瑞萨 RA MCU

RA4 快速设计指南

简介

本文档回答了常见的问题，并指出了单片机 (MCU) 的一些细节之处，除非通篇阅读硬件手册，否则可能会忽略这些细节。本文档并不适合取代硬件手册，而是对手册的一种补充，重点介绍大多数工程师在开始自己的设计时需要的一些关键项目。本文档还将从应用的角度探讨一些设计决策。

目标 MCU

RA4 MCU

1 电源 5
1.1 参考文档 6
2 仿真器支持 7
2.1 SWD 接口 8
2.2 JTAG 接口 9
2.3 使用 SCI 的串行编程接口 10
2.4 使用 SCI 的串行编程接口：支持 TrustZone® 的产品 11
2.5 多仿真器接口 12
2.6 仿真器连接的软件设置 13
2.6.1 SWD 和 JTAG 接口 13
2.6.2 跟踪端口 13
3 MCU 运行模式 14
3.1 RA4E2、RA4L1 和 RA4C1 MCU 的运行模式 16
4 选项设置区 17
4.1 选项设置区寄存器 18
5 时钟电路 19
5.1 复位条件 20
5.2 时钟频率要求 20
5.2.1 USB 通信要求 21
5.2.2 ROM 或 数据闪存编程和擦除要求 21
5.3 降低时钟生成电路（CGC）功耗 21
5.4 写入系统时钟控制寄存器 21
5.5 时钟设置示例 22
5.6 HOCO 精度 23
5.7 闪存接口时钟 23
5.8 电路板设计 23
5.9 外部晶振选择 23
5.10 外部时钟输入 24
6 复位要求和复位电路 25
6.1 引脚复位 26
6.2 上电复位 26
6.3 VRTC 域上电复位 26

6.4 VBATT 电压上电复位 26
6.5 独立看门狗定时器复位 26
6.6 看门狗定时器复位 27
6.7 电压监视器复位 27
6.8 深度软件待机复位 27
6.9 软件复位 27
6.10 其他复位 27
6.11 冷/热启动的确认 27
6.12 确定复位源 28
7 TrustZone®支持 29
7.1 Arm® TrustZone 技术的实施 29
7.2 支持 TrustZone 的仿真器 30
7.2.1 设备生命周期管理 32
7.2.2 ID 验证 32
8 存储器 33
8.1 SRAM 34
8.2 待机 SRAM 34
8.3 外设 I/O 寄存器 35
8.4 片上闪存 35
8.4.1 后台操作 36
8.4.2 ID 代码保护 36
8.4.3 闪存块保护 37
8.4.4 存储器保护单元 37
8.5 字节顺序的限制 38
9 寄存器写保护 38
10 I/O 端口配置 38
10.1 多功能引脚选择设计方案 39
10.2 设置端口用作 GPIO 39
10.2.1 内部上拉 40
10.2.2 开漏输出 40
10.2.3 驱动能力 40
10.3 设置和使用端口外设功能 41

10.4 设置和使用 IRQ 引脚 42
10.5 未使用的引脚 43
10.6 不存在的引脚 44
10.7 电气特性 44
11 模块停止功能 44
12 中断控制单元 45
13 低功耗 46
14 外部总线 50
14.1 总线错误监视 52
14.1.1 总线错误类型 52
14.1.2 发生总线错误时的处理 52
15 一般布局实践 53
15.1 数字域与模拟域 53
15.2 高速信号设计注意事项 54
15.3 信号组选择 54
16 参考文献 55
网页和支持 56

1 电源

RA MCU 有数字电源和模拟电源。电源使用以下引脚。

表 1. 数字电源

符号	功能名称	说明
VCC	电源	电源引脚。连接到系统电源。通过放置在 VCC 引脚附近的 0.1 µF 电容将此引脚连接至 VSS。
VSS	接地	接地
VCL	电源	通过 VCL 引脚附近的 0.1 µF 电容将此引脚连接至 VSS。电容值取决于具体的 MCU 产品群组。注 3
VCL0	电源	通过 VCL0 引脚附近的 0.1 µF 电容将此引脚连接至 VSS。电容值取决于具体的 MCU 产品群组。注 3
VBATT 注 2	备用电源	备用电源引脚。在没有 VCC 的情况下，为 RTC 和副时钟振荡器供电。当不使用 VBATT 引脚时，连接到 VCC 或 VSS。
VRTC 注4	电源	独立电源引脚。为 RTC 和副时钟振荡器供电。通过放置在 VRTC 引脚附近的 0.1 µF 电容将此引脚连接至 VSS。当不使用 VRTC 引脚时，连接到 VSS。
VCC_USB 注 1, 5	USB FS 电源	全速 USB 电源引脚。将此引脚连接到 VCC。通过放置在 VCC_USB 引脚附近的 0.1 µF 电容将此引脚连接至 VSS_USB。
VSS_USB 注 5	USB FS 接地	全速 USB 接地引脚。将此引脚连接到 VSS。

注：

1. RA4M1 的 VCC_USB 同时支持输入和输出。输入时被用作 USB 收发器的电源。输出时则为 USB LDO 稳压器的输出电压，此时需要连接外部电容。如果不使用 USB LDO 稳压器，请将此引脚连接至 VCC。使用稳压器时，请通过 0.1 µF 电容将此引脚连接至 VSS。
2. RA4E2、RA4L1、RA4C1无此引脚。
3. 请通过对应 MCU 的用户手册查找输出电容的值。
4. RA4C1有此引脚。
5. RA4C1无此引脚。

表 2. 模拟电源

符号	功能名称	说明
AVCC0	模拟电源	各个模块的模拟电压电源引脚。将此引脚连接到与 VCC 引脚相同的电压。
AVSS0	模拟接地	各个模块的模拟地。将此引脚连接到与 VSS 引脚相同的电压。
VREFH0 注 1	12 位 ADC 高基准电压	12 位 A/D 转换器的参考电源引脚。如果不使用这些功能，则将此引脚连接到 AVCC0。
VREFL0 注 1	12 位 ADC 低基准电压	12 位 A/D 转换器的模拟参考接地引脚。如果不使用这些功能，则将此引脚连接至 VSS。
VREFH 注 2, 3	12 位 ADC 和 DAC 模拟电源	12 位 A/D 转换器、单元 1（如有）和 D/A 转换器的参考电压输入引脚。如果不使用这些功能，则将此引脚连接到 AVCC0。
VREFL 注 2, 3	12 位 ADC 和 DAC 模拟地	12 位 A/D 转换器、单元 1（如有）和 D/A 转换器的参考接地引脚。如果不使用这些功能，则将此引脚连接至 VSS。

注：

1. 对于RA4M1而言，适用于14位 ADC。
2. RA4L1无此引脚。这些MCU 的 AVCC0 和 AVSS0 也可用作 12 位 ADC的模拟参考电压。
3. RA4C1无此引脚。

1.1 参考文档

有关 RA MCU 系列的电源的更多信息，请参见以下文档：

- R01UH0887 RA4M1 系列，RA4M1 系列用户手册：硬件
- R01UH0892 RA4M2 系列，RA4M2 系列用户手册：硬件
- R01UH0893 RA4M3 系列，RA4M3 系列用户手册：硬件
- R01UH0929 RA4E1 系列，RA4E1 系列用户手册：硬件
- R01UH0996 RA4E2 系列，RA4E2 系列用户手册：硬件
- R01UH1081 RA4L1 系列，RA4L1 系列用户手册：硬件
- R01UH1137 RA4C1 系列，RA4C1 系列用户手册：硬件
- R01UH0999 RA4T1 系列，RA4T1 系列用户手册：硬件

Arm Cortex-M4 和 Arm Cortex-M33 产品之间的章节编号可能会有所不同。

Overview 一章列出了每个封装中的电源引脚以及建议的旁路电容。

Resets 一章探讨了上电复位以及如何将其与其他复位源区分开。

Low Voltage Detection 一章详细介绍了可用于监视电源的低电压检测电路。**Option-Setting Memory** 一章还介绍了如何在启动时自动使能低电压检测 0 电路。

Battery Backup Function 一章介绍了如何为 RTC 和副时钟振荡器提供备用电池。

若要使用片上模数转换器 (ADC) 或数模转换器 (DAC)，请参见 *12-Bit A/D Converter (ADC12)* 或 *14-Bit A/D Converter (ADC14)* 及 *12-Bit D/A Converter (DAC12)*，了解有关如何为这些外设提供经过滤波的电源的详细信息。

表 3. RA4 MCU 系列用户手册：硬件

章节名称	说明
Overview	列出了各种封装产品中的电源引脚，并附有有关端接和旁路的说明。请参阅引脚功能表。
Resets	叙述了上电复位以及如何将其与其他复位源区分开来。
Voltage Detection Circuit	提供可用于监控电源的低电压检测电路的详细信息。
Low Power Modes	使用低功耗模式可以降低 MCU 的功耗。有关运行模式如何影响 MCU 中各种功能模块的电源要求的详细信息，请参阅本章。
Battery Backup Function	演示如何为 RTC 和副时钟振荡器提供备用电池。
12-bit A/D Converter 14-bit A/D Converter 12-bit D/A Converter	如果计划使用片上 A/D 或 D/A 转换器，可以参考这些章节在如何为这些外设提供滤波电源方面的详细说明。
Clock Generation Circuit	详细说明了如何配置以及如何使用可用时钟，包括 PCB 设计建议。

2 仿真器支持

RA4 MCU 产品支持使用 SWD 或 JTAG 通信进行调试，并可使用 SCI 通信进行串行编程。借助该仿真器，可以轻松地在调试和串行编程之间进行切换。

部分 RA4 MCU（如 RA4E2）不支持 JTAG 或 TRACE 功能。此类 MCU 必须使用 SWD 接口进行调试。

SWD 或 JTAG 仿真器接口应连接到符合 Arm® 标准的 10 引脚或 20 引脚插座。添加了 MD、TXD 和 RXD 引脚，以使用 SCI 通信进行串行编程。

必须使用串行编程接口对 Arm® TrustZone® IDAU 边界寄存器设置进行编程。对于支持 TrustZone® 的产品，建议使用板上线“或”门电路连接 P300/SWCLK/TCK 和 P201/MD 引脚，以同时使用调试和串行编程。

JTAG、SWD 和 SCI 信号必须有上拉电阻才符合 Arm® 标准。如果没有正确的上拉电阻，接口可能无法正常工作。而 RA4 MCU 中，上述信号引脚上的内部上拉电阻是默认使能的。当内部上拉电阻被使能时，这些信号便不再需要外部电阻。

仿真器支持有助于产品开发和原型设计，一旦进入生产环节，则不再需要仿真器支持。此种情况下，请依照各硬件手册中 Handling of Unused Pins 章节中的内容配置端口，也可参照本文档中 10.5 章节的内容。

2.1 SWD 接口

下图为使用串行线调试（SWD）时调试接口的典型连接方式。

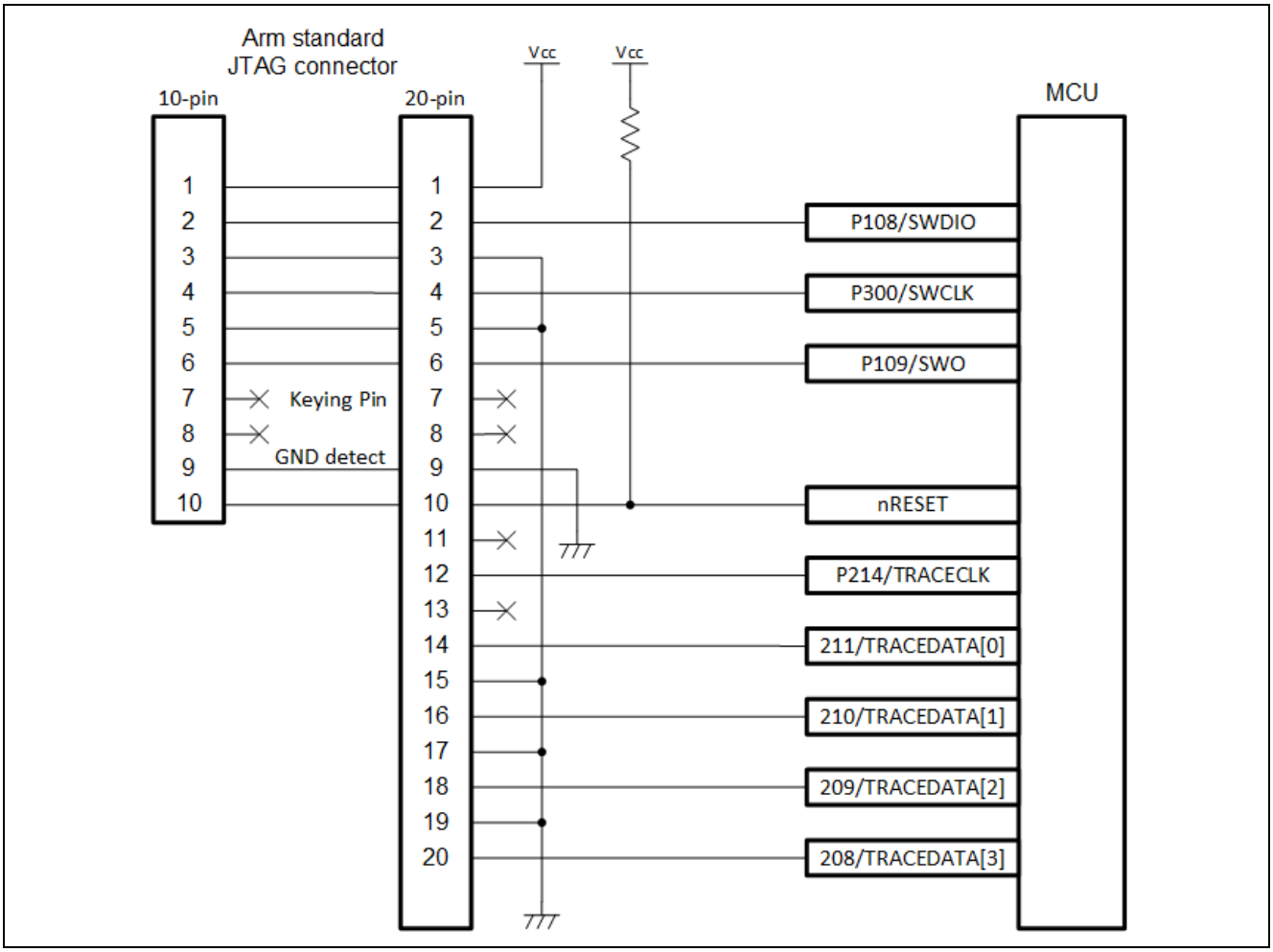


图 1. SWD 接口连接

注：

1. 用户系统复位电路的输出必须为集电极开路。
2. Arm Cortex-M4 产品（如 RA4M1）不具备 TRACECLK 和 TRACEDATA[n] 引脚。
3. 部分 RA4 Arm® Cortex®-M33 产品（如 RA4E2、RA4L1 和 RA4C1）不具备 SWO、TRACECLK 和 TRACEDATA[n] 引脚。

2.2 JTAG 接口

下图为使用 Arm® 标准 JTAG 调试接口时，调试接口的典型连接方式。

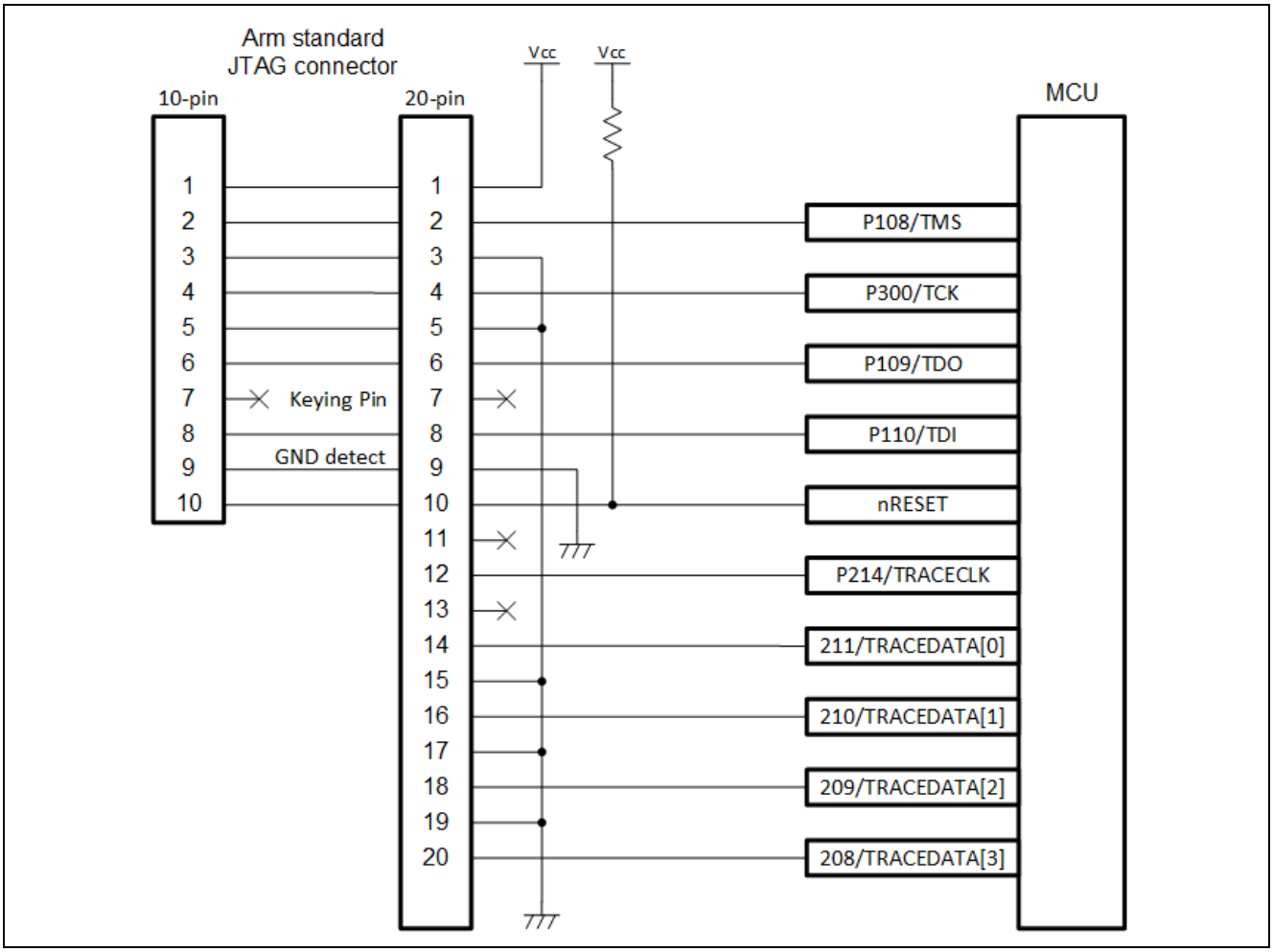


图 2. JTAG 接口连接

注：

1. 用户系统复位电路的输出必须为集电极开路。
2. Arm Cortex-M4 产品（如 RA4M1）不具备 TRACECLK 和 TRACEDATA[n] 引脚。
3. 部分 RA4 Arm® Cortex®-M33 产品（如 RA4L1 和 RA4C1）不具备 JTAG 接口。

2.3 使用 SCI 的串行编程接口

下图为使用 SCI 的串行编程接口的典型连接图。

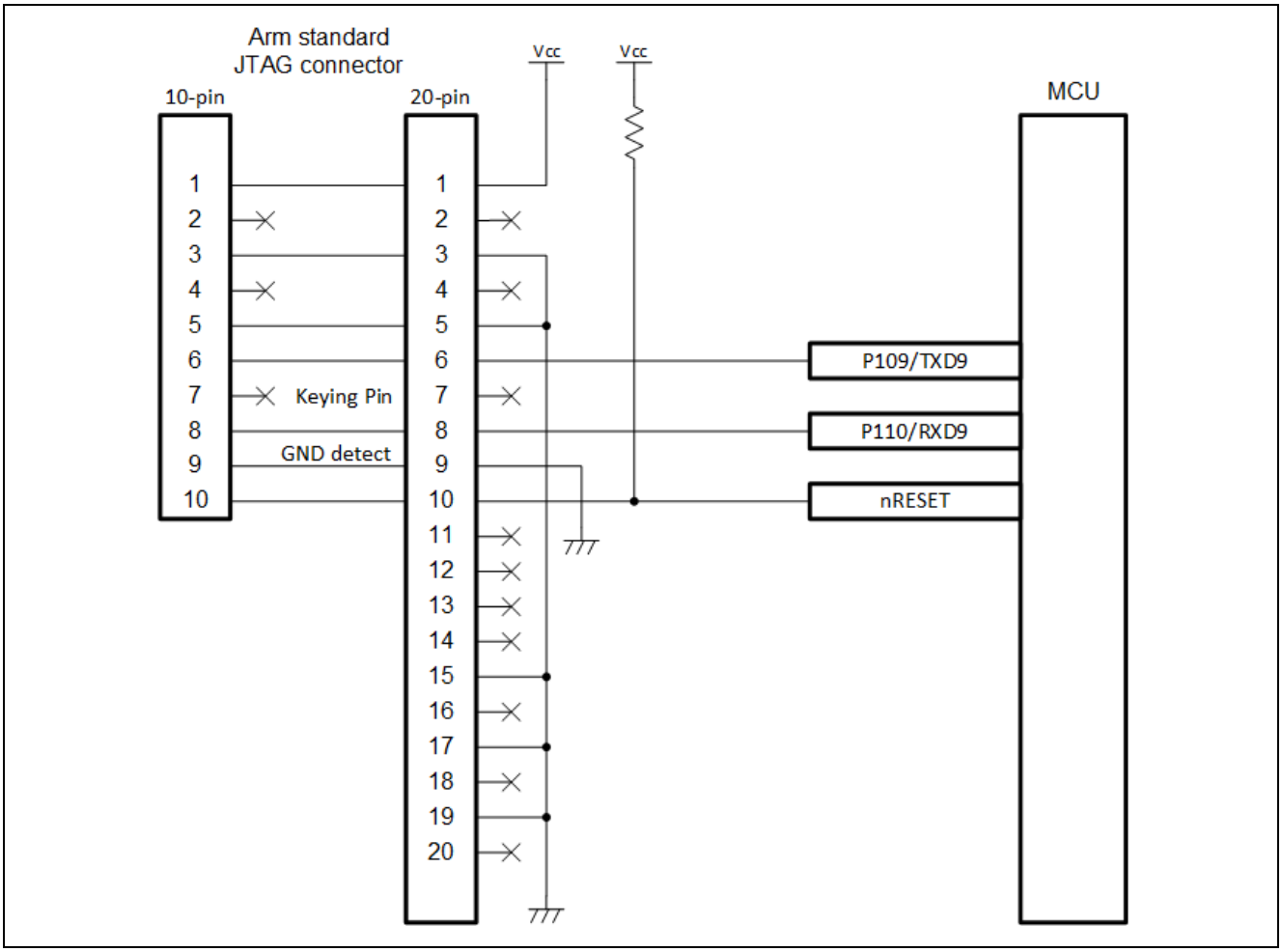


图 3. 使用 SCI 连接的串行编程接口

注：

1. 用户系统复位电路的输出必须为集电极开路。
2. MD 引脚不属于使用 SCI 连接的串行编程接口的部分，但是必须将其设为低电平以启动串行编程接口模式。

2.4 使用 SCI 的串行编程接口：支持 TrustZone® 的产品

图为使用 SCI 进行串行编程时，对于支持 TrustZone 的器件的典型接口连接方式。请注意，P201/MD 增加了额外连接，该连接用于实现对 Arm TrustZone IDAU 边界寄存器设置的编程。

RA4E2 系列 MCU 器件不在此要求范围内，属于例外情况。对于这些器件，IDAU 边界寄存器是在运行时通过应用程序进行设置的。更多信息，请参阅文档 Security Design with Arm® TrustZone® - IP Protection（R11AN0467EU）。

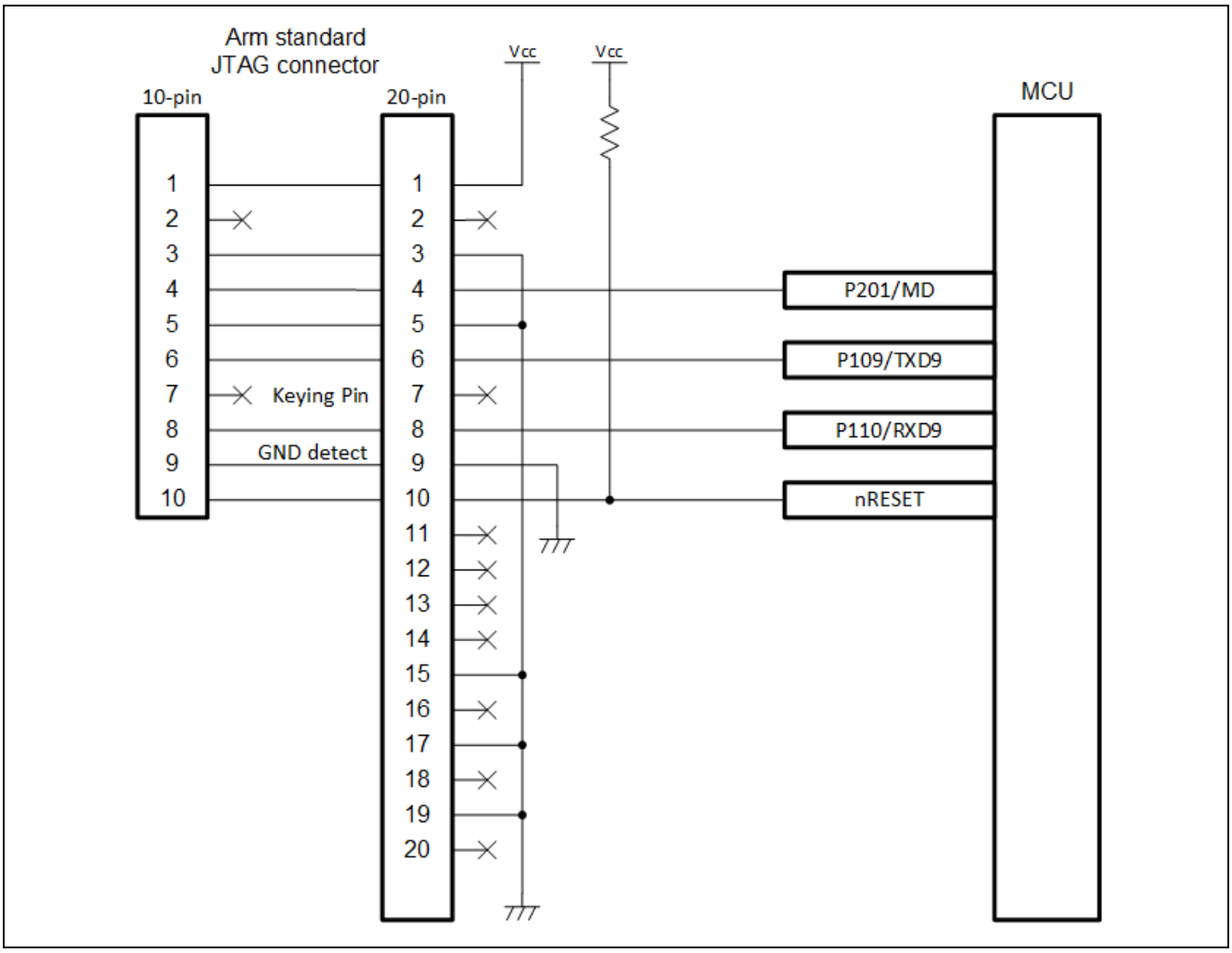


图 4. 使用 SCI 的串行编程接口：支持 TrustZone® 的产品

注：

1. 用户系统复位电路的输出必须为集电极开路。
2. P201/MD 必须连接到标准 ARM JTAG 接口的 Pin4 引脚，以支持对 Arm TrustZone IDAU 边界寄存器设置的编程。

2.5 多仿真器接口

下图为调试接口的典型连接方式，以支持多种仿真器类型，包括 SWD、JTAG、SCI 串行编程以及 Arm® TrustZone® 支持。

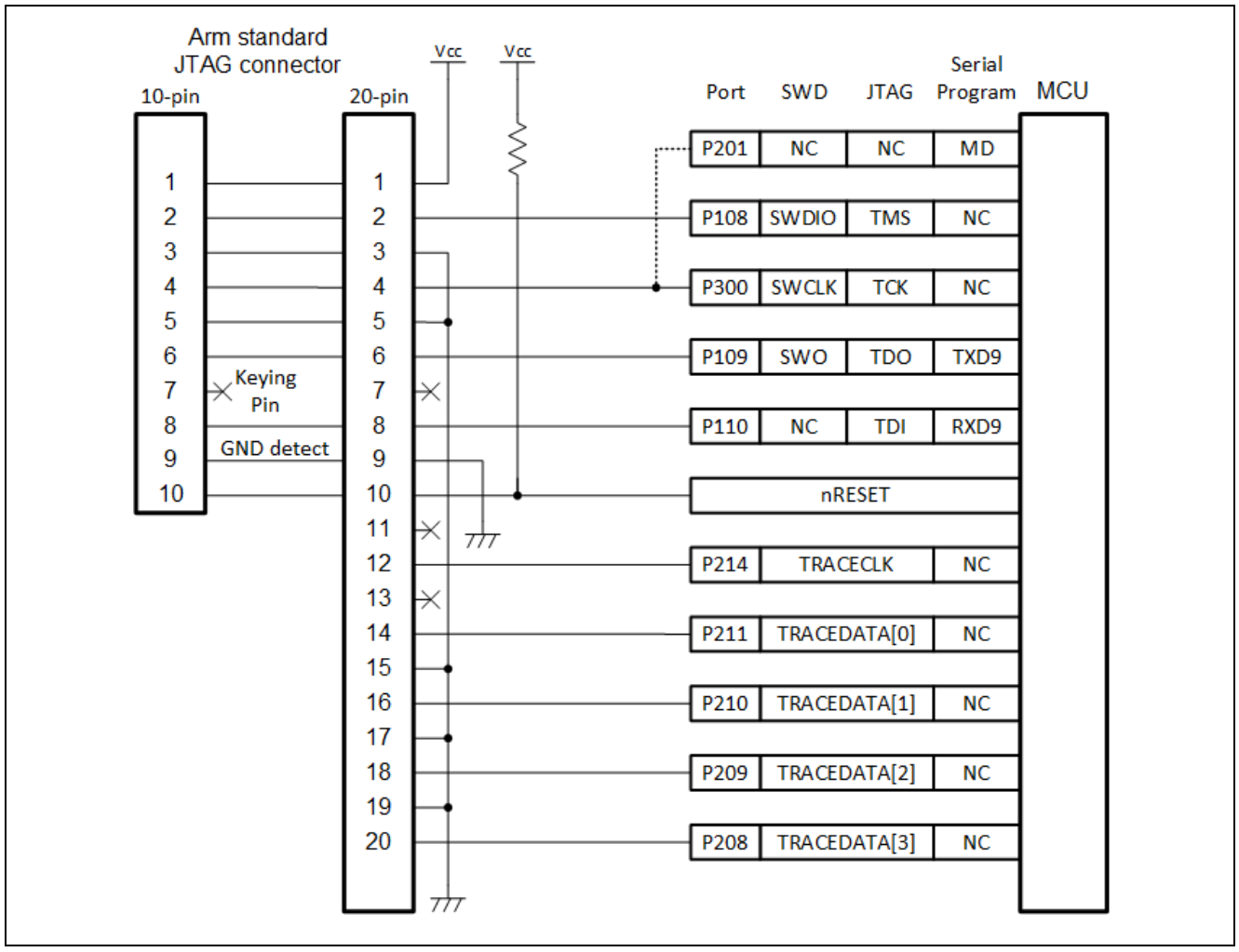


图 5. 多仿真器接口连接

注：

1. 目标产品上的复位电路必须是集电极开路。将 nRESET 信号上拉。请勿在此信号上放置电容，因为这会影响上电复位电路的运行。
2. 要在支持 TrustZone® 的产品上同时使用调试和串行编程，请使用线“或”门电路将 P201/MD 连接到 P300/SWCLK/TCK。
3. Arm Cortex®-M4产品（如RA4M1）不支持TRACECLK 或 TRACEDATA[n]引脚。
4. 部分 RA4 Arm® Cortex®-M33 产品（如 RA4E2、RA4L1 和 RA4C1）不支持 SWO、TRACECLK 和 TRACEDATA[n] 引脚。

将 P201/MD 连接到 P300/SWCLK 可使调试器控制 MCU 运行模式。更多详情请参阅第 3 章内容。

2.6 仿真器连接的软件设置

2.6.1 SWD 和 JTAG 接口

复位后，SWD 和 JTAG 引脚处于默认状态。表 4 为复位后相关引脚的默认值。

表 4. SWD/JTAG 引脚

引脚	P108	P109	P110	P300
功能	TMS/SWDIO	TDO/SWO	TDI	TCK/SWCLK

2.6.2 跟踪端口

在 RA4 产品中，4 位跟踪端口接口单元 (TPIU) 和串行线输出 (SWO) 提供跟踪输出。

必须先使能跟踪端口和时钟，调试器脚本才能使用它们。使用跟踪端口功能时，请避免将跟踪引脚用于其他功能。

表 5 为跟踪端口以及相关功能。

表 5. 跟踪端口

引脚	P208	P209	P210	P211	P214
功能	TDATA3	TDATA2	TDATA1	TDATA0	TCLK

Arm® Cortex®-M4 产品（如 RA4M1）不支持 TRACECLK 或 TRACEDATA[n]引脚。这些产品的跟踪数据可通过 SWO 引脚获取。

部分基于 Arm® Cortex®-M33 的产品（例如 RA4E2、RA4L1 和 RA4C1）不支持 TRACECLK、TRACEDATA[n] 和 SWO 引脚。这些产品的跟踪数据可通过 ETB 寄存器获取。

有关将 Arm® Cortex®-M33 产品上的跟踪端口与 SEGGER J-Trace Pro 一起使用的示例，请参见以下链接：

https://wiki.segger.com/RA6M4

有关将 Arm® Cortex®-M4 产品上的跟踪端口与 SEGGER J-Trace Pro 一起使用的示例，请参见以下链接：

https://wiki.segger.com/J-Link_Renesas_RA6M3

也可以在运行时通过使用 Renesas FSP 中的 Pin Configurator 来使能跟踪端口，但是在这种情况下，某些跟踪数据可能会丢失。

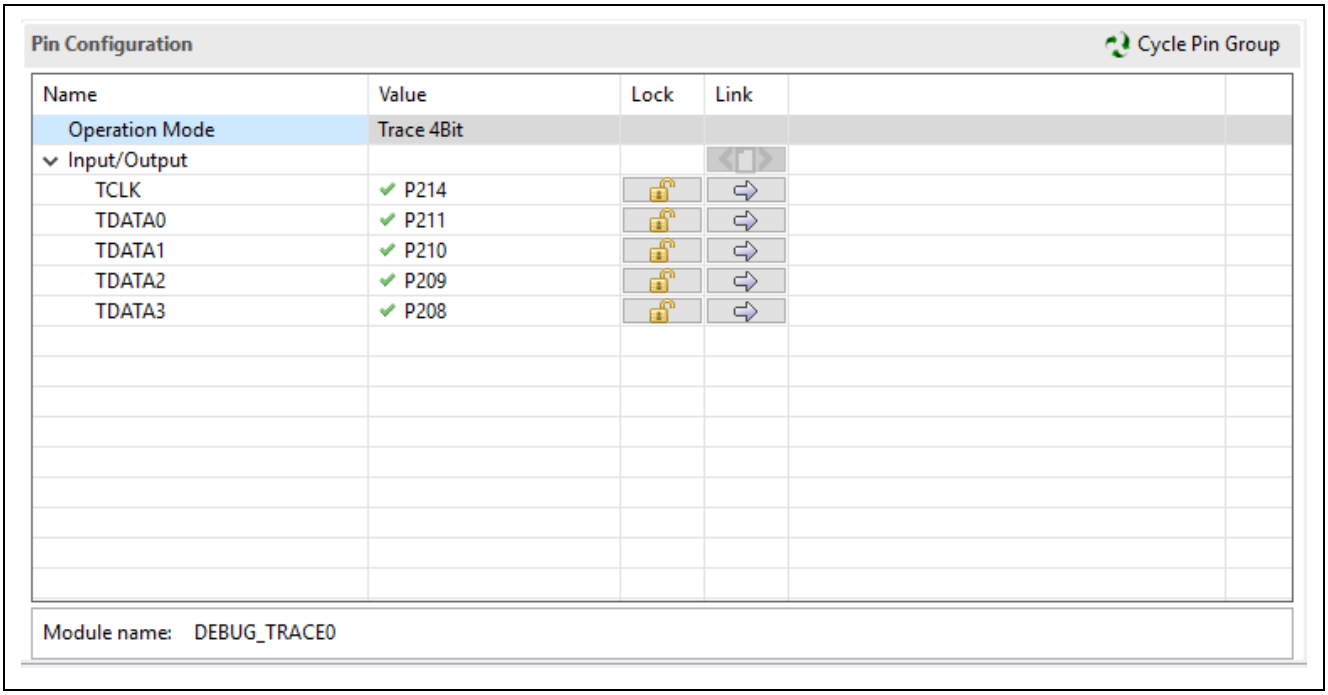


图 6. 使用 FSP Configurator 启用 RA4M2 跟踪端口

3 MCU 运行模式

大多数 RA4 MCU 复位后可以进入两种模式之一：单芯片模式或 SCI/USB Boot 模式。Boot 模式由 MD 引脚选择。RA4C1 不支持 USB boot 模式。

表 6. 复位时可用的运行模式

运行模式	MD	片上闪存	外部总线
单芯片/JTAG Boot 模式	1	使能	禁用
SCI/USB Boot 模式	0	使能	禁用

图 7 为由 Mode-Setting（MD）引脚决定的运行模式转换过程。

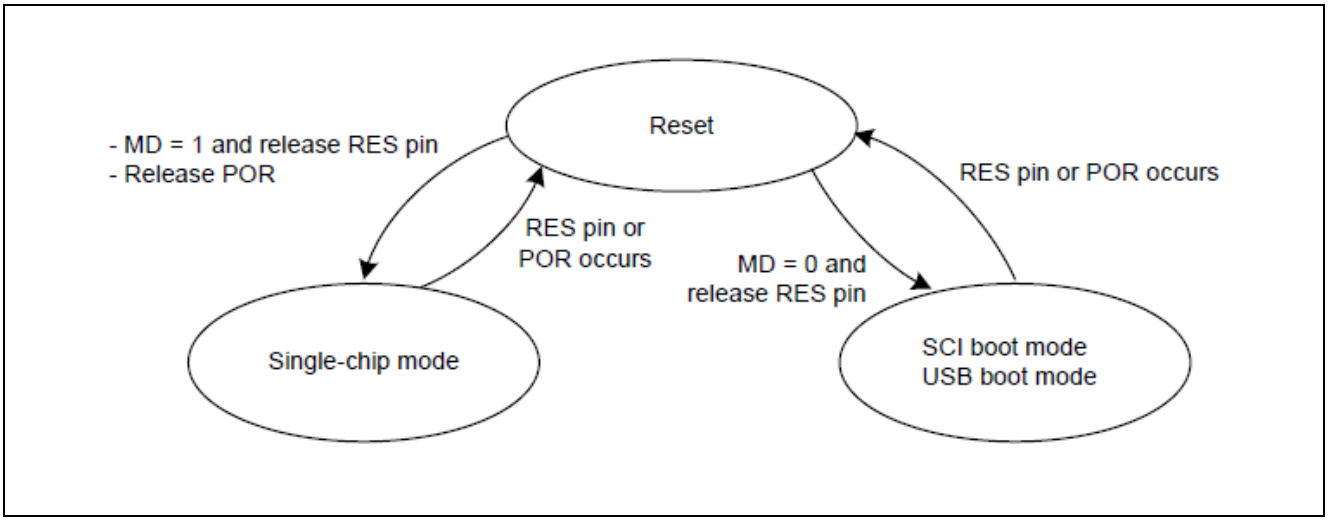


图 7. Mode-Setting 引脚电平与运行模式

典型的 MCU boot 模式电路包括一个跳线和一对电阻，允许选择将 MD 引脚连接到 VCC 或接地。RA4 MCU 的 P201/MD 包含一个内部上拉电阻，默认处于启用状态。内部上拉电阻可以代替图 8 中的外部上拉电阻。

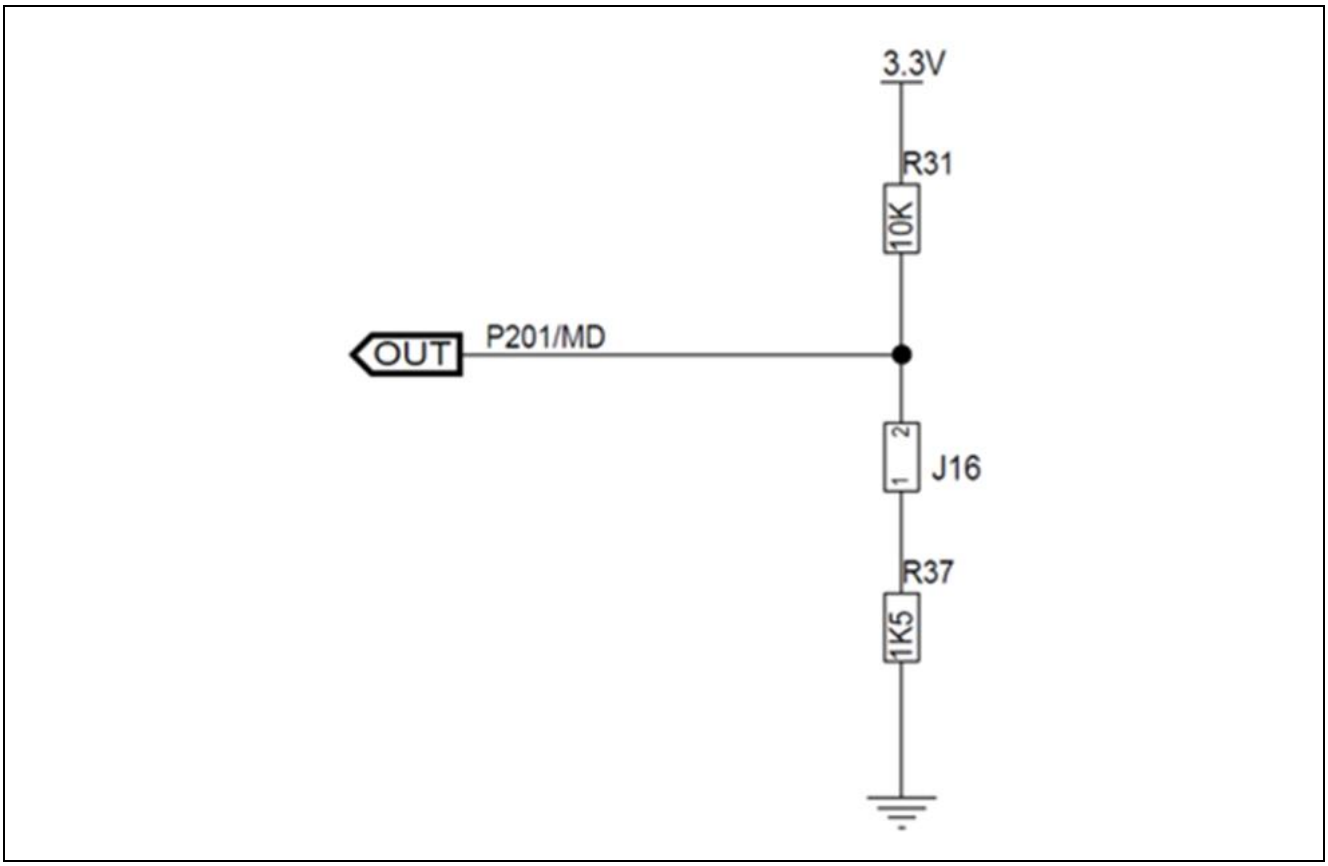


图 8. MCU Boot 模式选择的典型电路

3.1 RA4E2、RA4L1 和 RA4C1 MCU 的运行模式

除单芯片模式、SCI boot 模式和 USB boot 模式外，RA4E2、RA4L1 和 RA4C1 MCU 还包含 SWD boot 模式。RA4C1 不支持 USB boot 模式。

在该模式下，将使用 MCU 内部启动区中存储的内部闪存编程例程（SWD boot 程序）。通过使用 SWD 接口，可以从 MCU 外部对内部闪存（包括代码闪存和数据闪存）进行修改。要进入此模式，需要在 RES 引脚复位期间通过 SWD 接口输入请求。

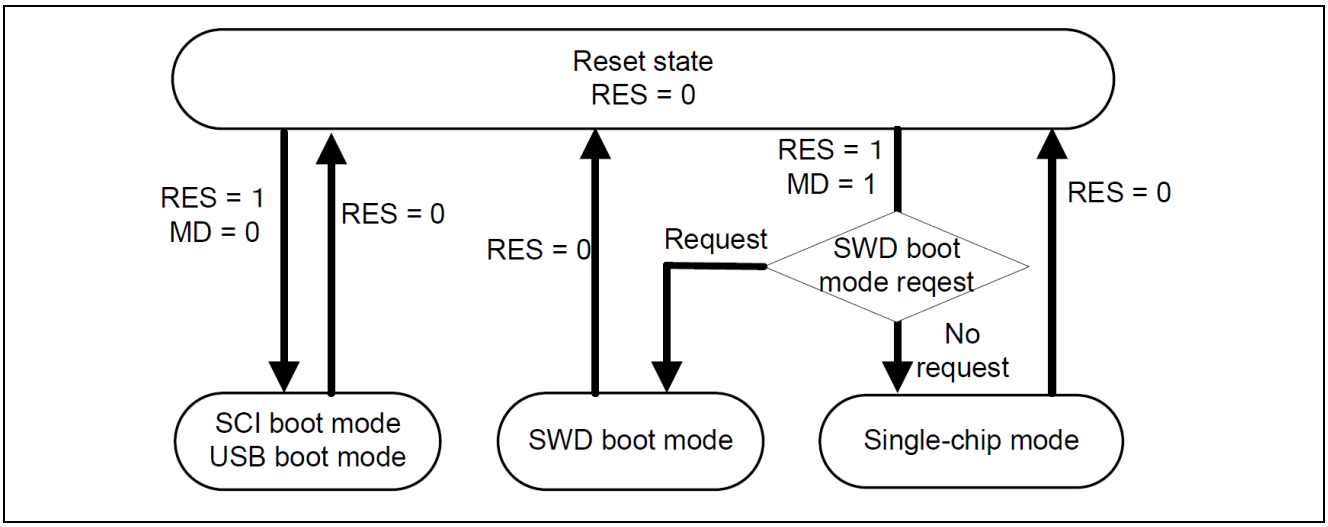


图 9. RA4E2 Mode-Setting 引脚电平与运行模式

4 选项设置区

选项设置存储器用于确定复位后 MCU 的状态。它被分配到闪存的配置设置区域和程序闪存区域。对于 Arm® Cortex®-M33 产品来说，选项设置区的大小和布局可能有所不同。

有关寄存器的详细介绍，请参见硬件手册中的 Option Setting Memory 一章。

闪存选项寄存器在代码闪存映射中占用一定空间。尽管寄存器位于 RA MCU 上保留闪存的一部分中，但是**有些用户可能会无意中将数据存储在这些位置**。用户必须进行检查，确保没有将多余数据写入这些位置，否则可能导致芯片发生意外行为。例如，闪存选项寄存器中的设置可以在复位后立即使能独立看门狗定时器 (IWDT)。如果存储在程序 ROM 中的数据无意中与选项设置存储器寄存器重叠，则有可能在不知情的情况下打开 IWDT。这将导致调试器与电路板之间的通信出现问题。

下图为选项设置存储器，它由 RA4M3（Arm® Cortex®-M33 产品）上的选项功能选择寄存器组成。不同产品的选项设置存储器可能有所不同。请参阅具体产品的 MCU 硬件用户手册以了解详细信息。

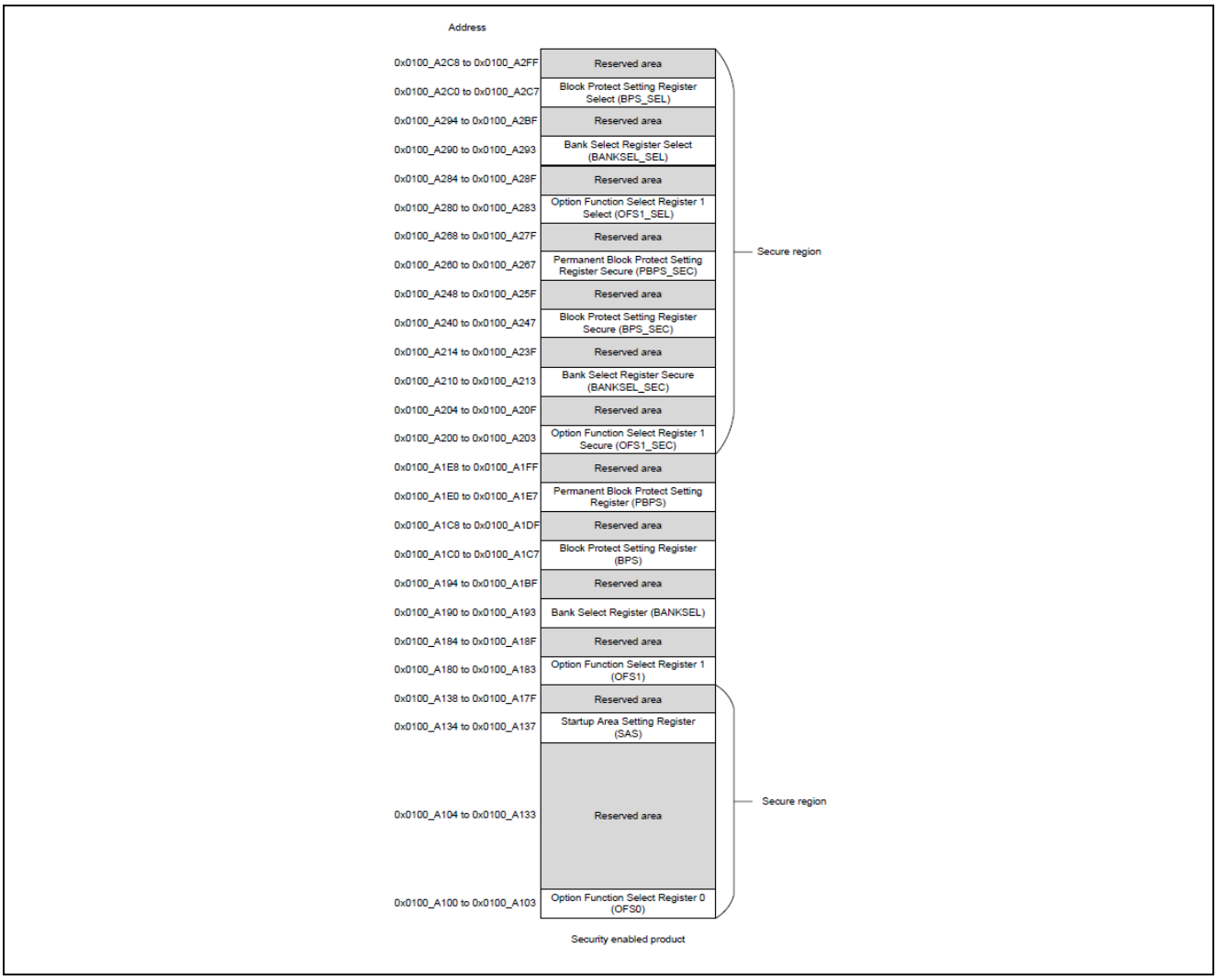


图 10. RA4M3 的选项功能选择寄存器

4.1 选项设置区寄存器

以下是选项设置存储器寄存器的概述。在启动之前，请确保已对其进行正确配置。

- OFS0 寄存器
  - 独立看门狗定时器 (IWDT) 自动启动
  - IWDT超时、频率、窗口操作、中断类型和低功耗模式行为
  - 看门狗定时器 (WDT) 自动启动
  - WDT超时、频率、窗口操作和中断类型
- OFS1 寄存器
  - 复位后 LVD0 使能
  - 复位后 HOCO 启动

Renesas FSP 配置器支持在 BSP 设置中设置选项存储器，如下图所示（以 RA4M3 MCU 为例）。通过 FSP 配置器进行的设置会反映在编译后的二进制文件中。

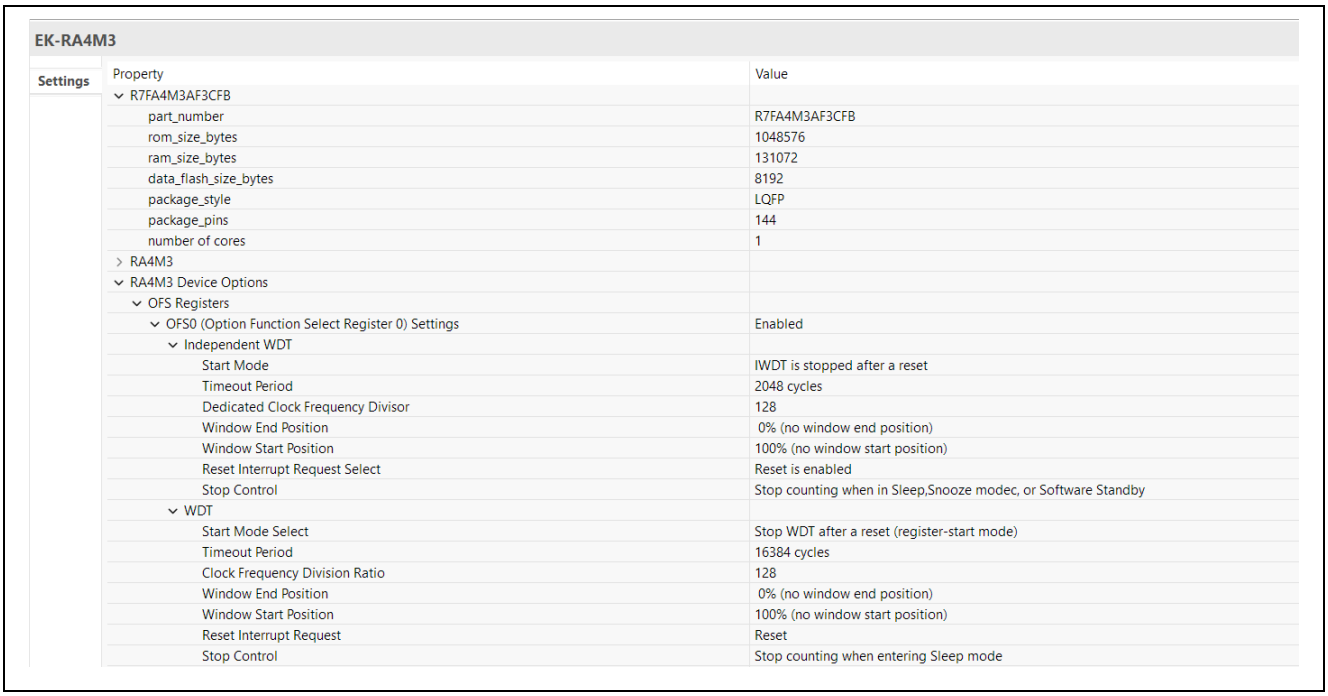


图 11. 在 FSP 中设置 RA4M3 MCU 的选项设置区

5 时钟电路

RA4 MCU 具有 6 个主要振荡器。其中有 5 个可以用作主系统时钟源。剩下的一个专用于独立看门狗定时器。在典型系统中，主时钟由外部晶体或时钟驱动。将此输入指向 PLL，并在此经过倍频后变为 PLL 时钟，然后分频为主系统时钟 (ICLK)、闪存时钟、外设模块时钟、外部总线时钟、跟踪时钟和 USB 时钟。此外，ICLK 还用于 CPU 时钟、DMAC 时钟和内部闪存时钟。对于某些 MCU，ICLK 也用于 SDRAM 时钟。有关时钟生成电路框图的信息，请参见硬件手册中 Clock Generation Circuit 一章。

每个时钟都有特定的容差和时序值。有关频率和时钟时序规格，请参见硬件手册中 Electrical Characteristics 一章的 AC Characteristics 部分。有关各种时钟频率之间关系的信息，请参见硬件手册中的 Clock Generation Circuit 一章。

表 7. RA4 振荡器

振荡源	输入源	频率	主要用途
主时钟	外部晶体/谐振器 -或- 外部时钟	8 MHz 至 24 MHz[注3,7] 最高 24 MHz[注10]	PLL输入、PLL2 输入[注5]、主系统时钟、CLKOUT、CAN 时钟、CAC 时钟、LCD 时钟[注10]、UART 时钟[注8]
副时钟 (SOSC)	外部晶体/谐振器	32.768 kHz	实时时钟、主系统时钟（低功耗模式）、CLKOUT、AGT 时钟、CAC 时钟、LCD时钟[注8]、UART 时钟[注8]、PLL_RTC 输入[注9]
高速片上振荡器 (HOCO)	片上振荡器	16/18/20 MHz[注8]	PLL 输入[注1]、主系统时钟、CLKOUT、CAC 时钟、LCD时钟[注2]、USB时钟[注8]、UART 时钟[注8]
中速片上振荡器 (MOCO)	片上振荡器	8 MHz	主系统时钟、CLKOUT、CAC 时钟、UART 时钟[注8]
低速片上振荡器 (LOCO)	片上振荡器	32.768 kHz	主系统时钟（低功耗模式下和主振荡器停止检测期间）、Systick 定时器[注1]、AGT 时钟、CLKOUT、CAC 时钟、实时时钟[注6]、UART 时钟[注8]
独立看门狗 (IWDT)	片上振荡器	15 kHz	独立看门狗定时器时钟、CAC 时钟

注：

1. Arm® Cortex®-M33 产品。
2. 在适用的 RA4 MCU 上。
3. RA4L1、RA4C1、RA4M1 和 RA4W1 的频率范围为 1 MHz 至 20 MHz。
4. Arm® Cortex®-M4 MCU 的 HOCO 频率选项为 24/32/48/64 MHz。RA4L1 和 RA4C1 的 HOCO 频率选项为 24/32/40/48/64/80 MHz。
5. 包含在 RA4M2、RA4M3 和 RA4E1 中。
6. RA4C1 的实时时钟不来源于 LOCO。
7. RA4W1 在低电压模式下的频率范围为 1 MHz 至 8 MHz。
8. RA4L1 和 RA4C1 具备此功能。
9. RA4C1 具备此功能。
10. RA4L1、RA4C1、RA4M1 和 RA4W1 的频率范围最高可达 20 MHz。

5.1 复位条件

复位后，RA4 MCU 开始运行，主时钟源为中速片上振荡器（MOCO）。在复位状态下，主振荡器和 PLL 默认关闭。此外，RA4L1 与 RA4C1 在复位时，副振荡器也默认关闭。HOCO 和 IWDT 的开启/关闭状态取决于选项设置区的设置（参见 3.1 章节）。

5.2 时钟频率要求

最小和最大时钟振荡频率如以下各表所示。有关振荡频率的详细信息以及外部和内部时钟源规格，请参阅 MCU 硬件手册中“Clock Generation Circuit”一章的“Overview”章节。更多详细信息请参阅硬件手册中“Electrical Characteristics”一章的“AC Characteristics”内容。

表 8. Arm Cortex-M4 MCU 内部时钟的频率范围

	ICLK	PCLKA	PCLKB	PCLKC	PCLKD
最大频率 [MHz]	48	48	32	64	64
最小频率 [MHz]	—	—	—	—	—

	FCLK [1]	UCLK	CANCLK	SLCDC
最大频率 [MHz]	32	48	20	64
最小频率 [MHz]	1	48	1	—

注 1：写入或擦除 ROM 或数据闪存时，FCLK 必须以至少 4 MHz 的频率运行。

表 9. Arm® Cortex®-M33 MCU 内部时钟的频率范围（RA4L1 及 RA4C1 除外）

	ICLK [1]	PCLKA [1]	PCLKB	PCLKC	PCLKD
最大频率 [MHz]	100	100	50	50	100
最小频率 [MHz]	—	—	—	—	—

	FCLK [1]	USBCLK	CANCLK
最大频率 [MHz]	50	48	24
最小频率 [MHz]	4	48	8

注 1：写入或擦除 ROM 或数据闪存时，FCLK 必须以至少 1 MHz 的频率运行。

表 10. RA4L1 及 RA4C1 MCU 内部时钟的频率范围

	ICLK	PCLKA	PCLKB	PCLKC	PCLKD
最大频率 [MHz]	80	80	40	48	80
最小频率 [MHz]	—	—	—	—	—

	FCLK [1]	USBCLK[2]	CANCLK
最大频率 [MHz]	48	48	20
最小频率 [MHz]	1	48	1

注 1：写入或擦除 ROM 或数据闪存时，FCLK 必须以至少 1 MHz 的频率运行。

注 2：RA4C1 不含此功能。

5.2.1 USB 通信要求

USB 2.0 全速模块（USBFS）需要 48 MHz 的 USB 时钟信号（UCLK 或 USBCLK）。

在 Arm Cortex-M33 器件上使用 USB，并且选择 HOCO 作为 PLL 的时钟源时，必须使能锁频环（FLL）功能。RA4L1、RA4C1 以及所有 Arm Cortex-M4 器件均不具备 FLL 功能。

使用 USB 外设时，主时钟振荡器频率仅限于特定的选择。这是由于时钟生成电路中可用的特定分频比，以及 USB 模块所需的 48 MHz 时钟所决定的。相关限制取决于关联寄存器中的倍频和分频设置。

对于 Arm Cortex-M4 器件，所使用的分频器取决于 PLLCCR2 寄存器（PLL 时钟控制寄存器 2）中控制位的设置。

对于 Arm Cortex-M33 产品，使用的分频器取决于 USBCKDIVCR 寄存器中 USBCKDIV 位的设置。USBCLK 也取决于 PLLCCR 和 PLL2CCR 寄存器中 PLL 的设置。RA4C1 不具备此功能。

有关乘除比的详细说明，请参见硬件手册中 Clock Generation Circuit 的内容。

5.2.2 ROM 或 数据闪存编程和擦除要求

对于 RA4L1 和 RA4C1，要对内部 ROM 和数据闪存执行编程和擦除操作，FCLK 必须至少为 1 MHz。

对于其他 RA4 器件，要对内部 ROM 和数据闪存执行编程和擦除操作，FCLK 必须至少为 4 MHz。

5.3 降低时钟生成电路（CGC）功耗

为了节省功耗，请尽可能将所有未使用的时钟（例如 BCLK）的分频器设置为最大可能值。另外，如果不使用时钟，请通过设置适当的寄存器来确保时钟已停止。下表为用于控制每个时钟源的寄存器。

表 11. 时钟源配置寄存器

振荡器	寄存器	说明
主时钟	MOSCCR	启动/停止主时钟振荡器
副时钟	SOSCCR	启动/停止副时钟振荡器
高速片上时钟（HOCO）	HOCOCR	启动/停止 HOCO
中速片上时钟（MOCO）	MOCOCR	启动/停止 MOCO
低速片上时钟（LOCO）	LOCOCR	启动/停止 LOCO

5.4 写入系统时钟控制寄存器

写入系统时钟分频控制寄存器（SCKDIVCR）和系统时钟源控制寄存器（SCKSCR）中的各个位字段时，应格外注意。

当外围模块时钟的时钟源切换为其他时钟源时，在时钟源切换期间，外围模块时钟周期会变长。详见图 12。因此必须在指令处理中增加延迟，以确保时钟在切换时钟源时仍保持稳定。

为保证时钟频率改变后的处理准确无误，首先写入相关的时钟控制寄存器改变频率，再从寄存器中读取值，最后进行后续处理。

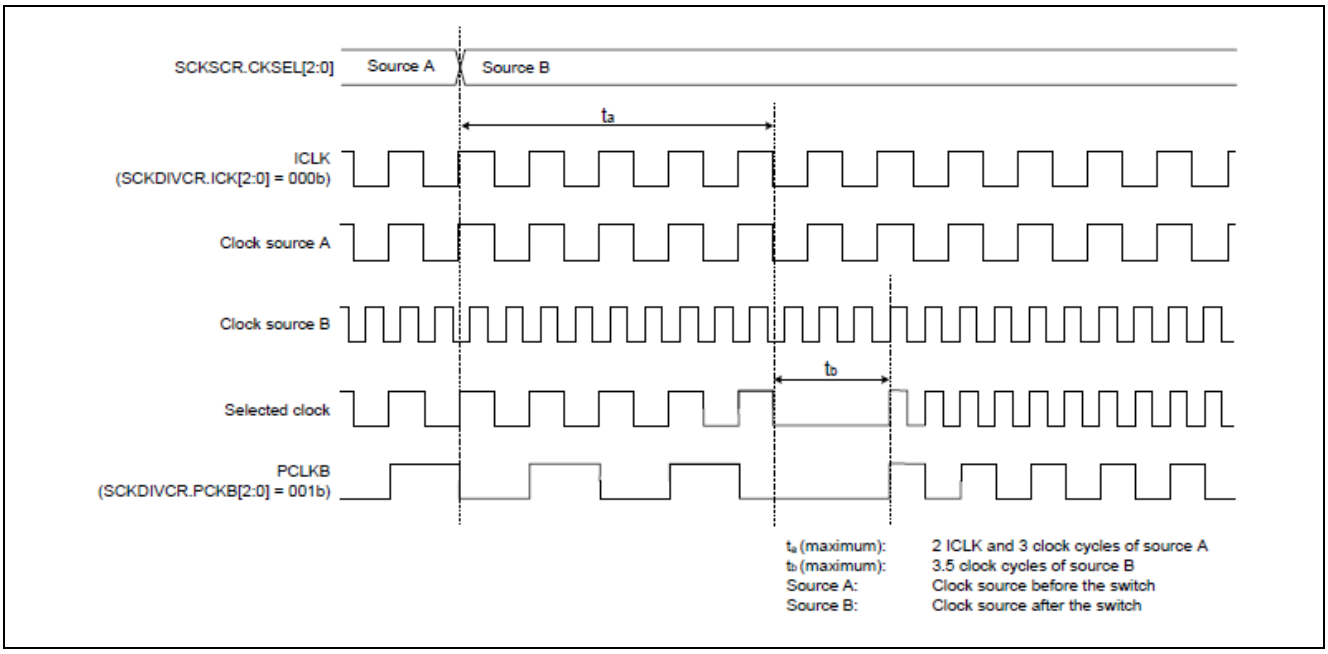


图 12. 切换时钟源的时序

5.5 时钟设置示例

瑞萨 FSP 为 RA MCU 提供了简单、可视化的时钟配置工具，以 RA4M3 为例，如下所示。

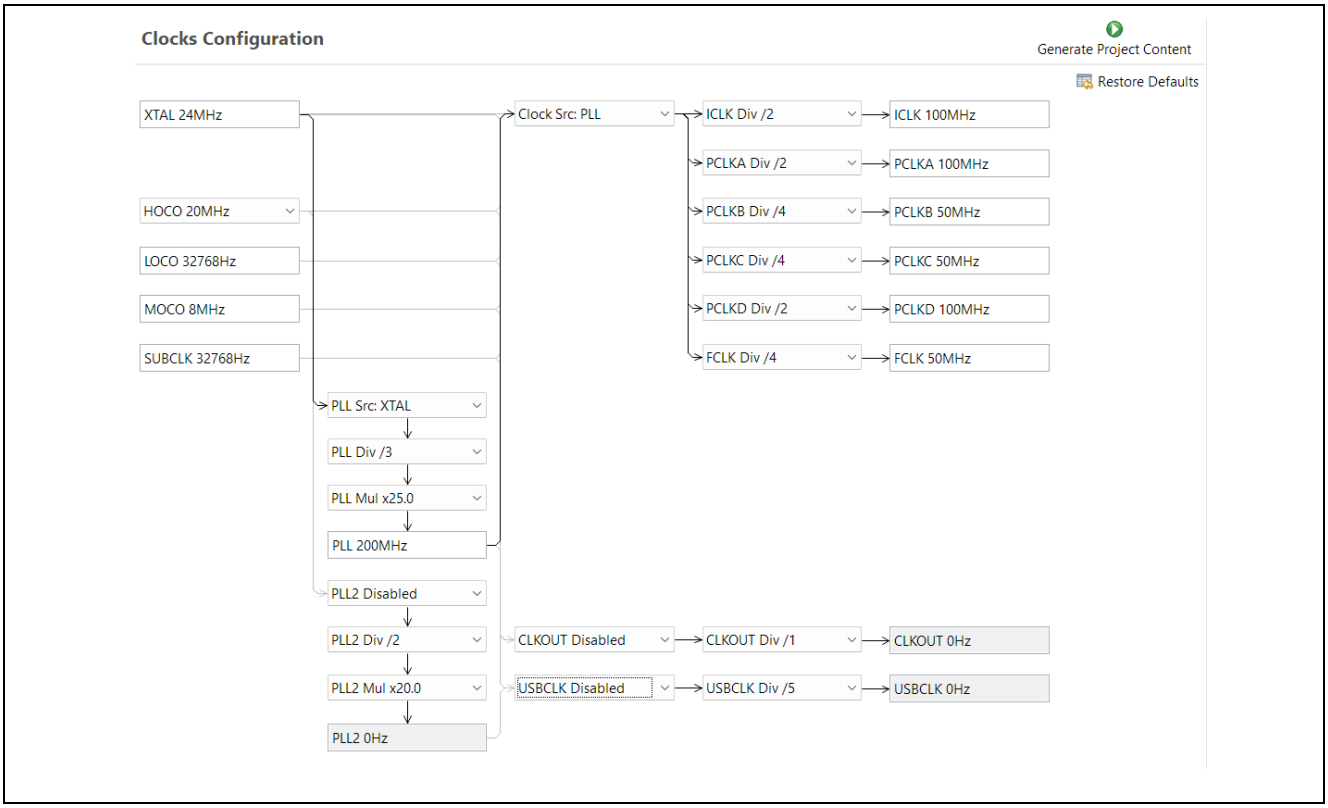


图 13. 使用 Renesas FSP 配置器进行时钟设置

5.6 HOCO 精度

内部高速片上振荡器（HOCO）的典型精度为 ±2% 或更高。针对各种环境工作温度（Ta）范围 HOCO 精度规格进行了特性规定。

对于大多数 Arm Cortex-M33 器件，可通过使能锁频环（FLL）功能来提高 HOCO 的精度，从而使时钟精度达到±0.25% 或更高。有关详细信息，请参见硬件手册中 Electrical Specifications 的内容。RA4L1 和 RA4C1 器件不具备此功能。

HOCO 可用作 PLL 电路的输入。当以这种方式使用 HOCO 时，不需要外部振荡器。如因空间限制或其他限制需要减少 PCB 设计中的元器件数量时，这便是一个优势。当然，要同时考虑时钟精度、性能权衡方面的限制，需要针对客户的应用进行评估。

5.7 闪存接口时钟

内部闪存（ROM 和 DF）的编程和擦除操作以及从数据闪存读取数据时，闪存接口时钟（FCLK）用作工作时钟。因此，FCLK 的频率设置会直接影响从数据闪存读取数据所需的时间。如果用户程序需要从数据闪存读取数据，或者需要对内部闪存执行编程或擦除操作，则建议使用最高 FCLK 频率。

当写入或擦除代码闪存（ROM）或数据闪存时，FCLK 必须以至少 4 MHz（RA4L1 及 RA4C1 为 1 MHz）的频率运行。请注意，FCLK 频率对从 ROM 读取数据或对 RAM 进行读写操作没有任何影响。

5.8 电路板设计

有关使用 CGC 的更多信息和电路板设计建议，请参阅硬件手册中 Clock Generation Circuit (CGC) 章节的 Usage Notes 部分。

通常而言，晶振及其负载电容应尽可能靠近 MCU 时钟引脚（XTAL/EXTAL、XCIN/XCOUT）。应避免在晶振和 MCU 之间出现任何其他信号。尽量减少每条走线上使用的连接通孔数量。

5.9 外部晶振选择

外部晶振可用作主时钟源。外部晶振连接在 MCU 的 EXTAL 和 XTAL 引脚上。其频率必须在主时钟振荡器的频率范围内。

晶振的选择很大程度上取决于具体的电路板设计。由于可用的晶振种类繁多，为 RA4 MCU 选用晶振时请仔细评估电气特性，以确定具体要求。

典型的晶振连接示例如下所示。

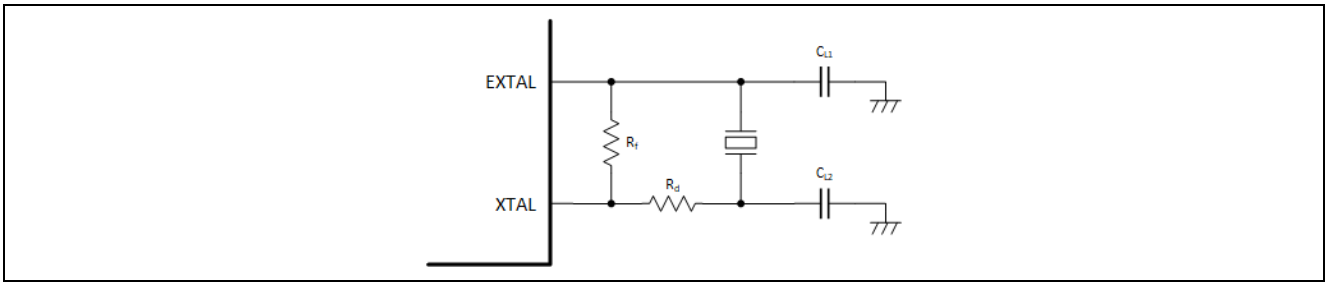


图 14. 晶振连接示例

在选择晶振和相关电容时，须进行详细评估。外部反馈电阻（Rf）和阻尼电阻（Rd）可根据晶振制造商的建议添加。

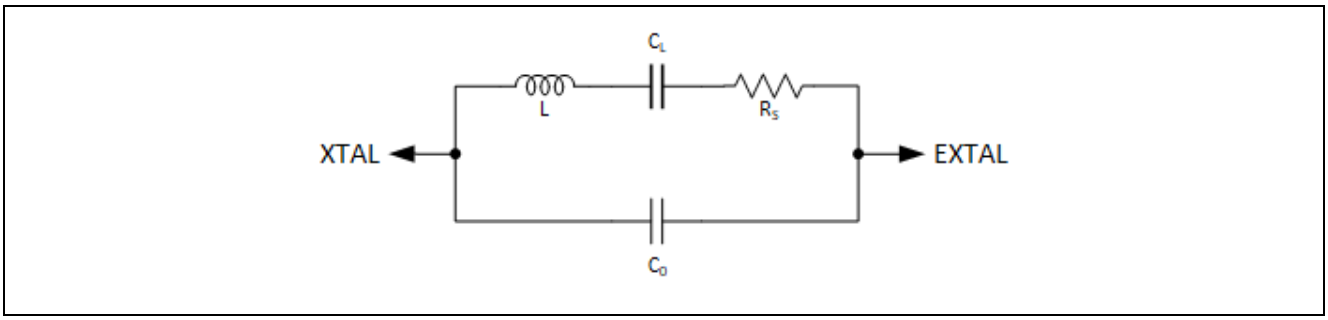


图 15. 晶振的等效电路

CL1 和 CL2 的电容值选择会影响内部时钟的精度。使用上图中晶振的等效电路对电路进行仿真，可了解 CL1 和 CL2 值的影响。如需获得更准确的结果，还要考虑与晶振元件之间的布线相关的杂散电容。

5.10 外部时钟输入

数字时钟输入可以用作主时钟源。图 16 为连接外部时钟输入的示例。若使用外部时钟信号运行振荡器，请将 MOMCR.MOSEL 位设为 1。XTAL 引脚变为高阻抗。

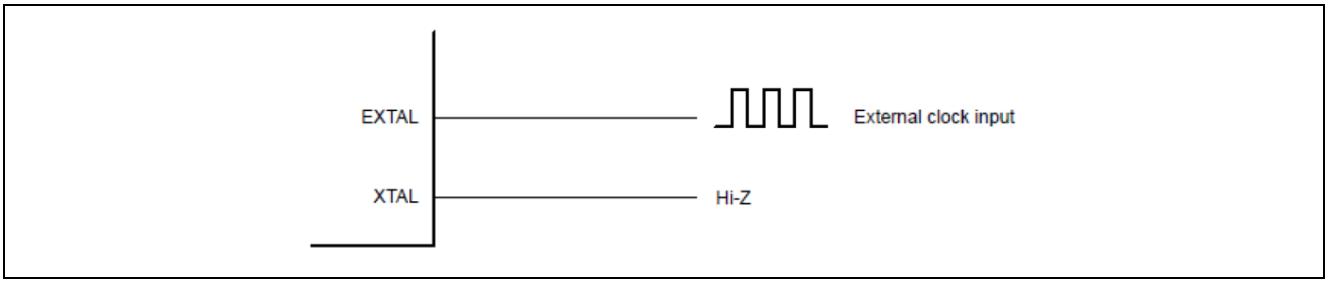


图 16. 外部时钟的等效电路

注：只有在主时钟振荡器停止运行时，才能改变外部时钟输入的频率。请不要在主时钟振荡器停止位（ MOSCCR.MOSTP 位 ）为 0 时更改外部时钟输入的频率。

6 复位要求和复位电路

Arm® Cortex®-M4 MCU 共有 14 种类型的复位。

表 12. Arm Cortex-M4 MCU 的复位

复位名称	源
引脚复位	RES# 引脚被拉低
上电复位	VCC 上升（电压检测：V_{POR}）
VBATT电压上电复位	VCC 下降（电压检测：$V_{DETBATT}$）
独立看门狗定时器复位	独立看门狗定时器下溢，或未执行刷新操作
看门狗定时器复位	看门狗定时器下溢，或未执行刷新操作
电压监视器 0 复位	VCC 下降（电压检测 V_{det0}）
电压监视器 1 复位	VCC 下降（电压检测 V_{det1}）
电压监视器 2 复位	VCC 下降（电压检测 V_{det2}）
SRAM 奇偶校验错误复位	检测到 SRAM 奇偶校验错误
SRAM ECC 错误复位	检测到 SRAM ECC 错误
总线主 MPU 错误复位	检测到总线主 MPU 错误
总线从 MPU 错误复位	检测到总线从 MPU 错误
堆栈指针错误复位	检测到堆栈指针错误
软件复位	寄存器设置

Arm® Cortex®-M33 MCU 的复位类型会因产品不同略有不同，共有 13 或 14 种类型的复位。

表 13. Arm® Cortex-M33 MCU 的复位

复位名称	源
引脚复位	RES# 引脚被拉低
上电复位	VCC 上升（电压检测：V_{POR}）
VRTC 域上电复位[注3]	VRTC 下降（电压检测：V_{VRTC_POR}）
独立看门狗定时器复位	独立看门狗定时器下溢，或未执行刷新操作
看门狗定时器复位	看门狗定时器下溢，或未执行刷新操作
电压监视器 0 复位	VCC 下降（电压检测 V_{det0}）
电压监视器 1 复位	VCC 下降（电压检测 V_{det1}）
电压监视器 2 复位	VCC 下降（电压检测 V_{det2}）
SRAM 奇偶校验错误复位	检测到 SRAM 奇偶校验错误
SRAM ECC 错误复位	检测到 SRAM ECC 错误
总线主 MPU 错误复位	检测到总线主 MPU 错误
TrustZone 错误复位	检测到 TrustZone 错误
缓存奇偶校验错误复位[注1]	检测到缓存奇偶校验错误
深度软件待机复位[注2]	通过中断信号取消深度软件待机模式
软件复位	寄存器设置

注：

1. 仅 RA4M3 及 RA4E2 具备此复位。
2. RA4L1 和 RA4C1 不具备深度软件待机（Deep Software Standby）复位。
3. 仅 RA4C1 具备此复位。

6.1 引脚复位

当 RES# 引脚被拉低时，所有处理都将中止，MCU 进入复位状态。要在运行中复位 MCU，应在指定的复位脉冲宽度内将 RES# 保持为低电平。有关时序要求的更详细信息，请参见硬件手册中 “Electrical Characteristics”一章的“Reset Timing”部分。另请参见本文档的第 2 节“仿真器支持”，了解与调试支持相关的复位电路的详细信息。

无需在 RES# 线路上使用外部电容，因为 POR 电路在内部将其保持为低电平以实现良好的复位，并且需要最小的复位脉冲来启动此过程。

6.2 上电复位

有两种情况会产生上电复位（POR）：

1. 上电时 RES# 引脚处于高电平状态。
2. VCC 降到 V_{POR} 以下时，RES# 引脚处于高电平状态。

当 VCC 超过上电复位电压（V_{POR}），且上电复位时间（t_{POR}）过后，芯片退出上电复位状态。上电复位时间是确保外部电源和 MCU 稳定工作而设置的时间段。请参阅硬件手册的“Electrical Characteristics”一章的“POR and PVD Characteristics”，了解电压电平和时序方面的详细信息。

由于 POR 电路要满足 RES# 与 VCC 同时为高电平，因此当仅使用上电复位功能时，不要在复位引脚上加电容。否则将延缓 RES# 相对于 VCC 的上升时间，从而导致 POR 电路无法正确识别上电条件。

当电源（VCC）降至 V_{POR} 或更低时，如果 RES# 引脚为高电平，则会产生上电复位。当 VCC 上升到 V_{POR} 以上并且 t_{POR} 时间过后，芯片退出上电复位状态。

上电复位后，RSTSR0 的 PORF 位被置为“1”。引脚复位后，PORF 位会被清 0。

6.3 VRTC 域上电复位

VRTC 域上电复位是由 VRTC 域的上电复位电路生成的内部复位。当 VRTC 超过 V_{RTC_POR}，并且经过规定时间（t_{RTCPOR}）后，将解除内部复位。当 VRTC 下降至 V_{RTC_POR} 以下时，VRTSR.VRTVLD 标志被清零，同时 VRTSR.PORDET 标志再次置位。

6.4 VBATT 电压上电复位

当 VCC 引脚电压下降时，由 VBATT 引脚向 RTC、LOCO 和副时钟振荡器供电。当 VCC 引脚处检测到电压下降时，电源连接将从 VCC 引脚切换到 VBATT 引脚上。

Arm Cortex-M4 产品发生这种情况时，会产生复位以标记从 VCC 到 VBATT 的切换。有关该复位的详情以及相关的 VBATT 配置，请参考硬件手册中 “Battery Backup Function” 的内容。该复位功能在 Arm Cortex-M33 产品中不存在。

6.5 独立看门狗定时器复位

该复位为独立看门狗定时器 (IWDT) 产生的内部复位。

当 IWDT 下溢时，可以选择产生独立看门狗定时器复位（可以改为产生 NMI），RSTSR1 中的 IWDTRF 位置为“1”。短暂延迟后，将取消 IWDT 复位。详情请参照硬件手册。

6.6 看门狗定时器复位

这是看门狗定时器 (WDT) 产生的内部复位。

当 WDT 下溢时，可以选择产生看门狗定时器复位（可以改为产生 NMI），RSTSR1 的 WDTRF 位置为“1”。短暂延迟后，将取消 WDT 复位。详情请参照硬件手册。

6.7 电压监视器复位

RA4 系列具备允许 MCU 在欠压期间防止不安全操作的电路。板上比较器根据三个参考电压 V_{det0}、V_{det1} 和 V_{det2} 检查电源电压。当电源下降到每个参考电压以下时，会产生中断或复位。检测电压 V_{det0}、V_{det1} 和 V_{det2} 均可从 3 个不同大小的值中选择。

当 Vcc 随后上升到超过 V_{det0}、V_{det1} 或 V_{det2} 时，经过稳定时间后，电压监视复位将被释放。

上电复位后，低电压检测功能默认被禁用。可以通过使用选项功能寄存器 OFS1 来使能电压监视。有关更多详细信息，请参见硬件手册中“Low Voltage Detection (LVD)”的内容。

发生 LVD 复位后，RSTSR0 中的 LVDnRF（n = 0、1、2）位置为“1”。

6.8 深度软件待机复位

这是在通过中断信号取消深度软件待机模式时产生的内部复位。

当取消深度软件待机模式时，将产生深度软件待机复位，并且时钟开始振荡。接收到中断后，经过深度待机取消等待时间（t_{DSBYWT} 34 ~ 35 个时钟周期）后，将取消复位并开始正常处理。有关深度软件待机模式的详细信息，请参见硬件手册中“Low Voltage Detection”的内容。

深度软件待机复位后，RSTSR0 中的 DPSRSTF 位置为“1”。

6.9 软件复位

该复位是通过将 SYSRESETREQ 位写入 Arm 内核的 AIRCR 寄存器产生的内部复位。当 SYSRESETREQ 位设为 1 时，产生软件复位，再经过内部复位时间 (t_{RESW2}) 后，将取消内部复位，CPU 进行复位异常处理。详情请参照 MCU 硬件手册。

有关 SYSRESETREQ 位的详细信息，请参照 Arm® Cortex®-M33 和 Cortex®-M4 的技术手册。

6.10 其他复位

MCU 内部的大多数外设功能，在特定故障条件下都可以产生复位。启用这些复位功能不需要进行硬件配置。有关每个外设功能产生复位的具体条件，请参阅硬件用户手册中对应章节的详细说明。

6.11 冷/热启动的确认

RA4 MCU 允许用户判断复位处理的原因。RSTSR2 中的 CWSF 标志位用于指示是上电复位导致了复位过程（冷启动），还是运行过程中输入的复位信号导致了复位过程（热启动）。

发生上电复位时，该标志置为“0”。否则，该标志不会置为“0”。通过软件向该标志写入 1 时会将其置为“1”。即使向该位写入“0”，也不会将其置“0”。

6.12 确定复位源

RA4 MCU 可由用户确定复位信号产生源。读取 RSTSR0 和 RSTSR1，可以判断是哪一种复位触发了当前复位。有关流程图，详情请参照硬件手册中的“Determination of Reset Generation Source”的内容。

以下代码示例展示了如何使用 Renesas FSP 中基于 CMSIS 的寄存器结构确定复位是由软件复位、深度软件待机还是上电复位导致的。

```
// Reset corresponding to each bit of RSTSR1 or RSTSR0.LVD1RF or RSTSR0.LVD2RF
if ( (0x0 != R_SYSTEM->RSTSR1) ||
     (0x1 == R_SYSTEM->RSTSR0_b.LVD1RF) ||
     (0x1 == R_SYSTEM->RSTSR0_b.LVD2RF) )
{
        // When resets are generated simultaneously by two or more sources
        // the two or more corresponding reset flags will be set to 1.
       if (0x1 == R_SYSTEM->RSTSR1_b.SWRF) {
           // Software reset
       } else {
          // Check other reset flags
       }
} else if (0x1 == R_SYSTEM->RSTSR0_b.DPSRSTF) {
   // Deep software standby reset
   // Some RA4 MCUs do not have this register.
} else if (0x1 == R_SYSTEM->RSTSR0_b.PORF) {
   // Power-on reset
} else {
   // Check other reset flags
}
```

7 TrustZone®支持

7.1 Arm® TrustZone 技术的实施

某些 RA4 MCU（如 RA4M2 和 RA4M3）具备 Arm® TrustZone® (TZ) 安全功能。关于实施 TZ 的详细信息，请参见 Arm 文档 (https://developer.arm.com/ip-products/security-ip/trustzone) 或相应的硬件手册。

Arm TZ 技术将 MCU 划分为安全和非安全分区，因此应用程序也分为安全和非安全应用程序。安全应用程序可以访问安全和非安全存储区和资源。非安全代码可以通过位于非安全可调用 (NSC) 区域中的一组所谓的模板来访问非安全存储器和资源以及安全资源。当从非安全分区中调用时，这可确保从单个访问点获取安全代码。默认情况下，MCU 在安全分区中启动。CPU 的安全状态可以是“安全”或“非安全”。

FSP 对 MCU 进行配置使其在复位时启动安全应用程序。这里的 CPU 安全状态既可以是“安全”，也可以是“非安全”。MCU 代码闪存、数据闪存和 SRAM 分为安全 (S) 和非安全 (NS) 区域。代码闪存和 SRAM 还包含一个称为非安全可调用 (NSC) 的区域。当产品生命周期为安全软件调试 (SSD) 状态时，可通过 SCI 或 USB boot 模式命令将这些存储器安全属性设置到非易失性存储器中。在应用程序执行之前，将存储器安全属性加载到实现定义属性单元 (IDAU) 外设和存储器控制器中，并且无法通过应用程序代码进行更新。

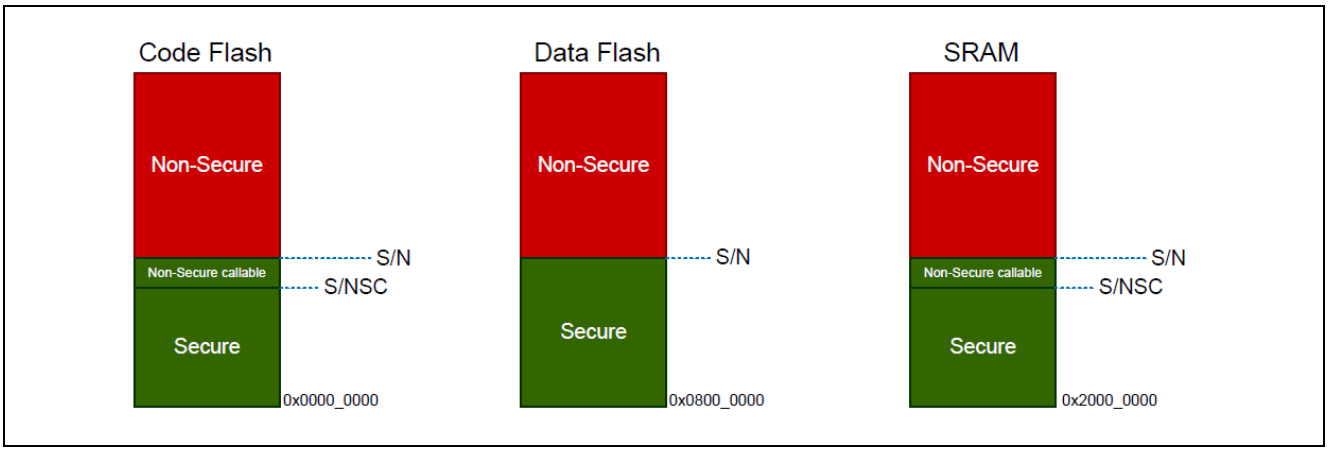


图 17. 安全（Secure）区和非安全（Non-Secure）区

注：所有外部存储器访问均被视为非安全访问。

代码闪存和 SRAM 可以分为安全、非安全和非安全可调用三个区域。来自非安全区域的所有安全存储器访问均必须通过非安全可调用网关，并以特定的安全网关 (SG) 汇编器指令为目标。这将强制访问固定位置的安全 API，并防止调用子功能等。未以 SG 指令为目标将产生 TZ 异常。支持 TZ 的编译器将使用 CMSE 扩展自动管理 NSC 模板的产生。这些安全、非安全和非安全可调用的区域均可以通过 FSP 轻松配置和管理。

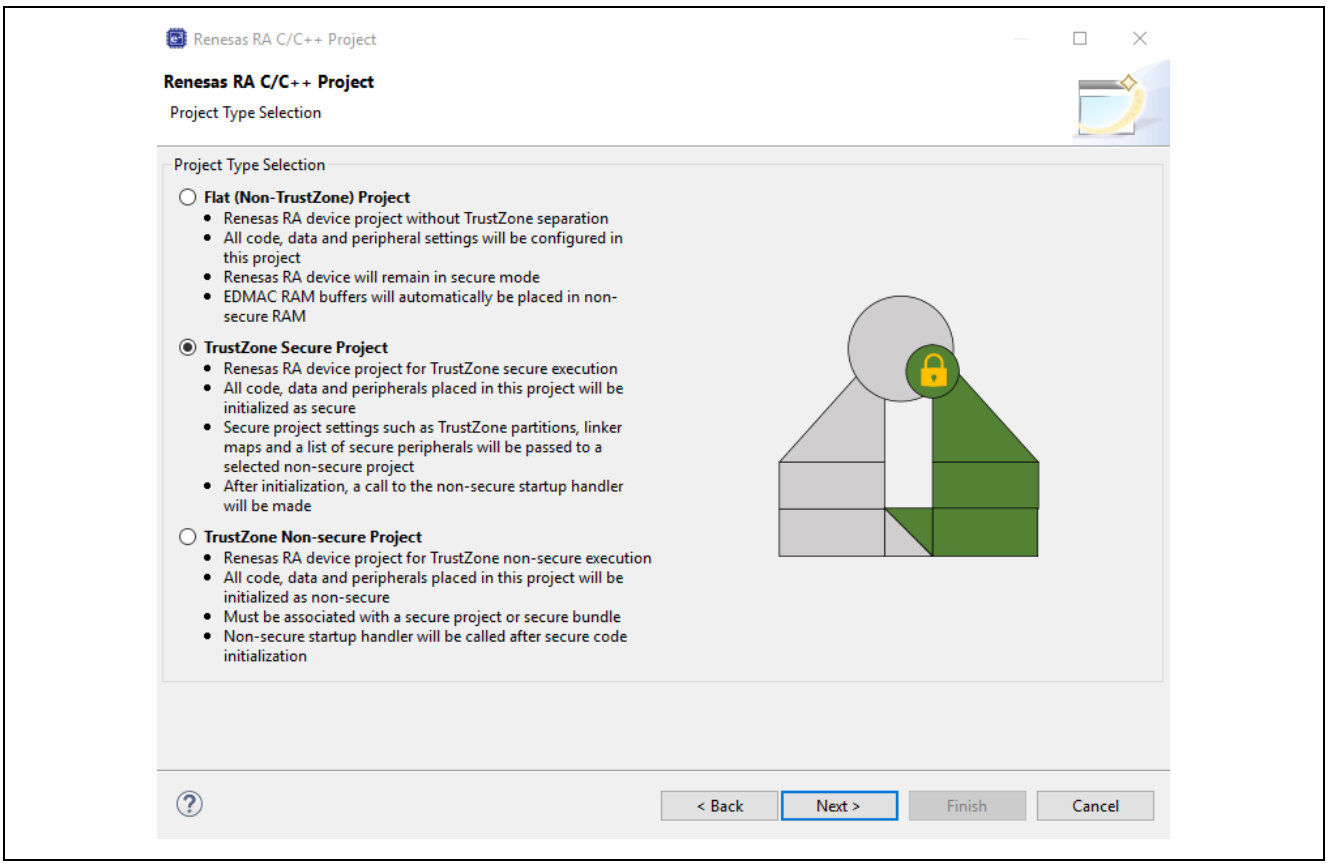


图 18. 通过 RA Project Generator 新建 TZ 使能工程

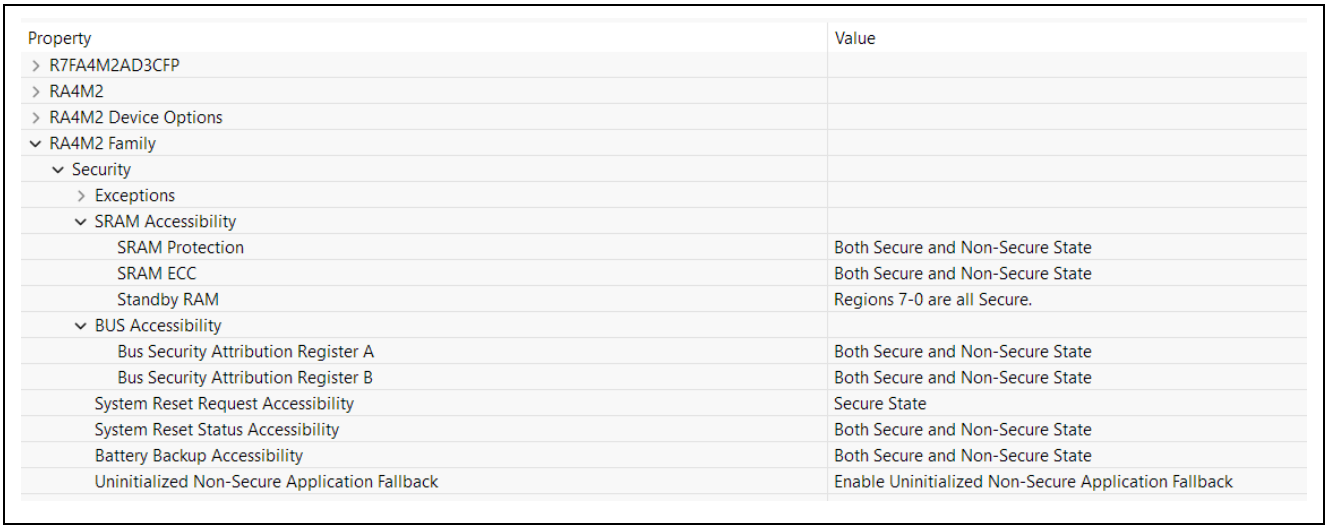


图 19. FSP Configurator 中安全区和非安全区的设置

7.2 支持 TrustZone 的仿真器

Renesas 提供的仿真器既支持通过 SWD 或 JTAG 通信进行调试，也支持通过 SCI 通信进行串行编程。借助该仿真器可以轻松地在调试和串行编程之间进行切换。下表为使用该仿真器时 10 引脚或 20 引脚插座的引脚分配。SWD 和 JTAG 的引脚分配符合 Arm® 标准，并增加了 MD、TXD 和 RXD 引脚，用于通过 SCI 通信进行串行编程。

除 RA4E2 外，必须使用串行编程接口对 Arm® TrustZone IDAU 边界寄存器设置进行编程。为了同时使用调试和串行编程，建议在电路板上使用“线或”电路连接 P300/SWCLK/TCK 和 P201/MD 引脚。

对于 RA4E2 MCU，IDAU 边界寄存器在运行时可以通过应用程序进行设置。更多信息，请参见 Security Design with Arm® TrustZone® - IP Protection 文档（R11AN0467EU）。

表 14. 仿真器引脚分配

引脚号	SWD	JTAG	通过 SCI 进行串行编程
1	VCC	VCC	VCC
2	P108/SWDIO	P108/SWDIO	NC
4	P300/SWCLK 以“线或”方式连接 P201/MD	P300/TCK 以“线或”方式连接 P201/MD	P201/MD
6	P109/SWO/TXD9	P109/TDO/TXD9	P109/TDO/TXD9
8	P110/SWO/RXD9	P110/TDI/RXD9	P110/TDI/RXD9
9	GNDdetect	GNDdetect	GNDdetect
10	nRESET	nRESET	nRESET
12	P214/TCLK	P214/TCLK	NC
14	P211/TDATA[0]	P211/TDATA[0]	NC
16	P210/TDATA[1]	P210/TDATA[1]	NC
18	P209/TDATA[2]	P209/TDATA[2]	NC
20	P208/TDATA[3]	P208/TDATA[3]	NC
3,5,15,17,19	GND	GND	GND
7	NC	NC	NC
11,13	NC	NC	NC

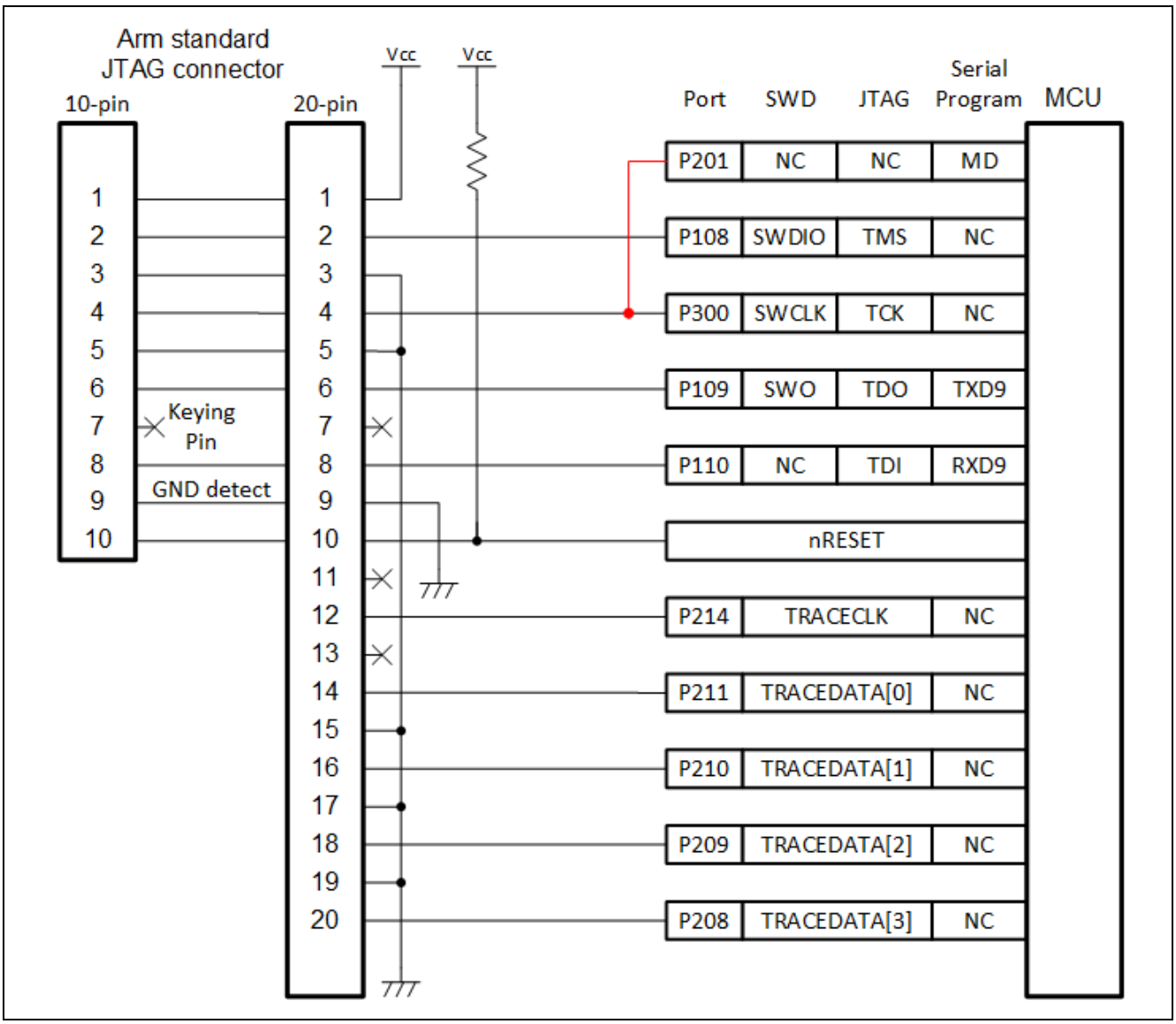


图 20. 支持 TrustZone® 的 MCU 的仿真器连接

7.2.1 设备生命周期管理

搭载了 Arm® Cortex®-M33 内核的 RA4 MCU 配备生命周期管理 (DLM) 功能，能够对产品从开发阶段到生产再到生命周期结束的整个过程进行管理。RA 系列 MCU 调试功能和串行编程功能由产品生命周期状态定义。

有关更多详细信息，请参见《*产品生命周期管理密钥安装*》 应用笔记。

7.2.2 ID 验证

RA4E2 MCU 通过 ID 验证来禁止编程和片上调试。根据存储在闪存中的 ID 代码，对 ID 代码进行验证或使其无效，并确定 ID 代码。当使能 ID 代码保护时，会将主机发送的 ID 代码与闪存中的 ID 代码进行比较，以判断二者是否匹配。只有当二者匹配时，才允许进行编程和片上调试。闪存中的 ID 代码由四个 32 位字组成。

ID 代码通过 OSIS（OCD/Serial Programmer ID Setting）寄存器进行设置。

8 存储器

RA4 MCU 支持 4 GB 的线性地址空间，范围为 0000 0000h 到 FFFF FFFFh，其中可以包含程序、数据和外部存储器总线。程序和数据存储器共用同一个地址空间；可使用单独的总线分别访问这两个存储器，从而提高性能并允许在同一个周期访问程序和数据。存储器映射中包含片上 RAM、外设 I/O 寄存器、程序代码闪存、数据闪存。

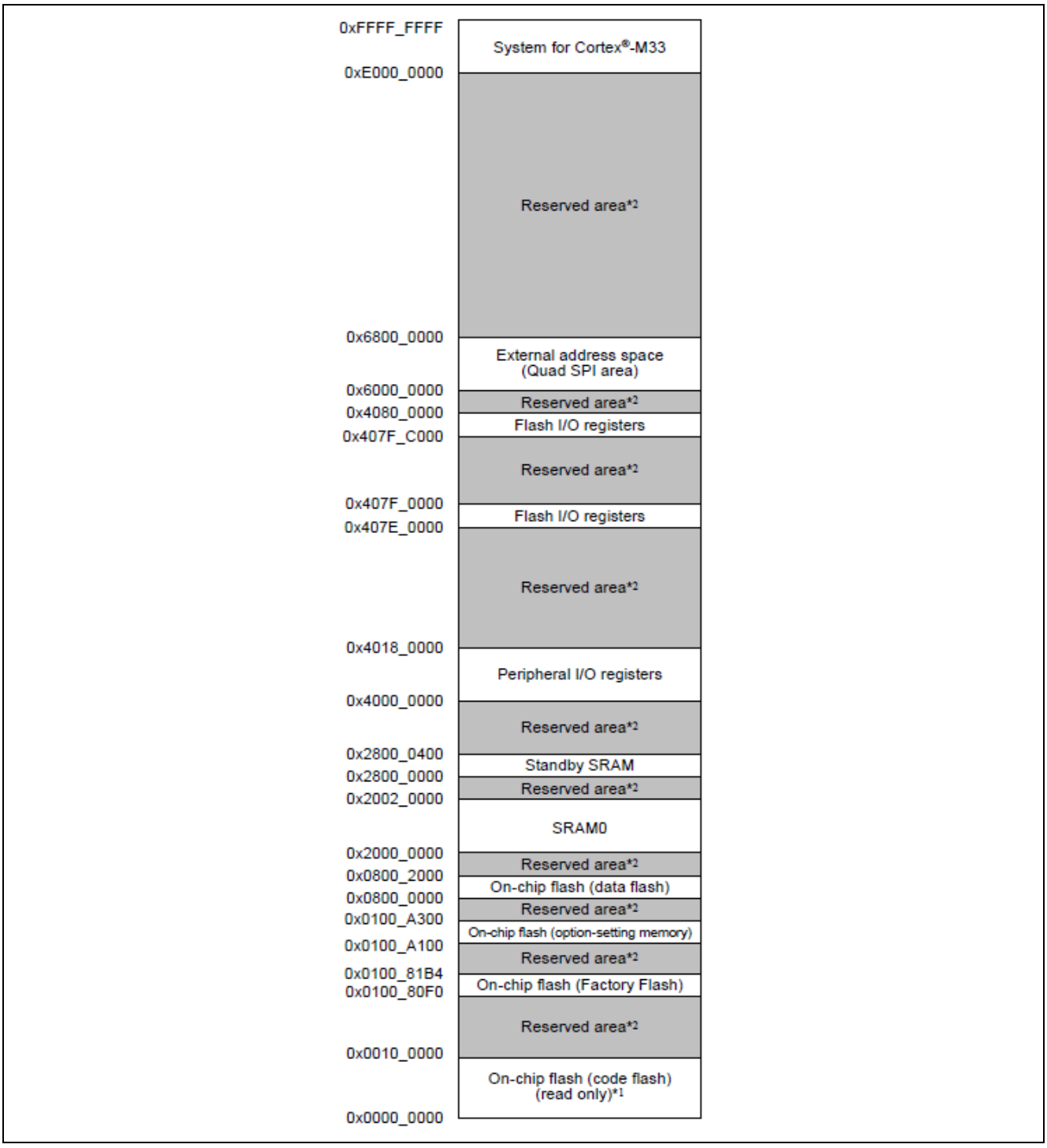


图 21. RA4M3 存储器映射

8.1 SRAM

RA4 MCU 提供带奇偶校验位或 ECC（纠错码）的片上高速 SRAM 模块。SRAM0 的前 32 KB 区域支持 ECC 。其他区域执行奇偶校验检查。下表列出了 SRAM 的规格。SRAM 模块的数量和容量因产品而异。有关详细信息，请参阅 MCU 用户手册中的 CPU 章节。

Table 42.1 SRAM specifications

Parameter	Without ECC	With ECC
SRAM capacity	SRAM0: 64 KB	SRAM0: 64 KB
SRAM address	SRAM0: 0x2001_0000 to 0x2001_FFFF	SRAM0: 0x2000_0000 to 0x2000_FFFF
Access	Can access with no wait. One wait access is set at initial state. For details, see section 42.3.9. Access Cycle	
Data retention function	Not available in deep standby mode	
Module-stop function	Module-stop state can be set to reduce power consumption	
Parity	Even parity with 8-bit data and 1-bit parity	No parity
Error checking	even-parity（Date:8bit, parity:1bit）	SEC-DED (Single-Error Correction and Double-Error Detection Code)
Security	TrustZone Filter is integrated for memory access and SFR access. Access to the memory space is controlled by setting the memory Security Attribution (SA). And, access to I/O space (SFR) space is controlled by setting the register SA. See section 42.3.6. TrustZone Filter function.	

图 22. RA4M3 SRAM 规格示例

8.2 待机 SRAM

RA4 MCU 的片上 SRAM，用于在深度软件待机模式下保持数据。下表为待机 SRAM 的规格。

在深度软件待机模式下，通过 DPSBYCR.DEEPCUT[1:0] 位控制是否向待机 SRAM 供电。如果 DPSBYCR.DEEPCUT[1:0] 位设置为 00b，则在深度软件待机模式下，待机 SRAM 中的数据会被保持。有关 DPSBYCR.DEEPCUT[1:0] 位的详细信息，请参见第 11 节“Low Power Modes”。

对于不支持深度软件待机模式的 MCU（例如 RA4L1 和 RA4C1），不存在待机 SRAM。

Table 43.1 Standby SRAM specifications

Item	Description
SRAM capacity	1 KB
SRAM address	0x2800_0000 to 0x2800_03FF
Access	Standby SRAM clock is the same clock as the PCLKB. See section 43.3.5. Access Cycle for details.
Data retention function	Data can be retained in deep standby mode. See section 43.3.1. Data Retention for details.
parity	Even parity (data: 8 bits, parity: 1 bit)
Module-stop function	Module-stop state can be set to reduce power consumption. See section 43.3.2. Setting for the Module-stop Function for details.
Security	Permits the read and write operations to Standby RAM following TrustZone Filter function. See section 43.3.4. TrustZone Filter function for details.

图 23. RA4M3 待机 SRAM 的规格

Renesas FSP 中的 LPM（低功耗模式）驱动程序提供了一个用于切断或保持待机 SRAM 电源的选项，如下图所示。仍然需要调用 LPM 驱动程序的 API 才能将所选设置写入 MCU 寄存器。

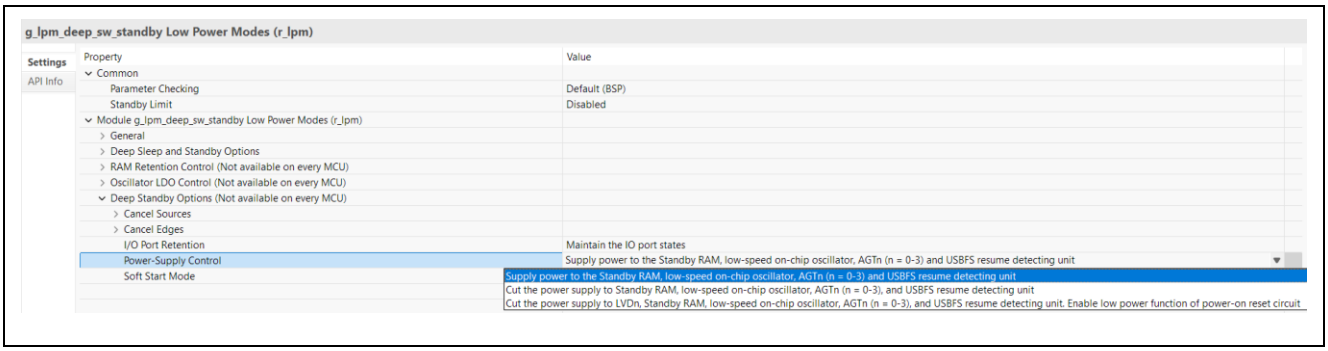


图 24. 通过 Renesas FSP Configurator 启用/禁用待机 SRAM 的电源

8.3 外设 I/O 寄存器

外设 I/O 寄存器块出现在存储器映射中的不同位置，具体取决于设备和当前的工作模式。大多数外设 I/O 寄存器占用的地址范围为 4000 0000h 到 400F FFFFh。但是，位置和大小可能会因产品而异。有关详细信息，请参见硬件手册。有关详细信息，可参见“I/O 寄存器”附录，也可参见每个外设功能的寄存器说明。该区域包含在所有工作模式下始终可用的寄存器。用于控制访问闪存的闪存 I/O 寄存器占据两个区域，即 407E 0000h 至 407E FFFFh 和 407F C000h 至 407F FFFFh。

Renesas FSP 包含采用 CMSIS 数据结构的 C 头文件，这些文件将特定器件的所有外设 I/O 寄存器映射到易于访问的 I/O 数据结构。

8.4 片上闪存

RA4 MCU 具有两个闪存部分：代码闪存和数据闪存，各部分的大小和可编程擦写次数因产品而异。闪存控制单元 (FCU) 控制闪存的编程和擦除。闪存应用程序命令接口 (FACI) 根据指定的 FACI 命令控制 FCU。

代码闪存旨在存储用户应用程序代码和常量数据。数据闪存旨在存储可能不时更新的信息，例如配置参数、用户设置或记录的数据。数据闪存区域中的编程和擦除单位远小于代码闪存的单位（数据闪存为 4 / 8 /16 字节，而代码闪存为 128 字节）。

数据闪存和代码闪存区域均可通过应用程序代码进行编程或擦除。这样，无需连接外部编程工具即可完成现场固件更新。

Renesas FSP 为代码闪存和数据闪存提供了 HAL 层驱动程序。

下图为代码闪存和数据闪存的规格。

Table 44.1 Specifications of flash memory (1 of 2)

Item	Code flash memory	Data flash memory
Memory capacity	User area: 1 Mbytes max	Data area: 8 Kbytes
Read cycle	See section 44.16.3. Access Cycle	See section 44.16.3. Access Cycle
Value after erasure	0xFF	Undefined
Programming/erasing method	• Programming and erasing the code flash memory and data flash memory, and programming the option-setting memory are handled by the FACI commands specified in the FACI command issuing area (0x407E_0000) (self-programming). • Programming/erasure through transfer by a serial-programmer via a serial interface (serial programming)	
Protection	Protects against erroneous rewriting of the flash memory	
Block swap function	The block swap structure makes a safe update for a part of non-secure application possible in case where programming is suspended.	Not available
Background operations (BGOs)	• The data flash memory can be read while the code flash memory is being programmed or erased. • The code flash memory can be read while the data flash memory is being programmed or erased.	
Units of programming and erasure	• Units of programming for the user area: 128 bytes • Units of erasure for the user area: Block units	• Unit of programming for the data area:4/8/16 bytes • Unit of erasure for the data area:64/128/256 bytes
Other functions	Interrupts can be accepted during self-programming. In the initial settings of this MCU, an expansion area of the option-setting memory can be set.	
On-board programming (four types)	Programming/erasure in boot mode (for the SCI interface) • The asynchronous serial interface (SCI9) is used. • The transfer rate is adjusted automatically. Programming/erasure in boot mode (for the USB interface) • USBFS is used. • Dedicated hardware is not required, so direct connection to a PC is possible. Programming/erasure in On-chip debug mode • JTAG or SWD interface is used Programming and erasure by self-programming • This allows code flash memory programming/erasure without resetting the system.	
Unique ID	A 16-byte ID code provided for each MCU	

图 25. RA4M3 MCU 的代码闪存和数据闪存规格

注：代码闪存的擦除状态为 FFh，但数据闪存的擦除状态不定。

8.4.1 后台操作

RA4 MCU 支持通过后台操作访问代码闪存和数据闪存。这意味着当开始执行编程或擦除操作时，用户可以继续从除正在操作的存储区以外的存储区中执行和访问存储器。例如，在对数据闪存进行擦除或编程操作时，CPU 可以从代码闪存执行应用程序代码。同样，在对代码闪存进行擦除或编程操作时，CPU 可以从 SRAM 执行应用程序代码。

8.4.2 ID 代码保护

具有 Cortex-M4 内核的 RA4 MCU 在选项设置存储区中有一个用作 ID 代码的 128 位存储器。如果此 ID 代码保留为空 (0xFF)，则不会使能任何保护，并且可以通过 boot 模式或使用片上调试器来访问 MCU。如果设置了 ID 代码，则用户可以控制对这些模式的访问。用户可以选择始终禁止连接，或者可以选择在输入匹配的 ID 代码时允许连接。有关更多信息，请参见“OCD/Serial Programmer ID Setting Register (OSIS)”和“ID Code Protection”以及 RA4 MCU 硬件手册中的相关内容。

对于搭载 Arm Cortex-M4 内核的 RA4 MCU，Renesas FSP 配置器提供了用于设置 ID 代码保护的选项。

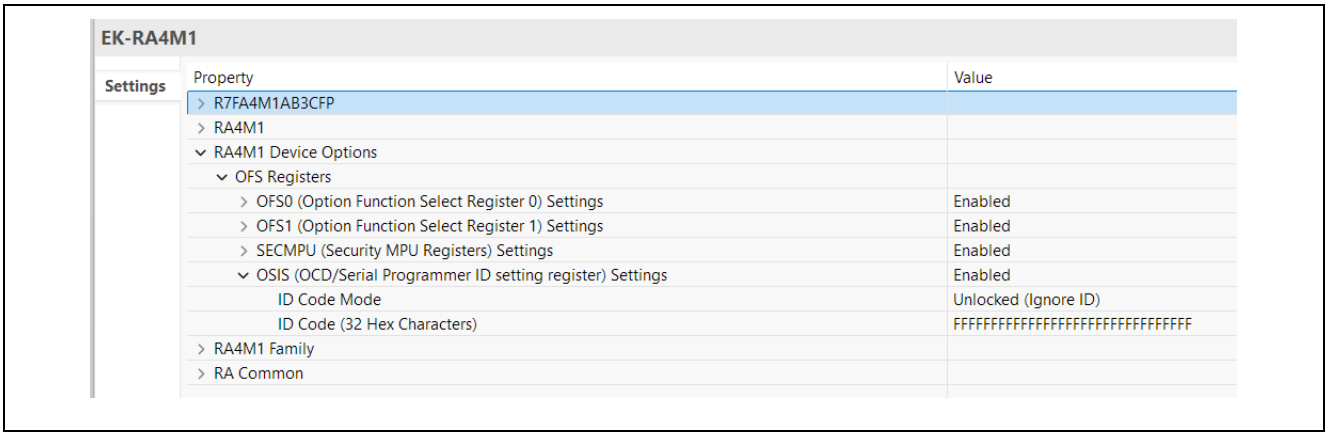


图 26. 通过 Renesas FSP Configurator 为 RA4M1（Arm Cortex-M4 Core）设置 ID 代码

注意：ID 码保护设置必须小心配置，以避免因误操作而造成 MCU 被锁定、无法访问。

8.4.3 闪存块保护

搭载 Cortex-M33 内核的 RA4 MCU 具有闪存块保护功能，可防止安全或非安全闪存区域被安全或非安全软件擦除或重新编程。值得注意的是，该保护功能同时适用于安全和非安全软件访问。

有关更多详细信息，请参阅 Securing Data at Rest Using the Arm® TrustZone®文档中的 Configuring the Flash Block Protection 的内容。

8.4.4 存储器保护单元

搭载 Cortex-M4 内核的 RA4 MCU 具有存储器保护单元 (MPU)。这些单元能够保护各种 MCU 区域免遭非法访问。具体选项包括允许读写、禁止写入和禁止读写。通过在特定的存储器地址上设置相应的常数值可选择这些选项之一。有关更多详细信息，请参见硬件手册中的“Memory Protection Unit”部分。

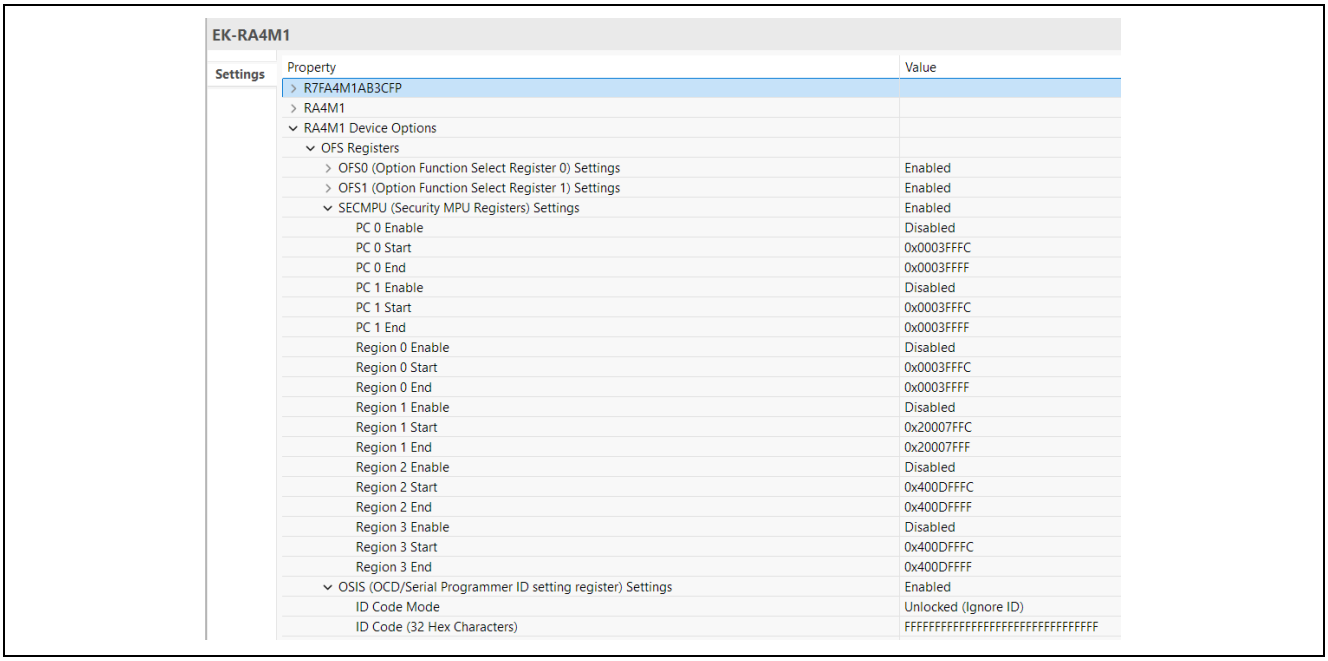


图 27. 通过 Renesas FSP 配置器为 RA4M1（Arm Cortex-M4 内核）设置 MPU

注：务必谨慎处理 MPU 设置，以避免发生可能导致阻止访问 MCU 区域的错误。

8.5 字节顺序的限制

包含指令代码的存储空间必须为小端格式，才能在 Arm® Cortex®-M 内核上执行代码。

9 寄存器写保护

寄存器写保护功能可防止重要寄存器因软件错误而被覆盖。使用保护寄存器 (PRCR) 设置要保护的寄存器。表 15 为 PRCR 位与要保护的寄存器之间的关联。

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
PRKEY[7:0]								—	—	—	PRC4 注 1	PRC3	—	PRC1	PRC0

图 28. PRCR 寄存器

注 1. Arm® Cortex®-M4 产品无此位。

表 15. PRCR 保护位

PRCR 位	说明
PRC0	• 与时钟生成电路有关的寄存器
PRC1	• 与低功耗模式有关的寄存器 • 与电池备用功能有关的寄存器
PRC3	• 与 LVD 有关的寄存器
PRC4 注 1	• 与安全功能有关的寄存器
PRKEY[7:0]	这些位控制对 PRCR 寄存器的写访问。要修改 PRCR 寄存器，可将 A5h 写入 8 个高地址位，将所需值写入 8 个低地址位，以将其用作 16 位单元。

注：

1. Arm® Cortex®-M4 产品无此位。
2. 每个 PRCR 位所适用的寄存器可能会因不同的 RA MCU 而异。有关更多信息，请参阅硬件用户手册中的“Register Write Protection”章节。

Renesas FSP 提供了两个 API（R_BSP_RegisterProtectEnable 和 R_BSP_RegisterProtectDisable），分别用于使能和禁用寄存器写保护。

10 I/O 端口配置

硬件手册中“I/O Ports”章节描述了基于外设选择和其他寄存器设置的引脚配置。以下列举了一些通用的内容。

需要注意的是，复位后到配置前，每个引脚将处于该引脚的默认状态。一些引脚的状态在短时间内可能与预期不符。无论使用哪种配置方法，都是这样的情况。用户需考虑这个现象可能对应用产生的影响，包括对其他系统功能的影响。

配置 I/O 端口时，可以直接写入寄存器，也可以通过 FSP 引脚配置功能进行配置。

10.1 多功能引脚选择设计方案

RA4 MCU 的大多数端口都具有多种外设功能。瑞萨提供了方便为每个 RA4 MCU 选择端口的工具，如：FSP 的引脚配置器。需要多个外设功能时，请在设计时使用以下方案选择端口功能。

- 先指定那些仅有一个端口选项的外设功能。例如，调试函数中的每个跟踪数据信号只有一个端口选项。当需要此功能时，请先分配这些端口。
- 然后指定那些具有有限几个端口选项的外设功能。例如，支持 OSPI 外设的产品通常只有两个端口选项用于每个 OSPI 信号。
- 最后指定那些具有多个端口选项的外设功能。例如，串行通信接口（SCI）通常有多个可用的端口选项。
- 某些外设功能端口选项是可互换的，而其他选项则必须在逻辑组中指定。例如，IIC 外设的某些端口带有后缀“_A”，而其他端口的信号名称中带有后缀“_B”，应选择具有相同后缀的端口。其他外设功能没有这种类型的后缀，端口可以互换，例如 USBFS 外设功能的 USB_VBUSEN 信号。详情请参阅本文档的 15.3 一节。

10.2 设置端口用作 GPIO

有两种方法可以设置端口用作 GPIO，一种是使用端口控制寄存器（PCNTR1），另一种是使用 PmnPFS 寄存器。

方法 1：端口控制寄存器 （PCNTR1）

- 通过将端口控制寄存器 1（PCNTR1）的端口方向位（PDRn）写“1”，引脚被设为输出。
- 端口方向位（PDRn）是可读/写的。将值设置为“1”后，引脚被设为输出。I/O 端口的默认状态为“0”（输入）。可以在 RA4 MCU 上读取端口方向寄存器。
- 相应端口控制寄存器（PCNTR1）的端口输出数据位（PODRn）是可读/写的。读取 PODR 时，读取的是输出数据锁存器的状态（而不是引脚电平）。
- 端口控制寄存器 2（PCNTR2）的端口输入位（PIDRn）是只读的。通过读取 PCNTR2 寄存器中的 PIDRn 位以读取引脚状态。

方法 2：端口 mn 引脚功能选择（PmnPFS）寄存器

- 端口模式寄存器（PMR）是可读/写的，用于指定单个引脚是用作 GPIO 还是外设引脚。复位后所有 PMR 寄存器都置为“0”，从而所有引脚被设为 GPIO。如果 PMR 寄存器置为“1”，则相应的引脚将用于外设功能。外设功能通过该引脚的 MPC 设置定义。
- 将引脚设置为输出时，建议先将端口的所需输出值写入数据锁存器，然后将方向寄存器设置为输出。虽然并非是在所有系统中都重要，但这样做可以防止正在被设置的端口上出现意外的输出毛刺。

一般来说，用 PCNTR1 配置端口能实现更快的访问，但可用的配置功能较少。而用 PmnPFS 寄存器会有更多可用的配置功能，但访问速度较慢。

通过 Renesas FSP 的 Pin Configuration 配置复位后的 GPIO 引脚，如下所示。配置 GPIO 后，可以使用 FSP 的 HAL 层 API 对其进行控制。

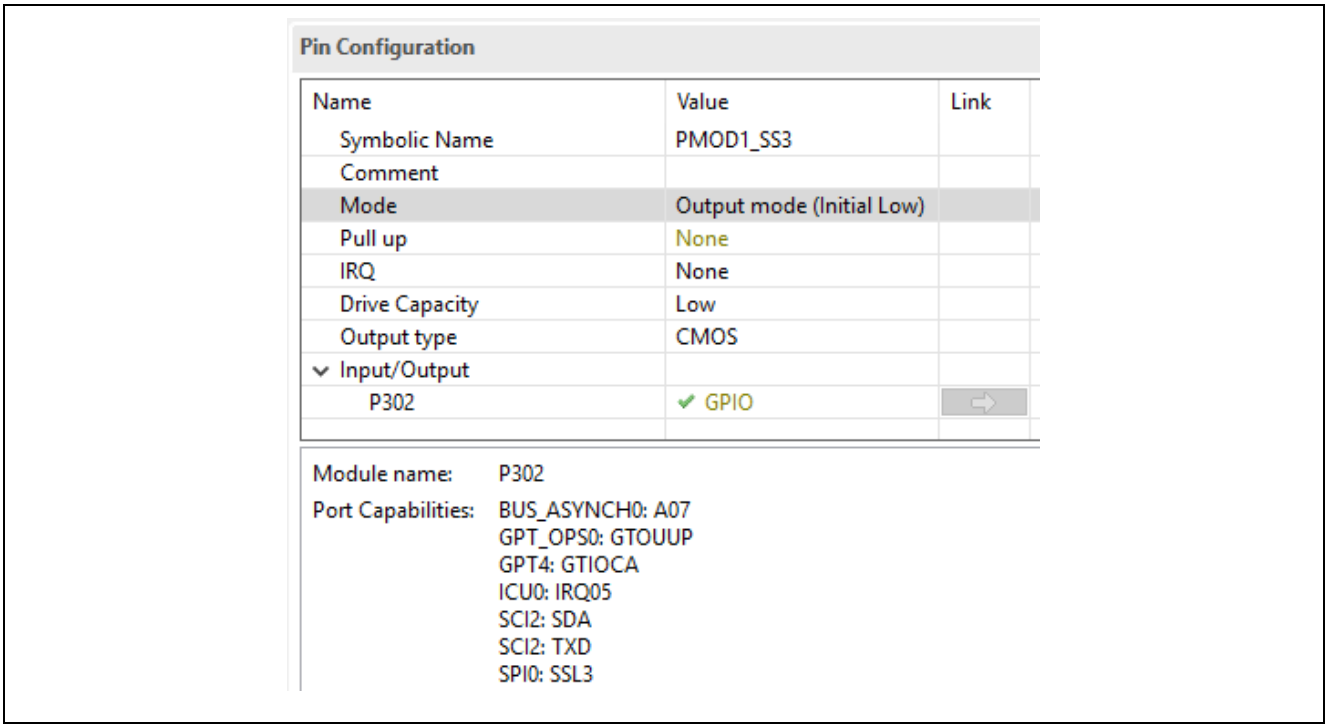


图 29. 通过 FSP 配置器将 P302 配置为输出低电平

10.2.1 内部上拉

- 端口 0 到 9 的大部分引脚都可以启用上拉电阻。上拉功能由每个端口 mn 引脚功能选择（PmnPFS）寄存器中的上拉控制位（PCR 位）进行控制。每个 PmnPFS 寄存器中的 PCR 位控制着对应端口上相应的引脚。
- 必须先在 PmnPFS 寄存器中通过相应的位将引脚设置位输入。将 PCR 位设置为“1”以启用上拉，将 PCR 位设置为“0”以禁用上拉。
- 复位后，除 SWD 接口引脚、JTAG 接口的非 TDO 引脚、MD 引脚以外，其他所有引脚的 PCR 位均被清零。
- 每当有引脚被指定为外部总线引脚、GPIO 输出或外设功能输出引脚时，上拉电阻就会自动关闭。

10.2.2 开漏输出

- 配置为输出的引脚通常用作 CMOS 输出。
- 端口 0 到 9 的大多数引脚都可以配置为 NMOS 开漏输出。
- 各个端口 mn 引脚功能选择寄存器（PmnPFS）的 N 沟道开漏控制位（NCODR）控制哪些引脚在开漏模式下工作。将各 PmnPFS 寄存器的 NCODR 位设为“1”，则对应的引脚被设置为输出开漏，设为“0”，则端口被设置为 CMOS 输出。

10.2.3 驱动能力

- 驱动能力的切换由各端口 mn 引脚功能选择寄存器（PmnPFS）的驱动能力控制位（DSCR）控制。
- 复位后，会将所有 DSCR 寄存器清零，因此会将所有引脚设置为低驱动能力输出。设置“00”以外的值会更改所选引脚的输出的驱动能力。
- 对于 Arm® Cortex®-M4 产品，所有引脚的最大总输出为 60 mA。对于 Arm Cortex-M33 产品，所有引脚的最大总输出为 80 mA。
- 驱动能力的差异如下所示。实际输出电流会因产品和引脚类型而有所不同。详细信息请参见 MCU 用户手册。

- RA4L1 和 RA4C1 的输出驱动能力是固定的。这些 MCU 的 PmnPFS 寄存器的 bit 10 和 bit 11 始终设置为 0。

表 16. 输出引脚驱动能力等级

典型输出引脚	DSCR[1:0]	驱动能力	最大值（mA）
单个引脚的允许输出电流	0 0	低驱动	4.0
单个引脚的允许输出电流	0 1	中驱动	8.0
单个引脚的允许输出电流	1 0	-	
单个引脚的允许输出电流	1 1	高驱动	32

高驱动输出引脚	DSCR[1:0]	驱动能力	最大值（mA）
单个引脚的允许输出电流	0 0	低驱动	4.0
单个引脚的允许输出电流	0 1	中驱动	8.0
单个引脚的允许输出电流	1 0	-	
单个引脚的允许输出电流	1 1	高驱动	40

输出驱动能力会对电路板设计的整体性能产生重大影响。在为每个输出选择驱动器能力时，应考虑以下几点。

- 建议先将所有引脚设置为低驱动器能力（默认）并评估性能。
- 根据电路板布局，设置为中等或高驱动能力的引脚可能会导致更高的 EMI 辐射。
- 长走线可能需要更高的驱动能力才能使信号准确地传到接收器。
- RA4L1 和 RA4C1 不具备选择驱动能力的功能。

10.3 设置和使用端口外设功能

端口 mn 引脚功能选择寄存器（PmnPFS）用于配置每个端口的特性。PSEL 位用来设置每个端口所选的外设功能。

- 由于大多数引脚具有多种功能，因此 RA4 MCU 具有引脚功能控制寄存器（PmnPFS 可用于更改分配给引脚的功能。
- 每个引脚都有自己的 PmnPFS 寄存器。
- 每个 PmnPFS 寄存器都可将引脚设为外设功能（PSEL 位）、IRQ 输入引脚（ISEL 位）或模拟输入引脚（ASEL 位）。如果 ASEL 位设为“1”（使用引脚作为模拟输入引脚），则应将引脚的 PMR 位设置为 GPIO 使用，并将引脚的 PDR 位设置为输入。
- 请参阅硬件手册中“I/O Ports”一章中“Peripheral Select Settings for each Product”的内容。
- 为防止外设引脚上出现意外的电压电平，请确保在修改引脚的 PmnPFS 寄存器之前先清除目标引脚的端口模式控制位（PMR）。
- 复位后，所有 PmnPFS 寄存器均处于写保护状态。如需对这些寄存器进行写操作，必须首先使用写保护寄存器（PWPR）来使能写入。
- 设置 PmnPFS 寄存器时应格外小心，切勿将一个功能分配给多个引脚。用户不应这样做，但 MCU 允许上述操作。如果发生这种情况，则引脚上的功能将处于未定义状态。
- 如果打算使用外部总线、以太网控制器或 USB，则在使用这些外设之前，必须先配置 MPC 中的其他寄存器。
- 下图为使用 FSP 引脚配置启用 QSPI 引脚的示例。

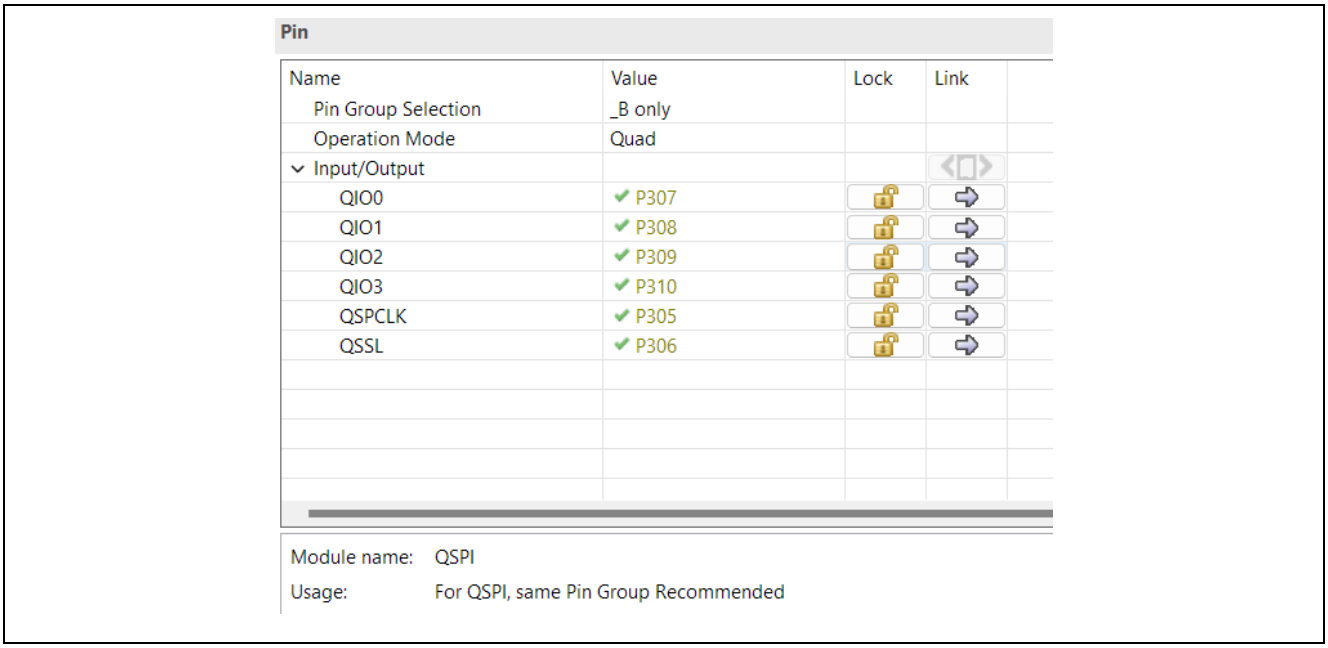


图 30. 使用 FSP 的 Pin Configurator 使能 QSPI 引脚

10.4 设置和使用 IRQ 引脚

- 某些端口引脚可用作硬件输入中断（IRQ）。请参阅硬件手册中“I/O Ports”一章中“Peripheral Select Settings for each Product”的内容，以了解您的 MCU 上哪些引脚可用。
- 某些 IRQ 引脚具有“-DS”后缀（例如 IRQ1-DS）。表示该引脚可用于将 MCU 从深度软件待机（Deep Software Standby）模式中唤醒。
- 注意：不能同时使用 IRQn 和 IRQn-DS。带有“-DS”和不带“-DS”后缀的相同中断在内部连接到同一中断，即使它们使用不同的外部引脚连接也是如此。
- 要将端口引脚设置为 IRQ 引脚，必须将引脚 PFS 寄存器中的中断输入功能选择位（ISEL）设置为“1”。
- 引脚可同时用于 IRQ 和外设功能。用户应在引脚的 PFS 寄存器中设置 ISEL 和 PSEL 位。
- 相同编号的 IRQ 功能只能在一个引脚上启用。
- IRQ 引脚可在检测到以下情况时触发中断：
  - 低电平
  - 下降沿
  - 上升沿
  - 上升沿和下降沿

  使用 IRQ 控制寄存器（IRQCRi）选择触发方式。
- 数字滤波器可用于 IRQ 引脚。滤波器基于以四种可选时钟速率（PCLK、PCLK/8、PCLK/32、PCLK/64）之一对信号进行重复采样。它们滤除短脉冲：在特定滤波速率下，任何少于 3 个样本的高脉冲或低脉冲。滤波器可用于滤除这些线路中的振铃和噪声，但是由于速率过快而无法滤除诸如机械开关弹跳之类的长时间事件。使能滤波会给硬件 IRQ 线增加一小段响应延迟时间（滤波时间）。
- 每个 IRQ 引脚可以单独使能数字滤波。此过程通过将每个 IRQ 的 IRQCRi 寄存器中的 IRQ 脚数字滤波器使能 (FLTEN) 位置 1 来实现。
- 每个 IRQ 引脚可以单独配置数字滤波的时钟速率。此过程通过将每个 IRQ 的 IRQCRi 寄存器中的 IRQ 引脚数字滤波器设置 (FCLKSEL[1:0]) 位置 1 来实现。
- 图 31 和图 32 为使用 Renesas FSP 使能和配置 IRQ 引脚的示例。

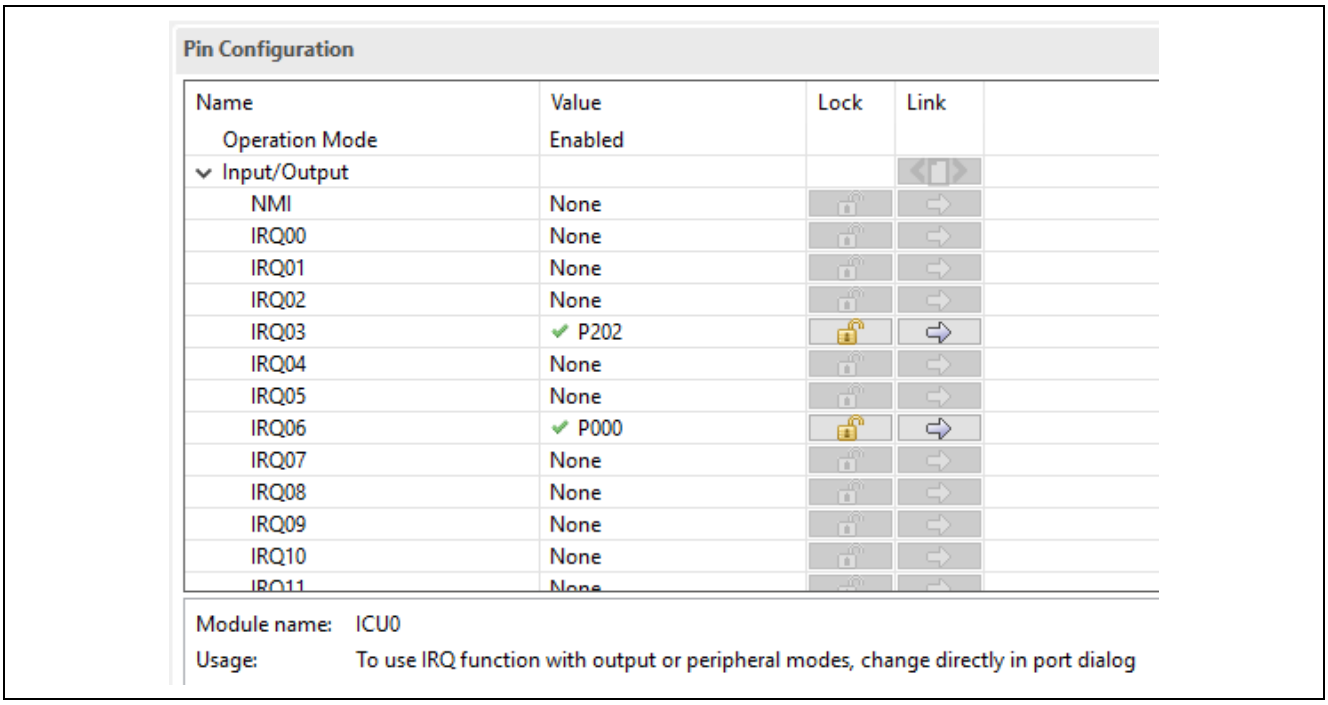


图 31. 使用 FSP 的 Pin Configurator 将 P202、P000 分别设为 IRQ03、IRQ06 输入

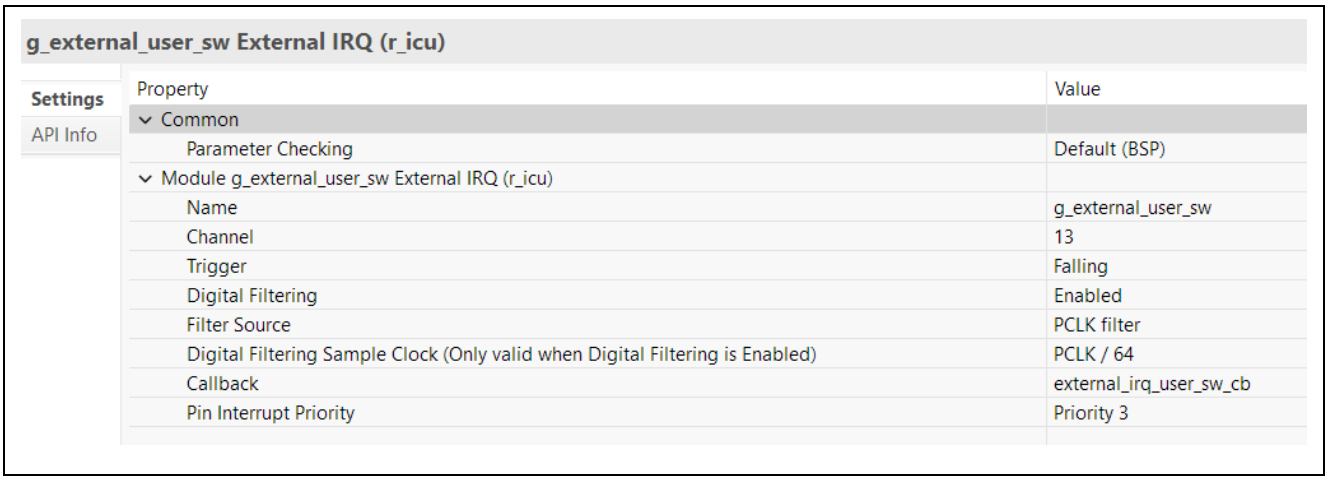


图 32. 通过 FSP 配置器配置 IRQ13

10.5 未使用的引脚

注意：某些引脚需要特定的端接处理：具体建议请参阅硬件手册的“Handling of Unused Pins”章节。

悬空的未使用引脚会消耗额外的功率，并使系统更易受到噪声问题的影响。使用以下方法可以处理未使用的引脚：

1. 第一种方法是将引脚设置为输入（复位后的默认状态），并使用电阻将引脚连接到 VCC 或 VSS。对于 MCU 而言，连接到 VCC 或 VSS 没有区别；但是，从系统噪声的角度来看，它可能各有优劣。VSS 可能是最典型的选择。 避免将引脚直接连接到 VCC 或 VSS，因为意外写入端口的方向寄存器（将引脚设置为输出）可能会造成输出短路。
2. 第二种方法是将引脚设置为输出。引脚电平是高电平还是低电平并不重要；但是，将引脚设置为输出并将输出设置为低电平，会在内部将引脚连接到接地层，这可能有助于解决整体系统噪声问题。将未使用的引脚设置为输出的一个缺点是端口的配置必须通过软件控制完成。

当 MCU 保持在复位状态时，直到方向寄存器设置为输出时，该引脚将是浮动输入状态，可能会消耗额外的电流。如果在此期间可以容忍额外的电流，则此方法无需第一种方法所需的外部电阻。

3. 将引脚保留为输入并用外部电阻端接的另一种变通方法是，使用 MCU 许多端口上可用的内部上拉电阻。这与将引脚设置为输出具有相同的限制（需要程序设置端口），但由于器件不会驱动引脚，因此确实限制了意外将引脚短路到地、相邻引脚或 VCC 的影响。

10.6 不存在的引脚

每个 RA4 MCU 系列均提供多种封装尺寸，总引脚数也各有不同。对于小于该 MCU 系列最大封装（通常为 100 引脚或 144 引脚）的任意封装，应将 PDR 寄存器中对应的不存在端口位设置为“1”（输出），并将 PODR 寄存器中对应的位设置为“0”。用户可通过查阅硬件用户手册中“I/O Ports”的“Specifications of I/O Ports”表，确认每种 MCU 封装上可用的端口。例如，端口 1 的引脚 0 和引脚 1 仅在 176 引脚封装中可用。请注意，对于不存在的引脚，无需进行其他额外处理。

10.7 电气特性

普通 GPIO 端口通常需要 CMOS 电平输入（高≥0.8 × VCC，低≤0.2 × VCC）。某些 GPIO 端口具有施密特触发器输入，其输入要求略有不同。有关详细信息，请参阅硬件手册中“Electrical Characteristics”章节的内容。

11 模块停止功能

为了最大限度地提高电源效率，RA4 MCU 允许通过写入模块停止控制寄存器 （MSTPCRi，i = A，B，C，D，E）来单独停止片上外设。一旦模块停止，就无法访问与该模块关联的寄存器。

复位后，除 DMAC、DTC 和 SRAM 模块外，大多数都处于模块停止状态。有关详细信息，请参阅硬件手册。

如需访问这些外设的寄存器，必须先将 MSTPCRi 寄存器中的相应位写“0”使该外设退出停止模式，然后才可以启用。

可通过将 MSTPCRi 寄存器中的相关位写“1”来停止该外设。

瑞萨 FSP 的 HAL 驱动程序会自动处理模块启动/停止功能。

12 中断控制单元

中断控制单元(ICU)用于控制连接到 NVIC、DTC 和 DMAC 模块的事件信号。ICU 同时也负责不可屏蔽中断的控制。 图 33 为 ICU 规格示例，图 34 通过示例说明了如何通过 I/O 引脚引发 IRQi 事件。有关具体信息，请参考各 RA4 MCU 的硬件手册。

Parameter		Description
Maskable interrupts	Peripheral function interrupts	• Interrupts from peripheral modules Number of sources: 223 (select factor within event list numbers 32 to 511)
	External pin interrupts	• Interrupt detection on low level*4, falling edge, rising edge, rising and falling edges. One of these detection methods can be set for each source • Digital filter function supported • 16 sources, with interrupts from IRQi (i = 0 to 15) pins.
	Interrupt requests to CPU (NVIC)	• 96 interrupt requests are output to NVIC.
	DMAC control	• The DMAC can be activated using interrupt sources*1 • The target interrupt source can be selected individually for every DMAC channels.
	DTC control	• The DTC can be activated using interrupt sources*1 • The method for selecting an interrupt source is the same as that of the interrupt request to NVIC.
Non-maskable interrupts*2	NMI pin interrupt	• Interrupt from the NMI pin • Interrupt detection on falling edge or rising edge • Digital filter function supported
	WDT underflow/refresh error*3	Interrupt on an underflow of the down-counter or occurrence of a refresh error
	IWDT underflow/refresh error*3	Interrupt on an underflow of the down-counter or occurrence of a refresh error
	Low voltage detection 1*3	Voltage monitor 1 interrupt of the voltage monitor 1 circuit (LVD_LVD1)
	Low voltage detection 2*3	Voltage monitor 2 interrupt of the voltage monitor 2 circuit (LVD_LVD2)
	RPEST*5	Interrupt on SRAM parity error
	RECCST*5	Interrupt on SRAM ECC error
	Bus master MPU error*5	Interrupt on MPU bus master error
	TZFST*5	TrustZone Filter error.
	CPEST*5	Cache RAM Parity error.
	Oscillation stop detection interrupt*3	Interrupt on detecting that the main oscillation has stopped
Low power modes		• Sleep mode: return is initiated by non-maskable interrupts or any other interrupt source • Software Standby mode: return is initiated by non-maskable interrupts. Interrupt can be selected in the WUPEN register. • Snooze mode: return is initiated by non-maskable interrupts. Interrupt can be selected in the SELSR0 and WUPEN registers. See section 13.2.17. SELSR0 : SYS Event Link Setting Register and section 13.2.18. WUPEN0 : Wake Up Interrupt Enable Register 0, section 13.2.19. WUPEN1 : Wake Up interrupt enable register 1.
TrustZone Filter		Available

图 33. RA4M3 ICU 的规格

Pin name	I/O	Description
NMI	Input	Non-maskable interrupt request pin
IRQi (i = 0 to 15)	Input	External interrupt request pins

图 34. RA4M3 ICU 的 I/O 引脚

下图为使用 Renesas FSP Configurator 使能和配置 Renesas FSP 中断的示例。通过 FSP 将 ICU 和中断配置为 HAL 驱动程序配置的一部分。

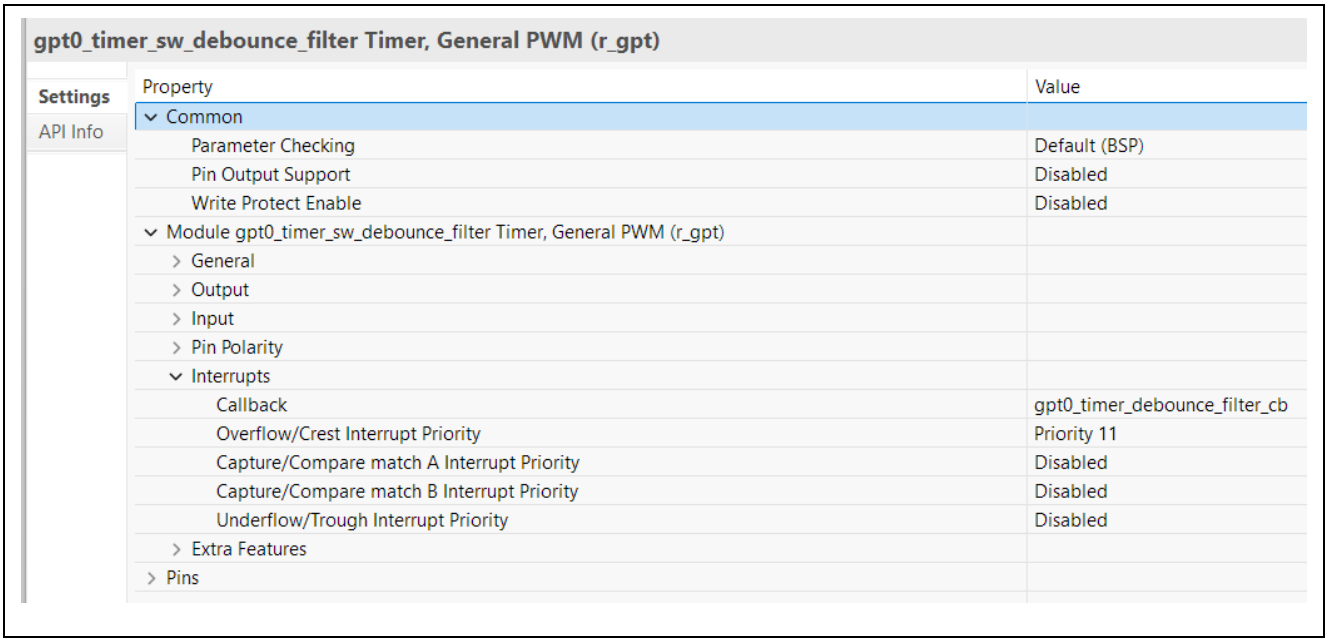


图 35. 使能 GTP0 溢出中断并设置将由中断服务程序调用的用户回调函数

13 低功耗

RA4 产品具有多种用于降低功耗的功能。其中包括设置时钟分频器、停止模块、在正常模式下选择电源控制模式以及转换为低功耗模式。有关更多详细信息，请参见硬件手册中“Low Power Modes”章节的内容。

RA4 MCU 支持四种不同类型的低功耗模式（LPM），具体会因 MCU 不同有差异。分别为：

- 休眠模式（Sleep Mode）
- 软件待机模式（Software Standby Mode）
- SNOOZE 模式（SNOOZE Mode）
- 深度软件待机模式（Deep Software Standby Mode）（并非所有 RA4 MCU 都支持此模式）

下表为可用于降低功耗的功能。

表 17. 低功耗模式的功能规格

项目	规范
通过修改时钟信号降低功耗	可以为系统时钟 (ICLK)、外设模块时钟（PCLKA, PCLKB, PCLKC, PCLKD）和闪存接口时钟 (FCLK) 单独选择分频比。注 1
模块停止	可以单独停止每个外设模块的功能。
低功耗模式	• 休眠模式 • 软件待机模式 • SNOOZE模式 • 深度软件待机模式
电源控制模式	多种工作电源控制模式： • 高速模式 • 中速模式注3 • 低电压模式注4 • 低速模式 • 副时钟振荡器速度模式
TrustZone® 滤波器注 2	可以为每个寄存器设置安全属性

注：

1. 有关详细信息，请参阅硬件手册中“Clock Generation Circuit”章节的内容。
2. 适用于支持 TrustZone ® 安全功能的产品
3. RA4 MCU不支持此模式。
4. 仅RA4M1支持此模式。

下表列出了转换到低功耗模式的条件、CPU 和外设模块的状态，以及取消每种模式的方法。

表 18. 低功耗模式

运行状态 注 1	休眠模式	软件待机模式	Snooze 模式	深度软件待机模式
进入条件	当 SBYCR.SSBY = 0 且满足以下任一条件时： **[条件 1]** ・执行 WFI 指令 ・CPU 无法受理有效的中断请求注 2。 （包括从执行 WFI 指令到完成向休眠模式转换期间的过渡过程） **[条件 2]** ・SCR.SLEEPONEXIT = 1 ・所有异常处理程序执行完成 ・CPU 无法受理有效的中断请求*1。 （包括从执行 WFI 指令到完成向休眠模式转换期间的过渡过程）	当 SBYCR.SSBY = 1 且 DPSBYCR.DPSBY = 0（如果存在该寄存器）时，满足以下任一条件： **[条件 1]** ・执行 WFI 指令 ・CPU 无法受理有效的中断请求注 2。 （包括从执行 WFI 指令到完成向休眠模式转换期间的过渡过程） **[条件 2]** ・SCR.SLEEPONEXIT = 1 ・所有异常处理程序执行完成 ・CPU 无法受理有效的中断请求注 1。 （包括从执行 WFI 指令到完成向休眠模式转换期间的过渡过程）	软件待机模式下的 Snooze 请求触发。 SNZCR.SNZE = 1	当 SBYCR.SSBY = 1 且 DPSBYCR.DPSBY = 1 时，满足以下任一条件： **[条件 1]** ・执行 WFI 指令 ・CPU 无法受理有效的中断请求注 2。 （包括从执行 WFI 指令到完成向休眠模式转换期间的过渡过程） **[条件 2]** ・SCR.SLEEPONEXIT = 1 ・所有异常处理程序执行完成 ・CPU 无法受理有效的中断请求注 1。 （包括从执行 WFI 指令到完成向休眠模式转换期间的过渡过程）
退出方式	所有中断。 该模式下可用的任何复位。	为此模式定义的中断。 该模式下可用的任何复位。	为此模式定义的中断。 该模式下可用的任何复位。	为此模式定义的中断。 该模式下可用的任何复位。
通过中断取消后的状态	程序执行状态	程序执行状态	程序执行状态	复位状态
通过复位取消后的状态	复位状态	复位状态	复位状态	复位状态

注：

1. 更多详细信息，请参阅硬件手册中的“Operating Conditions of Each Low Power Mode”。
2. 有效中断请求是指未被当前异常的优先级以及 BASEPRI 所设置的优先级屏蔽的任何中断/异常。此外，如果中断请求基于 IELSRn，则必须通过 NVIC_ISERn 使能该中断。

RA4 MCU 包含相应的寄存器设置，可使 MCU 在 Normal 模式和 Sleep 模式下以更低功耗运行。这些模式称为工作功耗控制模式，由 OPCCR 寄存器控制。

以下总结了工作功耗控制模式，以及各模式下允许的最大时钟和电压等级。并非所有 RA4 MCU 都支持全部功耗控制模式。请参阅硬件手册，以确认所用的 MCU 支持哪些模式。

模式	振荡器						
	PLL、PLL2 和 PLL_RTC 注2	高速片上振荡器	中速片上振荡器	低速片上振荡器	主时钟振荡器	副时钟振荡器	IWDT 专用片上振荡器
高速	适用	适用	适用	适用	适用	适用	适用
中速	适用	适用	适用	适用	适用	适用	适用
低电压	不适用	适用	适用	适用	适用	适用	适用
低速	不适用	适用	适用	适用	适用	适用	适用
副时钟振荡器速度	不适用	不适用	不适用	适用	不适用	适用	适用

注：

1. 虽然可以将 OPCCR 寄存器的值设置为任一种低功耗运行模式，但时钟和电压等级也必须设置为满足目标模式的要求。否则，OPCCR 寄存器中的设置将无法起到降低功耗的作用。
2. 仅 RA4C1 具备。

为了实现最低功耗，请在时钟生成电路中尽可能使用最大的分频比。

低功耗模式可由多种中断源解除，例如 RES 引脚复位、上电复位、电压监视器复位以及外设中断。有关不同低功耗模式的中断源列表，请参阅硬件手册中的“Low Power Modes”章节。

仅 Snooze 模式是通过 Snooze 请求从软件待机模式进入 Snooze 模式而触发的。进入其他低功耗模式时，需要在待机控制寄存器（SBYCR）中进行适当设置后执行 WFI 指令。

Renesas FSP 提供低功耗模式（LPM）驱动程序及驱动程序配置器，用于设置低功耗模式、唤醒源/解除源等。

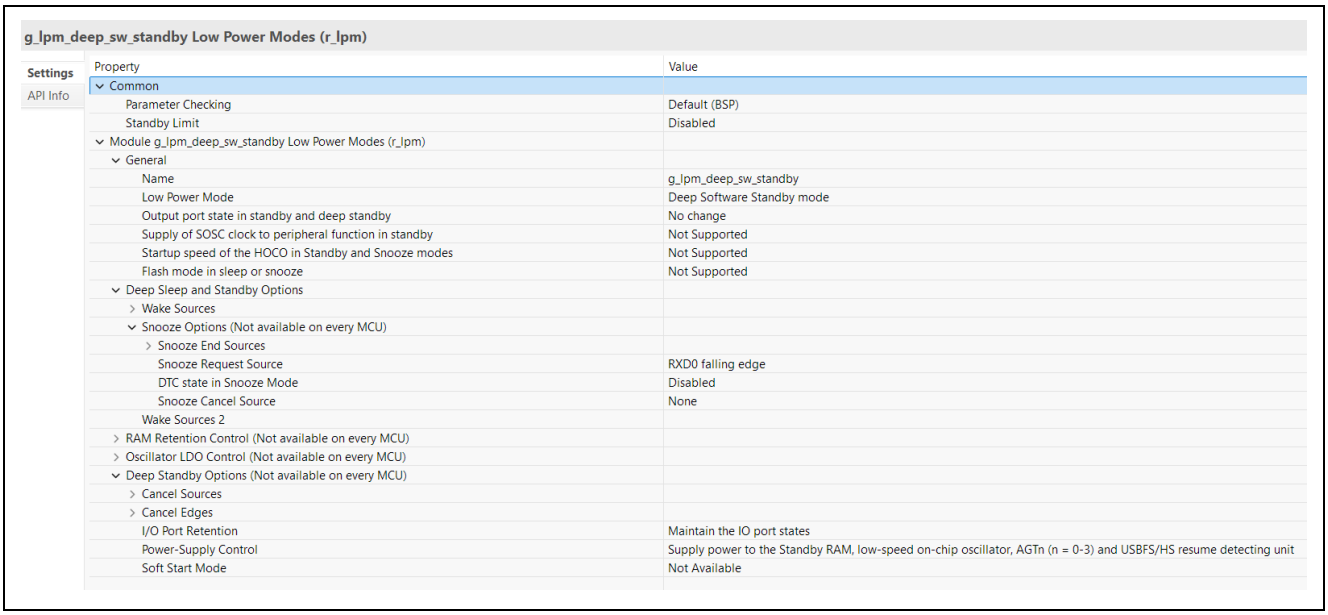


图 36. 使用 FSP 配置器为 RA4T1 设置低功耗模式

FSP 配置器设置特定 LPM 模式后，LPM 驱动程序的 API 可用于初始化 LPM 驱动程序并将 MCU 置于配置的 LPM 模式：

```
/* Open LPM driver and initialize LPM mode */
err = R_LPM_Open(&g_lpm_deep_sw_standby_ctrl, &g_lpm_deep_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_deep_sw_standby_ctrl);
/* Handle error */
if (FSP_SUCCESS != err)
  {
    return (err);
  }
```

有关如何在 LPM 下使用 FSP 配置器和 API 驱动，请参考 RA MCU 应用说明：Getting Started with Low Power Applications。

14 外部总线

Arm® Cortex®-M33 的 MCU 的总线由 32 位 AHB（Advanced High-performance Bus，高级高性能总线）总线矩阵组成。图 37 为总线主设备和总线从设备，图 38 为总线配置。

注：执行 Arm Cortex 指令代码时，存储器空间以小端格式访问。

Classification	Bus Master/Slave name	Bus I/F Max Freq	Sync Clock	Specifications
Bus Masters	Code bus (Cortex-M33)	100 MHz	ICLK	Connected to the CPU Instruction Cache for instructions and operands
	System bus (Cortex-M33)	100 MHz	ICLK	Connected to the CPU Data Cache for system
	DMAC / DTC	100 MHz	ICLK	Connected to the DMAC/DTC
Bus Slaves	FHBIU	100 MHz	ICLK	Connected to Code Flash memory and Configuration area
	FLBIU	50 MHz	FCLK	Connected to Data Flash memory, FACI
	S0BIU	100 MHz	ICLK	Connected to SRAM0 (Standby RAM)
	PSBIU	100 MHz	ICLK	Connected to peripheral system modules (DTC, DMAC, ICU, Flash, MPU, SRAM, Debug/Trace module, System controller and BUS controller)
	PLBIU	50 MHz	PCLKB	Connected to peripheral modules (CAC, ELC, I/O ports, POEG, RTC, WDT, IWDT, AGT, IIC, CAN, USBFS, SDHI, SSIE, TSN, and CTSU)
	PHBIU	100 MHz	PCLKA	Connected to peripheral modules (GPT, SCI, SPI, CRC, DOC, ADC12, DAC12 and SCE9)
	EQBIU (QSPI area)	100 MHz	PCLKA	Connected to the QSPI (External Memory Interface)

Note: FHBIU: Flash High speed Bus Interface Unit.
FLBIU: Flash Low speed Bus Interface Unit.
S0BIU: SRAM0 Bus Interface Unit.
PSBIU: Peripheral System Bus Interface Unit.
PLBIU: Peripheral Low speed Bus Interface Unit.
PHBIU: Peripheral High speed Bus Interface Unit.
EQBIU: External memory interface Qspi Bus Interface Unit.

图 37. RA4M3 总线规格

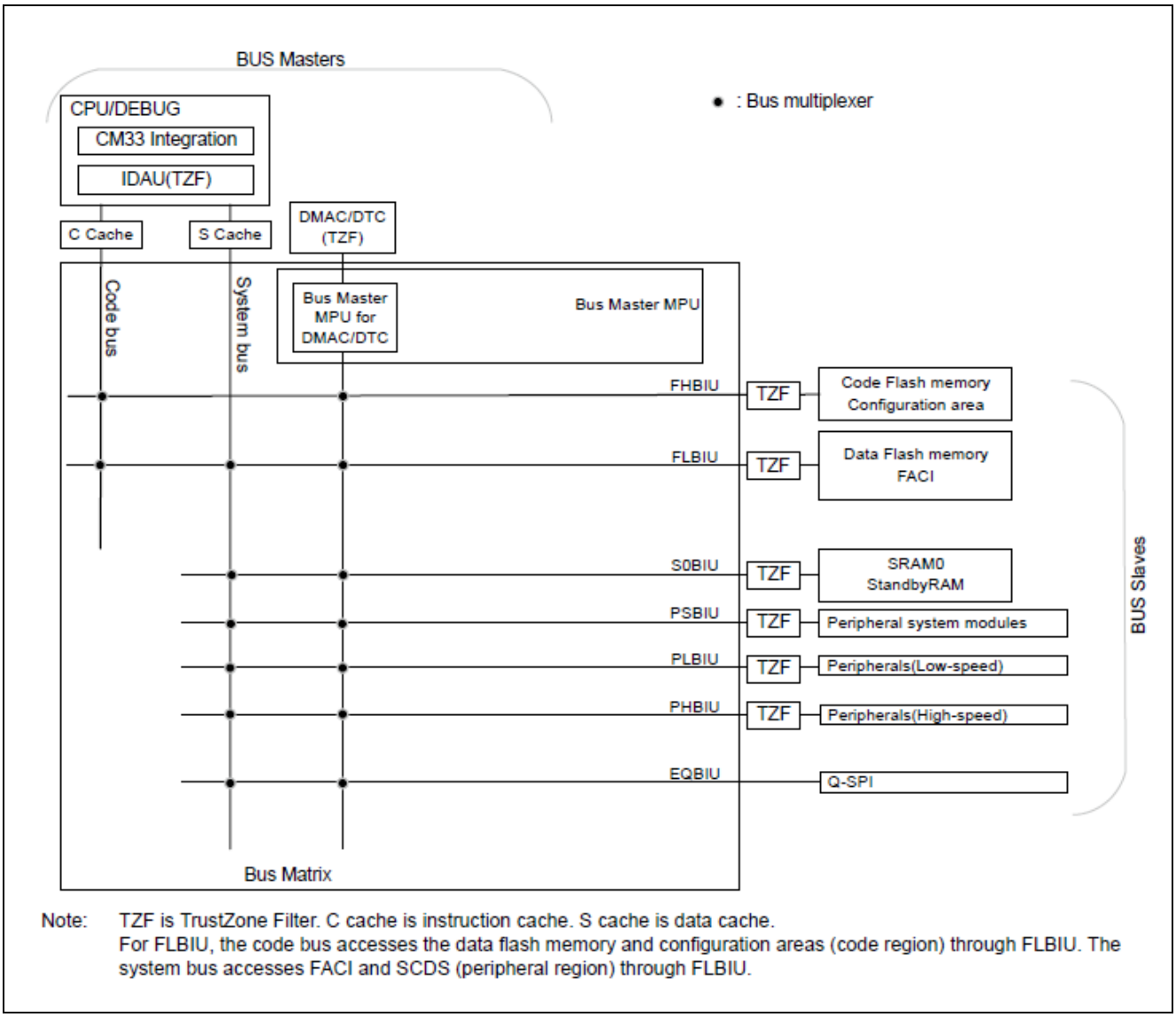


图 38. RA4M3 总线连接

14.1 总线错误监视

总线错误监视系统监视每个单独的区域，当检测到错误发生时，会通过 AHB-Lite 错误响应协议将错误返回给发请求的主 IP。

14.1.1 总线错误类型

每条总线上都可能发生以下类型的错误。

- 非法地址访问
- 总线主 MPU 错误
- TrustZone® 滤波器错误（仅限支持 Trustzone 的 MCU）。
- 每个从 IP 传输的总线错误

14.1.2 发生总线错误时的处理

发生总线错误时，无法保证正常运行，并且会将错误返回给发起请求的 Master IP。每个主控发生的总线错误均存储在 BUSnERRADD 和 BUSnERRSTAT 寄存器中。只能通过复位来清空这些寄存

器。有关更多信息，请参见硬件手册的 Bus Error Address Register (BUSnERRADD) 和 Bus Error Status Register (BUSnERRSTAT)的内容。

注：DMAC 和 DTC 不会收到总线错误，因此它们的运行不受总线错误的影响。

15 一般布局实践

15.1 数字域与模拟域

瑞萨 RA4 MCU 具有三种主要的引脚功能：电源、数字和模拟。

通常，电源引脚专用于电压和基准输入，不具有多种功能。电源引脚通常专用于 MCU 中的特定部分或域。例如，MCU 的主电源电压将为数字内核、许多数字外设功能和许多数字 I/O 引脚供电。数字域可以定义为数字电路、数字 I/O 引脚和相关电源引脚。指定用于模拟功能的电源引脚（如 AVCC0 和相关的 AVSS0）在 MCU 内提供特定的模拟电路，该电路与数字域电路分开。模拟域可以定义为模拟电路、模拟 I/O 引脚和相关电源引脚。

数字信号通常是与周期性时钟相关的重复开关模式。数字信号的转换往往是相对尖锐的边缘，转换之间具有稳定的高电平或低电平。在指定的时间范围内，每个信号必须稳定在可接受的电压电平（称为逻辑状态）。信号状态通常以预定的时钟间隔进行采样，使用时钟的边沿转换来评估相关的数据信号。 只要电压电平保持在指定范围内，数字信号电压电平的微小变化通常是可以接受的。然而，对数字信号的较大外部影响会对数字信号产生严重影响，这可能导致在对数据进行采样时出现不正确的逻辑状态。

模拟信号通常完全不同。模拟信号可能是周期性的，但模拟信号的评估通常是对一定范围内的电压测量，而不是逻辑状态。模拟信号的电压电平根据特定的触发事件进行采样，并使用 MCU 中的模拟电路处理测量结果。模拟测量的精度与采样电压电平的精度直接相关。任何不希望出现的外部影响，即使仅使模拟输入信号的电压电平发生轻微变化，也会影响测量的准确性。

由于瑞萨 RA4 MCU 的 I/O 引脚具有高度多路复用特性，因此许多 I/O 引脚可用于模拟或数字功能。这可能导致数字和模拟功能可能重叠并导致数据错误的情况。

为了最大程度地减少数字和模拟信号域之间的潜在问题，请考虑以下策略。

- 分配 I/O 引脚功能时，请选择引脚功能，以便模拟引脚和数字引脚尽可能在物理上分开。
- 每个模拟信号应尽可能与所有其他信号分开。
- PCB 布线应尽可能隔离每个模拟信号。避免在同一区域出现任何其他信号，无论是模拟信号还是数字信号。
- 确保模拟电源电压和模拟基准电压包括适当的交流滤波器。这可以是位于 MCU 电压引脚附近的推荐电容或适当的电感滤波器的形式。目标是提供纹波很小或没有纹波的供电电压和基准电压。
- 在 PCB 设计中使用专用电源层时，应避免在模拟电压区域出现数字信号，并避免在数字电压区域出现模拟信号。

对于高度敏感的应用，强烈建议使用仿真工具评估特定设计，以了解电路设计对性能的影响。这也包括精密传感器设计或超高速图总线接口等应用。有关每个外围功能的具体要求，请参阅硬件手册中的“Electrical Characteristics”一章。

15.2 高速信号设计注意事项

随着数字信号时钟速度的提高，外部干扰对这些信号的影响会变得更加显著。一些外设功能可归类为高速数字信号。对于高速数字信号，应考虑其他设计因素。

串扰是指一个信号的跳变对另一个附近信号产生感应影响的情况。当这种串扰效应足够强时，第一个信号可能会导致第二个信号出现误差。为了减少串扰的影响，请使用以下通用 PCB 布线指南。

- 在同一布线层上的信号之间提供足够的空间。通常，同一数字组的信号之间至少保留一个走线宽度空间，不同数字组的信号之间至少保留 3 ~ 5 倍走线宽度空间。
- 在同一布线层上的时钟信号和数据信号之间提供额外的空间。通常，在时钟和任何其他数字信号之间至少保持 3 ~ 5 倍走线宽度的空间。
- 避免在任何相邻的布线层上并行走数字信号。如果信号必须在相邻信号层上，请尽可能仅使用正交交叉。

如果可能，在信号层之间使用电源层或接地层将 PCB 信号层分开。电源层或接地层的实心铜可以充当数字信号的“屏蔽层”。

每个标准化接口都有特定的要求。为确保 PCB 的设计能够避免信号串扰问题，强烈建议参考设计中每个接口的相关标准。

15.3 信号组选择

某些引脚名称添加了_A、_B 或_C 后缀来指示信号组。分配某些外设功能（如 IIC、SPI、SSIE、ETHERC、SDHI 等）时，请选择具有相同后缀的功能引脚。在某些情况下，会针对每个信号组测量硬件手册的“Electrical Characteristics”一章中的 AC 特性。如果信号组混淆，则不能保证外设正常工作，并且所述的 AC 特性可能不适用。

如果外设功能的引脚名称没有信号组后缀，则可以安全地为每个功能信号选择最方便的引脚分配。

请参阅硬件手册“I/O Ports”一章中的“Peripheral Select Settings for each Product”和 PmnPFS 寄存器设置注意事项部分。

16 参考文献

制作本快速设计指南时参考了以下文档：

参考资料	文档编号	文档名称
1	R01UH0887	Renesas RA4M1 Group, User's Manual: Hardware
2	R01UH0892	Renesas RA4M2 Group, User's Manual: Hardware
3	R01UH0893	Renesas RA4M3 Group, User's Manual: Hardware
4	R01UH1081	Renesas RA4L1 Group, User's Manual: Hardware
5	R01UH1137	Renesas RA4C1 Group, User's Manual: Hardware
6	R01UH0577	RA Arm® TrustZone® Tooling Primer
7	R01UH0467	Security Design with Arm® TrustZone® - IP Protection
8	R20AN0468	Securing Data at Rest Using the Arm® TrustZone®
9	R11AN0475	Establishing and Protecting Device Identity using SCE9 and Arm® TrustZone®
10	R11AN0469	Device Lifecycle Management Key Installation

网页和支持

可从以下链接了解更多关于 RA 系列的要点，下载组件和相关文档，以获得支持。

RA 产品	www.renesas.com/ra
RA 产品支持论坛	www.renesas.com/ra/forum
RA FSP	www.renesas.com/FSP
瑞萨技术支持	www.renesas.com/support

更新履历

版本号	日期	说明	
		页数	概述
1.00	2021 年 7 月 21 日	–	初版发布
1.01	2021 年 12 月 10 日	21, 22	修改了对看门狗定时器以及独立看门狗定时器的复位发生条件的表述。
		23	寄存器 RSTSR0 和 RSTSR1 显示复位源。
		30	烧写数据闪存为 4 / 8 /16 字节。
1.02	2023 年 3 月 29 日	5 – 10, 12	删除了调试信号上外加上拉电阻的相关内容。
1.03	2024 年 5 月 20 日	–	新增了 RA4E1 与 RA4E2 的内容。
1.04	2026 年 3 月 27 日	–	新增了 RA4L1 与 RA4C1 的内容。

注意

1. 本文档中电路、软件和其他相关信息的描述仅用于说明半导体产品的操作和应用示例。用户应对产品或系统设计中电路、软件和信息的纳入或任何其他用途承担全部责任。对于您或第三方因使用这些电路、软件或信息而引起的任何损失和损害，Renesas Electronics 不承担任何责任。
2. Renesas Electronics 特此声明，对于因使用本文档中所述的 Renesas Electronics 产品或技术信息（包括但不限于产品数据、图纸、图表、程序、算法和应用示例）而引起的侵权或与第三方有关的专利、版权或其他知识产权的任何其他索赔，概不承担任何责任和赔偿。
3. 对 Renesas Electronics 或其他公司的任何专利、版权或其他知识产权均不授予任何明示、暗示或其他形式的许可。
4. 您应负责确定需要从任何第三方获得哪些许可，并在需要时为合法进口、出口、制造、销售、使用、分销或以其他方式处置包含 Renesas Electronics 产品的任何产品获得此类许可。
5. 不得对 Renesas Electronics 产品的全部或部分进行更改、修改、复制或逆向工程。对于因更改、修改、复制或逆向工程而导致您或第三方蒙受的任何损失或损害，Renesas Electronics 不承担任何责任。
6. Renesas Electronics 产品根据以下两个质量等级进行分类：“标准”和“优质”。Renesas Electronics 每种产品的预期应用取决于产品的质量等级，具体如下所示。
   “标准”：计算机、办公设备、通信设备、测试和测量设备、视听设备、家用电器、机械工具、个人电子设备、工业机器人等
   “优质”：运输设备（汽车、火车、轮船等）；交通管制（交通信号灯）；大型通信设备；关键金融终端系统；安全控制设备等
   除非在 Renesas Electronics 数据手册或 Renesas Electronics 其他文档中明确指定为高可靠性产品或用于恶劣环境的产品，否则 Renesas Electronics 产品不适合或不授权用于可能对人类生命构成直接威胁或造成人身伤害（人造生命支持设备或系统；手术植入物等），或者可能造成严重的财产损失（空间系统、海底中继器、核动力控制系统、飞机控制系统、关键设备系统、军事装备等）的产品或系统。对于因使用任何与 Renesas Electronics 数据手册、用户手册或其他 Renesas Electronics 文档不一致的 Renesas Electronics 产品而引起的您或任何第三方所造成的任何损坏或损失，Renesas Electronics 不承担任何责任。
7. 没有任何半导体产品是绝对安全的。尽管 Renesas Electronics 的硬件或软件产品中可能实施了任何安全措施或功能，Renesas Electronics 对因任何漏洞或侵骇（包括但不限于以任何未经授权的方式访问或使用 Renesas Electronics 产品或使用 Renesas Electronics 产品的系统）而产生的任何后果概不负责。RENESAS ELECTRONICS 不担保或保证 RENESAS ELECTRONICS 产品或使用 RENESAS ELECTRONICS 产品创建的任何系统不会被破坏，或者可免于数据损坏、攻击、病毒、干扰、黑客攻击、数据丢失或失窃或其他安全入侵（“漏洞问题”）。RENESAS ELECTRONICS 不承担由任何漏洞问题引起的或与之相关的任何和所有责任或义务。此外，在适用法律允许的范围内，RENESAS ELECTRONICS 不对本文件和任何相关或附带的软件或硬件提供任何和所有明示或暗示的保证，包括但不限于对适销性或对特定用途的适用性的暗示保证。
8. 使用 Renesas Electronics 产品时，请参见最新的产品信息（数据手册、用户手册、应用笔记、可靠性手册中的“处理和使用半导体器件的一般说明”等），并确保使用条件符合 Renesas Electronics 在最大额定值、工作电源电压范围、散热特性和安装等方面的规定。对于因在超出上述规定范围的条件下使用 Renesas Electronics 产品而引起的任何失常、故障或事故，Renesas Electronics 不承担任何责任。
9. 尽管 Renesas Electronics 致力于提高 Renesas Electronics 产品的质量和可靠性，但半导体产品具有特定的特性，例如在特定速率下发生故障以及在某些使用条件下出现故障。除非在 Renesas Electronics 数据手册或 Renesas Electronics 其他文档中指定为高可靠性产品或用于恶劣环境的产品，否则 Renesas Electronics 的产品将不受抗辐射设计的约束。用户应负责采取安全措施，以防止人身伤害、火灾造成的伤害，和/或因 Renesas Electronics 产品发生故障或失常而对公众造成的危险，例如硬件和设备的安全设计，包括但不限于冗余、火控和故障预防、针对老化退化的适当处理或任何其他适当的措施。由于对微型计算机软件进行评估非常困难且无实操性，因此用户有责任评估自己生产的最终产品或系统的安全性。
10. 请联系 Renesas Electronics 销售办事处，以获取有关环境事宜的详细信息，例如每个 Renesas Electronics 产品的环境相容性。用户有责任认真、充分地研究有关纳入或使用受控物质的适用法律和法规（包括但不限于欧盟 RoHS 指令），并按照所有适用法律和法规使用 Renesas Electronics 产品。对于因您未遵守适用的法律和法规而造成的损坏或损失，Renesas Electronics 不承担任何责任。
11. Renesas Electronics 产品和技术不得被用于或纳入为任何适用的本国或外国法律、法规所禁止制造、使用或销售的产品或系统范围内。用户应遵守由对当事方或交易拥有管辖权的任何国家/地区的政府颁布和管理的任何可适用的出口控制法律和法规。
12. 应由Renesas Electronics 产品的购买方或分销商，或者对产品进行分发、处置或以其他方式出售或转让给第三方的任何其他当事方，负责将本文档中阐明的内容和条件提前通知前述第三方。
13. 未经 Renesas Electronics 事先书面同意，不得以任何形式全部或部分重印、再现或复制本文档。
14. 如果对本文档中包含的信息或 Renesas Electronics 产品有任何疑问，请联系 Renesas Electronics 销售办事处。

（注 1）本文档中的“Renesas Electronics”是指 Renesas Electronics Corporation，也包括其直接或间接控制的子公司。
（注 2）“Renesas Electronics 产品”是指 Renesas Electronics 开发或制造的任意产品。

（版本 5.0-1 2020 年 10 月）

公司总部

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

联系信息

有关产品、技术、文档最新版本或离您最近的销售办事处的更多信息，请访问：www.renesas.com/contact/。

商标

Renesas 和 Renesas 徽标是 Renesas Electronics Corporation 的商标。所有商标和注册商标都是各自所有者的财产。

© 2026 Renesas Electronics Corporation。保留所有权利。