



EZ-CUBE

具有编程功能的片上调试仿真器

本资料所记载的内容，均为本资料发行时的信息，瑞萨电子对于本资料所记载的产品或者规格可能会作改动，恕不另行通知。
请通过瑞萨电子(中国)的主页确认发布的最新信息。

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.

2. Renesas Electronics does not assume any liability for infringement of 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.

No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.

3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.

4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.

5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.

6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.

7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anticrime systems; safety equipment; and medical equipment not specifically designed for life support.

"Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.

8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.

9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system manufactured by you.

10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.

11. This document may not be 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, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majorityowned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

前言

衷心感谢您使用瑞萨电子（中国）有限公司的EZ-CUBE 仿真器。
EZ-CUBE 仿真器（YRCNEZCUBE01）是用于瑞萨电子产MCU 的仿真器。
这些手册的最新版能从本公司开发环境的主页（<http://www.cn.renesas.com/EZ-CUBE>）下载。

重要事项

在使用本仿真器前，请务必仔细阅读并理解用户手册。
务必保管好用户手册，在使用中如有不明之处，请再次阅读。

仿真器：

本资料中的仿真器是指瑞萨电子（中国）有限公司的EZ-CUBE 仿真器，不包含客户的用户系统和主机。

仿真器的使用目的：

本仿真器是支援使用瑞萨单片机进行系统开发的装置，从软件和硬件两方面支援系统开发，但是不保证生产线上的使用。

请遵循此使用目的正确使用本仿真器，坚决拒绝本目的以外的使用。

仿真器的使用对象：

本仿真器仅供仔细阅读并理解用户手册的对象使用。在使用本仿真器时，需要电子电路、逻辑电路和单片机的基本知识。

使用仿真器时：

1. 本仿真器是用于程序开发和评价阶段的开发支援装置。在批量生产已开发的程序时，请客户务必事先进行安装评价和试验等判断是否适用。

2. 对于因使用本仿真器而产生的用户开发结果，本公司不承担任何责任。

3. 本公司努力提供有关本仿真器的缺陷对策。但是，不保证任何情况都能提供。

4. 本仿真器是为实验室用于程序开发和评价而准备的产品。

5. 本公司无法预测所有存在的潜在危险而可能引发的诸多状况和错误使用。因此，此用户手册中和贴在本仿真器上的并非所有的警告。请客户正确安全使用本仿真器。

6. 对于因本产品的偶发故障或者误动作而引起的客户直接和间接的损害，本公司不承担责任。

关于废弃：

在废弃本仿真器时，请当作工业废弃物，根据法令规定进行处理。

使用限制：

本仿真器是作为开发支持工具而开发的产品。因此，请不要用作嵌入式设备，也不要用于以下所示的开发用途：

1. 运输、交通车辆
2. 医疗（用于涉及生命安全的装置）
3. 航空宇宙
4. 原子能控制
5. 海底中继器
6. 军事产品、大规模杀伤性武器

关于产品的更改：

本公司采取不断改良本仿真器的设计和性能的方针。因此，在更改规格、设计和用户手册时，恕不另行通知。

关于权利:

1. 对于因使用本资料记载的信息、产品或者电路而引起的损害或者专利权等其他权利的侵犯, 本公司不承担任何责任。
2. 本资料不许诺第三者或者本公司的专利权以及其他权利的实施权。
3. 本用户手册和本仿真器享有著作权保护, 所有权利归属本公司。事先未经本公司的书面许可, 不得翻印、复制和转载本用户手册的部分或者全部内容。

关于图:

本用户手册的部分图可能和实物有差异。

术语

本手册使用的术语的含义请参照下表。

项目	说明
EZ-CUBE	EZ-CUBE 的通用名称
目标设备	用于仿真的设备
目标系统	用于调试的系统(用户创建的系统) 包括用户建立的软件和硬件。
固件	嵌入设备用来控制 EZ-CUBE 的程序
CS+	集成开发环境CS+
RFP	Renesas Flash Programmer, 用于进行 Flash 编程的 GUI 软件

注意事项

关于异常工作:

如果因外来噪声等干扰而引起仿真器工作异常, 请按以下步骤处理:

1. 结束仿真调试程序, 切断仿真器和用户系统的电源。
2. 在经过10 秒以后再次接通电源, 启动仿真调试程序。

操作本产品的注意事项

瑞萨电子(中国)有限公司对于用户未遵照用户手册进行的非正常及非惯例使用造成的产品失效和其他相关损失不承担任何责任。

目 录

第 1 章 概述	8
1.1 特点	8
1.2 使用EZ-CUBE前的注意事项	8
1.3 硬件规格	9
1.4 固件更新	9
1.5 标准配置	10
第 2 章 硬件名称和功能	11
2.1 EZ-CUBE各部件名称和功能	11
第 3 章 如何在RL78 微控制器上使用EZ-CUBE	13
3.1 目标系统设计	14
3.1.1 引脚配置	14
3.1.2 电路连接举例	15
3.1.3 reset引脚的连接	17
3.2 片上调试	18
3.2.1 调试功能	18
3.2.2 保留用户资源和设置安全ID	18
3.2.3 调试时的注意事项	22
3.3 Flash 编程	24
3.3.1 编程功能的规格定义	24
3.3.2 Flash编程时的注意事项	24
第 4 章 如何在 78K0R微控制器上使用EZ-CUBE	25
4.1 目标系统设计	26
4.1.1 引脚配置	27
4.1.2 电路连接示例	28
4.1.3 reset 引脚的连接	29
4.2 片上调试	30
4.2.1 调试功能	30
4.2.2 用户资源的保留, 设置安全ID 以及片上调试选项字节	30
4.2.3 调试的注意事项	34
4.3 Flash 编程	36
4.3.1 编程功能的规格定义	36
4.3.2 Flash 编程的注意事项	36
第 5 章 如何在 78K0 微控制器上使用EZ-CUBE	37
5.1 目标系统设计	38
5.1.1 引脚配置	38
5.1.2 电路连接示例	39
5.1.3 reset引脚的连接	42
5.1.4 目标系统设计的注意事项	43
5.1.5 时钟设置	43

5.2 片上调试	45
5.2.1 调试功能	45
5.2.2 用户资源的保留和安全ID的设置	45
5.2.3 调试的注意事项	49
5.3 Flash编程	52
5.3.1 编程功能的规格定义	52
5.3.2 Flash编程的注意事项	52
第6章 如何在RX微控制器上使用EZ-CUBE	53
6.1 目标系统设计	54
6.1.1 引脚配置	54
6.1.2 电路连接举例	55
6.1.3 连接时的注意事项	58
6.2 片上调试	59
6.2.1 调试功能	59
6.2.2 调试时的注意事项	59
6.3 Flash 编程	61
6.3.1 编程功能的规格定义	61
6.3.2 Flash编程时的注意事项	61
第7章 如何在V850 微控制器上使用EZ-CUBE	62
7.1 目标系统设计	63
7.1.1 引脚配置	63
7.1.2 电路连接举例	64
7.1.3 reset引脚的连接	66
7.2 片上调试	67
7.2.1 调试功能	67
7.2.2 保留用户资源和设置安全ID	67
7.2.3 调试时的注意事项	71
7.3 Flash 编程	73
7.3.1 编程功能的规格定义	73
7.3.2 Flash编程时的注意事项	73

第 1 章 概述

EZ-CUBE 是具有 flash 存储器编程功能的片上调试仿真器，可以用于调试程序或将程序烧写到内置 Flash 存储器微控制器中。

1.1 特点

- 片上调试
可以调试已连接到目标板上的微控制器。
- Flash编程
可以将程序写入控制器的内置flash存储器中。
- USB 连接
可以通过USB 2.0接口和主机连接。
工作时使用USB供电，无需外部电源。
- 支持产品范围广
EZ-CUBE能支持Renesas Electronics 8位到32位的内置Flash存储器微控制器。
 - RL78微控制器
 - 78K0R微控制器
 - 78K0 微控制器
 - RX微控制器
 - V850 微控制器

1.2 使用 EZ-CUBE 前的注意事项

第 1 章和第 2 章是有关 EZ-CUBE 的概要和基本规格说明，以下的章节针对不同的目标设备及其用途分别进行描述。

为了有效地使用本手册，您可以参照下表。

表 1-1. 用途对应章节

目标设备	用途	参照章节
RL78		第 3 章 如何在 RL78 系列微控制器上使用 EZ-CUBE
	目标系统设计	3.1 目标系统设计
	片上调试	3.2 片上调试
	Flash 编程	3.3 Flash 编程
78K0R		第 4 章 如何在 78K0R 系列微控制器上使用 EZ-CUBE
	目标系统设计	4.1 目标系统设计
	片上调试	4.2 片上调试
	Flash 编程	4.3 Flash 编程
78K0		第 5 章 如何在 78K0 系列微控制器上使用 EZ-CUBE
	目标系统设计	5.1 目标系统设计
	片上调试	5.2 片上调试
	Flash 编程	5.3 Flash 编程
RX		第 6 章 如何在 RX 系列微控制器上使用 EZ-CUBE

	目标系统设计	6.1 目标系统设计
	片上调试	6.2 片上调试
	Flash 编程	6.3 Flash 编程
V850		第7章 如何在 V850 系列微控制器上使用 EZ-CUBE
	目标系统设计	7.1 目标系统设计
	片上调试	7.2 片上调试
	Flash 编程	7.3 Flash 编程

1.3 硬件规格

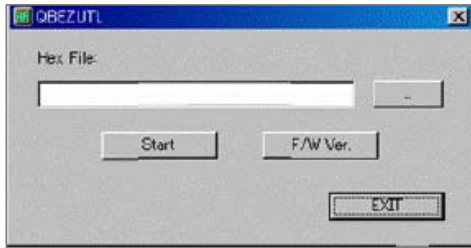
本节将说明 EZ-CUBE 的硬件规格。
相应片上调试和 Flash 存储器编程功能的相关规格在随后的章节中介绍。

表 1-2. 硬件配置

分类	项目	规格
EZ-CUBE	工作电压	USB 接口供电 (5 V)
	工作环境	温度: $\pm 0 \sim +40^{\circ}\text{C}$ 湿度: 40 - 80% RH (无结露)
	保存	温度: $-15 \sim +60^{\circ}\text{C}$ 湿度: 40 - 80% RH (无结露)
	外形尺寸	60 × 36 × 13mm
	重量	约 40 g
主机界面	目标主机	兼容 IBM PC/AT/MT
	目标操作系统	Windows XP(32bit), Vista, Windows 7, Windows 8
	USB	2.0
	USB 连接线	1.0 m
	消耗电流	最大 500 mA
目标界面	目标线缆	8 引脚线缆
	供电电压	5 V \pm 0.3V (典型值)
	供电电流	最大 100 mA
	电压范围	2.7 - 5.5 V

1.4 固件更新

- (1) **驱动程序安装**
如果没有安装用于 EZ-CUBE 的 USB 驱动程序, 请先安装它。(CD 光盘)
- (2) **USB 连接线**
将 EZ-CUBE 连接到主机, **不要将 EZ-CUBE 与目标板相连**。连接成功后状态指示灯显示为红色。
- (3) **EZ-CUBE 固件更新程序的启动**
启动 EZ-CUBE 固件更新程序【QBEZUTL.exe】, 如下所示。
QBEZUTL.exe 版本为 V1.14 及以上版本

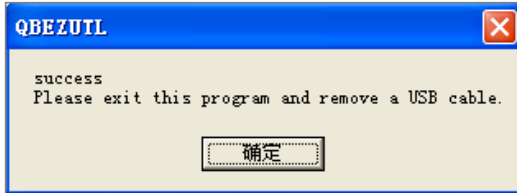


(4) **选择固件**

点击菜单栏中的[...]按钮，选择固件 (*.hex)。

(5) **更新固件**

点击[Start]按钮，开始更新固件。固件更新完成，显示以下对话框。



(6) **EZ-CUBE 固件更新程序的关闭**

固件更新完成，点击[EXIT]按钮，退出固件更新程序。

(7) **拔除 USB 线缆**

从 EZ-CUBE 或主机上拔除 USB 线缆。

1.5 标准配置

EZ-CUBE 本体一个

USB 电缆一根

8pin 目标线缆一根

软件光盘一张(包含: 安装软件、用户手册电子版)。

第 2 章 硬件名称和功能

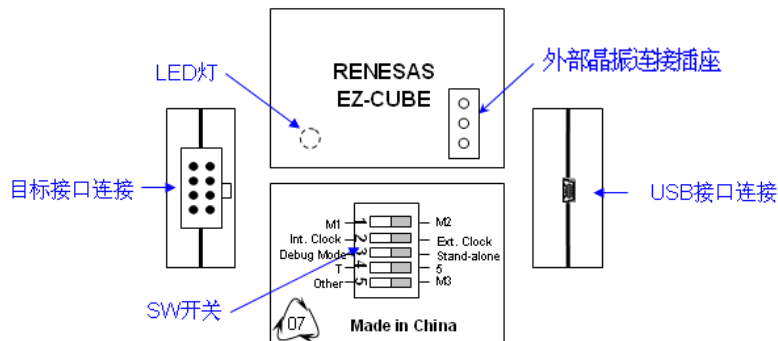
本章介绍 EZ-CUBE 的部件名称和功能及其附件。

本章所使用的部件名称将在后续的章节继续使用。本章介绍各种功能的概述。阅读本章后，读者将对 EZ-CUBE 有一个基本的认识，有利于理解接下来的章节。阅读本章内容同时，可以用于检查硬件设备是否存在缺损。

2.1 EZ-CUBE 各部件名称和功能

图 2-1 标明了 EZ-CUBE 各部件名称。具体功能请参照 (1) ~ (9)。

图 2-1. EZ-CUBE 各部件名称



(1) **SW-1 开关**

此开关取决于目标设备，详情请参见各系列章节。出厂时默认设定为“M2”。

(2) **SW-2 开关**

此开关用来设定目标系统振荡器，表 2-1 说明了具体含义，出厂时默认设定为“Int. Clock”。

表 2-1. 振荡器选择开关设定

设定	说明
Int. Clock	当选择“Int. Clock”，时钟以 8MHz 固定频率提供给目标板。（出厂设定）
Ext. Clock	如需要改变频率，必须选择“Ext. Clock”（在这种情况下，需使用外部晶振连接）。

(3) **SW-3 开关**

此开关选择目标系统的运行方式，表 2-2 说明了具体含义，出厂时默认设定为“Debug Mode”。

表 2-2. 运行模式选择开关设定

设定	说明
Debug Mode	当选择“Debug Mode”，运行命令从 GUI 调试器发出时，用户程序将运行。（出厂设定）
Stand Alone	当选择“Stand Alone”，EZ-CUBE 连接到目标板，产生复位，用户程序将自动运行。

(4) **SW-4 开关**

此开关用来设定目标系统电源模式，表 2-3 说明了具体含义，出厂时默认设定为“5”。

表 2-3. 电源选择开关设定

设定	说明
5	EZ-CUBE 向目标系统提供+5±0.3 V 电源 ^注 。（出产设定）
T	使用目标系统的电源，电压范围+2.7V~+5.5V。 即使在这种情况下，EZ-CUBE的VDD引脚线也必须与目标板的VDD连接，以调整目标板和EZ-CUBE之间的电压电平。

注 最大允许电流 100mA，所以请勿将 EZ-CUBE 连接到电流很大的目标系统上。在 EZ-CUBE 与主机连通后，主机始终保持向 EZ-CUBE 供电。

(5) **SW-5 开关**

此开关取决于目标设备，详情请参见各系列章节。出厂时默认设定为“Other”。

(6) **USB 连接器**

通过 USB 连接线，和主机进行连接的连接器。
使用 USB 2.0 对应的 mini-B 连接器。

(7) **目标连接器**

通过 2*4pin 的 8 引脚线缆，和目标系统进行连接的连接器。

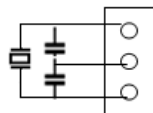
(8) **外部晶振连接插座（仅限 78K0 和 RX）**

SW-2 开关选为“Ext. Clock”。

在 EZ-CUBE 上的外部晶振位置安装振荡器或者振荡电路。（在调试器的配置对话框中，选择外部晶振为主时钟）。详细信息，请参考 CS+的用户手册。

安装振荡器，参考图 2-2。

图 2-2 在 EZ-CUBE 的外接晶振上安装振荡电路



(9) **模式灯**

根据硬件和软件的变化，模式灯会有不同的显示状态，请参照表 2-4。

表 2-4. 模式灯说明

模式灯颜色	显示状态	说明	
		USB 连接状态	软件工作状态
-	熄灭	未连接	未启动
Red	点亮	连接	上电或用户程序调试时中断
Green	点亮		用户程序调试正在运行

第3章 如何在 RL78 微控制器上使用 EZ-CUBE

本章描述了如何使用 EZ-CUBE 在 RL78 微控制器上进行片上调试和 Flash 编程。

片上调试是指通过芯片内置的调试功能对已经装载在目标系统上的微控制器进行调试的方法。由于这种调试方法是对板上的目标设备进行操作的，所以非常适合现场调试。

Flash 编程是将程序写入设备内置的 Flash 存储器的方法。可以在板上对设备进行擦除、写入和校验。

请先将 EZ-CUBE 的固件更新为用于 RL78，具体请参照 (1) - (3)。详细内容请参见 **1.4 固件更新**。

- (1) 使用 USB 缆连接 EZ-CUBE 和 PC。**不要将 EZ-CUBE 与目标板相连。**
- (2) 启动[QBEZUTL.exe]，并指定“RL78G10_OCD_FW.hex 或 RL78_OCD_FW (except G10).hex”固件文件。
- (3) 点击“Start”按钮，然后指定用于 RL78 固件下载到 EZ-CUBE。

如果首次用 RL78 微控制器作为目标设备来使用 EZ-CUBE，那么请仔细阅读以下内容。

3.1 目标系统设计

要用EZ-CUBE和目标系统进行通信，必须在目标系统上安装对应的通信电路。本节描述了相关电路的设计以及接口的安装。

3.2 片上调试

本节描述了用EZ-CUBE进行片上调试的系统配置和启动方法。

3.3 Flash 编程

本节描述了用 EZ-CUBE 进行 Flash 编程的系统配置和启动方法。

支持 RL78 MCU 列表

表3-1 显示了EZ-CUBE支持的RL78系列MCU列表及固件。

表 3-1 EZ-CUBE 支持RL78 MCU列表及固件

项目	内容	固件
支持MCU	RL78/G10	RL78G10_OCD_FW.hex
	RL78/G12,G13,G14,I1A	RL78_OCD_FW (except G10).hex
支持的操作模式	UART模式	

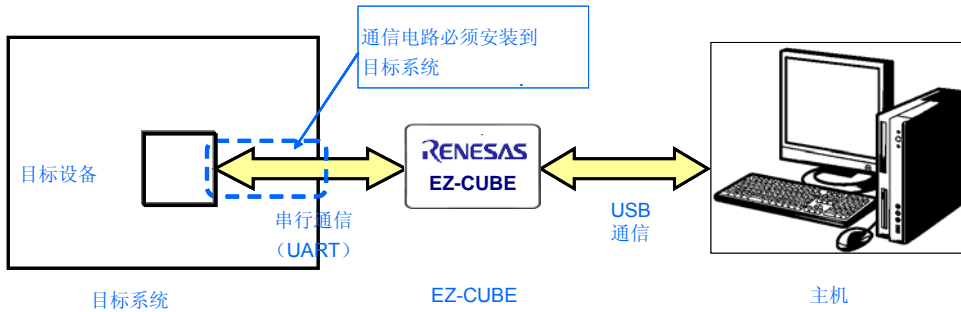
3.1 目标系统设计

为了实现片上调试和Flash编程功能，需要设计目标系统的电路，本节描述了相关的电路设计。

图3-1为EZ-CUBE通信的整体接口框图。如图所示，EZ-CUBE与目标系统的目标设备之间进行串行通讯。为了实现通讯，目标系统上必须安装用于通信的电路。请参考本节的内容设计合适的电路。

这里用于串行通信的引脚基本上和使用Flash编程器（例如EZ-CUBE）时使用的串行通信引脚相同，但是有一些设备并不支持这些引脚。

图3-1. 通信接口概况



3.1.1 引脚配置

本节将说明EZ-CUBE和目标系统之间使用的接口信号。表3-2中列出了引脚的配置情况。表3-3说明了各个引脚的功能。

表 3-2. 引脚配置

引脚编号	引脚名称 ^注
1	GND
2	RESET_IN
3	Vdd
4	FLMD0
5	CLK
6	RxD.
7	RESET_OUT
8	TOOLO

注 EZ-CUBE 内的信号名称。

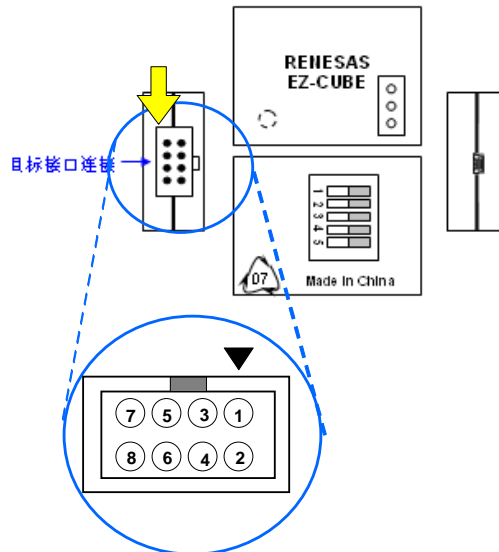


表 3-3. 引脚功能

引脚名称	IN/OUT*	描述
RESET_IN	IN	该引脚用于从目标系统输入 reset 信号
RESET_OUT	OUT	该引脚用于向目标系统输出 reset 信号
FLMD0	OUT	该引脚用于设定目标设备进入调试模式或编程模式
TOOL0	OUT	该引脚用于向目标设备发送命令/数据

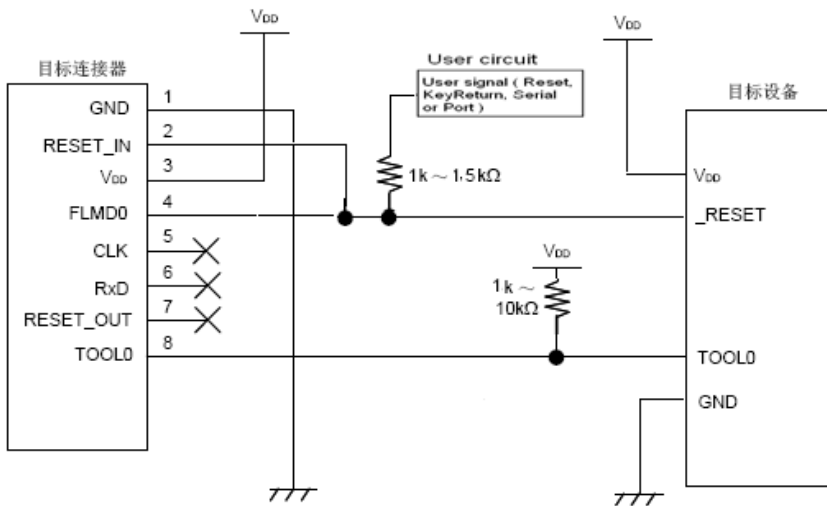
注 以 EZ-CUBE 为参照基准。

3.1.2 电路连接举例

目标系统上的电路设计会根据接口信号而有所不同。接口的引脚名称可能会因目标设备有所不同。详细信息请参考目标设备的用户手册。参照表 3-2 和相关的电路连接示例。

注意事项 在电路连接示例中出现的常数值为参考值。如果进行 flash 编程旨在量产，请彻底评估目标设备的规格是否满足要求。

图 3-2. RL78/G10, RL78/G12(20pin,24pin)电路连接示例



EZ-CUBE 开关设置 RL78/G10, G12 (20pin,24pin)

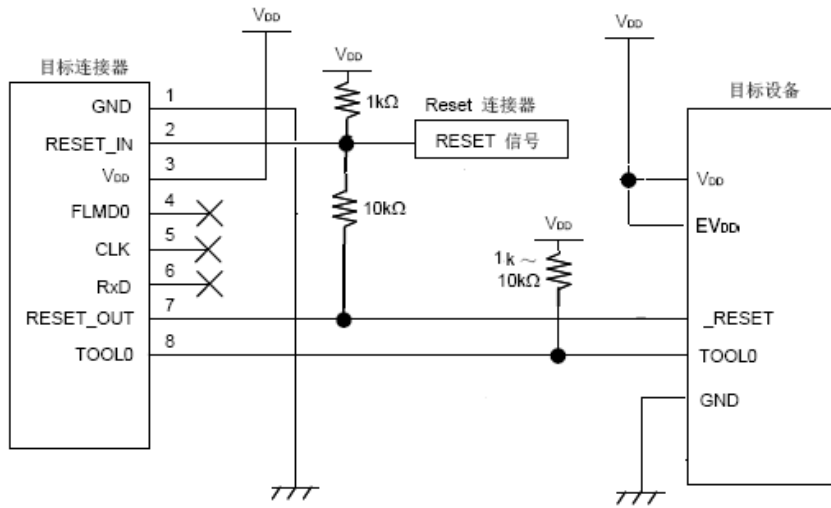
- SW-1 开关: 请选择“M2”.
- SW-2 开关: 请选择“ Int. Clock ”.
- SW-3 开关: 请选择“Debug Mode”.
- SW-4 开关: 请根据实际目标设备使用选择.
- SW-5 开关: 请选择“M3 ”.

CAUTION

注意目标系统的电源提供:

- 1. 在USB线缆处于连接状态时，请勿改变开关的设置。
- 2. 最大允许电流100mA，所以请勿将EZ-CUBE连接到电流很大的目标系统上。在EZ-CUBE与主机连通后，主机始终保持向EZ-CUBE供电。

图 3-3. RL78/G12(30pin), RL78/G13, RL78/G14, RL78/I1A 电路连接示例



EZ-CUBE 开关设置 RL78/G12 (30pin), G13,G14, I1A

- SW-1 开关: 请选择“M2”.
- SW-2 开关: 请选择“ Int. Clock ”.
- SW-3 开关: 请选择“Debug Mode”.
- SW-4 开关: 请根据实际目标设备使用选择.
- SW-5 开关: 请选择“Other ”.

! CAUTION

注意目标系统的电源提供:



1. 在USB线缆处于连接状态时, 请勿改变开关的设置.
2. 最大允许电流100mA, 所以请勿将EZ-CUBE连接到电流很大的目标系统上。在EZ-CUBE与主机连通后, 主机始终保持向EZ-CUBE供电.

3.1.3 reset 引脚的连接

本节描述了电路中 reset 引脚的连接，此处必须特别注意，电路连接示例见前面的章节。

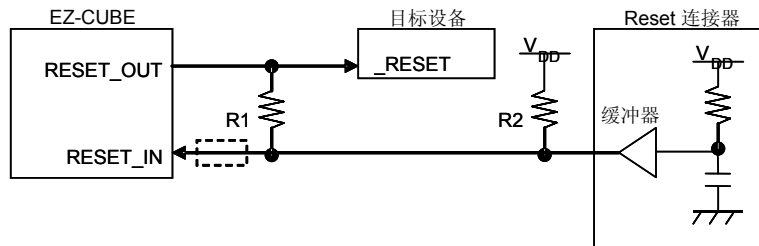
片上调试时，来自目标系统的 reset 信号输入 EZ-CUBE，被屏蔽后，再输入到目标设备。于是 reset 信号的连接根据是否连接 EZ-CUBE 而有所不同。

Flash 编程时，电路的设计必须保证目标系统的 reset 信号和 EZ-CUBE 的 reset 信号不冲突。

推荐通过电阻自动切换 reset 信号。图 3-4 说明了 3.1.2 电路连接示例中描述的 reset 引脚连接情况。

这种连接的设计是假定目标系统的复位电路包含有 N 沟道开漏缓冲器（输出阻抗：100 欧姆或更小）。当 EZ-CUBE 的 RESET_IN/OUT 逻辑反转时，VDD 或 GND 的电平可能会不稳定，所以请注意备注中所描述的条件。

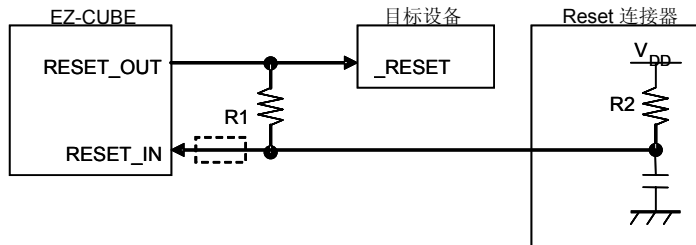
图 3-4. 包含缓冲器的复位电路的连接



备注 需确保 R1 的阻值至少是 R2 阻值的十倍，R1 为 10 kΩ 或更大。
如果复位电路的缓冲器是 CMOS 输出，则无需上拉电阻 R2。
当只用于 Flash 编程时，可以略去虚线框内的电路。

图 3-5 所示的电路连接情形是目标系统的复位电路不包含缓冲器，并且 reset 信号仅通过电阻或电容产生。设计电路时注意备注中描述的条件。

图 3-5. 不包含缓冲器的复位电路的连接



备注 需确保 R1 的阻值至少是 R2 阻值的十倍，R1 为 10 kΩ 或更大。
当只用于 Flash 编程时，可以略去虚线框内的电路。

3.2 片上调试

本节描述了使用 EZ-CUBE 进行片上调试的系统配置，启动/关闭顺序和调试中的若干注意事项。

3.2.1 调试功能

表 3-4 列出了以 RL78 微控制器作为目标设备，进行调试时的调试功能。

表 3-4. 调试功能

功能	规格定义
目标 MCU	RL78系列 RL78/G10, G12, G13, G14, I1A
安全	10 字节 ID 码验证
下载	支持
执行	全速执行，忽略断点执行，单步执行，下一步执行，运行至此，重启
硬件断点	1 个
软件断点	多个
RAM 监视	支持
调试所用的功能引脚	TOOL0

3.2.2 保留用户资源和设置安全 ID

用户必须做好下面的保留以保证 EZ-CUBE 和目标设备之间的通信以及实现每一个调试功能。使用 Renesas ELECTRONICS 的编译器，这些都可以通过 CS+ 选项设置。请参考下面的描述并设置，详细内容请查阅 CS+ 相关用户手册。

(a) 设置安全 ID 和片上调试选项字节

这个区域用来存放安全设置以防无认证的人读取 Flash 存储器。安全 ID 在内部 flash memory 的 0xC4 到 0xCD。安全 ID 功能就像启动调试器的密码。只有在调试器启动时输入的安全 ID 和存放在这个区域的安全 ID 匹配时，调试器才能正常启动。如果 ID 码不匹配，调试器根据片上调试选项字节区域设置的值对目标设备进行操作（参考表 3-6）。

[如何设置安全 ID]

有两种方式设置选项字节。如果方式（1）和（2）都设置，方式（2）优先。

- (1) 在用户程序地址 0xC4 到 0xCD 区域设置安全 ID。
- (2) 构建工具通用选项（CS+）中设置安全 ID。

(1) 用户程序在地址 0xC4 到 0xCD 设置

举例，安全 ID 设置为“0123456789ABCDEF1234”（不区分大小写）

表 3-5 安全 ID

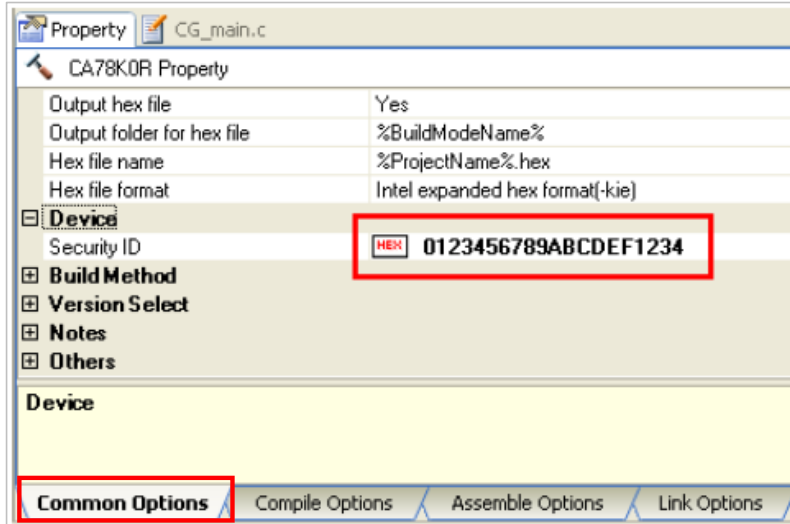
Address	Value
0xC4	0x01
0xC5	0x23
0xC6	0x45
0xC7	0x67
0xC8	0x89

0xC9	0xAB
0xCA	0xCD
0xCB	0xEF
0xCC	0x12
0xCD	0x34

(2) CS+设置

在通用选项 “device”中设置安全 ID 如图 3-6。

图 3-6. 安全ID设置样例



[如何在调试器启动进行安全 ID 验证]

当调试器连接到设备的安全 ID，必须由调试工具属性的连接设置中指定安全 ID。（默认安全 ID 在构建工具属性设置。）

(b) 设置片上调试选项字节

这个区域用来存放安全设置以防无认证的人读取 Flash 存储器。调试器根据设置的值对目标设备进行操作，如下表所示。

表 3-6. 片上调试选项字节设置和操作

设置值	描述	备注
0x04	不能进行调试	这个设置只在Flash编程或自编程时有效
0x85	不管安全ID码认证失败多少次都不会擦除片上Flash存储器	-
0x84	如果安全ID码认证失败擦除所有片上Flash存储器区域	-
其它	禁止设置	-

[如何设置]

有两种方式设置选项字节。如果方式（1）和（2）都设置，方式（2）优先。

（1）在用户程序地址0xC3设置在线调试选项字节。

（2）构建工具通用选项（CS+）中设置在线调试选项字节。

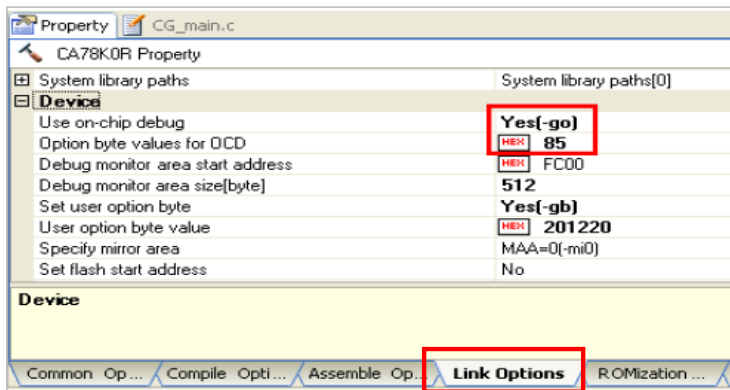
（1）在用户程序地址0xC3设置在线调试选项字节

在用户程序地址0xC3设置在线调试选项字节。

（2）构建工具通用选项（CS+）中设置在线调试选项字节 如图

3-7中显示了在CS+中的 [Link Options]选项卡设置。例 把控制值设置为0x85。

图 3-7. 片上调试选项字节样例



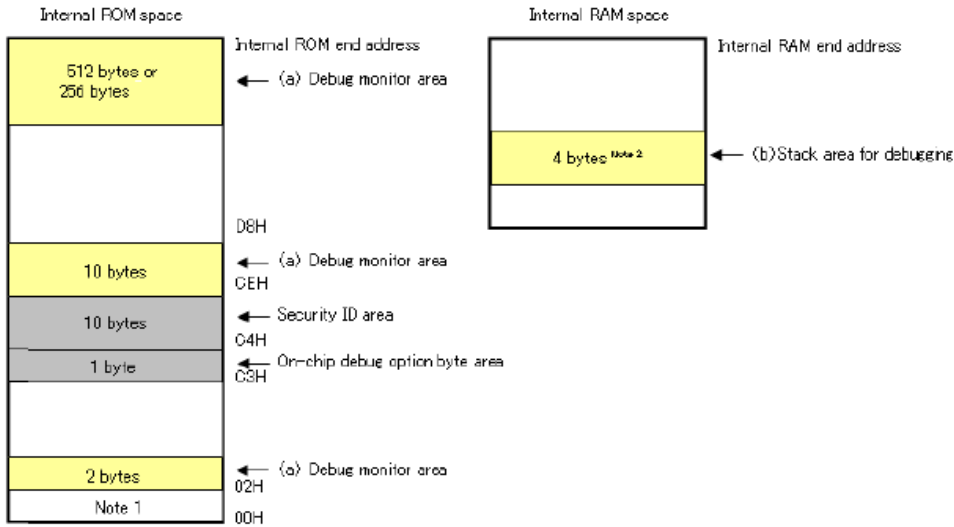
(c) 调试用区域的保留

图3-8中的阴影部分是存放调试监控程序的保留区域，所以用户程序或数据不能存放在这些空间。这些空间必须保证不被用户程序使用。

另外，这些区域也不能被用户程序重写。

调试保留区域内容说明，参见（1）和（2）。

图 3-8. 分配给调试监控程序的内存空间



注 1. 调试过程中，复位向量被重写，指向监控程序区。
2. 当执行自编程时，将是 12 字节。

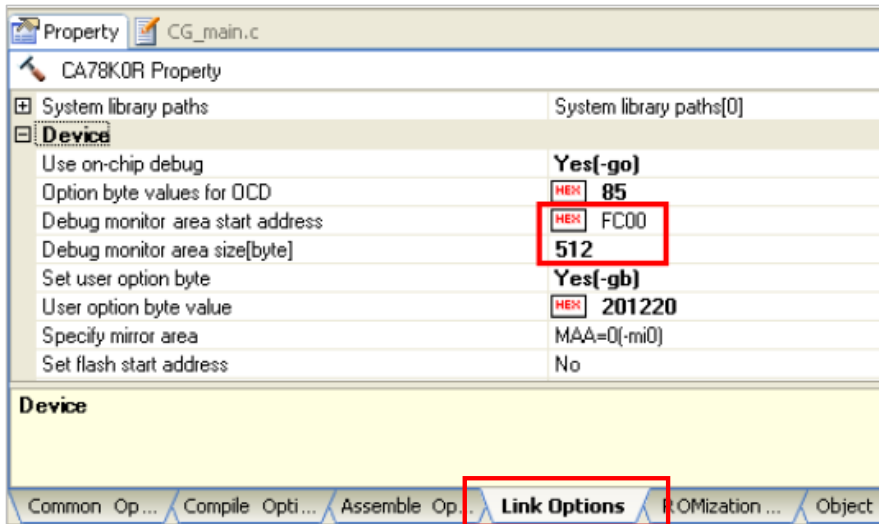
(1) 调试监控程序的保留区域

这是一个放置调试监控程序的区域。监控程序为调试通信接口和 CPU 的运行或者断点做初始化操作。用户程序或数据不能存放在片上调试选项字节周边的 22 个字节的区域以及内部 ROM 底端的 512 字节区域内。另外，复位向量会被重写并指向放置监控程序的地址。

[如何预留区域]

如果用户程序不使用这个区域，可以不进行预留。
为了避免在调试时出现问题，建议提前预留区域。
如图3-9 所示，使用 CS+，在[Link Options]选项卡的“Device”处进行设置。

图 3-9. 预留调试监测区域

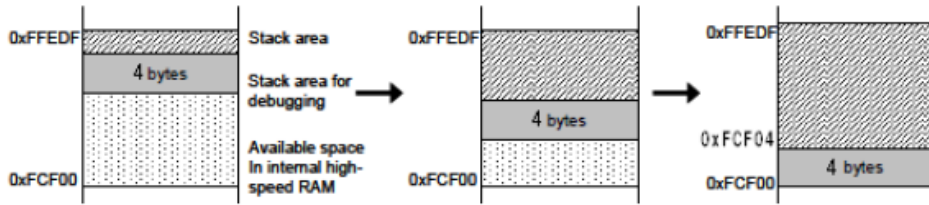


(2) 调试用堆栈区域的保留

这个区域需要 4 个字节作为调试的堆栈区域。因为这个区域分配在紧邻堆栈区域之前，所以这个区域的地址根据堆栈的增加和减少而有所改变。就是说，在使用的堆栈区域之外有 4 个额外的字节。

图 3-10 说明了堆栈区域增加时的情况，图中的内部高速 RAM 从 0xFCF00 开始。

图 3-10. 调试所用的堆栈地址的变化



[如何预留区域]

通过在用户程序使用堆栈之外再定义 4 个字节来设置堆栈指针。确保堆栈指针不会超过内部高速 RAM 的起始地址。

备注 关于如何预留自编程序的堆栈区域请参考自编程序手册。

3.2.3 调试时的注意事项

本节介绍使用 RL78 微控制器的片上调试功能时需要注意的事项。

请仔细阅读以下事项以正确使用 EZ-CUBE。

(1) 处理用于调试的设备

不要在批量生产的产品上安装曾用于调试的设备，因为 Flash 存储器在调试过程中被多次重写，Flash 存储器的重写次数无法保证。另外，不要把调试监控程序写入批量生产的产品。

(2) Flash 自编程

如果存放调试监控程序的区域被 Flash 自编程重写，调试器就不能正常操作。这个注意事项对这个区域的引导交换操作也适用。

(3) 复位后的操作

在外部复位或内部复位后，监控程序会进行调试初始化处理。因此，从产生复位到用户程序执行的时间和实际设备的操作有所不同。

(4) 调试后检查设备的运行

设备下载模块文件进行在线调试后，在没有 EZ-CUBE 时不能进行设备运行的检查。调试的设备包含片上调试的特殊程序，所以它与实际操作不同。

(5) 片上调试时功耗

设备的片上调试电路在片上调试时运行。

因此增加器件的功耗。

评估器件的功耗时，请不要连接调试器。

- (6) 片上调试选项字节设置（地址 C3H）
片上调试选项字节设置被调试器任意改写。
- (7) 操作电压不能写入 flash memory
如果片上调试时进行了如下步骤，调试器会输出一个错误和忽略该操作。因为这些操作包括了 Flash memory 的改写。
- <1> 内部 flash memory 写入
 - <2> 设置或取消软件断点
 - <3> 启动执行软件断点
 - <4> 软件断点进行单步执行
 - <5> 执行 Step-over, Return Out
 - <6> Come Here
 - <7> 指定、更改或取消硬件断点
 - <8> 内部复位
 - <9> 外围断点
- (8) 待机功能和 Break 功能之前的关系
Break 是 CPU 中断功能。在调试功能被打破时待机模式被释放。
- 停止用户程序的执行
 - 待机指令单步执行（执行指令后停止用户程序）
 - 伪实时 RAM 监视功能（break 被打破）
 - 伪动态 memory 修改
 - 执行用户程序断点设置
- (9) 使用 Step-in 的注意事项（单步执行）
当代码单步执行时，一些 SFR（特殊功能寄存器）的值可能会保持不变。如果代码单步执行时，一些 SFR（特殊功能寄存器）的值保持不变，单片机操作一直执行指令，而不是执行这些步骤。
代码单步：用户程序中的指令逐条执行
代码联系：用户程序从当前 PC 指针开始执行
- (10) flash memory CRC 功能仿真
请使用 IEUCBE 仿真告诉 CRC 操作。

3.3 Flash 编程

本节将介绍使用 EZ-CUBE 对 RL78 微控制器的 Flash 进行编程时的系统配置以及启动/关闭顺序。

3.3.1 编程功能的规格定义

表3-7. 编程功能的规格定义

功能	规格定义说明
主机接口	USB 2.0
目标接口	UART
目标系统电压	2.7到5.5 V (根据目标设备而定)
时钟源	可以提供8 MHz时钟信号 可以使用安装在目标系统上的时钟
电源	5 V \pm 0.3V (最大电流: 100 mA)
获取设备特定信息	使用RENEASAS ELECTRONICS的RL78参数文件
编程软件	RFP [®] (Renesas Flash Programmer)
安全标志的设置	有
独立操作	无 (必须连接主机)

注: 详情请见 RFP 用户手册

3.3.2 Flash 编程时的注意事项

本节将介绍进行 Flash 编程时需要注意的事项。请务必仔细阅读本节内容以正确使用 EZ-CUBE。

- 为提高写入的编程的质量, 请在使用 EZ-CUBE 之前了解以下内容并队列出的各项进行校验和评估。
 - 务必按照设备及 EZ-CUBE 的用户手册中给出的方法进行目标系统的电路的设计。
 - 务必按照设备、RFP 及 EZ-CUBE 各自的用户手册中介绍的方法进行操作。
 - 务必向目标系统提供稳定的电源。

第 4 章 如何在 78K0R 微控制器上使用 EZ-CUBE

本章描述了如何使用 EZ-CUBE 对 78K0R 微控制器进行片上调试和 Flash 编程。

片上调试是指通过芯片内置的调试功能对已经装载在目标系统上的微控制器进行调试的方法。由于这种调试方法是通过板上的目标设备实行的，所以非常适合现场调试。

Flash 编程是将程序写入设备内置的 Flash 存储器中的方法。可以在板上对设备进行擦除，写入和校验。

请先将 EZ-CUBE 的固件更新为用于 78K0R，具体请参照 (1) - (3)。详细内容请参见 **1.4 固件更新**。

- (1) 使用 USB 缆连接 EZ-CUBE 和 PC，**不要将 EZ-CUBE 与目标板相连**。
- (2) 启动【QBEZUTL.exe】，并指定“78K0R_OCD_FW.hex”固件文件。
- (3) 点击[Start]按钮，然后指定用于 78K0R 固件下载到 EZ-CUBE。

如果首次用 78K0R 微控制器作为目标设备来使用 EZ-CUBE，那么请仔细阅读以下内容。

4.1 目标系统设计

要用 EZ-CUBE 和目标系统进行通信，必须在目标系统上安装对应的通信电路。本节描述了相关电路的设计和连接插头的安装。

4.2 片上调试

本节描述了用 EZ-CUBE 进行片上调试的系统配置和启动方法。

4.3 Flash 编程

本节描述了用 EZ-CUBE 进行 Flash 编程的系统配置和启动方法。

支持 78K0R MCU 列表

表4-1 显示了EZ-CUBE支持的78K0R系列MCU列表及固件。

表 4-1 EZ-CUBE 支持78K0R MCU列表及固件

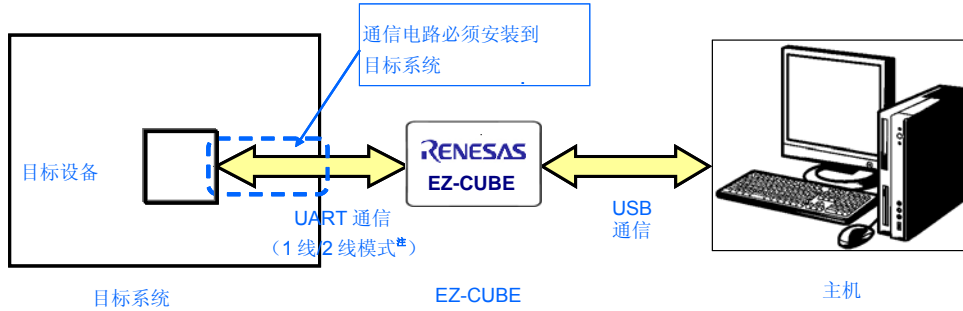
项目	内容	固件
支持MCU	78K0R/Kx3, Lx3	78K0R_OCD_FW.hex
支持的操作模式	UART模式	

4.1 目标系统设计

本节描述了片上调试和 Flash 编程所需目标系统的电路设计。

图 4-1 显示了 EZ-CUBE 通信的整体接口框图。如图所示，EZ-CUBE 和目标系统的目标设备之间进行串行通讯。为了通讯，目标系统上必须安装用于通信的电路。请参考本节的内容设计合适的电路。

图 4-1. 通信接口框图



注 单线模式：使用TOOL0引脚进行单线UART通信
 双线模式：使用TOOL0和TOOL1引脚进行双线UART通信

表 4-2. 单线模式和双线模式的不同之处

通信模式	Flash编程功能	调试功能
单线模式	可用	调试时所用的用户资源 内部ROM:1036字节 内部RAM:6字节(堆栈)
双线模式	可用	调试时所用的用户资源 内部ROM:100字节 内部RAM:6字节(堆栈)

4.1.1 引脚配置

本节描述了 EZ-CUBE 和目标系统之间使用的接口信号。表 4-3 列出了使用 8 引脚线缆时的引脚配置。表 4-4 描述了每个引脚的功能。在调试和编程的时候，引脚配置有很大区别，所以请根据后面的章节中描述的电路连接例题设计合适的电路。

表 4-3. 引脚配置

引脚编号	引脚名称 ^注
1	GND
2	RESET_IN
3	Vdd
4	FLMD0
5	CLK
6	RxD.
7	RESET_OUT
8	TxD

注 EZ-CUBE 内的信号名称。

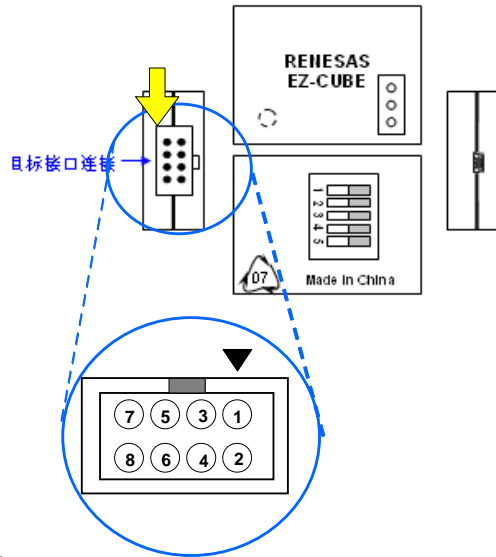


表 4-4. 引脚功能

引脚名称	IN/OUT ^注	描述
RESET_IN	IN	该引脚用于从目标系统输入 reset 信号
RESET_OUT	OUT	该引脚用于向目标系统输出 reset 信号
CLK	OUT	该引脚用于向目标系统输出时钟信号
FLMD0	OUT	该引脚用于设定目标设备进入调试模式或编程模式
RxD	IN	该引脚用于接收来自目标设备的命令/数据
TxD	OUT	该引脚用于向目标设备发送命令/数据

注 以 EZ-CUBE 为参照基准。

4.1.2 电路连接示例

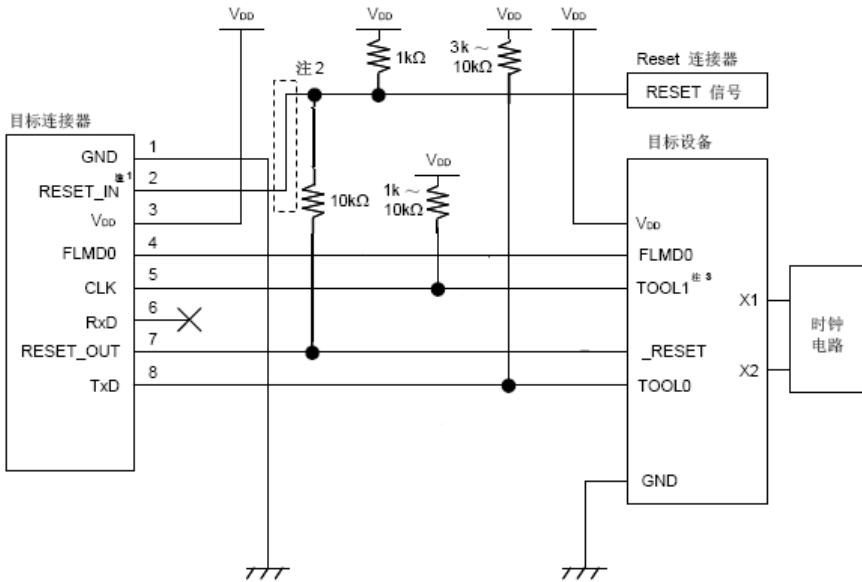
在调试和编程的时候，请根据描述的电路连接示例来设计合适的电路。

目标系统上的电路设计会根据接口信号而有所不同。接口的引脚名称可能会因目标设备有所不同。详细信息请参考目标设备的用户手册。

请确认使用目的，参照表 4-2 和相关的电路连接示例。

注意事项 在电路连接示例中出现的常数值为参考值。如果进行 flash 编程旨在量产时，请彻底评估目标设备的规格是否满足要求。

图 4-2. 电路连接示例



- 注**
1. 这种连接的设计是假定目标系统的复位电路包含有N沟道开漏缓冲器（输出阻抗：100欧姆或更小）。详细信息请参考4.1.3 reset 引脚的连接。
 2. 仅当flash编程时，不需要虚线所画的电路。
 3. 这个连接在双线通信模式下是必须的，但在单线模式下不需要。

EZ-CUBE 开关设置

- SW-1 开关: 请选择“M2”。
- SW-2 开关: 请选择“ Int. Clock ”。
- SW-3 开关: 请选择“Debug Mode”。
- SW-4 开关: 请根据实际目标设备使用选择。
- SW-5 开关: 请选择“Other”。

CAUTION

注意目标系统的电源提供:

1. 在USB线缆处于连接状态时，请勿改变开关的设置。
2. 最大允许电流100mA，所以请勿将EZ-CUBE连接到电流很大的目标系统上。在EZ-CUBE与主机连通后，主机始终保持向EZ-CUBE供电。

4.1.3 reset 引脚的连接

本节描述了电路中 reset 引脚的连接，此处必须特别注意，电路连接示例见前面的章节。

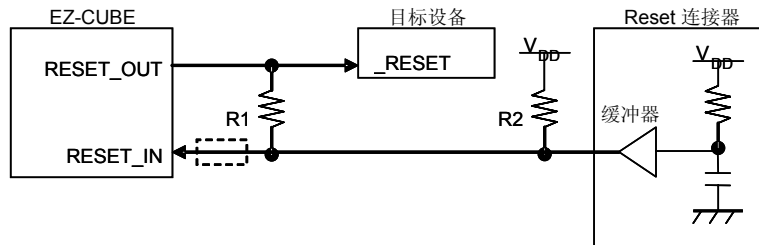
片上调试时，来自目标系统的 reset 信号输入 EZ-CUBE，被屏蔽后，再输入到目标设备。于是 reset 信号的连接根据是否连接 EZ-CUBE 而有所不同。

Flash 编程时，电路的设计必须保证目标系统的 reset 信号和 EZ-CUBE 的 reset 信号不冲突。

推荐通过电阻自动切换 reset 信号。图 4-3 说明了 4.1.2 电路连接示例中描述的 reset 引脚连接情况。

这种连接的设计是假定目标系统的复位电路包含有 N 沟道开漏缓冲器（输出阻抗：100 欧姆或更小）。当 EZ-CUBE 的 RESET_IN/OUT 逻辑反转时，VDD 或 GND 的电平可能会不稳定，所以请注意备注中所描述的条件。

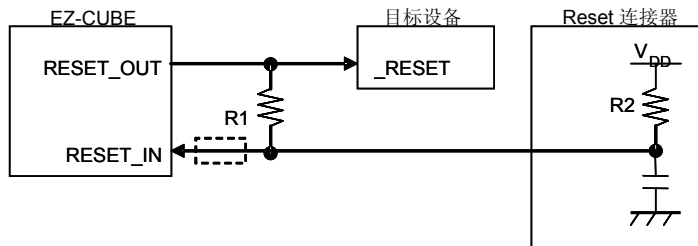
图 4-3. 包含缓冲器的复位电路的连接



备注 需确保 R1 的阻值至少是 R2 阻值的十倍，R1 为 10 kΩ 或更大。
如果复位电路的缓冲器是 CMOS 输出，则无需上拉电阻 R2。
当只用于 Flash 编程时，可以略去虚线框内的电路。

图 4-4 所示的电路连接情形是目标系统的复位电路不包含缓冲器，并且 reset 信号仅通过电阻或电容产生。设计电路时注意备注中描述的条件。

图 4-4. 不包含缓冲器的复位电路的连接



备注 需确保 R1 的阻值至少是 R2 阻值的十倍，R1 为 10 kΩ 或更大。
当只用于 Flash 编程时，可以略去虚线框内的电路。

4.2 片上调试

本节描述了使用EZ-CUBE进行片上调试的系统配置，启动/关闭顺序和调试中的若干注意事项。

4.2.1 调试功能

表4-5列出了以78K0R微控制器为目标设备时的调试功能。

表 4-5. 调试功能

功能	规格
目标MCU	78K0R系列 78K0R/ Kx3,Lx3
安全功能	10字节ID码验证
下载	支持
运行	全速执行，忽略断点执行，单步执行，下一步执行，运行至此，重启
硬件断点	1个
软件断点	多个
调试时所用的用户区域	单线模式：内部ROM：1036字节，内部RAM：6字节 ^注 双线模式：内部ROM：100字节，内部RAM：6字节 ^注
用于调试的功能引脚	单线模式：TOOL0 双线模式：TOOL0, TOOL1

注 详细情况请参考4.2.2用户资源的保留，设置安全ID以及片上调试选项字节。

4.2.2 用户资源的保留，设置安全ID以及片上调试选项字节

用户必须做好下面的保留以保证EZ-CUBE和目标设备之间的通信以及实现每一个调试功能。请参考下面的章节描述，在用户程序设置或使用构建工具属性。

(a) 安全ID区域

这个区域用来存放安全设置以防无认证的人读取Flash存储器。安全ID在地址0xC4到0xCD区域。安全ID功能就像启动调试器的密码。只有在调试器启动时输入的安全ID和存放在这个区域的安全ID匹配时，调试器才能正常启动。如果ID码不匹配，调试器根据片上调试选项字节区域设置的值对目标设备进行操作（参考表4-7）。如果用户忘记安全ID，请擦除flash memory，并重新设置ID。

[如何设置]

有两种方式设置选项字节。如果方式（1）和（2）都设置，方式（2）优先。

- （1）在用户程序地址0xC4到0xCD区域设置安全ID。
- （2）构建工具通用选项（CS+）中设置安全ID。

(1) 用户程序在地址0xC4到0xCD设置

举例，安全ID设置为“0123456789ABCDEF1234”（不区分大小写）

表 4-6 安全ID

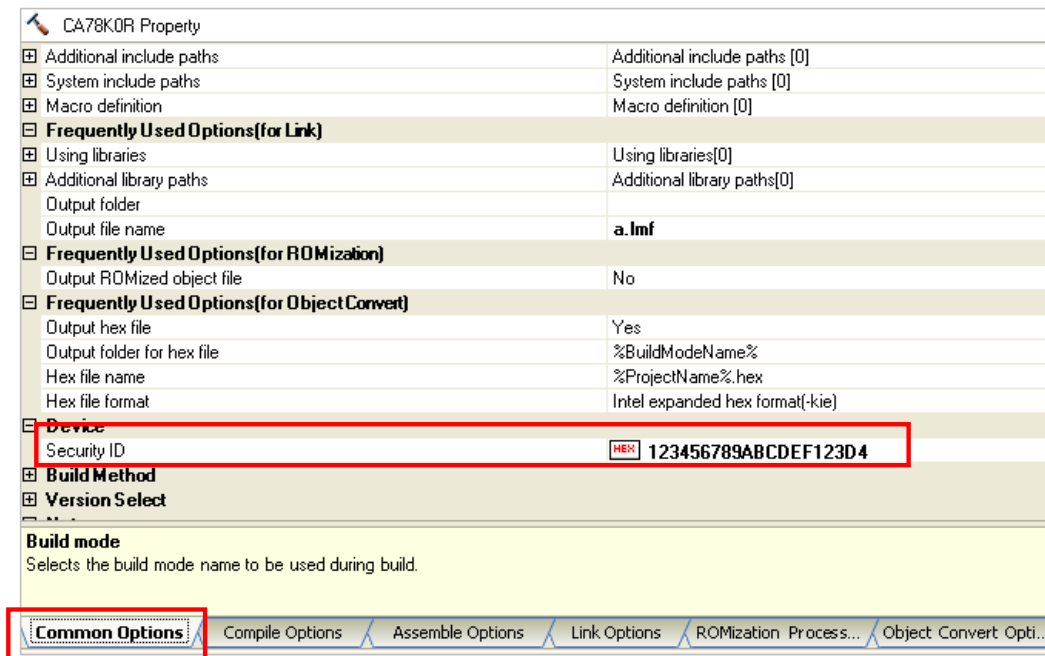
地址	值
0xC4	0x01
0xC5	0x23

0xC6	0x45
0xC7	0x67
0xC8	0x89
0xC9	0xAB
0xCA	0xCD
0xCB	0xEF
0xCC	0x12
0xCD	0x34

(2) CS+设置

在通用选项 “device”中设置安全 ID 如图 4-5。

图 4-5. 安全ID设置样例



(b) 片上调试选项字节设置

这个区域用来存放安全设置以防无认证的人读取 Flash 存储器。调试器根据设置的值对目标设备进行操作，如下表所示。

表 4-7. 片上调试选项字节设置和操作

设置值	描述	备注
0x04	不能进行调试。	这个设置只在Flash编程或自编程时有效
0x85	不管安全ID码认证失败多少次都不会擦除片上Flash存储器	-

0x84	如果安全ID码认证失败擦除所有片上Flash存储器区域	-
其它	禁止设置	-

[如何设置]

有两种方式设置选项字节。如果方式（1）和（2）都设置，方式（2）优先。

- (1) 在用户程序地址 0xC3 到 0xCD 区域设置在线调试选项字节。
- (2) 构建工具通用选项（CS+）中设置在线调试选项字节。

(1) 在用户程序地址0xC3到0xCD区域设置在线调试选项字节

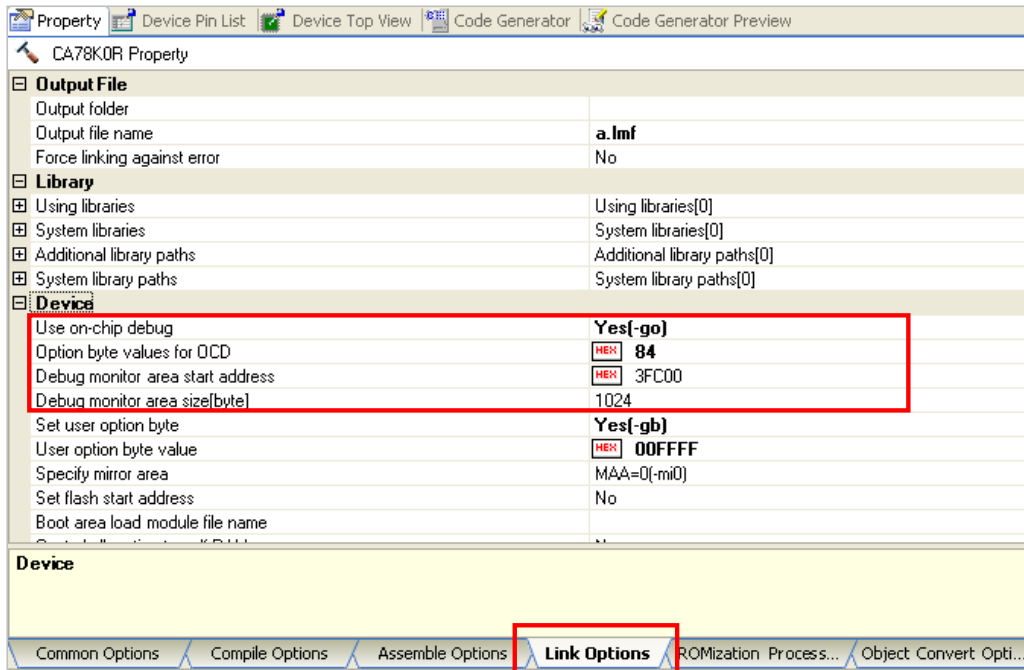
在用户程序地址0xC3到0xCD区域设置在线调试选项字节。

(2) 构建工具通用选项（CS+）中设置在线调试选项字节

如图4-6中显示了在CS+中的 [Link Options]选项卡设置“device”。例把

控制值设置为 0x84。

图 4-6. 预留调试监测区域和片上调试选项字节样例



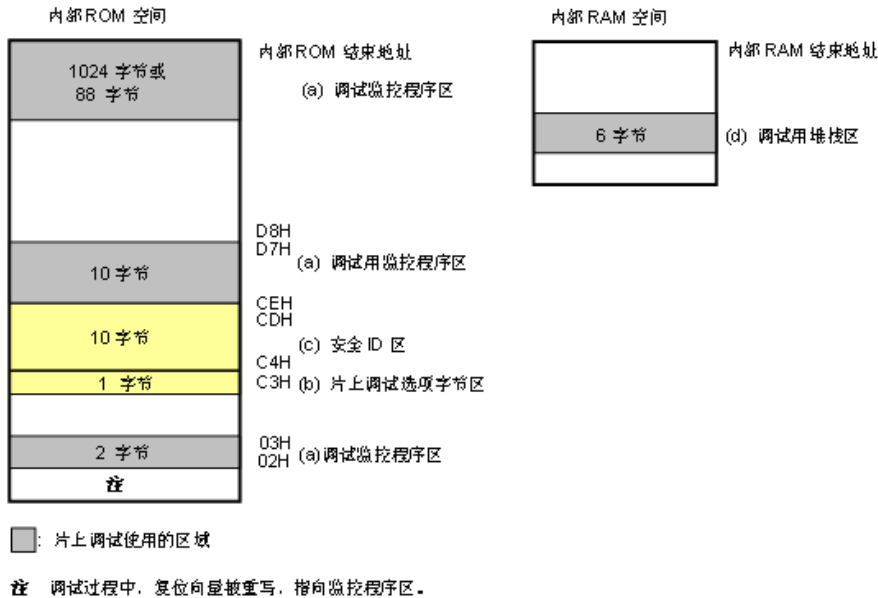
(c) 调试用堆栈区域的保留

图4-7中的阴影部分是存放调试监控程序的保留区域，所以用户程序或数据不能存放在这些空间。这些空间必须保证不被用户程序使用。

另外，这些区域也不能被用户程序重写。

调试保留区域内容说明，参见（1）和（2）。

图 4-7. 分配给调试监控程序的内存空间



(1) 调试监控程序的保留区域

这是一个放置调试监控程序的区域。监控程序为调试通信接口和 CPU 的运行或者断点做初始化操作。用户程序或数据不能存放在片上调试选项字节周边的 22 个字节的区域以及内部 ROM 底端的 1,024 字节区域内。另外，复位向量会被重写并指向放置监控程序的地址。

[如何预留区域]

为了防止调试启动时，占用造成的问题。推荐编译器预留区域。在如图4-6 中，使用 CS+在link选项卡设置“device”，预留此区域。

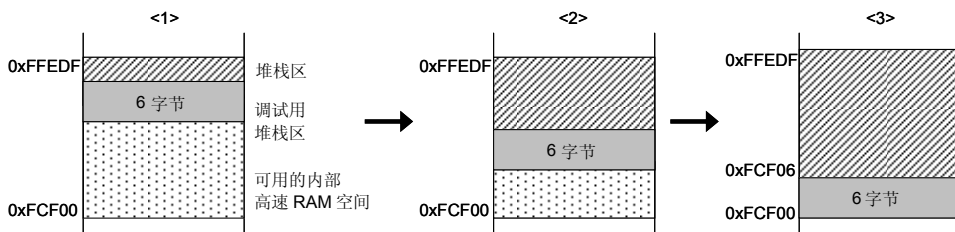
(2) 调试用堆栈区域的保留

这个区域需要 6 个字节作为调试的堆栈区域。因为这个区域分配在紧邻堆栈区域之前，所以这个区域的地址根据堆栈的增加和减少而有所改变。就是说，在使用的堆栈区域之外有 6 个额外的字节。

图 4-8 说明了堆栈区域增加时的情况，图中的内部高速 RAM 从 0xFCF00 开始。

备注 自编程时，将为12个字节。

图 4-8. 调试所用的堆栈地址的变化



[如何预留区域]

通过在用户程序使用堆栈之外再多定义 6 个字节来设置堆栈指针。确保堆栈指针不会超过内部高速 RAM 的起始地址。

备注 关于如何预留自编程序的堆栈区域请参考自编程手册。

4.2.3 调试的注意事项

本节描述了在 78K0R 微控制器上进行片上调试的注意事项。
请认真阅读下列事项，以保证 EZ-CUBE 的正常使用。

- (1) 处理用于调试的设备
不要在批量生产的产品上安装曾用于调试的设备，因为 Flash 存储器在调试过程中被多次重写，Flash 存储器的重写次数无法保证。另外，不要把调试监控程序写入批量生产的产品。
- (2) Flash 自编程
如果存放调试监控程序的区域被 Flash 自编程重写，调试器就不能正常操作。这个注意事项对这个区域的引导交换操作也适用。
- (3) 复位后的操作
在外部复位或内部复位后，监控程序会进行调试初始化处理。因此，从产生复位到用户程序执行的时间和实际设备的操作有所不同。
- (4) 使用实机调试而不使用 EZ-CUBE
如果使用实机调试而不使用 EZ-CUBE，请使用 RFP 写入用户程序。通过调试器下载的程序包括监控程序，这样的程序如果在有使 TOOL0 引脚变成低电平的操作时会出现误操作。
- (5) 调试器启动后的操作
调试器启动后，如果调试器配置对话框中的目标设备连接设置和上次的调试设置不同，内部 Flash 存储器会被擦除。
- (6) 通过 Flash 编程写入程序后的调试
如果通过 RFP 把程序写入内部 Flash 存储器，即使在片上调试选项字节设置中允许了片上调试，也不能进行片上调试。如果要对这样的目标设备要进行调试，可以先用 RFP 擦除内部 Flash 存储器后通过调试器下载程序。
- (7) 默认 LVI 起始功能设置(地址 C1H)
调试过程中，调试监控程序停止了 C1H 地址处 LVI 的默认启动功能。因此，即使调试结束，LVI 默认启动功能仍保持被停止状态，通过 flash 编程功能改变地址 C1H 的设置。
- (8) 片上调试选项字节设置（地址 C3H）
片上调试选项字节设置被调试器重写为任意值。
- (9) 操作电压不能写入 flash memory
如果片上调试时进行了如下步骤，调试器自动更改寄存器的设置，从而进行 flash memory 重写。如果执行下列操作<1>至<7>，而 flash memory 的重写已被禁用或与不被重写的电压下进行操作，调试器会输出一个错误和忽略该操作。
<1> 内部 flash memory 写入

- <2> 设置或取消软件断点
- <3> 启动执行软件断点
- <4> 软件断点进行单步执行
- <5> 执行 Step-over, Return Out
- <6> Come Here
- <7> 如果调试工具属性中允许 flash 编程被选中, 以下操作将不能执行
 - a)指定、更改或取消硬件断点
 - b)内部复位
 - c)外围断点

(10) 在单线模式下调试

在单线模式下执行调试的情况下 (通过在调试器配置对话框的 Connection with Target Board 区 Communication method 选择 1 line type(TOOL0)), 当内部高速振荡器被用于 CPU 的工作时钟时, 如果在调试器启动和中断发生之间频率变化太大 (改变寄存器时除外), 中断可能不会正常发生。这种情况在工作电压或温度改变太大时也可能发生。

(11) 待机功能和 Break 功能之前的关系

Break 是 CPU 中断功能。在调试功能被打破时待机模式被释放。

- 停止用户程序的执行
- 待机指令单步执行 (执行指令后停止用户程序)
- 伪实时 RAM 监视功能 (break 被打破)
- 伪动态 memory 修改
- 执行用户程序断点设置

(12) 使用 Step-in 的注意事项 (单步执行)

当代码单步执行时, 一些 SFR (特殊功能寄存器) 的值可能会保持不变。如果代码单步执行时, 一些 SFR (特殊功能寄存器) 的值保持不变, 单片机操作一直执行指令, 而不是执行这些步骤。

代码单步: 用户程序中的指令逐条执行

代码联系: 用户程序从当前 PC 指针开始执行

4.3 Flash 编程

本节描述了使用EZ-CUBE对78K0R微控制器进行Flash编程时的系统配置和启动/关闭顺序。

4.3.1 编程功能的规格定义

表 4-8. 编程功能的规格定义

功能	规格定义说明
主机接口	USB2.0
目标接口	UART（单线模式）
目标系统电压	2.7到5.5V（根据目标设备而定）
时钟提供	使用内部高速振荡器
电源	5V±0.3V（最大电流：100mA）
获取设备特定信息	使用RENESAS ELECTRONICS的78K0R参数文件
编程软件	RFP [®] (Renesas Flash Programmer)
安全标志设置	支持
脱机操作	不支持（必须连接主机）

注：详情请见RFP用户手册

4.3.2 Flash 编程的注意事项

本节描述了Flash编程的注意事项。请认真阅读，以保证EZ-CUBE的正确使用。

- 为了提高写入质量，请在使用EZ-CUBE之前对下列问题进行充分了解，并加以校验和评估。
 - 电路的设计要符合EZ-CUBE用户手册对设备的描述
 - 设备，RFP和EZ-CUBE的使用都要遵循各自手册的描述
 - 提供给目标系统的电源要稳定

第 5 章 如何在 78K0 微控制器上使用 EZ-CUBE

本章描述了如何使用 EZ-CUBE 在 78K0 微控制器上进行片上调试和 Flash 编程。

片上调试是指通过芯片内置的调试功能对已经装载在目标系统上的微控制器进行调试的方法。由于这种调试方法是通过板上的目标设备实行的，所以非常适合现场调试。

Flash 编程是将程序写入设备内置的 Flash 存储器的方法。可以在板上对设备进行擦除，写入和校验。

请先将 EZ-CUBE 的固件更新为用于 78K0，具体请参照 (1) - (3)。详细内容请参见 **1.4 固件更新**。

- (1) 使用 USB 缆连接 EZ-CUBE 和 PC。**不要将 EZ-CUBE 与目标板相连。**
- (2) 启动【QBEZUTL.exe】，并指定“78K0_OCD_FW.hex”固件文件。
- (3) 点击[Start]按钮，然后指定用于 78K0 固件下载到 EZ-CUBE。

如果首次用 78K0 微控制器作为目标设备来使用 EZ-CUBE，那么请仔细阅读以下内容。

5.1 目标系统设计

要使用 EZ-CUBE 和目标系统进行通信，必须在目标系统上安装对应的通信电路。本节描述了相关电路的设计和连接插头的安装。

5.2 片上调试

本节描述了用 EZ-CUBE 进行片上调试的系统配置和启动方法。

5.3 Flash 编程

本节描述了用 EZ-CUBE 进行 Flash 编程的系统配置和启动方法。

支持 78K0 MCU 列表

表5-1 显示了EZ-CUBE支持的78K0系列MCU列表及固件。

表 5-1 EZ-CUBE 支持78K0 MCU列表及固件

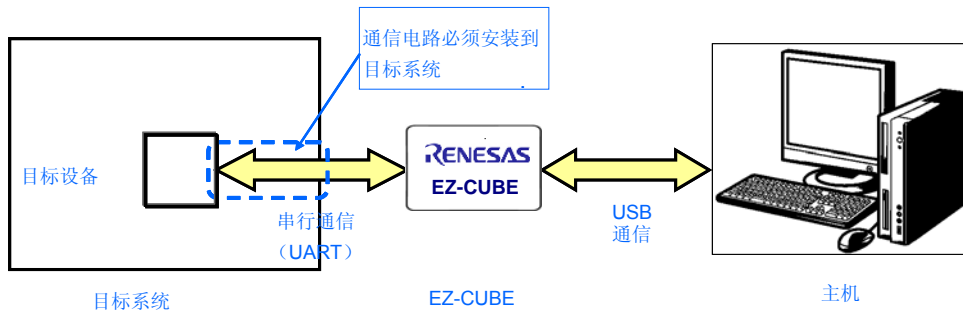
项目	内容	固件
支持MCU	78K0/Kx2, Lx3	78K0_OCD_FW.hex
支持的操作模式	UART模式	

5.1 目标系统设计

要使用 EZ-CUBE 和目标系统进行通信，必须在目标系统上安装对应的通信电路。本节描述了相关电路的设计和连接插头的安装。

图 5-1 显示了 EZ-CUBE 通信的整体接口框图。为了 EZ-CUBE 和目标系统之间的通信，目标系统上必须要安装通信电路，如图中左侧所示。请参考本节的内容进行合适的电路设计。

图 5-1 通信接口框图



5.1.1 引脚配置

本节将说明 EZ-CUBE 和目标系统之间使用的接口信号。表 5-2 中列出了引脚的配置情况。表 5-3 说明了各个引脚的功能。

表 5-2. 引脚配置

引脚编号	引脚名称 ^注
1	GND
2	RESET_IN
3	Vdd
4	FLMD0
5	CLK
6	RxD.
7	RESET_OUT
8	TxD

注 EZ-CUBE 内的信号名称。

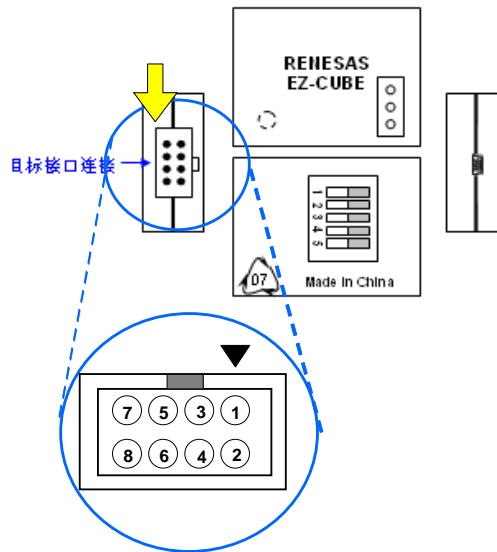


表 5-3. 引脚功能

引脚名称	IN/OUT*	描述
RESET_IN	IN	该引脚用于从目标系统输入 reset 信号
RESET_OUT	OUT	该引脚用于向目标系统输出 reset 信号
CLK	OUT	该引脚用于向目标系统输出时钟信号
FLMD0	OUT	该引脚用于设定目标设备进入调试模式或编程模式
RxD	IN	该引脚用于接收来自目标设备的命令/数据
TxD	OUT	该引脚用于向目标设备发送命令/数据

注 以 EZ-CUBE 为参照基准。

5.1.2 电路连接示例

在调试和编程的时候，引脚配置有很大区别，所以请根据描述的电路连接示例来设计合适的电路。

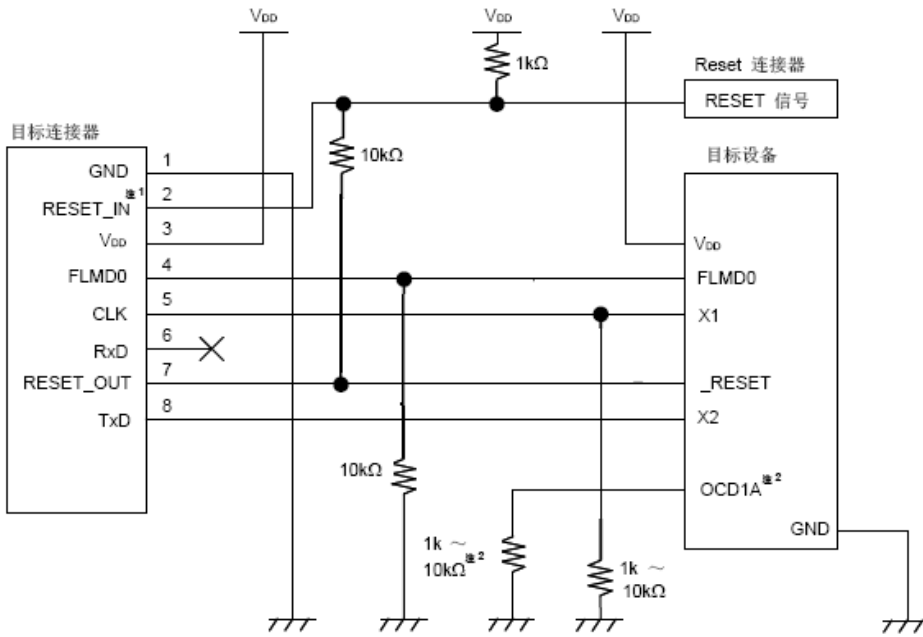
目标系统上的电路设计会根据接口信号而有所不同。接口的引脚名称可能会因目标设备有所不同。详细信息请参考目标设备的用户手册。

下面的（1）至（2）项是针对不同使用情况的主要用途。请确认使用目的，参照表 5-2 和相关的电路连接示例。

注意事项 在电路连接示例中出现的常数值为参考值。

- （1）进行片上调试，参考图 5-2
- （2）进行 Flash 编程，参考图 5-3

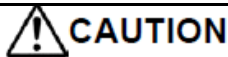
图 5-2 片上调试时



- 注：
1. 这种连接的设计是认为 RESET 信号是从 N 沟道开漏缓冲器（输出阻抗：100 欧姆或更小）输出的。详细信息请参考 5.1.3 reset 引脚的连接
 2. 各种不同设备上的 OCD1A 引脚名称可能不同，详细信息请参考目标设备的用户手册。

EZ-CUBE 开关设置

- SW-1 开关: 请选择“M2”.
- SW-2 开关: 请根据实际目标设备使用选择.
- SW-3 开关: 请选择“Debug Mode”.
- SW-4 开关: 请根据实际目标设备使用选择.
- SW-5 开关: 请选择“Other”.

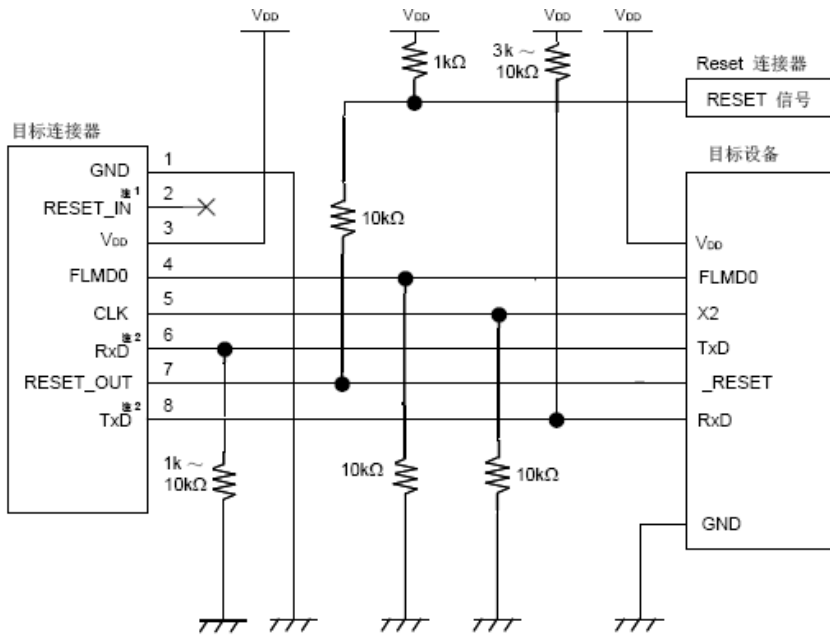


注意目标系统的电源提供:



1. 在USB线缆处于连接状态时，请勿改变开关的设置.
2. 最大允许电流100mA，所以请勿将EZ-CUBE连接到电流很大的目标系统上。在EZ-CUBE与主机连通后，主机始终保持向EZ-CUBE供电.

图 5-3 flash 编程时



- 注： 1. 这种连接的设计是认为 RESET 信号是从 N 沟道开漏缓冲器（输出阻抗：100 欧姆或更小）输出的。详细信息请参考 5.1.3 reset 引脚的连接。
2. 将目标设备的 TxD（发送方）连接到目标连接器的 RxD（接收方），同时将目标连接器的 TxD（发送方）连接到目标设备的 RxD（接收方）。

EZ-CUBE 开关设置

- SW-1 开关: 请选择“M1”.
- SW-2 开关: 请根据实际目标设备使用选择.
- SW-3 开关: 请选择“Debug Mode”.
- SW-4 开关: 请根据实际目标设备使用选择.
- SW-5 开关: 请选择“Other”.

CAUTION

注意目标系统的电源提供:

1. 在USB线缆处于连接状态时，请勿改变开关的设置.
2. 最大允许电流100mA，所以请勿将EZ-CUBE连接到电流很大的目标系统上。在EZ-CUBE与主机连通后，主机始终保持向EZ-CUBE供电.

5.1.3 reset 引脚的连接

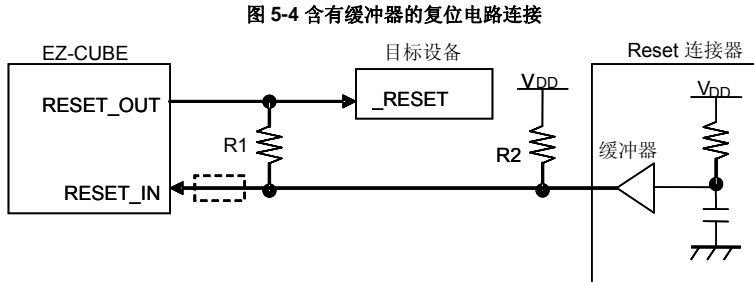
本节将介绍上一节的电路连接示例中提到的需要特别注意的 reset 引脚的连接方法。

片上调试时，来自目标系统的 reset 信号输入 EZ-CUBE，被屏蔽后，再输出到目标设备。于是 reset 信号的连接因为 EZ-CUBE 是否连接而有所不同。

编程时，电路的设计必须保证目标系统的 reset 信号和 EZ-CUBE 的 reset 信号不冲突。

推荐通过电阻自动切换 reset 信号。图 5-4 说明了 5.1.2 电路连接示例中描述的 reset 引脚连接情况。

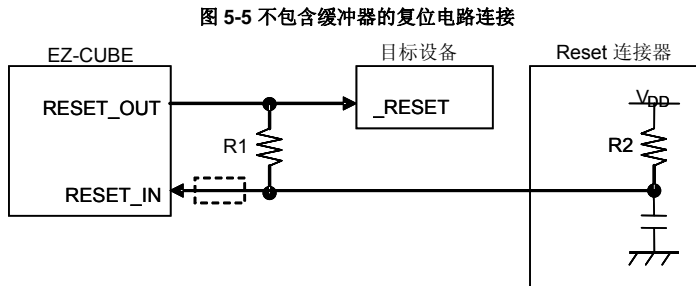
这种连接的设计是认为目标系统的 RESET 电路包含有 N 沟道开漏缓冲器（输出阻抗：100 欧姆或更小）。当 EZ-CUBE 的 RESET_IN/OUT 逻辑反转时，VDD 或 GND 的电平可能不稳定，所以请注意备注中所描述的条件。



备注：需确保R1的阻值至少是R2阻值的十倍，R1为10k或更大。

如果复位电路的缓冲器为CMOS输出，就不再需要R2上拉电阻。当仅进行Flash编程时，虚线框内的电路可以略去。

图 5-5 所示的电路连接示例描述了目标系统的复位电路中不包含缓冲器以及 reset 信号仅通过电阻或电容产生的情况。设计电路时需要注意备注中描述的条件。



备注：需确保R1的阻值至少是R2阻值的十倍，R1为10 k或更大。

只有在进行 Flash 编程时，虚线框内的电路才可以略去。

5.1.4 目标系统设计的注意事项

在设计目标系统时，请务必注意以下的事项。

- 目标系统电路布局时请不要将调试通信线路放在相邻的位置。如果必须这样布局，他们的连线要尽量短。
- 因为调试的通信信号用于高速通信，使用接地屏蔽以减少负载。
- 目标连接器和目标设备之间的距离要尽量短。
- 在产品出货之前，请使用跳线或者其他方式将 X1/OCD1A 和 X2/OCD1B 的引脚同目标连接器分离开来，以确保正常的时钟振荡。
- 在调试时，请将电容、反馈电阻以及其他用于振荡的元器件移去，以防止 X1/OCD1A 和 X2/OCD1B 引脚上所连接的负载引发信号失真。

5.1.5 时钟设置

片上调试使用的使用信号，可以由 X1 振荡器、内部振荡器提供。设置描述如下。

(1) 使用内部高速振荡器

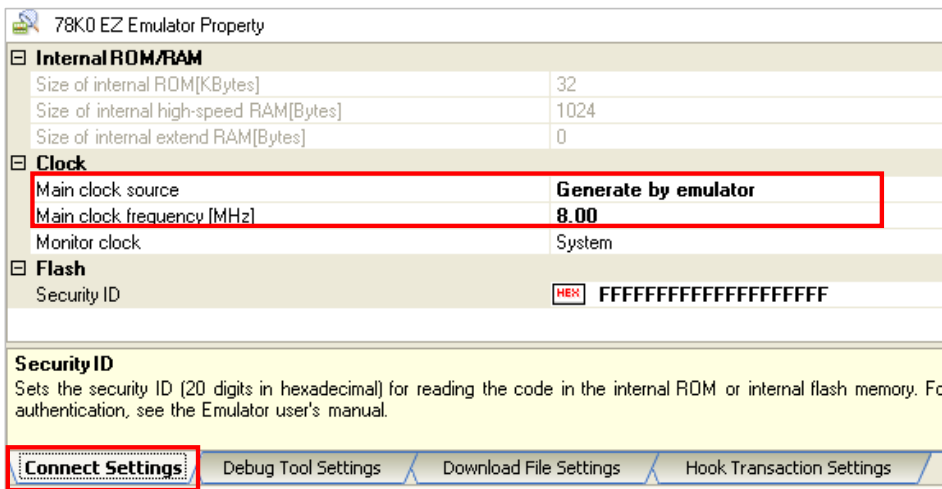
EZ-CUBE 可以提供时钟信号用作高速系统时钟（4、8 或 16MHz）。

SW-2 开关选为“Int. Clock”。

移除外部晶振的振荡器或振荡电路。（在 CS+的“Debug Tool”的[Connect Settings]选项卡中，时钟选择“Generate by emulator”为主时钟）。

这些设置的具体含义和详细内容请参考 CS+的用户手册。

图 5-6 时钟设置

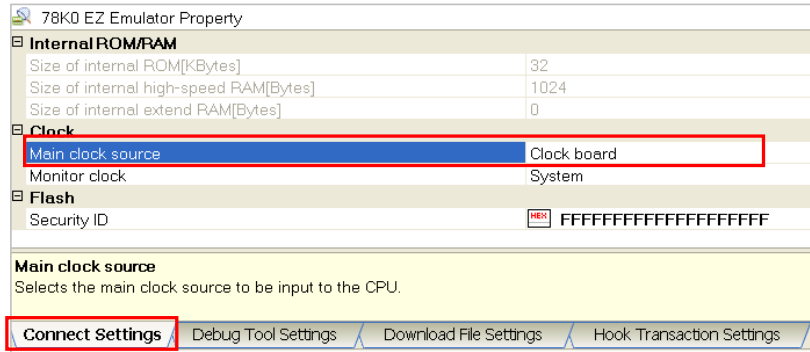


(2) 使用外部振荡器

SW-2 开关选为 “Ext. Clock”。

在EZ-CUBE 上的外部晶振插座位置安装振荡器或者振荡电路。（在调试器的配置对话框中，选择外部晶振为主时钟）。详细信息，请参考CS+的用户手册。

图 5-7 时钟设置



5.2 片上调试

本节描述了使用 EZ-CUBE 进行片上调试的系统配置，启动/关闭顺序和调试中的若干注意事项。

5.2.1 调试功能

以 78K0 微控制器作为目标设备，进行调试时的调试功能。

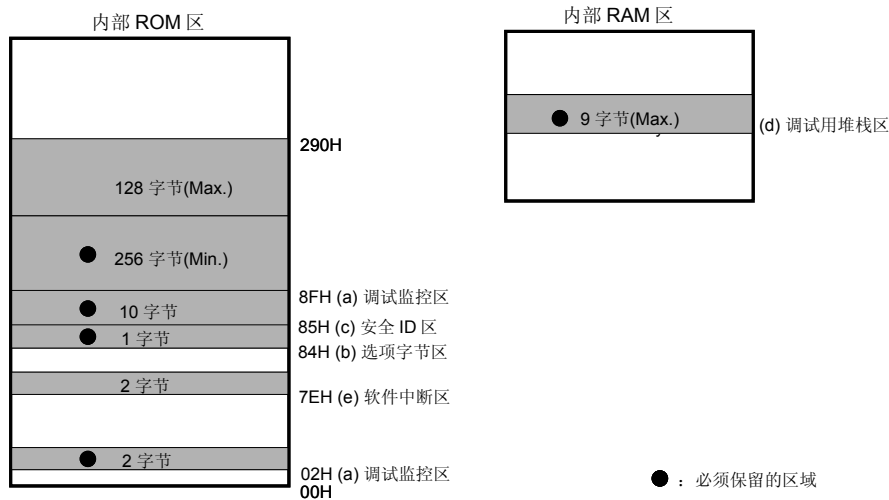
表5-4 调试功能

功能	规格定义
目标MCU	78K0系列 78K0/ Kx2, Lx3
安全	10字节ID码验证
下载	支持
执行	全速执行, 忽略断点执行, 单步执行, 下一步执行, 运行至此, 重启
硬件断点	1个
软件断点	多个
调试所用的用户空间	内部 ROM: 256 ~ 400 字节 内部RAM: 7 ~ 9 字节 根据设备而不同
用于调试的功能引脚	X1, X2

5.2.2 用户资源的保留和安全 ID 的设置

EZ-CUBE 使用用户存储区域（图 5-8 中阴影表示的区域）来完成自身和目标设备的通信，或者完成各种调试功能。用点（●）表示的区域在调试过程中会一直被占用，其他区域的占用对应实现各个调试功能。请参考后面描述的（a）至（e）项，并在用户程序中预留这些区域，或使用编译参数加以指定。

图5-8 使用EZ-CUBE时的保留区域



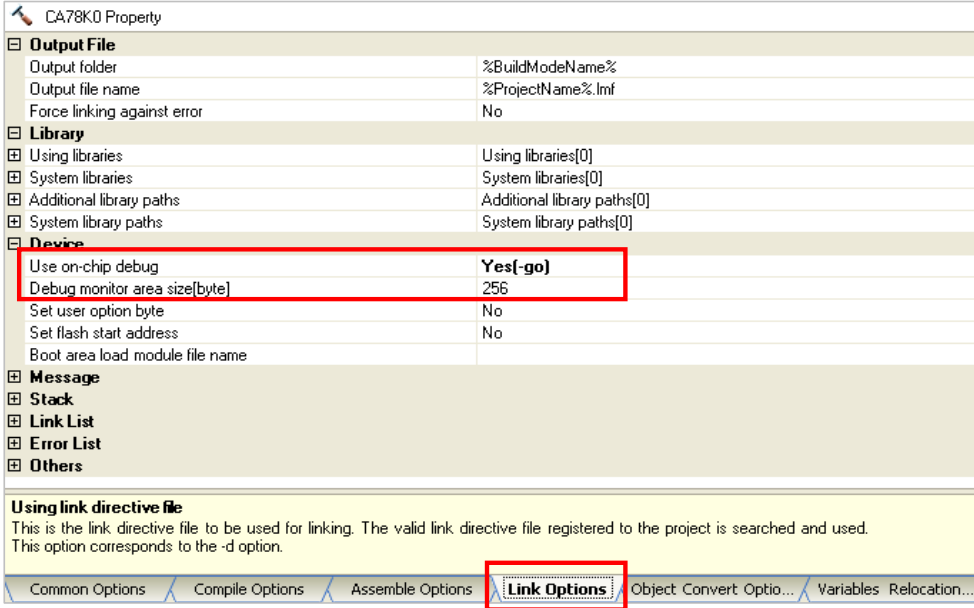
(a) 调试监控区域（必须保留此区域）

地址 0x02、0x03 以及 0x8F 开始的地址必须预留，以存放调试监控程序。务必保留此区域。监控程序执行初始化处理的通信接口，用于调试以及 CPU 运行/中断处理。如果这块区域被 Flash 自编程重写，则片上调试无法再进行。

[如何预留区域]

图 5-9 显示CS+设置预留区域举例。CS+的 link option 设置选项卡，[Use on-chip debug]设置为[Yes (-go)]，[Debug monitor area size]为256 字节，图 5-9 中红色区域。此预留区域0x02, 0x03, 和 0x8F 作为调试监视区。

图5-9 连接参数设置（调试监控区域）



(b) 选项字节区域（必要）

本区域用于安全设置，以防止对 Flash 存储器未经授权的读取。根据设置值的不同，目标设备的操作有所不同，具体如下表所示。选项字节区域的详细设置，请参见芯片的用户手册。

表5-5 选项字节设置和操作

可选值	描述	备注
0x00	即使连接了片上调试的仿真器（比如EZ-CUBE），仍然无法调试	这个设置仅对Flash编程有效
0x02	不论安全ID鉴别发生多少次错误，片上Flash存储器都不会被擦除	-
0x03	如果安全ID鉴别错误，片上Flash存储器的所有内容都会被擦除	-
其它	禁止设置	-

[如何设置]

有两种方式设置选项字节。如果方式（1）和（2）都设置，方式（2）优先。

(1) 用户程序设置

描述如何在用户程序实现选项字节的设置。请参考下面的汇编代码。

例 设置为 80H:03, 81H:00 82H:00 83H:00H 84H:02H

SSS CSEG AT 080H: "SSS"是任意符号名（长度不超过 8 个字符）

```

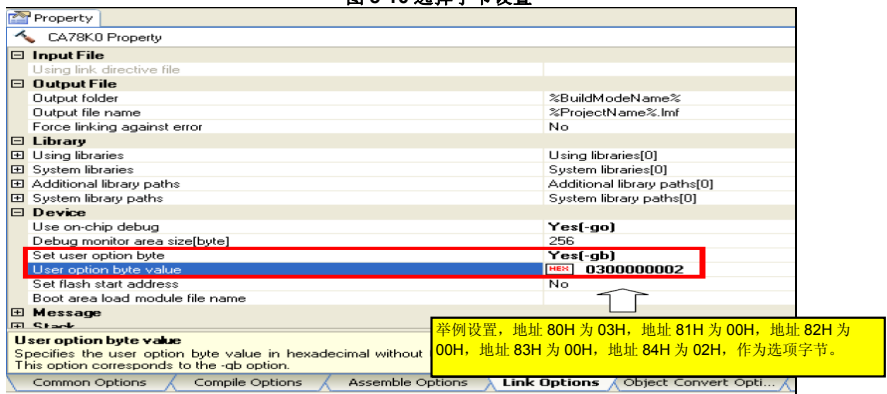
DB 03H;
DB 00H;
DB 00H;
DB 00H;
DB 02H;
    
```

注意事项 如果地址 0x84 的内容在自编程中被重写为 0x00，EZ-CUBE 和主机之间的通信就被禁止，此时即使重启调试器，调试和连接也无法进行。此时，只能通过 Flash 编程的方法擦除存储器内容。

(2) CS+设置

也可以通过 CS+来设置选项字节。如图 5-10 红色区域，[Set user option byte]选择 [Yes (-gb)]，地址80H 到 84H 指定选项字节值。

图 5-10 选择字节设置



(c) 安全 ID 区域（必须）

本区域用于安全设置，以防止对 Flash 存储器未经授权的读取。

安全 ID 功能作为启动调试器的密码。调试器启动之初，只有当输入的安全 ID 和本区域中写入的安全 ID 相符合时，调试器才可以启动。如果遗忘了安全 ID，通过擦除 Flash Memory 来设置新的安全 ID。

[如何设置]

有两种方式设置选项字节。如果方式（1）和（2）都设置，方式（2）优先。

(1) 用户程序设置

描述如何在用户程序实现安全 ID 的设置。举例，安全 ID 设置为“0123456789ABCDEF1234”（不区分大小写）

例) 地址 0x85 到 0x8E 设置安全 ID “0123456789ABCDEF1234”

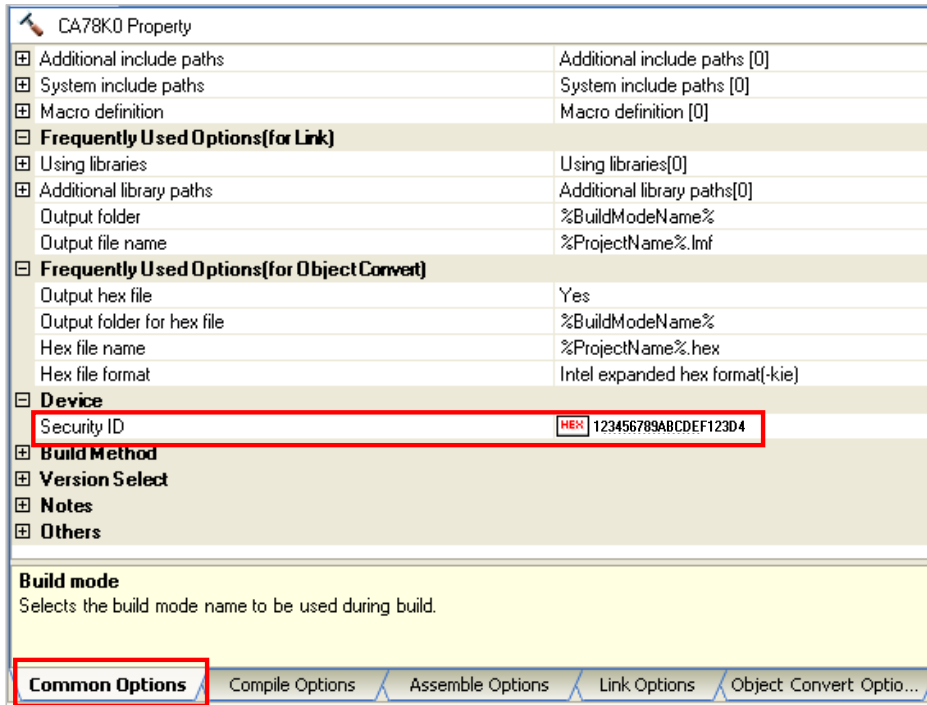
SSS CSEG AT 85H; “SSS” 任意符号名 (长度不超过 8 个字符).

DB 01H;
DB 23H;
DB 45H;
DB 67H;
DB 89H;
DB ABH;
DB CDH;
DB EFH;
DB 12H;
DB 34H;

(2) CS+设置

也可以通过CS+来设置安全ID。如图5-11 红色区域, [Security ID]设置, 地址85H 到8FH 指定值。

图5-11 连接参数设置 (安全ID)

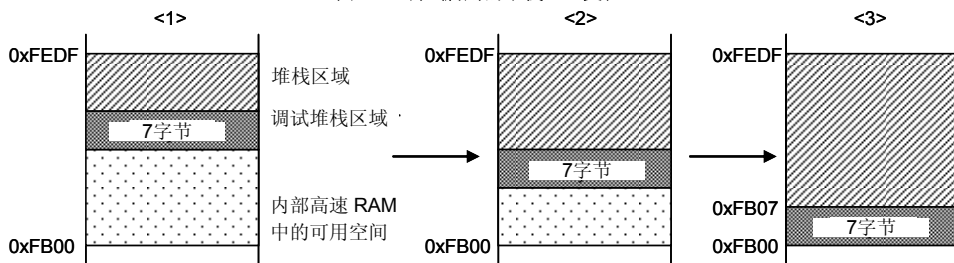


(d) 调试所需的堆栈区域 (必须保留此区域)

需要 7~9 个字节作为调试的堆栈区域。由于本区域被分配在紧邻堆栈区域之前, 所以本区域的实际地址会根据堆栈的增加和减少而改变。

图 5-12 说明了堆栈区域增加时的情况, 图中的内部高速 RAM 从 0xFB00 开始。

图 5-12 调试所用的堆栈地址变化



根据是否使用软件断点功能而有所不同，本区域所占用的大小也会改变。

表5-6 调试所用的堆栈区域大小

选项	调试所需的堆栈区域大小
标准	7字节
使用软件中断	9字节

[如何预留这段区域]

请参考下面的地址范围来设置堆栈指针。

例 当内部高速 RAM 从 0xFB00 开始时

- 当使用硬件断点时

在 0xFB07 到 0xFEDF 范围内[※]

- 当使用软件断点时（参见（e）项）

在 0xFB09 到 0xFEDF 范围内[※]

注

因为 EZ-CUBE 在中断时占用了 0xFEDF，请将堆栈指针初始值指向 0xFEDF，而不要指向 0xFEE0（当初始值设置为 0xFEDF，堆栈实际使用的值为 0xFEDE，所以就不会产生冲突）。另外，0xFEDF 被禁止用作堆栈区域，但是可以被用作变量区域。

(e) 软件断点区域

本区域用于软件断点。

[如何预留这段区域]

参考如下代码来预留区域。

```
SSS CSEG AT 07EH; "SSS"是任意符号名（长度不超过 8 个字符）
DB 0FFH, 0FFH
```

5.2.3 调试的注意事项

本节描述了在 78K0 微控制器上进行片上调试的注意事项。

请认真阅读下列事项，以保证 EZ-CUBE 的正常使用。

(1) 处理用于调试的设备

不要在批量生产的产品上安装曾用于调试的设备，因为 Flash 存储器在调试过程中曾被多次重写，Flash 存储器的重写次数无法保证。

(2) 片上调试时改写 Flash 存储器

如果片上调试时进行了如下步骤，则设备中的 Flash 存储器会被改写。

<1> 下载程序

<2> 设置或取消软件断点，程序执行

<3> 执行 Step-over, Return Out

<4> Come Here

<5> 如果配置对话框中的目标电源关闭被选中，以下操作将不能执行

a) 指定、更改或取消硬件断点

b) 内部复位

c) 外围断点

d) 程序执行

e) 软件复位（调试器执行复位）

<6> 当使用实时 RAM 监视功能时增加、更改或删除监视地址

<7> 使用软件断点时，设置软件断点

<8> 当调试器启动或终止

从 Flash 存储器编程的完成到转入 GUI 控制需要一段时间。

(3) 软件断点

在程序运行中，不要重写软件断点地址上的数据。这包括自编程和 RAM 重写。如果重写了，那么存放在这个地址上的指令可能失效。

(4) 自编程期间的引导交换

如果自编程重写了调试监视程序，调试器操作将不正确。仿真引导交换时也是如此。

(5) 堆栈指针初始化失败的中断功能

如果没有正确初始化堆栈指针，则在中断发生时，或者执行 PUSH 指令时，这个功能会执行一个断点。

如果在 reset 操作之后，紧接着有如下操作或者指令，堆栈指针初始化失败的断点功能就会失效。

- 设置一个软件断点
- 从寄存器窗口写堆栈指针
- 从存储器窗口写 Flash 存储器，等等

如果在堆栈指针没有正确初始化的时候发生了一个软件中断，在任务栏就会显示“Uninitialized Stack Pointer”。

如果发生了这种情况，后续的操作将无法正常进行，所以，请确保在用户程序中正确设置 SP。

(6) 下载 HEX 文件时的注意事项

当下载 HEX 文件时，请不要将目标转换参数 (-U) 的填充值指定为 0xFF 之外的值。

(7) 单步执行时的注意事项

在单步执行时，某些 SFRs（特殊功能寄存器）的内容不能被更新。这种情况下，用实时运行来执行指令。

(8) 代码单步时的注意事项

当代码单步执行时，一些 SFR（特殊功能寄存器）的值可能会保持不变。如果代码单步执行时，一些 SFR（特殊功能寄存器）的值保持不变，单片机操作一直执行指令，而不是执行这些步骤。

代码单步：用户程序中的指令逐条执行

代码联系：用户程序从当前 PC 指针开始执行

(9) POC 功能的仿真

当调试时目标系统的电源被关闭时，由于 EZ-CUBE 的 RESET_OUT 引脚，目标设备会进入复位状态。因此，目标设备的 POC 功能执行伪仿真功能。注意 POC 的检测电压和目标系统的电压有 ± 1 V 的误差。

5.3 Flash 编程

本节描述了使用 EZ-CUBE 对 78K0 微控制器进行 Flash 编程时的系统配置和启动/关闭顺序。

5.3.1 编程功能的规格定义

表5-7 编程功能的规格定义

功能	规格定义说明
主机接口	USB 2.0
主机接口	UART
目标系统电压	2.7到5.5 V (根据目标设备而定)
时钟源	可以提供8 MHz时钟 可以使用外部振荡器
电源	5 V ± 0.3V (最大电流: 100mA)
获取设备特定信息	使用RENASAS ELECTRONICS的78K0参数文件
编程软件	RFP [®] (Renesas Flash Programmer)
安全标识设置	支持
脱机操作	不支持 (必须连接主机)

注: 详情请见 RFP 用户手册

5.3.2 Flash 编程的注意事项

本节描述了 Flash 编程的注意事项。请认真阅读, 以保证 EZ-CUBE 的正确使用。

为了提高写入质量, 请在使用 EZ-CUBE 之前对下列问题进行充分了解, 并加以校验评估。

- 电路的设计要符合 EZ-CUBE 用户手册中对设备的描述。
- 设备, RFP 和 EZ-CUBE 的使用都要遵循各自手册的描述。
- 目标系统的供电电源要稳定。

第6章 如何在RX微控制器上使用EZ-CUBE

本章描述了如何使用EZ-CUBE在RX微控制器上进行片上调试和Flash编程。

片上调试是指通过芯片内置的调试功能对已经装载在目标系统上的微控制器进行调试的方法。由于这种调试方法是对板上的目标设备进行操作的，所以非常适合现场调试。

Flash编程是将程序写入设备内置的Flash存储器的方法。可以在板上对设备进行擦除、写入和校验。

请先将EZ-CUBE的固件更新为用于RX，具体请参照(1)-(3)。详细内容请参见**1.4 固件更新**。

- (1) 使用USB缆连接EZ-CUBE和PC。**不要将EZ-CUBE与目标板相连。**
- (2) 启动[QBEZUTL.exe](V1.14 or later)，并指定“RX_OCD_FW.hex”固件文件。
- (3) 点击“Start”按钮，然后指定用于RX固件下载到EZ-CUBE。

如果首次用RX微控制器作为目标设备来使用EZ-CUBE，那么请仔细阅读以下内容。

6.1 目标系统设计

要用EZ-CUBE和目标系统进行通信，必须在目标系统上安装对应的通信电路。本节描述了相关电路的设计以及接口的安装。

6.2 片上调试

本节描述了用EZ-CUBE进行片上调试的系统配置和启动方法。

6.3 Flash编程

本节描述了用EZ-CUBE进行Flash编程的系统配置和启动方法。

支持RX MCU列表

表6-1显示了EZ-CUBE支持的RX系列MCU列表及固件。

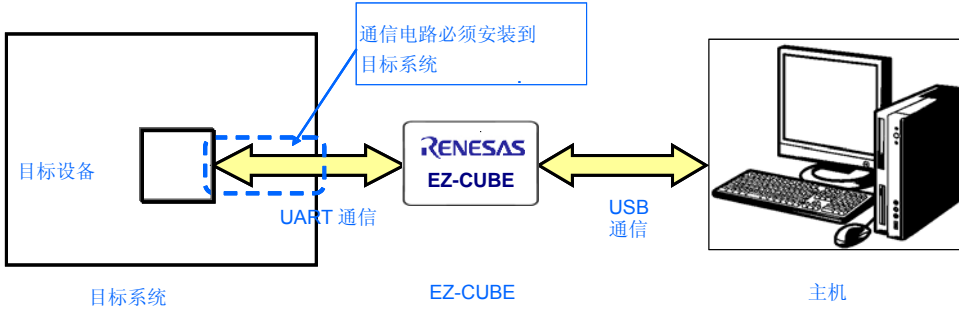
表 6-1 EZ-CUBE 支持RX MCU列表及固件

项目	内容	固件
支持MCU	RX600系列 RX63T群 [R5F563T4/5/6]	RX_OCD_FW.hex
	RX200系列 RX210, RX220 群	
	RX100系列 RX111 群	
支持的操作模式	Single chip模式	

6.1 目标系统设计

图 6-1 为 EZ-CUBE 通信的整体接口框图。

图6-1. 通信接口概况



6.1.1 引脚配置

本节将说明 EZ-CUBE 和目标系统之间使用的接口信号。表 6-2 中列出了引脚的配置情况。表 6-3 说明了各个引脚的功能。

表 6-2. 引脚配置

引脚编号	引脚名称 *1	
	RX63T [R5F563T4/5/6]	RX210, RX220, RX111
1	GND	GND
2	RES#	RES#
3	VCC	VCC
4	N.C.	N.C.
5	FINEC *2	N.C.
6	RxD *3	RxD *3
7	N.C.	N.C.
8	MD/FINED, TxD *3	MD/FINED, TxD *3

*1 EZ-CUBE 内信号名称

*2 EZ-CUBE 上焊接 16MHz 晶振，并且设置 SW-2 为 "Ext. Clock".

*3 使用 flash 编程器进行编程时，必须使用 TxD 和 RxD 引脚

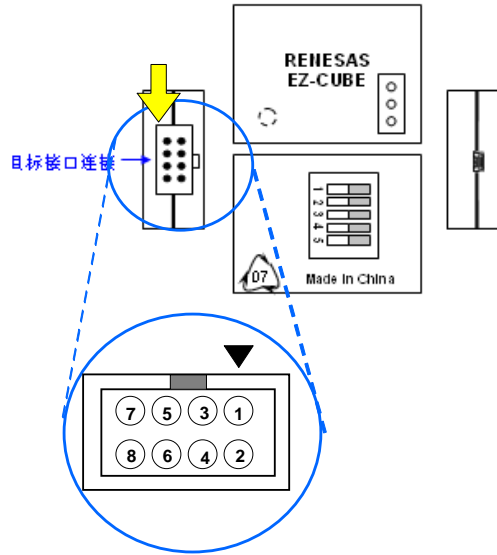


表 6-3. 引脚功能

引脚名称	IN/OUT *	描述
FINEC	OUT	通信时钟输出 (仅限 RX63T 的[R5F563T4/5/6])
MD/FINED, TxD	IN/OUT	调试通信数据输入-输出 (MD/FINED), 或 Flash 编程通信数据输出 (TxD)
RxD	IN	Flash 编程通信数据数据
RES#	IN/OUT	Reset 信号输入-输出
VCC	Power	电源电压输入 (2.7V - 5V) 或电源电压输出 (5V)
GND	Power	Ground

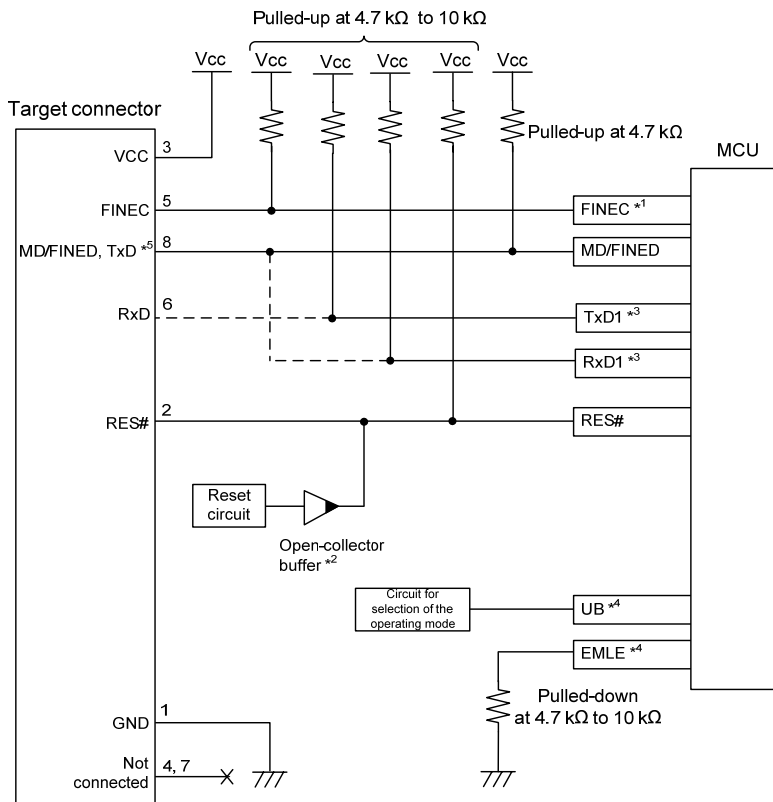
* 以EZ-CUBE为参照基准

6.1.2 电路连接举例

(1) 调试电路连接举例

图 6-2 显示了调试电路连接举例。

图 6-2. 调试电路连接示例



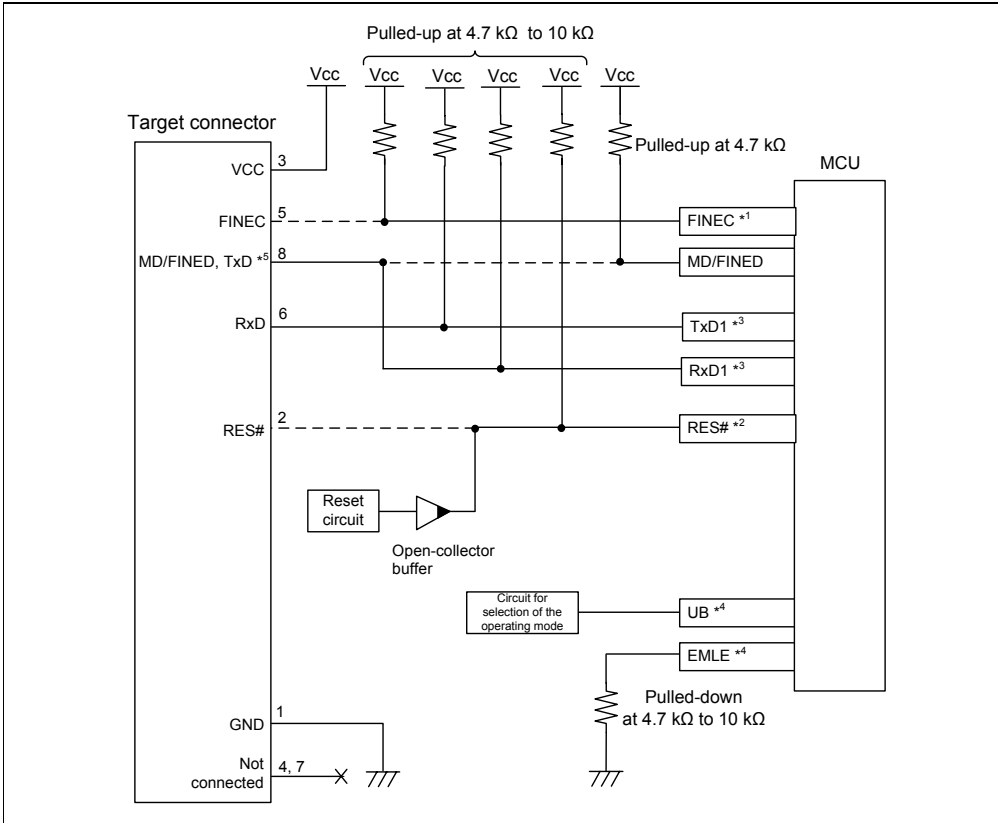
*1 调试RX63T的[R5F563T4/5/6, 需要连接FINEC 引脚。调试RX210, RX220 和RX111 时, 则不需要连接。此引脚可用做普通端口。

*2	目标系统的Reset电路输出，必须断开电容The output of the reset circuit of the target system must be open collector.
*3	调试时不使用TxD1和RxD1引脚，不必连接。
*4	参考“Notes on Connection” for detailed process of UB and EMLE pins.
*5	Connected pin of the MCU for debugging is different from for flash programming. Connect to MD/FINED pin of the MCU and pull-up RxD1 pin of the MCU.

(2) Flash 编程连接举例

图 6-3 显示了编程电路连接举例。

图6-3 Flash编程连接电路推荐



*1	FINEC 引脚不用于flash编程。可以不连接。
*2	RES# 引脚不用于flash编程。可以不连接。
*3	Flash 编程连接TxD1和RxD1引脚。
*4	UB和EMLE引脚的具体处理请参看“连接时的注意事项”。
*5	编程和调试连接有所不同。 编程时连接RxD1 引脚，并上拉MD/FINED连接。

EZ-CUBE 开关设置

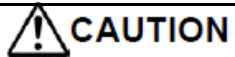
RX63T group [R5F563T4/5/6]

- SW-1 开关: 请选择“M2”.
- SW-2 开关: 请选择“Ext. Clock”，需要安装 16MHz 振荡器
- SW-3 开关: 请选择“Debug Mode”.
- SW-4 开关: 请选择“T”.
- SW-5 开关: 请选择“M3 ”.

RX210, RX220 group and RX111 group

- SW-1 开关: 请选择“M2”.
- SW-2 开关: 请选择“Int. Clock”

- SW-3 开关: 请选择“Debug Mode”.
- SW-4 开关: 请根据实际目标设备使用选择.
- SW-5 开关: 请选择“M3 ”.



注意目标系统的电源提供:



1. 在USB线缆处于连接状态时，请勿改变开关的设置.
2. 最大允许电流100mA，所以请勿将EZ-CUBE连接到电流很大的目标系统上。在EZ-CUBE与主机连通后，主机始终保持向EZ-CUBE供电.
3. RX111操作电压为3.6V

6.1.3 连接时的注意事项

(1) 关于FINEC和MD/FINED引脚

RX63T, FINE只支持2-Wire,使用FINEC和MD/FINED引脚。FINEC和MD/FINED引脚专门用于仿真器。FINEC引脚的任何复用功能都无法使用。

RX210, RX220及RX111, FINE支持一个1-Wire, 使用MD/FINED。只有MD/FINED引脚专门用于仿真器。不使用可以不连接FINEC引脚。FINEC引脚可以用来作为一个端口。

FINEC引脚上拉4.7kΩ到10kΩ。MD/FINED上拉4.7kΩ。不要将这些信号线平行或穿过其他高速信号线。MD/FINED信号和TXD信号被分配到相同的引脚。MCU调试时, 该管脚连接到MD/FINED引脚。

(2) 关于TXD和RXD引脚

TXD和RXD信号不能用于调试。这些引脚仅用于瑞萨Flash编程器的内部Flash编程。

如果MCU有多个TXD1或RXD1引脚, 确认各自的引脚在MCU的硬件手册Boot模式。

MD/FINED信号和TXD信号被分配到相同的引脚。使用Renesas Flash编程器时, 该引脚连接到RXD1引脚用于内部Flash编程。

(3) 关于RES引脚

仿真器使用RES#引脚。如果用户系统中有逻辑复位电路, 必须用开路缓冲器将仿真器连接器的RES#信号和用户逻辑的输出信号连接。在没有用户复位电路的情况下, 必须将仿真器连接器的RES#信号直接连接到单片机的RES#引脚。

(4) 关于EMLE引脚 (仅RX63T [R5F563T4/5/6])

在目标系统EMLE引脚下拉4.7 kΩ到10 kΩ。

(5) 关于UB引脚

UB引脚是进入用户引导模式和USB I/F模式的端口。具体哪个端口是UB引脚取决于使用的MCU。请参考使用的MCU硬件手册操作模式章节。

调试RX63T[R5F563T4/5/6]时, 因为没有用户引导模式, 所以可以不用对引脚进行处理。

调试RX210和RX220时, 在目标系统上UB引脚下拉4.7kΩ到10kΩ, 不会到用户引导模式上。

调试RX111时, 在目标系统上UB引脚上拉4.7kΩ到10kΩ, 不会到用户引导模式上。

(6) 关于VCC

将目标系统的VCC (电源) 连接到仿真器连接器的VCC。

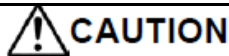
在电源电压为2.7V至5.5V的MCU的工作电压范围内使用模拟器。

EZ-CUBE可以提供5V电源一个简单的评价体系。最多可提供100 mA电流。从仿真器供给5V电源的情况下, 设置SW-4为“5”。

使用EZ-CUBE供电功能, 当检查到目标系统的供电电压。电压可能下降0.5V或更多, 因为它依赖于USB的VBUS电源电压。

(7) 关于GND

连接器的GND必须和MCU的VSS引脚的GND相同。



注意电源开/关:

在提供电源时, 必须确认用户系统的电源电路没有短路。必须在确认仿真器连接器的引脚排列没有问题后连接EZ-CUBE。如果发生错误连接, 主机、仿真器和用户系统就可能冒烟起火。

6.2 片上调试

本节描述了使用 EZ-CUBE 进行片上调试的系统配置，启动/关闭顺序和调试中的若干注意事项。

6.2.1 调试功能

表 6-4 列出了以 RX 微控制器作为目标设备，进行调试时的调试功能。

表 6-4. 调试功能

功能		规格定义		
		RX600 系列	RX200 系列	RX100 系列
目标 MCU		RX63T群 [R5F563T4/5/6]	RX210, RX220 群	RX111群
支持的电压		2.7 - 3.6 V	2.7 - 5 V	2.7 - 3.6 V
最大操作频率		100 MHz	50 MHz	32 MHz
操作模式		Single chip 模式		
下载		可用		
执行		Go, Stop, Step in, Step over, Return out, CPU Reset, Restart		
软件中断		最多256点		
事件	执行地址	最多8点	最多4点	
	数据访问	最多4点	最多2点	
	通过数	最多256次	无	
On-chip 中断	执行前PC中断	最多8点	最多4点	
	事件组合	OR/AND(累计)顺序		
	其他	跟踪中断		
跟踪(内部跟踪)		最多256个	最多64个(RX210) 最多32个(RX220)	最多32个
性能测试		最多2点的执行周期或者执行 次数(32位计数器 x 2)	最多1点的执行周期 (24位 计数器 x 1)	无
调试通信接口		FINEC, MD/FINED	MD/FINED	

6.2.2 调试时的注意事项

(1) 用户程序执行过程中复位

如果用户程序执行期间发生的内部复位，就不可能控制模拟器。不会产生如看门狗定时器产生的内部复位。

当目标系统的执行过程中发生了引脚复位，用户程序可能会挂起。如果检测引脚复位输入，输入从仿真器复位并重新执行用户程序，但它可能会成为一个中断状态，无法重新执行。

在仿真系统（[Memory]Memory 引用）复位（通过一个引脚复位，看门狗定时器等）和操作的情况下显示错误消息“超时错误。MCU 处于复位状态。系统复位发出？”。仿真器初始化和用户程序停止。系统复位后发出后，跟踪记录初始化。可以继续调试。

(2) 目标系统复位取消时间

超过的 VIH 电压在 100ms 内取消的复位输入。

(3) 对用于调试的设备的处理

由于调试时设备内部的 Flash 会被多次擦写并且无法确保擦写的次数，因此，不要将用于调试的设备安装在量产的产品中。此外，也不要将调试监控程序编程至量产的产品中。

(4) 高速时钟振荡器 (HOCO)

模拟器使用设备的内部高速时钟振荡器 (以下简称 HOCO) ,RX200 系列和 RX100 系列 MCU 通过 FINE 接口来实现通信。因此，不管相关寄存器设置，HOCO 始终处于振荡状态。

如果有一个 HOCO 频率和存储器存取之间的争用，存储器存取操作不能保证。

(5) 用户程序的最终评价

进入大规模生产阶段，被用编程器写入到 Flash memory ROM 之前，一定要进行程序的最终评估。请务必单独进行评估，不连接模拟器。

6.3 Flash 编程

本节将介绍使用EZ-CUBE对RX微控制器的Flash进行编程时的系统配置以及启动/关闭顺序。

6.3.1 编程功能的规格定义

表6-5. 编程功能的规格定义

功能	规格定义说明
主机接口	USB 2.0
目标接口	UART
目标系统电压	2.7到5.5 V（根据目标设备而定）
时钟源	可以提供8 MHz时钟信号 可以使用安装在目标系统上的时钟
电源	5 V ± 0.3V（最大电流：100 mA）
获取特定设备信息	使用Renesas Electronics的RX参数文件
编程软件	RFP [®] (Renesas Flash Programmer)
安全标志的设置	有
独立操作	无（必须连接主机）

注：详情请见RFP用户手册

6.3.2 Flash 编程时的注意事项

本节将介绍进行Flash编程时需要注意的事项。请务必仔细阅读本节内容以正确使用EZ-CUBE。

- 为提高写入的编程的质量，请在使用EZ-CUBE之前了解以下内容并队列出的各项进行校验和评估。
 - 务必按照设备及EZ-CUBE的用户手册中给出的方法进行目标系统的电路的设计。
 - 务必按照设备、RFP及EZ-CUBE各自的用户手册中介绍的方法进行操作。
 - 务必向目标系统提供稳定的电源。

第 7 章 如何在 V850 微控制器上使用 EZ-CUBE

本章描述了如何使用 EZ-CUBE 在 V850 微控制器上进行片上调试和 Flash 编程。

片上调试是指通过芯片内置的调试功能对已经装载在目标系统上的微控制器进行调试的方法。由于这种调试方法是对板上的目标设备进行操作的，所以非常适合现场调试。

Flash 编程是将程序写入设备内置的 Flash 存储器的方法。可以在板上对设备进行擦除、写入和校验。

请先将 EZ-CUBE 的固件更新为用于 V850，具体请参照 (1) - (3)。详细内容请参见 **1.4 固件更新**。

- (1) 使用 USB 缆连接 EZ-CUBE 和 PC。**不要将 EZ-CUBE 与目标板相连。**
- (2) 启动 [QBEZUTL.exe]，并指定“V850_OCD_FW.hex”固件文件。
- (3) 点击“Start”按钮，然后指定用于 V850 固件下载到 EZ-CUBE。

如果首次用 V850 微控制器作为目标设备来使用 EZ-CUBE，那么请仔细阅读以下内容。

7.1 目标系统设计

要用 EZ-CUBE 和目标系统进行通信，必须在目标系统上安装对应的通信电路。本节描述了相关电路的设计以及接口的安装。

7.2 片上调试

本节描述了用 EZ-CUBE 进行片上调试的系统配置和启动方法。

7.3 Flash 编程

本节描述了用 EZ-CUBE 进行 Flash 编程的系统配置和启动方法。

支持 V850 MCU 列表

表 7-1 显示了 EZ-CUBE 支持的 V850 系列 MCU 列表及固件。

表 7-1 EZ-CUBE 支持 V850 MCU 列表及固件

项目	内容	固件
支持 MCU	V850ES/Jx3	V850_OCD_FW.hex
	V850ES/Jx3-L	
支持的操作模式	UART 模式	

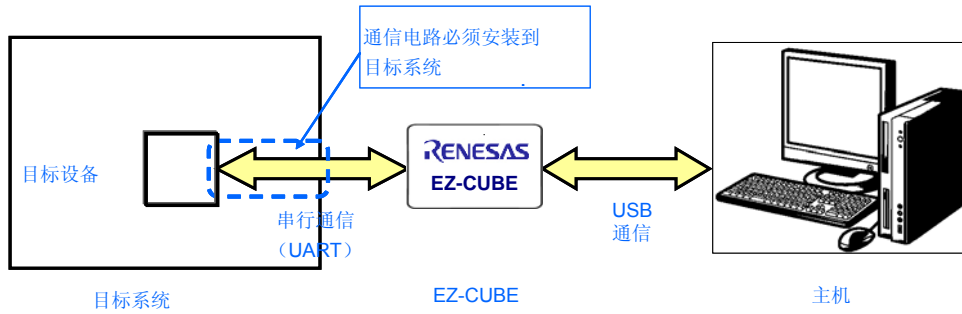
7.1 目标系统设计

为了实现片上调试和Flash编程功能，需要设计目标系统的电路，本节描述了相关的电路设计。

图7-1为EZ-CUBE通信的整体接口框图。如图所示，EZ-CUBE与目标系统的目标设备之间进行串行通讯。为了实现通讯，目标系统上必须安装用于通信的电路。请参考本节的内容设计合适的电路。

这里用于串行通信的引脚基本上和使用Flash编程器（例如EZ-CUBE）时使用的串行通信引脚相同，但是有一些设备并不支持这些引脚。

图7-1. 通信接口概况



7.1.1 引脚配置

本节将说明EZ-CUBE和目标系统之间使用的接口信号。表7-2中列出了引脚的配置情况。表7-3说明了各个引脚的功能。

表 7-2. 引脚配置

引脚编号	引脚名称 ^注
1	GND
2	RESET_IN
3	Vdd
4	FLMD0
5	CLK
6	RxD.
7	RESET_OUT
8	TxD

注 EZ-CUBE 内的信号名称。

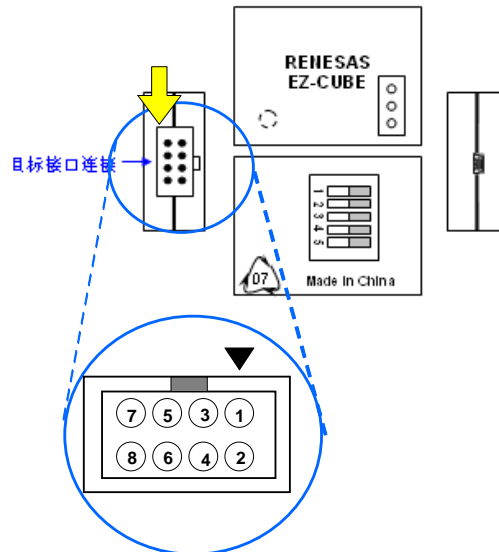


表 7-3. 引脚功能

引脚名称	IN/OUT*	描述												
RESET_IN	IN	该引脚用于从目标系统输入 reset 信号												
RESET_OUT	OUT	该引脚用于向目标系统输出 reset 信号	CLK	OUT	该引脚用于向目标系统输出时钟信号	FLMD0	OUT	该引脚用于设定目标设备进入调试模式或编程模式	RxD	IN	该引脚用于接收来自目标设备的命令/数据	TxD	OUT	该引脚用于向目标设备发送命令/数据
CLK	OUT	该引脚用于向目标系统输出时钟信号												
FLMD0	OUT	该引脚用于设定目标设备进入调试模式或编程模式												
RxD	IN	该引脚用于接收来自目标设备的命令/数据												
TxD	OUT	该引脚用于向目标设备发送命令/数据												

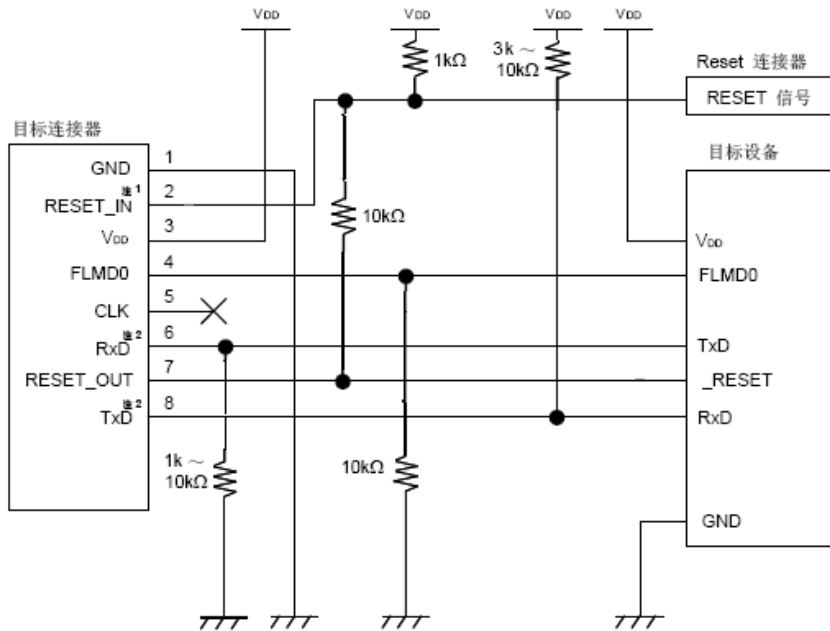
注 以 EZ-CUBE 为参照基准。

7.1.2 电路连接举例

目标系统上的电路设计会根据接口信号而有所不同。接口的引脚名称可能会因目标设备有所不同。详细信息请参考目标设备的用户手册。参照表 7-2 和相关的电路连接示例。

注意事项 在电路连接示例中出现的常数值为参考值。如果进行 flash 编程旨在量产，请彻底评估目标设备的规格是否满足要求。

图 7-2. 电路连接示例



- 注：
1. 该连接的设计是假定 RESET 信号从 N 沟道开漏缓冲器（输出阻抗：100 kΩ或更小）输出的。详细信息请参考 7.1.3 reset 引脚的连接。
 2. 将目标设备的 TxD（发送方）连接到目标连接器的 RxD（接收方），同时将目标连接器的 TxD（发送方）连接到目标设备的 RxD（接收方）。

EZ-CUBE 开关设置

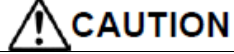
SW-1 开关: 请选择“M1”.

SW-2 开关: 请选择“Int. Clock”.

SW-3 开关: 请选择“Debug Mode”.

SW-4 开关: 请根据实际目标设备使用选择.

SW-5 开关: 请选择“Other ”.



注意目标系统的电源提供:



1. 在USB线缆处于连接状态时，请勿改变开关的设置.
2. 最大允许电流100mA，所以请勿将EZ-CUBE连接到电流很大的目标系统上。在EZ-CUBE与主机连通后，主机始终保持向EZ-CUBE供电.

7.1.3 reset 引脚的连接

本节将介绍上一节的电路连接举例中提到的需要特别注意的reset引脚的连接方法。

片上调试时，来自目标系统的reset信号输入EZ-CUBE被屏蔽后，再输出到目标设备。因此reset信号的连接根据是否连接EZ-CUBE而有所不同。

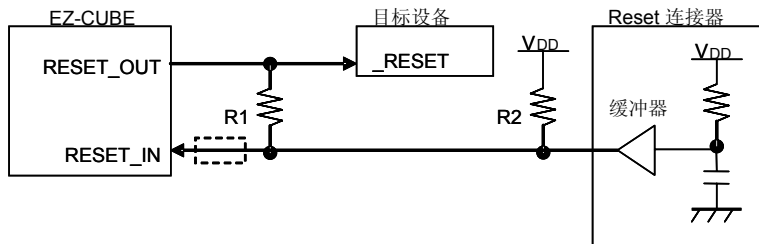
Flash编程时，电路的设计必须保证目标系统的reset信号和EZ-CUBE的reset信号不冲突。

推荐通过电阻自动切换 reset 信号。图 7-3 说明了 7.1.2 电路连接示例中描述的 reset 引脚连接情况。

图7-3说明了7.1.2 电路连接示例中描述的reset引脚的连接情况。

这种连接的设计是假定目标系统的复位电路包含有N沟道开漏缓冲器（输出阻抗：100Ω或更小）。当EZ-CUBE的RESET_IN/OUT逻辑反转时，VDD或GND的电平可能会不稳定，所以请注意备注中所描述的条件。

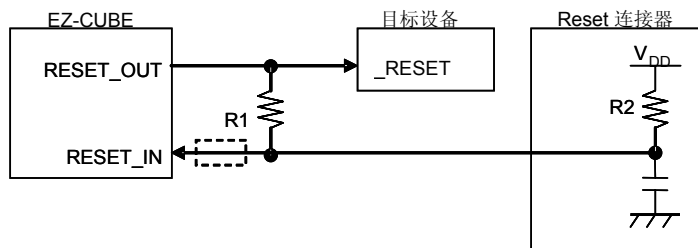
图 7-3. 包含缓冲器的复位电路的连接



备注 需确保R1的阻值至少是R2阻值的十倍，R1为10 kΩ或更大。
如果复位电路的缓冲器是CMOS输出，则无需R2上拉电阻。
当仅进行Flash编程时，虚线框内的电路就可以略去。

图7-4 所示的电路为目标系统的复位电路中不包含缓冲器，并且reset信号仅通过电阻或电容产生。设计电路时注意备注中描述的条件。

图 7-4. 不包含缓冲器的复位电路的连接



备注 需确保R1的阻值至少是R2阻值的十倍，R1为10 kΩ或更大。
当仅进行Flash编程时，虚线框内的电路就可以略去。

7.2 片上调试

本节描述了使用 EZ-CUBE 进行片上调试的系统配置，启动/关闭顺序和调试中的若干注意事项。

7.2.1 调试功能

表 7-4 列出了以 V850 微控制器作为目标设备，进行调试时的调试功能。

表 7-4. 调试功能

功能	规格定义
目标 MCU	V850系列 V850ES/Jx3, V850ES/Jx3-L
安全	10 字节 ID 码验证
下载	支持
执行	全速执行，忽略断点执行，单步执行，下一步执行，运行至此，重启
硬件断点	1 个
软件断点	多个
RAM 监视	支持
调试所用的功能引脚	RxD, TxD

7.2.2 保留用户资源和设置安全 ID

为实现目标设备与 EZ-CUBE 之间的通信以及各种调试功能，用户必须做以下几点准备。请参照本小节的内容修改用户程序或编译选项，以完成准备工作。

(a) 设置安全 ID

该功能用于防止存储器被盜读。内部 Flash 的地址 0x70 到 0x79 上会嵌入一个安全 ID。只有在调试器启动时输入的安全 ID 与内部 Flash 中地址 0x70 到 0x79 的安全 ID 一致时，调试器才可以正常启动。

但是，如果地址 0x79 的第 7 位的值为“0”，那么调试功能将被禁止使用。这种情况下，调试器将无法启动。对于量产的设备而言，通常调试功能都是被禁止的。

如果用户忘记了安全 ID 的值或希望重新打开调试功能，那么只有将 Flash 全部擦除后再次设置安全 ID。

[如何设置安全 ID]

在用户程序地址 0x70 到 0x79 区域设置安全 ID。

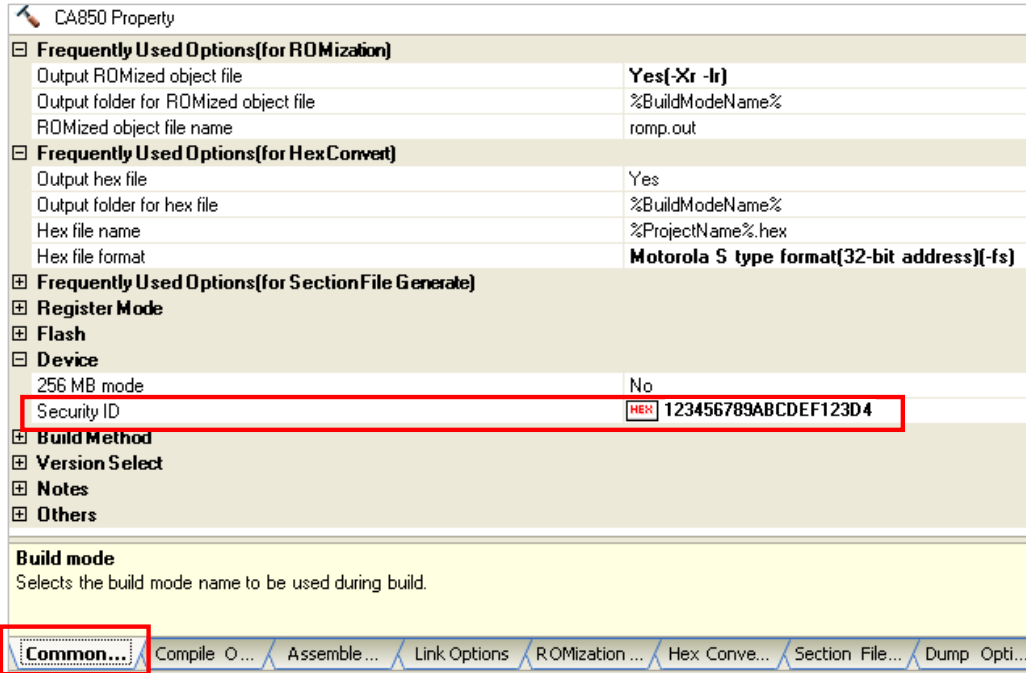
用户程序在地址 0x70 到 0x79 设置。举例，安全 ID 设置为“0123456789ABCDEF1234”（不区分大小写）

Address	Value
0x70	0x12
0x71	0x34
0x72	0x56
0x76	0x78
0x74	0x9A
0x75	0xBC
0x76	0xDE
0x77	0xF1
0x78	0x23

0x79	0xD4
------	------

如果使用Renesas编译器，在通用选项中设置安全ID。

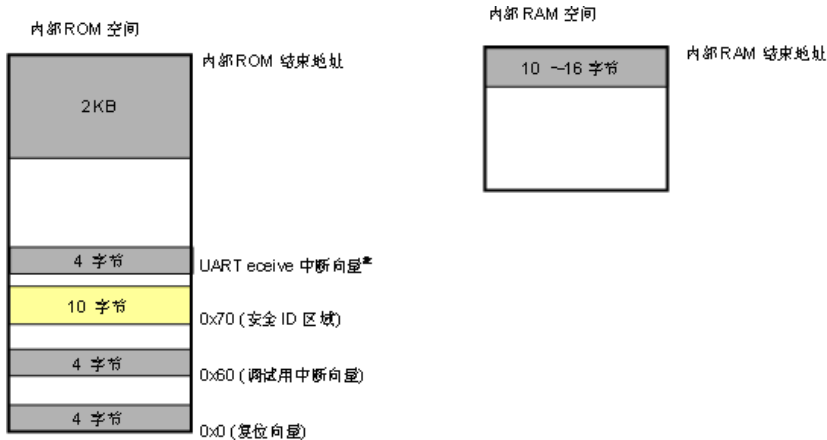
图7-5. 设置安全ID



• 预留存储器空间

图7-6中的阴影部分是存放调试监控程序的保留区域，因此，用户程序和数据不能分配到这些空间。

图 7-6. 分配给调试监控程序的存储器空间



■：调试区域

注： 如果存在接收错误中断的向量或接收状态中断的向量，必须保证是安全的。

- **通讯用的串行接口保留**

不能改变调试监控程序执行时，用作 EZ-CUBE 和目标设备间通讯的 UART 的状态。

(b) 复位向量

复位向量中保存着向调试监控程序跳转的跳转指令信息。

[如何预留区域]

如果用户程序不使用这块区域就没有必要一定要保留这块区域。下载程序时，调试器会自动以以下几种形式将复位向量改写。如果改写的形式与以下几种形式不符，那么调试器将产生一个错误警告。

- 从地址 0 开始连续放置两条 nop 指令

写入之前		写入之后
0x0 nop	→	跳至 0x0 地址上的调试监控程序
0x2 nop		0x4 xxxx
0x4 xxxx		

- 从地址 0 开始连续放置两条 0xFFFF 数据（被擦除过的设备）

写入之前		写入之后
0x0 0xFFFF	→	跳至 0x0 地址上的调试监控程序
0x2 0xFFFF		0x4 xxxx
0x4 xxxx		

- 地址 0 处放置 jr 指令（当使用 Renesas Electronics 的编译器时）

写入之前		写入之后
0x0 jr disp22	→	跳至 0x0 地址上的调试监控程序
		0x4 jr disp22 - 4

- 地址 0 处放置向调试监控程序跳转的指令

写入之前		写入之后
跳至 0x0 地址上的调试监控程序	→	不变化

(c) 调试监控程序的保留区域

调试监控程序执行对调试通讯接口的初始化处理和对 CPU 的运行或中断处理。内部 ROM 区域必须填满 0xFF。这个区域千万不要被用户程序重写。

[如何预留区域]

如果用户程序并未使用这一区域，那么用户也不必特意进行预留操作。

但是为了确保调试器启动时不会产生问题，我们还是推荐用户使用编译器将此区域预留。

下面是使用 Renesas Electronics 编译器 CA850 预留该方法举例。请在工程中加入如下所示的汇编语言源文件和链接指示代码。

- 汇编语言源文件（将以下代码写入汇编源文件中并加入到工程）

```
--为监控 ROM 节预留 2 KB 空间
.section "MonitorROM", const
.space 0x800, 0xff

--预留调试用的中断向量
.section "DBG0"
.space 4, 0xff

--预留串行通信用的中断向量
--请根据所使用的串行通信模式更改节名称
.section "INTCB0R"
.space 4, 0xff

--为监控 RAM 节预留 16 字节的空间
.section "MonitorRAM", bss
.lcomm monitorramsym, 16, 4 /* 定义 monitorramsym 符号 */
```

注 如果用语句“monitorramsym:”替换该行，仅执行一个符号定义，则下载速度会被提升。如果数值被填入一个洞（没有编码的区域），该效果不再显现。填入时，填入值必须是 0xFF 用于预留空间。

- 链接伪指令（将以下代码加入链接指示文件中）

下面的代码是以内部 ROM 结尾地址为 0xffff、内部 RAM 结尾地址为 0x3ffff 为例。

```
MROMSEG : !LOAD ?R V0x0ff800{
  MonitorROM = $PROGBITS ?A MonitorROM;
};

MRAMSEG : !LOAD ?RW V0x03ffff0{
  MonitorRAM = $NOBITS ?AW MonitorRAM;
};
```

(d) 预留串行通信接口

UART 用作 EZ-CUBE 和目标系统间的通讯。和 UART 相关的设置通过调试监控程序来执行，但是如果这个设置被用户程序改变，会产生通讯错误。

为了防止这种问题的发生，在用户程序里通讯串行接口必须保留。

[如何对串行通信接口进行预留]

请参考以下内容创建用户程序。

- 串行接口寄存器
不要在用户程序中对 UART 的相关寄存器进行设置。
- 中断屏蔽寄存器
使用 UART 时，不要屏蔽接收完成中断[※]。
注 不要屏蔽接收错误中断或者接收状态中断。
- 端口寄存器
使用 UART 时，不要将端口寄存器设置为 TxD 和 RxD 以外的模式。

7.2.3 调试时的注意事项

本节介绍使用 V850 微控制器的片上调试功能时需要注意的事项。

请仔细阅读以下事项以正确使用 EZ-CUBE。

- (1) 对用于调试的设备的处理
由于调试时设备内部的 Flash 会被多次擦写并且无法确保擦写的次数，因此，不要将用于调试的设备安装在量产的产品中。此外，也不要将调试监控程序编程至量产的产品中。
- (2) 下载注意事项
在调试时，CPU 复位才能下载。如果程序下载到 flash memory 进行 DMA 传输到内部 RAM，下载的程序可能无法正常执行。Break 不能执行。
- (3) 关于 ROM 校正功能
不要使用 ROM 校正功能，或发生其他意想不到的 break。
- (4) 关于电流消耗
在调试过程中，目标设备中的电流消耗比正常操作模式增加。因为在调试过程中，目标设备运行 OCD 单元。
- (5) 关于释放待机状态的调试功能
RRM 功能和 DMM 功能使用的情况下，内存读取或写入时，待机模式被释放。
- (6) Flash 自编程注意事项
Flash 环境期间不要进行 ROM 区中断。在监控 memory 的 RRM 功能情况下，执行临时中断。所以使用 flash 自编程时不要使用 RRM 功能。
使用调试接口不要修改调试监控区时。
- (7) 关于的 POC 功能和关闭仿真
确保在调试期间，不关闭目标系统的电源。在连接仿真器时检查 POC 功能的操作和关闭。在目标系统关断瞬间，调试器可能会挂起。

(8) 关于当使用屏蔽复位时 I/O 缓冲器

当引脚输入复位信号时，即使屏蔽功能屏蔽了复位，根据在目标设备，I/O 缓冲器（端口引脚）可进入复位状态。

(9) 当无法执行断点、RRM 功能和 DMM 功能操作时

以下情况发生时，将无法执行强制断点功能、RRM 功能和 DMM 功能。

- 禁止中断时 (DI)
- UART 接口中断被屏蔽
- 当设置了“禁止可屏蔽中断唤醒待机模式”时进入了待机模式
- 主时钟信号停止时，通信模式使用 UART 接口

(10) flash 编程的质量

为了提高写入质量，使用 EZ-CUBE 仿真器前充分理解、验证和评估。

- 该设备的用户手册中所描述的电路的设计。
- 用于每个用户的手册中所描述的设备 and 软件。
- 目标系统供电稳定。

(11) 使用实机调试

如果使用实机调试而不使用仿真器，使用编程软件写入用户程序。通过调试器下载程序包括监控程序，如果不通过 EZ-CUBE 控制可能发生程序故障。

(12) 关于看门狗定时器

调试监控程序强制停止看门狗定时器。因此，不能使用选项字节指定看门狗定时器停止。选项字节设置的详细信息，请参阅目标设备的用户手册。

(13) 关于外部复位

当发生外部复位（复位被屏蔽时除外）发生 **break**，或发生内部复位。

(14) 关于复位向量处理

不支持复位向量处理。

7.3 Flash 编程

本节将介绍使用 EZ-CUBE 对 V850 微控制器的 Flash 进行编程时的系统配置以及启动/关闭顺序。

7.3.1 编程功能的规格定义

表7-5. 编程功能的规格定义

功能	规格定义说明
主机接口	USB 2.0
目标接口	UART
目标系统电压	2.7到5.5 V (根据目标设备而定)
时钟源	可以提供8 MHz时钟信号 可以使用安装在目标系统上的时钟
电源	5 V \pm 0.3V (最大电流: 100 mA)
获取特定设备信息	使用Renesas Electronics的V850参数文件
编程软件	RFP [®] (Renesas Flash Programmer)
安全标志的设置	有
独立操作	无 (必须连接主机)

注: 详情请见 RFP 用户手册

7.3.2 Flash 编程时的注意事项

本节将介绍进行 Flash 编程时需要注意的事项。请务必仔细阅读本节内容以正确使用 EZ-CUBE。

- 为提高写入的编程的质量, 请在使用 EZ-CUBE 之前了解以下内容并队列出的各项进行校验和评估。
 - 务必按照设备及 EZ-CUBE 的用户手册中给出的方法进行目标系统的电路的设计。
 - 务必按照设备、RFP 及 EZ-CUBE 各自的用户手册中介绍的方法进行操作。
 - 务必向目标系统提供稳定的电源