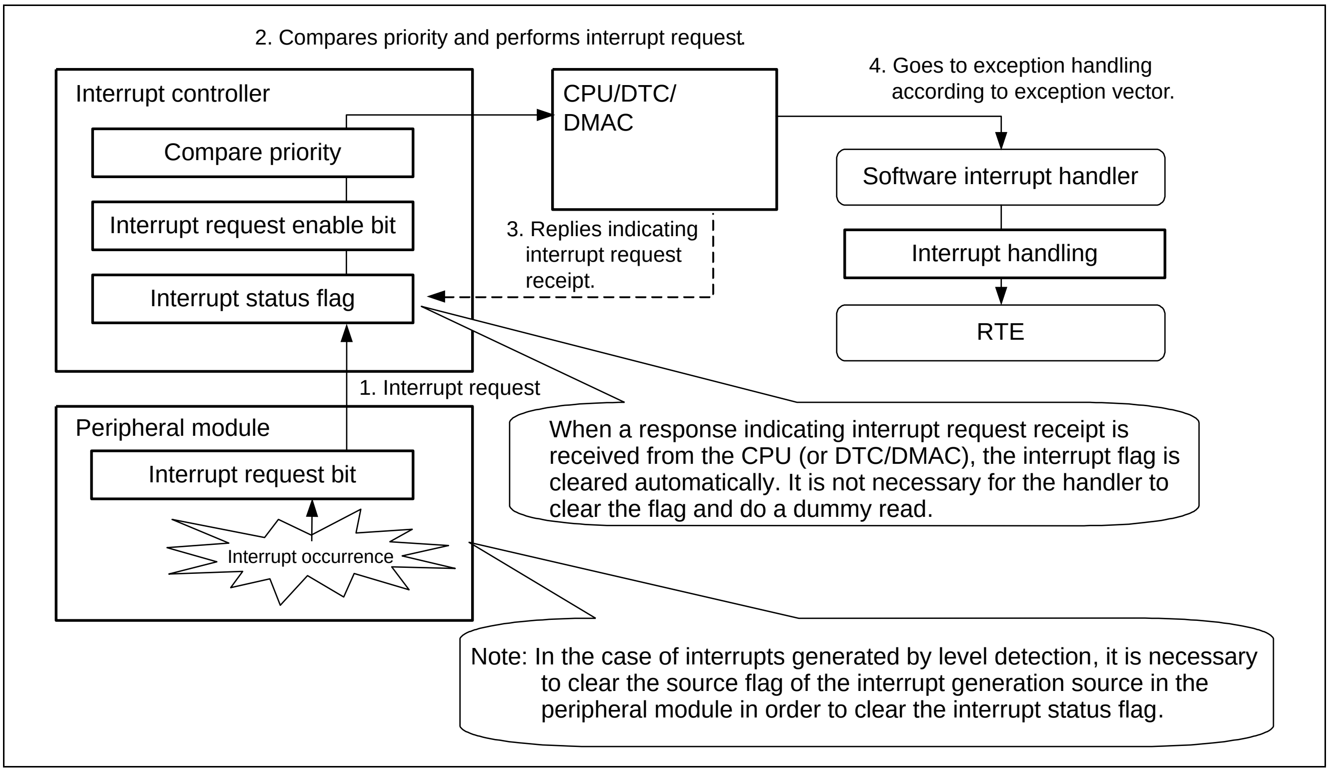

2. Compares priority and performs interrupt request.
Interrupt controller
Compare priority
Interrupt request enable bit
Interrupt status flag
CPU/DTC/ DMAC
4. Goes to exception handling according to exception vector.
Software interrupt handler
Interrupt handling
RTE
3. Replies indicating interrupt request receipt.
1. Interrupt request
Peripheral module
Interrupt request bit
Interrupt occurrence
When a response indicating interrupt request receipt is received from the CPU (or DTC/DMAC), the interrupt flag is cleared automatically. It is not necessary for the handler to clear the flag and do a dummy read.
Note: In the case of interrupts generated by level detection, it is necessary to clear the source flag of the interrupt generation source in the peripheral module in order to clear the interrupt status flag.

Figure 1.15 RX651 Peripheral Module Interrupt (Edge Detection)

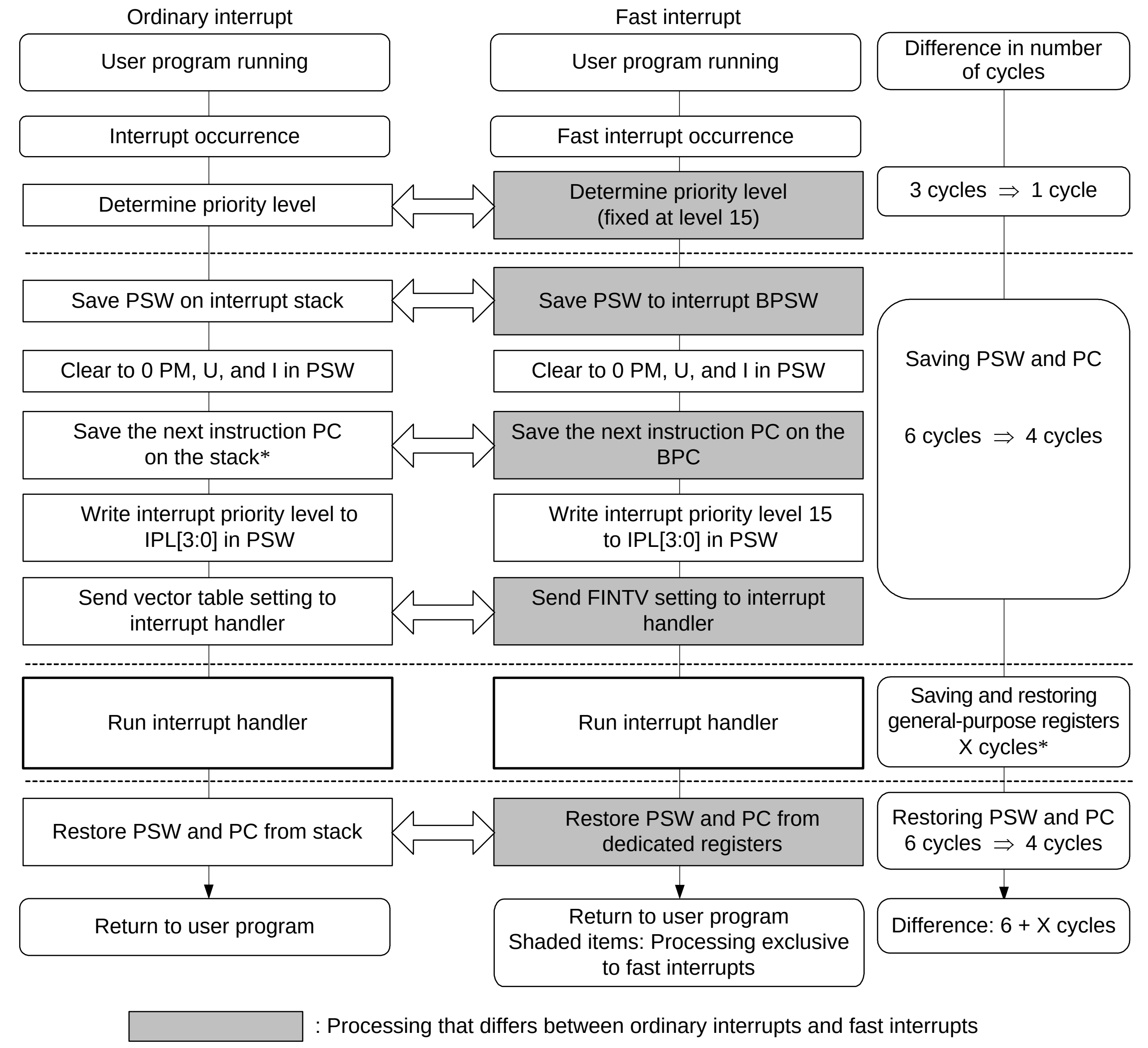
1.8.3 Fast Interrupt Control

In addition to ordinary interrupts, the RX651 supports fast interrupts.

Ordinary interrupt: After determining the interrupt priority it is necessary to save the contents of the control registers and general-purpose registers to the internal RAM or the external RAM by software.

Fast interrupt: Operation gives the interrupt the highest priority. When the interrupt occurs, the contents of the control registers are saved to dedicated registers, allowing interrupt activation to be realized faster than an ordinary interrupt.

It is possible to assign a portion of the general-purpose registers to exclusive use for interrupts by setting a compiler option. This eliminates the need to save and restore the contents of the general-purpose registers, further speeding up the interrupt.


Ordinary interrupt
Fast interrupt
User program running
User program running
Difference in number of cycles
Interrupt occurrence
Fast interrupt occurrence
Determine priority level
Determine priority level (fixed at level 15)
3 cycles ⇒ 1 cycle
Save PSW on interrupt stack
Save PSW to interrupt BPSW
Saving PSW and PC
6 cycles ⇒ 4 cycles
Clear to 0 PM, U, and I in PSW
Clear to 0 PM, U, and I in PSW
Save the next instruction PC on the stack*
Save the next instruction PC on the BPC
Write interrupt priority level to IPL[3:0] in PSW
Write interrupt priority level 15 to IPL[3:0] in PSW
Send vector table setting to interrupt handler
Send FINTV setting to interrupt handler
Run interrupt handler
Run interrupt handler
Saving and restoring general-purpose registers X cycles*
Restore PSW and PC from stack
Restore PSW and PC from dedicated registers
Restoring PSW and PC 6 cycles ⇒ 4 cycles
Return to user program
Return to user program Shaded items: Processing exclusive to fast interrupts
Difference: 6 + X cycles
▨ : Processing that differs between ordinary interrupts and fast interrupts

Note: * Fast interrupts have a single save register stage. Therefore, multistage fast interrupts are not possible. It is necessary to decide on a single function that will use fast interrupts.

Figure 1.16 Differences Between Ordinary Interrupts and Fast Interrupts

1.8.4 Digital Filter

The RX651 is provided with a digital filter function for the IRQ and NMI level signals. The sampling clock for the digital filter can be specified, and interrupt signals that do not last for at least three cycles of the sampling clock base are not accepted. This improves the system's noise tolerance.

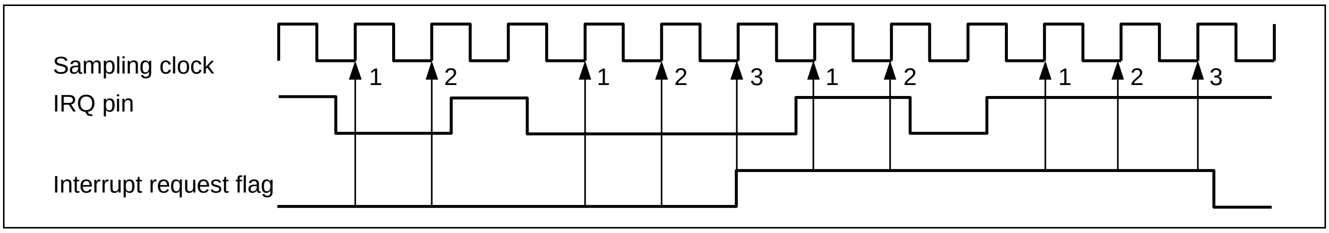


Figure 1.17 Digital Filter Operation Example

1.8.5 Multiple Interrupts

On the SH7044 if a high-priority interrupt occurs while a low-priority interrupt handler is running, the low-priority interrupt handler is suspended and the high-priority interrupt handler is executed. Once the high-priority interrupt handler finishes, the suspended low-priority interrupt handler is restarted.

On the RX651 if a high-priority interrupt occurs while a low-priority interrupt handler is running, the high-priority interrupt is not accepted until the low-priority interrupt handler finishes. This is because the PSW.I bit is cleared to 0 (interrupts are disabled) in a normal interrupt handler. In order to realize handling of multiple interrupts equivalent to that of the SH7044, it is necessary to set the PSW.I bit to 1 (interrupts are enabled) in the interrupt handler.

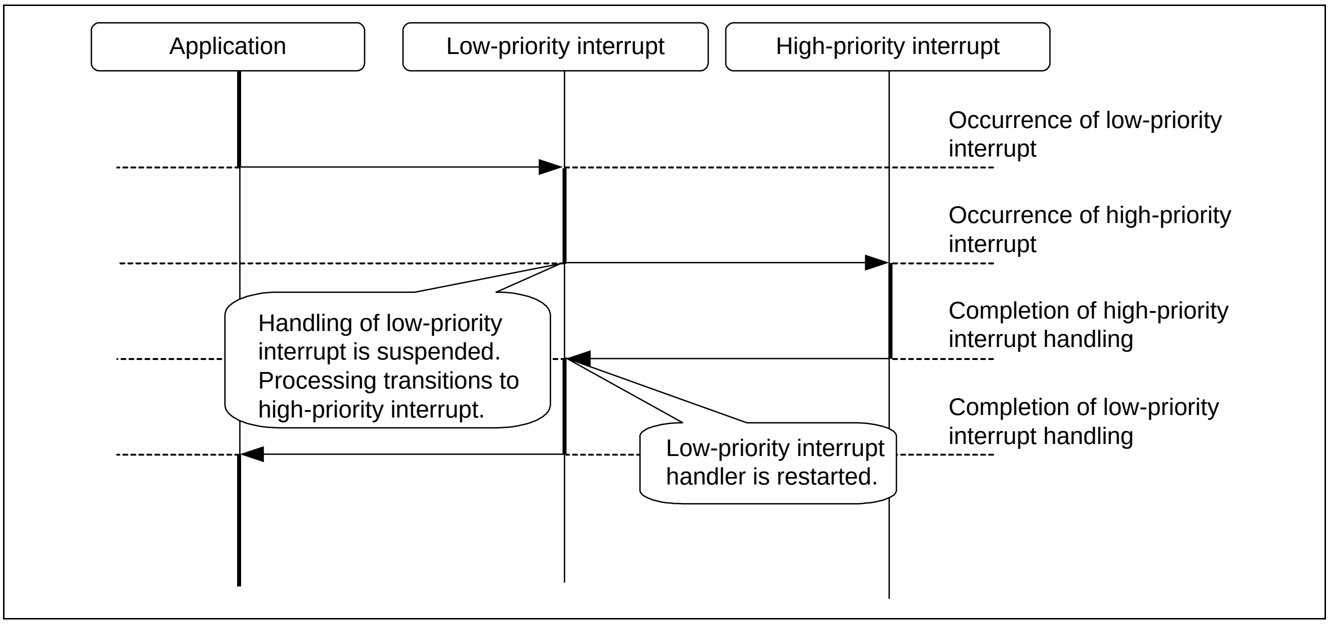

Application
Low-priority interrupt
High-priority interrupt
Occurrence of low-priority interrupt
Occurrence of high-priority interrupt
Completion of high-priority interrupt handling
Completion of low-priority interrupt handling
Handling of low-priority interrupt is suspended. Processing transitions to high-priority interrupt.
Low-priority interrupt handler is restarted.

Figure 1.18 SH7044 Multiple Interrupt Sequence

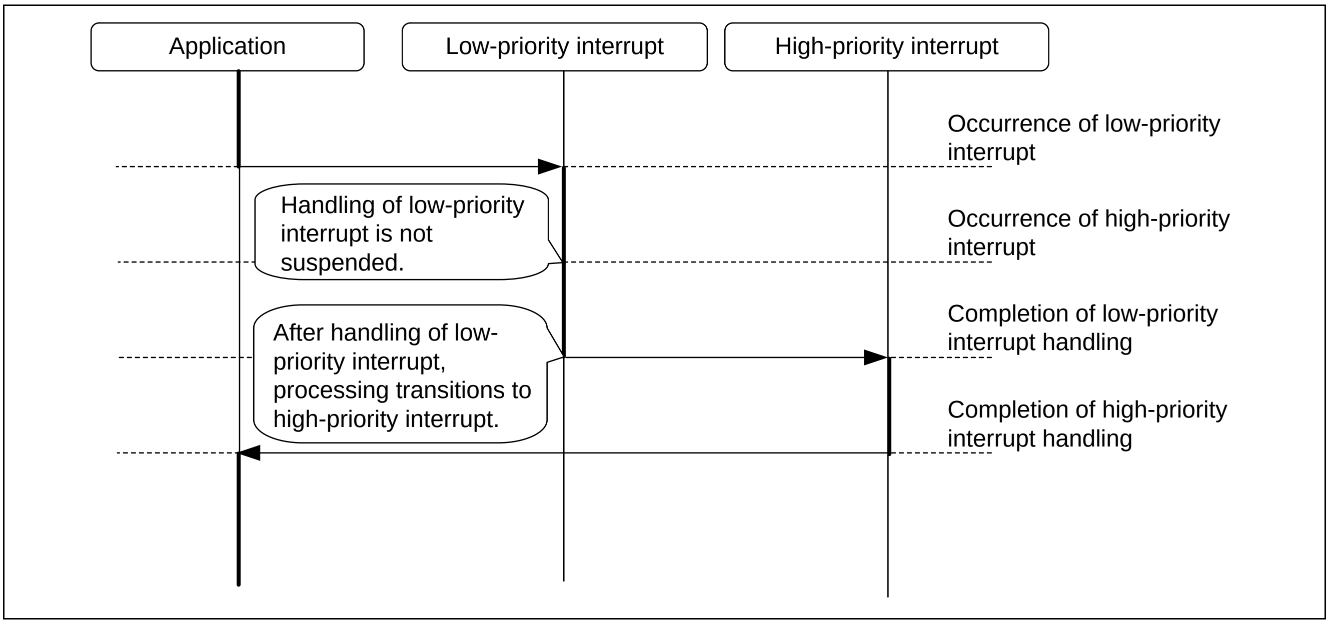

Application
Low-priority interrupt
High-priority interrupt
Occurrence of low-priority interrupt
Handling of low-priority interrupt is not suspended.
Occurrence of high-priority interrupt
After handling of low-priority interrupt, processing transitions to high-priority interrupt.
Completion of low-priority interrupt handling
Completion of high-priority interrupt handling

Figure 1.19 RX651 Interrupt Sequence (Not Controlled by PSW.I Bit)

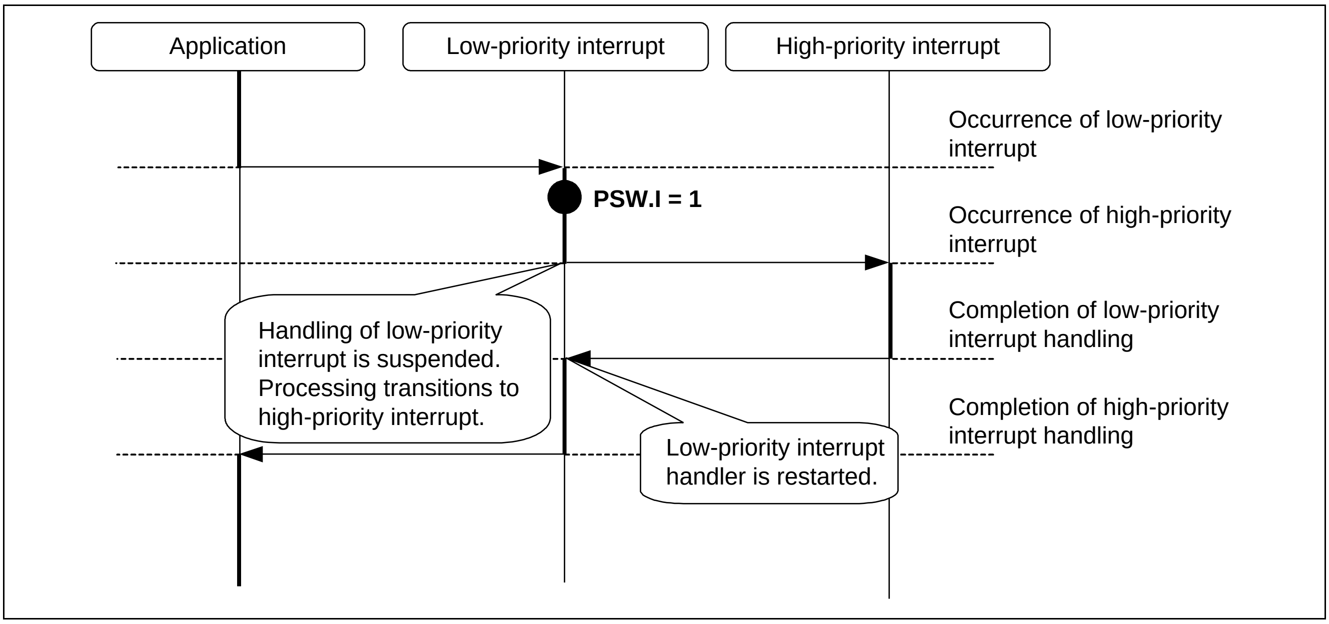

Application
Low-priority interrupt
High-priority interrupt
Occurrence of low-priority interrupt
PSW.I = 1
Occurrence of high-priority interrupt
Handling of low-priority interrupt is suspended. Processing transitions to high-priority interrupt.
Completion of low-priority interrupt handling
Low-priority interrupt handler is restarted.
Completion of high-priority interrupt handling

Figure 1.20 RX651 Interrupt Sequence (Controlled by PSW.I Bit)

1.8.6 Sselect interrupt

Among the RX651 interrupts, the interrupt factor of the peripheral module (Example: Some of the MTU and TPU interrupt factors) is allocated to the same vector as shown in Figure 1.21 selection interrupts. When using the unit selection function, it is necessary to select the interrupt source by means of a selector (register).

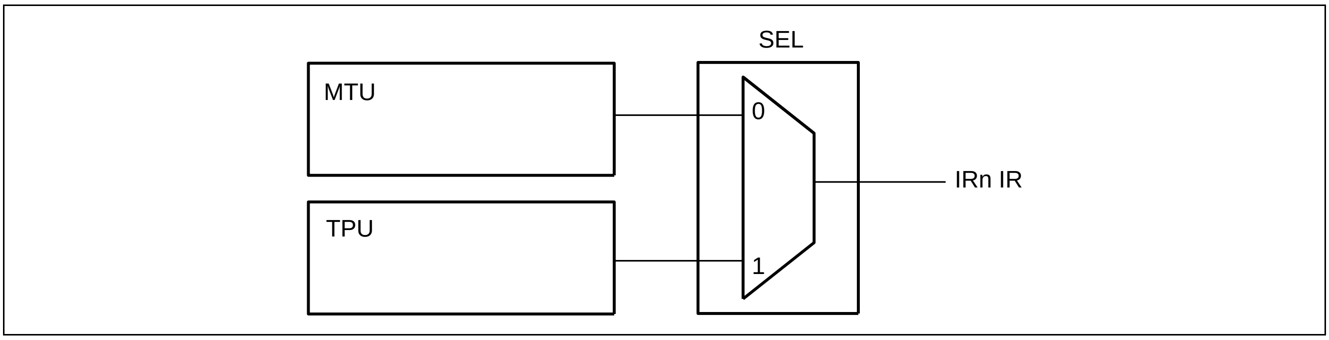


Figure 1.21 Unit Selection Function

1.8.7 Group Interrupts

Group interrupts allow multiple interrupt sources to be assigned to a single vector. Group interrupt detection is by means of a logical OR operation on all the interrupt requests assigned to the group. This means that when an interrupt request is detected, it is necessary to identify the interrupt request from among those in the group by means of software.

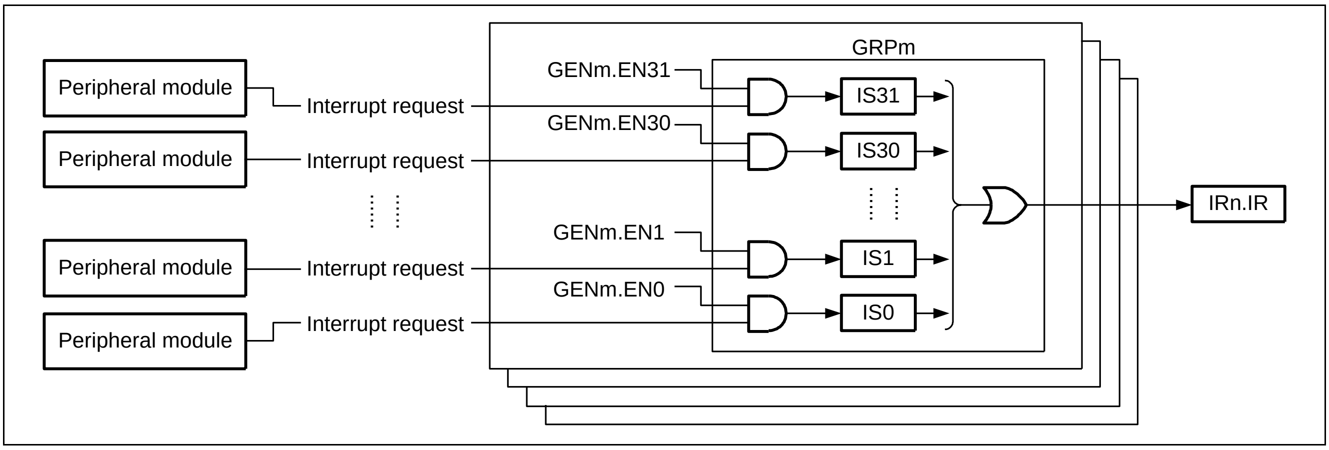


Figure 1.22 Group Interrupts

2. On-Chip Functions

2.1 List of On-Chip Functions

For more information about the peripheral features that are available only in RX651, see the User’s Manual: Hardware.

Table 2.1 List of Peripheral Functions

SH7044	RX651
Clock oscillator (CPG)	Clock generation circuit
User break controller (UBC)	—– Can be supported by the debugger function of the emulator
Data transfer controller (DTC)	Data transfer controller (DTCb)
Bus state controller (BSC)	Bus controller (BSC)
Direct memory access controller (DMAC)	DMA controller (DMACA) EXDMA controller (EXDMACa)
Multifunction timer pulse unit (MTU)	Multifunction timer pulse unit （MTU3a）
Watchdog timer (WDT)	Watchdog timer (WDTA) Independent watchdog timer (IWDTa)
Serial communication interface (SCI)	Serial communication interfaces（SCIg,SCIh, SCIi）
High-speed A/D converter (other than A mask) Mid-speed A/D converter (A mask)	12-bit A/D converter (S12ADFa)
Compare match timer (CMT)	Compare match timer (CMT) Compare match timer W（CMTW）
Pin function controller (PFC)	Multi-function pin controller (MPC)
I/O ports (I/O)	I/O port
Flash memory (256 KB)*1	Flash memory*2
RAM (4 KB)	RAM (maximum 256 KB)
Low power consumption function	Low power consumption function
Cache memory (CAC)	—
64/128 / 256kB mask ROM	It can be supported by the on-chip flash memory of RX
128kB PROM（ZTAT）	It can be supported by the on-chip flash memory of RX
256kB flash memory (F-ZTAT)	It can be supported by the on-chip flash memory of RX
—	Voltage detection circuit (LVDA)
	Clock frequency system measurement circuit (CAC)
	Battery backup function
	Register write protection function
	Memory protection unit (MPU)
	Event Link Controller (ELC)
	Port output enable 3（POE3a）
	16-bit timer pulse unit (TPUa)
	Programmable pulse generator (PPG)
	8-bit timer (TMR)
	Realtime clock （RTCd）
	Ethernet controller (ETHERC)
	Ethernet controller direct memory access controller (EDMAC)
	USB2.0FS Host/Function module (USBb)
	I^{2}C bus interface (RIICa)
	CAN module (CAN)
	Serial peripheral interface (RSPIc)

	Quad Serial Peripheral Interface (QSPI)
	CRC calculator (CRCa)
	SD Host Interface (SDHI)
	SD Slave Interface (SDSI)
	Multimedia Card Interface (MMCIF)
	Parallel data capture unit (PDC)
	Graphic LCD controller (GLCDC)
	2D drawing engine (DRW2D)
	Boundary scan
	AESa
	RNG
	Trusted Secure IP (TSIP)
	D / A converter (R12DAa)
	Temperature sensor (TEMPS)
	Data calculation circuit (DOC)
	Standby RAM

Notes: 1. Some versions of the SH7044 have on-chip mask ROM.

2. The RX651 group has up to 2 KB of on-chip flash memory (ROM) for storing code and up to 32 KB of on-chip flash memory for storing data (Data flash). For details, see the User’s Manual: Hardware.

2.2 I/O Ports

2.2.1 Number of I/O Ports

Table 2.2 Number of I/O Ports on SH7044 and RX651

Item	Package	Port Function
Number of I/O ports on SH7044	QFP-112	I/O: 74 Input: 8 Total: 82
Number of I/O ports on RX651	TFLGA-177 LFBGA -176 LQFP -176	I/O: 136 Input: 1 Pull-up resistor: 136 Open-drain output: 136 5 V tolerant: 19
	TFLGA-145 LQFP-144	I/O: 111 Input: 1 Pull-up resistor: 111 Open-drain output: 111 5 V tolerant: 18
	TFLGA-100 LQFP-100	I/O: 78 Input: 1 Pull-up resistor: 78 Open-drain output: 78 5 V tolerant: 17
	TFLGA-64	I/O: 41 Input: 1 Pull-up resistor: 41 Open-drain output: 41 5 V tolerant: 8
	LQFP-48	I/O: 42 Input: 1 Pull-up resistor: 42 Open-drain output: 42 5 V tolerant: 8

2.2.2 I/O Settings

Both the SH7044 and RX651 have multiplexed pins. Therefore, it is necessary to make pin settings to assign each pin to either general I/O or an on-chip module function.

On the SH7044 port functions are determined by settings made to the pin function controller (PFC). The I/O ports range from A to F, and with the exception of port F, which is input-only, each port can be assigned to either general I/O or an on-chip module function. Ports A to E are assigned to either general I/O or an on-chip module function the making settings in registers PnIOR and PnCR (n: port A to E). The general concept of I/O settings on the SH7044 and the functions of the various registers are described below.

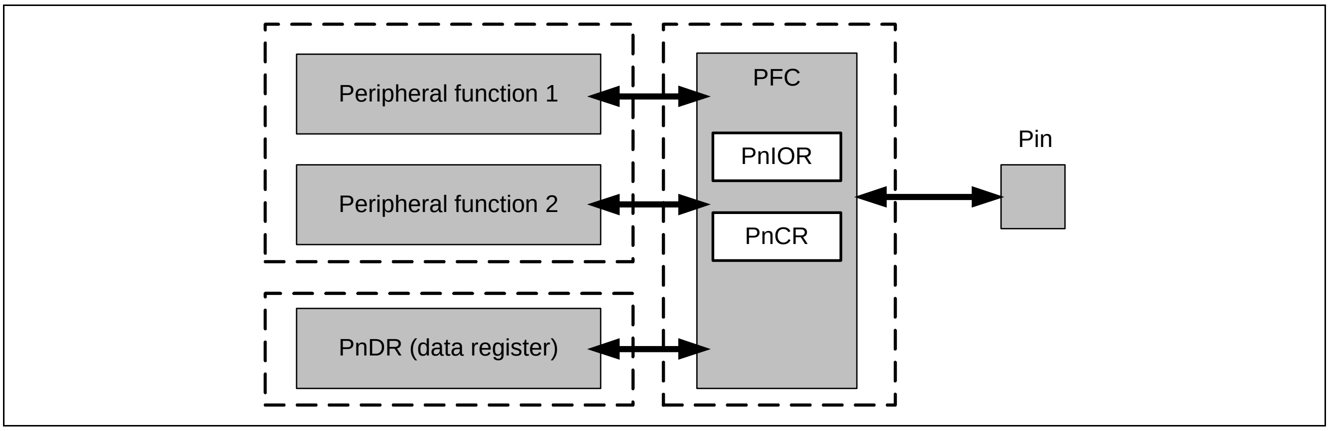


Figure 2.1 SH7044 I/O Settings

Table 2.3 Register Configuration on SH7044 for I/O Ports and Pin Function Controller

Module	Name	Function Name	Function
I/O port	PnDR	Port n data register	Port n data register
PFC	PnIOR	Port n IO register	Selects the port n I/O direction.
	PnCR	Port n control register	Selects the pin function.
	IFCR	IRQ function control register	Specifies the IRQ output pin state.

Note that the functions that can be assigned to pins and the functions that can be specified by the PFC differ according to the SH7044’s operation mode (microcontroller mode 0, 1, or 2, or single-chip mode).

The RX651 is provided with I/O ports 0 to 9, A to G, and J, and the configuration of the registers corresponding to these I/O ports is shown below. The port I/O registers include dedicated input and dedicated output registers.

The following types of I/O port settings are supported on the RX651.

- Open drain control register: Port output format selection
  CMOS output, N-channel open-drain output, or P-channel open-drain output
- Pull-up control register: Input pull-up resistor on/off selection
- Drive capacity control register: Selection between normal drive output and high drive output
- 5 V tolerant input ports are provided.

As on the SH7044, the pins are multiplexed, so it is necessary to make pin function settings in the I/O port module and the multi-function pin controller (MPC).

The I/O settings for RX651 are shown in Figure 2.2: RX651 I/O settings.

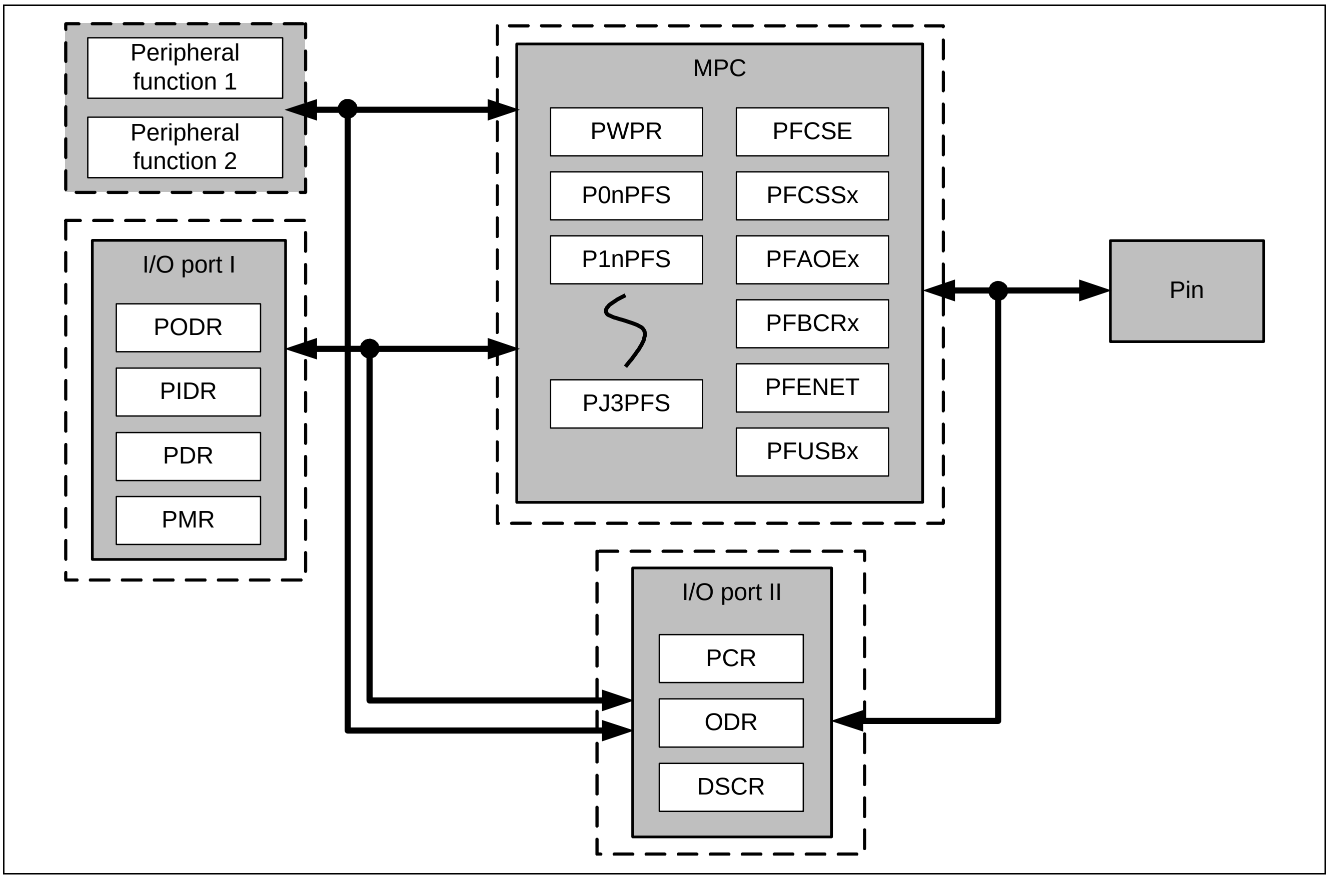


Figure 2.2 I/O Settings on the RX651

To use a pin for general I/O, it is sufficient to make settings in the I/O port registers (settings in PMR, PDR, ODR, PCR, and DSCR). Table 2.4 lists the registers in which the settings are made. Figure 2.3 is a flowchart of the setting procedure.

To use a pin for a peripheral function, the pin must be assigned to the peripheral function in the pin function control register (PxnPFS) in the MPC. Tables 2.4 and 2.5 list the registers in which the settings are made. Figure 2.4 is a flowchart of the setting procedure.

For example settings for use with peripheral functions that include general I/O, see the section describing the specific peripheral function.

Table 2.4 RX651 I/O Port Register Configuration

Register	Function Name	Function
PDR	Port direction register	Specifies input or output for pins selected as general I/O ports.
PODR	Port output register	Stores pin output data for general output ports.
PIDR	Port input register	Reflects pin states for general input ports.
PMR	Port mode register	Used for port pin function settings. Specifies whether each pin is used as a general I/O port or for a peripheral function.
ODR0	Open drain control register 0	Selects the port output format from among the following: • CMOS output • N-channel open drain • P-channel open drain
ODR1	Open drain control register 1	Selects the port output format from among the following: • CMOS output • N-channel open drain
PCR	Pull-up control register	Turns the port input pull-up resistor on or off.
DSCR	Drive capacity control register	Specifies the drive capacity. • Normal drive output • High drive output
DSCR2	Drive capacity control register	Specifies the drive capacity. • Normal / high drive output • High drive output for high-speed interface

Table 2.5 RX651 Multi-Function Pin Controller Registers

Register	Function Name	Function
PWPR	Write-protect register	Write-protect function for PxxPFS register xx: 0n to 9n, An to Gn, J3
P0nPFS	P0n pin function control register	Register for selecting the pin function (port 0 pin function selection)
P1nPFS	P1n pin function control register	Register for selecting the pin function (port 1 pin function selection)
P2nPFS	P2n pin function control register	Register for selecting the pin function (port 2 pin function selection)
PFnPFS	PFn pin function control register	Register for selecting the pin function (port F pin function selection)
PJnPFS	PJnpin function control register	Register for selecting the pin function (port J pin function selection)
PFCSE	CS output enable register	Disables or enables output on CSn# (n: 0 to 7).
PFCSS0	CS output pin select register 0	Selects output pins for CS0 to CS3.
PFCSS1	CS output pin select register 1	Selects output pins for CS4 to CS7.
PFAOE0	Address output enable register 0	Settings when using pins for address bus
PFAOE1	Address output enable register 1	Settings when using pins for address bus
PFBCR0	External bus control register 0	Settings when using pins for external bus
PFBCR1	External bus control register 1	Settings when using pins for external bus
PFBCR2	External bus control register 2	Settings when using pins for external bus
PFBCR3	External bus control register 3	Settings when using pins for external bus
PFENET	Ethernet control register	Ethernet mode setting (PMII or MII)

The initialization sequence when using RX651 I/O ports for general I/O is shown below.

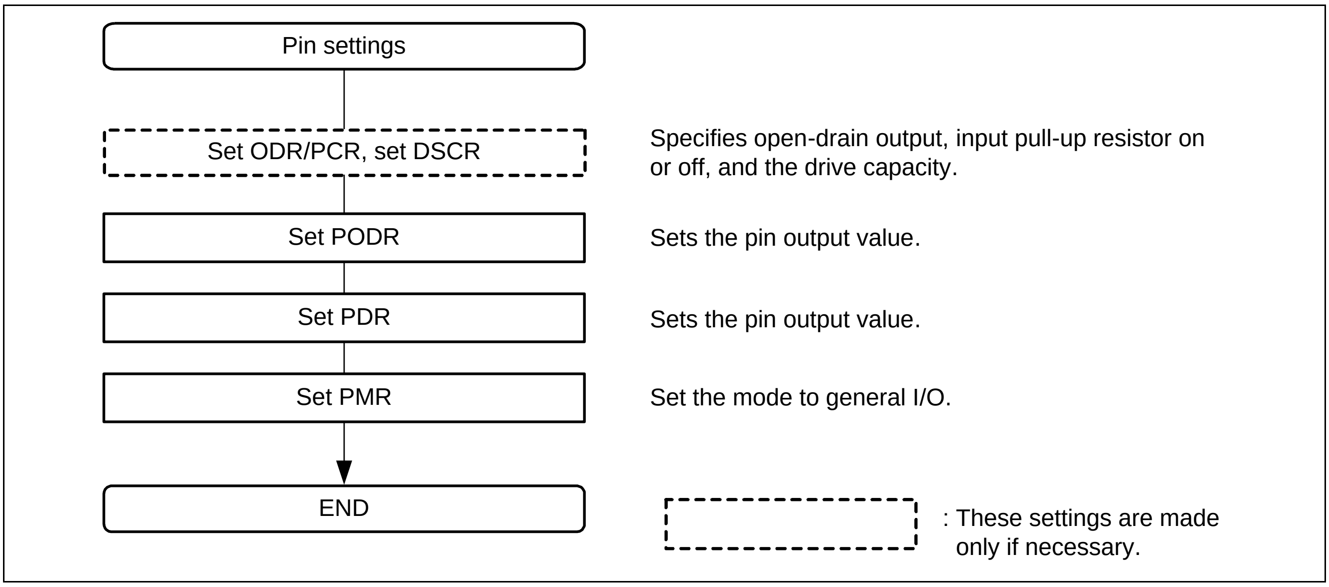
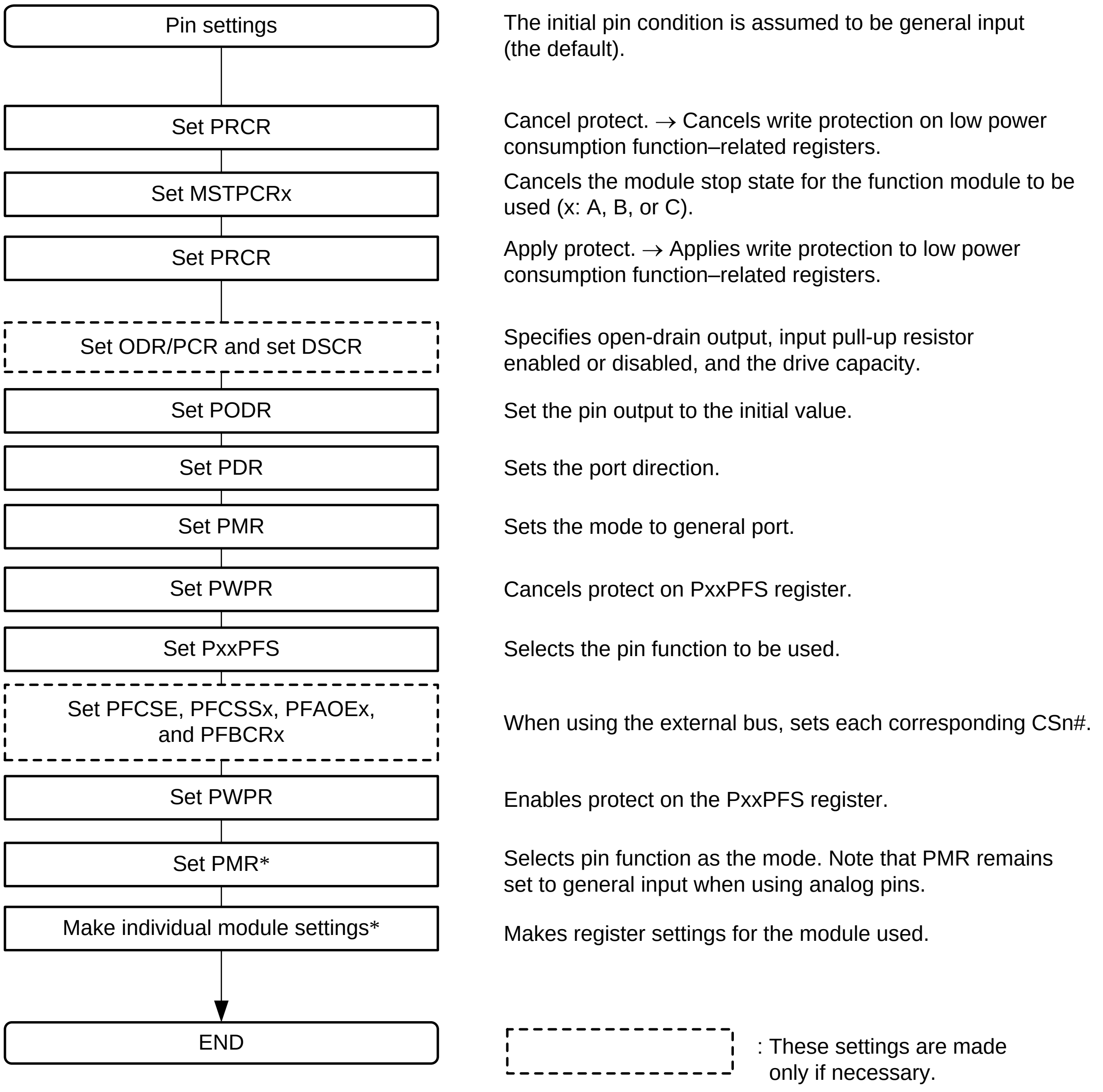

Pin settings
Set ODR/PCR, set DSCR
Specifies open-drain output, input pull-up resistor on or off, and the drive capacity.
Set PODR
Sets the pin output value.
Set PDR
Sets the pin output value.
Set PMR
Set the mode to general I/O.
END
: These settings are made only if necessary.

Figure 2.3 Using RX651 I/O Ports for General I/O

The initialization sequence when assigning pin functions to RX651 I/O ports is shown below.


Pin settings
The initial pin condition is assumed to be general input (the default).
Set PRCR
Cancel protect. → Cancels write protection on low power consumption function–related registers.
Set MSTPCRx
Cancels the module stop state for the function module to be used (x: A, B, or C).
Set PRCR
Apply protect. → Applies write protection to low power consumption function–related registers.
Set ODR/PCR and set DSCR
Specifies open-drain output, input pull-up resistor enabled or disabled, and the drive capacity.
Set PODR
Set the pin output to the initial value.
Set PDR
Sets the port direction.
Set PMR
Sets the mode to general port.
Set PWPR
Cancels protect on PxxPFS register.
Set PxxPFS
Selects the pin function to be used.
Set PFCSE, PFCSSx, PFAOEx, and PFBCRx
When using the external bus, sets each corresponding CSn#.
Set PWPR
Enables protect on the PxxPFS register.
Set PMR*
Selects pin function as the mode. Note that PMR remains set to general input when using analog pins.
Make individual module settings*
Makes register settings for the module used.
END
: These settings are made only if necessary.

Note: * The order of PMR settings and module settings differs according to the module.

Figure 2.4 Assigning Pin Functions to RX651 I/O Ports

Note: For details on the MPC settings used to assign functions to pins, see the section describing the specific peripheral function.

On the RX651 the individual modules are in the stopped state*[1] by default. Therefore, it is necessary to cancel module stop with the module stop control register (MSTPCRx) of the low power consumption function before making peripheral function settings. In addition, write protection has been applied to MSTPCRx by the register write protection function. Thus, to overwrite MSTPCRx it is necessary to first make it writeable by using the protect register (PRCR).

Note: 1. The DMAC, DTC, and RAM are in the operable state by default.

2.2.3 General I/O

General I/O port setting examples for the SH7044 and RX651 are shown below.

An example of using PB2 for SH7044 and P34 for RX651 as general-purpose inputs is shown in "Table 2.6 Settings for general-purpose input".

Table 2.6 Pin Settings for General Input

Procedure		SH7044 Setting Example	RX651 Setting Example
1	Set the pin I/O direction to input.	PBIOR.PB2IOR = 0	PORT3.PDR.B4 = 0

2	Set general pins as general ports.	PBCR2.PB2MD1 = 0
		PBCR2.PB2MD0 = 0
			PORT3.PMR.B4 = 0

An example of using PB2 for SH7044 and P34 for RX651 as general-purpose inputs is shown in "Table 2.7 Settings for general-purpose input". The output value is 1.

Table 2.7 Pin Settings for General Output

Procedure		SH7044 Setting Example	RX651 Setting Example
1	Set the pin to output.	PBDR.PB2DR = 1	PORT3.PODR.B4 = 1
2	Set the pin I/O direction to output.	PBIOR.PB2IOR = 1	PORT3.PDR.B4 = 1

3	Set pins as general ports.	PBCR2.PB2MD1 = 0
		PBCR2.PB2MD0 = 0
			PORT3.PMR.B4 = 0

2.3 Buses

This section describes the points of difference between the bus specifications of the two microcontrollers.

2.3.1 Comparison of Specifications

The main differences between the buses of the SH7044 and RX651 are shown below.

Table 2.8 SH7044 and RX651 Bus Comparison

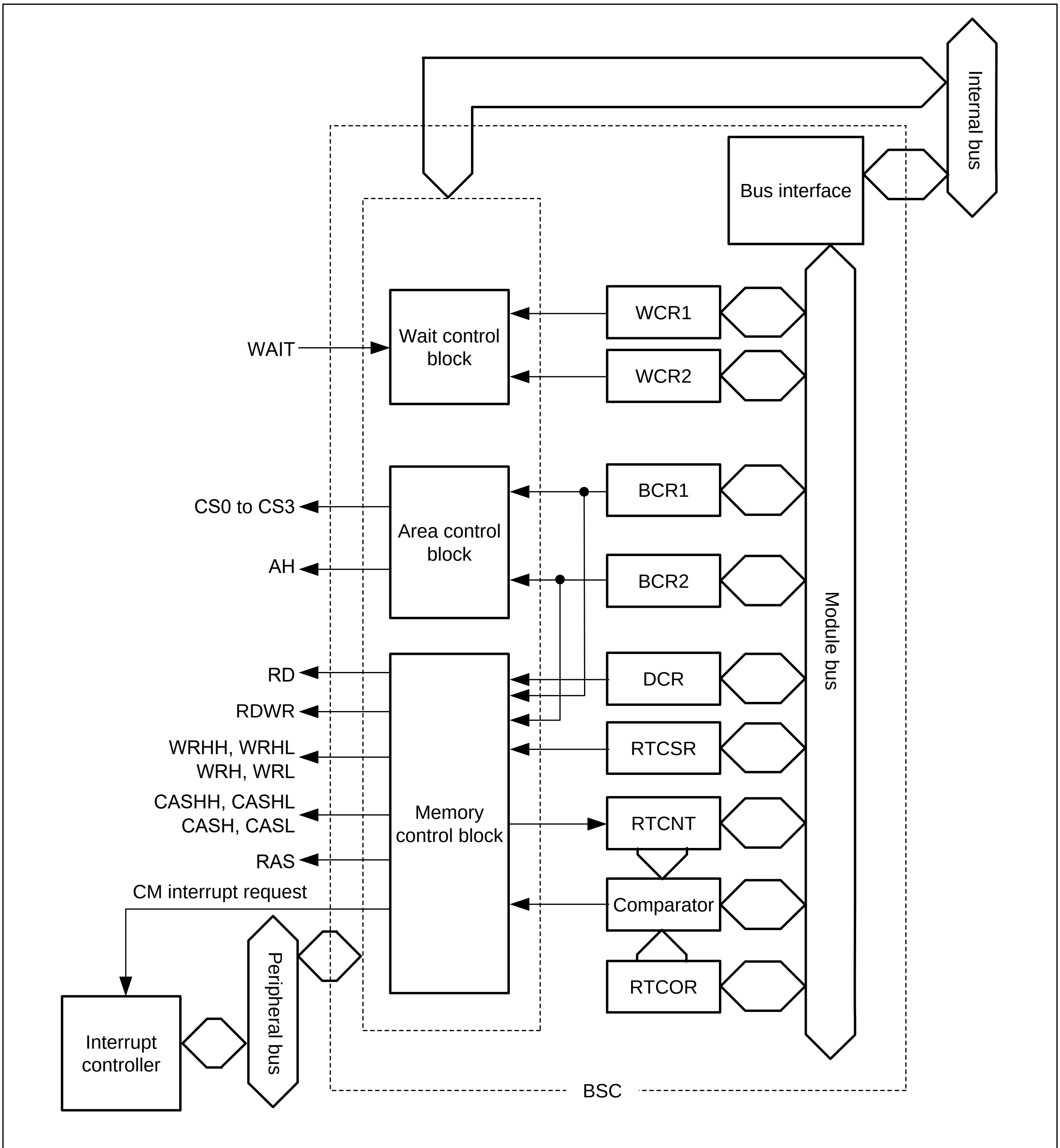
Item	SH7044	RX651
External bus address space	• External address spaces CS0 to CS3 (4 MB each) Notes: 1. CS0 is 2 MB when on-chip ROM is enabled. 2. 4 MB in on-chip ROM disabled mode.	• External address spaces CS0 to CS7 (16 MB)
DRAM/SDRAM dedicated space	DRAM space (maximum 16 MB)	SDRAM space (maximum 128 MB)
Bus width	Settable to 8 or 16 bits by area.	Settable to 8, 16, or 32 bits by area.
Endianness	Big-endian (fixed)	The endianness can be set independently for each area.*
Bus arbitration	• CPU bus and external bus have fixed priority.	• External bus: Priority selectable from the following: 1) fixed priority, 2) toggle priority • Internal bus: Priority selectable from the following: 1) fixed priority, 2) toggle priority
SRDAM refresh	• CAS Bifo RAS refresh and self-refresh support	• Auto fresh and self-refresh support
Interrupt request occurrence	• The refresh counter can be used as an interval timer	• Can be supported by timers such as MTU3
External bus arbitration	• Possible	• Not possible
Other access control	• Output of _RAS and _CAS signals for DRAM • Ability to generate a RAS precharge time assurance Tp cycle • DRAM burst access function • Ability to specify the DRAM refresh interval • Ability to insert wait cycles using an external _WAIT signal • Ability to access address data multiplexed I/O devices	CS area • Ability to insert recovery cycles • Cycle wait function • CSn# signal timing setting • RD# and WR# signal timing control • Write access mode • Ability to access address data multiplexed I/O devices SDRAM area • Multiplexed output of row and column addresses • CAS latency setting Write buffer • Write buffer function

Note: * See 1.2.2.

2.3.2 Bus Configuration

The bus configurations of the SH7044 and RX651 are compared below.

The configuration of the SH7044's bus state controller is shown below.

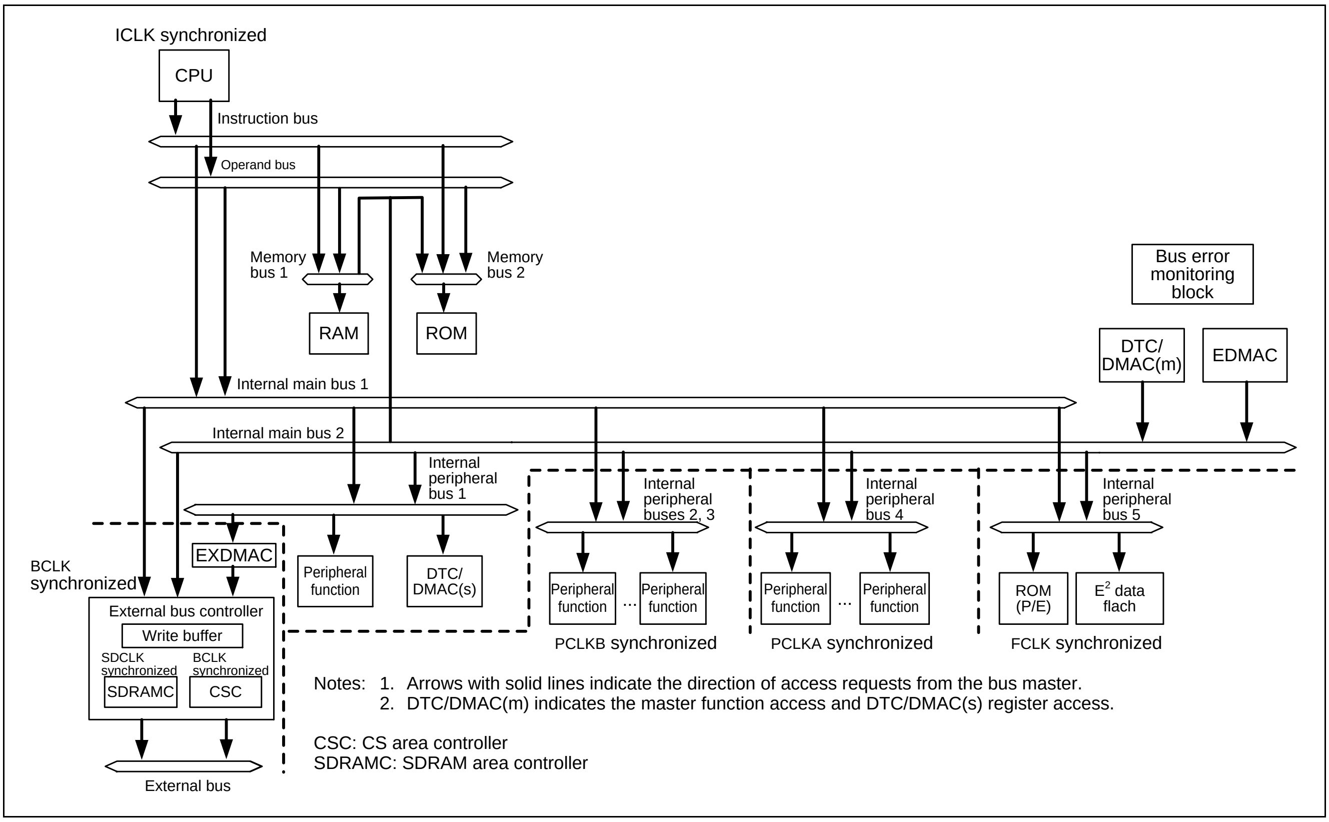


WCR1: Wait control register 1
WCR2: Wait control register 2
BCR1: Bus control register 1
BCR2: Bus control register 2

DCR: DRAM area control register
RTCSR: Refresh timer control/status register
RTCNT: Refresh timer counter
RTCOR: Refresh time constant register

Figure 2.5 SH7044 Bus State Controller Configuration

The bus configurations of the RX651 is shown below.



Notes: 1. Arrows with solid lines indicate the direction of access requests from the bus master.
2. DTC/DMAC(m) indicates the master function access and DTC/DMAC(s) register access.

CSC: CS area controller
SDRAMC: SDRAM area controller

Figure 2.6 RX651 Bus Configuration

The bus types on the RX651 are listed below. The RX651 has a different bus architecture than the SH7044, and the memory buses, internal buses, and peripheral buses each have multiple stages. This enables parallel operation by the CPU and DMAC or DTC, and between the modules on the peripheral buses, thereby speeding up operation overall.

Table 2.9 RX651 Buses

Bus	Connected modules, etc.	Clock
CPU buses (instruction bus and operand bus)	Instruction bus: CPU, on-chip Memory Operand bus: CPU, on-chip Memory	ICLK
Memory bus 1	On-chip RAM	ICLK
Memory bus 2	Code flash memory	ICLK
Internal main bus 1	CPU	ICLK
Internal main bus 2	DTC, DMAC, Extended bus master	ICLK
Internal peripheral bus 1	DTC, DMAC, EXDMAC, interrupt controller, bus error monitoring block	ICLK (EXDMA: PCLKB)
Internal peripheral bus 2	Peripheral functions (peripheral functions other than those connected to peripheral buses 1, 3, 4, and 5)	PCLKB
Internal peripheral bus 3	USBb, Standby RAM	PLCKB
Internal peripheral bus 4	EDMAC, ETHERC, MTU3, SCIi, RSPI, AES	PLCKA
Internal peripheral bus 5	GLDC, DRW	PLCKA
Internal peripheral bus 6	Code flash memory, Data flash memory	FCLK
External buses (CS areas)	External devices	BCLK
External buses (SDRAM)	SDRAM	SDCLK

ICLK: System clock PCLKA: Peripheral clock A PCLKB: Peripheral clock B
FCLK: FlashIF clock BCLK: External bus clock SDCLK: SDRAM clock

2.4 Interrupt Controller（ICUB）

2.4.1 IRQ Usage Example

A setting example using IRQ3 is shown below. PB5 is used as the IRQ3 input pin on the SH7044. P33 is used as the IRQ3 input pin on the RX651.

Table 2.10 Interrupt Initial Setting Example (IRQ3 Settings)

Procedure		SH7044	RX651

1	Make I/O port settings.	PBIOR.PB5IOR = 0
		(general input pin setting)
		PBCR2.PB5MD1, PBCR2.PB5MD0 = 01b
		(IRQ3 interrupt input pin)
			PORT3.PDR.B3 = 0
			(P33 input setting)
			PORT3.PMR.B3 = 0
			(P33 GPIO setting)
			MPC.PWPR.B0WI = 0
			MPC.PWPR.PFSWE = 1
			(PFS write enabled)
			MPC.P33PFS.ISEL = 1
			(interrupt function setting IRQ3-DS)
			MPC.PWPR.PFSWE = 0
			(PFS write disabled)
			MPC.PWPR.B0WI = 1

2	Make interrupt controller settings.	ICR.IRQ3S = 1
		(IRQ detection: Falling edge)
		IPRA = 0x000F
		(bits 3 to 0: interrupt level 15)
			IRQCR3.IRQMD = 1
			(IRQ detection: Falling edge)
			IRQFLTE0.FLTEN3 = 1
			(IRQ3 digital noise filter enabled)
			IRQFLTC0.FCLKSEL3 = 3;
			(sampling PCLK/64)
			IR067 = 0 (interrupt flag cleared)
			IER08.IEN3 = 1 (IRQ3 enabled)
			IPR067 = 15 (interrupt level 15)

2.5 Data Transfer Controller (DTCb)

2.5.1 Comparison of Specifications

On both the SH7044 and RX651 the transfer information is located in RAM, and DTC vectors are used to specify transfer information. The basic operation of the three transfer modes (normal transfer mode, repeat transfer mode, and block transfer mode) is the same on both microcontrollers. The DTC specifications of the SH7044 and RX651 are listed below.

Table 2.11 Comparison of DLC Specifications on SH7044 and RX651

Item	SH7044	RX651
Transfer modes	• Normal transfer mode • Repeat transfer mode • Block transfer mode	
Activation sources	• • External interrupt • Peripheral function interrupt • Software trigger	
Activation enable/disable control	Activated by DTC activation enable register of interrupt controller.	
Transfer spaces	Transfer between the following spaces is possible: • On-chip memory space • On-chip peripheral module space (excluding DMAC and DTC) • External memory space • Memory-mapped external device space Note: One of the specified areas must be in the on-chip memory space or on-chip peripheral module space.	
Transfer units	May be specified as 8, 16, or 32 bits. Block size: May be specified within range of 0 to 65,535.	1 data unit: May be specified as 8, 16, or 32 bits. 1 block: May be specified within range of 1 to 256 data units.
Number of transfers	• Normal transfer mode : 1 to 65536 times • Repeat transfer mode : 1 to 256 times (repeat after the specified number of times) • Block transfer mode : 1 to 65536 times	
CPU interrupt requests	• An interrupt generated by a CPU interrupt request may be used as the DTC activation source. • A CPU interrupt at single data unit transfer-end may be used. • A CPU interrupt after transfer of a specified number of data units may be used.	
Method	Control information is allocated for each interrupt source by using DTC vectors.	
Other	Chain transfer	• Chain transfer • Sequence transfer • The following functions can be used to shorten the transfer duration and reduce memory usage: • Transfer information read skipping • Write-back skipping • Short-address mode • Event link • Light back disabled • Displacement addition

2.5.2 Register Configuration

The register configuration of the DTC is shown below.

Table 2.12 List of DTC Registers on SH7044 and RX651

Item		SH7044	RX651
Transfer mode selection		DTC mode register (DTMR) DTC mode 1, 0 (MD1 or MD0)	DTC mode register A (MRA) DTC transfer mode select bits
Selection of repeat area or block area as transfer destination or transfer source		DTC mode register (DTMR) DTC transfer mode select (DTS)	DTC mode register B (MRB) DTC transfer mode select bits
Data transfer size selection		DTC mode register (DTMR) DTC data transfer size 1 or 0 (SZ1 or SZ0)	DTC mode register A (MRA) DTC data transfer size bits
Transfer source: Address state after transfer		DTC mode register (DTMR) Source address mode 1 or 0 (SM1 or SM0)	DTC mode register A (MRA) Transfer source address addressing mode bits
Transfer destination: Address state after transfer		DTC mode register (DTMR) Destination address mode 1 or 0 (DM1 or DM0)	DTC mode register B (MRB) Transfer destination address addressing mode bits
Chain transfer selection	Transfer-end/ continue, enable/ disable	DTC mode register (DTMR) DTC chain enable (CHNE)	DTC mode register B (MRB) DTC chain transfer enable bit (CHNE)
	Continuous transfer/ transfer at change of transfer counter	—	DTC mode register B (MRB) DTC chain transfer select bit (CHNE)
Interrupt request enable/ disable		DTC mode register (DTMR) DTC interrupt select (DISEL)	DTC mode register B (MRB) DTC interrupt select bit (DISEL)
DTC transfer suspend/ resume by NMI		DTC mode register (DTMR) DTCNMI mode (NMIM)*1	—
Transfer source address		DTC source address register (DTSAR)	DTC transfer source register (SAR)
Transfer destination address		DTC destination address register (DTDAR)	DTC transfer destination register (DAR)
Initial address		DTC initial address register (DTIAR) *1*1*2	—
Transfer count specification		DTC transfer count register A (DTCRA) Specifies the transfer count.	DTC transfer count register A (CRA) Specifies the transfer count.
Block transfer mode	Data unit transfer count	DTC transfer count register A (DTCRA) Specifies the block transfer count.	DTC transfer count register B (CRB) Specifies the block transfer count.
	Block length specification	DTC transfer count register B (DTCRB) Specifies the block length.	DTC transfer count register A (CRA) Specifies the block length.
DTC activation disable/enable		DTC enable register (DTER) DTC enable bit	DTC activation enable register (ICU.DTCERn)
DTC module operate/stop		—	DTC module start register (DTCST) DTC module start bit
Base address		DTC information base register (DTBR)*2	DTC vector base register (DTCVBR)
Full address mode/ Short address mode		—	DTC address mode register (DTCADMOD)
NMI interrupt generation enable/disable		DTC control/status register (DTCSR) NMI flag bit (NMIF)	Non-maskable interrupt status register (ICU.NMISR) NMI status flag

Item	SH7044	RX651
DTC activation by software enable/disable	DTC control/status register (DTCSR) DTC software activation enable bit (SWDTE)	Software interrupt activation register (ICU.SWINTR) Software interrupt activation bit (SWINT)
DTC vector address setting for DTC activation by software	DTC control/status register (DTCSR) Software activation vectors 7 to 0 (DTVEC7 to DTVEC0)	DTC status register (DTCSTS) VECN[7:0] bits (DTC-activating vector number monitoring bits)
Showing of DTC transfer operation state	—	DTC status register (DTCSTS) DTC active flag
Read skipping enable	—	DTC Control Register (DTCCR) DTC transfer information read skipping enable bit
Possibility to add displacement values	—	DTC mode register C (MRC) Displacement addition bit
Base address setting for index placement	—	DTC index table base register (DTCIBR)
Sequence transfer End execution	—	DTC operation register (DTCOR) Sequence transfer end bit
Sequence transfer settings	—	DTC sequence transfer permission register (DTCSQE) Sequence transfer vector number specification bit Sequence transfer permission bit
Specifying a displacement value	—	DTC Address displacement register (DTCDISP)

Notes: 1. The initial address setting in SH is set in the transfer source (SAR) / transfer destination (DAR) register of RX.

2. The contents of the SH information base register are included in the contents of the address indicated by the RX DTC vector base register.

2.5.3 Transfer Modes

The differences in the operation of the transfer modes is described below.

Table 2.13 Normal Transfer Mode

Item	SH7044	RX651
Transfer size	1 byte, 1 word, or 1 longword	1 byte, 1 word, or 1 longword
Transfer count	1 to 65,536	1 to 65,536

Table 2.14 Repeat Transfer Mode (The method of specifying the repeat area differs.)

Item	SH7044	RX651
Transfer size	1 byte, 1 word, or 1 longword	1 byte, 1 word, or 1 longword
Transfer count	1 to 256	1 to 256
Repeat area specification method	The repeat mode and whether either the source or destination is the repeat area is specified in the mode register. The repeat address is specified in the repeat initial address register.	The concept of the repeat initial address does not apply, and the initial value of SAR or DAR is repeated.

Table 2.15 Block Transfer Mode (The way of conceptualizing the single block size differs.)

Item	SH7044	RX651
Transfer size	Transferring a single block	Transferring a single block
Single block size	1 to 65,536 bytes	1 to 256 data units The data unit can be byte, word, or longword.
Transfer count	1 to 65,536	1 to 65,536

2.5.4 Activation Source Setting

On the SH7044 activation sources of the DTC are set in the DTC enable registers (DTEA to DTEE). On the RX651 DTC activation sources are set in the DTC activation enable registers (DTCERn, n: vector number) of the interrupt controller, and this enables DTC activation by interrupts.

2.5.5 DTC Vector Configuration

The DTC vector configuration on the SH7044 and RX651 is shown below.

On the SH7044 the DTC vector table starts from the fixed address 400h. The upper 16 bits of the transfer information addresses are stored in the DTC information base register (DTBR), and the lower 16-bit address for each activation source is stored in the DTC vector table.

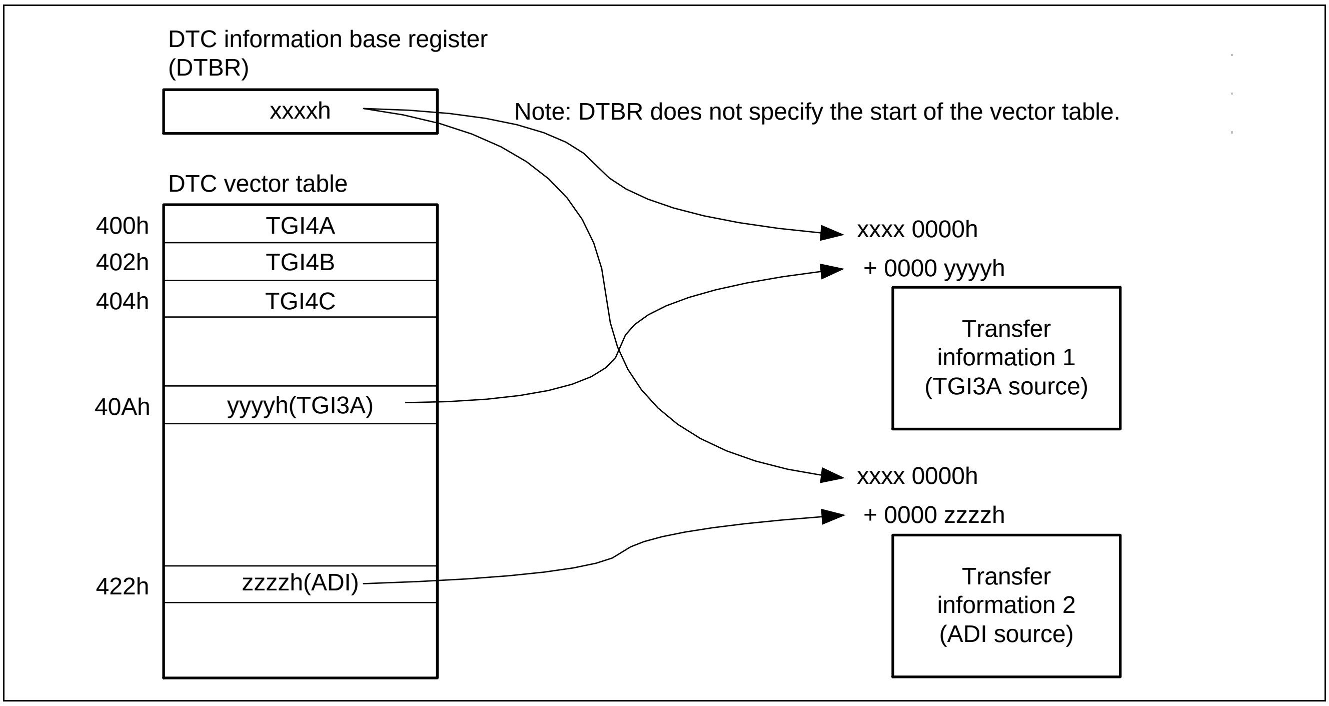

DTC information base register
(DTBR)
xxxxh
Note: DTBR does not specify the start of the vector table.
DTC vector table
400h
TGI4A
402h
TGI4B
404h
TGI4C
40Ah
yyyyh(TGI3A)
422h
zzzzh(ADI)
xxxx 0000h
+ 0000 yyyyh
Transfer information 1 (TGI3A source)
xxxx 0000h
+ 0000 zzzzh
Transfer information 2 (ADI source)

Figure 2.7 DTC Vector Configuration on SH7044

On the RX651 the vector table start address is specified by the DTC vector base register (DTVBR).. Vectors can be set in 1 KB units within the range from 0000 0000h to 7FF FC00h and from F800 0000h to FFFF FC00h. As with interrupt vectors, the vectors are numbered 0 to 255, and a 32-bit transfer information address can be specified for each vector. In contrast to the SH7044's DTC vector table, which starts from the fixed address 400h, on the RX651 the start address can be set in the DTC vector base register, so there is more flexibility in specifying the DTC vector table area.

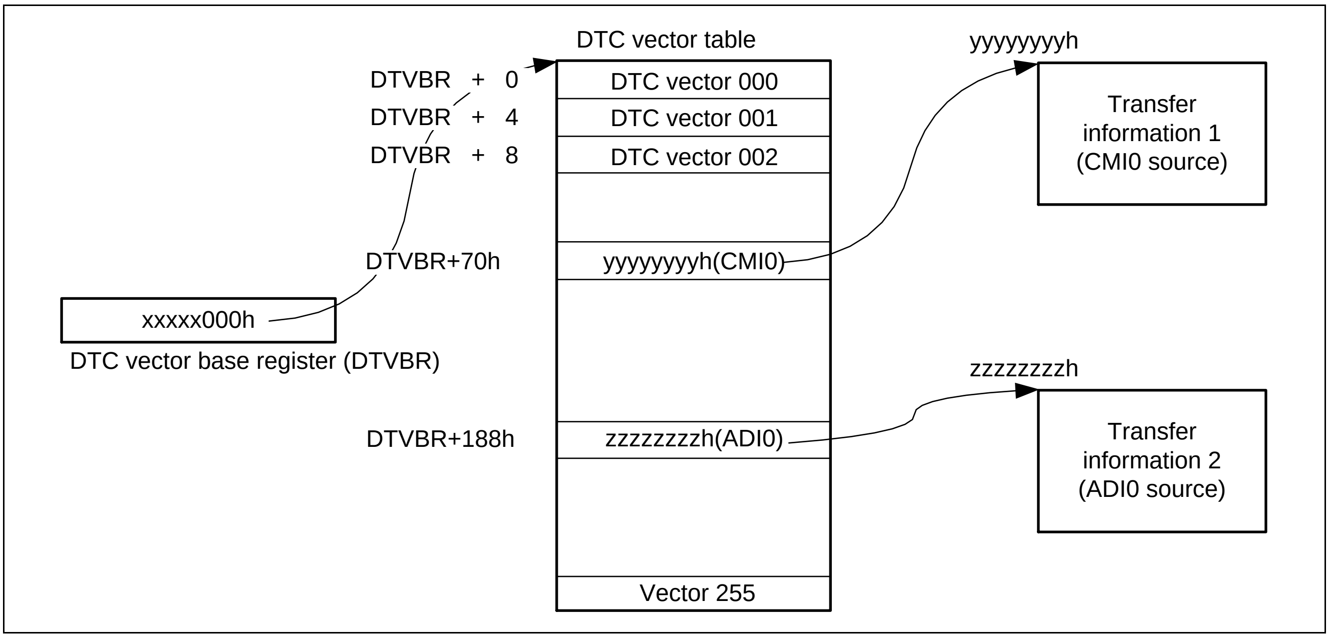
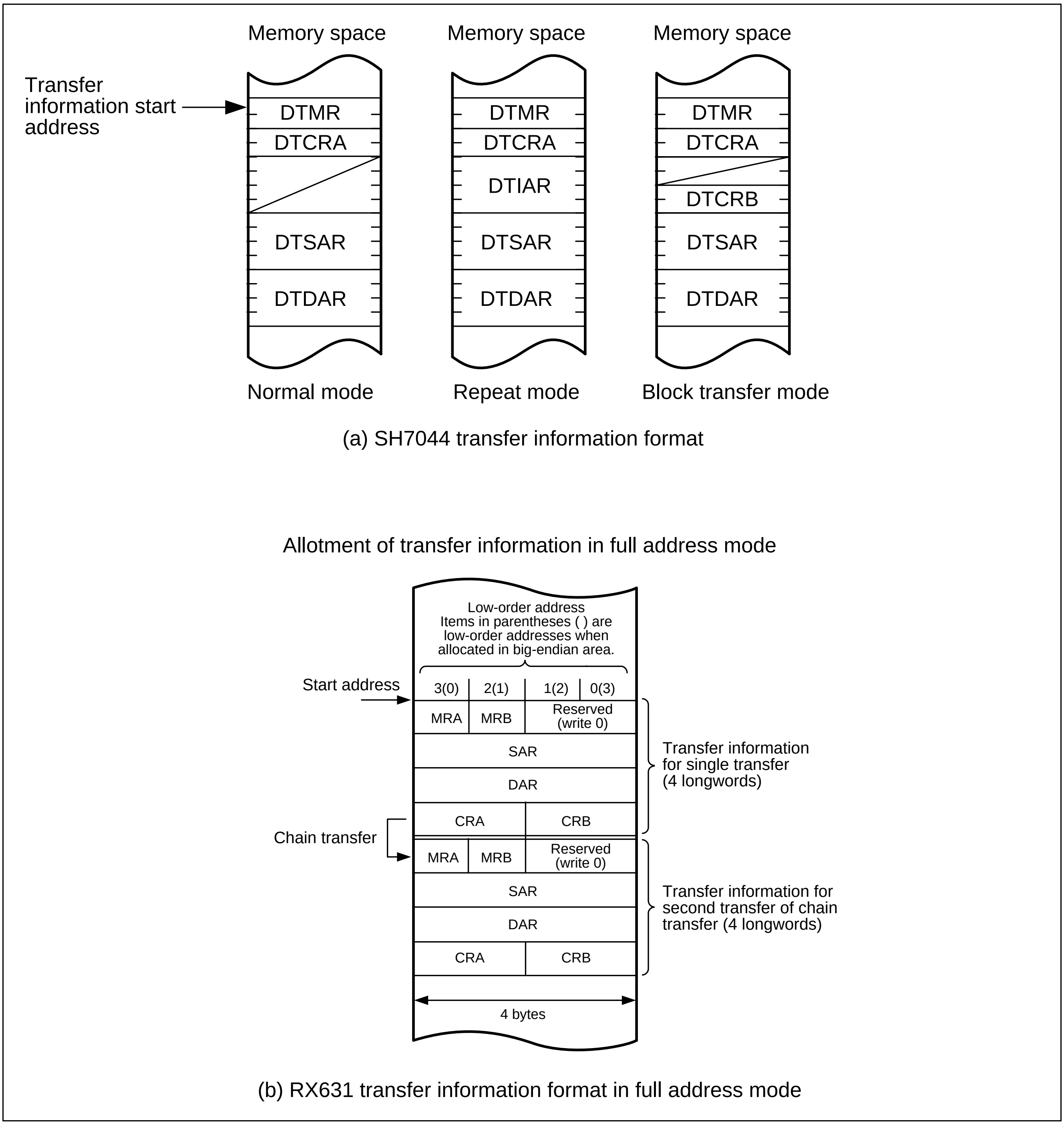


Figure 2.8 DTC Vector Configuration on RX651

2.5.6 Allocation of Transfer Information

The format of transfer information differs between the SH7044 and the RX651.

On the SH7044 a different transfer information format is used for each transfer mode. On the RX651 all transfer modes use the same transfer information format. Note, however, that on the RX651 the DTC transfer information is affected by the endianness setting. The transfer information format of each mode on the SH7044 (a) and the full-address mode transfer information format on the RX651 (b) are shown below.


Memory space
Memory space
Memory space
Transfer information start address
DTMR
DTCRA
DTSAR
DTDAR
DTMR
DTCRA
DTIAR
DTSAR
DTDAR
DTMR
DTCRA
DTCRB
DTSAR
DTDAR
Normal mode
Repeat mode
Block transfer mode
Allotment of transfer information in full address mode
Low-order address
Items in parentheses ( ) are low-order addresses when allocated in big-endian area.
Start address
3(0)
2(1)
1(2)
0(3)
MRA
MRB
Reserved (write 0)
SAR
DAR
CRA
CRB
Transfer information for single transfer (4 longwords)
Chain transfer
MRA
MRB
Reserved (write 0)
SAR
DAR
CRA
CRB
Transfer information for second transfer of chain transfer (4 longwords)
4 bytes

(a) SH7044 transfer information format

(b) RX631 transfer information format in full address mode

Figure 2.9 DTC Transfer Information Formats on SH7044 and RX651

The RX651 supports a short-address mode in which addresses can be specified in 24 bits. The size of the transfer information is four longwords in full-address mode but only three longwords in short-address mode. This reduces the time it takes the DTC to read transfer information and enables it to start up faster. In addition, less RAM is needed to store the transfer information. The transfer information format in short-address mode is shown below.

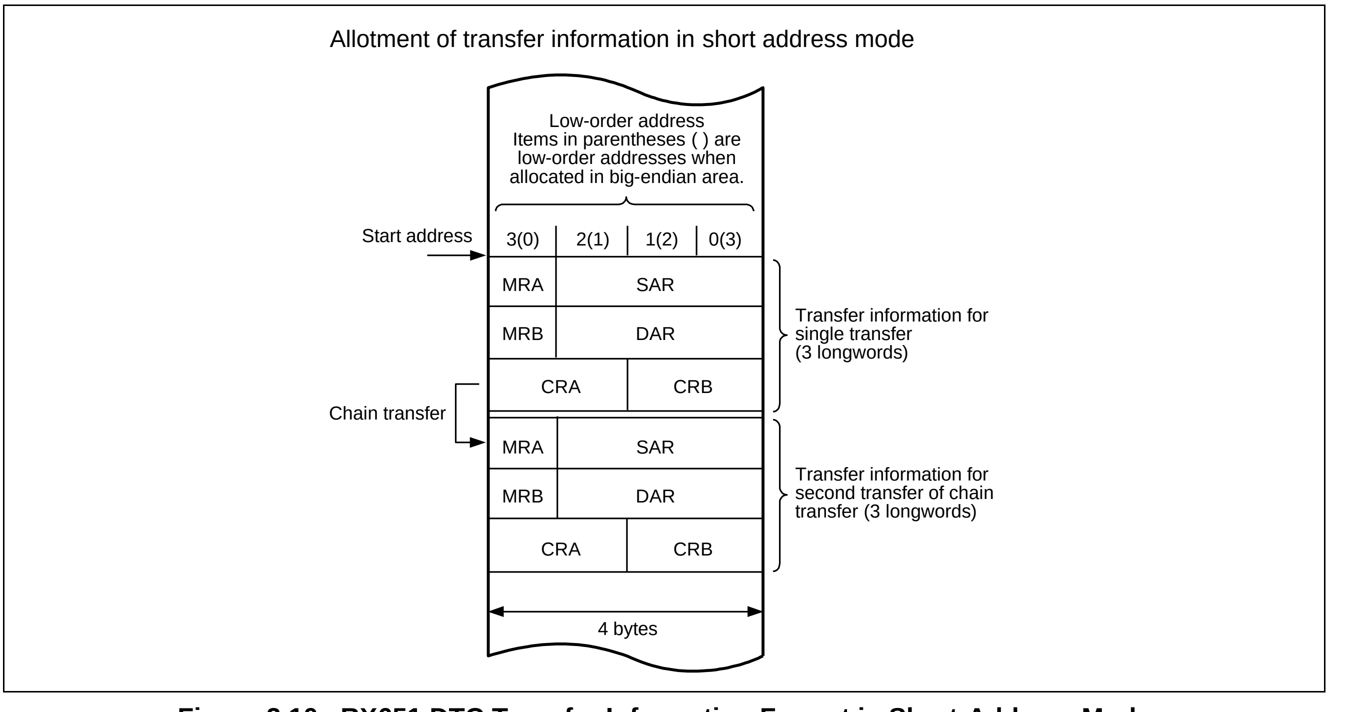

Allotment of transfer information in short address mode
Low-order address
Items in parentheses ( ) are low-order addresses when allocated in big-endian area.
Start address
3(0)
2(1)
1(2)
0(3)
MRA
SAR
MRB
DAR
CRA
CRB
Chain transfer
MRA
SAR
MRB
DAR
CRA
CRB
Transfer information for single transfer (3 longwords)
Transfer information for second transfer of chain transfer (3 longwords)
4 bytes

Figure 2.10 RX651 DTC Transfer Information Format in Short-Address Mode

Short-address mode supports 16 megabytes of transfer space in address ranges 00000000h to 007FFFFFh and FF800000h to FFFFFFFFh (excluding reserved areas).

2.5.7 Module Stop

The initial state of the peripheral modules of the RX651 is stopped, due to the low power consumption function. However, the initial state of the DTC is operational, so there is no need to cancel the module stop state. Module stop can be applied to the DTC, but doing so also stops the DMAC because the same control bit in the module stop control register is used for both the DTC and the DMAC. (The EXDAMC and EDMAC are controlled separately.)

2.6 Direct Memory Access Controller (DMACAa)

Direct memory access control functionality is implemented on the SH7044 by an on-chip DMAC and on the RX651 by an on-chip DMACA and by a dedicated on-chip EXDMACa for transfers between external areas. The internal bus configuration of the RX651 differs from that of SH microcontrollers. It supports independent data transfers by CPU instruction execution and by the DMAC or DTC for improved transfer performance.

2.6.1 Comparison of Specifications

The functions and features of the SH7044 and RX651 are shown below.

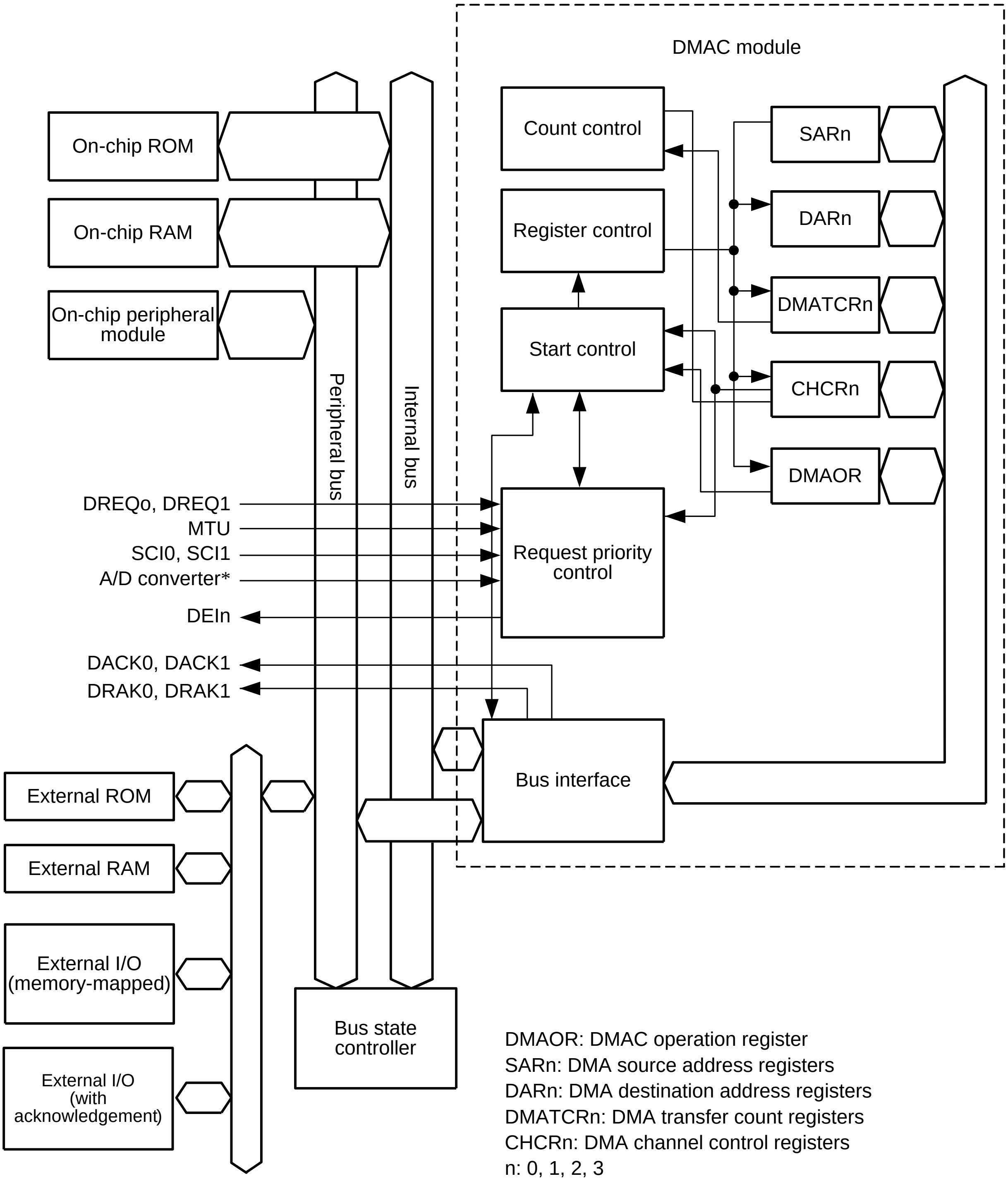
Table 2.16 Comparison of SH7044 (DMAC) and RX651 (DMACA and EXDMACa) Functions

		SH7044	RX651	
Item		DMAC	DMACA	EXDMACa
Number of channels		4 channels	8channels	2 channels
Maximum transfer count (maximum transfer data unit count on RX)		16 M (16,777,216)	64M data units (block transfer mode max. total transfer count: 1,024 data units × 65535 blocks) Free running mode settable	1 M data units (block transfer mode max. total transfer count: 1,024 data units × 1,024 blocks)
DMA activation sources		• External request • On-chip module request • Auto request (Equivalent to software trigger)	• External requests not supported. • On-chip module request • Software trigger • External interrupts	• External request • On-chip module request • Software trigger
Channel priority		Selectable between the following: • ① CH0> CH1> CH2> CH3 • ② CH0> CH2> CH3> CH1 • ③ CH2> CH0> CH1> CH3④Round robin	Fixed (channel 0 > channel 1 > channel 2 ···> channel7)	Fixed (channel 0 > channel 1)
Transfer data	1 data unit	8 bits, 16 bits, 32 bits	8 bits, 16 bits, 32 bits	8 bits, 16 bits, 32 bits
	Repeat size	—	Data units: 1 to 1,024	Data units: 1 to 1,024
	Block size	—	Data units: 1 to 1,024	Data units: 1 to 1,024
	Cluster size	—	—	Data units: 1 to 8
Transfer modes		• None (The transfer mode on the SH is equivalent to normal transfer mode on the RX.)	• Normal transfer mode • Repeat transfer mode • Block transfer mode	• Normal transfer mode • Repeat transfer mode • Block transfer mode • Cluster transfer mode
Bus modes		• Cycle-steal mode • Burst mode	—	—
Address modes		• Single address mode • Dual address mode	—	• Single address mode • Dual address mode
Address update mode		• Address is fixed. • Address is incremented. • Address is	• Address is fixed. • Offset addition • Address is incremented.	• Address is fixed. • Offset addition • Address is incremented.

		decremented.	• Address is decremented.	Address is decremented.
Interrupt request	Transfer-end interrupt	Generated after completion of the number of transfers specified by the transfer counter.	Normal transfer mode: After the specified number of transfers is completed Repeat transfer mode: After the transfer of the specified number of repeats is completed Block transfer mode: After the transfer of the specified number of blocks is completed Cluster transfer mode: After the transfer of the specified number of clusters is completed (EXDMACa only)	
	Transfer escape-end interrupt	—	Generated after completion of data transfer equivalent to the repeat size or when the extended repeat area overflows.	
Other		• Source address reload function	• Extended repeat area function • Event link function	• Extended repeat area function

2.6.2 DMAC Block Diagram

A block diagram of the SH7044's DMAC is shown below.



DMAOR: DMAC operation register
SARn: DMA source address registers
DARn: DMA destination address registers
DMATCRn: DMA transfer count registers
CHCRn: DMA channel control registers
n: 0, 1, 2, 3

Note: * A/D on other than A mask products, A/D1 on A mask products.

Figure 2.11 SH7044 DMAC Block Diagram

A block diagram of the RX651's DMACA is shown below.

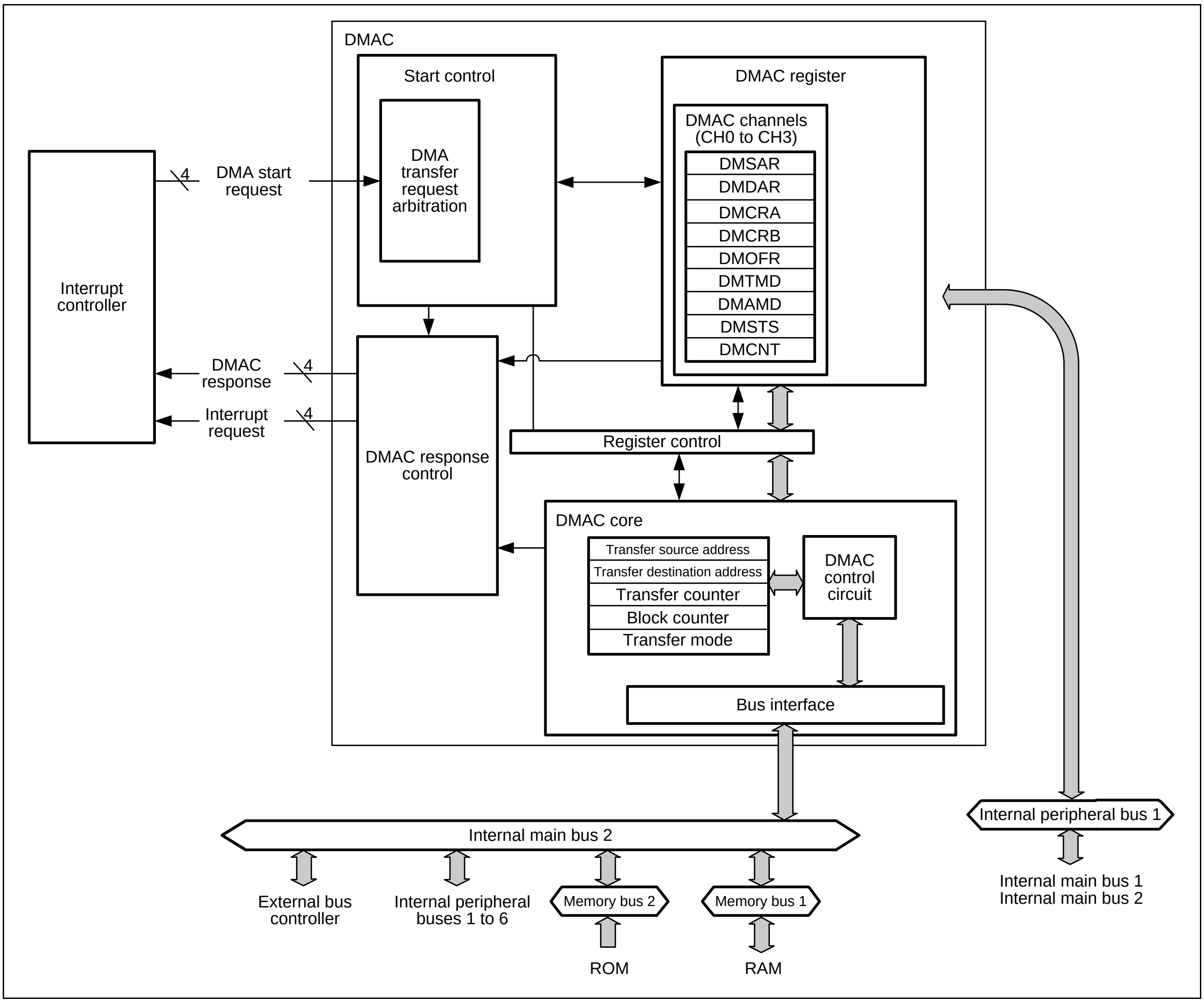


Figure 2.12 RX651 DMACA Block Diagram

A block diagram of the RX651's EXDMACa is shown below.

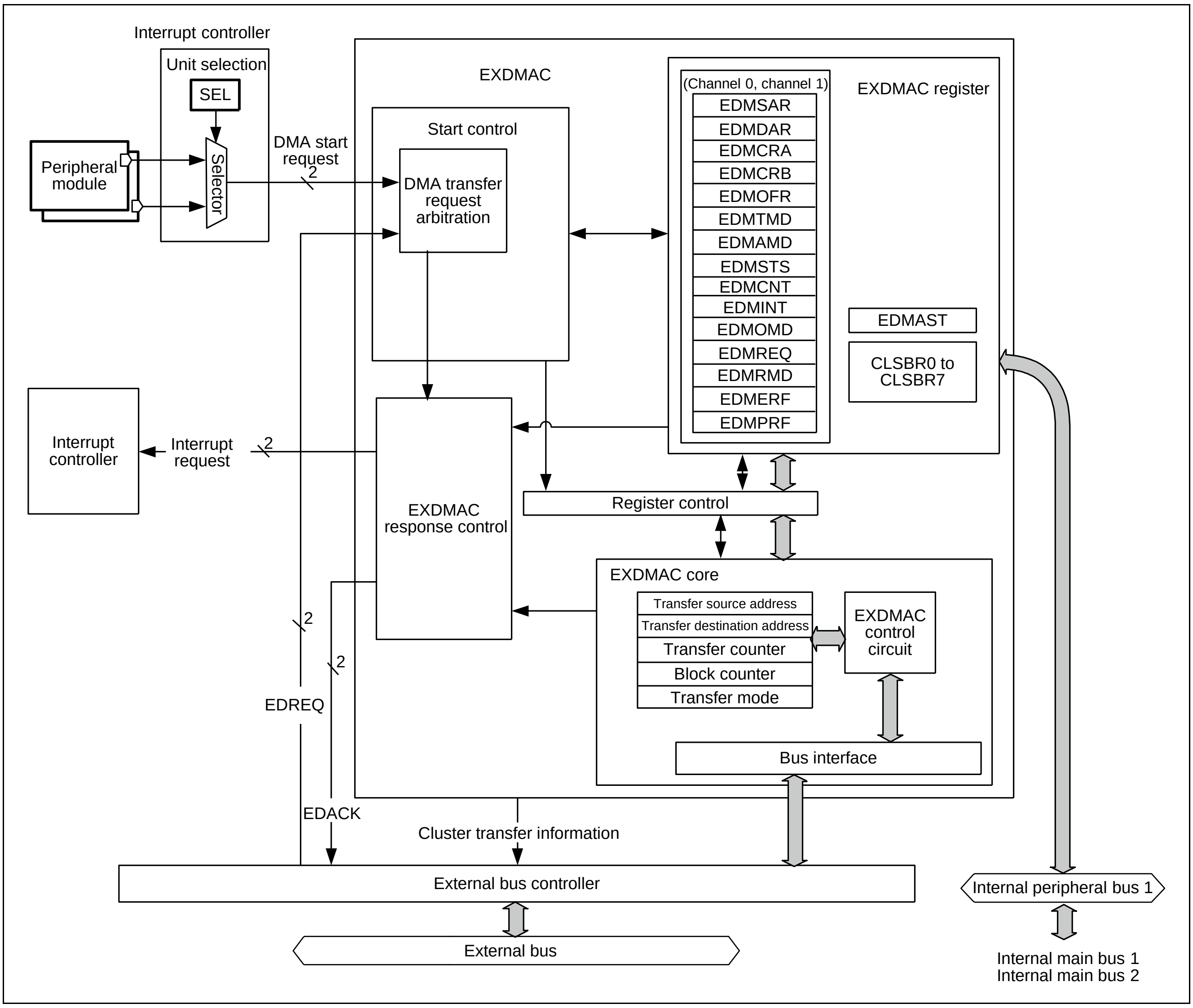


Figure 2.13 RX651 EXDMACa Block Diagram

2.6.3 Comparison of Registers

Table 2.17 "SH7044 / RX651 DMAC / DMACA Register Comparison" compares the DMAC registers of the SH7044 and the DMACA registers of the RX651.

Table 2.18 "SH7044 / RX651 DMAC / EXDMACa Register Comparison" compares the DMAC registers of the SH7044 and the EXDMACa registers of the RX651.

Table 2.17 SH7044/RX651 DMAC/DMACA Register Comparison

SH7044	RX651
DMAC n: 0 to 3	DMACA m: 0 to 7
DMA operation register (DMAOR)	DMA module start register (DMAST)
DMA source address register n (SARn)	DMA transfer source register m (DMACm.DMSAR)
DMA destination register n (DARn)	DMA transfer destination register m (DMACm.DMDAR)
DMA transfer count register n (DMATCRn)	DMA transfer counter register m (DMACm.DMCRA)
DMA channel control register (CHCRn)	DMA block transfer count register m (DMACm.DMCRB)
—	DMA transfer mode register m (DMACm.DMTMD)
	DMA interrupt setting register m (DMACm.DMINT)
	DMA address mode register m (DMACm.DMAMD)
	DMA transfer enable register m (DMACm.DMCNT)
	DMA software start register m (DMACm.DMREQ)
	DMA status register m (DMACm.DMSTS)
	DMA activation source flag control register m (DMACm.DMCSL)
	DMA offset register (DMAC0.DMOFR)
	DMA interrupt status register m (DMACm.DMIST)

Note: In the register symbols above, n and m represent the respective DMA channel numbers.

Table 2.18 SH7044/RX651 DMAC/EXDMACa Register Comparison

SH7044	RX651
DMAC n: 0 to 3	EXDMACa p: 0 to 1
DMA operation register (DMAOR)	EXDMA module start register (EDMAST)
DMA source address register n (SARn)	EXDMA transfer source register p (EXDMACp.EDMSAR)
DMA destination register n (DARn)	EXDMA transfer destination register p (EXDMACp.EDMDAR)
DMA transfer count register n (DMATCRn)	EXDMA transfer counter register p (EXDMACp.EDMCRA)
DMA channel control register (CHCRn)	EXDMA block transfer count register p (EXDMACp.EDMCRB)
—	EXDMA output setting register p (EXDMACp.EDMOMD)
	EXDMA transfer mode register p (EXDMACp.EDMTMD)
	EXDMA interrupt setting register p (EXDMACp.EDMINT)
	EXDMA address mode register p (EXDMACp.EDMAMD)
	EXDMA transfer enable register p (EXDMACp.EDMCNT)
	EXDMA software start register p (EXDMACp.DEMREQ)
	EXDMA status register p (EXDMACp.EDMSTS)
	EXDMA external request sense mode register p (EXDMACp.EDMRMD)
	EXDMA external request flag register p (EXDMACp.EDMERF)
	EXDMA peripheral request flag register p (EXDMACp.EDMPRF)
	EXDMA offset register (EXDMAC0.EDMOFR)
	Cluster buffer register y (CLDBR0 to CLDBR7)

Note: In the register symbols above, n and p represent the respective DMA channel numbers.

2.6.4 DMA Activation Sources and Settings

Table 2.19 "DMA Activation Source Comparison" lists the types of transfer activation sources of the respective DMAC modules.

Table 2.19 DMA Activation Source Comparison

	SH7044	RX651	
DMA Activation Sources	**DMAC**	**DMACA**	**EXDMACa**
Activation by software	Supported	Supported	Supported
Activation by external device via request pin	Supported (activation by _DREQ signal)	Not supported	Supported (activation by EDREQn signal)
Activation by peripheral module	Supported	Supported (activation by interrupt via external interrupt input also supported)	Supported

On the SH7044, DMA activation by peripheral module requires that the activation source be specified by a resource selector setting in the DMA channel control register (RS3 to RS0 in CHCRx). On the RX651 (DMACA) DMA activation by peripheral module requires specification of the activation source's vector number in the DMAC activation request select register (DMRSRm, m: channel 0 to 3) of the interrupt controller.

2.6.5 Transfer Sources and Destinations

The transfer sources and destinations supported by each DMA are listed below.

Table 2.20 SH7044 DMAC Transfer Sources and Destinations

	Transfer Destination				
Transfer Sources	**External Device with DACK**	**External Memory**	**Memory-Mapped External Device**	**On-Chip Memory**	**On-Chip Peripheral Module**
External Device with DACK	Not supported	●	●	Not supported	Not supported
External Memory	●	○	○	○	○
Memory-Mapped External Device	●	○	○	○	○
On-Chip Memory	Not supported	○	○	○	○
On-Chip Peripheral Module	Not supported	○	○	○	○

●: Single address mode transfers supported. ○: Dual address mode transfers supported.

Table 2.21 RX651 DMACA Transfer Sources and Destinations

	Transfer Destination				
Transfer Sources	**External Device with EDACK**	**External Memory**	**Memory-Mapped External Device**	**On-Chip Memory**	**On-Chip Peripheral Module**
External Device with DACK	Not supported	Not supported	Not supported	Not supported	Not supported
External Memory	Not supported	○	○	○	○
Memory-Mapped External Device	Not supported	○	○	○	○
On-Chip Memory	Not supported	○	○	○	○
On-Chip Peripheral Module	Not supported	○	○	○	○

○: Transfers supported.

Table 2.22 RX651 EXDMACa Transfer Sources and Destinations

	Transfer Destination				
Transfer Sources	**External Device with EDACK**	**External Memory**	**Memory-Mapped External Device**	**On-Chip Memory**	**On-Chip Peripheral Module**
External Device with EDACK	Not supported	●	●	Not supported	Not supported
External Memory	●	○	○	Not supported	Not supported
Memory-Mapped External Device	●	○	○	Not supported	Not supported
On-Chip Memory	Not supported	Not supported	Not supported	Not supported	Not supported
On-Chip Peripheral Module	Not supported	Not supported	Not supported	Not supported	Not supported

●: Single address mode transfers supported. ○: Dual address mode transfers supported.

2.6.6 Transfer Modes

The transfer modes of the SH7044 and RX651 are described below.

The concept of transfer mode does not apply on the SH7044. When switching to the RX651, the equivalent transfer mode is normal transfer mode. However, if the source address reload function was used on the SH7044, it is possible to achieve equivalent results on the RX651 by using repeat mode to repeat the source address for four transfer units. This makes it possible to reproduce the transfer method of the SH7044 by using the transfer modes of the RX651.

Table 2.23 RX651 Transfer Modes

Transfer Mode	DMACA	EXDMACa	Remarks
Normal transfer	○	○	Equivalent to the transfer method of the SH7044
Repeat transfer	○	○	Usable as a substitute for source address reload on the SH7044
Block transfer	○	○	
Cluster transfer	Not supported	○	

2.6.7 Address Modes

The SH7044 has two address modes: single address mode and dual address mode.

The EXDMACa of the RX651 has a single address mode and a dual address mode like the SH7044. In single address mode a DMA transfer can be completed in a single bus cycle. Two bus cycles are required to complete a DMA transfer in dual address mode. On the DMACA the address mode concept does not apply, but the method of specifying addresses and the operation are equivalent to dual address mode on the SH7044.

2.6.8 Bus Modes

On the SH7044 the bus mode can be specified as either cycle-steal mode or burst mode. In cycle-steal mode the bus is released to another bus master when a single transfer finishes. In burst mode the bus is not released after the start of a DMA transfer until the transfer finishes.

On the RX651 it is not possible to specify the bus mode of the DMACA or EXDMACa. This is because the bus architecture differs from that of the SH7044. The RX651 supports parallel operation when the bus master accesses a different slave. On the RX651 it is possible for the DMAC to perform transfers between the peripheral bus and the external bus while the CPU is accessing the ROM to fetch CPU instructions or the RAM to manipulate operands.

Figure 2.14 Parallel Bus Operation shows an example in which the DMAC accesses the peripheral bus and the external bus using internal main bus 2 while the CPU is accessing the ROM and RAM.

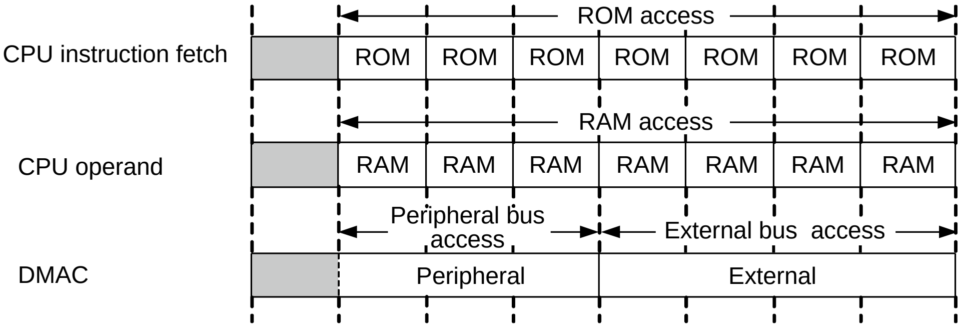


Figure 2.14 Parallel Bus Operation

2.6.9 Module Stop

The initial state of the peripheral modules of the RX651 is stopped, due to the low power consumption function. However, the initial state of the DMACA and EXDMAC is operational, so there is no need to cancel the module stop state. Module stop can be applied to the DMACA, but doing so also stops the DTC because the same control bit in the module stop control register is used for both the DTC and the DMAC.

2.7 Multifunction Timer Pulse Unit 3(MTU3a)

2.7.1 Comparison of Specifications

Table 2.24 Comparison of MTU Specifications on SH7044 and RX651

Item		SH7044	RX651
Pulse I/O		Maximum 16	Maximum 28
Pulse input		—	3
Count clock		Selectable for each channel among six clocks based on the internal clock (ϕ) and eight clocks employing external clocks (TCLKA, TCLKB, TCLKC, and TCLKD).	11 types can be selected for each channel MTU0 is 14 types, MTU2 is 12 types, MTU5 is 10 kinds, MTU1 & MTU2 (LWA = 1) 4 types)
Function settings	MTU0 to 4 MTU6 *1 MTU7 *1 MTU8 *1	• Compare match waveform output • Input capture function • Synchronous operation — Synchronized writing to multiple timers (TCNT) (Except MTU8) — Clearing synchronized with compare match or input capture (Except MTU8) — I/O with various registers in synchronization with counter (Except MTU8) • PWM mode — PWM output with user-specified duty (Except MTU8) — Up to 12-phase PWM output combined with synchronous operation (Except MTU8)	
	MTU0, MTU3, MTU4 MTU6*1 MTU7*1 MTU8*1	• Support for buffer operation settings •	
	MTU1, MTU2	Up- or down-counting of two-phase encoder pulses in phase counting mode	
	MTU3, MTU4 MTU6 *1 MTU7 *1 MTU8 *1	• Complementary PWM and reset synchronous PWM operation by interlocking operation of MTU3 / MTU4 and MTU6 / MTU7 enables 6-phase positive / negative total 12-phase output. • Double buffer function can be set in complementary PWM mode	
	MTU3,4	In conjunction with MTU0, the AC synchronous motor (brushless DC motor) drive mode with complementary PWM and reset synchronous PWM can be set, and waveform output of two types (chopping, level) is selectable.	
	MTU5*1		Capable of operation as a dead-time compensation counter
	MTU0/5*1 MTU1,2 MTU8 *1	—	32-bit phase counting mode specifiable by combining MTU1 and MTU2 and through interlocked operation with MTU0/MTU5 and MTU8
Complementary PWM mode		Interrupts at counter peaks and troughs (Interrupts on crests and troughs of counter values)	
Interrupt sources (See separate listing for details.)		23	43
Buffer operation		Automatic transfer of register contents	
Trigger generation		A/D converter start trigger	A/D converter start trigger PPG output trigger

DMAC activation	MTU0 to MTU 4 MTU5 to MTU 8*1	TGR compare match or input capture Note: On the SH7044 the registers are named TGRnA (n: channel number)	
	MTU4,7	—	Overflow interrupt
DTC activation	MTU0 to MTU 4 MTU5 to MTU 8*	TGR compare match or input capture,	
	MTU4 MTU8*1	TGR compare match or input capture, TCNT overflow or underflow	
A/D conversion start triggers		MTU0 to MTU4: TGRA compare match or input capture	MTU0 to MTU4,6,7: TGRA compare match or input capture MTU0: TGRE compare match MTU4,7: TCNT underflow during complementary PWM mode (valley)
PPG triggers		—	MTU0 to MTU3: TGRA and TGRB compare match or input capture
A/D conversion start request delay function		—	MTU4: Start request at match of TADCORA or TADCORB and TCNT
Interrupt skipping function		—	MTU3: TGRA compare match interrupt skipping MTU4: TCIV interrupt skipping

*1 Only RX651

Table 2.25 List of MTU Interrupt Sources on SH7044 and RX651

	SH7044/RX651					RX651			
Item	**MTU0**	**MTU1**	**MTU2**	**MTU3**	**MTU4**	**MTU5**	**MTU6**	**MTU7**	**MTU8**
Compare match/input capture nA	○	○	○	○	○		Δ	Δ	Δ
Compare match/input capture nB	○	○		○	○		Δ	Δ	Δ
Compare match/input capture nC	○			○	○		Δ	Δ	Δ
Compare match/input capture nD	○			○	○		Δ	Δ	Δ
Overflow	○	○	○	○	○		Δ	Δ	Δ
Underflow		○	○		○			Δ*1	
Compare match nE	Δ								
Compare match nF	Δ								
Compare match/input capture nU						Δ			
Compare match/input capture nV						Δ			
Compare match/input capture nW						Δ			

n: Channel number ○: Compatible between SH7044 and RX651 Δ: Added on RX651

*1: Valid only in complementary PWM mode

2.7.2 Handling of Interrupt Flags

The RX651's MTU2a and the SH7044's MTU are software compatible. With the exception of changes to the timer status register (TSR) interrupt flags, it is possible to migrate the functions of MTU0 to MTU4and MTU6 to MTU8 without changing the registers. (It is necessary to make separate changes to the initial settings, such as the pin settings.) The one significant difference is that on the RX651 the timer status register (TSR) contains no interrupt flags. Nevertheless, it is possible to implement equivalent processing by using the interrupt request register (IR) in the interrupt controller corresponding to the MTU (IR142 and above).

The RX651 MTU interrupt is assigned to selective interrupt A.

The interrupt controller's selective interrupt A status flag (PIARk.PIRn) is not automatically cleared, but it does not affect interrupt request generation.

See Chapter 1.8 for interrupts.

2.7.3 List of Registers

Whether or not changes to the register settings are needed when switching from the SH7044 to the RX651 is indicated below.

Table 2.26 List of MTU Registers (1/3)

Register Name	SH7044 (MTU)	RX651 (MTU3a)	Change
Timer control register	TCR0 to TCR4	MTU0.TCR to MTU4.TCR MTU6.TCR to MTU84.TCR	◎
		MTU5.TCRU/V/W	*1
Timer control register 2		MTU0.TCR2 MTU3.TCR2 to MTU4.TCR2 MTU6.TCR2 to MTU8.TCR2	*1
Timer mode register	TMDR0 to TMDR4	MTU0.TMDR1	◎
		MTU1.TMDR1 to MTU2.TMDR1 MTU3.TMDR1 to MTU4.TMDR1 MTU6.TMDR1 to MTU8.TMDR1	*1
Timer mode register 2		MTU1.TMDR2A MTU2.TMDR2B	*1
Timer mode register 3		MTU.TMDR3	*1
Timer I/O control register	TIOR0H, TIOR3H, TIOR4H	MTU0.TIORH, MTU3.TIORH,. MTU6,TIORH	◎
	TIOR1, TIOR2	MTU1.TIOR, MTU2.TIOR	◎
	TIOR0L, TIOR3L, TIOR4L	MTU0.TIORL, MTU3.TIORL, TU4.TIORL	◎
		MTU6.TIORH, MTU8.TIORH	*1
Timer compare match clear register		TCNTCMPCLR	*1
Timer interrupt enable register	TIER0	MTU0.TIER	MTU0.TIER
	TIER1, TIER2	MTU1.TIER , MTU2.TIER	◎
	TIER3, TIER4	MTU3.TIER,MTU6.TIER	◎
		MTU4.TIER、MTU7.TIER、MTU8.TIER	*1
Timer status register	TSR0		
	TSR1, TSR2	MTU1.TSR MTU2.TSR	∆
	TSR3, TSR4	MTU3.TSR MTU4.TSR	∆
		MTU6.TSR MTU7.TSR	

◎: Registers with identical bit assignments on the SH7044 and RX651

○: Registers where the RX651 has new functions (bits) assigned. (Except for the new function bits, the bit assignments are identical.)

∆: On the RX651 these registers contain no interrupt flags.

Note: * Registers with no equivalents on the SH7044. (These registers are for new functions added in the MTU2. When migrating programs that use the SH7044's MTU, the initial values can be used unaltered without any problem.)

Table 2.27 List of MTU Registers （2/3）

Register Name	SH7044 (MTU)	RX651 (MTU3a)	Change
Timer buffer operation transfer mode register		MTU0.TBTM、MTU3.TBTM、MTU4.TBTM、MTU6.TBTM、MTU7.TBTM	*1
Timer input capture control register		MTU1.TICCR	
Thai machine synchro clear register		MTU6.TSYCR	*1
Timer counter	TCNT0~4	MTU0.TCNT to MTU4.TCNT	◎
		MTU5.TCNT to MTU7.TCNT	*1
Timer long word count		MTU1.TCNTLW	
Timer general register	TGR0,3,4（A,B,C,D）	MTU0.TGRA to D MTU3.TGRA to D MTU4.TGRA to D	◎
		MTU0.TGRE,F MTU1.TGRC to F MTU2.TGRC to F MTU3.TGRE,F MTU4.TGRE,F MTU5.TGRA to F MTU6.TGRA to F MTU7.TGRA to F	*1
	TGR1,2（A,B）	MTU1.TGRA,B MTU2.TGRA,B	◎
Timer longword general register		MTU1.TGRALW MTU1.TGRBLW	
Timer start register	TSTR	MTU.TSTR	◎
Timer synchro register	TSYR	MTU.TSYR	◎
Timer counter Synchro start register		MTU.TCSYSTR	
Timer read / write enable register		MTU.TRWERA、 MTU.TRWERB	*1
Timer output master permission register	TOER	MTU.TOERA、	◎
		MTU.TOERB	*1
Timer output control register 1	TOCR	MTU.TOCR1A	○
		MTU.TOCR1B	*1

◎: Registers with identical bit assignments on the SH7044 and RX651

○: Registers where the RX651 has new functions (bits) assigned. (Except for the new function bits, the bit assignments are identical.)

∆: On the RX651 these registers contain no interrupt flags.

Note: *1 Registers with no equivalents on the SH7044. (These registers are for new functions added in the MTU2. When migrating programs that use the SH7044's MTU, the initial values can be used unaltered without any problem.)

Table 2.27 List of MTU Registers （3/3）

Register Name	SH7044（MTU）	RX651（MTU3a）	Change
Timer output control register 2		MTU.TOCR2A、MTU.TOCR2B	*1
Timer output level buffer register		MTU.TOLBRA、MTU.TOLBRB	*1
Timer gate control register A		MTU.TGCRA	*1
Timer sub-counter		MTU.TCNTSA、MTU.TCNTSB	*1
Timer cycle data register		MTU.TCDRA、MTU.TCDRB	*1
Timer cycle buffer register		MTU.TCBRA、MTU.TCBRA	*1
Timer dead time data register		MTU.TDDRA、MTU.TDDRB	*1
Timer dead time enable register		MTU.TDERA、MTU.TDERB	*1
Timer buffer transfer setting register		MTU.TBTERA、MTU.TBTERB	*1
Timer waveform control register		MTU.TWCRA、MTU.TWCRB	*1
Noise filter control register n		MTU0.NFCR0、MTU1.NFCR1、MTU2.NFCR2、MTU3.NFCR3、MTU4.NFCR4、MTU6.NFCR6、MTU7.NFCR7、 MTU8.NFCR8	*1
Timer waveform control register		MTU.TWCRA、MTU.TWCRB	*1
Noise filter control register n		MTU0.NFCR0、MTU1.NFCR1、MTU2.NFCR2、MTU3.NFCR3、MTU4.NFCR4、MTU6.NFCR6、MTU7.NFCR7、 MTU8.NFCR8	*1
Noise filter control register 5		MTU5.NFCR5	*1
Timer A / D conversion start request control register		MTU4.TADCR	*1
Timer A / D conversion start request cycle setting register		MTU4.TADCORA、 MTU4.TADCORB、 MTU7.TADCORA、 MTU7.TADCORB	*1
Timer A / D conversion start request cycle setting buffer register		MTU4.TADCOBRA、 MTU4.TADCOBRB、 MTU7.TADCOBRA、 MTU7.TADCOBRB	*1
Timer interrupt decimation mode register		MTU.TITMRA、MTU.TITMRB	*1
Timer interrupt decimation setting register 1		MTU.TITCR1A	*1
Timer interrupt decimation count counter 1		MTU.TITCNT1A	*1
Timer interrupt decimation setting register 2		MTU.TITCR2A	*1
Timer interrupt decimation count counter 2		MTU.TITCNT2A	*1

◎: Registers with identical bit assignments on the SH7044 and RX651

○: Registers where the RX651 has new functions (bits) assigned. (Except for the new function bits, the bit assignments are identical.)

∆: On the RX651 these registers contain no interrupt flags.

Note: *1 Registers with no equivalents on the SH7044. (These registers are for new functions added in the MTU2. When migrating programs that use the SH7044's MTU, the initial values can be used unaltered without any problem.)

2.7.4 Unit Selection Function

Some interrupt sources of the MTU and TPU are assigned to common vectors. It is therefore necessary when using the MTU to specify which interrupt will be using each vector by setting the corresponding selector.

2.7.5 Module Stop

The initial state of the peripheral modules of the RX651 is stopped, due to the low power consumption function. The initial state of the MTU is stopped as well. Do not fail to cancel the module stop state when making settings to the module. Before accessing the module stop control register to cancel the module stop state, first cancel register write protection.

2.8 Watchdog Timers（WDTA）

2.8.1 Comparison of Specifications

The SH7044 incorporates the WDT as its watchdog timer module. The RX651 incorporates, in addition to the WDTA, the IWDTa, which operates on a dedicated independent clock. The specifications of these modules are compared below.

Table 2.27 Comparison of WDT, WDTA, and IWDTa Specifications on SH7044 and RX651

	SH7044	**RX651**	
Item	**WDT**	**WDTA**	**IWDTa**
Clock source	System clock (ϕ)	Peripheral clock (PCLK)	IWDT dedicated clock (IWDTCLK)
Clock frequency division ratio	ϕ/2, 64, 128, 256, 512, 1024, 4096, 8192	PCLK/4, 64, 128, 512, 2048, 8192	IWDTCLK/1, 16, 32, 64, 128, 256
Count operation	8-bit up-counter	14-bit down-counter	14-bit down-counter
Operating modes	• Watchdog timer mode • Interval timer mode	Change with option setting memory instead of operation mode concept —Output reset allowed (Equivalent to watchdog timer mode) —Interrupt Request allowed (Equivalent to interval timer mode)	Change with option setting memory instead of operation mode concept —Output reset allowed (Equivalent to watchdog timer mode) —Interrupt Request allowed (Equivalent to interval timer mode)
Count start condition	Timer enable bit in timer control register set to “enabled”	Selectable between the following: 1. Automatic count start after a reset (auto-start mode) 2. Count start by refresh operation (register start mode)	Selectable between the following: 1. Automatic count start after a reset (auto-start mode) 2. Count start by refresh operation (register start mode)
Count stop condition	Watchdog timer mode • Overflow • Power-on reset Interval timer mode • Timer enable bit in timer control register set to “disabled” • Power-on reset	• Underflow • Reset (down-counter, return to register initial value) • Refresh error	• Underflow • Reset (down-counter, return to register initial value) • Refresh error
Operation at overflow/ underflow	Watchdog timer mode • WDTOVF output • Internal reset Interval timer mode • Interrupt	• When reset output is permitted • Internal reset • When interrupt request output is enabled • Interrupt	• When reset output is permitted • • Internal reset • When interrupt request output is enabled • Interrupt
Other	—	Window function The following are specified by settings in option function select register 0: • Clock frequency division ratio	Cooperation with event link controller ・Window function ・Can operate even with low power consumption The following are specified

		• Refresh window start/end • Timeout period • Operation at underflow	by settings in option function select register 0: • Clock frequency division ratio • Refresh window start/end • Timeout period • Operation at underflow

—: Function not implemented on SH7044.

2.8.2 Count Start condition

The SH7044 group starts counting with a "1" write to the timer enable bit.

With the RX651, you can select the register start mode (similar to the SH7044 group) that starts counting by writing to the register in the option function selection register, or the auto start mode that automatically starts counting after reset.

In the RX651 auto-start mode, counting automatically starts after a reset is released in accordance with the settings in option function select register (OFS0) in the ROM.

In register start mode, counting is started by refresh operation after the respective registers are set after a reset is released.

2.8.3 Interrupt

The RX651's WDTA and IWDTa interrupts support both Non-maskable interrupt and interrupts.

The interrupt status flag (IRn.IR) of the interrupt controller is automatically cleared when an interrupt is received.

See Chapter 1.8 for interrupts.

2.8.4 Refresh Operation

The RX651 refreshes the count by writing “00h” to the WDT refresh register (WDTRR) and then writing “FFh”.

Writing to the WDT refresh register must be done within the refresh permission period.

To refresh the IWDTa count, write to the IWDT refresh register (IWDTRR) within the refresh permission period.

Table 2.28 Comparison of refresh operations

Item	**SH7044 Group**	**RX651（WDTA）**
Refresh Condition	Write to the watchdog timer counter (TCNT)	Write "FFh" after writing "00h" to the refresh register (WDTRR) within the Refresh permission period
Initial value of counter after refresh	The value written to the watchdog timer counter (TCNT)	Register start mode — The value selected by the timeout period selection bit (WDTCR.TOPS) of the WDT control register. Auto start mode — Option Function The value selected by the WDT timeout period selection bit (OFS0.WDTTOPS) in the selection register.

2.8.5 Register write limit

SH7044 Group and RX651 both have limited register writes for WDT. The register write limits are shown below.

Table 2.29 SH7044 group register write limit

Item	Write limit
Timer counter（TCNT） Reset control / Status register（RSTCSR） — Reset enable（RSTCSR.RSTE） — Reset select（RSTCSR.RSTS）	Write in the word size of the following configuration — high byte: "5Ah" — Low byte: Write data
Timer counter／Status register（TCSR） Reset control / Status register（RSTCSR） — Watchdog timer overflow flag（RSTCSR.WOVF）	Write in the word size of the following configuration — High byte: "A5h" — Low byte: Write data

Table 2.30 RX651 Register write limits

Item	Write limit
WDT control register（WDTCR） WDT reset control register（WDTRCR） IWDT control register（IWDTCR） IWDT reset control register（IWDTRCR） IWDT count stop control register（IWDTCSTPR）	Writable only once between reset release and first refresh operation

2.8.6 Interrupt

The RX651's WDTA and IWDTa interrupts support both Non-maskable interrupt and interrupts.

The interrupt status flag (IRn.IR) of the interrupt controller is automatically cleared when an interrupt is received.

See Chapter 1.8 for interrupts.

2.8.7 Module Stop

The initial state of the peripheral modules of the RX651 is stopped, due to the low power consumption function. However, the WDTA and IWDTa have no module stop function. Their initial operating state is determined by settings in the option-setting memory. Note that when all modules are stopped, the WDTA stops counting and retains its state. The operation of the IWDTa when all modules are stopped is selectable between operational and stopped by a setting in the option-setting memory.

2.9 Serial Communication Interface（SCIg, SCIi, SCIh）

2.9.1 Comparison of Specifications

In contrast to the SCI of the SH7044, the RX651 integrates the SCIg/SCIi/SCIh. In addition to the conventional asynchronous and clock-synchronous transfer modes, the SCIc provides smartcard (IC card) interface support as an extended asynchronous mode. In addition, it supports simple I^{2}C bus interface single master operation and simple SPI bus interface mode. SCI2 has an extended serial interface. For details of the transfer modes that are not supported on the SH7044, refer to the User's Manual: Hardware.

Table 2.31 SCI Differences

Item		SH7044	RX651
Number of channels		2 channels (SCI0, SCI1)	13 channels SCIg：SCI0 to 9 SCIi：SCI10 to 11 SCIh：SCI12
Serial communication modes		• Asynchronous • Clock-synchronous	• Asynchronous • Clock-synchronous • Smartcard interface • Simple I^{2}C bus • Simple SPI bus
Transfer speed		Any bit rate may be selected using the on-chip baud rate generator.	
Full-duplex communication		Transmit block: Support for continuous transmission using double-buffer configuration Receive block: Support for continuous reception using double-buffer configuration	
Data transfer		LSB-first only	Selectable between LSB-first and MSB-first (MSB-first only on simple I^{2}C bus)
Interrupt sources		• Transmit data-empty • Transmit-end • Receive data-full • Receive error	• Transmit data-empty • Transmit-end • Receive data-full • Receive error • Start condition* • Restart condition* • Stop condition generation-end* Note: * Used in simple I^{2}C mode.
Asynchronous mode	Data length	7 bits, 8 bits	7 bits, 8 bits,9 bits
	Stop bits	1 bit, 2 bits	
	Parity	Even parity, odd parity, or no parity	
	Receive error detection	Parity error, overrun error, or framing error	
	Hardware flow control	No	Yes (controllable using CTS and RTSn pins)
	Data match detection	No	It is possible to compare the received data and the comparison data register and generate an interrupt request if they match.
	Break detection	Possible by reading level of RxDn pin directly when a framing error occurs	
	Clock source	Selectable between internal and external clock	Selectable between internal and external clock Ability to input transfer rate clock from TMR (SCI5 and SCI6)
	Multi-processor communication	Yes	
	Noise	No	On-chip digital noise filter for input on

	cancellation		RXDn pins
Clock-synchronous mode	Data length	8 bits	
	Receive error detection	Overrun error	
	Hardware flow control	No	Yes (controllable using CTS and RTSn pins)
Smartcard interface		No	Yes
Simple I^{2}C mode		No	Yes
Simple SPI mode		No	Yes
Extended serial mode		No	Implemented on SCIh (SCI12) only

A comparison of the on-chip SCI registers is shown below.

Table 2.32 SCI Communication Registers

SH7044	RX651	Changed
Transmit data register (TDR)	Transmit data register (TDR)	◎
Transmit shift register (TSR)	Transmit shift register (TSR)	◎
Receive data register (RDR)	Receive data register (RDR)	◎
Receive shift register (RSR)	Receive shift register (RSR)	◎
Serial mode register (SMR)	Serial mode register (SMR)	◎
Serial control register (SCR)	Serial control register (SCR)	◎
Serial status register (SSR)	Serial status register (SSR/ SSRFIFO)	◎*1
Bit rate register (BBR)	Bit rate register (BBR)	◎
	Smartcard mode register (SCMR)	○
	Serial extended mode register (SEMR)	○
	Noise filter setting register (SNFR)	○*2
	I^{2}C mode registers 1 to 3 (SIMR1 to SIMR3)	*2
	I^{2}C status register (SISR)	—
	SPI mode register (SPMR)	*2*3
	Extended serial mode enable register (ESMER)	—
	Control registers 0 to 3 (CR0 to CR3)	—
	Port control register (PCR)	—
	Interrupt control register (ICR)	—
	Status register (STR)	—
	Status clear register (STCR)	—
	Control field 0 data register (CF0DR)	—
	Control field 0 compare enable register (CF0CR)	—
	Control field 0 receive data register (CF0RR)	—
	Primary control field 1 data register (PCF1DR)	—
	Secondary control field 1 data register (SCF1DR)	—
	Control field 1 compare enable register (CF1CR)	—
	Control field 1 receive data register (CF1RR)	—
	Timer control register (TCR)	—
	Timer mode register (TMR)	—
	Timer prescaler register (TPRE)	—
	Timer count register (TCNT)	—
	Receive data registers H, L, HL (RDRH, RDRL, RDRHL)	—
	Receive FIFO Data Register (FRDR)	—
	Transmit data registers H, L, HL (TDRH, TDRL, TDRHL)	—
	Transmit FIFO Data Register (FTDR)	—
	Modulation duty register (MDDR)	—
	FIFO control register (FCR)	—
	FIFO data count register (FDR)	—
	Line status register (LSR)	—
	Comparative data register (CDR)	—
	Data comparison control register (DCCR)	—
	Serial port register (SPTR)	—

◎: Registers with identical bit assignments on the SH7044 and RX651

○: Registers not present on the SH7044 that are required when using functions.

—: Registers with no equivalents on the SH7044. (When migrating programs that use the SH7044's SCI, the initial values can be used unaltered without any problem.)

Notes: 1. Only TDRE and RDRF differ.

The function of some bits differs depending on the smart card interface mode and non-smart card interface mode, and the FIFO mode and non-FIFO mode.

2. When migrating programs, the initial values can be used unaltered.
3. For information on register settings required when performing flow control using the CTS and RTS pins, and register bit assignments, see the User's Manual: Hardware.

2.9.2 Clock source selection

The RX651 can select the TMR clock input (SCI5, SCI6, SCI12 only) as the clock source when communicating in asynchronous mode.

Also, the SH7044 group operates on the clock of the port generator when the internal clock is selected, and the clock is fixed at 16 bits when the external clock is selected, while the RX65N can be selected from 8 bits or 16 bits.

2.9.3 Interrupt

The SH7044 group and RX651 can activate DTC and DMAC by interrupting with receive data full and transmit data empty.

Table 2.33 shows a list of Interrupt sources for the SH7044 group and RX72M.

See Chapter 1.8 for interrupts.

Table 2.33 List of SCI interrupt factors

Priority	Interrupt sources	Start by interrupt	
		SH7044 group	**RX651**
high ↑ Low	Receive error	Not possible	Impossible
	Receive data full	DMAC/DTC can be activated	DMAC/DTC can be activated
	Transmit data empty		
	Transmit end	Not possible	Not possible

2.9.4 Switching SCIs

Differences such as the following should be borne in mind when switching from the SH7044's SCI to the SCIc or SCId on the RX651:

1. TDRE and RDRF

The transmit register-empty (TDRE) and receive data-full (RDRF) flags in the serial status register of the SH7044 are not implemented on the SCIc or SCId modules of the RX651. The TDRE and RDRF flags on the SH7044 correspond to the IR (TXI) and IR (RXI) flags, respectively, of the interrupt controller on the RX651.2.

Determination of one-bit period and clock source selection

For communication in asynchronous mode, external clock input or TMR clock input can be selected as the clock source for determining the one-bit period by a setting in the serial extended mode register (SEMR). Also, the number of base clock cycles per one-bit period can be set to 8 or 16.

3. Digital noise filter

   The digital noise filter is activated or disabled by a setting in the serial extended mode register (SEMR). When enabling the noise filter, make sure to make the appropriate noise filter clock select setting in the noise filter setting register (SNFR).

4. Receive error interrupt

   The receive error interrupt is assigned to a group interrupt. The use of a group interrupt means that receive errors for 12 channels, SCI0 to SCI12, are assigned to a single vector. Therefore, when a receive error interrupt is generated it is necessary to detect the channel on which the error occurred by means of the ISn (n: channel number) flags in group interrupt source register 12 (GRP12). Within each channel, error handling for overrun errors, framing errors, and parity errors is the same as on the SH7044.

2.9.5 Module Stop

The initial state of the peripheral modules of the RX651 is stopped, due to the low power consumption function. The initial state of the SCI is also stopped. Do not fail to cancel the module stop state when making settings to the module. Before accessing the module stop control register to cancel the module stop state, first cancel register write protection.

2.10 10-bit A / D converter

2.10.1 Comparison of Specifications

The RX651 does not have a 10-bit A / D converter.

Use a 12-bit A / D converter (S12ADFa). (See 12-bit A / D converter)

2.11 12-bit A / D converter (S12ADFa)

2.11.1 Comparison of Specifications

The features and features of the SH7044 high-speed A/D Converter and RX651 12-bit A/D converters (S12ADFa) are shown below.

Table 2.34 Comparison of High-Speed A/D Converter Specifications on SH7044 and RX651

	SH7044	RX651
Item	**High-Speed A/D Converter**	**12-Bit A/D Converter (S12ADa)**
Resolution	10 bits	12 bits
Number of input channels	8 channels	2 units (S12AD, S12AD1) S12AD: 8 channels, S12AD1: 21 channels + 1 expansion
A/D conversion method	Successive approximation	Successive approximation
Conversion speed	2.9 µs per 1channel (operating frequency: 28 MHz)	Per channel (0.48 µs) (12-bit converter mode) Per channel (0.45 µs) (10-bit converter mode) Per channel (0.42 µs) (8-bit converter mode) (A / D converter clock ADCLK = 60MHz operation)
Operating modes	• Selectable between select mode and group mode • Selectable between single mode and scan mode	• Operation mode can be set individually for 2 units • Scan mode — Continuous scan mode — Single scan mode — Group scan mode With group priority control
A/D conversion start conditions	• Software trigger • Trigger by timer (MTU) • Asynchronous trigger (ADTRG pin)	• Software trigger • Trigger by timer (MTU、TPU、TMR、ELC) • Asynchronous trigger (ADTRG0# pin, ADTRG1 # pin)
Other functions	• Buffer operation • 2-channel simultaneous sampling	• Channel-only sample and hold function • Adjustable number of sampling states • Self-diagnosis function of 12-bit A / D converter • A / D conversion value addition mode and average mode can be selected • Analog input disconnection detection assist function • Double trigger mode • 12/10/8 bit conversion switching function • A / D data register auto clear function • Extended analog input function • Compare function • Event link function
Operations linked to A/D conversion-end interrupt	• CPU interrupt generation • DMAC or DTC activation	• Various CPU interrupts are generated for each mode • A compare interrupt request is generated when the comparison condition of the digital compare function is satisfied • Start DMAC or DTC
Low power consumption function	None	Support for module stop state setting

Conversion targets	AN pin	AN pin Internal reference voltage Temperature sensor Extended input

2.11.2 Input Channels

The SH7216 group consists of eight channels.

On the RX651 the S12ADC comprises two units, S12AD and S12AD1, one with eight channels and the other with 21 channels. As on the SH7216 Group, on the RX651 each unit incorporates an A/D converter. Simultaneous operation is possible, but continuous scan operation spanning the two units is not supported.Figure 2.15 compares the A/D converter configurations of the SH7216 Group and RX72M

Figure 2.15 compares the A/D converter configurations of the SH7216 Group and RX72M.

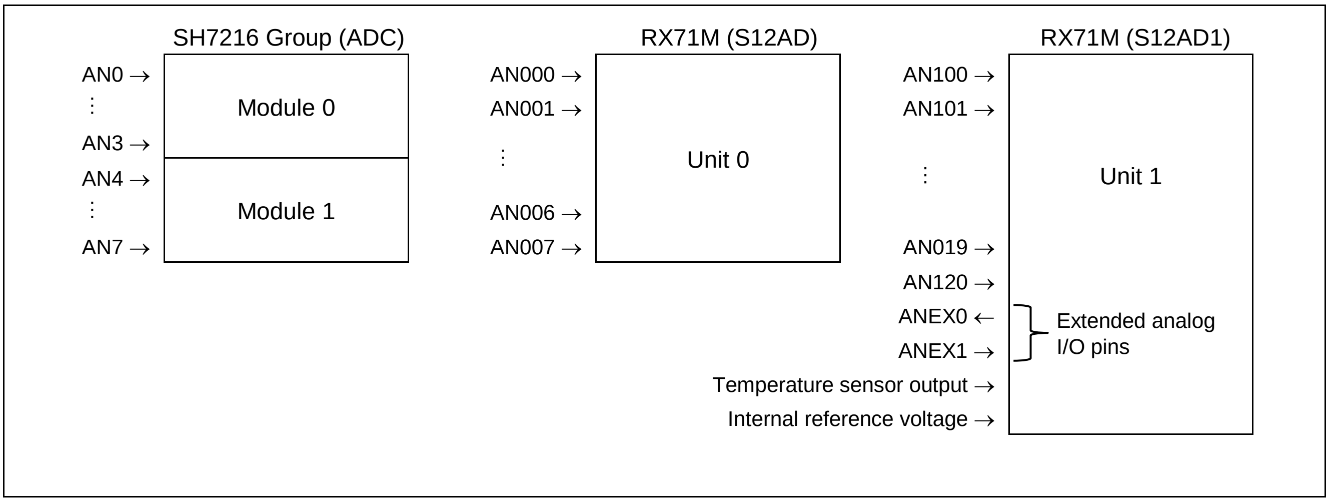


Figure 2.15 Comparison of A/D Converter Configurations

2.11.3 Scanning Sequence

Table 2.35 lists the scanning sequence when all channels are specified.

Table 2.35 A/D Converter Scanning Sequence

Microcontroller	A/D Converter	Conversion Sequence
SH7216 Group	ADC (module 0)	AN0 ⇒ AN1 ⇒ AN2 ⇒ AN3
	ADC (module 1)	AN4 ⇒ AN5 ⇒ AN6 ⇒ AN7
RX72M	S12AD	AN0 ⇒ AN1 ⇒ omitted ⇒ AN6 ⇒ AN7 ⇒ Temperature sensor output ⇒ Internal reference voltage It is possible to select group A priority control for group scan operation.
	S12AD1	AN100 ⇒ AN101 ⇒ omitted ⇒ AN119 ⇒ AN120 ⇒Temperature sensor output ⇒ Internal reference voltage It is possible to select group A priority control for group scan operation.

2.11.4 Operating Modes

The operation of the SH7044's high-speed A/D converter is determined by the following mode settings in combination.

- Channel designation mode
  Select mode: A single channel is specified.
  Group mode: Multiple channels are specified.
- Converter operation mode
  Single mode: A/D conversion is activated once.
  Scan mode: A/D conversion is activated repeatedly.

Table 2.36 SH7044 High-Speed A/D Converter Operating Modes

Operating Mode	Single Mode	Scan Mode
Select mode	1 conversion of 1 channel	Repeated conversions of 1 channel
Group mode	1 conversion of multiple channels	Repeated conversions of multiple channels

The corresponding operating modes, when switching from the SH7044's high-speed A/D converter, are listed below.

Table 2.37 A/D Converter Operating Mode Correspondences

No.	SH7044 (high-speed A/D converter)	RX651 (S12ADFa)
1	Select single mode	Single scan mode (only 1 channel specified)
2	Select scan mode	Continuous scan mode (only 1 channel specified)
3	Group single mode	Single scan mode (multiple channels specified)
4	Group scan mode	Continuous scan mode (multiple channels specified)

2.11.5 Interrupt

On the RX651 the S12ADFa interrupts are assigned to group interrupt BL1 and to software configurable interrupt B. The group BL1 interrupt status flag (GRPBL1.ISn) is cleared automatically when the corresponding bit in the module's status register is cleared. The software configurable interrupt B status flag (PIBRk.PIRn) is not cleared automatically, but there is no effect on the generation of interrupt requests.

Refer to 1.8, Interrupt Handling for information about interrupts.

2.11.6 Module Stop

The initial state of the peripheral modules of the RX651 is stopped, due to the low power consumption function. The initial state of the A/D converter modules (S12ADFa) is also stopped. Do not fail to cancel the module stop state when making settings to these modules. Before accessing the module stop control register to cancel the module stop state, first cancel register write protection.

2.11.7 Other Differences

The 12-bit A/D converter on the RX651 has no functions equivalent to simultaneous sampling, low-power conversion mode, and buffer operation, all of which are supported by the high-speed A/D converter on the SH7044.

2.12 Compare Match Timer (CMT)

2.12.1 Comparison of Specifications

Table 2.38 Comparison of SH7044 and RX651 CMT Specifications

Item	SH7044	RX651
Clock	Each channel selectable among 4 internal clocks (ϕ/8, ϕ/32, ϕ/128, and ϕ/512)	Each channel selectable among 4 internal clocks (PCLK/8, PCLK/32, PCLK/128, and PCLK/512)
Number of units (channels)	1 unit (total 2 channels)	2 units (total 4 channels)
Interrupt sources	Support for separate compare match interrupt requests for each (CMI0 and CMI1)	Support for separate compare match interrupt requests for each (CMI0, CMI1, CMI2, and CMI3)

2.12.2 CMT Replacement

The CMT of the SH7044 and the CMT of the RX651 are software compatible. However, the compare match timer control and status registers (CMCSR0 and CMCSR1) on the RX651 do not contain interrupt flags, so it is necessary to use the interrupt controller’s interrupt flags instead. In addition, it is not necessary to clear the flags in the compare match interrupt handler. (The interrupt controller automatically clears the associated flag when an interrupt is accepted.) A comparison of the compare match timer registers of the SH7044 and RX651 is shown below.

Table 2.39 List of Compare Match Timer Registers

Register Name	SH7044	RX651	Changed
Channel	Channel 0,1	Unit 0 (channel 0,1)	◎
Compare match timer start register	CMSTR	CMSTR0	◎
Compare match timer control/status registers	CMCSR0, CMCSR1	CMT0.CMCR, CMT1.CMCR	◎*1
Compare match timer counters	CMCNT0, CMCNT1	CMT0.CMCNT, CMT1, CMCNT	◎
Compare match timer constant registers	CMCOR0, CMCOR1	CMT0.CMCOR, CMT1, CMCOR	◎
Unit 1	—	Unit 1 (channels 2, 3)	○
—		CMSTR1	○
		CMT2.CMCR, CMT3.CMCR	○*
		CMT2.CMCNT, CMT3, CMCNT	○
		CMT2.CMCOR, CMT3, CMCOR	○

◎: Registers with identical bit assignments on the SH7044 and RX651
○: Unit 1 registers. The bit assignments are the same as for unit 0.
Note: * These registers so not contain interrupt flags. Use the IR bits of the interrupt controller instead.

2.12.3 Module Stop

The initial state of the peripheral modules of the RX651 is stopped, due to the low power consumption function. The initial state of the CMT is also stopped. Do not fail to cancel the module stop state when making settings to the module. Before accessing the module stop control register to cancel the module stop state, first cancel register write protection.

2.13 Flash Memory

2.13.1 Comparison of Specifications

Table 2.40 Comparison of Flash Memory Specifications on SH7044 and RX651

Item	SH7044	RX651
Size	• 256 KB	• ROM area User area: Max. 2 MB
Block size × block count	• 1 KB × 4 (4 KB) • 28 KB × 1 (28 KB) • 32 KB × 7 (224 KB)	• Linear mode • Blocks 0 to 7 8K bytes x 8 (64K bytes) • Blocks 8 to 69 32K x 61 (1952K bytes) • Dual mode • Blocks 0 to 7 8K bytes x 8 (64K bytes) • Blocks 8 to 37 32K x 29 (464K bytes) • Blocks 38 to 45 8K bytes x 8 (64K bytes) • Blocks 46 to 75 32K x 29 (464K bytes)
Operating modes	• Program mode • Erase mode • Program verify mode • Erase verify mode	Flash sequencer (FCU) Control the FCU according to the FACI command. • FACI command program Block erase Multi-block erase P / E suspend P / E resume Clear status forced termination Blank check Configuration settings
Write and erase units	• Write: 32-byte units • Erase: Block units	• Write — Code Flash memory program Unit: 128 bytes — Data flash memory program Unit: 4 bytes • Erase — Erase unit: 1 block
Write count	100 times	1,000 times
Programming modes	• On-board programming — Boot mode — User programming mode • Writer mode	• On-board programming — Boot mode (SCI/USB/FINE) • Single-chip mode
Other	• Automatic bit rate adjustment • RAM-based flash memory emulation function • Protect mode	• Automatic bit rate adjustment • Suspend/resume function • ROM code protection function • Supports background operations

Note: P / E: Program / Erase

When rewriting the on-chip Flash with RX651, "Flash Module Firmware Integration Technology" can be used.

The Flash Module Firmware Integration Technology makes it easy to implement rewriting of the RX651's on-chip See the following application note below for how to use it and how to embed it in application.:

Flash module Firmware Integration Technology (R01AN2184EJ)

2.14 Low Power Consumption Function

2.14.1 Comparison of Mode Specifications

The low-power modes on the SH7044 are sleep mode and standby mode. The states of the clock, CPU, and on-chip modules in each mode are listed below:

Table 2.41 SH7044 Low-Power Modes

Item	Clock	CPU	On-Chip Modules
Sleep mode	Operating	Stopped	Operating
Standby mode	Stopped	Stopped	Stopped

The low-power modes on the RX651 are sleep mode, all-module clock stop mode, software standby mode, and deep software standby mode. Table 2.38 lists the states of the on-chip modules in each mode.

Table 2.42 RX651 Low-Power Modes

Entering and Exiting Low Power Consumption Modes and Operating States	Sleep Mode	All-Module Clock Stop Mode	Software Standby Mode	Deep Software Standby Mode
Transition condition	**Control register + instruction**	**Control register + instruction**	**Control register + instruction**	**Control register + instruction**
Method of release other than reset	**Interrupt**	**Interrupt*1**	**Interrupt*2**	**Interrupt*3**
State after release*4	**Program execution state (interrupt processing)**	**Program execution state (interrupt processing)**	**Program execution state (interrupt processing)**	**Program execution state (interrupt processing)**
Main clock oscillator	Operation possible	Operation possible	Operation possible *5	Operation possible *5
Sub-clock oscillator	Operation possible	Operation possible	Operation possible *6	Operation possible *6
High-speed on-chip oscillator	Operation possible	Operation possible	Stopped	Stopped
Low-speed on-chip oscillator	Operation possible	Operation possible	Stopped	Stopped
IWDT dedicated on-chip oscillator	Operation possible *7	Operation possible *7	Operation possible *7	Stopped (settings undetermined) *7
PLL	Operation possible	Operation possible	Stopped	Stopped
CPU	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings undetermined)
RAM and expansion RAM	Operation possible (settings retained)	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings undetermined)
Standby RAM	Operation possible (settings retained)	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings retained/ undetermined) *8
Flash memory	Operating	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings retained)
USBFS host/function module (USBb)	Operation possible	Stopped *9	Stopped *9	Stopped (settings retained/ undetermined) *10
Watchdog timer (WDT A)	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings undetermined)
Independent watchdog timer (IWDT)	Operation possible *7	Operation possible *7	Operation possible *7	Stopped (settings undetermined) *7
Realtime clock (RTC)	Operation possible	Operation possible	Operation possible	Operation possible
8-bit timer (unit 0, unit 1) (TMR)	Operation possible	Operation possible*11	Stopped (settings retained)	Stopped (settings undetermined)
Port Output Enable (POE)	Operating possible	Operating possible*12	Stopped (settings retained)	Stopped (settings undetermined)
Voltage detection circuit (LVDA)	Operation possible	Operation possible	Operation possible	Operation possible *13, *14
Power-on reset circuit	Operating	Operating	Operating	Operating *14
Peripheral modules	Operation possible	Stopped (settings retained)	Stopped (settings retained)	Stopped (settings undetermined)
I/O ports	Operating	Settings retained *15	Settings retained *16	Settings retained *16

“Operating possible” means that operating or stopped can be controlled by the control register setting.

“Stopped (Retained)” means that internal register values are retained and internal operations are suspended.

“Stopped (Undefined)” means that internal register values are undefined and power is not supplied to the internal circuit.

Note 1. “Interrupts” here indicates an external pin interrupt (the NMI or IRQ0 to IRQ15) or any of peripheral interrupts (the 8-bit timer, RTC alarm, RTC periodic, IWDT, USB suspend/resume, voltage monitoring 1, voltage monitoring 2, and main-clock oscillation stop detection).

Note 2. “Interrupts” here indicates an external pin interrupt (the NMI or IRQ0 to IRQ15) or any of peripheral interrupts (the RTC alarm, RTC periodic, IWDT, USB suspend/resume, voltage monitoring 1, and voltage monitoring 2 interrupts).

Note 3. “Interrupts” here indicates a certain external pin interrupt source pin (the NMI, IRQ0-DS to IRQ15-DS, SCL2-DS, SDA2-DS, or CRX1-DS) or any of peripheral interrupts (the RTC alarm, RTC periodic, USB suspend/resume, voltage monitoring 1, and voltage monitoring 2 interrupts). However, these interrupts are enabled only when the corresponding bit in the deep standby interrupt enable registers i (DPSIERi) (i = 0 to 3) is set to 1. When the pin functions have “-DS” appended to their names, they can also be used as triggers for release from deep software standby. Also, USBb is not released from deep software standby mode using USB0_OVRCURB multiplexed with pin P22.

Note 4. This does not include release initiated by the RES# pin reset, power-on reset, voltage monitoring reset, or independent watchdog-timer reset. The transition is to the reset state when release is initiated by one of these reset sources.

Note 5. Operation or stopping can be selected by the main clock oscillator forced oscillation bit (MOFXIN) in the main clock oscillator forced oscillation control register (MOFCR).

Note 6. Operation or stopping is selected by the sub-clock oscillator control bit (RTCEN) in the RTC control register 3 (RCR3).

Note 7. Operation or stopping is selected by the setting of the IWDT sleep mode count stop control bit (IWDTSLCSTP) in the option function select register 0 (OFS0) in IWDT auto start mode. If the OFS0.IWDTSLCSTP bit is 0 (disabling stopping of the counter when a transition to low power consumption mode is made), the transition is to software standby mode rather than deep software standby mode. In any mode other than IWDT auto start mode, operation or stopping is selected by the setting of the sleep mode counter stop control bit (SLCSTP) in the IWDT counter stop control register (IWDTCSTPR). If the IWDTCSTPR.SLCSTP bit is 0 (disabling stopping of the counter when a transition to low power consumption mode is made), the transition is to software standby mode rather than deep software standby mode.

Note 8. Retention or undefined is selectable by the setting of the deep cut bits in the deep standby control register (DPSBYCR.DEEPCUT[1:0]).

Note 9. Detection of USB resumption is possible.

Note 10. Disabling or enabling of detection of USB resumption is controllable by the deep cut bits in the deep standby control register (DPSBYCR.DEEPCUT[1:0]). When detection of USB resumption is enabled, the values of the registers in the USB resume detecting unit are only held even in deep software standby mode.
Also, USBb is not released from deep software standby mode using USB0_OVRCURB multiplexed with pin P22.

Note 11. Stopping or operation is controlled by the module-stop setting bits (MSTPA4 and MSTPA5, respectively) in module-stop control register A (MSTPCRA) for 8-bit timers 0 and 1 (unit 0) and 2 and 3 (unit 1).

Note 12. When a source condition for POE interrupts is satisfied while POE interrupts are enabled and the chip is in all-module clock stop
mode, the flag for the source condition is retained but return from all-module clock stop mode does not proceed. If a different source initiates return from all-module clock stop mode in this situation, the POE interrupt is generated after that.

Note 13. If the voltage monitoring 1 circuit mode selection bit in the voltage monitoring 1 circuit control register 0 (LVD1CR0.LVD1RI) or the voltage monitoring 2 circuit mode selection bit in the voltage monitoring 2 circuit control register 0 (LVD2CR0.LVD2RI) is 1, the transition is to software standby mode rather than deep software standby mode.

Note 14. When the deep cut bits in the deep standby control register (DPSBYCR.DEEPCUT[1:0]) are set to 11b and the LSI enters deep
software standby mode, the voltage detection circuit stops and the low power consumption function of the power-on reset circuit is enabled.

Note 15. If pin P53 is being used for the BCLK signal, operation continues with as-is output of BCLK. While the 8-bit timer and RTC are operated, the related pins continue operation.

Note 16. Retention of levels or placement in the high-impedance state is selectable for the address bus and bus control signals (CS0# to
CS7#, RD#, WR0# to WR3#, WR#, BC0# to BC3#, ALE, CKE, SDCS#, RAS#, CAS#, WE#, and DQM0 to DQM3) by the output port enable bit (OPE) in the standby control register (SBYCR).

2.14.2 Mode Transitions

The transition diagram between each mode of RX651 is shown in Figure 2.15 “RX651 Mode Transition Diagram”.

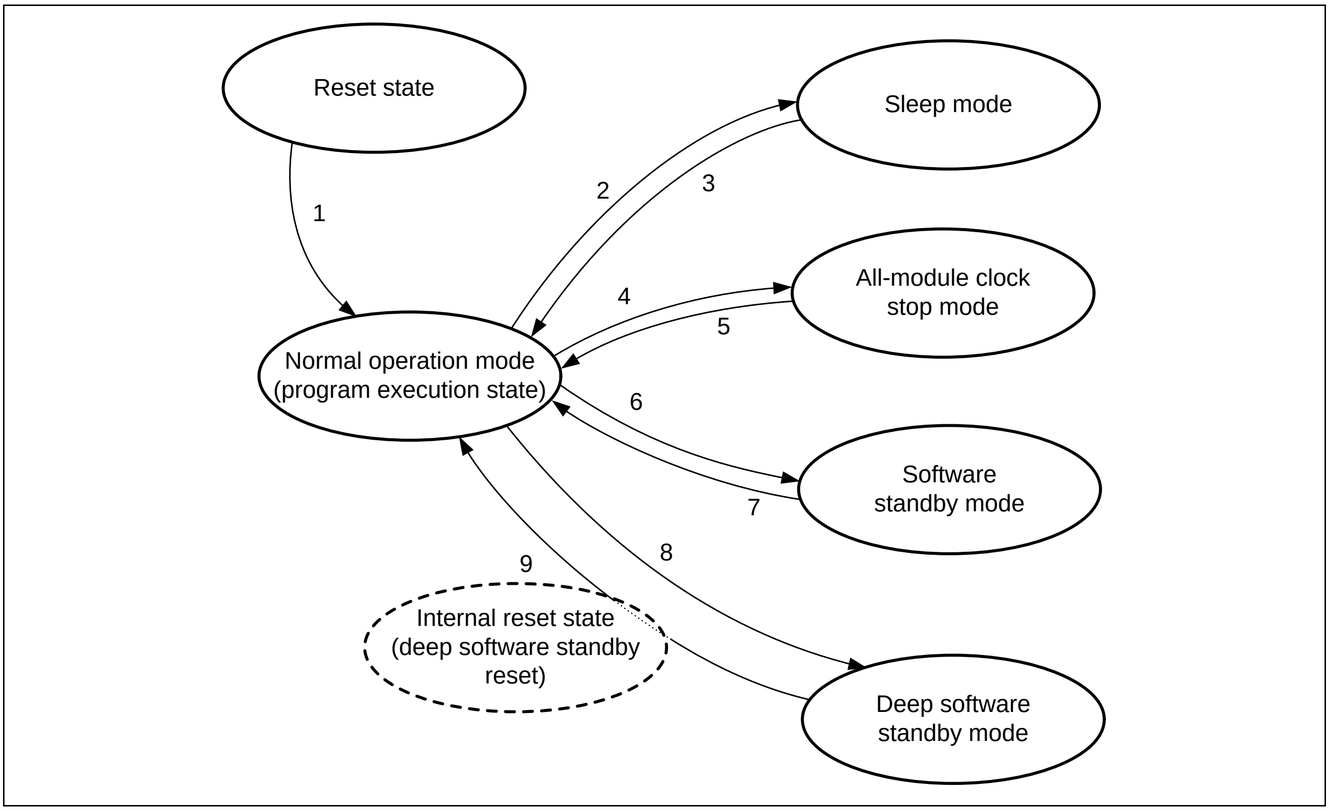

Reset state
Sleep mode
All-module clock stop mode
Normal operation mode (program execution state)
Software standby mode
Internal reset state (deep software standby reset)
Deep software standby mode
1
2
3
4
5
6
7
8
9

Figure 2.16 RX651 Mode Transitions

The events and transition conditions shown in figure 2.30 are listed below:

Table 2.43 List of RX651 Mode Transitions and Events

No.	Event	Transition Condition (The following conditions are specified before the event.)
1	RES# pin = high	—
2	WAIT instruction executed	SBYCR.SSBY = 0
3	All interrupts	—

4	WAIT instruction executed	SBYCR.SSBY = 0 MSTPCRA.ACSE = 1 MSTPCRA = FFFF FF[C-F]Fh
		MSTPCRB = FFFF FFFFh MSTPCRC[31:16] = FFFFh
		MSTPCRD = FFFF FFFFh

5	External and peripheral interrupts	External pin interrupts (NMI, IRQ0 to IRQ15) Peripheral function interrupts (8-bit timer, RTC alarm, RTC cycle, IWDT, USB suspend/resume, voltage monitor 1, voltage monitor 2, oscillation stop detection)*
6	WAIT instruction executed	SBYCR.SSBY = 1, DPSBYCR.DPSBY = 0
7	External and peripheral interrupts	External pin interrupts (NMI, IRQ0 to IRQ15) Peripheral function interrupts (RTC alarm, RTC cycle, IWDT, USB suspend/resume, voltage monitor 1, voltage monitor 2)*
8	WAIT instruction executed	SBYCR.SSBY = 1, DPSBYCR.DPSBY = 1
9	External and peripheral interrupts	Some pins used as external pin interrupt sources (NMI, IRQ0-DS to IRQ15-DS, SCL2-DS, SDA2-DS, CRX1-DS), peripheral function interrupts (RTC alarm, RTC cycle, USB suspend/resume, voltage monitor 1, voltage monitor 2)* After one of the above interrupts occurs the internal reset state lasts for a specified duration, after which the internal reset and deep software standby mode are canceled at the same time, and the CPU operates in normal operation mode using the LOCO (recovery after a reset).

Note: * Each interrupt has detailed conditions. For descriptions, see the User's Manual: Hardware.

3. Sample Code

For the settings of each function, use the smart configurator to set according to the application and generate the code.

4. Reference Documents

4.1 Reference Documents

Section 4.1 lists the documents referenced in the preparation of this application note. When referring to the documents listed below, substitute the latest version if a newer version is available. The latest versions of these documents can be confirmed and downloaded from the Renesas Electronics Website.

Table 4.1 Reference Documents

Reference Documents
SH7040 Series User's Manual Hardware (REJ09B0031-0600H)
SH-1/SH-2/SH-DSP Software Manual （REJ09B0228-0700）
RX65N Group, RX651 Group User’s Manual: Hardware (R01UH0590EJ0230)
RX Family User’s Manual: Software (R01US0071EJ)
Flash Memory User's Manual Hardware Interface (R01UH0602EJ)
Flash Module Firmware Integration Technology (R01AN2184EJ)
SH7040 Series: On-Chip I/O (ADJ-502-052A)

Website and Support

Renesas Electronics Website

https://www.renesas.com/jp/en

Inquiries
Contact Us | Renesas

Revision History

		Description	
Rev.	Date	Page	Summary
1.00	Sep 22, 2022	—	First edition issued

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 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 VIL (Max.) and VIH (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 VIL (Max.) and VIH (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 systemevaluation 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

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/.

Trademarks

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