**Introduction**

This document answers common questions and points out subtleties of the RA2 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**

RA2 MCU Series

**Contents**

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Power Supplies</td>
<td>4</td>
</tr>
<tr>
<td>1.1</td>
<td>References</td>
<td>5</td>
</tr>
<tr>
<td>2.</td>
<td>Emulator Support</td>
<td>6</td>
</tr>
<tr>
<td>2.1</td>
<td>SWD Interface</td>
<td>7</td>
</tr>
<tr>
<td>2.2</td>
<td>Serial Programming Interface using SCI</td>
<td>8</td>
</tr>
<tr>
<td>2.3</td>
<td>Combination Debug Interface</td>
<td>9</td>
</tr>
<tr>
<td>3.</td>
<td>MCU Operating Modes</td>
<td>9</td>
</tr>
<tr>
<td>4.</td>
<td>Option Setting Memory</td>
<td>10</td>
</tr>
<tr>
<td>4.1</td>
<td>Option Setting Memory Registers</td>
<td>11</td>
</tr>
<tr>
<td>5.</td>
<td>Clock Circuits</td>
<td>12</td>
</tr>
<tr>
<td>5.1</td>
<td>Reset Conditions</td>
<td>13</td>
</tr>
<tr>
<td>5.2</td>
<td>Clock Frequency Requirements</td>
<td>13</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Requirements for USB Communications</td>
<td>14</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Requirements for Programming and Erasing ROM or Data Flash</td>
<td>14</td>
</tr>
<tr>
<td>5.3</td>
<td>Lowering Clock Generation Circuit (CGC) Power Consumption</td>
<td>14</td>
</tr>
<tr>
<td>5.4</td>
<td>Writing the System Clock Control Registers</td>
<td>14</td>
</tr>
<tr>
<td>5.5</td>
<td>Clock Setup Example</td>
<td>15</td>
</tr>
<tr>
<td>5.6</td>
<td>HOCO Accuracy</td>
<td>15</td>
</tr>
<tr>
<td>5.7</td>
<td>Flash Interface Clock</td>
<td>16</td>
</tr>
<tr>
<td>5.8</td>
<td>Board Design</td>
<td>16</td>
</tr>
<tr>
<td>5.9</td>
<td>External Crystal Resonator selection</td>
<td>16</td>
</tr>
<tr>
<td>5.10</td>
<td>External Clock Input</td>
<td>17</td>
</tr>
<tr>
<td>6.</td>
<td>Reset Requirements and the Reset Circuit</td>
<td>17</td>
</tr>
<tr>
<td>6.1</td>
<td>Pin Reset</td>
<td>18</td>
</tr>
<tr>
<td>6.2</td>
<td>Power-On Reset</td>
<td>18</td>
</tr>
</tbody>
</table>
6.3 Independent Watchdog Timer Reset .............................................................. 18
6.4 Watchdog Timer Reset .................................................................................. 18
6.5 Voltage-Monitoring Resets ........................................................................... 19
6.6 Software Reset .............................................................................................. 19
6.7 Other Resets .................................................................................................. 19
6.8 Determination of Cold/Warm Start ............................................................... 19
6.9 Determining the Reset Source ...................................................................... 19

7. Memory ............................................................................................................. 20
7.1 SRAM .............................................................................................................. 20
7.2 Peripheral I/O Registers ................................................................................ 21
7.3 On-Chip Flash Memory .................................................................................. 21
7.3.1 Background Operation .............................................................................. 22
7.3.2 ID Code Protection .................................................................................... 22
7.3.3 Memory Protection Unit .......................................................................... 23
7.4 Restriction on Endianness ........................................................................... 24

8. Register Write Protection .................................................................................. 24

9. I/O Port Configuration ...................................................................................... 25
9.1 Multifunction Pin Selection Design Strategies ............................................. 25
9.2 Setting Up and Using a Port as GPIO ............................................................ 25
9.2.1 Internal Pull-Ups ...................................................................................... 27
9.2.2 Open-Drain Output .................................................................................. 27
9.2.3 Drive Capacity ........................................................................................ 27
9.3 Setting Up and Using Port Peripheral Functions .......................................... 27
9.4 Setting Up and Using IRQ Pins ................................................................... 28
9.5 Unused Pins ................................................................................................... 30
9.6 Nonexistent Pins .......................................................................................... 31
9.7 Electrical Characteristics ............................................................................. 31

10. Module Stop Function .................................................................................... 31

11. Interrupt Control Unit .................................................................................... 31

12. Low Power Consumption .............................................................................. 33

13. Buses .............................................................................................................. 36
13.1 Bus Error Monitoring .................................................................................. 37
13.1.1 Bus Error Types ...................................................................................... 37
13.1.2 Operation When a Bus Error Occurs ...................................................... 37

14. 24-Bit Sigma-Delta A/D Converter (SDADC24) .......................................... 37

15. Operational Amplifier (OPAMP) with Configurable Switches ..................... 39
16. General Layout Practices ................................................................. 40
16.1 Digital Domain vs. Analog Domain .......................................... 40
16.2 High Speed Signal Design Considerations ............................. 41
16.3 Signal Group Selections .............................................................. 41

17. References .................................................................................... 42

Revision History .............................................................................. 44
1. Power Supplies

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

**Table 1. Digital Power Supplies**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VCC</td>
<td>Power supply</td>
<td>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.</td>
</tr>
<tr>
<td>VSS</td>
<td>Ground</td>
<td>Ground</td>
</tr>
<tr>
<td>VCL</td>
<td>Power supply</td>
<td>Connect this pin to VSS via a capacitor close to the VCL pin. The value depends upon the specific MCU group.</td>
</tr>
<tr>
<td>VCC_USB(^1)</td>
<td>USB FS power supply</td>
<td>USB Full-Speed power supply pin. Connect this pin to VCC. Connect this pin to VSS_USB via a 0.1-µF capacitor placed close to the VCC_USB pin.</td>
</tr>
<tr>
<td>VSS_USB</td>
<td>USB FS ground</td>
<td>USB Full-Speed ground pin. Connect this pin to VSS.</td>
</tr>
<tr>
<td>VCC_DCDC(^2)</td>
<td>Switching regulator power supply</td>
<td>In DCDC mode, connect this pin to VCC. Connect this pin to VSS through a 0.1-µF capacitor placed close to the VCC_DCDC pin. In LDO mode, leave this pin open.</td>
</tr>
<tr>
<td>VSS_DCDC(^2)</td>
<td>Switching regulator ground pin</td>
<td>In DCDC mode, connect this pin to VSS. In LDO mode, leave this pin open.</td>
</tr>
<tr>
<td>VLO</td>
<td>Switching regulator pin</td>
<td>In DCDC mode, connect this pin to an external inductor then connect it to VCL pin. Place the inductor and capacitor close to the pin.</td>
</tr>
</tbody>
</table>

Notes:
1: Only used on RA2A1. VCC_USB can be either input or output. As input, it supplies the USB transceiver. As output, it is the voltage out from the USB LDO Regulator, and needs an external capacitor. When the USB LDO Regulator is not used, connect to VCC. When the regulator is used, connect to VSS through a 1.0-µF capacitor.
2: Only used on RA2L1. Please see the chapter Internal Voltage Regulator in the RA2L1 Hardware User’s Manual for implementation guidelines.
Table 2. Analog Power Supplies

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVCC0³</td>
<td>Analog power supply</td>
<td>Analog voltage supply pin for the respective modules. Connect this pin to the same voltage as the VCC pin.</td>
</tr>
<tr>
<td>AVSS0³</td>
<td>Analog ground</td>
<td>Analog ground for the respective modules. Connect this pin to the same voltage as the VSS pin.</td>
</tr>
<tr>
<td>VREFH0¹</td>
<td>12-bit ADC high reference voltage</td>
<td>Reference voltage input pin for the 12-bit A/D. Connect this pin to AVCC0 if the 12-bit A/D converter is not used.</td>
</tr>
<tr>
<td>VREFL0¹</td>
<td>12-bit ADC low reference voltage</td>
<td>Analog reference ground pin for the 12-bit A/D converter. Connect this pin to AVSS0 if the 12-bit A/D converter is not used.</td>
</tr>
<tr>
<td>VREFH²</td>
<td>12-bit DAC analog supply</td>
<td>Reference voltage input pin for the D/A converter. Connect this pin to AVCC0 if the D/A converter is not used.</td>
</tr>
<tr>
<td>VREFL²</td>
<td>12-bit DAC analog ground</td>
<td>Reference ground pin for the D/A converter. Connect this pin to AVSS0 if the D/A converter is not used.</td>
</tr>
<tr>
<td>AVCC1²</td>
<td>24-bit SDADC analog supply voltage</td>
<td>Analog voltage supply pin for the 24-bit SDADC module.</td>
</tr>
<tr>
<td>AVSS1²</td>
<td>24-bit SDADC analog ground</td>
<td>Analog ground pin for the 24-bit SDADC module.</td>
</tr>
<tr>
<td>VREFI²</td>
<td>24-bit SDADC external reference supply</td>
<td>External voltage reference for the 24-bit SDADC, multiplexed with the SBIAS output. As an input, this pin has a voltage range of 0.8 V to 2.4 V, set in 0.2-V increments. Connect this pin to AVSS1 through a 0.22-µF capacitor.</td>
</tr>
</tbody>
</table>

Notes:
1: For RA2A1, this applies to the 16-bit ADC.
2: Present on RA2A1 only.
3: Not present on RA2E2
4: For RA2E2, connect VREFH0 to VCC and VREFL0 to VSS if the 12-bit ADC is not used.

1.1 References

Further information regarding the power supply for the RA MCU Group can be found in the following documents:
- R01UH0852 RA2E1 Group, RA2E1 Group User’s Manual: Hardware
- R01UH0919 RA2E2 Group, RA2E2 Group User’s Manual: Hardware
- R01UH0853 RA2L1 Group, RA2L1 Group User’s Manual: Hardware

Chapter numbers may vary between devices.
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) or the Digital to Analog Converter (DAC), refer to one of the following chapters in the respective Hardware User’s Manuals for details on how to provide filtered power supplies for these peripherals:
- 12-Bit A/D Converter (ADC12)
- 16-Bit A/D Converter (ADC16)
- 24-Bit Sigma-Delta A/D Converter (SDADC24)
- 8-Bit D/A Converter (DAC8)
- 12-Bit D/A Converter (DAC12)
Table 3. RA2 MCU Groups, User's Manual: Hardware

<table>
<thead>
<tr>
<th>Chapter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>Lists power pins in each package with notes on termination and bypassing.</td>
</tr>
<tr>
<td>Resets</td>
<td>Discusses the Power-On Reset and how to differentiate this from other reset sources.</td>
</tr>
<tr>
<td>Low Voltage Detection</td>
<td>Provides details on the Low-Voltage Detection Circuit that can be used to monitor the power supply.</td>
</tr>
<tr>
<td>Low Power Modes</td>
<td>Using low power modes may allow you to reduce the voltage of the power supply. See this chapter for details on how operating modes affect power supply requirements.</td>
</tr>
<tr>
<td>12-Bit A/D Converter</td>
<td>If you plan to use the on-chip A/D or D/A converters, these chapters give details on how to provide filtered power supplies for these peripherals.</td>
</tr>
<tr>
<td>16-Bit A/D Converter</td>
<td></td>
</tr>
<tr>
<td>24-Bit Sigma-Delta A/D Converter</td>
<td></td>
</tr>
<tr>
<td>8-Bit D/A Converter</td>
<td></td>
</tr>
<tr>
<td>12-bit D/A Converter</td>
<td></td>
</tr>
<tr>
<td>Clock Generation Circuit</td>
<td>Provides detailed descriptions on how to configure and use the available clocks, including PCB design recommendations.</td>
</tr>
</tbody>
</table>

2. Emulator Support

RA2 MCU devices have an emulator interface that supports both debugging using SWD communication and serial programming using SCI communication. This emulator makes it easy to switch between debugging and serial programming.

The SWD emulator interface can be connected to an Arm®-standard 10-pin or 20-pin socket. TXD and RXD pins are added for serial programming using SCI communication.

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 related MCU Hardware User’s Manual. See also section 9.5 in this document.
2.1 SWD Interface

Figure 1. SWD Interface Connections

**Note:** The output of the reset circuit of the user system must be open collector.
2.2 Serial Programming Interface using SCI

**Figure 2. Serial Programming Interface using SCI Connections**

**Note**: The output of the reset circuit of the user system must be open collector.
2.3 Combination Debug Interface

Figure 3. Debug Interface for both SWD and Programming with SCI Connections

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

3. MCU Operating Modes

The RA2 MCU series can enter one of two modes after reset: Single-chip mode or SCI/USB boot mode. The boot mode is selected by the MD pin:

Table 4. Operating Modes Available at Reset

<table>
<thead>
<tr>
<th>Operating Mode</th>
<th>MD</th>
<th>On-Chip Flash Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-chip mode</td>
<td>1</td>
<td>Enable</td>
</tr>
<tr>
<td>SCI/USB boot mode</td>
<td>0</td>
<td>Enable</td>
</tr>
</tbody>
</table>

Figure 4 shows operating mode transitions as determined by the mode-setting pin.
Figure 4. Mode Setting Pin Level and Operating Mode

Note that USB boot mode is only available on devices that have the USB Full-Speed peripheral, such as RA2A1. For the RA2A1 MCU group, USB boot mode doesn’t exist in 32-pin products.

A typical MCU boot mode circuit includes a jumper and a couple of resistors to allow selections to connect the MD pin to VCC or Ground.

Figure 5. Typical Circuit for MCU Boot Mode Selection

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 of setting are different for the two areas.

The registers are detailed in the Option Setting Memory chapter in the Hardware User’s Manual.

The flash option registers occupy space in the code flash memory map. Although the registers are located in a portion of the flash memory that was 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 flash option registers 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 RA2A1. The Option Setting Memory may be different for each device. Please consult MCU User’s Manual for the specific device details.
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
  - Watchdog Timer (WDT) auto start
  - WDT timeout, frequency, windowing, and interrupt type

- **OFS1 register**
  - LVD0 enable after reset
  - HOCO startup after reset
Renesas FSP Configurator supports setting of option memory in BSP settings, as shown in Figure 7 for RA2A1 MCU.

![Figure 7. Option Memory Settings in FSP Configuration for RA2A1 MCU](image)

### 5. Clock Circuits

Most RA2 MCUs have six primary oscillators. Five of these may be used as the source for the main system clock. The remaining oscillator is dedicated to the Independent Watchdog Timer. 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. For some devices, the clock distribution also includes additional clocks for ADC and USB peripherals. Note that RA2E2 devices do not have external crystal or clock input pins. Refer to the Hardware User's Manual’s Clock Generation Circuit chapter for the block diagram of the clock generation circuit.

### Table 5. RA2 Oscillators

<table>
<thead>
<tr>
<th>Oscillator</th>
<th>Input Source</th>
<th>Frequency</th>
<th>Primary Uses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main clock(^4) (MOSC)</td>
<td>External crystal/resonator</td>
<td>1 MHz to 20 MHz(^3)</td>
<td>Main system clock (ICLK), Peripheral clocks, CAN clock, CAC clock, CLKOUT, flash clock(^3), SDADC clock(^1)</td>
</tr>
<tr>
<td></td>
<td>-or-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>External clock</td>
<td>Up to 20 MHz</td>
<td></td>
</tr>
<tr>
<td>Sub-clock(^4) (SOSC)</td>
<td>External crystal/resonator</td>
<td>32.768 kHz</td>
<td>Real-time clock, main system clock (ICLK) in low power modes, CLKOUT, AGT clock, CAC clock</td>
</tr>
<tr>
<td>High-speed on-chip (HOCO)</td>
<td>On-chip oscillator</td>
<td>24/32/48/64 MHz</td>
<td>Main system clock (ICLK), Peripheral clocks, CAC clock, CLKOUT, USB clock(^2), SDADC clock(^1)</td>
</tr>
<tr>
<td>Middle-speed on-chip (MOCO)</td>
<td>On-chip oscillator</td>
<td>8 MHz</td>
<td>Main system clock (ICLK) at startup, Peripheral clocks, CLKOUT, CAC clock</td>
</tr>
<tr>
<td>Low-speed on-chip (LOCO)</td>
<td>On-chip oscillator</td>
<td>32.768 kHz</td>
<td>Main system clock (ICLK) in low power modes and during main oscillator stop detection, Peripheral clocks, Systick timer, AGT clock, CLKOUT, CAC clock, Real-Time clock, Independent Watchdog Timer clock</td>
</tr>
<tr>
<td>Independent Watchdog (IWDT)</td>
<td>On-chip oscillator</td>
<td>15 kHz</td>
<td>Independent Watchdog Timer clock, CAC clock</td>
</tr>
</tbody>
</table>

Note 1: SDADC clock is only present on RA2A1 devices.
2: USB clock is only present on RA2A1 devices.
3: On RA2A1 devices, the flash clock (FCLK) is driven independently from MOSC, and can be sourced from MOSC, SOSC, HOCO, MOCO, or LOCO.
4: Not present on RA2E2

Some devices, such as RA2E2, do not include the option for an external crystal or clock. In this case, the oscillator sources for the main clock are limited to the on-chip oscillators.

#### 5.1 Reset Conditions

After reset, RA2 MCUs begin running with the middle-speed on-chip oscillator (MOCO) as the main clock source. At reset, the main oscillator is stopped by default. The HOCO and 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 can be found in the AC Characteristics section of the Electrical Characteristics chapter in the MCU Hardware User's Manual.

### Table 6. Frequency Range for RA2 MCU Internal Clocks

<table>
<thead>
<tr>
<th></th>
<th>ICLK (^4)</th>
<th>PCLKB</th>
<th>PCLKD</th>
<th>FCLK (^1)</th>
<th>UCLK (^2)</th>
<th>CANMCLK (^3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Frequency [MHz]</td>
<td>48</td>
<td>32</td>
<td>64</td>
<td>32</td>
<td>48</td>
<td>20</td>
</tr>
<tr>
<td>Minimum Frequency [MHz]</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>1</td>
<td>48</td>
<td>1</td>
</tr>
</tbody>
</table>

Note 1: Only on RA2A1. The FCLK must run at a frequency of at least 1 MHz when programming or erasing ROM or data flash.
2: For devices with USB peripheral function.
3: For devices with CAN peripheral function.
4: For RA2E1, RA2E2, and RA2L1, ICLK must run at a frequency of at least 1 MHz when programming or erasing ROM or data flash.
5.2.1 Requirements for USB Communications
The USB 2.0 Full-Speed Module (USBFS) requires a 48-MHz USB clock signal (UCLK).

For RA2 devices with a USBFS Module, the HOCO is the clock source for UCLK. Therefore, the HOCO must be configured for 48-MHz operation when the USBFS Module is used.

5.2.2 Requirements for Programming and Erasing ROM or Data Flash
For RA2A1, the FCLK must be at least 1 MHz to perform programming and erasing on internal ROM and data flash.

For all other RA2 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 7. Clock Source Configuration Registers

<table>
<thead>
<tr>
<th>Oscillator</th>
<th>Register</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main clock&lt;sup&gt;1&lt;/sup&gt;</td>
<td>MOSCCR</td>
<td>Starts/stops main clock oscillator</td>
</tr>
<tr>
<td>Sub-clock&lt;sup&gt;1&lt;/sup&gt;</td>
<td>SOSCCR</td>
<td>Starts/stops sub-clock oscillator</td>
</tr>
<tr>
<td>High-speed on-chip (HOCO)</td>
<td>HOCOCR</td>
<td>Starts/stops HOCO</td>
</tr>
<tr>
<td>Middle-speed on-chip (MOCO)</td>
<td>MOCOCR</td>
<td>Starts/stops MOCO</td>
</tr>
<tr>
<td>Low-speed on-chip (LOCO)</td>
<td>LOCOCR</td>
<td>Starts/stops LOCO</td>
</tr>
</tbody>
</table>

Note 1: Not present on RA2E2 devices.

5.4 Writing the System Clock Control Registers
Care should be taken when writing to the individual bit fields in the System Clock Division Control Register (SCKDIVCR) and System Clock Source Control Register (SCKSCR).

When the clock source of the peripheral module clock is switched, the duration of the peripheral module clock cycle becomes longer during the clock source transition period. See Figure 8.

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

Renesas FSP provides a simple, visual clock configuration tool for RA2A1 MCU shown as follows. 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.

5.6 HOCO Accuracy

The internal high-speed on-chip oscillator (HOCO) runs at 24 MHz, 32 MHz, 48 MHz, or 64 MHz for RA2 devices, with a typical accuracy of +/-2% or better. HOCO accuracy specifications are characterized for various ambient operating temperature (Ta) ranges. See the Electrical Characteristics of the Hardware User’s Manual for the clock accuracy specifications in the desired temperature range.

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.

For RA2E2 devices, there are no external crystal or external clock inputs. An internal clock (HOCO, MOCO, or LOCO) must be selected as the main system clock.

### 5.7 Flash Interface Clock

On RA2A1 devices, the Flash Interface Clock (FCLK) is used as the operating clock when programming and erasing internal flash (ROM and DF) and for reading from the data flash. For other RA2 devices, 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 FCLK or ICLK frequency is recommended.

The flash programming 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 (XTAL/EXTAL, 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 RA2 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 RA2 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 10. Example of Crystal Resonator Connection](image)

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

### 5.10 External Clock Input

A digital clock input may be used as the main clock source for most RA2 devices. Figure 12 shows an example of connecting an external clock input. To operate the oscillator with an external clock signal, set the MOMCR.MOSEL bit to 1. The XTAL pin becomes high impedance.

**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 12 or 13 types of resets for Arm® Cortex-M23 devices.
Table 8. RA2 Device Resets

<table>
<thead>
<tr>
<th>Reset Name</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin reset</td>
<td>RES# is driven low</td>
</tr>
<tr>
<td>Power-on reset</td>
<td>VCC rises (voltage detection: VPOR)</td>
</tr>
<tr>
<td>Independent watchdog timer reset</td>
<td>The independent watchdog timer underflows, or a refresh occurs</td>
</tr>
<tr>
<td>Watchdog timer reset</td>
<td>The watchdog timer underflows, or a refresh occurs</td>
</tr>
<tr>
<td>Voltage monitor 0 reset</td>
<td>VCC falls (voltage detection Vdet0)</td>
</tr>
<tr>
<td>Voltage monitor 1 reset</td>
<td>VCC falls (voltage detection Vdet1)</td>
</tr>
<tr>
<td>Voltage monitor 2 reset</td>
<td>VCC falls (voltage detection Vdet2)</td>
</tr>
<tr>
<td>SRAM parity error reset</td>
<td>SRAM parity error detection</td>
</tr>
<tr>
<td>SRAM ECC error reset (^1)</td>
<td>SRAM ECC error detection</td>
</tr>
<tr>
<td>Bus master MPU error reset</td>
<td>Bus master MPU error detection</td>
</tr>
<tr>
<td>Bus slave MPU error reset</td>
<td>Bus slave MPU error detection</td>
</tr>
<tr>
<td>Stack pointer error reset</td>
<td>Stack pointer error detection</td>
</tr>
<tr>
<td>Software reset</td>
<td>Register setting</td>
</tr>
</tbody>
</table>

Note 1: Not present on RA2E1 or RA2E2 devices.

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 of this document, 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.

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\(_{\text{POR}}\).

After VCC has exceeded the power-on reset voltage (V\(_{\text{POR}}\)) and the power-on reset time (t\(_{\text{POR}}\)) 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\(_{\text{POR}}\), a power-on reset is generated. The chip is released from the power-on state after VCC has risen above V\(_{\text{POR}}\) and the t\(_{\text{POR}}\) has elapsed.

After a power on reset, the PORF bit in RSTSR0 is set to 1. Following a pin reset, PORF is cleared to 0.

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 IWDTREF 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 Watchdog Timer Reset

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

When the WDT overflows, a watchdog timer reset is optionally generated (NMI can be generated instead), and the WDTRF bit in RSTSR1 is set to a 1. After a short delay the WDT reset is canceled. Refer to MCU User’s Manual for the specific timing.
6.5 Voltage-Monitoring Resets

The RA2 MCU family includes circuitry that allows the MCU to protect against unsafe operation during brownouts. On-board comparators check the supply voltage against three reference voltages, \( V_{\text{det0}} \), \( V_{\text{det1}} \), and \( V_{\text{det2}} \). As the supply dips below each reference voltage an interrupt or a reset can be generated. The detection voltages \( V_{\text{det0}} \), \( V_{\text{det1}} \), and \( V_{\text{det2}} \) are each selectable from 3 different levels.

When \( V_{CC} \) subsequently rises above \( V_{\text{det0}}, V_{\text{det1}}, \) or \( V_{\text{det2}} \), 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 register OFS1. For more details, see the chapter Low Voltage Detection (LVD) in the Hardware User’s Manual.

After an LVD Reset, the LVDnRF \((n = 0, 1, 2)\) bit in RSTSR0 is set to 1.

6.6 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 \((t_{\text{RESW2}})\) 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.

6.7 Other Resets

Most 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.

6.8 Determination of Cold/Warm Start

The RA2 MCUs allow the user to determine the cause of the reset processing. The CWSF flag in RSTSR2 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.

6.9 Determining the Reset Source

The RA2 MCUs allow the user to determine the reset signal generation source. Read RSTSR0, RSTSR1, and RSTSR0 to determine which reset was the source of the reset. Refer to the Hardware User’s Manual section Determination of Reset Generation Source for the flow diagram.

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.

```c
/* Power on Reset */
if(1 == R_SYSTEM->RSTSR0_b.PORF)
{
    /* Do something */
}

/* Software Reset */
if(1 == R_SYSTEM->RSTSR1_b.SWRF)
{
    /* Do something */
}
```
7. Memory

The RA2 MCUs support a 4-GB linear address space ranging from 0000 0000h to FFFF FFFFh that can contain program, data, and external memory bus. 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 memory map are regions for on-chip RAM, peripheral I/O registers, program code flash, and data flash.

![RA2A1 Memory Map](image)

**Figure 13. RA2A1 Memory Map**

### 7.1 SRAM

The RA2 MCUs provide on-chip SRAM modules with either parity-bit checking or ECC (Error Correction Code). The following table lists the SRAM specifications for RA2A1 MCU. The SRAM capacity varies by device. Consult the Hardware User’s Manual for specifics.
Figure 14. RA2A1 SRAM Specification

### 7.2 Peripheral I/O Registers

Blocks of peripheral I/O registers appear at various locations in the memory map 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 I/O Registers 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 407E FFFFh.

The Renesas FSP contains 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 RA2 MCUs feature two flash memory sections: code flash and data flash, which vary in size and programmable cycle capacity by device. The Flash Control Unit (FCU) controls programming and erasure of the 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 be updated from time to time 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 RA2A1 devices, the code flash memory uses 64-bit units for programming and 2-KB units for erasure, while the data flash memory uses 8-bit units for programming and 1-KB 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 enables field firmware updates without having to connect an external programming tool.

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.
### 7.3.1 Background Operation

RA2 MCUs support background operations for code flash and 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. Also, the CPU can execute application code from SRAM while the code flash memory is being erased or programmed. The only exception to this rule is that the data flash cannot be accessed during code flash programming or erasing.

### 7.3.2 ID Code Protection

RA2 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 through boot mode or 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 RA2 MCU Hardware User’s Manual for more information.

Renesas FSP configurator provides options to set up ID code protection for RA2 MCUs as shown in Figure 16.
Figure 16. ID Code Setup for RA2A1 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.

7.3.3 Memory Protection Unit

RA2 MCUs have a Memory Protection Unit (MPU). The MPU can protect various MCU regions from illegal access. The choices include allowing both reading and writing, prohibiting writing, and prohibiting writing and reading. Select one of these options by setting the corresponding constant value at the specific memory address. See Memory Protection Unit in the MCU Hardware User's Manual for more details.
Figure 17. MPU Setup for RA2A1 Using Renesas FSP Configurator

Note: MPU settings must be handled carefully to prevent mistakes that may result in blocking accesses to an MCU region.

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 9 lists the association between the PRCR bits and the registers to be protected.

<table>
<thead>
<tr>
<th>b15</th>
<th>b14</th>
<th>b13</th>
<th>b12</th>
<th>b11</th>
<th>b10</th>
<th>b9</th>
<th>b8</th>
<th>b7</th>
<th>b6</th>
<th>b5</th>
<th>b4</th>
<th>b3</th>
<th>b2</th>
<th>b1</th>
<th>b0</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRKEY[7:0]</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

Figure 18. PRCR Register
Table 9. PRCR Protection Bits

<table>
<thead>
<tr>
<th>PRCR bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRC0</td>
<td>Registers related to the Clock Generation Circuit: SCKDIVCR, SCKSCR, MEMWAIT, MOSCCR, HOCOCR, MOCOCR, CKOCR, OSTDCCR, OSTDSCR, MOCOUTCR, HOCOUTCR, MOSCWTCR, MOMICR, SOSCCR, SOMCCR, LOCCOR, LOCOUTCR, HOCOWTCR, SOMRGR, SDADCCKCR, LPOPT</td>
</tr>
<tr>
<td>PRC1</td>
<td>Registers related to the low power modes: SBYCR, SNZCR, SNZEDCR, SNZEDCR0, SNZREQCR, FLSTOP, OPCCR, SOPCCR, SYOCDCR, PSMCR, DCDCTL, VCCSEL</td>
</tr>
<tr>
<td>PRC3</td>
<td>Registers related to the LVD: LVD1CR1, LVD1SR, LVD2CR1, LVD2SR, LVCMPCR, LVDLVLR, LVD1CR0, LVD2CR0</td>
</tr>
<tr>
<td>PRKEY[7:0]</td>
<td>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.</td>
</tr>
</tbody>
</table>

Note: Not all registers may be included on all RA2 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 RA2 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 RA2 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 RA2 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 typically only has two options for the CLKOUT signal.
- Assign peripheral functions with multiple port options last. One example would be the Serial Communications Interface (SCI) which typically has many available port options.
- The Pin Lists section in the RA2 Hardware User’s Manuals show some peripheral port functions with a suffix (such as “_A”) at the end of the function name. For RA2 devices, this type of suffix can be ignored when assigning port functions. Also see Section 16.3 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 Register (PCNTR1), or the PmnPFS registers.
Method 1: Port Control Register (PCNTR1)

- Select a pin as an output by writing a 1 to the Port Direction bit (PDRn) in Port Control Register 1 (PCNTR1).
- 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 RA2 MCUs.
- The Port Output Data bits (PODRn) in the corresponding Port Control Register (PCNTR1) 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 Control Register 2 (PCNTR2) are read only. Read the PIDRn bit in the PCNTR2 register to read the pin state.

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

- The Port Mode Register (PMR) is read/write. It is used to specify whether individual pins function as GPIO or as peripheral pins. Out of reset, all PMR registers are set to 0 which sets all pins to work as GPIO. If a PMR register is set to 1 then that corresponding pin will be used for peripheral functions. The peripheral function is defined by that pin’s MPC setting.
- 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 PCNTR1 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 19. After the GPIO is configured, it can be controlled using HAL layer APIs in FSP.

![Figure 19. Configuring P107 as Output and Low using FSP Configurator](image-url)
9.2.1 Internal Pull-Ups
- Most pins on ports 0 through 9 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 all PCR registers are cleared to 0, therefore all pull-up resistors are disabled.
- 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.

9.2.3 Drive Capacity
The RA2A1 MCU group can enable an output drive capacity. For RA2A1, the drive capacity can be set to low- or middle-drive capacity. The other RA2 MCU groups don’t have this capability.
- Drive capacity switching is controlled by the Drive Capacity Control Register (DSCR) bits in each Port mn Pin Function Select (PmnPFS) register.
- Out of reset, all DSCR registers are cleared to 0. Therefore all pins are set to low drive output. Setting a value other than 00 will change the drive capacity of the output for the selected pin.
- The maximum total output of all pins is dependent on the specific MCU group and device package. Please see the Electrical Characteristics’ section of the Hardware User’s Manual for details.
- The typical differences the drive levels are shown below. Actual output current levels vary by device and pin type. See the specific MCU Hardware User’s Manual for details.

<table>
<thead>
<tr>
<th>Typical output pins</th>
<th>DSCR[1:0]</th>
<th>Drive Capacity</th>
<th>Max (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permissible output current per pin</td>
<td>0 0</td>
<td>Low Drive</td>
<td>4.0</td>
</tr>
<tr>
<td>Permissible output current per pin</td>
<td>0 1</td>
<td>Middle Drive</td>
<td>8.0</td>
</tr>
<tr>
<td>IIC Fast Mode and SPI</td>
<td>1 0</td>
<td>Middle Drive</td>
<td>8.0</td>
</tr>
<tr>
<td>Invalid setting; do not use</td>
<td>1 1</td>
<td>--</td>
<td></td>
</tr>
</tbody>
</table>

Output drive capacity can have a significant impact on overall performance of a board design. The following points should be considered when selecting the drive capacity for each output.
- It is recommended to start with all pins set to low-drive capacity (default) and evaluate the performance.
- Depending on the board layout, pins set to middle-drive capacity may result in higher EMI radiation.
- Long traces may require higher drive capacity for signals to propagate correctly to the receiver.

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 RA2 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 (ASEL bit). If the ASEL bit is set to 1 (use pin as analog input pin) then the pin’s PMR bit should be set for GPIO use and the pin’s PDR bit should be set for input.

Refer to the Peripheral Select Settings for each Product section in the I/O Ports chapter of the Hardware User’s Manual.

In order to ensure that no unexpected edges are input or output on peripheral pins, make sure to clear the Port Mode Control (PMR) bit for the targeted pin before modifying the pin’s PmnPFS register.

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 20 shows an example of enabling SPI0 pins using FSP Pin configuration.

Figure 20. 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:
- Low level
- Falling edge
- Rising edge
- Rising and falling edges

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

Digital filtering is available for IRQ pins. The filters are based on repetitive sampling of the signal at one of four selectable clock rates (PCLKB, PCLKB/8, PCLKB/32, PCLKB/64). This filters out short pulses: any high or low pulse less than 3 samples at the filter rate. The filters are useful for filtering out ringing and
noise in these lines but are much too quick for filtering out long events like mechanical switch bounce. Enabling filtering adds a short bit of latency (the filter time) to the hardware IRQ lines.

- Digital filtering can be enabled for each IRQ pin independently. This is done by setting the IRQ Pin Digital Filter Enable (FLTEN) bit in the IRQCRi register for each IRQ.
- The clock rate for digital filtering is configurable for each IRQ pin independently. This is done by setting the IRQ Pin Digital Filter Setting (FCLKSEL[1:0]) bits in the IRQCRi register for each IRQ.
- Figure 21 and Figure 22 show examples of enabling and configuring IRQ pins using Renesas FSP.

![Figure 21. Enable P004 as IRQ03 Input Using Pin Configurator in Renesas FSP](image-url)
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 RA2 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 100 pins, 64 pins, or 24 pins), set the corresponding bits of nonexistent ports in the PDR register to 1 (output) 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. For example, P007 and P008 on port 0 are only available on 100-pin packages. 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 RA2 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 23 shows an example of the ICU specifications, and Figure 24 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 RA2 MCU Group.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interrupts</td>
<td>Peripheral function interrupts</td>
</tr>
<tr>
<td></td>
<td>• Interrupts from peripheral modules</td>
</tr>
<tr>
<td></td>
<td>Number of sources: 132 (select factor within event list numbers 9 to 141)</td>
</tr>
<tr>
<td>External pin interrupts</td>
<td>• Interrupt detection on low level, falling edge, rising edge, rising and falling</td>
</tr>
<tr>
<td></td>
<td>edges. One of these detection methods can be set for each source.</td>
</tr>
<tr>
<td></td>
<td>• Digital filter function supported</td>
</tr>
<tr>
<td></td>
<td>• 8 sources, with interrupts from IRQ0 to IRQ7 pins.</td>
</tr>
<tr>
<td>DTC control</td>
<td>The DTC can be activated by interrupt sources</td>
</tr>
<tr>
<td>Interrupt sources for NVIC</td>
<td>32 sources</td>
</tr>
<tr>
<td>Non-maskable interrupts</td>
<td></td>
</tr>
<tr>
<td>NMI pin interrupt</td>
<td>• Interrupt from the NMI pin</td>
</tr>
<tr>
<td></td>
<td>• Interrupt detection on falling edge or rising edge</td>
</tr>
<tr>
<td></td>
<td>• Digital filter function supported</td>
</tr>
<tr>
<td>Oscillation stop detection</td>
<td>Interrupt on detecting that the main oscillation has stopped</td>
</tr>
<tr>
<td>interrupt</td>
<td></td>
</tr>
<tr>
<td>WDT underflow/refresh error</td>
<td>Interrupt on an underflow of the down-counter or occurrence of a refresh error</td>
</tr>
<tr>
<td>IWDT underflow/refresh error</td>
<td>Interrupt on an underflow of the down-counter or occurrence of a refresh error</td>
</tr>
<tr>
<td>Voltage monitor 1 interrupt</td>
<td>Voltage monitor interrupt of low voltage detection 1 (LVD_LVD1)</td>
</tr>
<tr>
<td>Voltage monitor 2 interrupt</td>
<td>Voltage monitor interrupt of low voltage detection 2 (LVD_LVD2)</td>
</tr>
<tr>
<td>RPEST</td>
<td>Interrupt on SRAM parity error</td>
</tr>
<tr>
<td>RECCST</td>
<td>Interrupt on SRAM ECC error</td>
</tr>
<tr>
<td>BUSST</td>
<td>Interrupt on MPU bus slave error</td>
</tr>
<tr>
<td>BUSMST</td>
<td>Interrupt on MPU bus master error</td>
</tr>
<tr>
<td>SPEST</td>
<td>Interrupt on CPU stack pointer monitor</td>
</tr>
</tbody>
</table>

Figure 23. RA2A1 ICU Specifications

<table>
<thead>
<tr>
<th>Pin name</th>
<th>I/O</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NMI</td>
<td>Input</td>
<td>Non-maskable interrupt request pin</td>
</tr>
<tr>
<td>IRQ0 to IRQ7</td>
<td>Input</td>
<td>External interrupt request pins</td>
</tr>
</tbody>
</table>

Figure 24. RA2A1 ICU I/O Pins

Figure 25 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.
12. Low Power Consumption

The RA2 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.

RA2 MCUs support three different types of LPM. These are:

- Sleep mode
- Software Standby mode
- Snooze mode

The following table is an overview of the functions available for reducing power consumption.
Table 11. Specifications of Low Power Mode Functions

<table>
<thead>
<tr>
<th>Item</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reducing power consumption by modifying clock signals</td>
<td>The frequency division ratio can be selected independently for the system clock (ICLK), peripheral module clock (PCLKB, PCLKD, CLKOUT), and flash interface clock (FCLK).*1</td>
</tr>
<tr>
<td>Module stop</td>
<td>Functions can be stopped independently for each peripheral module.</td>
</tr>
</tbody>
</table>
| Low power modes | • Sleep mode  
• Software Standby mode  
• Snooze mode |
| Power control modes | Operating power control modes:  
• High-speed mode  
• Middle-speed mode  
• Low-speed mode  
• Low-voltage mode*2  
• Subosc-speed mode |

Notes: 1. For details, see the chapter “Clock Generation Circuit” in the Hardware User’s Manual.  
2. Only RA2A1 group supports low-voltage mode.

RA2L1 MCU group devices can be operated in switch regulator (DCDC) mode. In DCDC mode, only Normal mode and Sleep mode can be supported. The system cannot transition to Software Standby mode or Snooze mode while in DCDC mode.

Additionally, RA2L1 devices in LDO mode and either low-speed mode or Subosc-speed mode cannot transition to DCDC mode. In DCDC mode, only high-speed mode and middle-speed mode are supported.

Table 12 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 12. Low Power Consumption Modes

<table>
<thead>
<tr>
<th>State of operation*1</th>
<th>Sleep Mode</th>
<th>All-Module Clock Stop Mode</th>
<th>Software Standby Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transition condition</td>
<td>WFI instruction while SBYCR.SSBY=0</td>
<td>WFI instruction while SBYCR.SSBY=1 and DPSBYCR.DPSBY=0</td>
<td>Snooze request trigger in Software Standby mode. SNZCR.SNZE=1</td>
</tr>
<tr>
<td>Canceling method</td>
<td>All interrupts. Any reset available in the mode.</td>
<td>Interrupts defined for this mode. Any reset available in the mode.</td>
<td>Interrupts defined for this mode. Any reset available in the mode.</td>
</tr>
<tr>
<td>State after cancellation by an interrupt</td>
<td>Program execution state (interrupt processing)</td>
<td>Program execution state (interrupt processing)</td>
<td>Program execution state (interrupt processing)</td>
</tr>
<tr>
<td>State after cancellation by a reset</td>
<td>Reset state</td>
<td>Reset state</td>
<td>Reset state</td>
</tr>
</tbody>
</table>

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

RA2 devices include register settings that allow the MCU to operate with lower power consumption in Normal mode and Sleep mode. These modes are referred to as the Operating Power Control Modes and are controlled by the OPCCR register.

The following is a summary of the Operating Power Consumption Control modes and the maximum permissible clocking and voltage levels under each mode.
Table 13. Available Oscillators in Each Operating Power Consumption Control Mode

<table>
<thead>
<tr>
<th>Mode</th>
<th>Oscillator</th>
<th>High-speed on-chip oscillator</th>
<th>Middle-speed on-chip oscillator</th>
<th>Low-speed on-chip oscillator</th>
<th>Main clock oscillator</th>
<th>Sub-clock oscillator</th>
<th>IWDT-dedicated on-chip oscillator</th>
</tr>
</thead>
<tbody>
<tr>
<td>High-speed</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Middle-speed</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Low-speed</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Low-voltage*1</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Subosc-speed</td>
<td>N/A</td>
<td>N/A</td>
<td>Available</td>
<td>N/A</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
</tbody>
</table>

Notes 1: Only RA2A1 group supports low-voltage mode.
2: Not present on RA2E2

While it may be possible to set the value in the OPCCCR 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 OPCCCR register will not have any effect in lowering power consumption.

In order to achieve the lowest power numbers, use the maximum possible dividers in the clock generation circuits.

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 26. Set Up Low Power Mode Using Renesas FSP Configurator](image)

**Figure 26. 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:
13. Buses

The buses in RA2 MCUs consist of a main bus and a slave interface. Figure 27 lists the main bus and the slave interface. Figure 28 shows the bus configuration.

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

<table>
<thead>
<tr>
<th>Bus type</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main bus</td>
<td></td>
</tr>
<tr>
<td>System bus (CPU)</td>
<td>• Connected to CPU</td>
</tr>
<tr>
<td>DMA bus</td>
<td>• Connected to on-chip memory and internal peripheral bus.</td>
</tr>
<tr>
<td>Slave Interface</td>
<td></td>
</tr>
<tr>
<td>Memory bus 1</td>
<td>• Connected to code flash memory</td>
</tr>
<tr>
<td>Memory bus 3</td>
<td>• Connected to code flash memory by DMA bus</td>
</tr>
<tr>
<td>Memory bus 4</td>
<td>• Connected to SRAM0</td>
</tr>
<tr>
<td>Internal peripheral bus 1</td>
<td>• Connected to system control related to peripheral modules</td>
</tr>
<tr>
<td>Internal peripheral bus 3</td>
<td>• Connected to peripheral modules (CAC, ELC, I/O Ports, POEG, RTC, WDT, IWDT, IIC, CAN, ADC16, DAC12, DOC, GPT, SCI, SPI, and CRC)</td>
</tr>
<tr>
<td>Internal peripheral bus 5</td>
<td>• Connected to peripheral modules (KINT, AGT, USBFS, DAC8, OPAMP, ACMPHIS, ACMPFLP, SDADC24, and CTSU)</td>
</tr>
<tr>
<td>Internal peripheral bus 7</td>
<td>• Connected to Secure IPs</td>
</tr>
<tr>
<td>Internal peripheral bus 9</td>
<td>• Connected to flash memory (in P/E) and data flash memory</td>
</tr>
</tbody>
</table>

**Figure 27.** RA2A1 Bus Specifications

```c
/* 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);
}
```
13.1 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
- Bus master MPU error
- Bus slave MPU error
- Timeout

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. 24-Bit Sigma-Delta A/D Converter (SDADC24)
The RA2A1 MCU Group includes one 24-bit Sigma-Delta A/D Converter. This is the only MCU Group in the
RA2 family that includes an SDADC. Figure 29 and Figure 30 list specifications of SDADC24 converter.
Refer to the “24-bit Sigma-Delta A/D Converter (SDADC24)” chapter in the RA2A1 MCU Hardware User’s
Manual for more details.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specifications</th>
</tr>
</thead>
</table>
| Input channel                                  | Single-ended input mode: Up to 10 channels*1 (external inputs: 8 channels, input from the internal OPAMP: 2 channels)  
|                                                | Differential input mode: Up to 5 channels (external inputs: 4 channels, input from the internal OPAMP: 1 channel) |
| A/D conversion method                          | Sigma-delta conversion method                                                                       |
| Resolution                                     | 24 bits                                                                                                 |
| Analog input                                   | • Single-ended input  
|                                                |  - Conversion is possible with single-ended input on both positive and negative channels.  
|                                                |  • Differential input.                                                                                |
| Oversampling frequency                         | • Normal A/D conversion mode: 1 MHz  
|                                                |  • Low-power A/D conversion mode: 0.125 MHz.                                                         |
| Power control                                  | • Power-on/power-off can be selected for filtered, PGA, and sigma-delta A/D converter power              |
|                                                |  • Power-on/power-off can be selected for ADBGR, SBIAS, and ADREG power                               |
|                                                |  • VREF reference voltage (SBIAS/VREF) can be set (step: 0.2 V, range: 0.8 to 2.4 V)  
|                                                |  Note: 2.4 V can be set in external VREF (VREF) mode only                                           |
|                                                |  • Sensor reference voltage (SBIAS) can be activated independently.                                    |
| Programmable gain instrumentation amplifier (PGA) | • The gain of an instrumentation amplifier can be set for each channel.  
|                                                |  (∗1 to ∗32 can be set by a combination of \( G_{SET1} \) and \( G_{SET2} \))                      |
|                                                |  - \( G_{SET1} \): range of the previous-stage amplifier: 1, 2, 3, 4, 8  
|                                                |  - \( G_{SET2} \): range of the next-stage amplifier: 1, 2, 4, 8                                     |
|                                                |  • The offset voltage can be adjusted for each channel by using a D/A converter connected to the next-stage amplifier  
|                                                |  - Offset voltage adjustment (-164 to +164 mV, 31 levels: 5 bits)  
|                                                |  • PGA offset can be measured as self-diagnosis                                                       |
|                                                |  • Disconnection detection assist: possible on both positive and negative sides in single-ended input mode. |
| Data registers                                  | • One A/D conversion result register and one A/D conversion average value register.  
|                                                |  - The channel number that corresponds to an A/D conversion result can be checked with a special register  
|                                                |  - An overflow flag is provided for A/D conversion results                                            |
|                                                |  • Differential input model: code is 2’s complement                                                    |
|                                                |  • Single-ended input mode: straight binary                                                           |
|                                                |  • Reverse output can be selected for the conversion results of the single-ended negative channel.     |
| Operation clock                                | • The 24-bit sigma-delta A/D converter reference clock is generated from the peripheral clock output by the clock generation circuit according to the SDADC24 operation mode: 1/1, 1/2, 1/3, 1/4, 1/5, 1/6, 1/16, 1/12, or 1/16 can be selected  
|                                                |  • The SDADC24 reference clock/oversampling clock changes according to the mode as follows:  
|                                                |  - Normal A/D conversion mode: 4 MHz/1 MHz  
|                                                |  - Low-power A/D conversion mode: 500 kHz/125 kHz.                                                    |
|                                                |  Note: When the A/D converter is used in low-power A/D conversion mode, the specified frequency of the SDADC24 reference clock is divided by 8 by using an internal frequency divider. |
| Conversion start condition                     | • Software trigger                                                                                     |
|                                                |  • Hardware trigger (ELC_SDADC24).                                                                    |
| Operation mode                                 | • Continuous scan mode                                                                                 |
|                                                |  • Single scan mode                                                                                   |
|                                                |  • One-shot operation.                                                                                 |
| Oversampling rate                              | • 64, 128, 256, 512, 1024, or 2048 can be selected                                                   |
|                                                |  • Can be set for each channel.                                                                        |

Figure 29. SDADC24 Specifications (1 of 2)
15. Operational Amplifier (OPAMP) with Configurable Switches

The RA2A1 MCU has Operational Amplifiers (OPAMP) that can be used to amplify small analog input voltages and output the amplified voltages. The MCU has a total of three differential operational amplifier units, each with two input pins and one output pin.

The operational amplifiers have the following functions:

- OPAMP0 and OPAMP1 can be used to input signals to the Low-Power Analog Comparator (ACMPLP) and the 24-bit Sigma-Delta A/D Converter (SDADC24).
- High-speed mode (high-current consumption), middle-speed mode (medium-current consumption), and low-power mode (slow-speed response) are supported and any mode can be selected based on trade-offs between the response speed and current consumption.
- Operation can be started by a trigger from the Asynchronous General Purpose Timer (AGT).
- Operation can be stopped by a 16-bit A/D conversion end trigger.
- All units have switches that can select input signals. Additionally, OPAMP0 has a switch that can select the output pin.
- The output of the OPAMP can be output from the AMP0O to AMP2O pins without passing through the switch.
- The I/O signals of all OPAMP units can be used for the input signals to the ADC16.
- The signal output from the DAC8 and DAC12 can be used as the positive input signal for each OPAMP.
- A voltage follower circuit can be configured by connecting the output signal from an OPAMP to the negative input signal of the same OPAMP.

Renesas FSP provides an Operational Amplifier driver and driver configurator to set up OPAMPs, pin connections, etc. Figure 31 shows an example of setting up OPAMP0 to form a voltage follower with P500, P501, P502 as plus input pin, minus input pin, output pin respectively. The plus input is also set up to connect to DAC12 output internally in the following example.
Figure 31. Example of Setting RA2A1’s OPAMP Using FSP Configurator

After an OPAMP is set up by FSP Configurator, you can use OPAMP driver’s APIs such as R_OPAMP_Open and R_OPAMP_Start to initialize and start OPAMP operation. Refer to the Operational Amplifier (OPAMP) chapter in RA2A1 Hardware User’s Manual and FSP User’s Manual for more details.

16. General Layout Practices

16.1 Digital Domain vs. Analog Domain

Renesas RA2 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 AVCC0 and the associated AVSS0) 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 RA2 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 supply voltages and 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 voltage supply and 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.

16.2 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.

16.3 Signal Group Selections

Some pin names have an added _A, _B, _C, _D, _E, or _F suffix to indicate signal groups. For RA2 devices, the suffix can be ignored when assigning functionality. 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.

17. References

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

<table>
<thead>
<tr>
<th>Reference</th>
<th>Document Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>R01UH0888</td>
<td>Renesas RA2A1 Group, User’s Manual: Hardware</td>
</tr>
<tr>
<td>2</td>
<td>R01UH0852</td>
<td>Renesas RA2E1 Group, User’s Manual: Hardware</td>
</tr>
<tr>
<td>3</td>
<td>R01UH0853</td>
<td>Renesas RA2L1 Group, User’s Manual: Hardware</td>
</tr>
<tr>
<td>4</td>
<td>R01UH0919</td>
<td>Renesas RA2E2 Group, User’s Manual: Hardware</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>Rev.</th>
<th>Date</th>
<th>Description</th>
<th>Page</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>Sep.14.21</td>
<td></td>
<td>—</td>
<td>Initial release</td>
</tr>
<tr>
<td>1.01</td>
<td>Feb.24.22</td>
<td></td>
<td>—</td>
<td>Minor corrections</td>
</tr>
</tbody>
</table>


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 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.

5. 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; underwater 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.

6. 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 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.

7. 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.

8. 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.

9. 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.

10. 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.

11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.

12. 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.4.0-1 November 2017)

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

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

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

© 2022 Renesas Electronics Corporation. All rights reserved.