Renesas RA Family

RA0 Quick Design Guide

Introduction

This document answers common questions and points out subtleties of the RA0 MCU that might be missed unless the hardware manual was extensively reviewed. The document is not intended to be a replacement for the hardware manual. It is intended to supplement the manual by highlighting some key items most engineers will need to start their own design. It also discusses some design decisions from an application point of view.

Target Device

RA0 MCU Series

Contents

1. Power Supplies

The RA0 family has digital power supplies and analog power supplies. The power supplies use the following pins.

Table 1. Digital Power Supplies

Symbol	Name	Description
VCC	Power supply	Power supply pin. Connect to the system power supply. Connect this pin to VSS via a 0.1 µF capacitor placed close to the VCC pin.
VSS	Ground	Ground.
VCL	Power supply	Connect this pin to VSS via a 0.47 µF to 1.0 µF capacitor close to the VCL pin.

Table 2. Analog Power Supplies

Symbol	Name	Description
VREFH0	12-bit ADC high reference voltage	Reference voltage input pin for the 12-bit A/D when not used as GPIO.
VREFL0	12-bit ADC low reference voltage	Analog reference ground pin for the 12-bit A/D converter when not used as GPIO.

1.1 References

Further information regarding the power supply for the RA MCU Group can be found in the following documents:

- R01UH1040 RA0E1 Group, *RA0E1 Group User's Manual: Hardware*

The **Overview** chapter lists power pins in each package with recommended bypass capacitors.

The **Resets** chapter discusses the Power-On Reset and how to differentiate this from other reset sources.

The **Low Voltage Detection** chapter provides details on the Low-Voltage Detection Circuit that can be used to monitor the power supply. The **Option-Setting Memory** chapter additionally describes how to enable Low-Voltage Detection 0 Circuit automatically at startup.

If you plan to use the on-chip Analog to Digital Converters (ADC), refer to the **12-Bit A/D Converter (ADC12)** chapter in the Hardware User's Manual for details on how to provide filtered power supplies for these peripherals.

Table 3. RA0 MCU Groups, User's Manual: Hardware

Chapter Name	Description
Overview	Lists power pins in each package with notes on termination and bypassing.
Resets	Discusses the Power-On Reset and how to differentiate this from other reset sources.
Low Voltage Detection	Provides details on the Low-Voltage Detection Circuit that can be used to monitor the power supply.
Low Power Modes	Using low power modes may allow you to reduce the power consumption of the MCU. See this chapter for details on how operating modes affect power supply requirements.
12-Bit A/D Converter	If you plan to use the on-chip A/D converter, these chapters give details on how to provide filtered power supplies for these peripherals.
Clock Generation Circuit	Provides detailed descriptions on how to configure and use the available clocks, including PCB design recommendations.

2. Emulator Support

RA0 MCU devices have an emulator interface that supports debugging using SWD communication.

The SWD emulator interface can be connected to an Arm®-standard 10-pin or 20-pin socket.

To comply with the Arm® specification, pull up resistors are required on the SWD signals. Without the correct pull up resistors, the interface may not function correctly. However, RA0 MCU devices have internal pull up resistors that are enabled by default for these signals. When the internal pull up resistors are enabled, no external resistors are required on these signals.

Emulator support is useful for product development and prototyping, but may not be needed once a design moves to production. If emulator support is no longer needed for a design, make sure to configure the ports according to the *Handling of Unused Pins* section of the MCU Hardware User's Manual. Also see section 9.5 in this document.

2.1 SWD Interface

The following diagram shows the typical connectivity of the debug interface when using Serial Wire Debug (SWD).

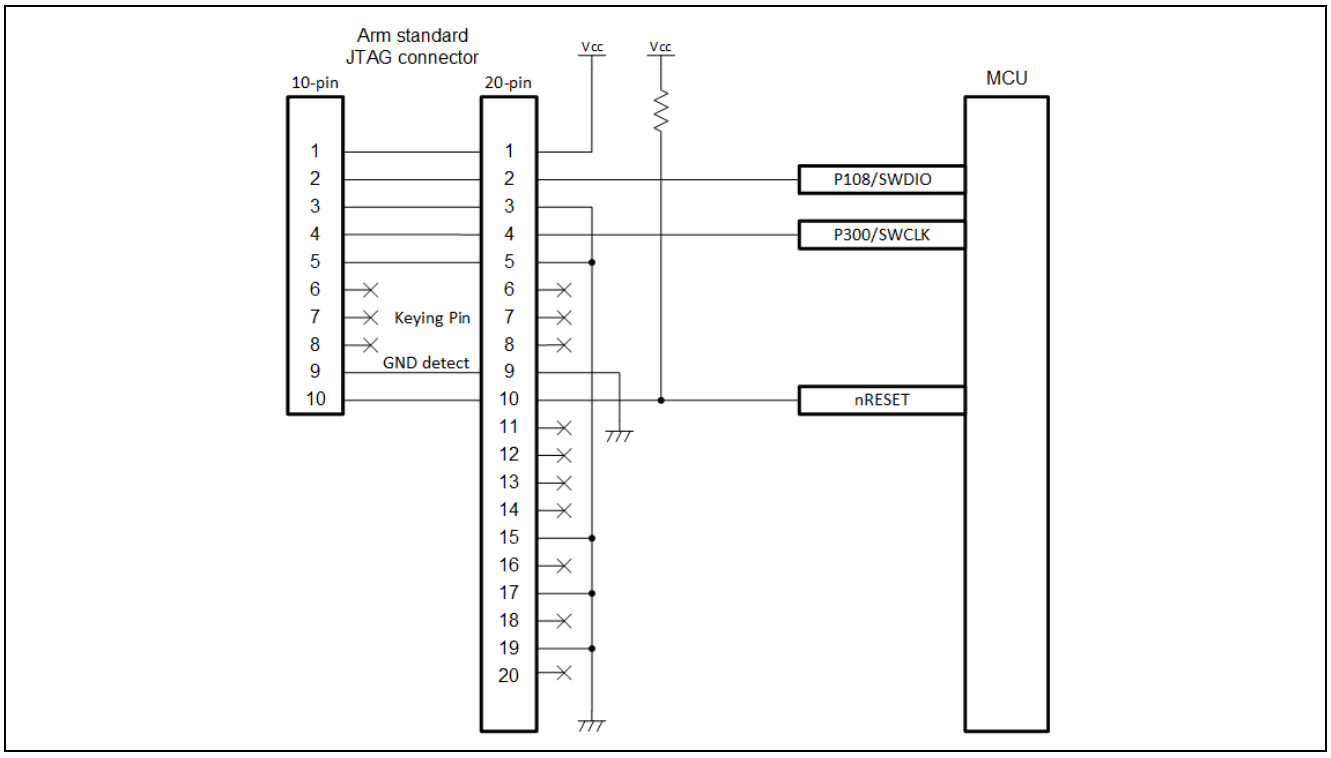


Figure 1. SWD Interface Connections

Note: The output of the reset circuit of the user system must be open collector.

3. MCU Operating Modes

After reset, the RA0 MCU series will enter Single-chip mode. This will occur for both RES pin resets and POR resets.

Figure 2 shows the operating mode transitions.

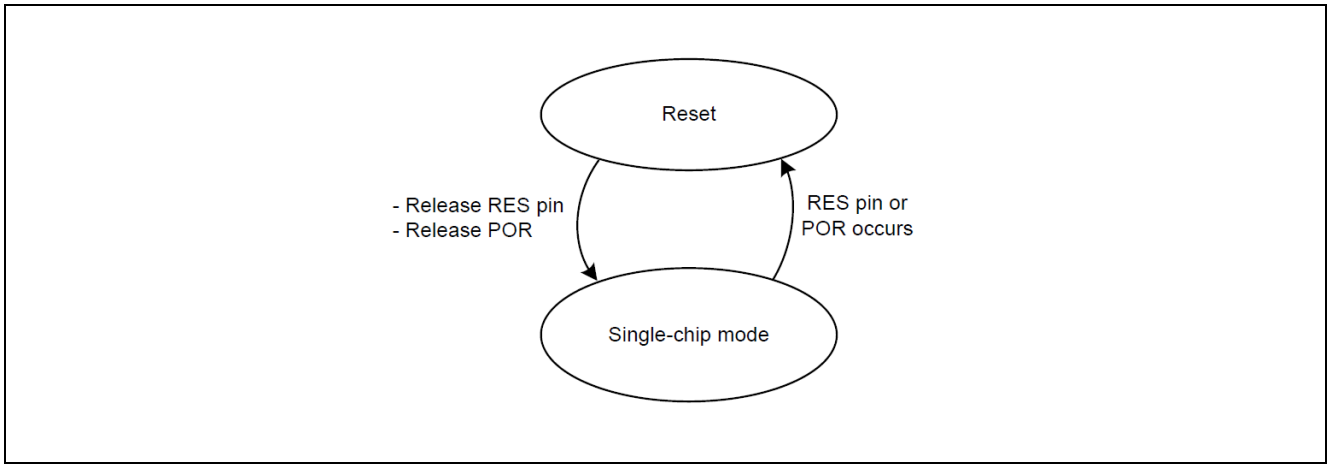


Figure 2. Operating Mode Transitions

It is important to note that RA0 MCU devices do not have an MD (Mode) pin. This is different from other Renesas RA devices. RA0 MCU devices do not support SCI Boot Mode, so there is no need for the MD pin.

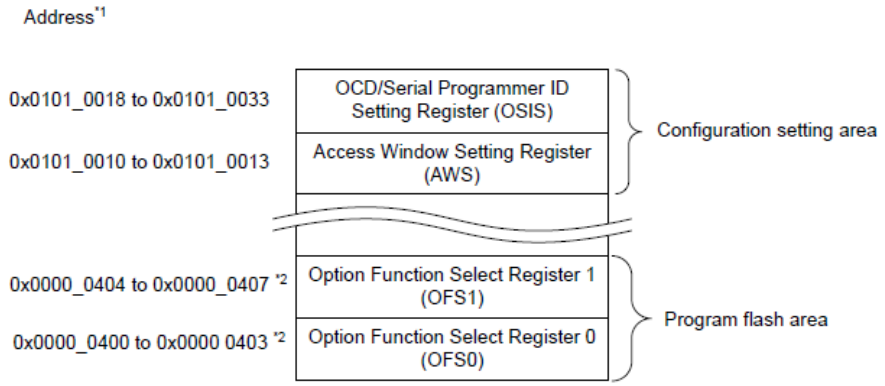
4. Option-Setting Memory

The option-setting memory determines the state of the MCU after a reset. It is allocated to the configuration setting area and the program flash area of the flash memory. The available methods for configuration are different for the two areas.

The registers are detailed in the *Option-Setting Memory* chapter in the Hardware User's Manual.

Options-Setting Memory registers occupy space in the code flash memory map. Although the registers are located in a portion of the flash memory that is reserved on the RA MCUs, **it is possible that some customers may store data in these locations inadvertently**. The user must check to ensure that no unwanted data is written to these locations or else unexpected behavior of the chip may result. Additionally, when using binary files for programming, the user must ensure that reserved areas of memory are not programmed due to the addresses of the configuration setting area. For instance, settings in the option-setting memory can enable the Independent Watchdog Timer (IWDT) immediately after reset. If data stored in program ROM inadvertently overlaps the option-setting memory register, it is possible to turn on the IWDT on without realizing it. This will cause the debugger to have communications problems with the board.

The figure below shows the option-setting memory area, which consists of the option function select registers on RA0E1. The Option-Setting Memory may be different for each device. Please consult MCU User's Manual for the specific device details.



Note 1. The option-setting memory must be allocated to the user area of the flash memory.
Note 2. The address of these registers will be changed when the boot swap is set. See section 6.2.1. OFS0 : Option Function Select Register 0 and section 6.2.2. OFS1 : Option Function Select Register 1 for details.

Figure 3. Option-Setting Memory for RA0E1

4.1 Option-Setting Memory Registers

The following is a summary of the Option-Setting Memory registers. Make sure that they are configured properly before startup.

- OFS0 register
  - Independent Watchdog Timer (IWDT) auto start
  - IWDT timeout, frequency, windowing, interrupt type, and low power mode behavior
- OFS1 register
  - LVDAS (LVD0 Circuit Start) enable after reset.
  - HOCO startup settings after reset.
  - Port 206 / RES function selection.
  - Flash Read Protection after reset.

Renesas FSP Configurator supports setting of Option Function Select Registers in BSP settings, as shown in Figure 4 for RA0E1 MCU.

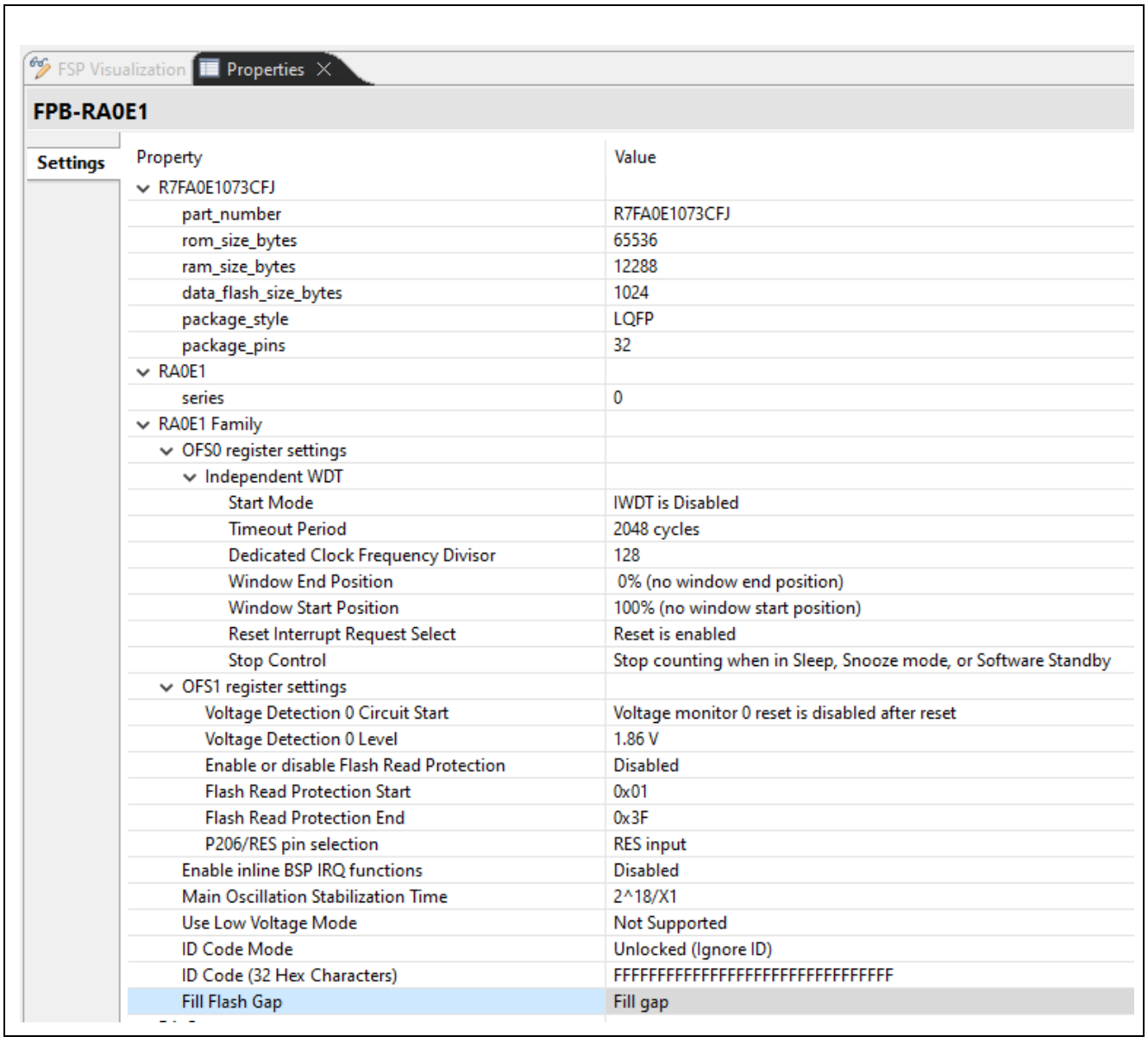


Figure 4. Option Function Select Register Settings in FSP Configuration for RA0E1 MCU

5. Clock Circuits

RA0 MCUs have five primary oscillators. All five of these may be used as the source for the main system clock. In a typical system, the main clock is driven with an external crystal or clock. This input is directed to internal selectors and frequency dividers, where it is further directed to the main system clock (ICLK), flash clock, CPU clock, and peripheral module clocks.

Note: Some RA0 devices may multiplex the Main Clock Oscillator and Sub Clock Oscillator input pins. Refer to the *Clock Generation Circuit* chapter in the *Hardware User's Manual's* for the block diagram of the clock generation circuit.

Each clock has specific tolerances and timing values. Refer to the Hardware User's Manual's AC Characteristics section in the Electrical Characteristics chapter for the Frequency and Clock Timing specifications. Refer to the Hardware User's Manual's Clock Generation Circuit chapter for the relationship between the various clock frequencies.

Table 4. RA0 Oscillator Source Specifications

Oscillator	Input Source	Frequency	Primary Uses
Main clock (MOSC)	External crystal/resonator -or-	1 MHz to 20 MHz	System clock (ICLK): To CPU, Flash, SRAM, Flash-IF, Peripheral module clock
	External clock	Up to 20 MHz	
Sub-clock (SOSC)	External crystal/resonator	32.768 kHz	Real-time clock, system clock (ICLK) in low power modes, PCLBUZ0, TAU clock, UARTA clock, TML32 clock.
High-speed on-chip (HOCO)	On-chip oscillator	24/32 MHz	System clock (ICLK) at startup, CPU, Flash, SRAM, Flash-IF, Peripheral module clock.
Middle-speed on-chip (MOCO)	On-chip oscillator	1 MHz to 4 MHz	System clock (ICLK), Peripheral clocks, PCLBUZ0, TAU clock, TML32 clock, UARTA clock.
Low-speed on-chip (LOCO)	On-chip oscillator	32.768 kHz	System clock (ICLK) in low power modes, Peripheral clocks, Systick timer, TAU, TML32 clock, UARTA clock, PCLBUZ0, Real-Time clock, Independent Watchdog Timer clock[1]

Note 1: The LOCO will oscillate if the IWDT is enabled, regardless of the LOCO register (LOCOCR) settings.

Some RA0 devices may multiplex the Main Clock Oscillator and Sub Clock Oscillator input pins. In this case, users must carefully consider the tolerances of the on-chip oscillators as noted in the Oscillators Characteristics section in the MCU Hardware User's Manual.

5.1 Reset Conditions

After reset, RA0 MCUs begin running with the high-speed on-chip oscillator (HOCO) as the main clock source. At reset, the main clock oscillator is stopped by default and the pins are set to port mode. The SOSC, LOCO, MOCO, are stopped by default after reset. The IWDT may be on or off depending on the settings in the Option-Setting Memory (see section 4).

5.2 Clock Frequency Requirements

Minimum and maximum frequencies are shown in the following tables. Details can be found in the Overview section of the Clock Generation Circuit chapter in the MCU Hardware User's Manual, including external and internal clock source specifications. Additional details on the tolerances of each clock input source can be found in the *Oscillator Characteristics* section of the *Electrical Characteristics* chapter in the MCU Hardware User's Manual.

Table 5. Frequency Range for RA0 MCU Internal Clocks

	ICLK [1]	RTCCLK	IWDTCLK	SWCLK	CLKOUT	SYSTICCLK
Max Frequency [Hz]	32M	32768	16384	12.5M	16M	32768
Min Frequency [Hz]	—	128	16384	—	—	32768

Note 1: ICLK must run at a frequency of at least 1 MHz when programming or erasing ROM or data flash.

5.2.1 Requirements for Programming and Erasing ROM or Data Flash

For all RA0 devices, the ICLK must be at least 1 MHz to perform programming and erasing on internal ROM and data flash.

5.3 Lowering Clock Generation Circuit (CGC) Power Consumption

To aid in saving power, set the dividers for any unused clocks to the highest possible value whenever possible. Also, if not using a clock, then make sure that it has been stopped by setting the appropriate register(s). The registers for controlling each clock source are shown in the table below.

Table 6. Clock Source Configuration Registers

Oscillator	Register	Description
Main clock	MOSCCR	Starts/stops main clock oscillator
Sub-clock	SOSCCR	Starts/stops sub-clock oscillator
High-speed on-chip (HOCO)	HOCOCR	Starts/stops HOCO
Middle-speed on-chip (MOCO)	MOCOCR	Starts/stops MOCO
Low-speed on-chip (LOCO)	LOCOCR	Starts/stops LOCO

After a clock source has been started, a wait period is required to be observed before using it as a source for an internal clock. The status of the wait period can be read and controlled through registers available in the CGC peripheral block such as HOCOSF, and OSTSB.

All internal clocks can be output to the Clock/Buzzer output pins. This allows measuring the clock frequency accurately.

The frequency of all internal on-chip oscillator sources can be trimmed using special function registers HIOTRM, MIOTRM, and LIOTRM. The clock correction resolution of trimming registers, along with other applicable parameters, are specified in the On-chip Oscillators Characteristics sub-section of the Electrical Characteristics.

5.4 Writing the System Clock Control Registers

Care should be taken when writing to the individual bit fields in the clock control registers. Follow the register list, ICLK source diagram and timing diagram for clock source switching shown in the section *System Clock (ICLK)* in the Hardware User's Manual.

When the clock source of the system clock (ICLK) is switched, the duration of the system clock cycle becomes longer during the clock source transition period. See Figure 5.

To ensure correct processing after the clock frequency changes, first write to the relevant Clock Control registers to change the frequency, then read the value from the register, and finally perform the subsequent processing.

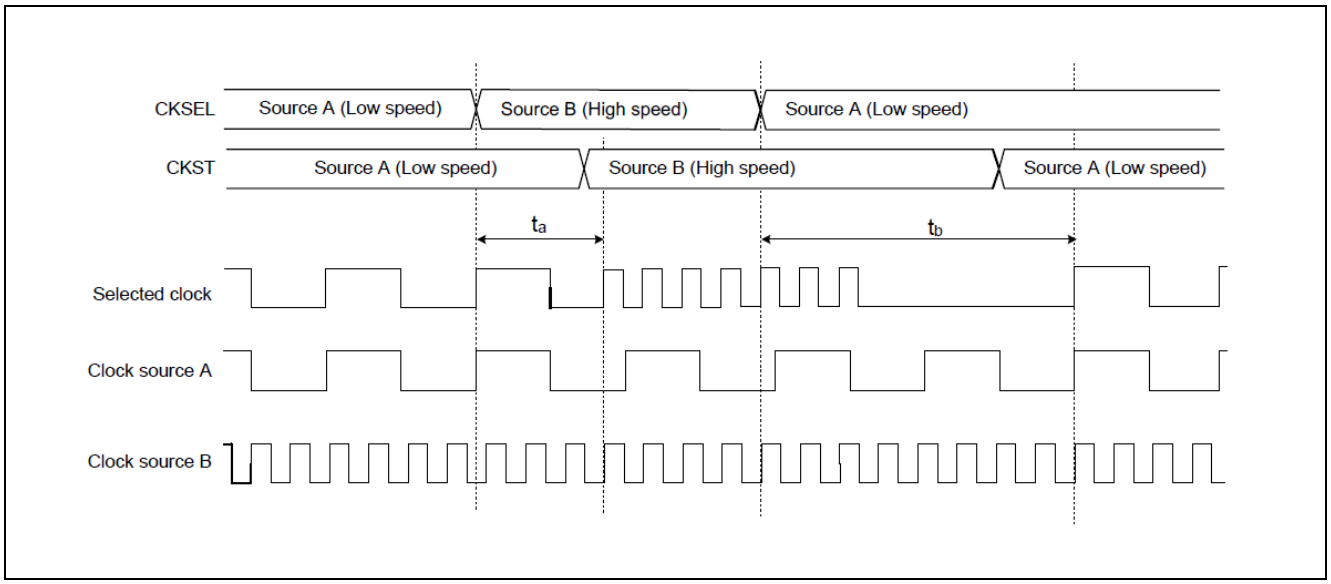


Figure 5. Timing of Clock Source Switching

5.5 Clock Setup Example

Renesas FSP provides a simple, visual clock configuration tool for all RA0 MCUs. An example for RA0E1 MCU group is shown below. This configurator configures code in the board support package to initialize the Clock Generation Circuit based on user selection, with proper precautions as indicated in the MCU Hardware User's Manual.

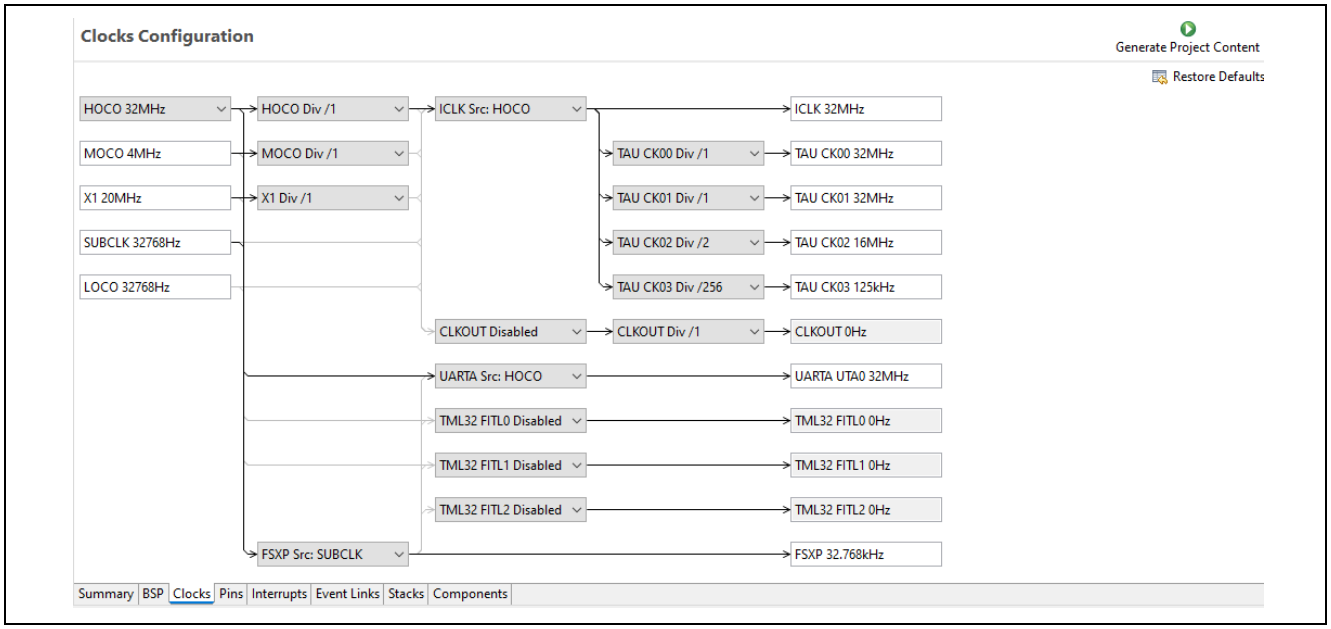


Figure 6. Clock Settings Using Renesas FSP Configurator

5.6 HOCO Accuracy

The internal high-speed on-chip oscillator (HOCO) runs at 24 MHz or 32 MHz for RA0 devices. Once the HOCO Clock Oscillation Stabilization Flag (HOCOSF) has been set to 1, the HOCO will have a typical accuracy of +/-1% or better. See the Electrical Characteristics of the *Hardware User's Manual* for the clock accuracy specifications.

The HOCO may be used as an input to the clock generation circuit. When the HOCO is used this way, no external oscillator is required. This may be an advantage when space constraints or other limitations require a reduced component count in a PCB design. However, there are performance tradeoffs and limitations due to the clock accuracy which should be evaluated for your application.

5.7 Flash Interface Clock

For RA0 devices, the System Clock (ICLK) is used as the operating clock when programming and erasing the internal flash.

Therefore, the frequency setting of the relevant clock will have a direct impact on the amount of time it takes to read from the data flash. If the user's program is reading from the data flash, or performing programming or erasures on internal flash, then using the maximum ICLK frequency is recommended.

The system clock must run at a frequency of at least 1 MHz when writing or erasing ROM or data flash. Please note that the clock frequency does not have any impact upon reading from ROM or reading and writing to RAM.

5.8 Board Design

Refer to the *Usage Notes* section of the *Clock Generation Circuit (CGC)* chapter in the Hardware User's Manual for more information on using the CGC and for board design recommendations.

In general, place the crystal resonator and its load capacitors as close to the MCU clock pins (X1/X2, XCIN/XCOUT) as possible. Avoid routing any other signals between the crystal resonator and the MCU. Minimize the number of connecting vias used on each trace.

5.9 External Crystal Resonator Selection

An external crystal resonator may be used as the main clock source for most RA0 devices. The external crystal resonator is connected across the EXTAL and XTAL pins of the MCU. The frequency of the external crystal resonator must be in the frequency range of the main clock oscillator.

Selection of a crystal resonator will be largely dependent on each unique board design. Due to the large selection of crystal resonators available that may be suitable for use with RA0 MCU devices, carefully evaluate the electrical characteristics of the selected crystal resonator to determine the specific implementation requirements.

The following diagram shows a typical example of a crystal resonator connection.

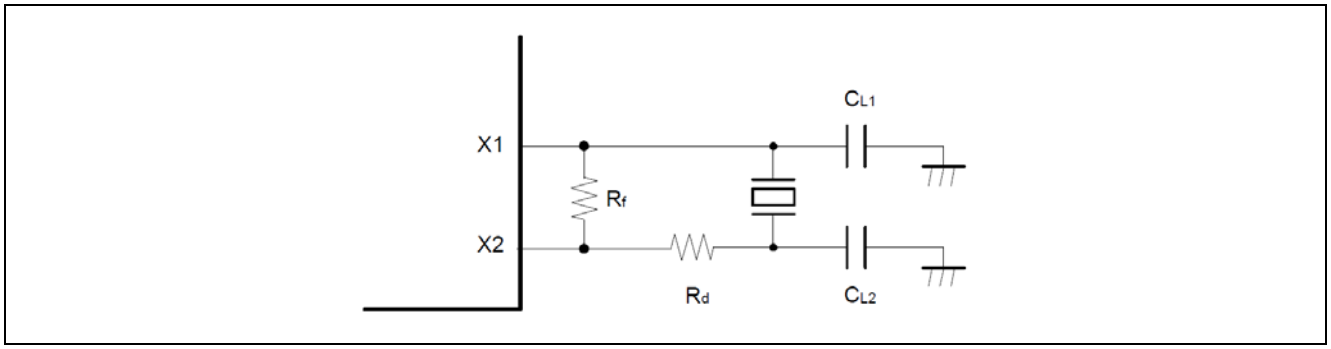


Figure 7. Example of Crystal Resonator Connection

Careful evaluation must be used when selecting the crystal resonator and the associated capacitors. The external feedback resistor (R_f) and damping resistor (R_d) may be added if recommended by the crystal resonator manufacturer.

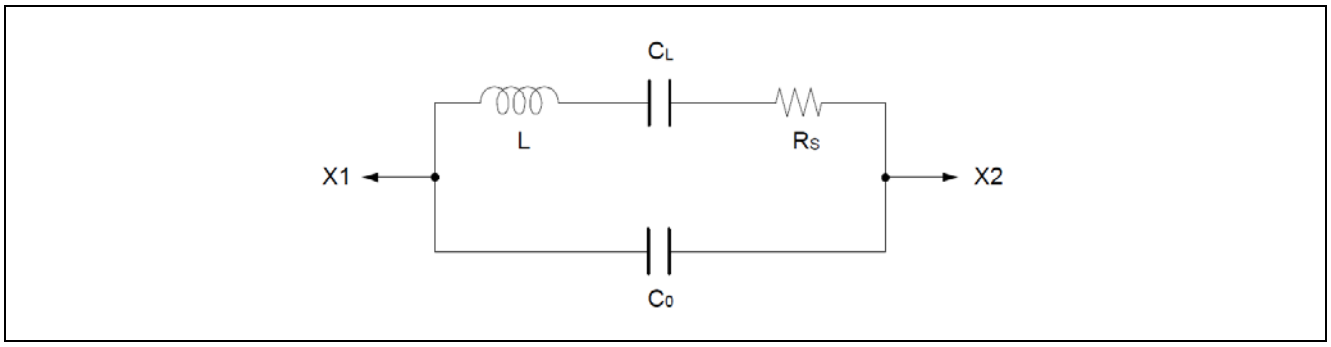


Figure 8. Equivalent Circuit of the Crystal Resonator

Selection of the capacitor values for CL1 and CL2 will affect the accuracy of the internal clock. To understand the impact of the values for CL1 and CL2, the circuit should be simulated using the equivalent circuit of the crystal resonator in the figure above. For more accurate results, also take into account the stray capacitance associated with the routing between the crystal resonator components. The Renesas RA Family Design Guide for Sub-Clock Circuits provides details on board design specification for both the external crystal and sub-clock oscillator.

5.10 External Clock Input

A digital clock input may be used as the main clock source for most RA0 devices. Figure 9 shows an example of connecting an external clock input. To operate the oscillator with an external clock signal, set the CMC:MOSEL[1:0] bits to 11b. The X1 pin becomes high impedance.

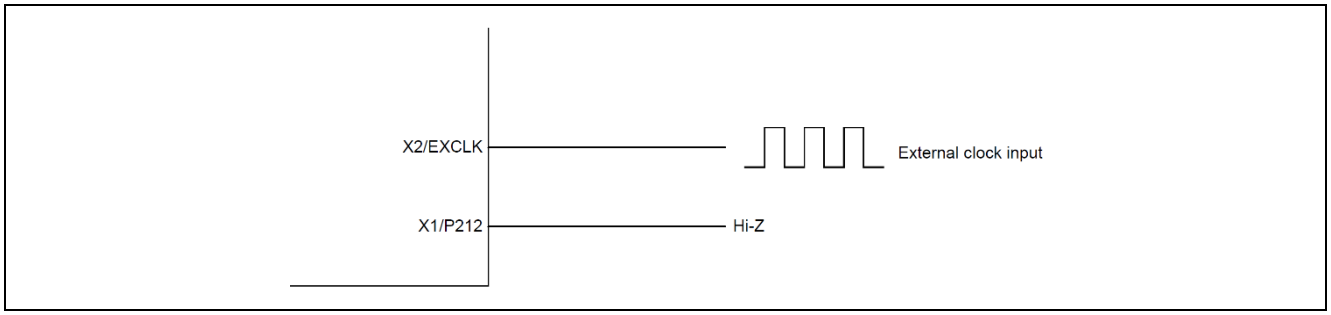


Figure 9. Equivalent circuit for external clock

Note: The frequency of the external clock input can only be changed when the main clock oscillator is stopped. Do not change the frequency of the external clock input when the setting of the Main Clock Oscillator Stop bit (MOSCCR.MOSTP) is 0.

6. Reset Requirements and the Reset Circuit

There are 7 types of resets for RA0 Arm® Cortex®-M23 devices.

Table 7. RA0 Device Resets

Reset Name	Source
Pin reset	RES is driven low
Power-on reset	VCC rises (voltage detection: VPOR)
Independent watchdog timer reset	The independent watchdog timer underflows, or a refresh occurs
Voltage monitor 0 reset	VCC falls (voltage detection Vdet0)
Voltage monitor 1 reset	VCC falls (voltage detection Vdet1)
SRAM parity error reset	SRAM parity error detection
Software reset	Register setting

6.1 Pin Reset

When the RES pin is driven low, all processing is aborted and the MCU enters a reset state. To reset the MCU while it is running, hold RES low for the specified reset pulse width. Refer to the Reset Timing section of the *Electrical Characteristics* chapter of the *Hardware User's Manual* for more detailed timing

requirements. Also refer to section 2, Emulator Support for details on reset circuitry in relation to debug support.

There is no need to use an external capacitor on the RES line because the POR circuit holds it low internally for a good reset and a minimum reset pulse is required to initiate this process.

Once the power is turned on, P206 functions as the RES input. The PORTSELB bit of the option select register1 (OFS1) defines whether this port operates as P206 or RES. When this pin is set to P206, do not input the low level to this pin during a reset by the power-on-reset (POR) circuit and during the period from release from the reset by the POR circuit to the start of normal operation. If input of the low level continues during this period, the chip will remain in the reset state in response to the external reset. The on-chip pull-up resistor is enabled after power is turned on.

6.2 Power-On Reset

There are two conditions that will generate a power-on reset (POR):

1. If the RES pin is in a high-level state when power is supplied.
2. If the RES pin is in a high-level state when VCC is below V_{POR}.

After VCC has exceeded the power-on reset voltage (V_{POR}) and the power-on reset time (t_{POR}) has elapsed, the chip is released from the power-on reset state. The power-on reset time is a period that allows for stabilization of the external power supply and the MCU. Refer to the POR and LVD Characteristics section of the *Electrical Characteristics* chapter of the *Hardware User's Manual* for voltage level and timing details.

Because the POR circuit relies on having RES high concurrently with VCC, don't place a capacitor on the reset pin. This will slow the rise time of RES in relation to VCC, preventing the POR circuit from properly recognizing the power-on condition.

If the RES pin is high when the power supply (VCC) falls to or below V_{POR}, a power-on reset is generated. The chip is released from the power-on state after VCC has risen above V_{POR} and the t_{POR} has elapsed.

After a power on reset, the PORF bit in PORSR is set to 0. To properly detect a power on reset, this bit must be set to 1 by the user before a power on reset occurs.

6.3 Independent Watchdog Timer Reset

This is an internal reset generated by the Independent Watchdog Timer (IWDT).

When the IWDT underflows, an independent watchdog timer reset is optionally generated (NMI can be generated instead) and the IWDTRF bit in RSTSR1 is set to a 1. After a short delay the IWDT reset is canceled. Refer to MCU User's Manual for the specific timing.

6.4 Voltage-Monitoring Resets

The RA0 MCU family includes circuitry that allows the MCU to protect against unsafe operation during brownouts. On-board comparators check the supply voltage against two reference voltages, V_{det0} and V_{det1}. As the supply dips below each reference voltage an interrupt or a reset can be generated. The detection voltage V_{det0} is selectable from 6 different levels in the OFS1 register. The detection voltage V_{det1} is selectable from 18 different levels in the LVD1CR register.

When VCC subsequently rises above V_{det0} or V_{det1}, release from the voltage-monitoring reset proceeds after a stabilization time has elapsed.

Low Voltage Detection is disabled after a power on reset. Voltage monitoring can be enabled by using the Option Function Select register (OFS1) for V_{det0}, or the LVD1CR register for V_{det1}. For more details, see the chapter Low Voltage Detection (LVD) in the Hardware User's Manual.

After an LVD Reset, the LVIRF bit in RESF is set to 1.

6.5 Software Reset

The software reset is an internal reset generated by a software setting of the SYSRESETREQ bit in the AIRCR register in the Arm core. When the SYSRESETREQ bit is set to 1, a software reset is generated. When the internal reset time (tRESW2) elapses after the software reset is generated, the internal reset is canceled and the CPU starts the reset exception handling. Refer to MCU User's Manual for the specific timing.

For details on the SYSRESETREQ bit, see the Arm® Cortex®-M23 Technical Reference Manuals.

6.6 Other Resets

Peripheral functions within the MCU can generate a reset under specific fault conditions. No hardware configuration is required to enable these resets. Refer to the relevant chapters in the *Hardware User's Manual* for details of the conditions that will generate a reset for each peripheral function.

6.7 Determination of Cold/Warm Start

The RA0 MCUs allow the user to determine the cause of the reset processing. The PORSR.PORF register flag indicates whether a power on reset caused the reset processing (cold start) or a reset signal input during operation caused the reset processing (warm start.)

The flag is set to 0 when a power on reset occurs. Otherwise, the flag is not set to 0. The flag is set to 1 when 1 is written to it through software. It is not set to 0 even when 0 is written to it.

6.8 Determining the Reset Source

The RA0 MCUs allow the user to determine the reset signal generation source. Read the RESF register to determine which reset was the source of the reset. Refer to the *Hardware User's Manual* table for *Reset detect flags initialized by each reset source* for more information on which flags are modified depending on the source of the interrupt.

The following sample code shows how to determine if a reset is caused by Power On Reset or Software Reset using CMSIS based register structure in Renesas FSP.

```
/* Power on Reset */
if(1 == R_SYSTEM->PORSR_b.PORF)
{
  /* Do something */
}
…
/* Software Reset */
if(1 == R_SYSTEM->RESF_b.SWRF)
{
  /* Do something */
}
```

7. Memory

The RA0 MCUs support a 4 GB linear address space ranging from 0000 0000h to FFFF FFFFh that can contain program, and data. Program and data memory share the address space. Separate buses are used to access each, increasing performance and allowing same-cycle access of program and data. Contained within the address space are regions for on-chip RAM, peripheral I/O registers, program code flash, and data flash. All reserved areas are off-limits during execution. Undefined behavior may occur if reserved areas are accessed.

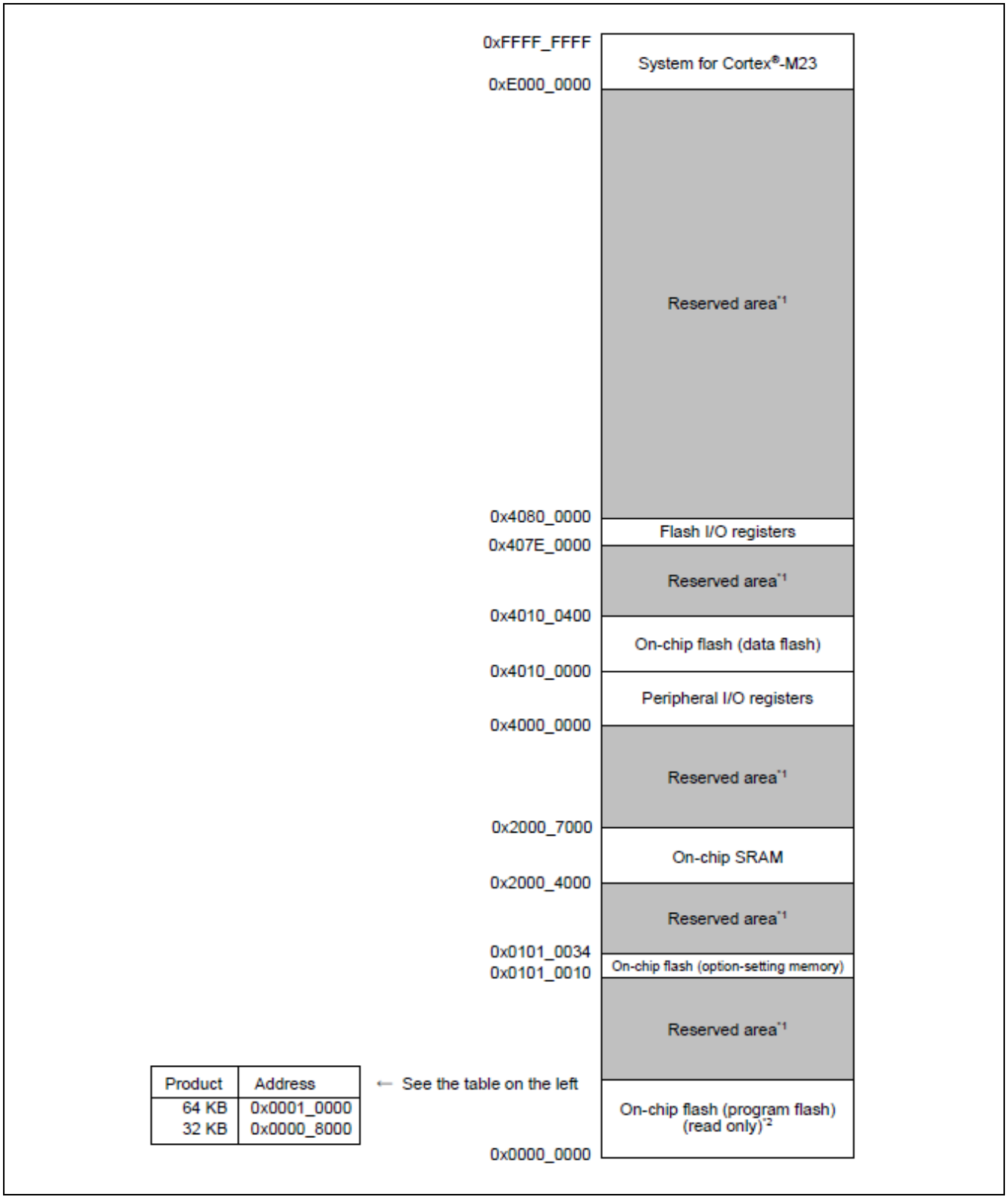


Figure 10. RA0E1 Memory Map

7.1 SRAM

The RA0 MCUs provide on-chip SRAM modules. The following table lists the SRAM specifications for RA0E1 MCU. The SRAM capacity varies by device. Consult the *Hardware User's Manual* for specifics.

Table 27.1 SRAM specifications

Parameter	Description
SRAM capacity	SRAM0: 12 KB
SRAM address	SRAM0: 0x2000_4000 to 0x2000_6FFF
Access*1	0 wait for both reading and writing
Parity	Even parity with 8-bit data and 1-bit parity
Error checking	Even parity error check

Figure 11. RA0E1 SRAM Specification

7.2 Peripheral I/O Registers

Blocks of peripheral I/O registers appear at various locations in the address space depending on the device and the current operating mode. The majority of peripheral I/O registers occupy a region from address 4000 0000h to 400F FFFFh. However, this may vary in location and size by device. Consult the *Hardware User's Manual* for specifics. Details can be found in the appendix, and also in the register descriptions for each peripheral function. This region contains registers that are available at all times in all modes of operation. Flash I/O registers to control access flash memory occupy the region from 407E 0000h to 407F FFFFh.

The Renesas FSP provide C header files in CMSIS data structure that map all of the peripheral I/O registers for a specific device to easily accessible I/O data structures.

7.3 On-Chip Flash Memory

The RA0 MCUs feature two flash memory sections: code flash and data flash, which vary in size and programmable cycle capacity by device. The Flash Control Block (FCBU) controls programming and erasure of flash memory. The Flash Application Command Interface (FACI) controls the FCU in accordance with the specified FACI commands.

The code flash is designed to store user application code and constant data. The data flash is designed to store information that may need frequent updates such as configuration parameters, user settings, or logged data. The units of programming and erasure in the data flash area are smaller than that of the code flash. For example, on RA0E1 devices, the code flash memory uses 32-bit units for programming and 2 KB units for erasure, while the data flash memory uses 8-bit units for programming and 256B units for erasure. The unit sizes vary by device. See the Flash Memory chapter in the *Hardware User's Manual* for details.

Both the data flash and code flash areas can be programmed or erased by application code. This mode is called self-programming. This enables field firmware updates without having to connect an external programming tool. In self-programming, it is necessary to supply a stable HOCO clock to the flash memory to generate the program voltage and erase voltage. Therefore, if the HOCO is stopped where another clock source is selected as the system clock, it is necessary to start the HOCO operation and ensure that the oscillation is in a stable state before executing the self-programming.

Renesas FSP provides HAL layer drivers for both code flash memory and data flash memory.

The following figure shows example specifications of code flash memory and data flash memory.

Table 28.1 Code flash memory and data flash memory specifications

Parameter	Code flash memory	Data flash memory
Memory capacity	• 64-KB/32-KB of user area • Configuration setting area (See section 6, Option-Setting Memory)	1-KB of data area
Read cycle	• A read operation takes 2 cycles	• A read operation takes 6 cycles
Value after erasure	0xFF	0xFF
Programming/erasing method	• Programming and erasure of code and data flash memory through the FCB commands specified in the registers • Programming of flash memory by user program (self-programming)*1.	
Security function	Protection against illicit tampering with or reading of data in flash memory	
Protection	Protection against erroneous overwriting of flash memory	
Background operation (BGO)	Code flash memory can be read during data flash memory programming	
Units of programming and erasure	• 32-bit units for programming in user area • 2-KB units for erasure in user area.	• 8-bit units for programming in data area • 256B units for erasure in data area.
Other functions	Interrupts accepted during self-programming	
	Option-setting memory can be set in the initial MCU settings	
On-board programming	Programming in on-chip debug mode: • SWD interface used • Dedicated hardware not required. Programming by a routine for code and data flash memory programming within the user program: • Allows code and data flash memory programming without resetting the system.	

Note 1. HOCO should be stably oscillated. See section 28.9. Self-programming.

Figure 12. Specifications of Code Flash Memory and Data Flash Memory on RA0E1 MCU

7.3.1 Background Operation

RA0 MCUs support background operations (BGO) for data flash. This means that when a program or erase starts, the user can keep executing and accessing memory from memory areas other than the one being operated on. For example, the CPU can execute application code from code flash while the data flash memory is being erased or programmed.

7.3.2 ID Code Protection

RA0 MCUs have a 128-bit memory in option-setting memory area that is used as an ID code. If this ID code is left blank (0xFF's) then no protection is enabled and access to the MCU is allowed using the on-chip debugger. If the ID code is set, then the user can control access to these modes. The user can choose to always disallow connections or can choose to allow connections when a matching ID code is input. Refer to the OCD/Serial Programmer ID Setting Register (OSIS) and ID Code Protection and sections of *RA0 MCU Hardware User's Manual* for more information.

Renesas FSP configurator provides options to set up ID code protection for RA0 MCUs as shown in Figure 13.

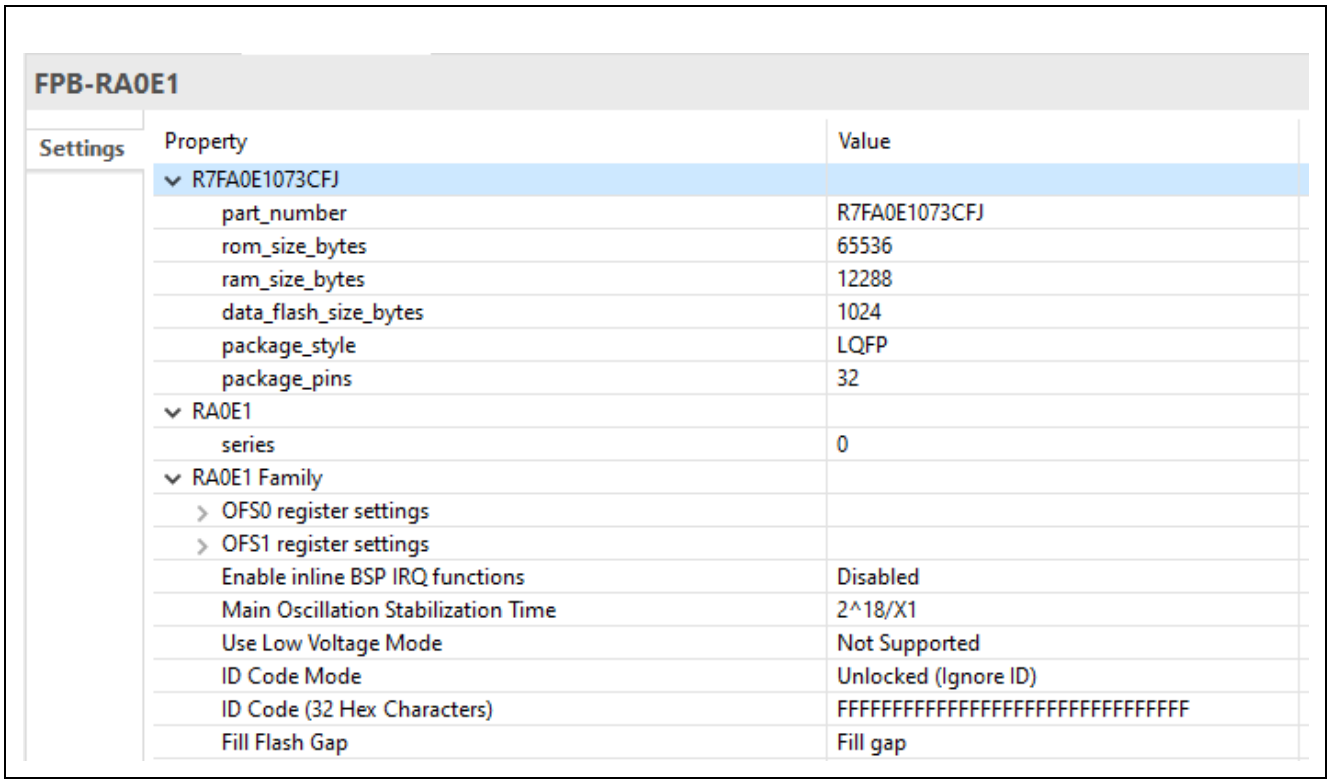


Figure 13. ID Code Setup for RA0E1 Using Renesas FSP Configurator

Note: ID code protection settings must be handled carefully to prevent mistakes that may result in blocking access to the MCU.

7.4 Restriction on Endianness

Memory space must be little-endian to execute code on the Arm® Cortex®-M core.

8. Register Write Protection

The register write protection function protects important registers from being overwritten because of software errors. The registers to be protected are set with the Protect Register (PRCR). Table 8 lists the association between the PRCR bits and the registers to be protected.

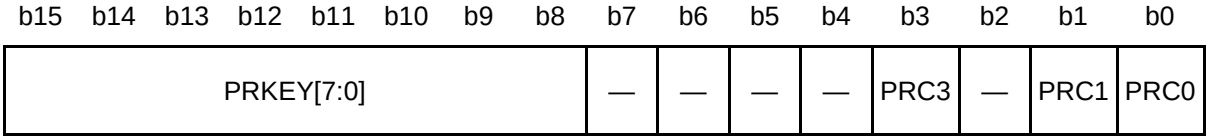


Figure 14. PRCR Register

Table 8. PRCR Protection Bits

PRCR bit	Description
PRC0	• Registers related to the Clock Generation Circuit: CMC, SOMRG, FOCOSCR, FMAINSCR, FSUBSCR, ICLKSCR, MOSCCR, SOSCCR, LOCOCR, HOCOCR, MOCOCR, OSTS, HOCODIV, MOCODIV, MOSCDIV, OSMC, LIOTRM, MIOTRM
PRC1	• Registers related to the low power modes: • SBYCR, PSMCR, SYOCDCR
PRC3	• Registers related to the LVD: LVD1CR, LVD1MKR, LVD1SR
PRKEY[7:0]	These bits control write access to the PRCR register. To modify the PRCR register, write A5h to the eight higher-order bits and the wanted value to the eight lower-order bits as a 16-bit unit.

Note**:** Not all registers may be included on all RA0 devices. Please refer to the *Register Write Protection* section of the *Hardware User's Manual* for more details.

Renesas FSP supplies two APIs (`R_BSP_RegisterProtectEnable` and `R_BSP_RegisterProtectDisable`) to enable and disable Register Write Protection respectively.

9. I/O Port Configuration

The *I/O Ports* section of the *Hardware User's Manual* describes exact pin configurations based on peripheral selection and other register settings. Some general information is listed as follows.

It is important to note that after a reset, each pin will be in the default state for that pin until the configuration is applied. For RA0 devices, all I/O pins operate as input pins immediately after a reset. There may be a small period where some pins may be in an undesirable state. This will be true regardless of what configuration approach is used. The user should consider the impact this may have for each application, including how this may affect other system features.

The IO Port Configuration may be set by either directly writing to registers or using the FSP Pin Configurator.

9.1 Multifunction Pin Selection Design Strategies

Most ports on the RA0 Series of MCUs can have multiple peripheral functions. Tools, such as the Pin Configurator in FSP, are available from Renesas to assist with port selection for each RA0 device. When several peripheral functions are needed, use the following design strategies to help with port function selection.

- Assign peripheral functions with only one port option first. For example, there is only one port option for each debug signal in the debug function. When this function is needed, assign these ports first.
- Assign peripheral functions with limited port options next. For example, the CLKOUT function only has three options for the CLKOUT signal on RA0E1.
- Assign peripheral functions with multiple port options last. One example would be the Serial Array Unit (SAU) which typically has several available port options.
- The Pin Lists section in the RA0 *Hardware User's Manuals* show some peripheral port functions with a suffix (such as "_A") at the end of the function name. For RA0 devices, this type of suffix can be ignored when assigning port functions for most peripheral. Please see the I/O Ports section, Peripheral Select

Settings for Each Product in the RA0 *Hardware User's Manual* for details. Also see section 9.3 in this document.

9.2 Setting Up and Using a Port as GPIO

There are two methods for setting up and using a port as GPIO, either using the Port Control Registers (PODRm, PDRm, PIDRm, POSRm, EORRm, EOSRm), or the PmnPFS registers.

Method 1: Port Control Registers

- Select a pin as an output by writing a 1 to the appropriate bit of the Port Direction Register (PDRm).
- The Port Direction bits (PDRn) are read/write. Setting the value to a 1 selects the pin as an output. Default state for I/O Ports is 0 (input). The port direction registers can be read on the RA0 MCUs.
- The Port Output Data bits (PODRn) in the corresponding Port Output Data Register (PODRm) are read/write. When the PODR is read the state of the output data latch (not the pin level) is read.
- The Port Input bits (PIDRn) in Port State Register (PIDRm) are read only. Read the PIDRn bit in the PIDRm register to read the pin state.

Method 2: Port mn Pin Function Select (PmnPFS) Registers

- The Port Mode Registers are read/write. They are used to specify whether individual pins function as GPIO or as peripheral pins.
- When setting a pin as an output, it is recommended that the desired output value of the port be written to the data latch first, then the direction register should be set to an output. Though not important in all systems, this prevents an unintended output glitch when the port is set up.

In general, using the Port Control Registers to configure a port will provide faster access but will have fewer configuration features available. Using the PmnPFS registers will have more configuration features available but will have slower access.

Renesas FSP provides a Pin Configurator to configure the GPIO pin after reset as shown in Figure 15. Configuring P008 as Output and Low using FSP Configurator. After the GPIO is configured, it can be controlled using HAL layer APIs in FSP.

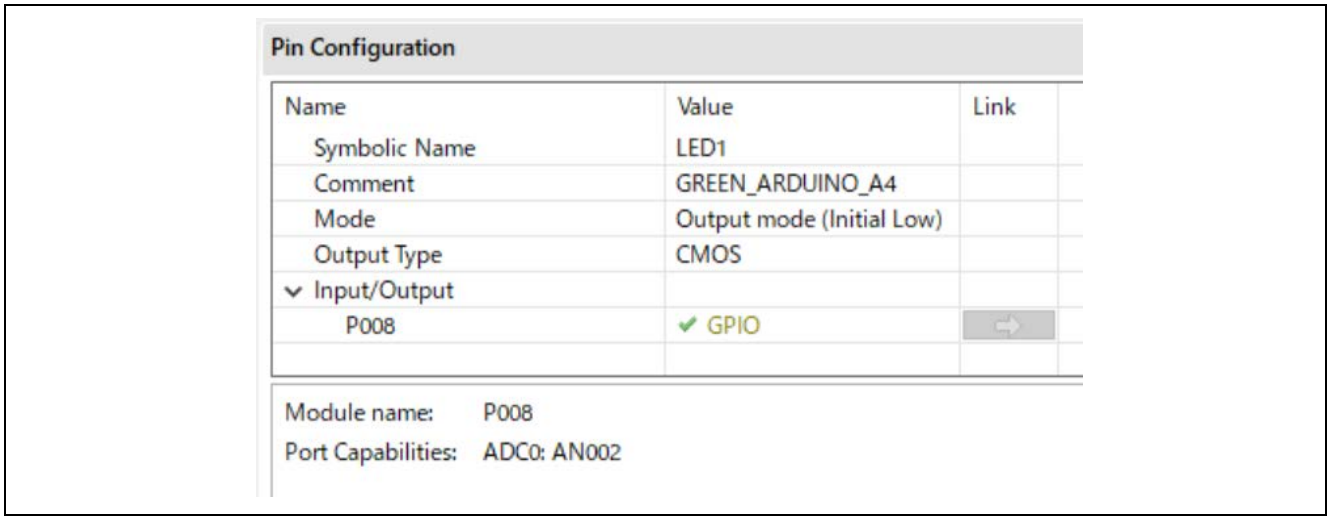


Figure 15. Configuring P008 as Output and Low using FSP Configurator

9.2.1 Internal Pull-Ups

- Most port pins can enable a pull-up resistor. The pull-up is controlled by the Pull-Up bit (PCR) bit in each Port mn Pin Function Select (PmnPFS) register. The PCR bit in each PmnPFS register controls the corresponding pin on the port.
- The pin must first be set as an input with the associated bit in the PmnPFS register. Set the PCR bit to 1 to enable the pull-up and to 0 to disable it.
- Out of reset most PCR registers are cleared to 0, therefore the pull-up resistors are disabled. See the *Hardware User's Manual* for exceptions.
- The pull-up is automatically turned off whenever a pin is designated as an external bus pin, a GPIO output, or a peripheral function output pin.

9.2.2 Open-Drain Output

- Pins configured as outputs normally operate as CMOS outputs.
- Most pins on ports 0 through 9 can be configured as an NMOS open-drain output.
- The N-channel open-drain control (NCODR) bit in each Port mn Pin Function Select (PmnPFS) Register controls which pins operate in open-drain mode. Setting the applicable bit in each register to a 1 makes the output open-drain. Setting the applicable bit in each register to a 0 sets the port to CMOS output.

9.3 Setting Up and Using Port Peripheral Functions

The Port mn Pin Function Select Registers (PmnPFS) are used to configure the characteristics of each port. The PSEL bits select the peripheral function selected for each port.

- Since most pins have multiple functions, the RA0 MCUs have Pin Function Control Registers (PmnPFS) that allow you to change the function assigned to a pin.
- Each pin has its own PmnPFS register.
- Each PmnPFS register allows a pin to be used for peripheral function (PSEL bits), as an IRQ input pin (ISEL bit), or as an analog input pin (PMC bit). To use a pin as an analog function, set the Pin Mode Control bit (PMC) to 1, the N-Channel Open-Drain Control bit (NCODR) to 0, and the Port Direction bit (PDR) to 0 in the Port mn Pin Function Select Register PmnPFS_A.
- Refer to the Peripheral Select Settings for each Product section in the I/O Ports chapter of the Hardware User's Manual.
- All PmnPFS registers are write protected out of reset. In order to write to these registers, first enable writing using the Write-Protect Register (PWPR).
- Care should be taken when setting PmnPFS registers so that a single function is not assigned to multiple pins. The user should not do this but the MCU will allow it. If this occurs the function on the pins will be undefined.

Figure 16 shows an example of enabling SAU_SPI1 pins using FSP Pin configuration.

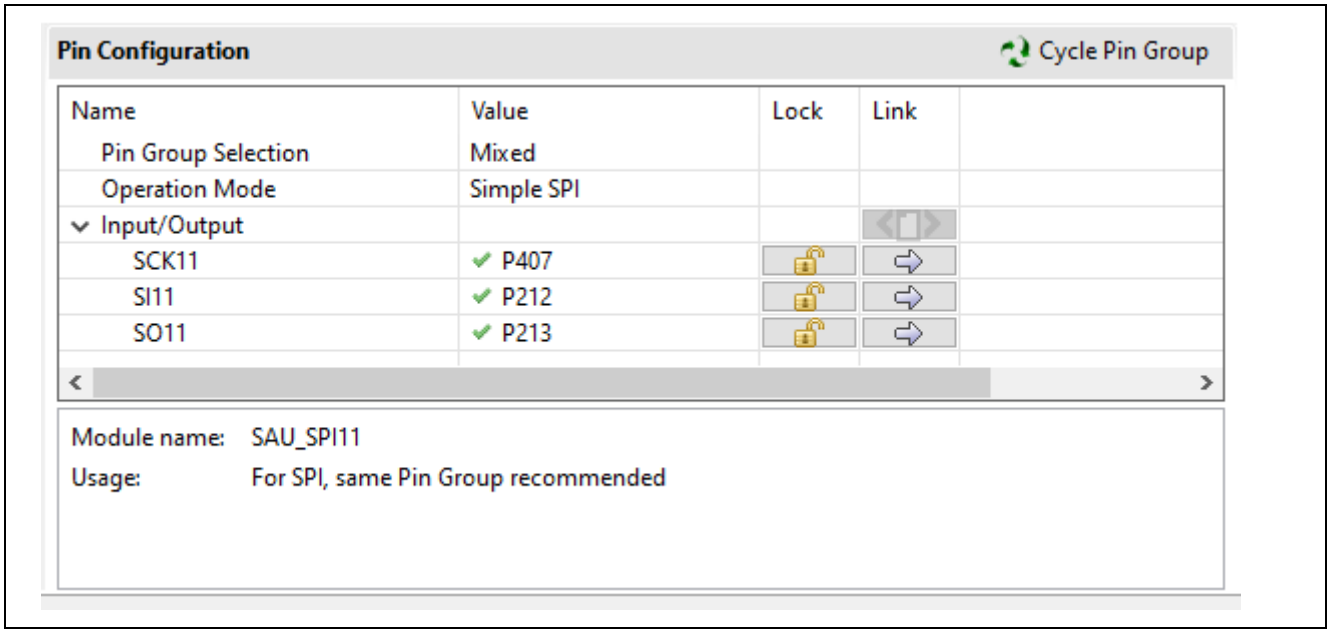


Figure 16. Enabling SPI0 pins Using Pin Configurator in Renesas FSP

9.4 Setting Up and Using IRQ Pins

- Certain port pins can be used as hardware interrupt lines (IRQ). See the Peripheral Select Settings for each Product section in the I/O Ports chapter of the Hardware User's Manual for information on which pins are available for your MCU.
- To set a port pin to be used as an IRQ pin, the Interrupt Input Function Select bit (ISEL) in the pin's PFS register must be set to 1.
- Pins can be used for both IRQ and peripheral functions simultaneously. To enable this, set both the ISEL and PSEL bits in the pin's PFS register.
- IRQ functions of the same number must only be enabled on one pin.
- IRQ pins can trigger interrupts on detection of:
  - Falling edge
  - Rising edge
  - Rising and falling edges

  The IRQ Control Registers (IRQCRi) controls which trigger is selected.

Figure 17 and Figure 18 show examples of enabling and configuring IRQ pins using Renesas FSP.

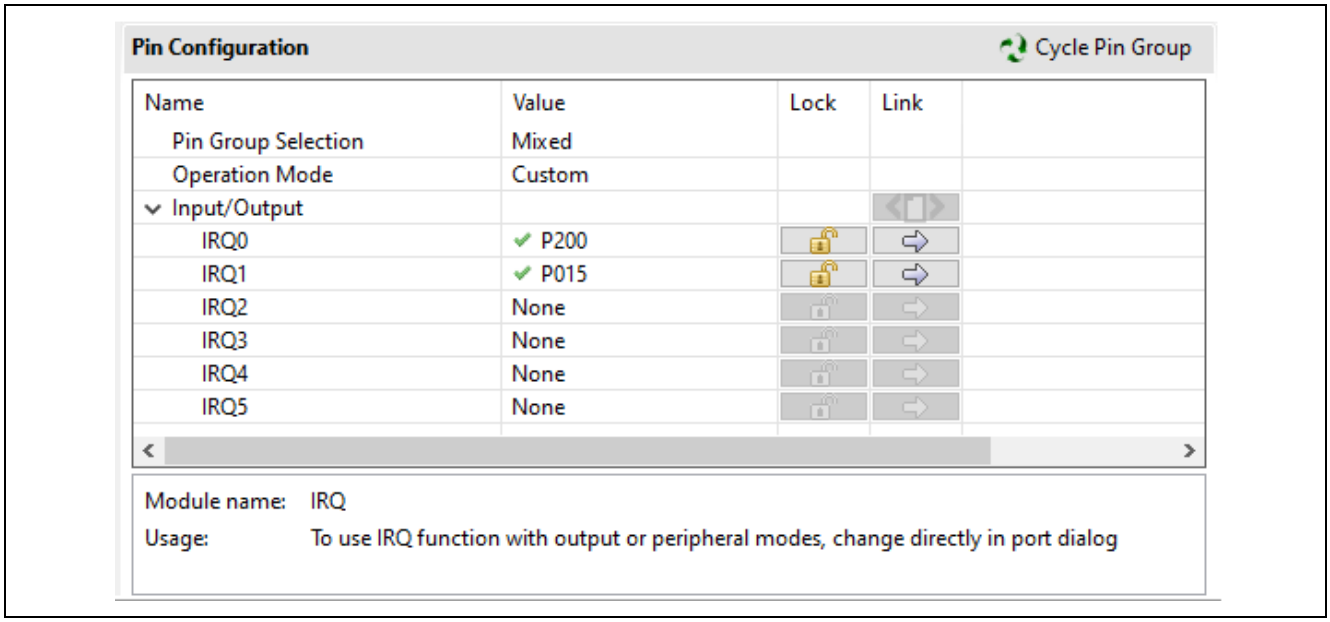


Figure 17. Enable P200 and P015 as IRQ0 and IRQ1 Input Using Pin Configurator in Renesas FSP

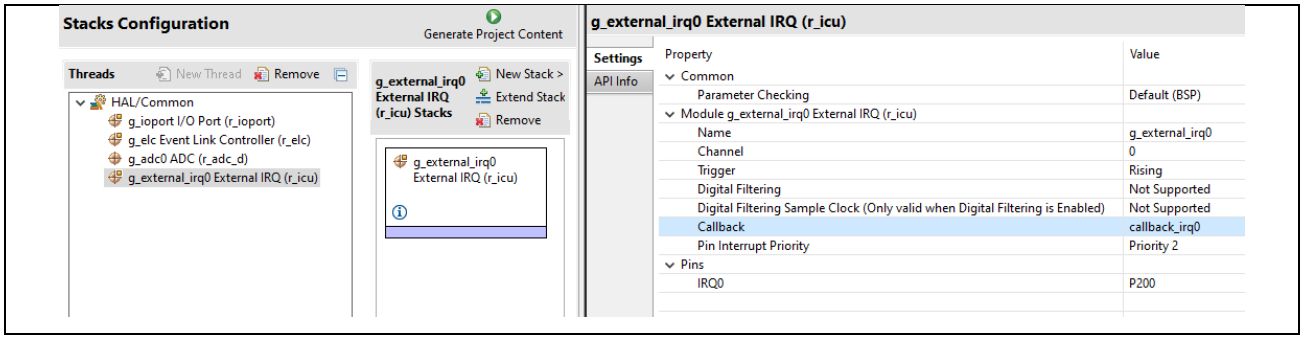


Figure 18. Configure IRQ0 Using Renesas FSP Configurator

9.5 Unused Pins

Note: Some pins require specific termination: See the *Handling of Unused Pins* section of the *Hardware User's Manual* for specific recommendations.

Unused pins that are left floating can consume extra power and leave the system more susceptible to noise problems. Terminate unused pins with one of the methods detailed here:

1. The first option is to set the pin to an input (the default state after reset) and connect the pin to VCC or VSS using a resistor. There is no difference to the MCU between one connection or another; however, there may be an advantage from a system noise perspective. VSS is probably the most typical choice. Avoid connecting a pin directly to VCC or VSS since an accidental write to the port's direction register that sets the pin to an output could create a shorted output.
2. A second method is to set the pin to an output. The pin level may be set high or low. However, setting the pin as an output and making the output low connects the pin internally to the ground plane. This may help with overall system noise concerns. A disadvantage of setting unused pins to outputs is that the configuration of the port must be done via software control. While the MCU is held in reset and until the direction register is set for output, the pin will be a floating input and may draw extra current. If the extra current can be tolerated during this time, this method eliminates the external resistors required in the first method.
3. A variation on leaving the pins as inputs and terminating them with external resistors uses the internal pull-ups available on many ports of the MCU. This has the same limitation as setting the pins to outputs

(requires the program to set up the port) but it does limit the effect of accidental pin shorts to ground, adjacent pins or VCC since the device will not be driving the pin.

9.6 Nonexistent Pins

Each RA0 MCU group is available in multiple package sizes, with different total pin counts. For any package smaller than the largest package for that MCU group (typically 24 pins, 20 pins, or 16 pins), set the corresponding bits of nonexistent ports in the PDR register to 0 and in the PODR register to 0. The user can see which ports are available on each MCU package by reviewing the Specifications of I/O Ports table in the I/O Ports section of the Hardware User's Manual. Note that no additional handling of nonexistent pins is required.

9.7 Electrical Characteristics

Normal GPIO ports typically require CMOS level inputs (High ≥ 0.8 * VCC, Low ≤ 0.2 * VCC). Some GPIO ports have Schmitt Trigger inputs, which have slightly different input requirements. See the *Hardware User's Manual* section *Electrical Characteristics* for more information.

10. Module Stop Function

To maximize power efficiency, the RA0 series of MCUs allow on-chip peripherals to be stopped individually by writing to the Module Stop Control Registers (MSTPCRi, i=A, B, C, D). Once a module stops, access to the module registers is not possible.

After a reset, most of the modules are placed in module-stop state, except for DTC. See *Hardware User's Manual* for details.

Before accessing any of the registers for a peripheral, the peripheral must be enabled by taking it out of stop mode by writing a 0 to the corresponding bit in the MSTPCRi register.

Peripherals may be stopped by writing a 1 to the proper bit in the MSTPCRi register.

HAL drivers in Renesas FSP handle module start/stop function automatically.

11. Interrupt Control Unit

The Interrupt Controller Unit (ICU) controls which event signals are linked to the Nested Vector Interrupt Controller (NVIC) and Data Transfer Control (DTC) modules. The ICU also controls non-maskable interrupts. Figure 19 shows an example of the ICU specifications, and Figure 20 shows an example of the ability to raise the IRQi event from the I/O pins. Refer to the *Hardware User's Manual* for details for each RA0 MCU Group.

Item		Description
Maskable interrupts	Peripheral function interrupts	• Interrupts from peripheral modules Number of sources: 33
	External pin interrupts	• Interrupt detection on falling edge, rising edge, rising and falling edges. One of these detection methods can be set for each source • 6 sources, with interrupts from IRQi (i = 0 to 5) pins.
	Interrupt requests to CPU (NVIC)	• 39 interrupt requests are output to NVIC.
	DTC control	• The DTC can be activated using interrupt sources*1 • The method for selecting an interrupt source is the same as that of the interrupt request to NVIC.
Non-maskable interrupts*2	NMI pin interrupt	• Interrupt from the NMI pin • Interrupt detection on falling edge or rising edge
	IWDT underflow/refresh error*3	Interrupt on an underflow of the down-counter or occurrence of a refresh error
	Low voltage detection 1*3	Voltage monitor 1 interrupt of the voltage monitor 1 circuit (LVD_LVD1)
	RPEST	Interrupt on SRAM parity error
Low power modes		• Sleep mode: return is initiated by non-maskable interrupts or any other interrupt source • Software Standby mode: return is initiated by non-maskable interrupts. Interrupt can be selected in the SBYEDCRn register. • Snooze mode: return is initiated by non-maskable interrupts. Interrupt can be selected in the SBYEDCRn register. See section 11.2.14. SBYEDCR0 : Software Standby/Snooze End Control Register 0 and section 11.2.15. SBYEDCR1 : Software Standby/Snooze End Control Register 1.

Note 1. For the DTC activation sources, see Table 11.5.
Note 2. Non-maskable interrupts can be enabled only once after a reset release.
Note 3. These non-maskable interrupts can also be used as maskable interrupts. When used as maskable interrupts, do not change the value of the NMIER register from the reset state. To enable voltage monitor 1 interrupts, set the LVD1CR1.IRQSEL bits to 1.

Figure 19. RA0E1 ICU Specifications

Table 11.2 ICU I/O pins

Pin name	I/O	Description
NMI	Input	Non-maskable interrupt request pin
IRQi (i = 0 to 5)	Input	External interrupt request pins

Figure 20. RA0E1 ICU I/O Pins

Figure 21 is an example of using a Renesas FSP configurator to enable and configure an interrupt using Renesas FSP. The ICU and interrupts are configured as part of the HAL driver configuration through FSP.

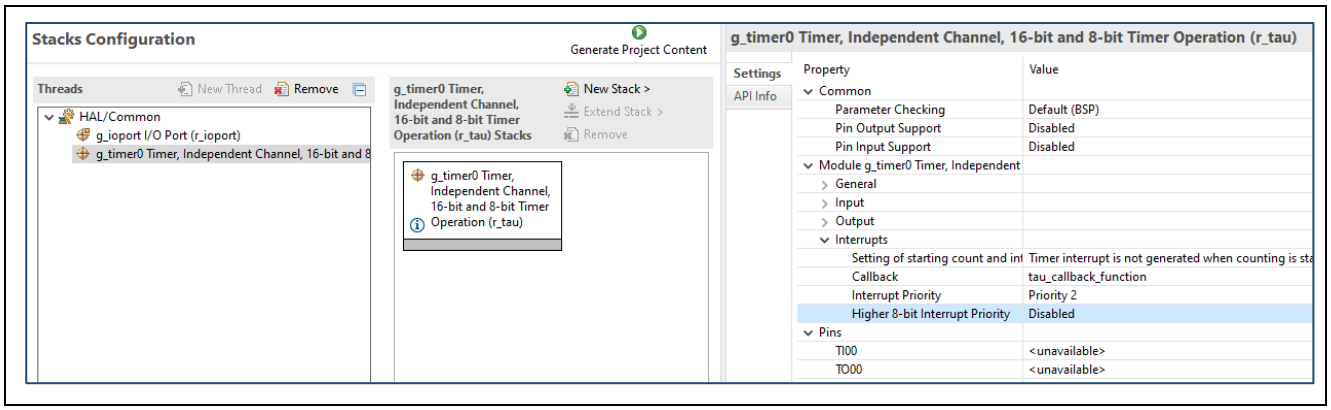


Figure 21. Enable TAU Interrupt and Set User Callback Functions Invoked by Interrupt Service Routine

12. Low Power Consumption

The RA0 devices have several functions for reducing power consumption. These include setting clock dividers, stopping modules, selecting power control mode in Normal mode, and transitions to low power modes. Refer to the Low Power Modes chapter in the *Hardware User's Manual* for more details.

RA0 MCUs support three different types of Low Power Modes (LPM.) These are:

- Sleep mode – The CPU is stopped, but all other peripherals and clock sources continue to operate (fastest wake-up)
- Snooze mode – The CPU is stopped, but some peripherals may turn on required clock sources
- Software Standby mode – The CPU and most peripherals and clock sources are stopped (most power savings)

The following table is an overview of the functions available for reducing power consumption.

Table 9. Specifications of Low Power Mode Functions

Item	Specification
Reducing power consumption by modifying clock signals	The frequency division ratio can be selected for HOCO, MOCO, and MOSC*1
Module stop	Functions can be stopped independently for each peripheral module.
Low power modes	• Sleep mode • Snooze mode • Software Standby mode
Power control modes	Power consumption can be reduced in Normal, Sleep, and Snooze mode by selecting an appropriate operating power control mode according to the operating frequency and voltage. Four operating power control modes are available: ● High-speed mode ● Middle-speed mode ● Low-speed mode ● Subosc-speed mode

Notes: 1. For details, see the chapter *Clock Generation Circuit* in the *Hardware User's Manual*.

Table 10 lists the conditions to transition to low power modes, the states of the CPU and the peripheral modules, and the method for cancelling each mode.

Table 10. Low Power Consumption Modes

State of operation*[1]	Sleep Mode	Software Standby Mode	Snooze Mode
Transition condition	WFI instruction while SBYCR.SSBY=0	WFI instruction while SBYCR.SSBY=1	Snooze request trigger in Software Standby mode. SBYCR.SSBY=1
Canceling method	All interrupts. Any reset available in the mode.	Interrupts defined for this mode. Any reset available in the mode.	Interrupts defined for this mode. Any reset available in the mode.
State after cancellation by an interrupt	Program execution state (interrupt processing)	Program execution state (interrupt processing)	Program execution state (interrupt processing)
State after cancellation by a reset	Reset state	Reset state	Reset state

Notes: 1. Refer to the table Operating Conditions of Each Low Power Mode in the *Hardware User's Manual* for additional details.

RA0 devices include register settings that allow the MCU to operate with lower power consumption in Normal mode Sleep mode and Snooze mode. These modes utilize the FLMODE register to reduce power consumption in Normal mode, Sleep mode, and Snooze mode.

The following is a summary of the Operating Power Consumption Control modes and the maximum permissible clocking and voltage levels under each mode.

Table 11. Available Oscillators in Each Operating Power Consumption Control Mode

Mode	Oscillator				
	High-speed on-chip oscillator	**Middle-speed on-chip oscillator**	**Low-speed on-chip oscillator**	**Main clock oscillator**	**Sub-clock oscillator**
High-speed	Available	Available	Available	Available	Available
Middle-speed	Available	Available	Available	Available	Available
Low-speed	Available	Available	Available	Available	Available
Subosc-speed	N/A	N/A	Available	N/A	Available

While it may be possible to set the value in the FLMODE register to any of the low power operating modes, clocking and voltage levels must also be set to meet the requirements of the desired mode. Otherwise, the settings in the FLMODE register will not have any effect in lowering power consumption.

To achieve the lowest power numbers, use the maximum possible dividers in the clock generation circuits.

The following diagram is an example of the relative power consumption for each operating power control mode, set by the MODE bits in the FLMODE and the CKST bit in the ICLKSCR register.

Operating power control mode	MODE[1:0] bits	ICLKSCR.CKST bit	Power consumption
High-speed mode	11b	0	High
Middle-speed mode	10b	0	↓
Low-speed mode	01b	0	↓
Subosc-speed mode	xxb	1	Low

Figure 22. Operating Power Control Mode

Low power modes are canceled by various interrupt sources such as RES pin reset, power-on reset, voltage monitor reset, and peripheral interrupts. Refer to the Low Power Modes section in the *Hardware User's Manual* for a list of interrupt sources for different LPMs.

Only Snooze mode is triggered by a Snooze request to enter snooze mode from Software Standby mode. The transitions to other LPMs are done by executing a WFI instruction with appropriate settings in the Standby Control register (SBYCR).

Renesas FSP provides a low power mode (LPM) driver and driver configurator to set up low power mode, wake source/cancel source, and so forth.

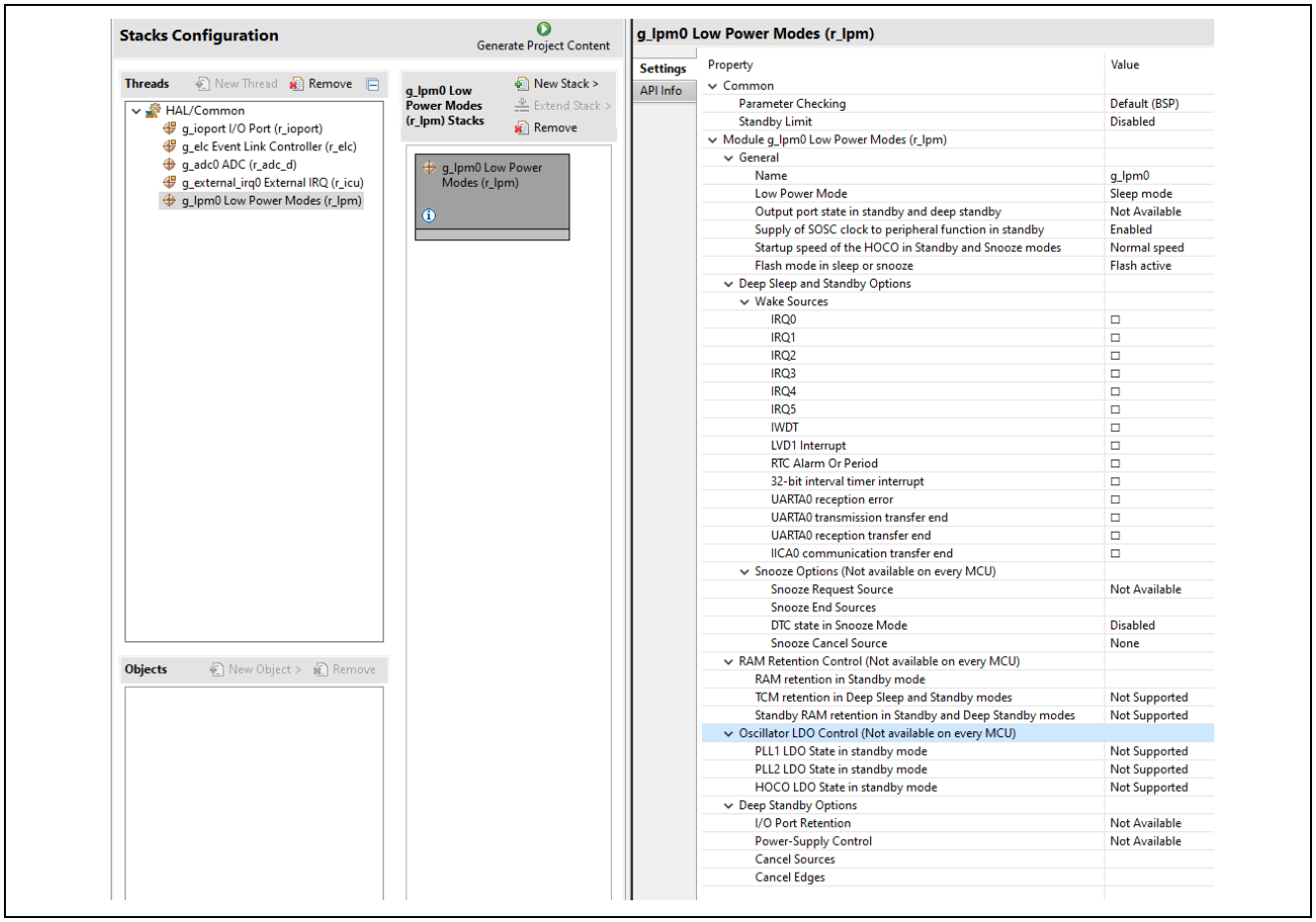


Figure 23. Set Up Low Power Mode Using Renesas FSP Configurator

After a specific LPM mode is set up by the FSP Configurator, the LPM driver's API can be used to initialize LPM driver and place MCU in configured LPM mode, as shown in the following example:

```
/* Open LPM driver and initialize LPM mode */
err = R_LPM_Open(&g_lpm_sw_standby_ctrl, &g_lpm_sw_standby_cfg);
/* Handle error */
if (FSP_SUCCESS != err)
 {
   return (err);
 }
/* Transition to configured LPM mode: Deep Software Standby Mode */
err = R_LPM_LowPowerModeEnter(&g_lpm_sw_standby_ctrl);
/* Handle error */
if (FSP_SUCCESS != err)
 {
   return (err);
 }
```

13. Buses

The buses in RA0 MCUs consist of a main bus and a slave interface. Figure 24 lists the main bus and the slave interface. Figure 25 shows the bus configuration.

Note: Memory space must be little-endian in order to execute Arm® Cortex® code.

Table 12.1 Bus specifications

Bus type		Description
Main bus	System bus (CPU)	• Connected to CPU • Connected to on-chip memory and internal peripheral bus
	DMA bus	• Connected to DTC • Connected to on-chip memory and internal peripheral bus
Slave interface	Memory bus 1	Connected to code flash memory
	Memory bus 4	Connected to SRAM0
	Internal peripheral bus 1	Connected to system control related to peripheral modules
	Internal peripheral bus 3	• Connected to peripheral modules (ELC, IWDT, MSTP and CRC) • Connected to peripheral modules (I/O Ports, ADC12, SAU0, SAU1, TAU, RTC, IICA, UARTA, TML32, and PCLBUZ)
	Internal peripheral bus 7	Connected to TRNG
	Internal peripheral bus 9	Connected to code flash memory (in P/E (Programming/Erasure)), data flash memory

Figure 24. RA0E1 Bus Specifications

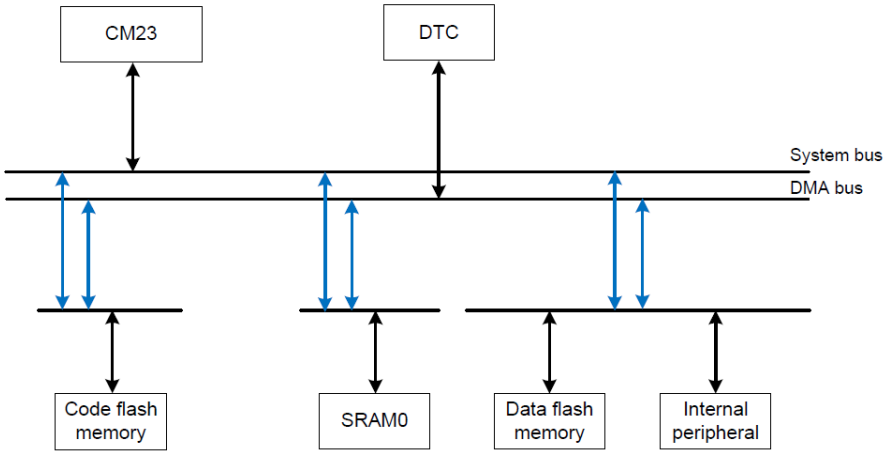


Figure 25. RA0 MCU Simplified Bus Configuration

13.1 Bus Error Monitoring

The monitoring system monitors each individual area. Whenever it detects an error, it returns the error to the requesting master IP using the AHB-Lite error response protocol.

13.1.1 Bus Error Types

The following types of errors can occur on each bus:

- Illegal address access

13.1.2 Operation When a Bus Error Occurs

When a bus error occurs, operation is not guaranteed, and the error is returned to the requesting master IP. The bus error information that occurred in each master is stored in the BUSnERRADD and BUSnERRSTAT

registers. These registers must be cleared by reset only. For more information, see the Bus Error Address Register (BUSnERRADD) and Bus Error Status Register (BUSnERRSTAT) sections in the Hardware User's Manual.

Note: The DTC does not receive bus errors, so its operation is not affected by bus errors.

14. General Layout Practices

14.1 Digital Domain vs. Analog Domain

Renesas RA0 Microcontroller devices have three primary types of pin functions: Power, Digital, and Analog.

Generally, power pins are dedicated for voltage and reference input and do not have multiple functions. Power pins are typically dedicated to specific portions, or domains, within the MCU. For example, the main supply voltage for the MCU will provide power to the digital core, many of the digital peripheral functions and many of the digital I/O pins. The digital domain can be defined as the digital circuitry, digital I/O pins, and the related power pins. Power pins which are designated for analog functions (such as VREFH0 and the associated VREFL0 reference voltage pins) supply specific analog circuitry within the MCU, which is separate from the digital domain circuitry. The analog domain can be defined as the analog circuitry, analog I/O pins, and the related power pins.

Digital signals are typically repetitive, switched patterns that are associated with periodic clocks. The transitions on digital signals tend to be relatively sharp edges, with stable levels of high or low between the transitions. Each signal must be stable at an acceptable voltage level, referred to as a logic state, within a specified timeframe. The state of the signal is typically sampled at predetermined clock intervals, using the edge transition of a clock to evaluate the associated data signals. Small variations in the voltage level of digital signals are typically acceptable, as long as the level remains within a specified range. However, large external influences on digital signals can have an acute influence on a digital signal, which can result in an incorrect logic state at the moment when the data is sampled.

Analog signals are usually quite different. Analog signals may be periodic, but the evaluation of an analog signal is typically a measurement of voltage over a range instead of logic state. The voltage level of an analog signal is sampled based on a specific trigger event, and the resulting measurement is processed using the analog circuitry in the MCU. The accuracy of an analog measurement is directly related to the accuracy of the sampled voltage level. Any unwanted external influence which may change the voltage level of an analog input signal, even slightly, can influence the accuracy of the measurement.

Due to the highly multiplexed nature of the I/O pins on Renesas RA0 MCU devices, many I/O pins can be used for either analog or digital functions. This can result in situations where digital and analog functions may overlap and result in data errors.

To minimize potential problems between digital and analog signal domains, consider the following guidelines:

- When assigning I/O pin functions, select pin functions such that analog pins and digital pins are physically separated as much as possible.
- Each analog signal should be separated from all other signals as much as possible.
- PCB routing should isolate each analog signal as much as possible. Avoid routing any other signals, either analog or digital, in the same area.
- Ensure that analog reference voltages include appropriate AC filters. This may be in the form of recommended capacitors located near the MCU voltage pin, or appropriate inductive filters. The goal is to provide reference voltage with little or no voltage ripple.
- When using dedicated power layers in a PCB design, avoid routing digital signals in the areas of analog voltages, and avoid routing analog signals in the areas of digital voltages.

For highly sensitive applications, it is highly recommended to evaluate the specific design using simulation tools to understand the effect that circuit design has on the performance. For example, this may include applications such as precision sensor designs, or very high-speed digital bus interfaces. Refer to the Electrical Characteristics chapter in the *Hardware User's Manual* for the specific requirements for each peripheral function.

14.2 High Speed Signal Design Considerations

As clock speeds for digital signals increase, the influence of external stimuli on those signals can become more significant. Some peripheral functions can be classified as High Speed digital signals. Additional design considerations should be made for high-speed digital signals.

Crosstalk is a condition where transitions on one signal have an inductive influence on another nearby signal. When this crosstalk effect is strong enough, the first signal may cause errors on the second signal. To reduce the effects of crosstalk, use the following general PCB routing guidelines.

- Provide sufficient space between routed signals on the same routing layer. Generally, keep a minimum of one trace width space between signals of the same digital group, and a minimum of 3-5 trace widths space between signals of different digital groups.
- Provide extra space between clock signals and data signals on the same routing layer. Generally, keep a minimum of 3-5 trace widths space between clocks and any other digital signals.
- Avoid parallel routing of digital signals on any adjacent routing layers. If signals must be routed on adjacent signals layers, try to use only orthogonal crossings wherever possible.

If possible, separate PCB signal layers using power or ground layers between signal layers. The solid copper of the power or ground layer can act as a shield for the digital signals.

Each standardized interface will have specific requirements. To ensure that the PCB is designed to avoid signal crosstalk problems, we strongly suggest referring to the relevant standards for each interface in the design.

14.3 Signal Group Selections

Some pin names have an added _A, _B, _C, _D, _E, or _F suffix to indicate signal groups. For RA0 devices, the suffix can be ignored when assigning functionality, except for SAU and IICA. For SAU and IICA, only signals, except for SCL11 and SCK11, bearing the same suffix can be selected. For these two signal groups, the simultaneous use of the same signal with different suffixes is prohibited. For all other signal groups, it is safe to select the most convenient pin assignment for each function signal.

Refer to the sections Peripheral Select Settings for each Product and Notes on the PmnPFS Register Setting in the I/O Ports chapter of the *Hardware User's Manual*.

15. References

The following documents were used in creating this Quick Design Guide. Visit Renesas website for the latest version of each of these documents.

Reference	Document Number	Description
1	R01AN6277	Renesas RA Family Design Guide for Sub-Clock Circuits
2	R01UH1040	Renesas RA0E1 Group, User’s Manual: Hardware

Website and Support

Visit the following URLs to learn about key elements of the RA family, download components and related documentation, and get support.

RA Product Information	www.renesas.com/ra
RA Product Support Forum	www.renesas.com/ra/forum
RA Flexible Software Package	www.renesas.com/FSP
Renesas Support	www.renesas.com/support

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Apr.09.24	—	Initial release

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/.