RA Family

Hardware Manual Guide (peripheral functions)

Introduction

This manual is an easy-to-understand summary of each peripheral functional chapter of the hardware manual, using the RA4L1 and RA6M5 as examples.The purpose of this document is to provide a much deeper understanding of the functions by using it with the hardware manual.For more detailed usage instructions, please refer to the application notes provided on each page.This document does not cover all the information described in the manual. For detailed information such as precautions for each function, please check the hardware manual for the product you want to use.

Please also refer to the Hardware Manual Guide: Electrical Characteristics(R01TU0457EJ)

Target Device

RA Family

Contents

1. Low power consumption

1.1 Low Power Consumption function by product group

The Low Power Consumption functions for each product are shown in the table below.

Product Group	Low power modes				Power control modes			
	Sleep	Software Standby	Deep Software Standby	Snooze	High-speed	Middle-speed	Low-speed	Subosc-speed
RA4L1	✓	✓	-	✓	✓	✓	✓	✓
RA6M5	✓	✓	✓	✓	✓	-	✓	✓

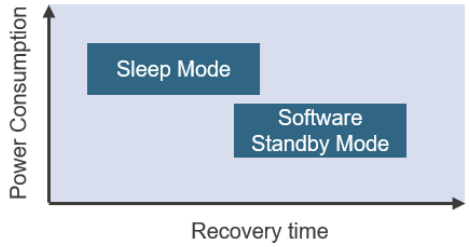
Some products may not support certain modes, so please refer to the hardware section of the user's manual for each product for details.

1.1.1 Overvies of low Power Consumption function in the RA4L1 GROUP

The following summarize the low power consumption funtion featuers of the RA4L1 group

"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.

Low power consumption mode	CPU	Oscillator / Power management				Peripheral modules						Memory		I/O Port
		Main clock	HOCO MOCO PLL	LOCO Sub-clock RTC LVD	POR	IWDT I3C AGT UARTA ACMPLP SLCDC IRQ NMI	WDT	USBFS	IIC0	ADC12 DAC12 CTSU DTC SCI0 DOC ELC	Other	SRAM	Flash memory	
Sleep	Stop (retained)	Selectable	Selectable	Selectable	Operating	Selectable	Selectable	Selectable	Selectable	Selectable	Selectable	Selectable	Operating	Operating
Software Standby	Stop (retained)	Stop	Stop	Selectable	Operating	Selectable	Stop (retained)	Stop (retained) *2	Selectable	Stop (retained)	Stop (retained)	Stop (retained)	Stop (retained)	Stop (retained)
Snooze*1	Stop (retained)	Selectable	Selectable	Selectable	Operating	Selectable	Stop (retained)	Operation prohibited *2	Selectable *3	Selectable	Operation prohibited	Selectable	Stop (retained)	Operating



*1 : For details of snooze mode, see this page.

*2 : USB resumption detection is possible.

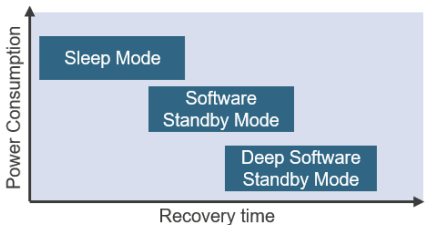
*3 : Wake-up interrupt operation only.

1.1.2 Overvies of low Power Consumption function in the RA6M5 GROUP

The following summarize the low power consumption funtion featuers of the RA6M5 group

"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.

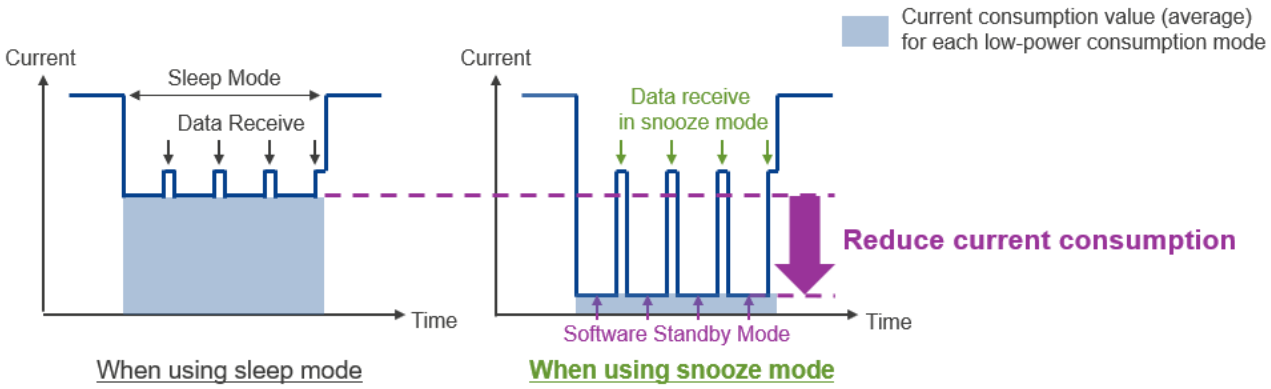
Low power consumption mode	CPU	Oscillator / Power management				Peripheral modules						Memory			I/O Port
		Main-clock HOCO MOCO PLL PLL2	LOCO Sub-clock RTC LVD AGT	POR	EBCLK	IWDT IIC0 IRQ	NMI IRQ-DS	WDT	ADC12 DAC12 CTSU SCI0 DTC DOC ELC	USBFS USBHS	Other	SRAM	Standby RAM	Flash memory	
Sleep	Stop (retained)	Selectable	Selectable	Operating	Selectable	Selectable	Selectable	Selectable	Selectable	Selectable	Selectable	Selectable	Selectable	Operating	Operating
Software Standby	Stop (retained)	Stop	Selectable	Operating	Stop (retained)	Selectable	Selectable	Stop (retained)	Stop (retained)	Stop (retained)	Stop (retained)	Stop (retained)	Stop (retained)	Stop (retained)	Stop (retained)
Snooze*1	Stop (retained)	Selectable	Selectable	Operating	Operation prohibited	Selectable *2	Selectable	Stop (retained)	Selectable	Operation prohibited	Operation prohibited	Selectable	Selectable	Stop (retained)	Operating *3
Deep Software Standby	Stop (Undefined)	Stop	Selectable *4	Operating	Stop (retained)	Stop (Undefined)	Selectable	Stop (Undefined)	Stop (Undefined)	Stop (retained/ Undefined)	Stop (Undefined)	Stop (retained)	Stop (retained/ Undefined)	Stop (retained)	Stop (retained)



*1 : For details of snooze mode, see this page.
*2 : For the IIC0,Wake-up interrupt operation only possible.
*3 : Stop Only EBCLK pin:(retained)
*4 : Stop Only AGT4~5:(Undefined)

1.2 Snooze mode

During software-standby mode, some peripheral functions can operate without wakening the CPU (intermittent operation). By performing intermittent operation in software standby and snooze mode, the total current consumption can be reduced from mA to uA level compared to sleep mode operation.



1.3 High-speed, Middle-speed, Low-speed, OSC mode

By selecting the appropriate operating power control mode according to the operating frequency, power consumption can be reduced in normal, sleep, and snooze modes. In each mode, it will operate within the following operating voltages and operating frequency ranges

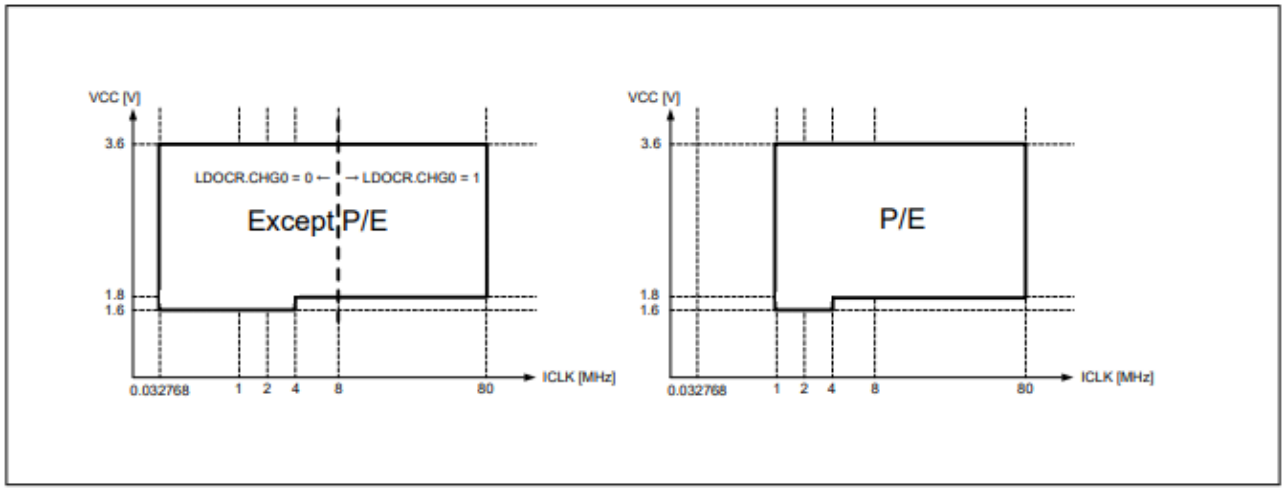


Figure 10.2 Operating voltages and frequencies in High-speed mode

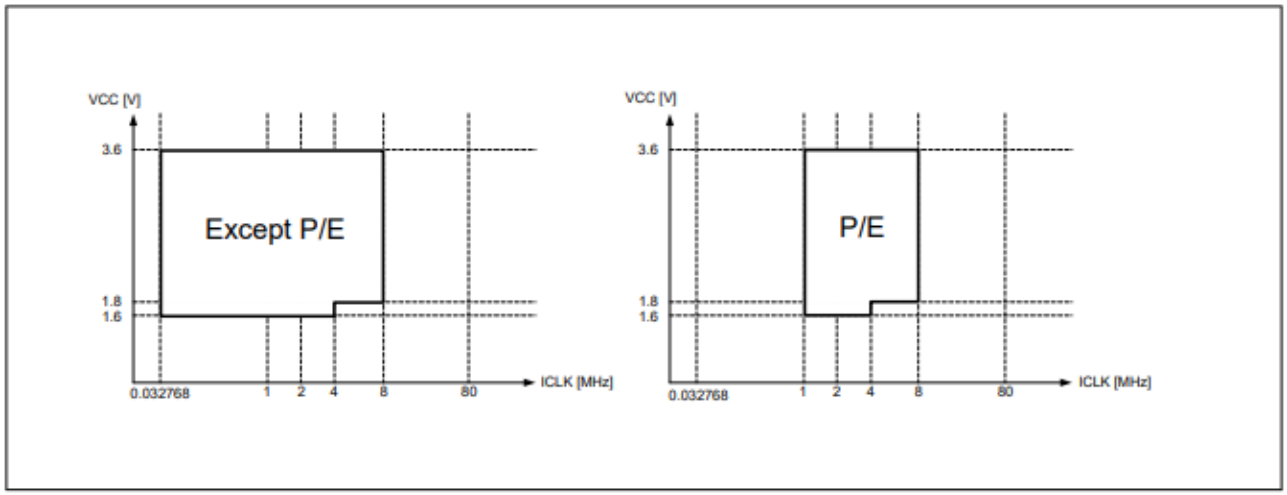


Figure 10.3 Operating voltages and frequencies in Middle-speed mode

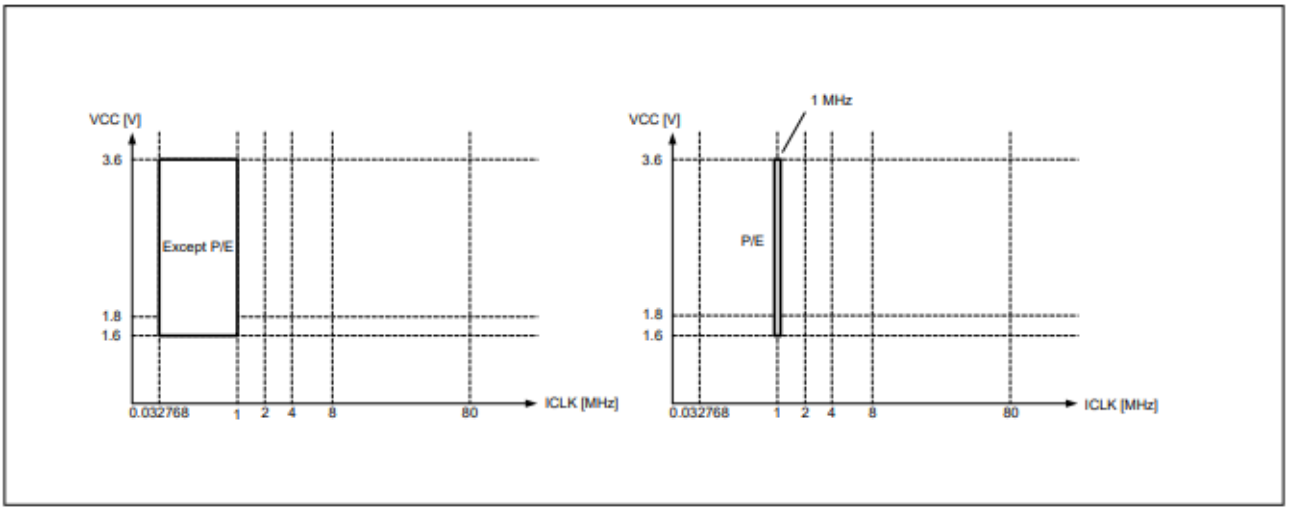


Figure 10.4 Operating voltages and frequencies in Low-speed mode

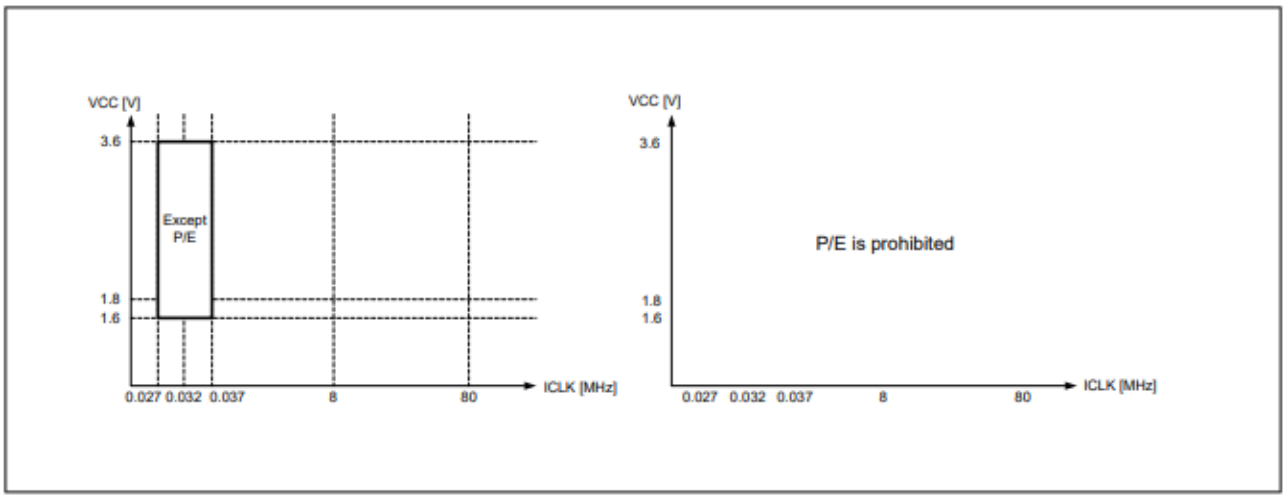


Figure 10.5 Operating voltages and frequencies in Subosc-speed mode

1.4 Tips for reducing power consumption

Introduce tips for reducing power consumption using the RA4L1 as an example. The following information depends on the product you are using. For details, please refer to the User's Manual: Hardware of each product.

(1) Utilize High-speed / Middle-speed / Low-speed / Subosc-speed operation mode as appropriate

Power consumption can be further reduced by lowering the operating frequency of the system when entering sleep or deep sleep mode.

Example) Comparison of current consumption in sleep mode and operation of all peripheral modules (typical)

	High-speed	Middle-speed	Low-speed	Subosc-speed
Current consumpion	22.8mA	3.4mA	0.94mA	15uA

(2) Stop unused peripheral modules

There is a large difference in current consumption when "all modules are operated" or "all modules are stopped". Therefore, power consumption can be reduced by stopping unused peripheral modules.

Example) Comparison of current consumption of peripheral module operation/stop in High-speed sleep mode (typical)

	All peripheral operation	Stop all peripherals
Current consumption	22.8mA	5.63mA

(3) I/O Port

The current consumption of the I/O port for which the output is selected can be reduced by the following process. (If it is connected to a pull-up resistor, set the output level to High, and if it is connected to a pull-down resistor, set the output level to Low.)

(4) Clock generation circuit

When using low power mode, observe the following clock frequency settings: Here are some examples of the highest operating frequency and the circumferential ratio. For more information, refer to the chapter on Clock Generation Circuit in the user manual.

Table 8.2 Clock generation circuit specifications for the internal clocks (1 of 3)

Item	Clock source	Clock supply	Specification
System clock (ICLK)	MOSC/SOSC/HOCO/MOCO/ LOCO/PLL	CPU, DTC, DMAC, Flash, RAM, I/O ports	Up to 80 MHz Division ratio: 1/2/4/8/16/32/64
Peripheral module clock A (PCLKA)	MOSC/SOSC/HOCO/MOCO/ LOCO/PLL	Peripheral modules (QSPI, SCI, SPI, CRC, DOC, ADC12, DAC12, GPT bus clock, CNECC, IrDA, I3C)	Up to 80 MHz Division ratio: 1/2/4/8/16/32/64

Note: Restrictions on setting clock frequency: ICLK ≥ PCLKA ≥ PCLKB, PCLKD ≥ PCLKA ≥ PCLKB
ICLK ≥ FCLK
Restrictions on clock frequency ratio: (N: integer, and up to 64)
ICLK : FCLK = 1:1 when ICLK ≤ 48 MHz, ICLK : FCLK = 2:1 when 48 MHz < ICLK ≤ 80 MHz
ICLK : PCLKA = N:1, ICLK : PCLKB = N:1, ICLK : PCLKC = N:1 or 1:N , ICLK : PCLKD = N:1 or 1:N
If the A/D converter is enabled, the clock frequency ratio is constrained as follows:
PCLKA : PCLKC = 1:1 or 2:1 or 4:1 or 8:1 or 1:2 or 1:4

(5) Changing the Regulator (LDO) Operating Mode

The LDO has normal operating mode and high-performance mode. Power consumption can be minimized by operating in the normal mode.

(6) Sub-clock oscillator drive capability switching function

This function is implemented in some products of the RA4L1 Group.
The current consumption decreases in the following order: Normal mode> Low power mode 1, >Low power mode 2, > Low power mode 3. In addition, to achieve low-power consumption modes 1, 2, and 3, the external crystal must be a low-CL product. Therefore, when requesting a matching evaluation, please inform the oscillator manufacturer of the above requirements, or refer to the application note below to perform the matching evaluation.

■Application Note
Renesas RX and RA families Design Guide for Main Clock Circuits and Sub Clock Circuits R01AN7202EJ

1.5 Application notes for Low Power Consumption function

Please refer to the following APN for details on how to use the Low Power Consumption function.

Application note name	Document No.
RA2L1 Group Capacitive Touch Low Power Guide	R01AN6266EJ
RA6M2 Group Capacitive Touch Low Power Guide	R11AN6473EJ
RA2E3 Group RA2E3 HS4001 Low Power Sensor System Example	R01AN7744EJ
Renesas RA0 Family Low Power Consumption Guide	R01AN7893EJ
RA Family Examples of Transitioning to Low Power Modes	R01AN8013EJ

2. I/O port

2.1 List of port functions

“Input pull-up function”, “Open-drain output function”,” Drive capacity switching function”, “5V tolerant setting” are also valid for other signals (such as serial and other peripheral functions) that share pins with general-purpose I/O ports.

Table 19.2 I/O port functions (1 of 2)

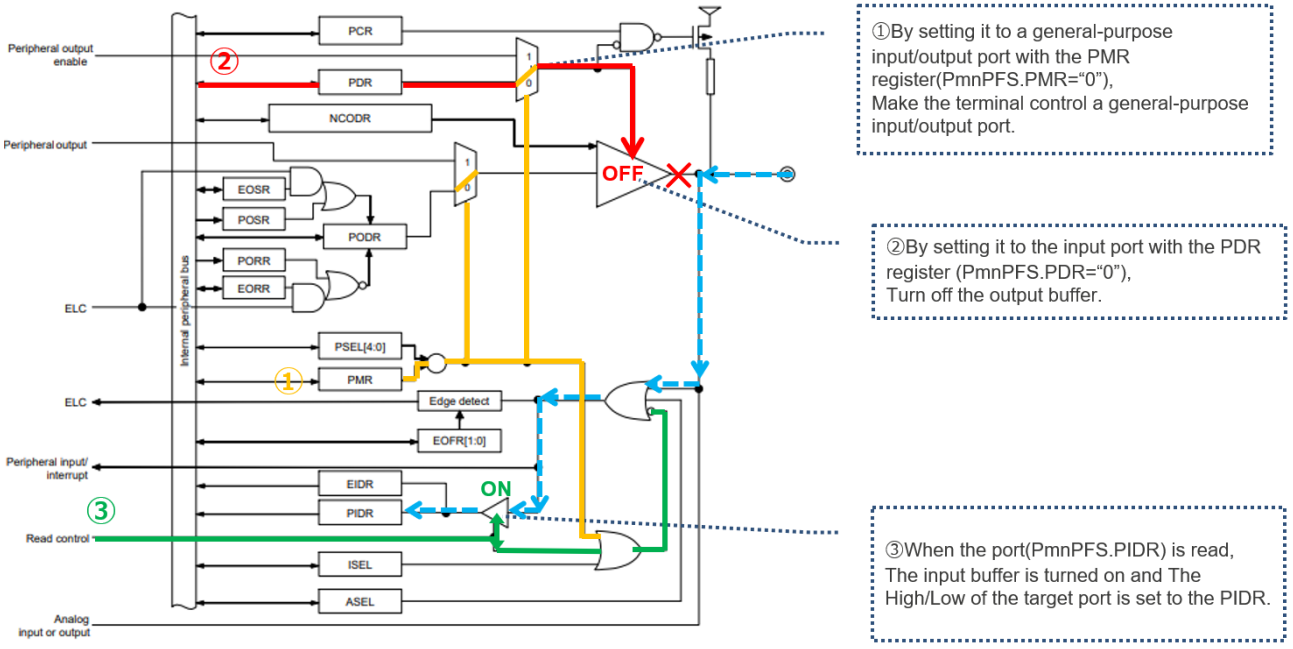
Port	Port name	Input pull-up	Open-drain output	Drive capacity switching	5V tolerant	I/O
PORT0	P000 to P010, P014, P015	✓	✓	Low	—	Input / Output
PORT1	P100 to P107	✓	✓	Low, middle, high, high speed high drive	—	Input / Output
	P108 to P115	✓	✓	Low, middle, high	—	Input / Output
PORT2	P200	✓	—	—	—	Input
	P201	✓	✓	Low	—	Input / Output
	P202 to P204, P207, P212, P213	✓	✓	Low, middle, high	—	Input / Output
	P208 to P211, P214	✓	✓	Low, middle, high, high speed high drive	—	Input / Output
	P205, P206	✓	✓	Low, middle, high	✓	Input / Output
PORT3	P300 to P315	✓	✓	Low, middle, high	—	Input / Output
PORT4	P400, P401, P407 to P415	✓	✓	Low, middle, high	✓	Input / Output
	P402 to P406	✓	✓	Low, middle, high	—	Input / Output
PORT5	P500 to P508, P513	✓	✓	Low, middle, high	—	Input / Output
	P511, P512	✓	✓	Low, middle, high	✓	Input / Output

*This function is described using the RA6M5 as an example.

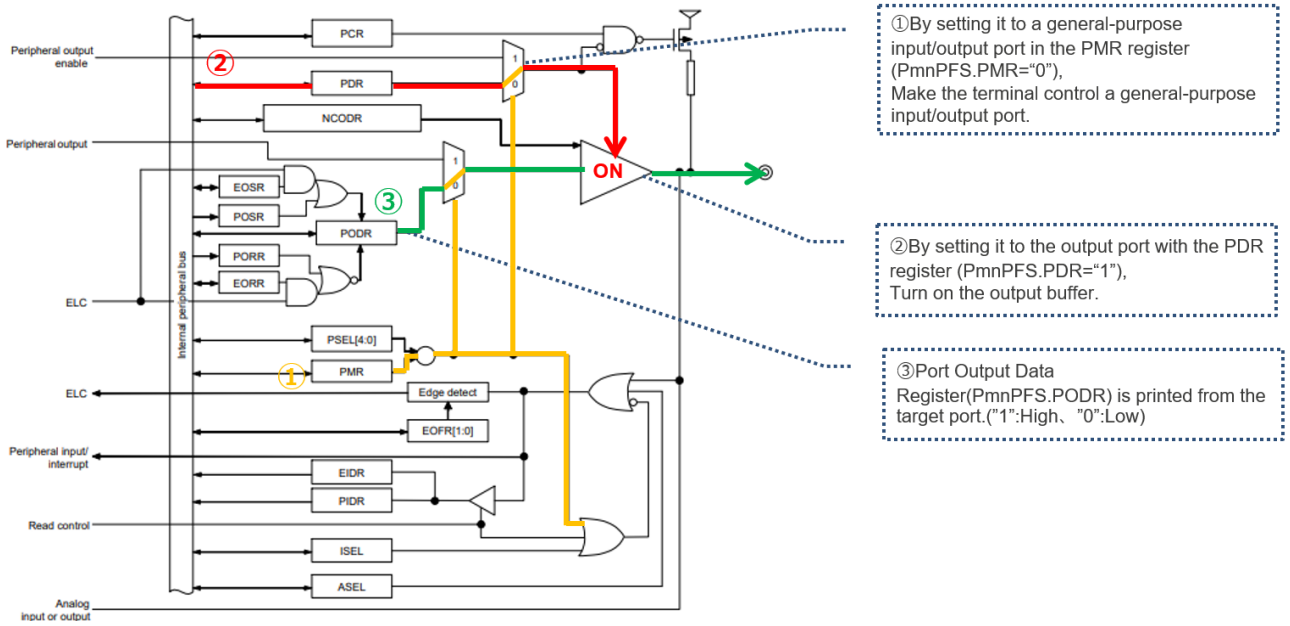
2.2 Additional information for the Input/Output port configuration page

Additional information for the Unput/Output port configuration is shown.

(1) General I/O input port

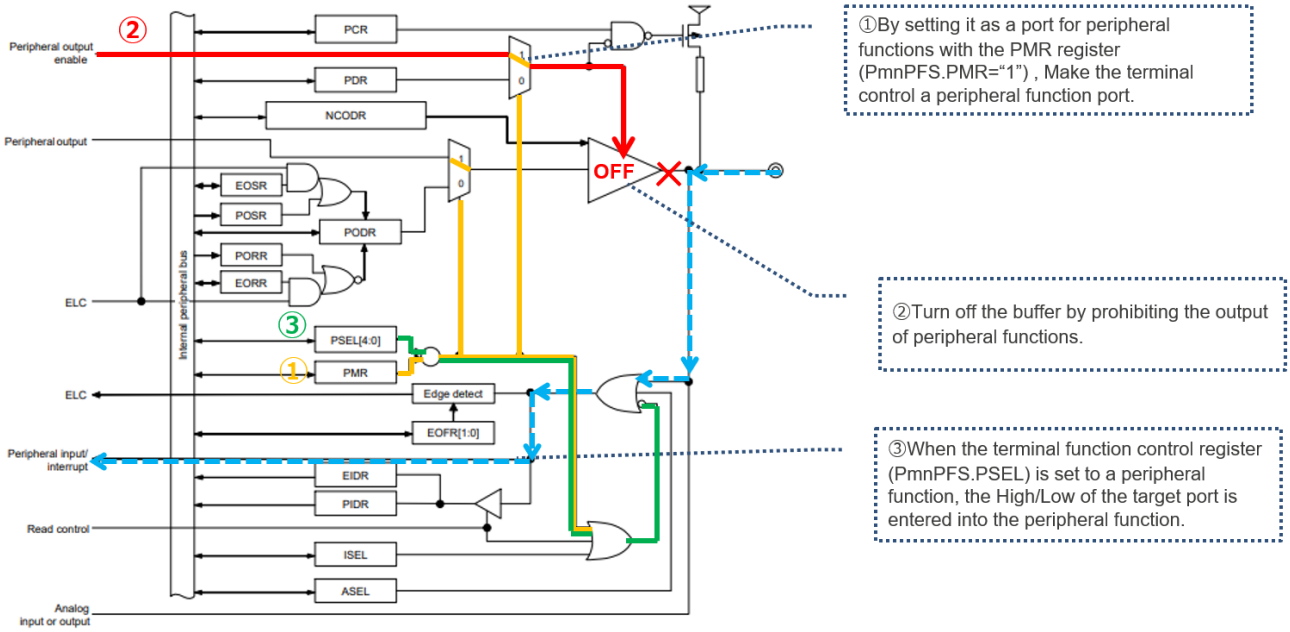


(2) General I/O output port



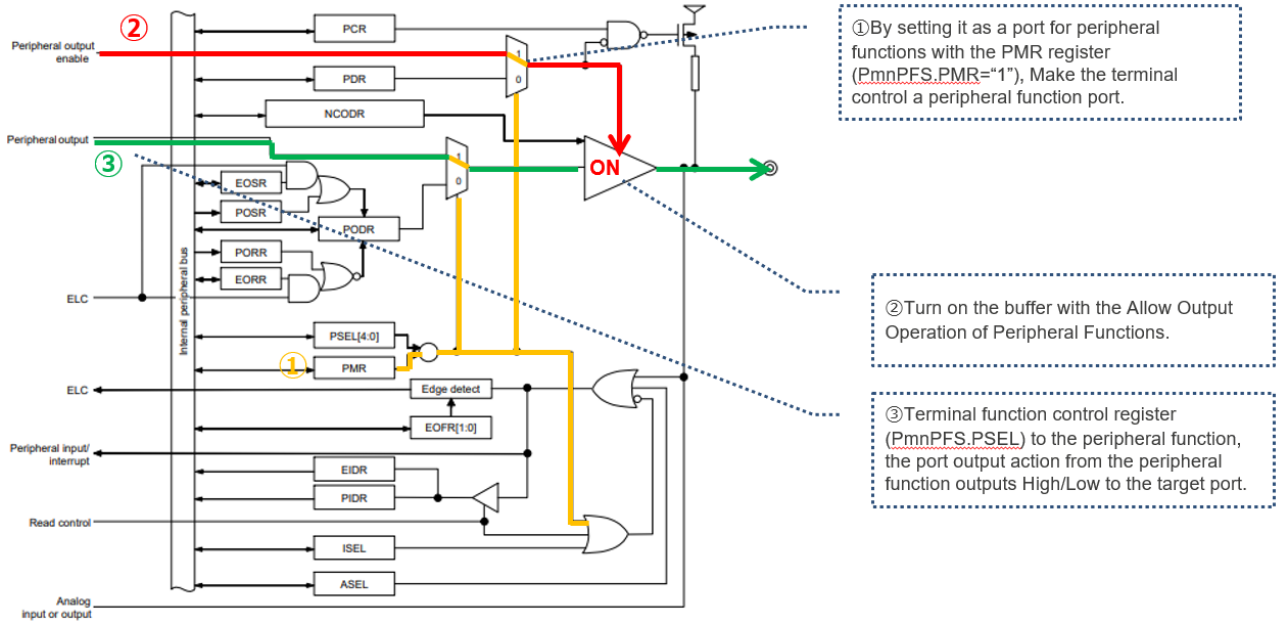
(3) Pheropheral funcion Input

When the pin function of the port is set to "Use the pin as peripheral function" and the peripheral function to be used is the input operation.

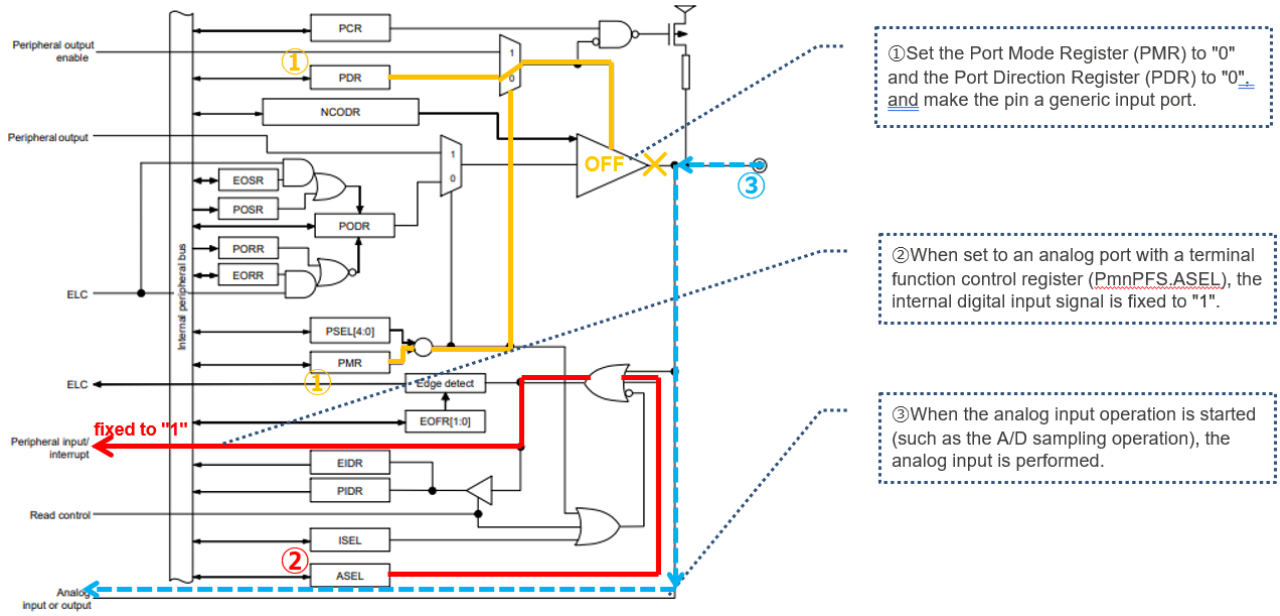


(4) Peripheral function output

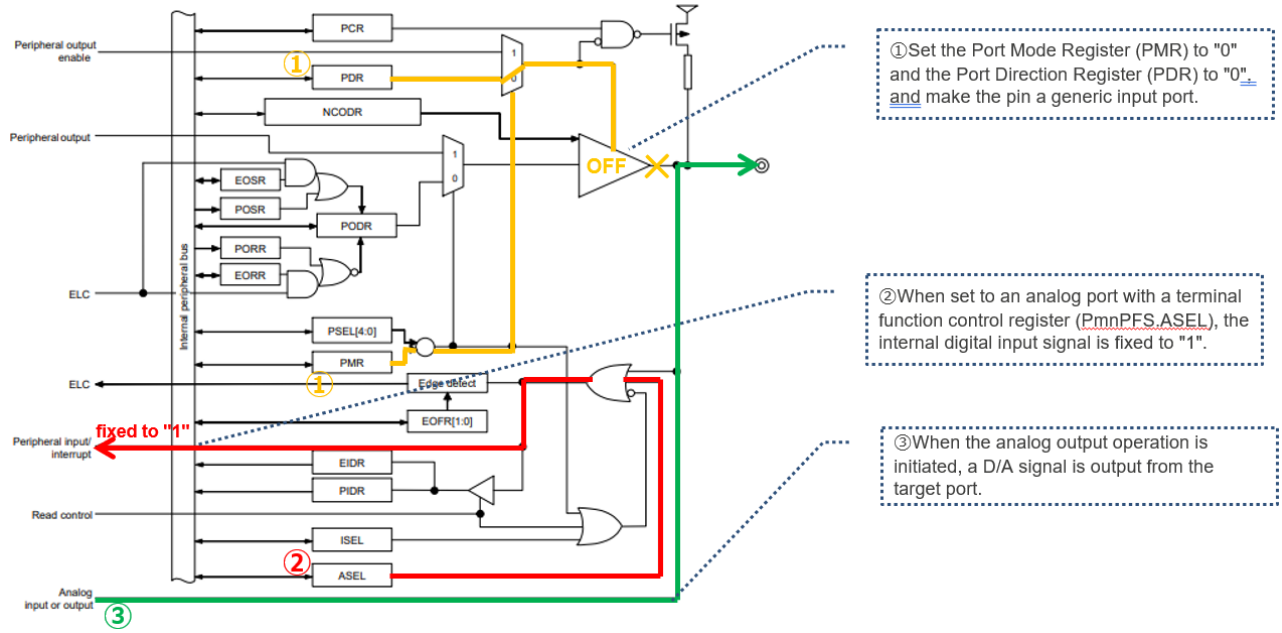
When the pin function of the port is set to "Use the pin as peripheral function" and the peripheral function to be used is the output operation.



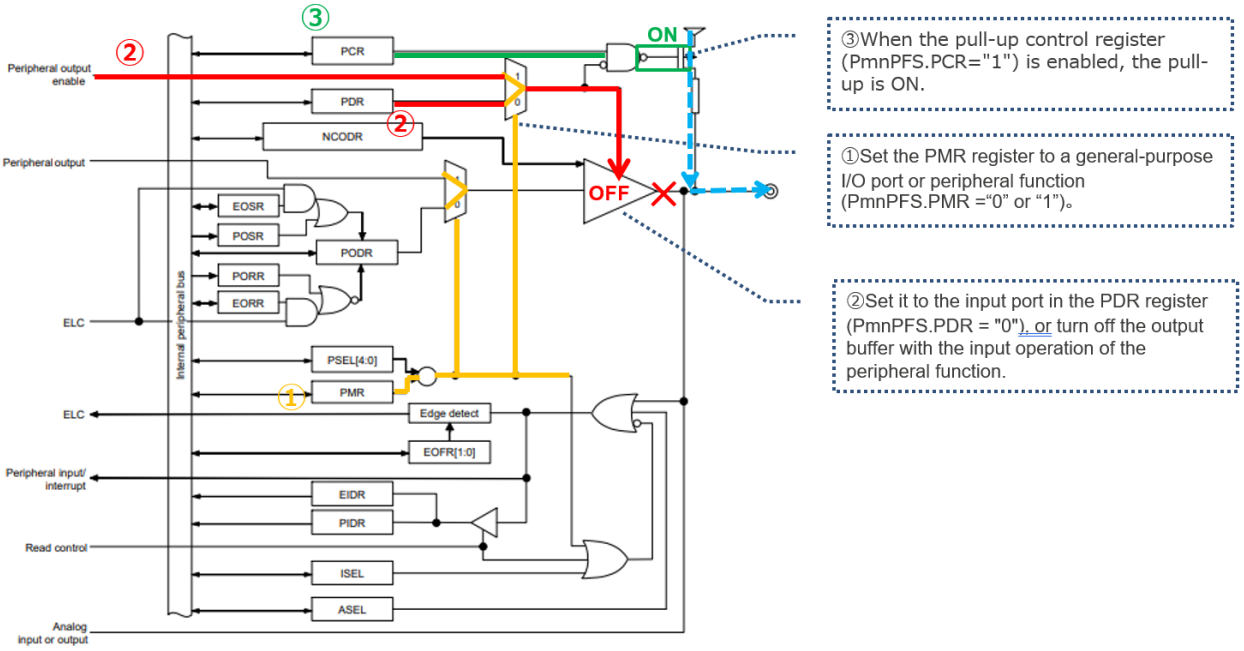
(5) analog function input



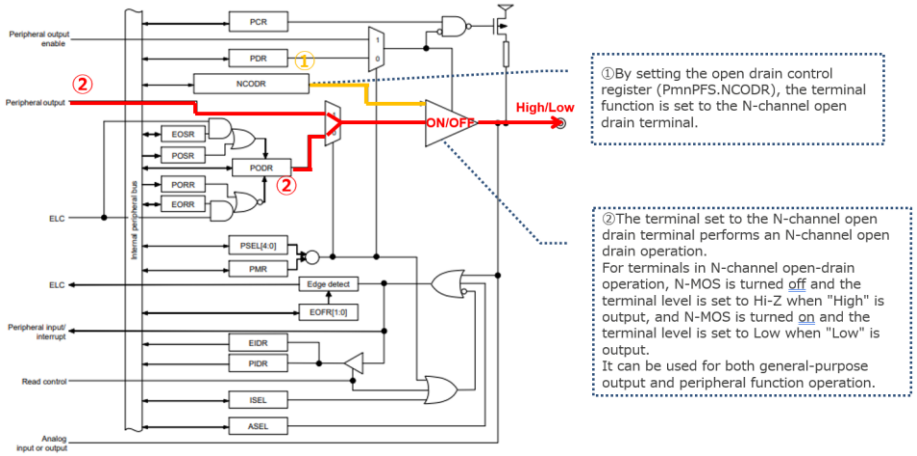
(6) analog function output



(7) Pull-up



(8) N-channel open-drain



2.3 Notice for I/O Port Settings

(1) Common

When PmnPFS.PIDR is read, the pin status is read regardless of PmnPFS.PDR and PmnPFS.PMR registers. Pull-up enables the input pull-up resistor of the pin corresponding to the bit with the PmnPFS.PCR register set to “1” when the pin is in the input state. The pull-up resistor is disabled during a reset. The input-pull-up function, open-drain output function, drive capacity switching function and 5V tolerant setting are also valid for other signals that share a port with the general-purpose I/O port.

(2) Interrupt pins

The ISEL bit is set when used as an IRQ input pin (external pin interrupt). It can also be used in combination with peripheral functions.
However, it is prohibited to enable IRQn of the same number by two or more pins.

(3) Analog port

When setting the pins as analog pins with ASEL bit, set the relevant bit of the port mode register (PmnPFS.PMR) and the relevant bit of the port direction register (PmnPFS.PDR) to “0” to set the relevant pins as general-purpose inputs, set PmnPFS.ASEL bit to “1”. The pin status cannot be read at this time.

(4) Peripheral function port

The PmnPFS.PSEL[5:0] bit should be changed with the PmnPFS.PMR bit set to “0”.

(5) Other

For ports to which RIIC and RI3C are assigned, set the PmnPFS.PCR bit to “0”.
(Pull-up is automatically turned off for peripheral function outputs other than RIIC and RI3C.)

3. Watchdog timer/Independent Watchdog Timer (WDT/IWDT)

3.1 Functional Comparison of WDT and IWDT

The functional comparison between WDT and IWDT is as follows.

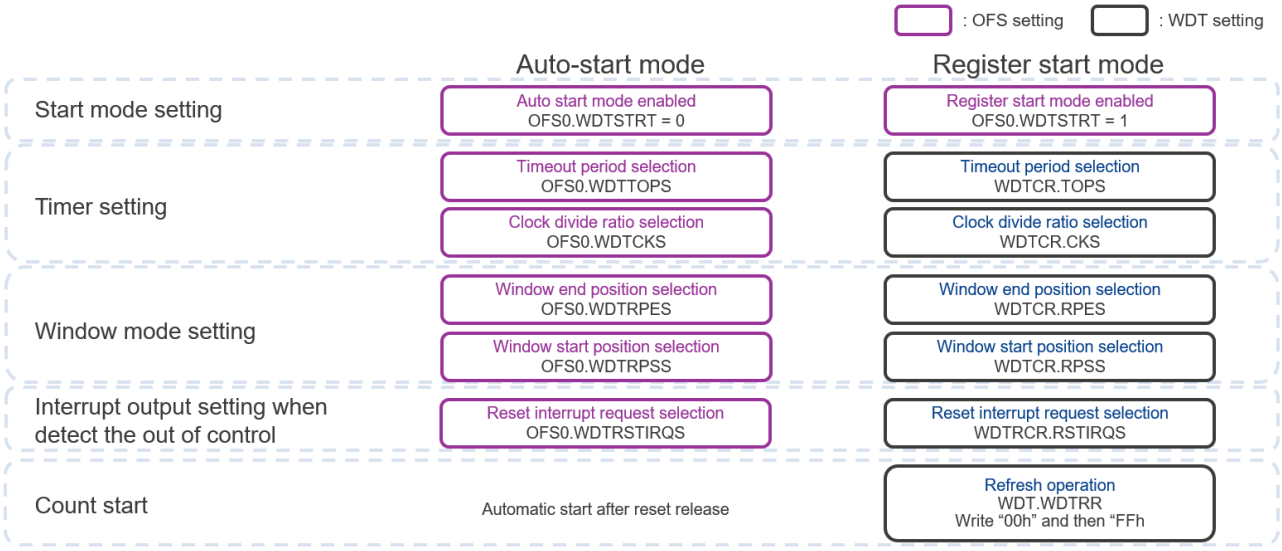
Blue text : Functional difference

Item	WDT	IWDT
Count source	Peripheral module clock	**IWDT-dedicated clock (IWDTCLK)**
Clock divide ratio	4/64/128/512/2048/8192	**1/16/32/64/128/256**
Conditions for starting the counter	・Auto-start mode: Counting automatically starts after a reset is released ・Register start mode: Counting is started by refresh operation	**・Auto-start mode : Counting automatically starts after a reset**
Conditions for stopping the counter	・Reset ・In low power consumption states(Depends on register setting) ・A counter underflows or a refresh error occurs (only in register start mode)	・Reset ・In low power consumption states(Depends on register setting)
Window function	Yes	Yes
Reset output sources	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)
Non-maskable interrupt/ interrupt sources	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)	・Down-counter underflows ・Refreshing outside the refresh-permitted period (refresh error)
Reading the counter value	Enable	Enable
Event link function	Yes	Yes
Output signal (Internal signal)	・Reset output ・Interrupt request output ・Sleep mode count stop control output	・Reset output ・Interrupt request output ・Sleep mode count stop control output
Operations during low power consumption mode	Run / Stop (Selectable) *1	Run / Stop (Selectable) *1

*1 Each low-power mode works differently. For details, see the Low power consumption page.

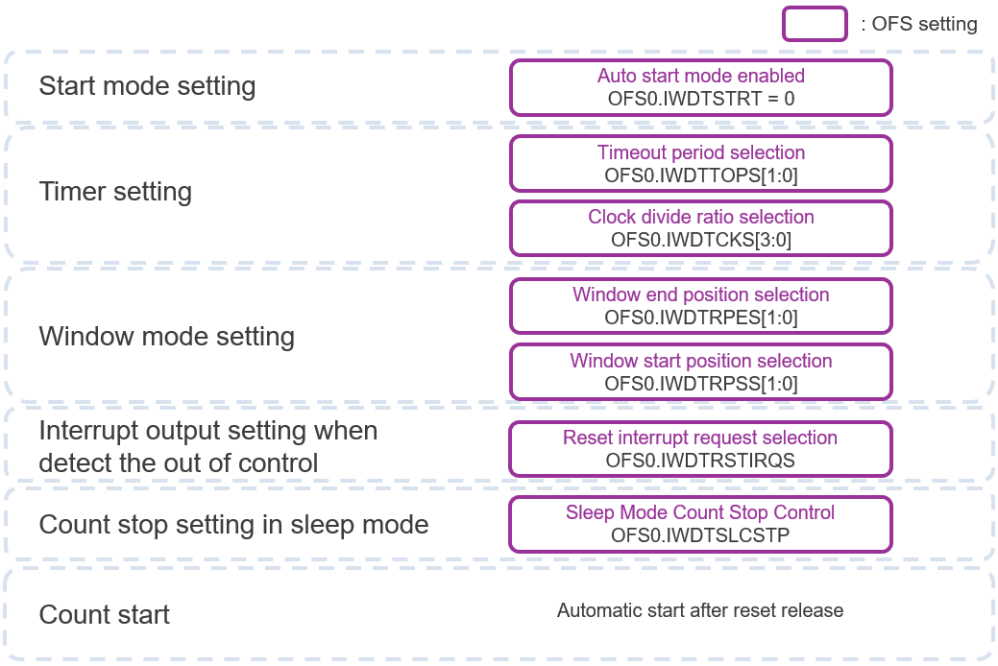
3.2 WDT settings list

The watchdog timer has two modes of operation: auto start mode and register start mode. Each mode is configured by setting registers in the Option Setting Memory (OFS) or WDT. The following figure shows the registers that need to be set in each operation mode.



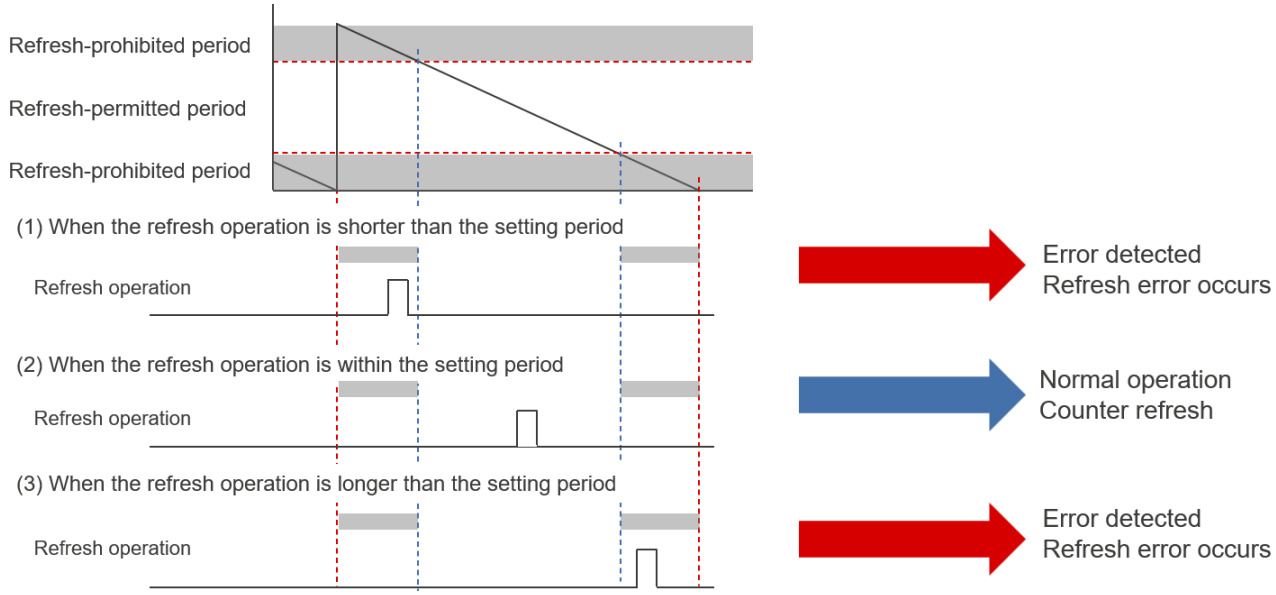
3.3 IWDT settings list

IWDT support auto start mode only. Auto start mode is configured by setting registers in the Option Setting Memory (OFS). The following figure shows the registers that need to be set in each operation mode.



3.4 Function explanation: Window mode

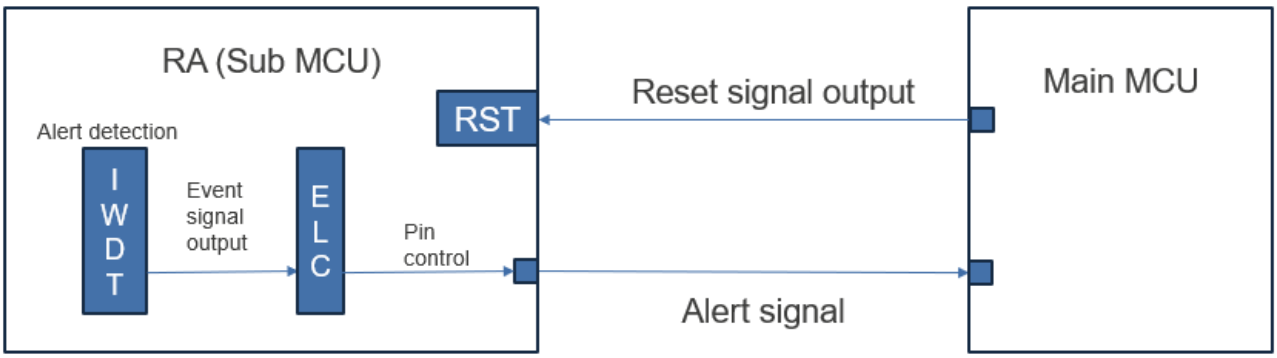
In window mode, set the permitted and prohibited periods for refresh operations. If a refresh operation is executed during the prohibited period, a refresh error occurs and the system is judged to be out of control. In addition to CPU running out of control, this function can also detect deviations in the processing cycle, making it suitable for applications with high periodicity and reliability requirements.



3.5 Use case

WDT or IWDT can output signals to the ELC when a refresh error or underflow occurs and can output external signals from I/O ports using the ELC without going through the CPU.

The figure below shows how to detect an alert in IWDT, send an alert notification to the external main MCU, and execute a hardware reset from the main MCU. The main MCU can check the status of the sub-MCUs and monitor the number of alerts.



*When using ELC, non-maskable interrupt request or interrupt request output must be enabled (IWDTRCR.RSTIRQS = 0).

3.6 Note

(1) Stopping the WDT and IWDT

When WDT or IWDT is operated, it cannot be stopped except under the count stop condition. The stop conditions are as follows.

- Reset
- Low power consumption state (IWDT can be enabled/disabled by register setting)
- In case underflow or refresh error occurs (only in register start mode)

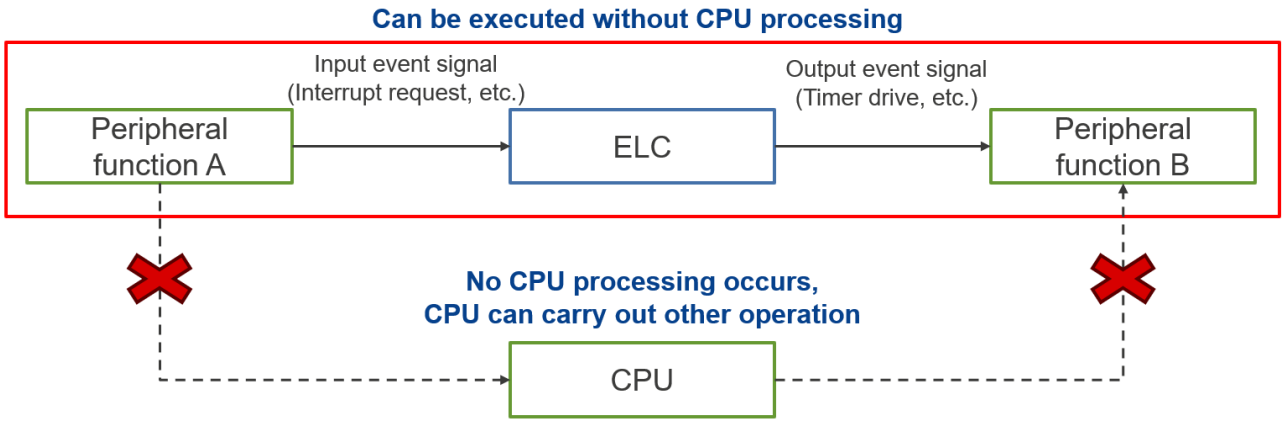
(2) Debugging on an emulator

When debugging on the emulator, the WDT and IWDT counts stop during the break. The stopped counts are restarted during program execution.

4. Event link controller (ELC)

4.1 ELC Overview

The Event Link Controller (ELC) is a function that uses an event signal generated from one peripheral to activate another peripheral without involving the CPU. It is possible to trigger multiple functions with an event generated by one module.



4.2 Comparing ELC and CPU Interrupts

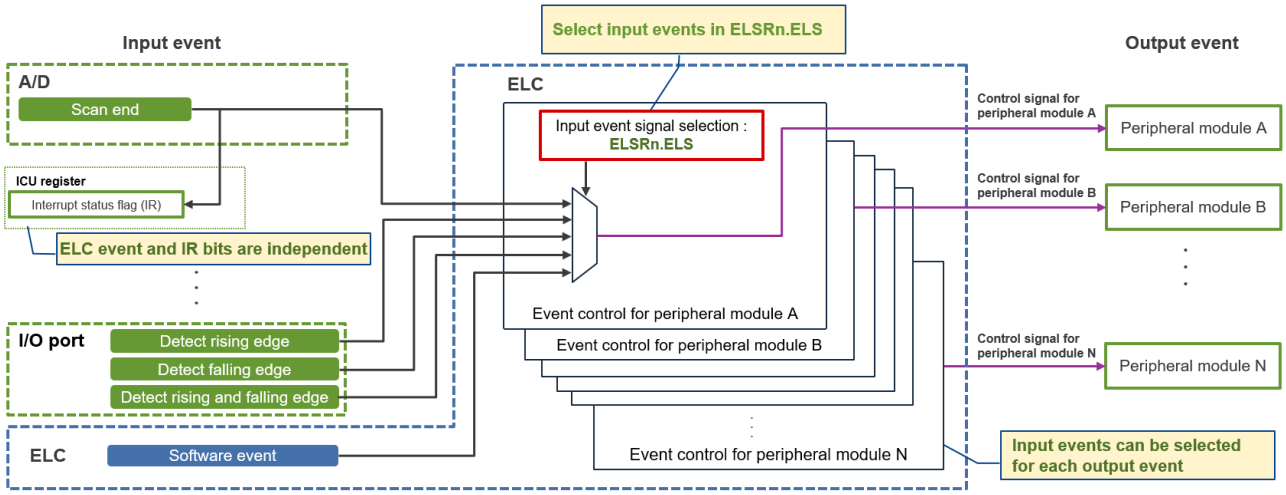
The following compares the case where the event link controller is used with the case where a CPU interrupt is used.

Blue: Benefit

Item	ELC	CPU interrupt
CPU usage	**No**	Yes
Number of simultaneous output events for one input event	**Multiple**	Multiple (However, executed one by one depending on the users software)
Number of output events executed when multiple input events occur	**Multiple output events can be executed in parallel**	Sequentially executed, one by one, in order of priority

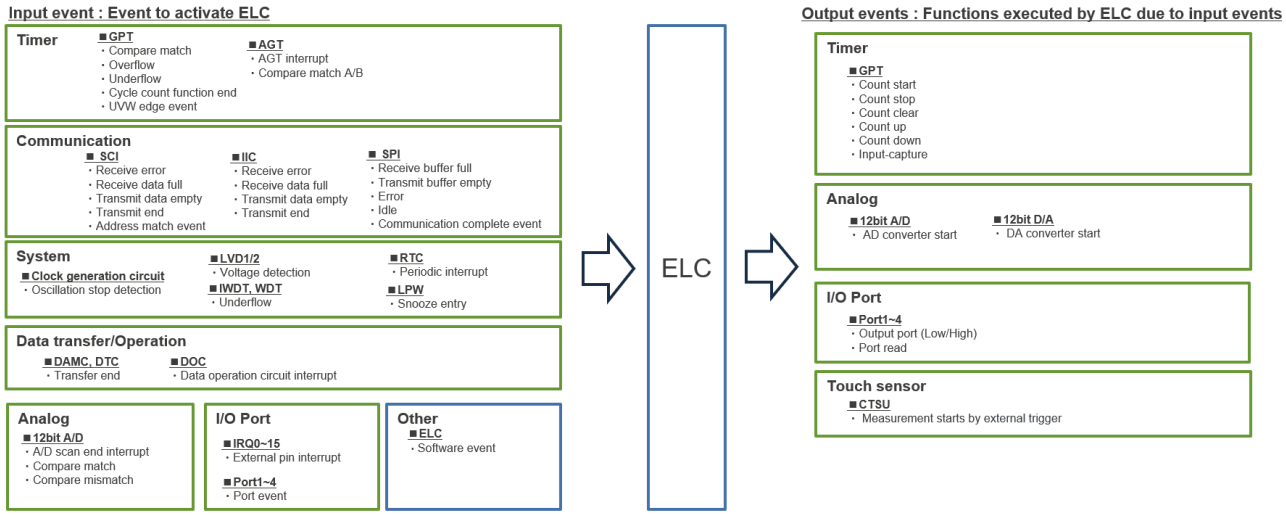
4.3 How to connect ELC input and output event

The figure below shows the structure of the Event Link Controller. ELC is controlled by one control register (ELSRn) for each output event. Multiple input events can be set by one output event. Input events and interrupt status flags (IR bits) are independent and have no effect on CPU interrupts.



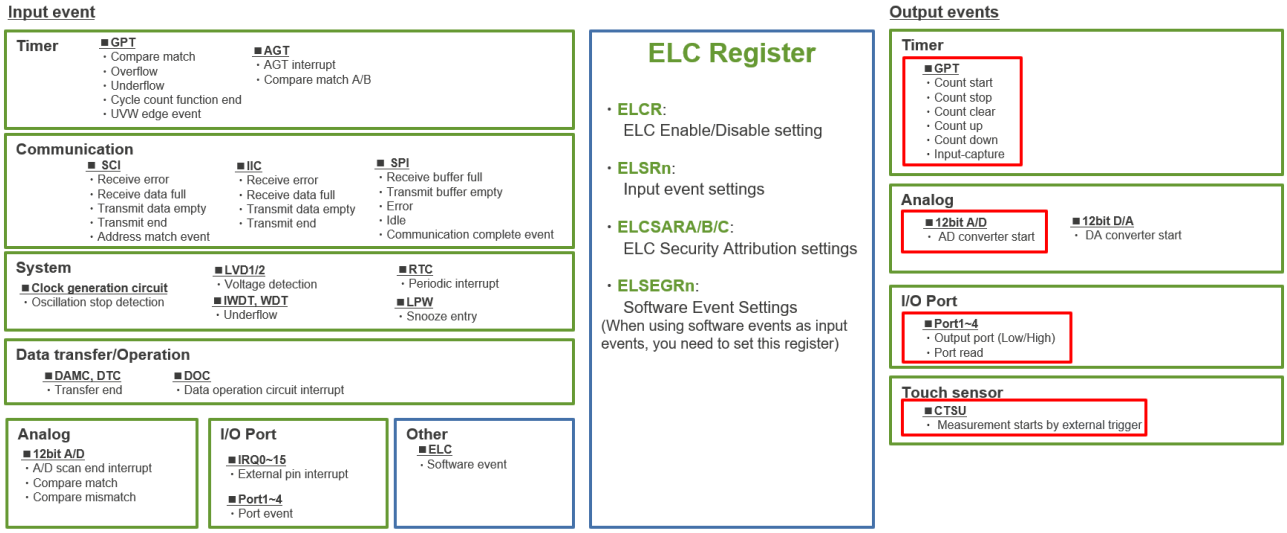
4.4 Summary of ELC input and output events

Summary of input and output events. The following is an example of RA6M5. For details, please refer to the “ELC” chapter on manual of each RA product.



4.5 Related Register when using ELC

ELC is controlled by the ELCR and ELSRn registers. When using the functions in the red frame, the register settings within each function are also needed.



The following figure shows the registers that need to be set on the function side other than ELC.

Output event

Timer

■GPT

- Count start
  GTSSR: Count start factor setting
- Count stop
  GTPSR: Count stop factor setting
- Count clear
  GTCSR: Count clear factor setting
- Count up
  GTUPSR: Count up factor setting
- Count down
  GTDNSR: Count down factor setting
- Capture operation
  GTICnSR: Input capture factor setting

Analog

■12bit A/D

- Event enable
  ADCSR.GBADIE:
  Group Scan End Interrupt and ELC Event Enable(Group B only)
- Conversion start
  ADCSR.EXTRG:
  A/D conversion start trigger selection
  ADSTRGR.TRSn:
  A/D conversion start trigger selection

Touch sensor

■CTSU

- Measurement starts by external trigger
  CTSUCR0.CTSUCAP: Measurement start trigger selection

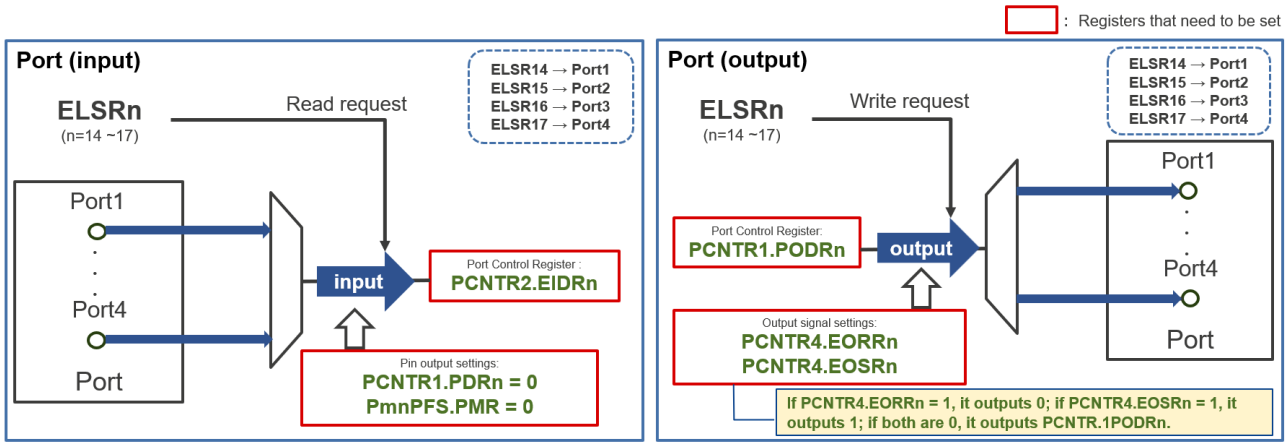
I/O port

PCNTR2.EDIRn: Latches the input data of the corresponding pin when an event occurs.
PCNTR4.EORRn: Sets the output value of the corresponding pin when an event occurs (No output change / Drive High).
PCNTR4.EOSRn: Sets the output value of the corresponding pin when an event occurs (No output change / Drive Low).

4.6 How to set ELC

There are two types of events for a port:

- Input: When an event is entered, the pin level of the number corresponding to event n is sent to PCNTR2.EIDRn.
- Output: When an event is input, the value set by PCNTR1PODRn is output from the pin of the number corresponding to event n. By setting PCNTR4.EORRn or PCNTR4.EOSRn, you can change the output value from pin.



4.7 Supplementary information & notes for ELC settings

(1) Supplementary information

ELC can operate even when CPU is stopped, such as in sleep mode.
However, please note that it will not operate in a mode where the clock supplies to the ELC and the target peripheral module is stopped, as described in the "Usage Notes" section of the ELC chapter. Input events used for the ELC can also be used as a transferring source for DTC.

(2) Precautions

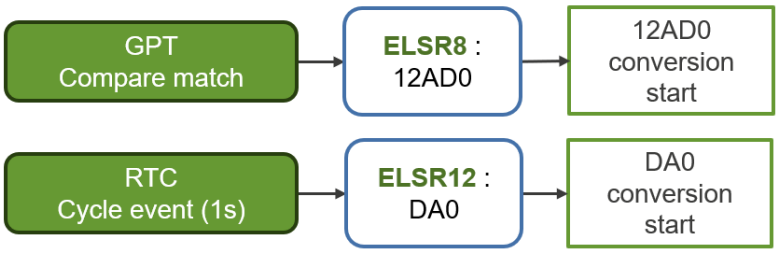
Be sure to read the precautions for using the ELC in the section "Usage Notes" in ELC chapter. Be sure to read the "Usage Notes" for each of the peripheral modules to be used.

4.8 Use case

ELC can be used for a variety of purposes, the following is the example.

(1) Use case 1: Execute peripheral function without CPU operation

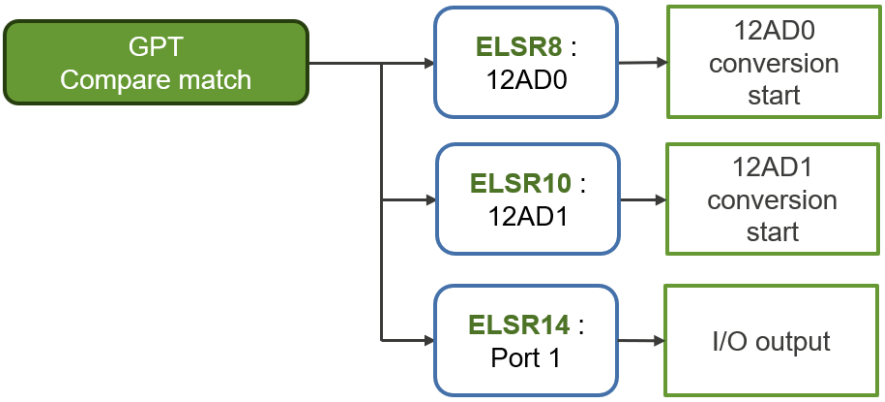
The following shows an example in which a peripheral module is executed at a fixed period without CPU operation by utilizing ELC.



e.g.: Starting a 12 bits A/D or D/A converter cycle by the compare match timer or a RTC cycle event

(2) Use case 2: Execute several different peripherals with one event signal

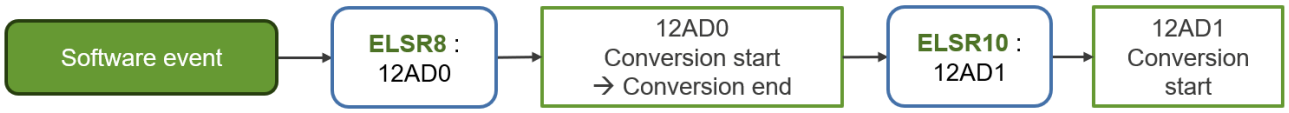
The following shows how to use ELC to activate several peripheral modules with a single input-event signal.



e.g.: Simultaneously operating 12AD0, 12AD1 and I/O output at GPT compare match.

(3) Use case 3: Execute sequentially of peripherals by connecting them together

The following shows an example of starting several peripheral modules in succession with a single input-event signal by utilizing ELC.



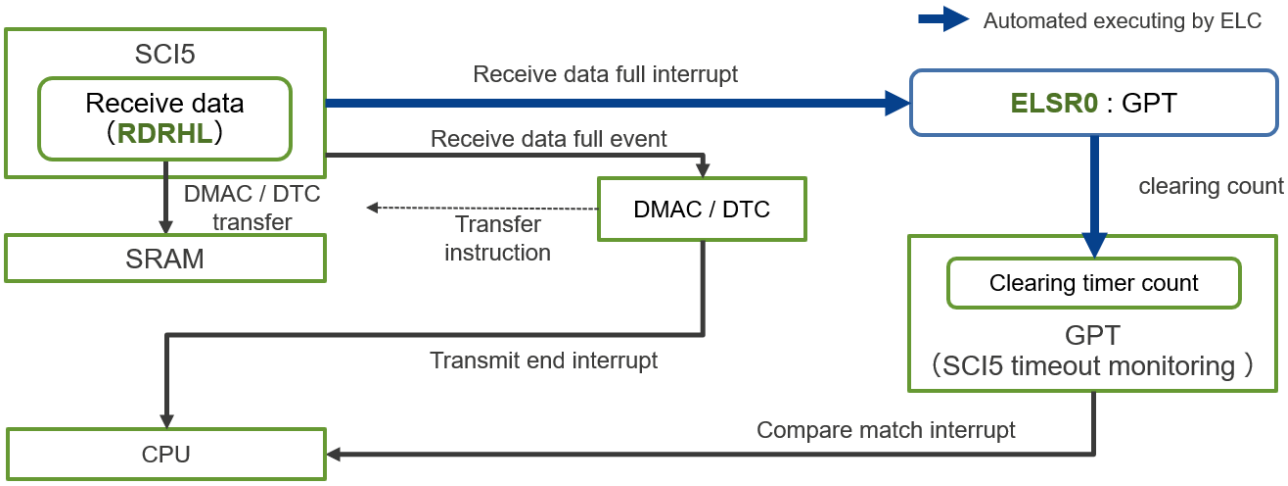
e.g.: Conversion of 12AD0 is started by software event, conversion of 12AD1 is started by 12AD0 conversion end as input event.

4.9 Advanced Example

Detailed examples are introduced in the following.

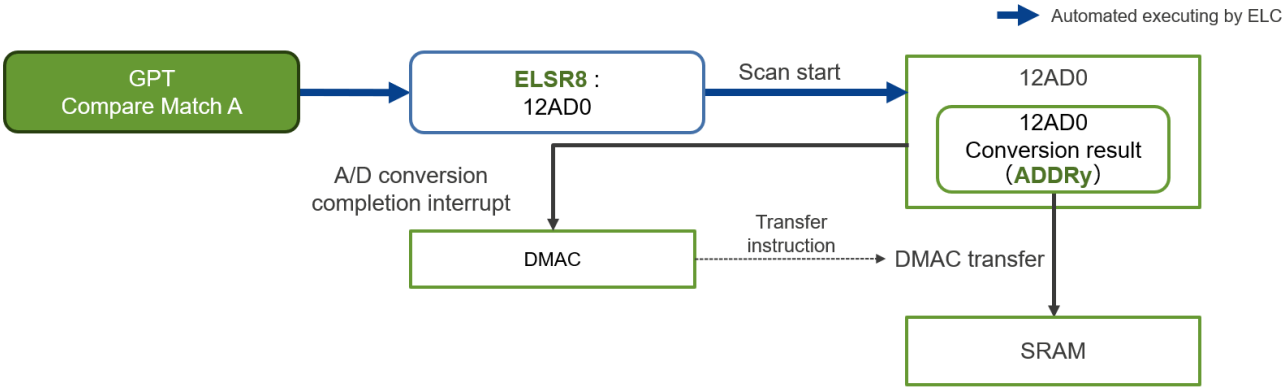
(1) Example 1: Perform SCI data transfer and reset after timeout without using the CPU

This is an example of using the GPT to automatically detect a timeout from the end of one SCI reception to the end of the next in a system requiring SCI reception at regular intervals. The CPU is not involved at all. The SCI receive data full interrupt is used as an ELC input event, and the ELC clears the GPT timer count. The timer is configured to compare-match according to the reception interval. If the counter clear is not performed in time, a compare match interrupts occurs to notify the CPU of the timeout. Concurrently, the receive data full interrupt is also used as a transfer request to transfer the received data to SRAM via DMAC/DTC.



(2) Example 2: Periodically AD converting & save the conversion result in SRAM without CPU operation

By combining the ELC and DMAC, the results from a A/D conversion can be moved to SRAM automatically without CPU intervention. The ELC triggers the start of A/D conversion using a GPT compare match as an input event. Additionally, the A/D conversion end interrupt serves as a transfer request factor, allowing the DMAC to store the A/D conversion results into SRAM. By configuring the DMAC to free-run mode, there is no need to reconfigure the transfer count.



4.10 List of Application Notes using ELC

Below are examples of applications using ELC. For details, please refer to each application note.

Application note name	Document No
RA2E1/RA2E2 Group Low Power Application (Use of ADC, DTC and ELC at Snooze mode) for FPB-RA2E1 and FPB-RA2E2	R30AN0392
Sensorless Vector Control of PM Motor for Renesas Flexible Motor Control Series, Equipped with Flying Start	R01AN8230
Renesas RA8 Series Getting Started with RA8 Power Management Techniques	R01AN7342

5. External bus controller

5.1 Bus Controller configuration

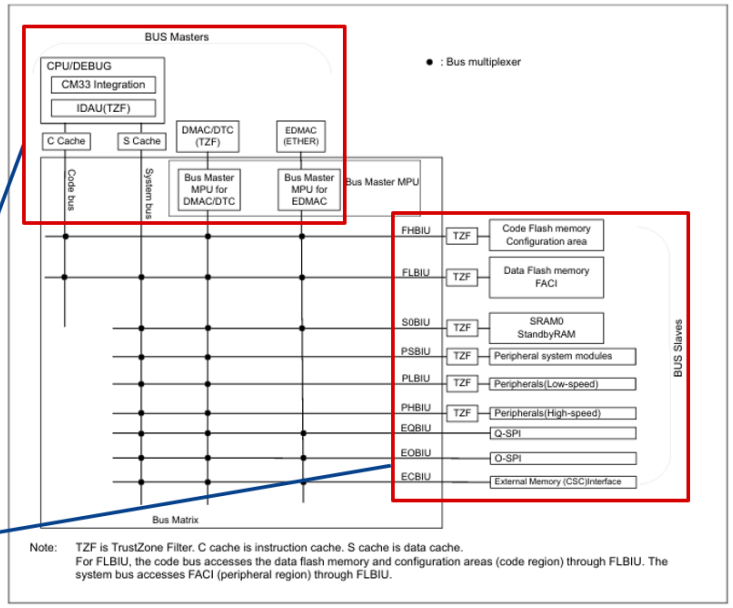
RA internal buses are divided into master and slave sections, with the master and slave connected in a matrix pattern.

BUS Masters

There are four buses and three masters: CPU, DMAC/DTC, and EDMAC.
When each master accesses different slaves, parallel operations is possible.

BUS Slaves

There are nine buses, each with their own corresponding slave.
EQBIU/EBIU/ECBIU are classified as external buses.
By setting the BUSSCNT register corresponding to each slave, you can set the master arbitration for each slave.



5.2 Arbitration

Mean of arbitration is the priority set for each bus to prevent collisions in bus access. By setting the value of the BUSSCNT<slave>.ARBS bit corresponding to the slave, you can decide the arbitration for each master for each slave. Below are examples of bit settings and priority for the FHBIU bus.

BUSSCNT FHBIU.ARBS[1:0]	The priority for each bus (Arbitration)		
	EDMAC	DMAC / DTC	CPU
00	High	Middle	Low
10	High*	High*	Middle
11	High*	High*	High*

* Operates in a round-robin manner between the same priority levels

5.3 External bus

The external address space is divided into several CS areas and SPI area.

Bus connection can be made by setting the access method and waiting according to the specifications of the other device and controlling the output waveform.

■CS areas

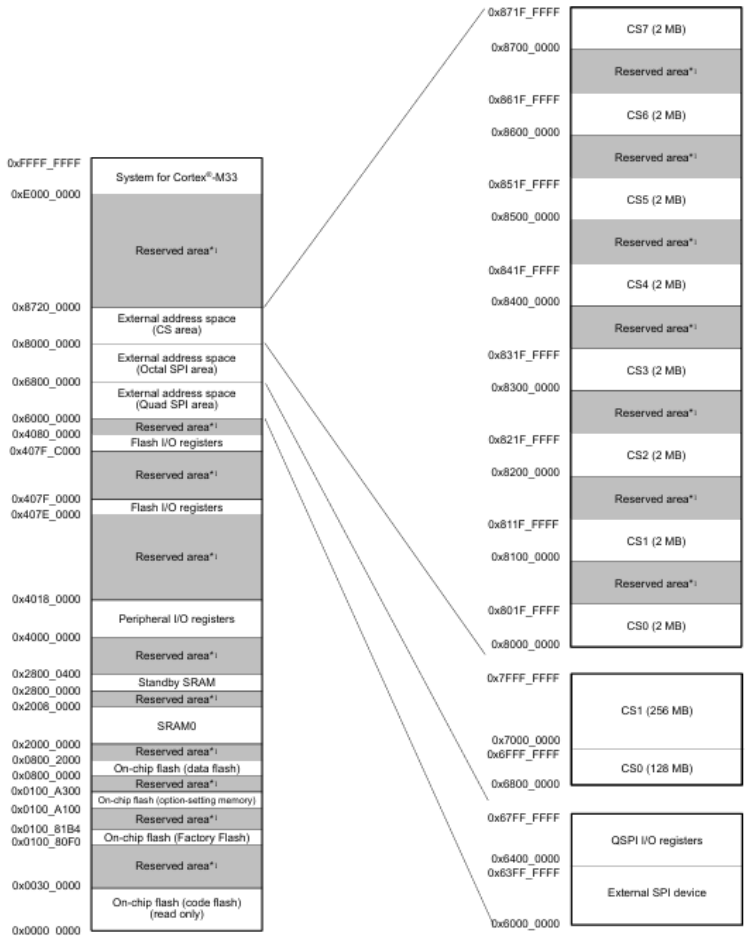
Some products support this function.

For each space, you can set the following individually:

- Access Methods (Separate bus or Address/data multiplexed bus)
- Bus width
- Recovery cycles, Cycle wait, External/Internal wait control
- Write access mode (Single write strobe mode/byte strobe mode)
- endian mode
- Page Read Access/Page Write Access

■SPI areas

Some RA products have QSPI and OSPI interface. Below SPI means both interfaces. SPI areas are supporting some RA products. Each space is matched with external storage areas, so it provides high-speed access through serial communication. For details, see the "Serial Peripheral Interface (SPI)", "Quad Serial Peripheral Interface (QSPI) ", “Octa Serial Peripheral Interface (OSPI)” on manual of each RA product.



5.3.1 Mode and pin list for External Bus

The following is a list of the modes of the external bus and the available pins.

Pin	CS areas				No Access State
	Separate bus		Address/data multiplexed bus		
	Byte strobe	Single write strobe	Byte strobe	Single write strobe	
A23~A0	✓	✓			
D15～D0	✓	✓			high-impedance
A0/D0~A15/D15*			✓	✓	
BC0#~BC3#		✓		✓	High
CS0#~CS7#	✓	✓	✓	✓	High
RD#	✓	✓	✓	✓	High
WR0#/WR#	✓ (Use as WR0#)	✓ (Use as WR#)	✓ (Use as WR0#)	✓ (Use as WR#)	High
WR1#~WR3#	✓		✓		High
ALE			✓	✓	Low
WAIT#	✓	✓	✓	✓	
BCLK	✓	✓	✓	✓	

* A0/D0~A15/D15 are dedicated to address/data multiplexing.
It cannot be used as an address on a separate bus.

5.3.2 Mode and registers for external bus

The following is a list of the modes of the external bus and the registers.

Register name and settings		CS area	Bus error	Other
CSnCR	Operation Enable, Bus width, Endian mode, Access method (Separate or Multiplex)	✓		
CSnREC	Recovery Cycle (Read, Write)	✓		
CSRECEN	Recovery Cycle Insertion Enable	✓		
CSnMOD	Write Access Mode Select, External Wait Enable, Page Read/Write Access Enable	✓		
CSnWCR1,2	Wait Control (Normal Write/Read, Page Write/Read, CS Extension Cycle (Read/Write), Write Data Output Extension/Output Wait, Address Cycle, Assert (RD,WD,CS))	✓		
BUSnERRCLR	Bus Error Status Clearing		✓	
DMACDTCERRCLR	DMAC/DTC Error Clearing		✓	
BUSSARA,B	Bus Security Settings			✓
BUSSCNT<slave*>	Slave Bus Arbitration Settings			✓

* : Inside to <>, the slave's name is written.

Since the number of masters varies for each slave, the size of the arbitration setting bit area varies.

Please be careful when setting bits.

5.3.3 Mode and registers for external bus (Detail info. For CS)

The following are the register details that require each mode and configuration of the external bus in CS areas.

W: Write, R: Read

Register and settings				Separate	Address/data multiplex	Page access	Recovery
Items	Register	Bit	Details				
Settings related to mode	CSnCR	EXENB	Operation Enable	✓	✓	-	-
		BSIZE[1:0]	External Bus Width Select	✓	✓	-	-
		EMODE	Endian Mode	✓	✓	-	-
		MPXEN	Address/Data Multiplexed I/O Interface Select	✓	✓	-	-
	CSnMOD	WRMOD	Write Access Mode Select (Byte strobe/Single write strobe)	✓	✓	-	-
		EWENB	External Wait Enable	✓	✓	-	-
		PRENB	Page Read Access Enable	-	-	✓(R)	-
		PWENB	Page Write Access Enable	-	-	✓(W)	-
		PRMOD	Page Read Access Mode Select (Normal access compatible/External data read continuous assertion)	-	-	✓(R)	-
Settings related to Recovery cycle	CSnREC	RRCV[3:0]	Read Recovery count setting	-	-	-	✓(R)
		WRCV[3:0]	Write Recovery count setting	-	-	-	✓(W)
	CSRECEN	RCVEN0~7	Separate Bus Recovery Cycle Insertion Enable	-	-	-	✓(R)
		RCVENM0~7	Multiplexed Bus Recovery Cycle Insertion Enable	-	-	-	✓(W)
Wait Control	CSnWCR1	CSPWWAIT[4:0]	Page Write Cycle Wait Select	-	-	✓(W)	-
		CSPRWAIT[4:0]	Page Read Cycle Wait Select	-	-	✓(R)	-
		CSWWAIT[4:0]	Normal Write Cycle Wait Select	✓(W)	✓(W)	-	-
		CSRWAIT[4:0]	Normal Read Cycle Wait Select	✓(R)	✓(R)	-	-
	CSnWCR2	CSROFF[2:0]	Read-Access CS Extension Cycle Select	✓(R)	✓(R)	-	-
		CSWOFF[2:0]	Write-Access CS Extension Cycle Select	✓(W)	✓(W)	-	-
		WDOFF[2:0]	Write Data Output Extension Cycle Select	✓(W)	✓(W)	-	-
		AWAIT[1:0]	Address Cycle Wait Select		✓(W)	-	-
		RDON[2:0]	RD Assert Wait Select	✓(R)	✓(R)	-	-
		WRON[2:0]	WR Assert Wait Select	✓(W)	✓(W)	-	-
		WDON[2:0]	Write Data Output Wait Select	✓(W)	✓(W)	-	-
		CSON[2:0]	CS Assert Wait Select	✓(R/W)	✓(R/W)	-	-

5.3.4 Pin settings for External bus interface

Pin settings for external bus interface are performed by setting Pin Function Select Register (PmnPFS.PSEL[4:0]). Please disable write prohibition in Write-Protect Register (PWPR) before operating PSEL [4:0]. Below is an example of the RA6M5.

Pin	PSEL[4:0]	Signal	Number of pins		
			176 pins	144 pins	100 pins
P100	01011b	D00[A00/D00]	✓	✓	✓
P101		D01[A01/D01]	✓	✓	✓
P102		D02[A02/D02]	✓	✓	✓
P103		D03[A03/D03]	✓	✓	✓
P104		D04[A04/D04]	✓	✓	✓
P105		D05[A05/D05]	✓	✓	✓
P106		D06[A06/D06]	✓	✓	✓
P107		D07[A07/D07]	✓	✓	✓
P111		A05	✓	✓	✓
P112		A04	✓	✓	✓
P113		A03	✓	✓	✓
P114		A02	✓	✓	✓
P115		A01	✓	✓	✓
P202	01011b	WR1/BC1	✓	✓	-
P203		A19	✓	✓	-
P204		A18	✓	✓	-
P205		A16	✓	✓	✓
P206		WAIT	✓	✓	✓
P207		A17	✓	✓	✓
P208		CS4	✓	✓	✓
P209		CS5	✓	✓	✓
P210		CS6	✓	✓	✓
P211		CS7	✓	✓	✓

Pin	PSEL[4:0]	Signal	Number of pins			
			176 pins	144 pins	100 pins	
					BGA	LQFP
P301	01011b	A06	✓	✓	✓	✓
P302		A07	✓	✓	✓	✓
P303		A08	✓	✓	✓	✓
P304		A09	✓	✓	-	✓
P305		A10	✓	✓	-	✓
P306		A11	✓	✓	-	✓
P307		A12	✓	✓	-	✓
P308		A13	✓	✓	-	-
P309		A14	✓	✓	-	-
P310		A15	✓	✓	-	-
P311		CS2	✓	✓	-	-
P312		CS3	✓	✓	-	-
P313		A20	✓	✓	-	-
P314		A21	✓	-	-	-
P315		A22	✓	-	-	-
P504	01011b	ALE	✓	✓	✓	✓
P600		RD	✓	✓	✓	✓
P601		WR/WR0	✓	✓	✓	✓
P602		EBCLK	✓	✓	✓	✓
P603		D13[A13/D13]	✓	✓	-	-
P604		D12[A12/D12]	✓	✓	-	-
P605		D11[A11/D11]	✓	✓	-	-
P608		A00/BC0	✓	✓	✓	✓
P609		CS1	✓	✓	✓	✓
P610		CS0	✓	✓	✓	✓
P612		D08[A08/D08]	✓	✓	-	-
P613		D09[A09/D09]	✓	✓	-	-
P614		D10[A10/D10]	✓	✓	-	-
P800		D14[A14/D14]	✓	✓	-	-
P801		D15[A15/D15]	✓	✓	-	-
P900		A23	✓	-	-	-

5.3.5 CS area (Write Access mode)

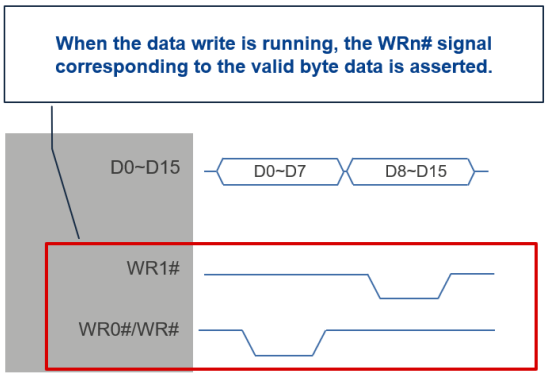
There are two write access modes in the CS space: Byte Strobe and Single Write Strobe mode.

Byte strobe : This mode uses multiple write signals (WR0#, WR1#) for writing processing. Since different writing signals are used for each byte position, byte control signals are not used.

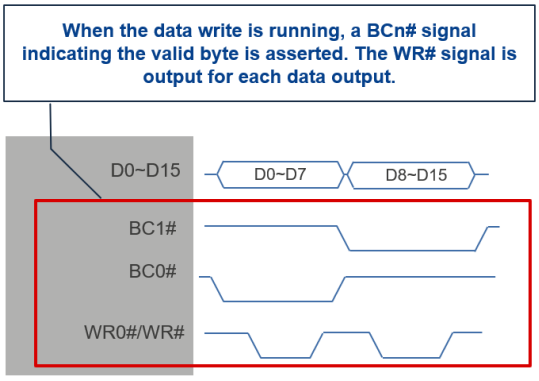
Single write strobe : This mode performs write processing using a single write signal (WR#). Use the byte control signal (BC0#, BC1#) to specify the effective byte position.

5.3.6 Write Access mode

Write access mode support Byte strobe mode and Single write strobe mode. An example of each signal output is shown below.



Signal output example for Byte Strobe Write Access (16bit access to 4n+1 address in 16bit bus area)



Signal output example for Single write Strobe Write Access (16bit access to 4n+1 address in 16bit bus area)

5.3.7 Endian

The endian type for the external bus can be set for each CS area. In addition, multiple accesses occur in a single transfer request depending on the set endian type and access address. If you access a data size of more than the bus width, page access may occur after the second time the number of accesses. By specifying the access address according to the bus width of the space to which the access is being accessed, the increase in the number of accesses can be reduced.

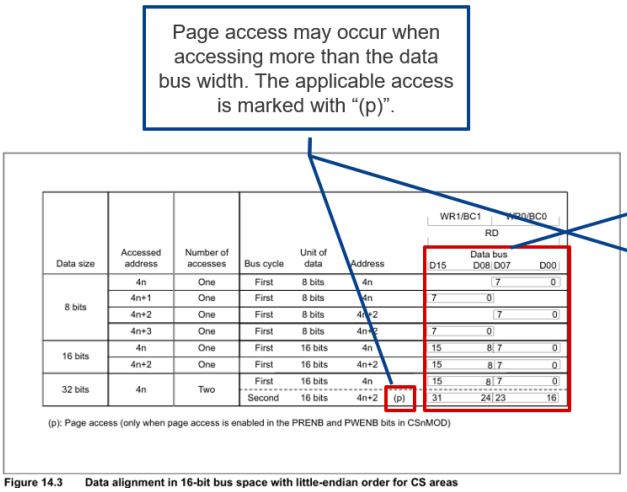


Figure 14.3 Data alignment in 16-bit bus space with little-endian order for CS areas

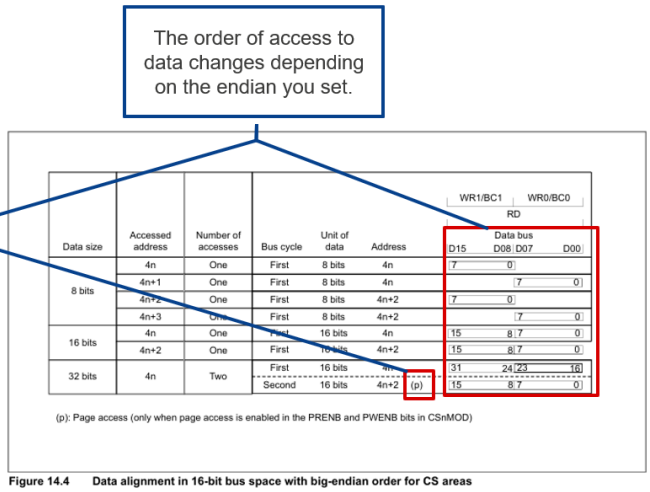
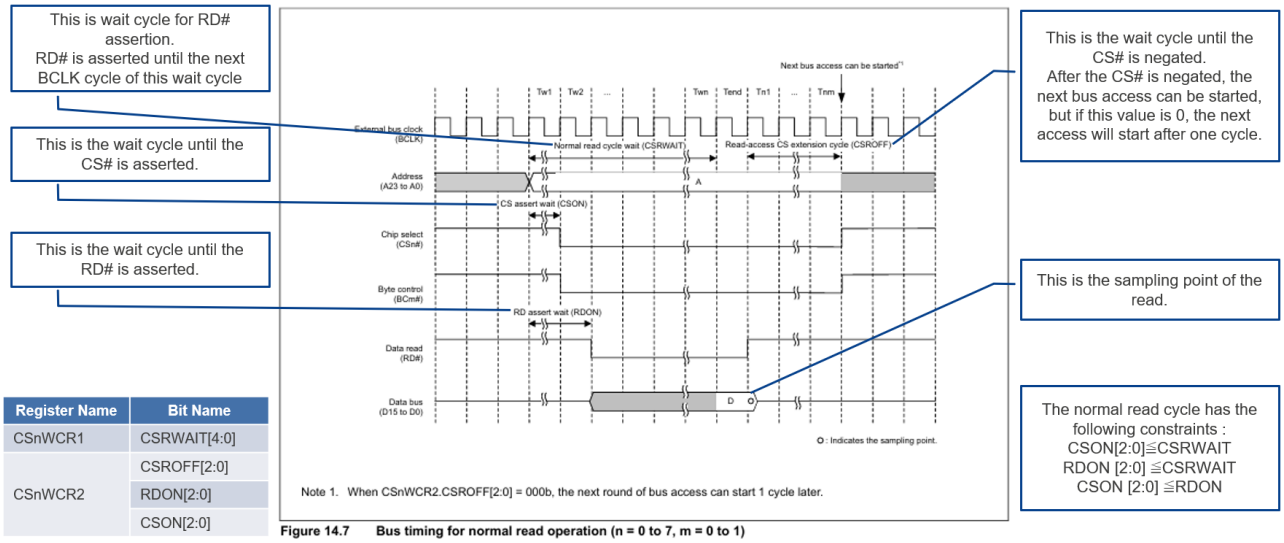


Figure 14.4 Data alignment in 16-bit bus space with big-endian order for CS areas

5.3.8 Each wait setting

(1) Normal read timming

Example of the timing when it is “Normal read". Please insert the wait to match the connected device and control the output waveform.



Register Name	Bit Name
CSnWCR1	CSRWAIT[4:0]
CSnWCR2	CSROFF[2:0]
	RDON[2:0]
	CSON[2:0]

Note 1. When CSnWCR2.CSROFF[2:0] = 000b, the next round of bus access can start 1 cycle later.

Figure 14.7 Bus timing for normal read operation (n = 0 to 7, m = 0 to 1)

(2) Normal write timming

Example of the timing when it is “Normal write". Please insert the wait to match the connected device and control the output waveform.

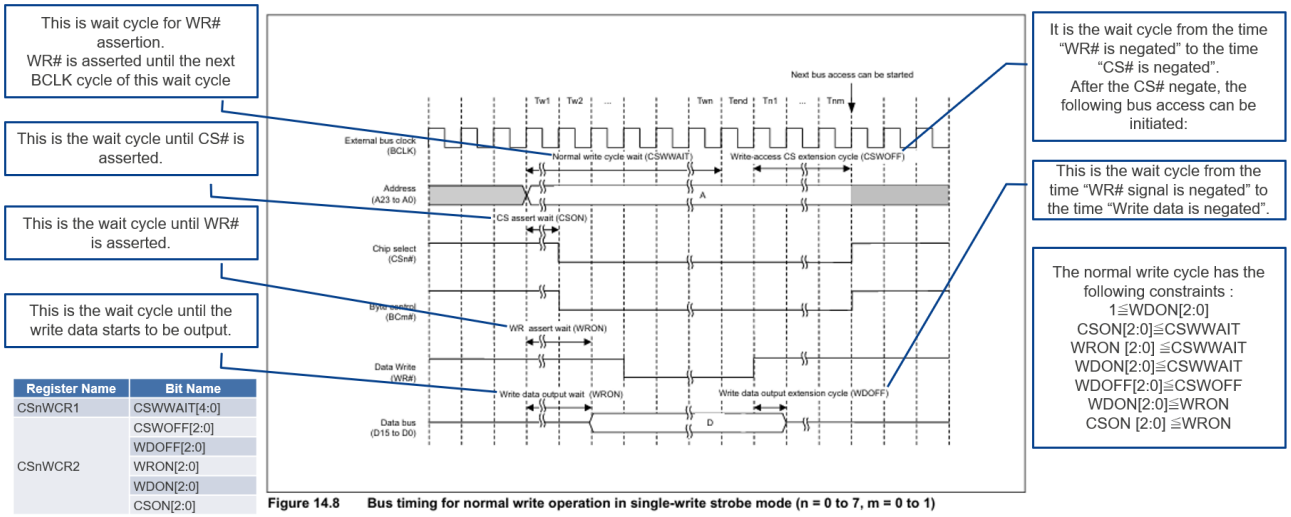


Figure 14.8 Bus timing for normal write operation in single-write strobe mode (n = 0 to 7, m = 0 to 1)

5.3.9 External bus Clock and BCLK pin

Depending on the product, BCLK pin output must be set to 1/2 of the internal BCLK. Here are some examples of how this works and precautions.

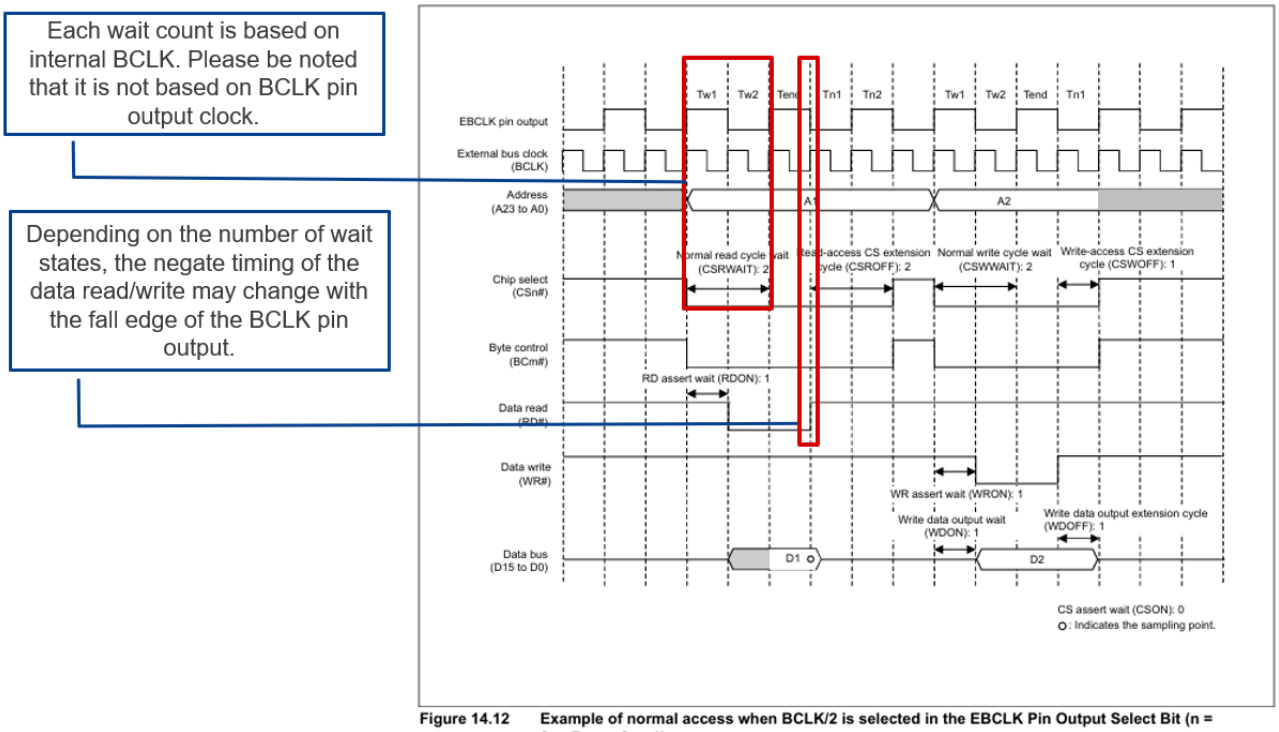


Figure 14.12 Example of normal access when BCLK/2 is selected in the EBCLK Pin Output Select Bit (n = 0 to 7, m = 0 to 1)

5.3.10 Page Access

Page access occurs when multiple accesses occur in a single transfer request. Page access can have a different number of waiting between data accesses than normal access.

During the second and subsequent accesses, the page read (write) cycle wait may be inserted instead of the normal read(write) cycle weight of normal access.

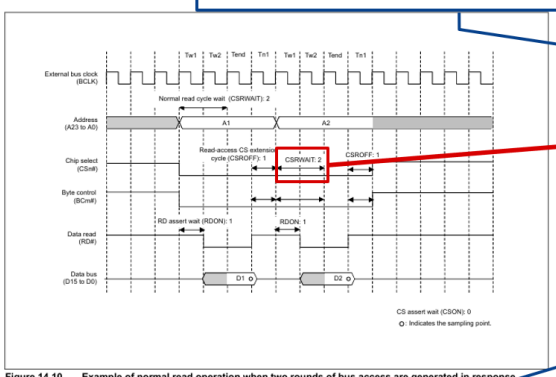


Figure 14.10 Example of normal read operation when two rounds of bus access are generated in response to a single transfer request (n = 0 to 7, m = 0 to 1)

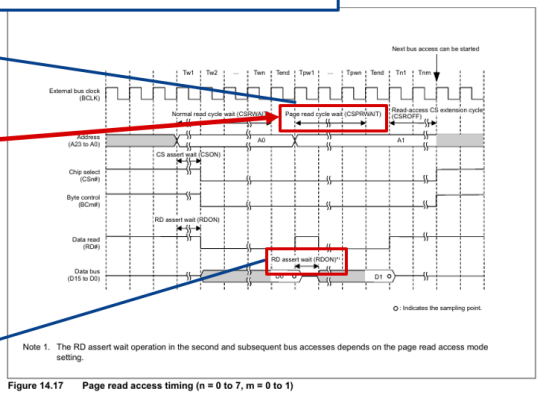


Figure 14.17 Page read access timing (n = 0 to 7, m = 0 to 1)

Register Name	Bit Name
CSnWCR1	CSPWWAIT[4:0](Write)
	CSPRWAIT[4:0](Read)

The WR assert wait is as valid as the first time. On the other hand, RD assert wait is handled as follows :
CSnMOD.PRMOD bit is 0 : RD assert is valid as in the first time, and the RD signal between them is negated.
CSnMOD.PRMOD bit is 1 : RD assert is valid as in the first time, and the RD signal continues to assert during that time.

5.3.11 Address / Data Multiplexed Bus

The following is access example of Address/Data Multiplexed Bus. Please insert the wait to match the connected device and control the output waveform.

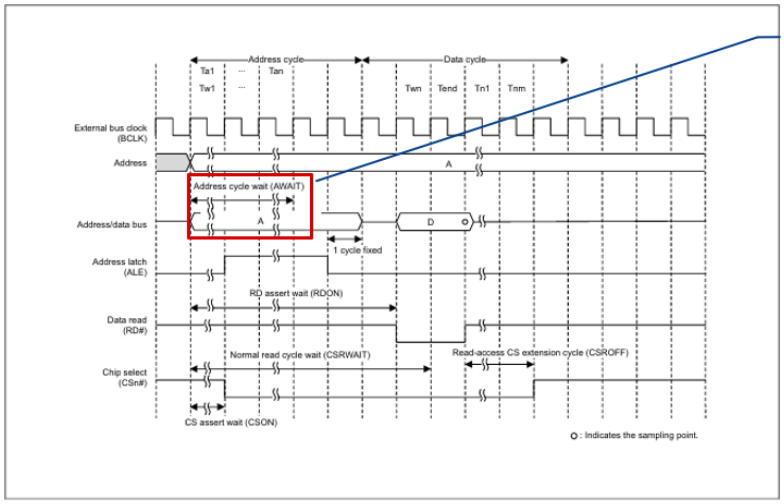


Figure 14.23 Example of read access operation with address/data multiplexed I/O interface (n = 0 to 7)

In the first address cycle, set the negate timing for the ALE.

Address/Data Multiplexed Bus has the following constraints :

< Read >	< Write >
CSON[2:0]≦CSRWAIT	CSON≦CSWWAIT
RDON[2:0]≦CSRWAIT	WRON [2:0] ≦CSWWAIT
CSON [2:0] ≦RDON	WDON[2:0]≦CSWWAIT
AWAIT[1:0]+2≦RDON	WDOFF[2:0]≦CSWOFF
CSON[2:0]≦AWAIT	WDON[2:0]≦WRON
	CSON [2:0] ≦WRON
	AWAIT[1:0] +2≦WRON
	AWAIT[1:0] +2≦WDON
	CSON[2:0] ≦AWAIT

< Read >

Register Name	Bit Name
CSnWCR1	CSRWAIT[4:0]
CSnWCR2	CSROFF[2:0]
	RDON[2:0]
	CSON[2:0]
	AWAIT[4:0]

< Write >

Register Name	Bit Name
CSnWCR1	CSWWAIT[4:0]
CSnWCR2	CSWOFF[2:0]
	WDOFF[2:0]
	WRON[2:0]
	WDON[2:0]
	CSON[2:0]
	AWAIT[4:0]

5.3.12 Recovery Cycles

RA can set recovery cycles (extended CS# pin negate period) between external access and external access. Need to set the CS Negate period according to the connected device. In addition, if two or more external bus accesses are required for a single transfer request from the bus master, and the following conditions for inserting recovery cycles are met, the insertion of recovery cycles into the intermediate access is as follows.

(1) For normal access case:

When page access is disabled: Recovery cycles are inserted into intermediate access
When page access is enabled: Recovery cycles are not inserted into intermediate access

(2) For Address/Data Multiplexed case:

Recovery cycles are inserted into intermediate access regardless of whether page access is enabled/disabled

Access Type	External Address Space	Insertion of Recovery Cycles	Enable/Disable recovery cycles insertion settings	
			Separate bus	Multiplexed bus
Read access after Read access	Same area	CSnREC.RRCV[3:0]*	CSRECEN.RCVEN0	CSRECEN.RCVENM0
	Different area		CSRECEN.RCVEN1	CSRECEN.RCVENM1
Write access after Read access	Same area		CSRECEN.RCVEN2	CSRECEN.RCVENM2
	Different area		CSRECEN.RCVEN3	CSRECEN.RCVENM3
Read access after Write access	Same area	CSnREC.WRCV[3:0]	CSRECEN.RCVEN4	CSRECEN.RCVENM4
	Different area		CSRECEN.RCVEN5	CSRECEN.RCVENM5
Write access after Write access	Same area		CSRECEN.RCVEN6	CSRECEN.RCVENM6
	Different area		CSRECEN.RCVEN7	CSRECEN.RCVENM7

*: When CSnWCR2.CSROFF[2:0] is 000b (Read CS extend cycle is 0), The next bus start cycle will be one cycle later. Even if you do not insert a cycle in this setting, the one cycle negate period will be inserted.

The following is an example of an insertion of recovery cycles.

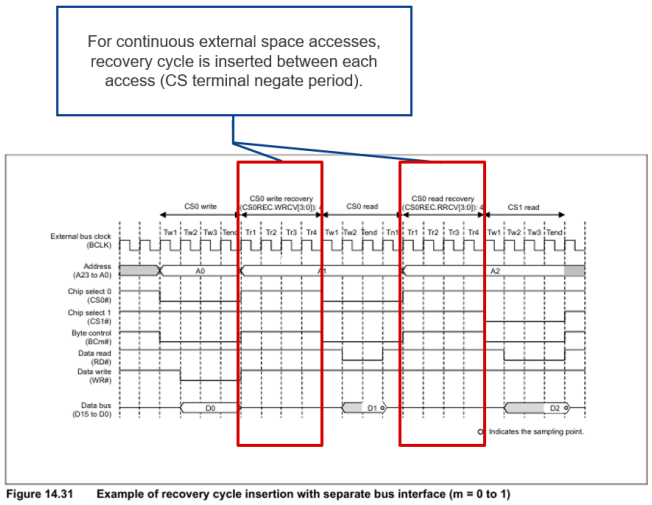


Figure 14.31 Example of recovery cycle insertion with separate bus interface (m = 0 to 1)

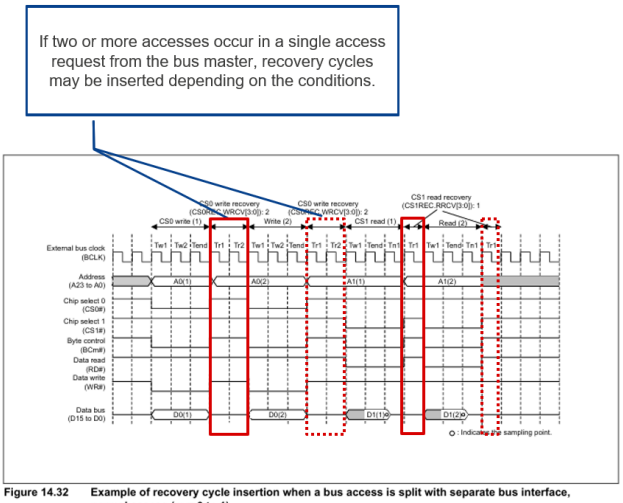


Figure 14.32 Example of recovery cycle insertion when a bus access is split with separate bus interface, normal access (m = 0 to 1)

5.3.13 Bus Error

RA can detect five main types of bus errors. Each time an error occurs, each bit in BUS Error Status Register (BUSn.ERRSTAT) is set to 1. If an error occurs simultaneously, bits will be activated in the following priority order; Bus Master MPU Error > Illegal Address Access Error, Slave Bus Error. Illegal Address Access Error and Slave Bus Error do not occur simultaneously.

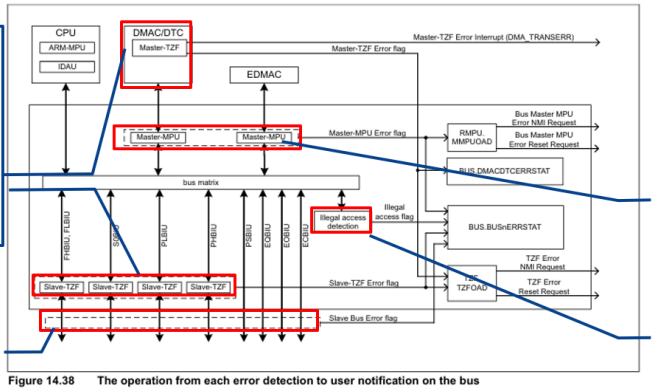
Master-TZF Error

Slave-TZF Error

This error occurs when non-secure access is made to an area protected by the TrustZone filter. Sleep mode is canceled by a reset generated by this error. If you select this error as the trigger factor for a non-maskable interrupt, set the NMICR register, NMICLR register, and NMIER register.

Slave Bus Error

This error occurs in each slave bus. Errors such as timeouts fall into this category.



Bus Master MPU Error

This error occurs when DMAC, DTC, or EDMAC access the protected area set by the MPU. Sleep mode is canceled by a reset generated by this error. This error can be selected as a trigger factor for a non-maskable interrupt.
For details of this error, please refer to the "Memory Protection Unit (MPU)" chapter on manual of each RA product.

Illegal Address Access Error

This error occurs when access to an unauthorized address is detected within the bus matrix.

Figure 14.38 The operation from each error detection to user notification on the bus

(1) Illegal Access Error

Area that can detect illegal address access errors is decided for each bus. Below is an example of the RA6M5.

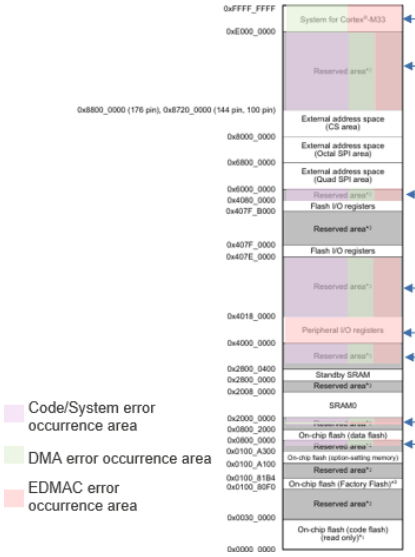


Table. Conditions leading to illegal address access errors

Address	Slave bus	Master bus			
		CPU		DMA	EDMAC
		Code	System		
0xE000_0000~0xFFFF_FFFF	System for Cortex®-M33			E	E
0x8800_0000~0xDFFF_FFFF	Reserved		E	E	E
0x8000_0000~0x87FF_FFFF	ECBIU		-	-	-
0x6800_0000~0x7FFF_FFFF	EOBIU		-	-	-
0x6000_0000~0x67FF_FFFF	EQBIU		-	-	-
0x4080_0000~0x5FFF_FFFF	Reserved		E	E	E
0x407E_0000~0x407F_FFFF	FLBIU		-	-	-
0x4018_0000~0x407D_FFFF	Reserved		E	E	E
0x4010_0000~0x4017_FFFF	PHBIU		-	-	E
0x4008_0000~0x400F_FFFF	PLBIU		-	-	E
0x4000_0000~0x4007_FFFF	PSBIU		-	-	E
0x2801_0000~0x3FFF_FFFF	Reserved		E	E	E
0x2000_0000~0x2800_FFFF	S0BIU		-	-	-
0x1010_0000~0x1FFF_FFFF	Reserved	E		E	E
0x1000_0000~0x100F_FFFF	Reserved	-		E	E
0x0804_0000~0x0FFF_FFFF	Reserved	E		E	E
0x0800_0000~0x0803_FFFF	FLBIU	-		-	-
0x0200_0000~0x07FF_FFFF	Reserved	E		E	E
0x0000_0000~0x01FF_FFFF	FHBIU	-		-	-

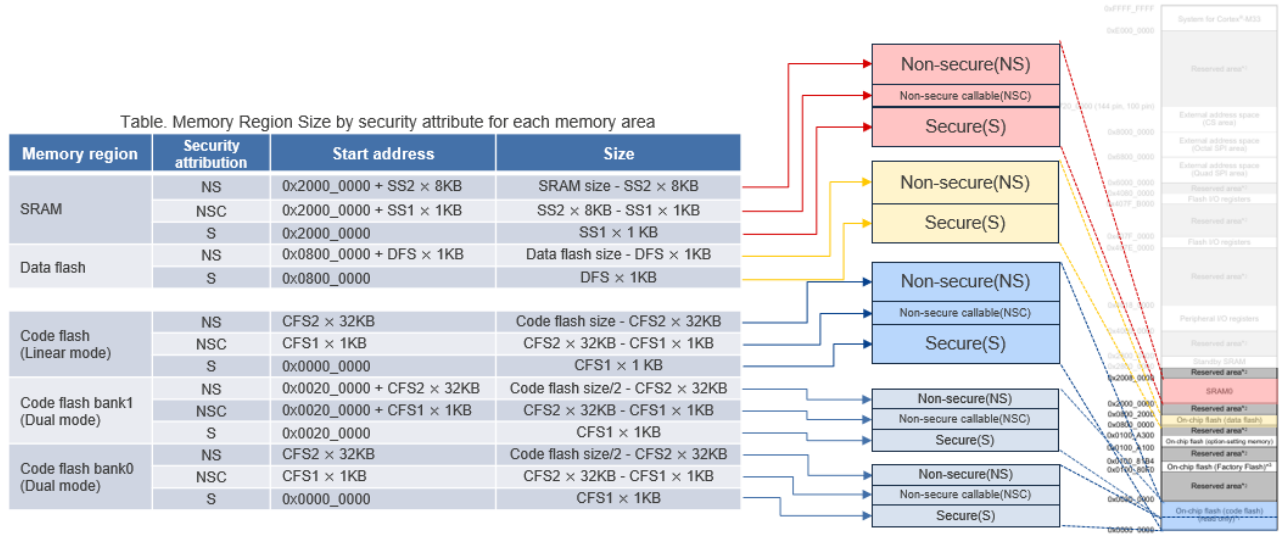
E : A bus error occurs.
- : A bus error has not occurred.

(2) TZF error

If non-secure access is made to a memory area set to Secure or Non-secure callable, a TrustZone access error will occur. Size of the area where Security Attribution is set is decided by the value of Security Attribution Monitor Register corresponding to each area. For details, please refer to the "Security Features" chapter on manual of each RA product. Below is an example of the RA6M5.

Table. Memory Region Size by security attribute for each memory area

Memory region	Security attribution	Start address	Size
SRAM	NS	0x2000_0000 + SS2 × 8KB	SRAM size - SS2 × 8KB
	NSC	0x2000_0000 + SS1 × 1KB	SS2 × 8KB - SS1 × 1KB
	S	0x2000_0000	SS1 × 1 KB
Data flash	NS	0x0800_0000 + DFS × 1KB	Data flash size - DFS × 1KB
	S	0x0800_0000	DFS × 1KB
Code flash (Linear mode)	NS	CFS2 × 32KB	Code flash size - CFS2 × 32KB
	NSC	CFS1 × 1KB	CFS2 × 32KB - CFS1 × 1KB
	S	0x0000_0000	CFS1 × 1 KB
Code flash bank1 (Dual mode)	NS	0x0020_0000 + CFS2 × 32KB	Code flash size/2 - CFS2 × 32KB
	NSC	0x0020_0000 + CFS1 × 1KB	CFS2 × 32KB - CFS1 × 1KB
	S	0x0020_0000	CFS1 × 1KB
Code flash bank0 (Dual mode)	NS	CFS2 × 32KB	Code flash size/2 - CFS2 × 32KB
	NSC	CFS1 × 1KB	CFS2 × 32KB - CFS1 × 1KB
	S	0x0000_0000	CFS1 × 1KB



5.4 Cache

The specifications for the RA cache are detailed below, featuring the RA6M5 configuration as an example.

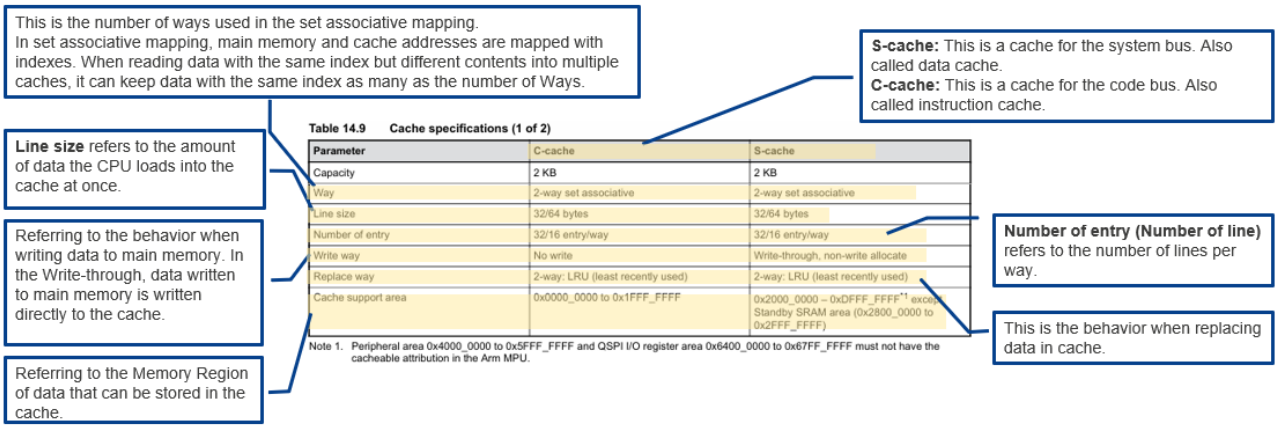


Table 14.9 Cache specifications (1 of 2)

Parameter	C-cache	S-cache
Capacity	2 KB	2 KB
Way	2-way set associative	2-way set associative
Line size	32/64 bytes	32/64 bytes
Number of entry	32/16 entry/way	32/16 entry/way
Write way	No write	Write-through, non-write allocate
Replace way	2-way: LRU (least recently used)	2-way: LRU (least recently used)
Cache support area	0x0000_0000 to 0x1FFF_FFFF	0x2000_0000 – 0xDFFF_FFFF*1 except Standby SRAM area (0x2800_0000 to 0x2FFF_FFFF)

Note 1. Peripheral area 0x4000_0000 to 0x5FFF_FFFF and QSPI I/O register area 0x6400_0000 to 0x67FF_FFFF must not have the cacheable attribution in the Arm MPU.

5.5 Early forwarding function

Early Forwarding Function bring forward the timing of CPU reads data when a data request is made during cache fill. If the Address of cache fill and Address of CPU read request are the same, the CPU reads immediately without waiting for the entire cache fill to complete.

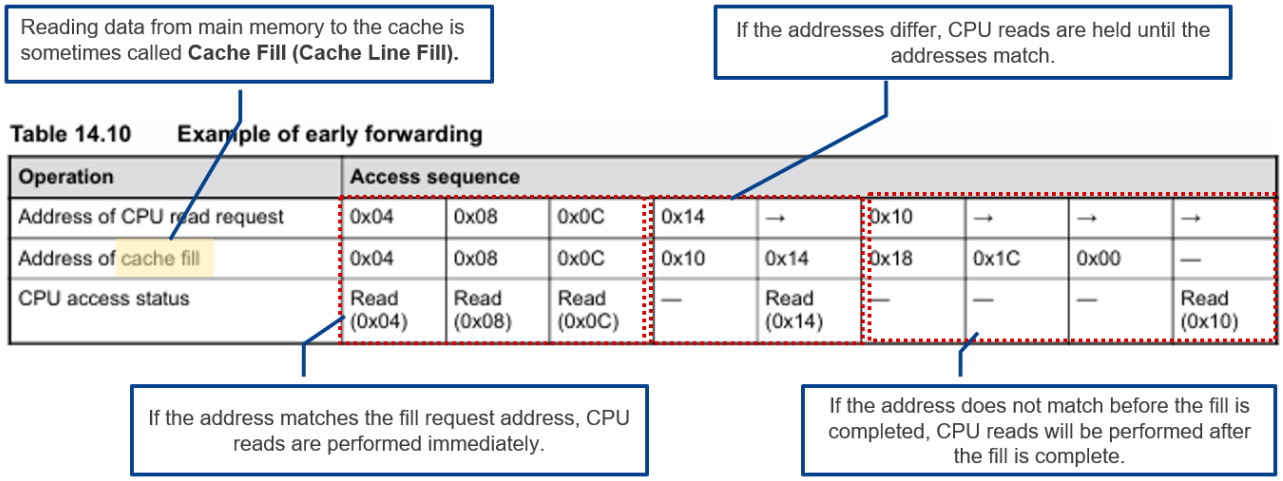


Table 14.10 Example of early forwarding

Operation	Access sequence								
Address of CPU read request	0x04	0x08	0x0C	0x14	→	0x10	→	→	→
Address of cache fill	0x04	0x08	0x0C	0x10	0x14	0x18	0x1C	0x00	—
CPU access status	Read (0x04)	Read (0x08)	Read (0x0C)	—	Read (0x14)	—	—	—	Read (0x10)

5.6 Appendix: Terminology

This is an explanation page for some terms that appear in the "Buses" chapter "Cache" section of manual of each RA product.

(1) Cache Flush

This process occurs when the amount of data in the cache exceeds a certain amount.
After writing data from the cache to main memory, the data in the cache is cleared.

(2) Cache Line

This is the unit of data to be written to the cache.

(3) Coherency

It refers to the consistency of data between main memory and cache. If there is a discrepancy between the main memory and cache data contents, correct processing cannot be performed. Techniques for maintaining coherency include Write-through, where the contents of the main memory write are directly to the cache, and Write-back, where data ejected from the cache is written to main memory.

(4) Cacheability

This attribute is assigned to addresses in main memory that can use cache. Available areas are assigned as Cacheable, while unavailable areas are assigned as non-cacheable.

5.7 List of Application Notes using Buses

Below is an example of an application using an external bus. For details, please refer to each application note.

Apprication note name	Document No
Renesas RA Family Driving an LCD Using the External Memory Bus on the RA6M4	R01AN7454
Renesas RA8 Series Getting Started with RA8 Memory Architecture, Configurations and Topologies	R01AN7088

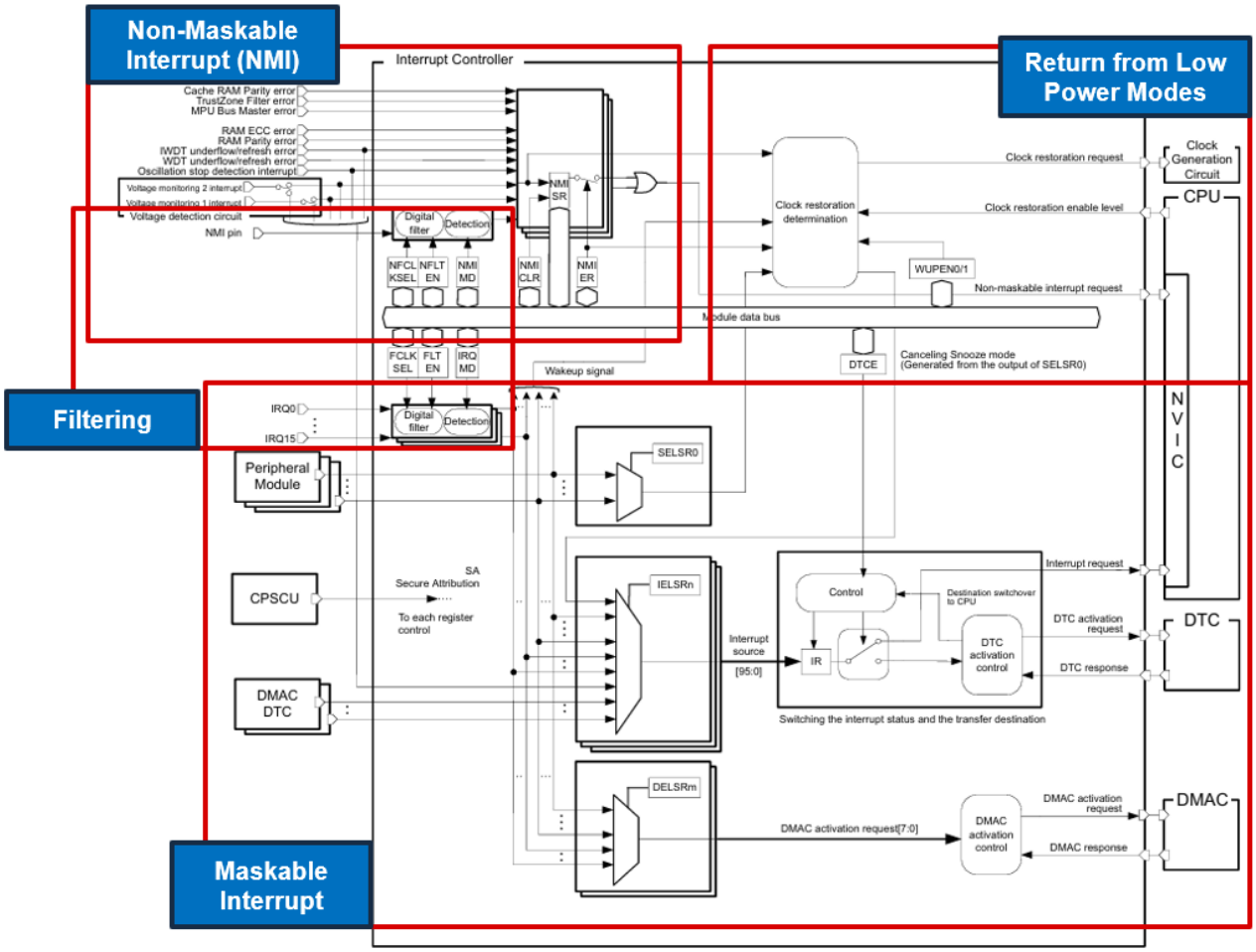
6. Interrupt Controller Unit (ICU)

6.1 Specification summary of Interrupt Controller

Structure of the interrupt controller can be divided into four sections, as shown in the diagram on the right.

This guide explains the following four sections in order.

- Maskable Interrupt
- Non-Maskable Interrupt
- Filtering
- Return from Low Power Modes



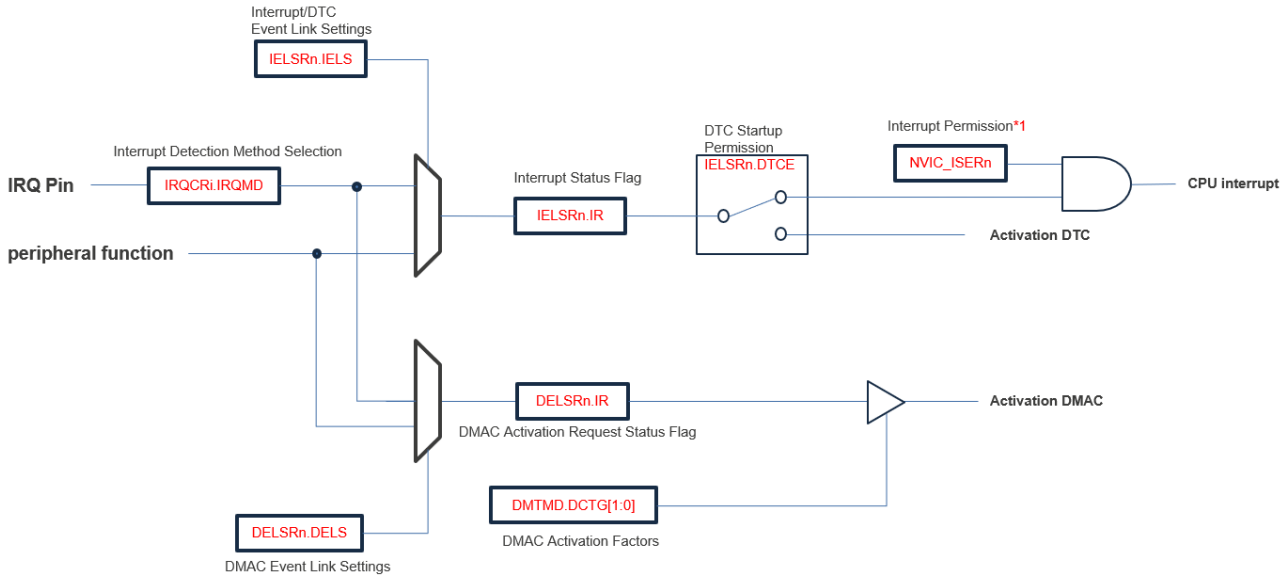
6.2 Maskable interrupt

The features of maskable interrupts are shown below.

- This interrupt occurs when IRQ Pin or peripheral functions act as an interrupt source.
- Maskable interrupt request pins IRQn (n = 0~15) include a digital filter function.
- Interrupt request source (and interrupt name) is managed by Interrupt Vector Table and Event Table.
  For details, please refer to the "Vector Table" section on manual of each RA product.
- Maskable interrupts can trigger CPU interrupts or request a transfer by the DTC/DMAC.
- When generating a maskable interrupt, you need to set not only register for the interrupt function but also registers for peripheral functions and the CPU.
- If the manual only mentions 'interrupt', it refers to a maskable interrupt.
  Unless it is labeled as a non-maskable interrupt, it refers to a non-maskable interrupt.

6.2.1 Maskable interrupt structure

Below are the structures of maskable interrupt and the register names that need to be configured. For details of register, please refer to “Interrupt Controller Unit” chapter “Register Descriptions” section on manual of each RA product. *



*: The diagram shows an example of the RA6M5.

*1: This is the NVIC register.

For detail, please refer to Reference provided at the end of this chapter of this guide.

6.2.2 Vector Table

RA Family is configured to use both interrupt vector tables and event tables.

▪In interrupt vector tables, interrupt request source is not directly assigned. Instead, the correspondence between IRQ numbers and IELSRn registers is defined. You can set an event number for the IELS [8:0] bit in each IELSRn register.

▪The event table defines association between event number and interrupt request source (and interrupt name). By configuring registers, you can assign same interrupt vector to different interrupt request sources.

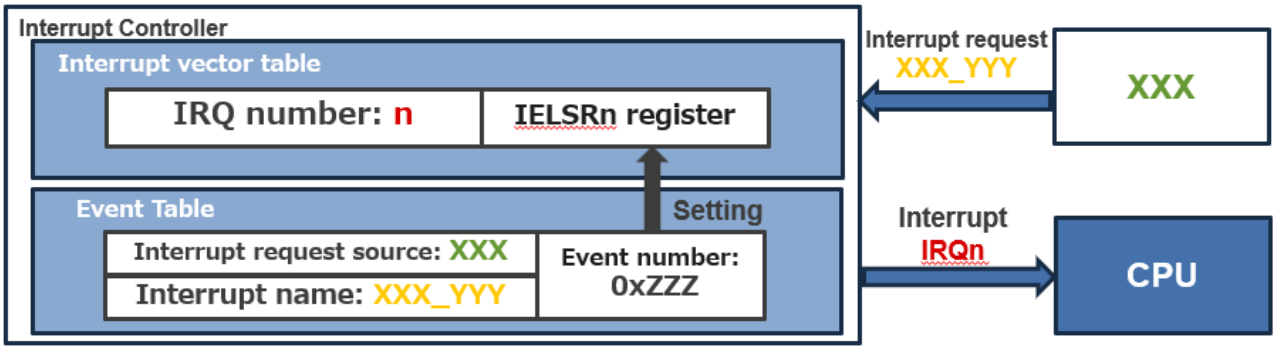


Figure. Example of interrupts associated with vector tables

Below is an example of associating interrupt vectors and interrupt factors within RA Family.

Table 13.3 Interrupt vector table (1 of 3)

Exception number	IRQ number	Vector offset	Source	Description
0	—	0x000	Arm	Initial stack pointer
15	—	0x03C	Arm	System Tick Timer (SysTick)
16	0	0x040	ICU.IELSR0	Event selected in the ICU.IELSR0 register
17	1	0x044	ICU.IELSR1	Event selected in the ICU.IELSR1 register
18	2	0x048	ICU.IELSR2	Event selected in the ICU.IELSR2 register

IRQ0 is assigned to the ICU.IELSR0 register. By default, it is not associated with the origin of specific interrupt request.

Table 13.4 Event table (7 of 9)

Event number	Interrupt request source	Name	IELSRn		DELSRn	Canceling Snooze	Canceling Software Standby	Canceling Deep Software Standby
			Connect to NVIC	Invoke DTC	Invoke DMAC			
0x111	GPT9	GPT9_CCMPA	✓	✓	✓	—	—	—
0x160	ADC120	ADC120_ADI	✓	✓	✓	—	—	—
0x161		ADC120_GBADI	✓	✓	✓	—	—	—
0x162		ADC120_CMPAI	✓	—	—	—	—	—

ICU.IELSR0 By setting the event number in register, source of the IRQ0 interrupt is decided.
IELSR0.IELS[8:0] = 0x160, the interrupt request is determined to originate from ADC120 and named ADC120_ADI.

6.2.3 Selecting Interrupt Request Recipient

The choice of interrupt request destination involves the following three registers.

IELSRn: CU Event Link Setting Register n (n = 0～95)

DELSRn: DMAC Event Link Setting Register n (n = 0～7)

IRQCRi: IRQ Control Register i (i = 0～15)

▪ If the DMAC or DTC is selected as interrupt request destination from IRQi pin, you must configure IRQCRi register to enable edge detection.

▪ If IELSRn is set to “DTC activation is disabled”, events specified by IELSRn will be output to CPU (NVIC).

▪ If IELSRn is set to “DTC activation is enabled”, event specified in the IELSRn register will be output to DTC, and DTC will be activated.

▪ Event specified in DELSRn register is output to DMAC.

▪ After DMAC activates, you can send an interrupt request to CPU. In this case, you need to set DMAC interrupt to IELSRn as the source of request and set “DTC activation is disabled”.

(1) Activation DTC

Supplementary information regarding DTC activation via interrupts is provided below.

Table 13.5 Operation when DTC becomes interrupt request destination

Interrupt request destination	DISEL*1	Remaining transfer operations	Operation per request	IR*2	Interrupt request destination after transfer
DTC*3	1	≠ 0	DTC transfer → CPU interrupt	Cleared on interrupt acceptance by the CPU	DTC
		= 0	DTC transfer → CPU interrupt	Cleared on interrupt acceptance by the CPU	CPU (IELSRn.DTCE bit is automatically cleared)
	0	≠ 0	DTC transfer	Cleared at the start of DTC data transfer after reading DTC transfer data	DTC
		= 0	DTC transfer → CPU interrupt	Cleared on interrupt acceptance by the CPU	CPU (IELSRn.DTCE bit is automatically cleared)

If you want to use DTC as the interrupt request destination, check the settings for the MRB.DISEL bit in addition to DTCST and IELSRn registers.
For event numbers that allow DTC startup, please refer to "Vector Table" section on manual of each RA product.

17.2.3 MRB : DTC Mode Register B

Base address: DTCVBR

Offset address: 0x02 + 0x4 × Vector number
(Inaccessible directly from the CPU. See section 17.3.1. Allocating Transfer Information and DTC Vector Table)

Bit position:	7	6	5	4	3	2	1	0
Bit field:	CHNE	CHNS	DISEL	DTS	DM[1:0]		—	—
Value after reset:	x	x	x	x	x	x	x	x

Bit	Symbol	Function	R/W
5	DISEL	DTC Interrupt Select 0: Generate an interrupt request to the CPU when the specified data transfer is complete 1: Generate an interrupt request to the CPU each time DTC data transfer is performed	—

(2) Activation DMAC

This shows path when DMAC serves as both destination and source for interrupt requests.

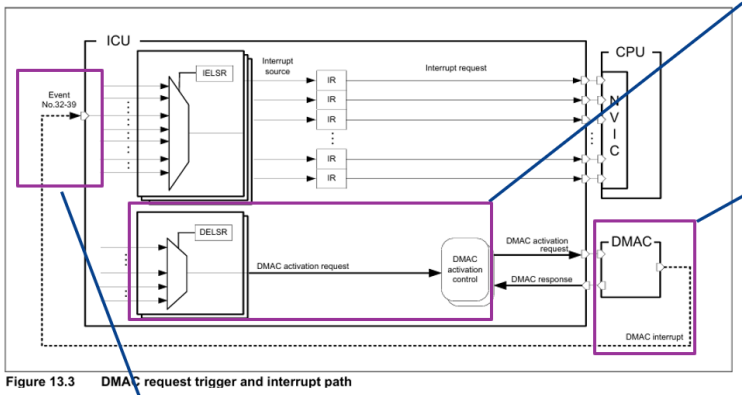


Figure 13.3 DMAC request trigger and interrupt path

If DMAC will be **set as interrupt request destination**, set the event number in **DELSRn register** in addition to setting DMACm.DMTMD register.
For Event numbers that allow DMAC to be activated, please refer to "Vector Table" section on manual of each RA product.

If DMAC will be **set as source of interrupt request**, set event numbers 32~39 (0x020~0x027) in **IELSRn register**.
DMAC can make interrupt requests to CPU (NVIC) or DTC.

Unable to activate DMAC for DMAC interrupts.

Table 13.4 Event table (1 of 9)

Event number	Interrupt request source	Name	IELSRn		DELSRn
			Connect to NVIC	Invoke DTC	Invoke DMAC
0x020	DMAC0	DMAC0_INT	✓	✓	—
0x021	DMAC1	DMAC1_INT	✓	✓	—
0x022	DMAC2	DMAC2_INT	✓	✓	—
0x023	DMAC3	DMAC3_INT	✓	✓	—
0x024	DMAC4	DMAC4_INT	✓	✓	—
0x025	DMAC5	DMAC5_INT	✓	✓	—
0x026	DMAC6	DMAC6_INT	✓	✓	—
0x027	DMAC7	DMAC7_INT	✓	✓	—

6.3 Non-Maskable Interrupt

Non-maskable interrupts are summarized below.

▪This interrupt occurs when a CPU abnormality is detected and a catastrophic failure may have occurred in system.

▪In RA Family, all of following interrupts are collectively defined as non-maskable interrupts.

NMI Pin	Oscillation Stop Detection	IWDT/WDT Underflow/Refresh Error	SRAM ECC Error	Bus Master MPU Error
Cache RAM Parity Error	TrustZone Filter Error	Low Voltage Detection 1/2	SRAM Parity Error	

▪It can only be used on CPU and cannot be used to boot DTC or DMAC.

▪It takes precedence over all other interrupts.

▪By default, it is prohibited. To use it, you need to set up NMIR register.

▪If a non-maskable interrupt occurs, there may be a system abnormality, so please consider a mechanism that prevents reverting to the original process after interrupting.

▪ A digital filter function is provided for NMI pin interruption.

▪The following interrupts can also be used as maskable interrupts.

WDT Underflow

IWDT Underflow

Low Voltage Detection 1

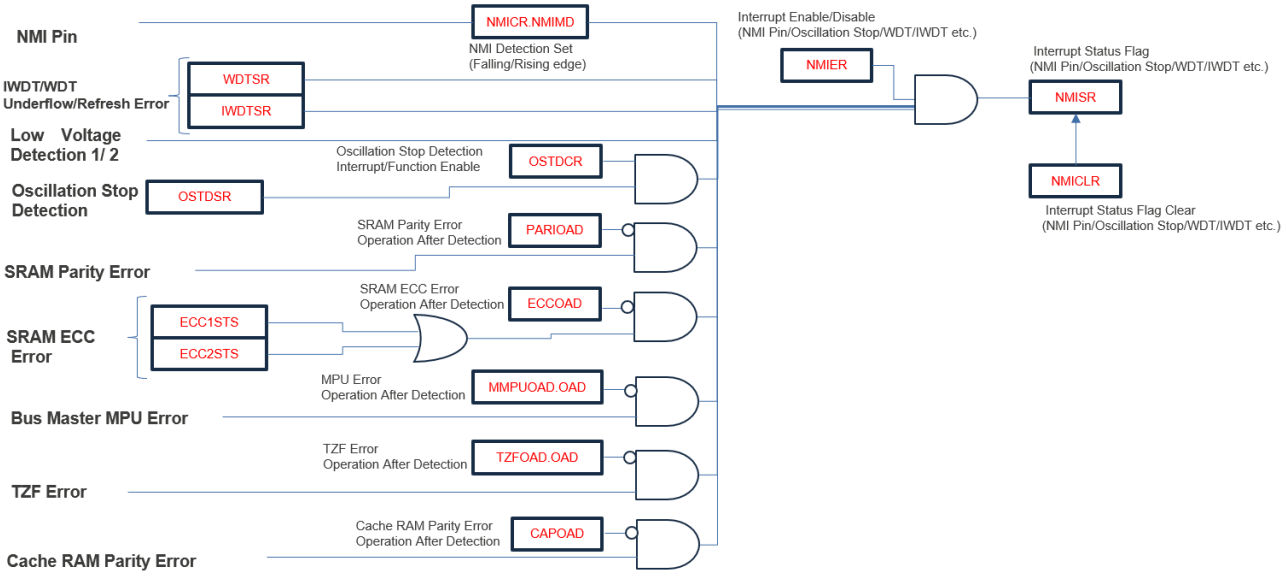
Low Voltage Detection 2

Oscillation Stop Detection interrupt

These interrupt request sources (and interrupt names) are defined in the event table. For detail, please refer to “Event Number" section on manual of each RA product.

6.3.1 Non-maskable interrupt structure

Below are structures of the non-maskable interrupt and the register names that need to be configured. For details of register, please refer to “Interrupt Controller Unit” chapter “Register Descriptions” section on manual of each RA product. *



* : The diagram shows an example of the RA6M5.

6.4 Digital Filter

▪ NMI pins and IRQ pins can remove noise through filtering.

▪ Only input signal with matching three times in a row level passes through.

▪Noise filter processing cannot be used in the following modes. Therefore, when using the filtering, please disable filter before these modes migration and enable it after disabling the mode.

Software Standby mode

Deep Software Standby mode

Snooze mode

When using filtering, registers and bits that need to be configured are shown below.

Input Type	Digital Filter Enable/Disable	Digital Filter Sampling Clock Select	Detection Sense Setting
NMI pin (NMICR register)	NFLTEN	NFCLKSEL	NMIMD
IRQ pin (IRQCRi register)	FLTEN	FCLKSEL	IRQMD

6.5 Returning from low-power consumption mode

Interrupt can be used to return from each low-power consumption mode. Below is a summary of important points to keep in mind when using interrupts to return from low-power modes.

Table. Notes on returning from low-power consumption mode by interrupt

	Sleep mode	Software Standby mode	Deep Software Standby mode	Snooze mode
Operation after interrupt	Execute interrupt used as return factor	Execute interrupt used as return factor	Perform internal reset	Execute interrupt used as return factor
Factors for Comeback	All interrupts	Partial interrupts*1	Partial interrupts*1	Partial interrupts*1
Digital filter	Enable	Disable	Disable	Disable
Registers that require configuration	NMIER register NVIC_ISERn register*2	NMIER register WUPEN register NVIC_ISERn register*2	NMIER register WUPEN register NVIC_ISERn register*2	SELSR0 register IELSRn register NVIC_ISERn register*2

*1: For selectable interrupts as factor returnable from each mode, see table “Interrupt Source for canceling Snooze, Software Standby, and Deep Software Standby Modes “in "Low Power Modes “chapter and table "Event table" in "Interrupt Controller Unit (ICU) " chapter, on manual of each RA products.

*2: This is the NVIC register. For detail, please refer to Reference provided at the end of this chapter of this guide.

▪Returning from sleep mode is possible from any interruption source.

▪In software standby mode and deep software standby mode, there are only a limited number of interrupts that can be used as recovery factors. For details, please refer to the “Recovery from Low Power State” section on manual of each RA product.

▪When running in low-power mode other than deep software standby mode, even if an event selected by IELSRn and DELSRn is detected, interrupt processing will not be performed immediately. Actual handling of interrupts by CPU and DMAC (acknowledges) is performed after resuming normal mode.

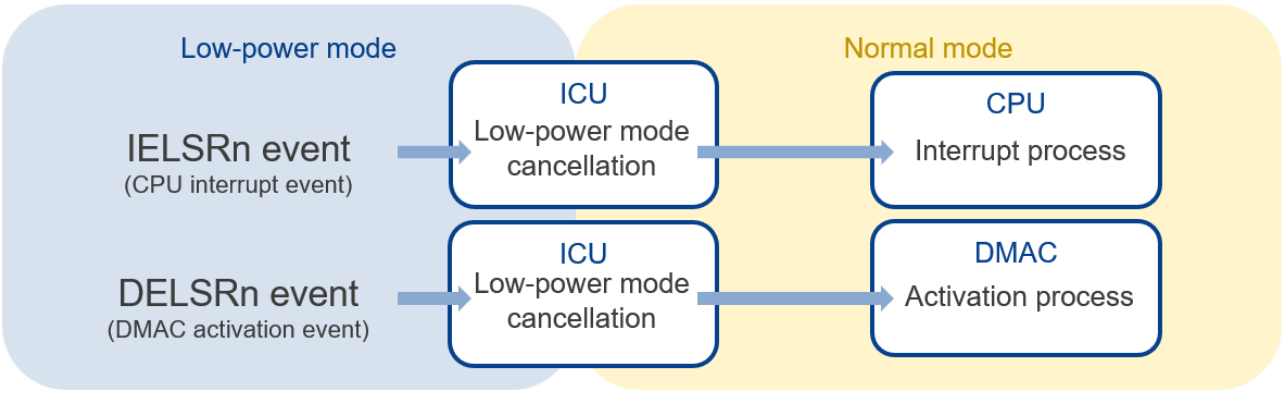


Table. Returning operation from low-power mode other than deep software standby mode

6.6 Reference

For information on NVIC's internal registers, please refer to following manual.

▪ARM Limited., ARM® Cortex®-M33 Processor Technical Reference Manual (ARM 100230)

You can download it from the official ARM website.

Arm Cortex-M33 Processor Technical Reference Manual

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Apr. 2026	-	First edition issued
2.00	Jul. 2026	16~42	Added event link controller, external bus controller, and interrupt controller.

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

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

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

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

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

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

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

(Rev.5.0-1 October 2020)

Corporate Headquarters

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

Trademarks

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

Contact information

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