

## R7F0C019

R01AN3058CC0100

Rev.1.00

## DMA 控制器（UART 连续接收） CC-RL

2015.12.31

### 要点

本篇应用说明介绍了利用 DMA 控制器进行 UART 连续接收的方法。将 UART 接收到的数据，通过 DMA 控制器按照顺序保存到内部 RAM 中。

### 对象 MCU

R7F0C019

本篇应用说明也适用于其他与上面所述的 MCU 具有相同 SFR（特殊功能寄存器）定义的产品。关于产品功能的改进，请参看手册中的相关信息。在使用本篇应用说明的程序前，需进行详细的评价。

## 目录

|                                 |    |
|---------------------------------|----|
| 1. 规格 .....                     | 3  |
| 2. 动作确认条件 .....                 | 5  |
| 3. 硬件说明 .....                   | 6  |
| 3.1 硬件配置示例 .....                | 6  |
| 3.2 使用引脚一览 .....                | 6  |
| 4. 软件说明 .....                   | 7  |
| 4.1 操作概要 .....                  | 7  |
| 4.2 选项字节设置一览 .....              | 8  |
| 4.3 常量一览 .....                  | 8  |
| 4.4 变量一览 .....                  | 9  |
| 4.5 函数一览 .....                  | 9  |
| 4.6 函数说明 .....                  | 10 |
| 4.7 流程图 .....                   | 14 |
| 4.7.1 整体流程 .....                | 14 |
| 4.7.2 初始化函数 .....               | 14 |
| 4.7.3 系统函数 .....                | 15 |
| 4.7.4 初始化端口 .....               | 16 |
| 4.7.5 CPU 时钟设置 .....            | 18 |
| 4.7.6 SAU0 初始化设置 .....          | 19 |
| 4.7.7 UART0 初始化设置 .....         | 20 |
| 4.7.8 DMA 控制器的初始化设置 .....       | 21 |
| 4.7.9 主函数处理 .....               | 27 |
| 4.7.10 主函数初始化 .....             | 28 |
| 4.7.11 DMA 控制器用户初始化设置 .....     | 29 |
| 4.7.12 UART0 通信开始处理（屏蔽中断） ..... | 30 |
| 4.7.13 DMA0 动作开始 .....          | 34 |
| 4.7.14 DMA0 传送结束中断 .....        | 37 |
| 4.7.15 UART0 接收错误中断 .....       | 38 |
| 4.7.16 DMA0 软件触发传送开始处理 .....    | 39 |
| 5. 参考例程 .....                   | 40 |
| 6. 参考文献 .....                   | 40 |
| 公司主页和咨询窗口 .....                 | 40 |

1. 规格

本篇应用说明介绍了使用 DMA 控制器，传送 UART 通信连续接收数据的使用方法。使用串行阵列单元（SAU）进行 UART 通信，接收来自对方设备发送过来的 ASCII 码数据（1 个字节×5 次）。然后，使用 DMA 控制器将数据从 SFR 传送到 RAM 中。以 UART 接收结束中断作为触发，将接收数据保存到 RAM 中。达到之前设定的传送次数（5 次）后，将接收数据转换为可以看作“5 位数的 10 进制表示 ASCII 码”的数值。一旦接收数据的累计值超过阈值（100000）后，将点亮 LED。

相关外围功能和用途，请参见“表 1.1”。使用 DMA 控制器进行 UART 连续接收的操作概要，请参见“图 1.1”，DMA 控制器的时序图，请参见“图 1.2”。

表 1.1 相关外围功能及用途

| 外围功能    | 用途                     |
|---------|------------------------|
| DMA 控制器 | 将 UART 接收数据传送到内部 RAM 中 |
| 串行阵列单元  | 进行 UART 接收             |

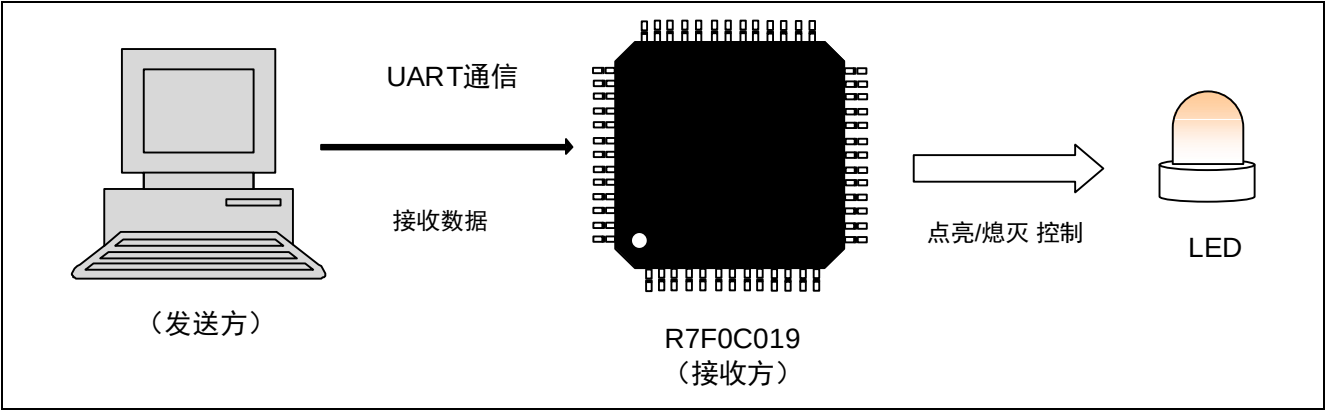


图 1.1 操作概要

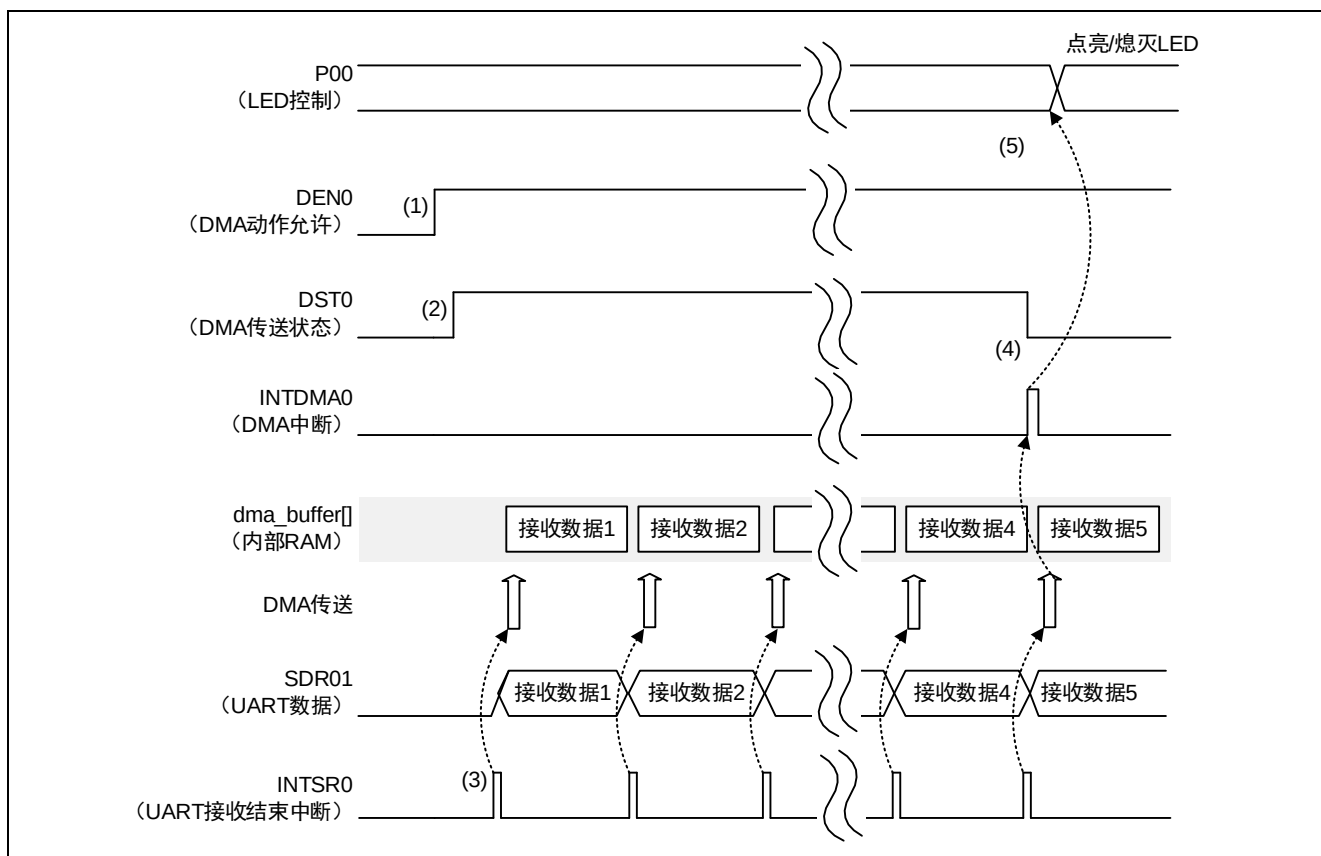


图 1.2 DMA 控制器的时序图

- (1) 设定  $DEN0 = 1$ ，DMA 进入动作允许状态。
- (2) 进行 DMA 的各种设定后，设定  $DST0 = 1$ ，进入 DMA 触发等待状态。
- (3) UART 接收结束中断 (INTSR0) 触发 DMA，将串行数据寄存器 01 (SDR01) 中的值传送到 dma\_buffer (内部 RAM) 中。
- (4) 当 DMA 传送次数达到规定次数 (5 次) 时，解除 DMA 触发等待状态 ( $DST0 = 0$ )，产生 DMA 中断 (INTDMA0)。
- (5) DMA 中断 (INTDMA0) 产生后，将 dma\_buffer 中保存的接收数据转换为数值。根据累计值控制 P00 的输出，并点亮或者熄灭 LED。

## 2. 动作确认条件

本应用说明中的参考例程，是在下面的条件下进行动作确认的。

表 2.1 动作确认条件

| 项目     | 内容                                                                                                             |
|--------|----------------------------------------------------------------------------------------------------------------|
| 所用微控制器 | R7F0C019L                                                                                                      |
| 工作频率   | 高速内部振荡器（HOCO）时钟：24MHz<br>CPU/外围功能时钟：24MHz                                                                      |
| 工作电压   | 5.0V（工作电压范围：2.9V~5.5V）<br>LVD 工作模式（V <sub>LVD</sub> ）：复位模式<br>上升沿：2.81V（2.76V~2.87V）<br>下降沿：2.75V（2.70V~2.81V） |
| 集成开发环境 | CS+ for CC V3.02.00（瑞萨电子开发）                                                                                    |
| C 编译器  | CC-RL V1.01.00（瑞萨电子开发）                                                                                         |

3. 硬件说明

3.1 硬件配置示例

本篇应用说明中使用的硬件配置示例，请参见“图 3.1”。

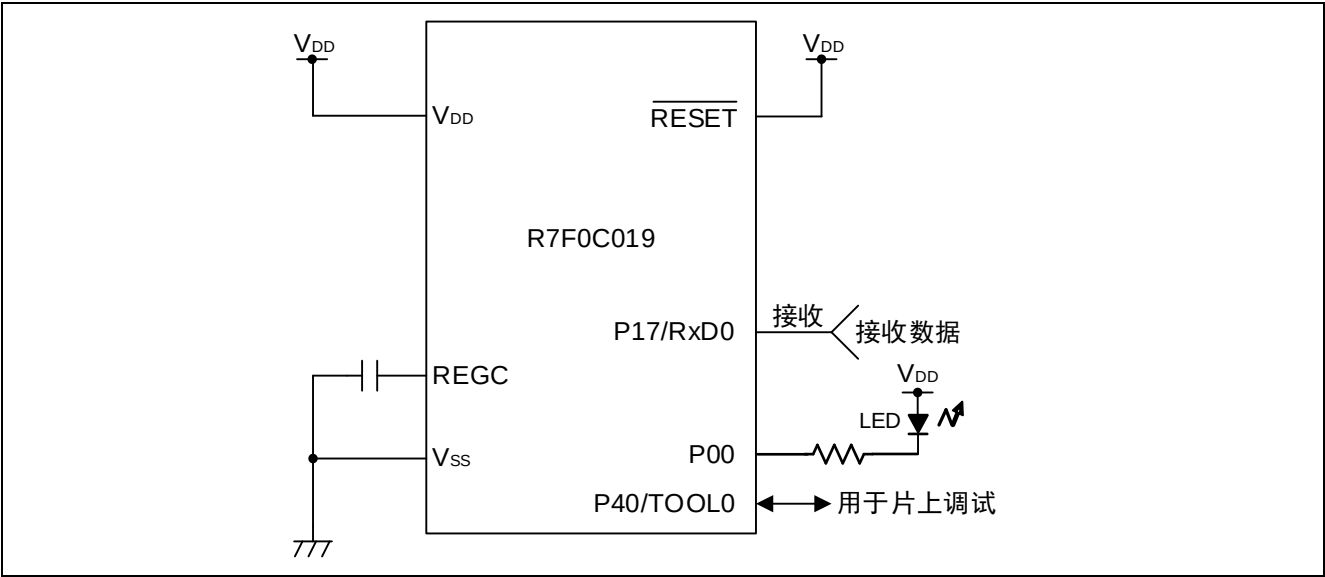


图 3.1 硬件配置

注意：1. 上述硬件配置图是为了表示硬件连接情况的简化图。在实际电路设计时，请注意根据系统具体要求进行适当的引脚处理，并满足电气特性的要求（输入专用引脚请注意分别通过电阻上拉到  $V_{DD}$  或是下拉到  $V_{SS}$ ）。

2. 请将  $V_{DD}$  电压值保持在由 LVD 设定的复位解除电压  $V_{LVD}$  以上。

3.2 使用引脚一览

使用的引脚及其功能，请参见“表 3.1”。

表 3.1 使用的引脚及其功能

| 引脚名            | 输入/输出 | 内容         |
|----------------|-------|------------|
| P17/SEG42/RxD0 | 输入    | 串行数据接收用引脚  |
| P00/SEG43      | 输出    | LED 点灯控制端口 |

## 4. 软件说明

### 4.1 操作概要

本应用说明中，使用 DMA 控制器，将 UART 接收到的数据保存到内部 RAM 中。DMA 控制器传送完 5 字节的数据后，将内部 RAM 中保存的接收数据转换为 5 位的数值。当累计值超过 100000 时，点亮 LED，没有达到时，熄灭 LED。

（1）进行 DMA 控制器的初始化设定

<设定条件>

- 设定传送方向为 SFR→内部 RAM
- 设定 DMA 启动源为 UART 接收传送结束中断请求（INTSR0）
- 设定传送数据长度为 8 位
- 设定传送源 SFR 的地址为 SDR01 的地址（0x000FFF12）
- 设定传送目标 RAM 的地址为变量 dma\_buffer[]的起始地址
- 设定传送次数为 5 次

（2）进行 SAU0 的初始化设定

<设定条件>

- 设定 SAU0 为 UART 模式，只接收
- 奇偶校验设定为偶校验
- 设定数据传送顺序为 LSB 优先
- 设定数据长度为 8 位
- 接收数据电平设定为标准
- 设定传送速度为 9600bps
- 设定错误中断（INTSRE0）有效
- 设定 INTSRE0 的中断优先级为低优先级
- 中断源使用 UART 接收的传送结束中断（INTSR0）

（3）将通道 1 的动作开始触发（SS01 位）设定为“1”，UART 进入通信待机状态。屏蔽传送结束中断（SRMK0 = 1），禁止 INTSR0 中断处理。但是，为了防止接收错误发生时的溢出错误，允许错误中断（INTSRE0）的向量中断处理。

（4）设定 DMA 控制器进入触发等待状态。

（5）执行 HALT 命令，进入 HALT 模式，等待 DMA 传送结束中断（INTDMA0）的发生。

（6）传送结束中断请求（INTSR0）发生，DMA 控制器更新接收数据。

（7）错误中断（INTSRE0）发生时，向量中断处理内，保存错误状态，并且软件触发 DMA 进行传送。

（8）DMA 传送结束中断（INTDMA0）解除 HALT 模式（UART 接收错误发生时，将发生错误时接收的数据在内的 5 字节传送结束后，产生 DMA0 传送结束中断（INTDMA0））。此后，确认错误状态。UART 接收没有发生错误时，执行（9）的处理。UART 接收发生错误时，清除错误状态，执行（11）的处理。

（9）接收数据（5 字节）转换为 ASCII 码表示的 5 位 10 进制数值。

（10）当（9）的累计值超过 100000 的时候，清除累计值，点亮 P00 连接的 LED。累计值不超过 100000 的时候，保持累计值，熄灭 P00 连接的 LED。

（11）初始化 DMA 的设定，再次进入触发等待状态。

（12）再度进入 HALT 模式，等待 DMA 传送结束中断（INTDMA0）。

## 4.2 选项字节设置一览

选项字节的设置，请参见“表 4.1”。

表 4.1 选项字节设置

| 地址            | 设定值       | 内容                                                                      |
|---------------|-----------|-------------------------------------------------------------------------|
| 000C0H/010C0H | 11101111B | 看门狗定时器动作停止<br>(复位后, 停止计数)                                               |
| 000C1H/010C1H | 01111111B | LVD 复位模式<br>检测电压:<br>上升沿 2.81V (2.76V~2.87V)<br>下降沿 2.75V (2.70V~2.81V) |
| 000C2H/010C2H | 11100000B | HS 模式、HOCO: 24MHz                                                       |
| 000C3H/010C3H | 10000100B | 允许片上调试                                                                  |

## 4.3 常量一览

参考例程中使用的常量，请参见“表 4.2”。

表 4.2 参考例程中使用的常量

| 常量                     | 数值         | 内容             |
|------------------------|------------|----------------|
| TOTAL_LIMIT            | 100000     | 接收值的累计值上限      |
| _12_DMA0_SFR_ADDRESS   | 0x12       | DMA 传送的 SFR 地址 |
| _FE20_DMA0_RAM_ADDRESS | 0xFE20     | DMA 传送的 RAM 地址 |
| _0005_DMA0_BYTE_COUNT  | 0x0005     | DMA 传送次数       |
| P_LED                  | P0_bit.no0 | LED 点亮控制端口     |



#### 4.4 变量一览

参考例程中使用的全局变量，请参见“表 4.3”。

表 4.3 参考例程中使用的全局变量

| 类型       | 变量名            | 内容            | 使用的函数                                                                   |
|----------|----------------|---------------|-------------------------------------------------------------------------|
| uint8_t  | g_DmaUartError | UART 接收错误状态变量 | R_DMAC0_Create_UserInit<br>r_dmac0_interrupt<br>r_uart0_interrupt_error |
| uint8_t  | dma_buffer[5]  | 数据接收缓冲器       | R_DMAC0_Create_UserInit<br>r_dmac0_interrupt                            |
| uint32_t | rx_total       | 接收值累计缓冲器      | R_DMAC0_Create_UserInit<br>r_dmac0_interrupt                            |

#### 4.5 函数一览

参考例程中使用的函数，请参见“表 4.4”。

表 4.4 函数

| 函数名                           | 概要                 |
|-------------------------------|--------------------|
| hdwinit                       | 初始化函数              |
| R_Systeminit                  | 系统函数               |
| R_PORT_Create                 | 初始化端口              |
| R_CGC_Create                  | CPU 时钟设置           |
| R_SAU0_Create                 | SAU0 初始化设置         |
| R_UART0_Create                | UART0 初始化设置        |
| r_uart0_interrupt_error       | UART0 接收错误中断       |
| R_DMAC0_Create                | DMA 控制器的初始化设置      |
| R_DMAC0_Start                 | DMA0 动作开始          |
| r_dmac0_interrupt             | DMA0 传送结束中断        |
| main                          | 主函数处理              |
| R_MAIN_UserInit               | 主函数初始化             |
| R_DMAC0_Create_UserInit       | DMA 控制器用户初始化设置     |
| R_UART0_MaskStart             | UART0 通信开始处理（屏蔽中断） |
| R_DMAC0_Set_SoftwareTriggerOn | DMA0 软件触发传送开始处理    |

## 4.6 函数说明

本节对参考例程中使用的函数进行说明。

### [函数名] hdwinit

|     |                                                                                            |
|-----|--------------------------------------------------------------------------------------------|
| 概要  | 初始化函数                                                                                      |
| 头文件 | r_cg_macrodriver.h, r_cg_userdefine.h, r_cg_cgc.h, r_cg_serial.h, r_cg_dmac.h, r_cg_port.h |
| 声明  | void hdwinit(void)                                                                         |
| 说明  | 执行系统函数。                                                                                    |
| 参数  | 无                                                                                          |
| 返回值 | 无                                                                                          |
| 参考  | 无                                                                                          |

### [函数名] R\_Systeminit

|     |                                                                                            |
|-----|--------------------------------------------------------------------------------------------|
| 概要  | 系统函数                                                                                       |
| 头文件 | r_cg_macrodriver.h, r_cg_userdefine.h, r_cg_cgc.h, r_cg_serial.h, r_cg_dmac.h, r_cg_port.h |
| 声明  | void R_Systeminit(void)                                                                    |
| 说明  | 对本应用说明中使用的外围功能进行初始化设置。                                                                     |
| 参数  | 无                                                                                          |
| 返回值 | 无                                                                                          |
| 参考  | 无                                                                                          |

### [函数名] R\_PORT\_Create

|     |                                                    |
|-----|----------------------------------------------------|
| 概要  | 初始化端口                                              |
| 头文件 | r_cg_macrodriver.h, r_cg_port.h, r_cg_userdefine.h |
| 声明  | void R_PORT_Create(void)                           |
| 说明  | 执行 I/O 端口的初始化设置。                                   |
| 参数  | 无                                                  |
| 返回值 | 无                                                  |
| 参考  | 无                                                  |

### [函数名] R\_CGC\_Create

|     |                                                   |
|-----|---------------------------------------------------|
| 概要  | CPU 时钟设置                                          |
| 头文件 | r_cg_macrodriver.h, r_cg_cgc.h, r_cg_userdefine.h |
| 声明  | void R_CGC_Create(void)                           |
| 说明  | 执行 CPU 时钟的初始化设置。                                  |
| 参数  | 无                                                 |
| 返回值 | 无                                                 |
| 参考  | 无                                                 |

---

[函数名] R\_SAU0\_Create

---

|     |                                                      |
|-----|------------------------------------------------------|
| 概要  | SAU0 初始化设置                                           |
| 头文件 | r_cg_macrodriver.h, r_cg_serial.h, r_cg_userdefine.h |
| 声明  | void R_SAU0_Create(void)                             |
| 说明  | 执行 SAU0 的初始化设置。                                      |
| 参数  | 无                                                    |
| 返回值 | 无                                                    |
| 参考  | 无                                                    |

---

[函数名] R\_UART0\_Create

---

|     |                                                      |
|-----|------------------------------------------------------|
| 概要  | UART0 初始化设置                                          |
| 头文件 | r_cg_macrodriver.h, r_cg_serial.h, r_cg_userdefine.h |
| 声明  | void R_UART0_Create(void)                            |
| 说明  | 执行 UART0 的初始化设置。                                     |
| 参数  | 无                                                    |
| 返回值 | 无                                                    |
| 参考  | 无                                                    |

---

[函数名] r\_uart0\_interrupt\_error

---

|     |                                                                   |
|-----|-------------------------------------------------------------------|
| 概要  | UART0 接收错误中断                                                      |
| 头文件 | r_cg_macrodriver.h, r_cg_serial.h, r_cg_dmac.h, r_cg_userdefine.h |
| 声明  | static void __near r_uart0_interrupt_error(void)                  |
| 说明  | 清除错误标志，读取 SDR01 寄存器的值，并保存错误状态，软件触发 DMA 传送。                        |
| 参数  | 无                                                                 |
| 返回值 | 无                                                                 |
| 参考  | 无                                                                 |

---

[函数名] R\_DMAC0\_Create

---

|     |                                                    |
|-----|----------------------------------------------------|
| 概要  | DMA0 控制器的初始化设置                                     |
| 头文件 | r_cg_macrodriver.h, r_cg_dmac.h, r_cg_userdefine.h |
| 声明  | void R_DMAC0_Create(void)                          |
| 说明  | 执行 DMA0 的初始化设置。                                    |
| 参数  | 无                                                  |
| 返回值 | 无                                                  |
| 参考  | 无                                                  |

---

[函数名] R\_DMAC0\_Start

---

|     |                                                    |
|-----|----------------------------------------------------|
| 概要  | DMA0 动作开始                                          |
| 头文件 | r_cg_macrodriver.h, r_cg_dmac.h, r_cg_userdefine.h |
| 声明  | void R_DMAC0_Start(void)                           |
| 说明  | 开始 DMA0 动作。                                        |
| 参数  | 无                                                  |
| 返回值 | 无                                                  |
| 参考  | 无                                                  |

---

[函数名] r\_dma0\_interrupt

---

|     |                                                                                                       |
|-----|-------------------------------------------------------------------------------------------------------|
| 概要  | DMA0 传送结束中断                                                                                           |
| 头文件 | r_cg_macrodriver.h, r_cg_dmac.h, r_cg_userdefine.h                                                    |
| 声明  | static void __near r_dmac0_interrupt(void)                                                            |
| 说明  | 指定的 DMA 传送次数结束的时候，执行此中断处理。<br>接收数据和之前的接收值进行累计加法运算，当超过规定值 100000 时，清除累计值，将 P00 连接的 LED 点亮。此后，再次启动 DMA。 |
| 参数  | 无                                                                                                     |
| 返回值 | 无                                                                                                     |
| 参考  | 无                                                                                                     |

---

[函数名] main

---

|     |                                                                                           |
|-----|-------------------------------------------------------------------------------------------|
| 概要  | 主函数处理                                                                                     |
| 头文件 | r_cg_macrodriver.h, r_cg_cgc.h, r_cg_serial.h, r_cg_dma.h, r_cg_userdefine.h, r_cg_port.h |
| 声明  | void main(void)                                                                           |
| 说明  | 执行主函数处理。                                                                                  |
| 参数  | 无                                                                                         |
| 返回值 | 无                                                                                         |
| 参考  | 无                                                                                         |

---

[函数名] R\_MAIN\_UserInit

---

|     |                                                                                           |
|-----|-------------------------------------------------------------------------------------------|
| 概要  | 主函数初始化                                                                                    |
| 头文件 | r_cg_macrodriver.h, r_cg_cgc.h, r_cg_serial.h, r_cg_dma.h, r_cg_userdefine.h, r_cg_port.h |
| 声明  | void R_MAIN_UserInit(void)                                                                |
| 说明  | 执行主函数初始化处理。                                                                               |
| 参数  | 无                                                                                         |
| 返回值 | 无                                                                                         |
| 参考  | 无                                                                                         |

---

[函数名] R\_DMACH0\_Create\_UserInit

---

|     |                                                   |
|-----|---------------------------------------------------|
| 概要  | DMA 控制器用户初始化设置                                    |
| 头文件 | r_cg_macrodriver.h, r_cg_dma.h, r_cg_userdefine.h |
| 声明  | void R_DMACH0_Create_UserInit(void)               |
| 说明  | 执行 DMA 控制器的用户初始化设置。                               |
| 参数  | 无                                                 |
| 返回值 | 无                                                 |
| 参考  | 无                                                 |

---

[函数名] R\_UART0\_MaskStart

---

|     |                                                      |
|-----|------------------------------------------------------|
| 概要  | UART0 通信开始处理（屏蔽中断）                                   |
| 头文件 | r_cg_macrodriver.h, r_cg_serial.h, r_cg_userdefine.h |
| 声明  | void R_UART0_MaskStart(void)                         |
| 说明  | 屏蔽 UART0 传送结束中断（INTSR0）的状态下，开始 UART 通信。              |
| 参数  | 无                                                    |
| 返回值 | 无                                                    |
| 参考  | 虽然函数 R_UART0_Start 也有相同的目的，但是当需要屏蔽中断时，请使用这个函数。       |

---

[函数名] R\_DMAC0\_Set\_SoftwareTriggerOn

---

|     |                                                    |
|-----|----------------------------------------------------|
| 概要  | DMA0 软件触发传送开始处理                                    |
| 头文件 | r_cg_macrodriver.h, r_cg_dmac.h, r_cg_userdefine.h |
| 声明  | void R_DMAC0_Set_SoftwareTriggerOn(void)           |
| 说明  | 设置 DMA0 的软件触发，开始 DMA 传送（1 次）。                      |
| 参数  | 无                                                  |
| 返回值 | 无                                                  |
| 参考  | 无                                                  |

4.7 流程图

4.7.1 整体流程

本篇应用说明中参考例程的整体流程，请参见“图 4.1”。

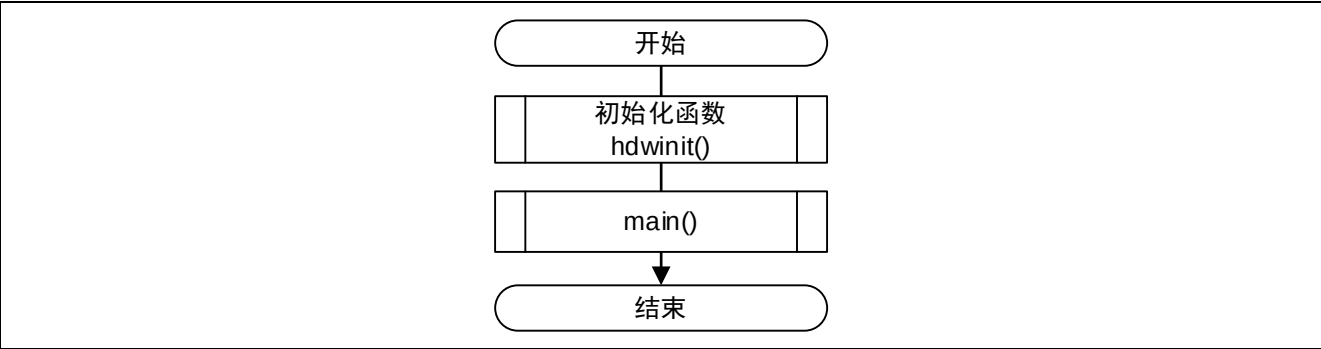


图 4.1 整体流程图

4.7.2 初始化函数

初始化函数的流程，请参见“图 4.2”。

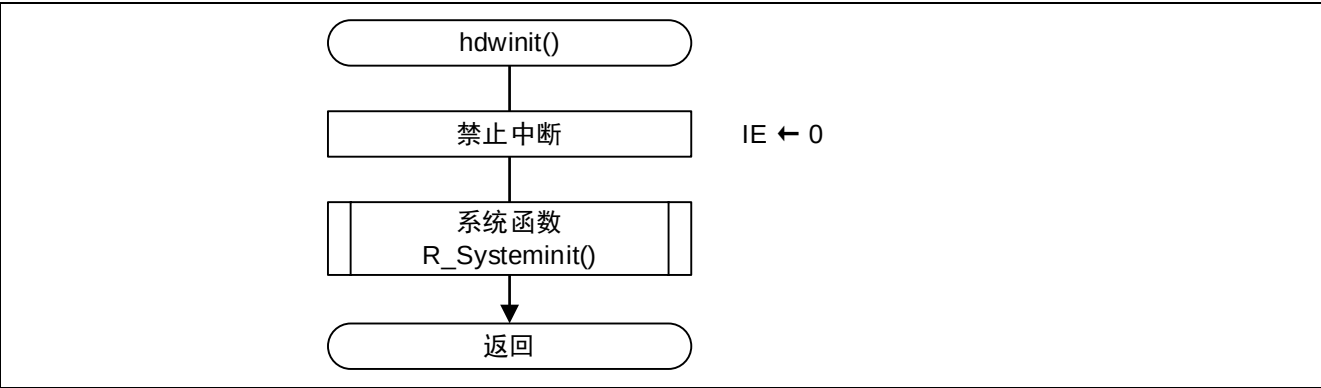


图 4.2 初始化函数

4.7.3 系统函数

系统函数的流程，请参见“图 4.3”。

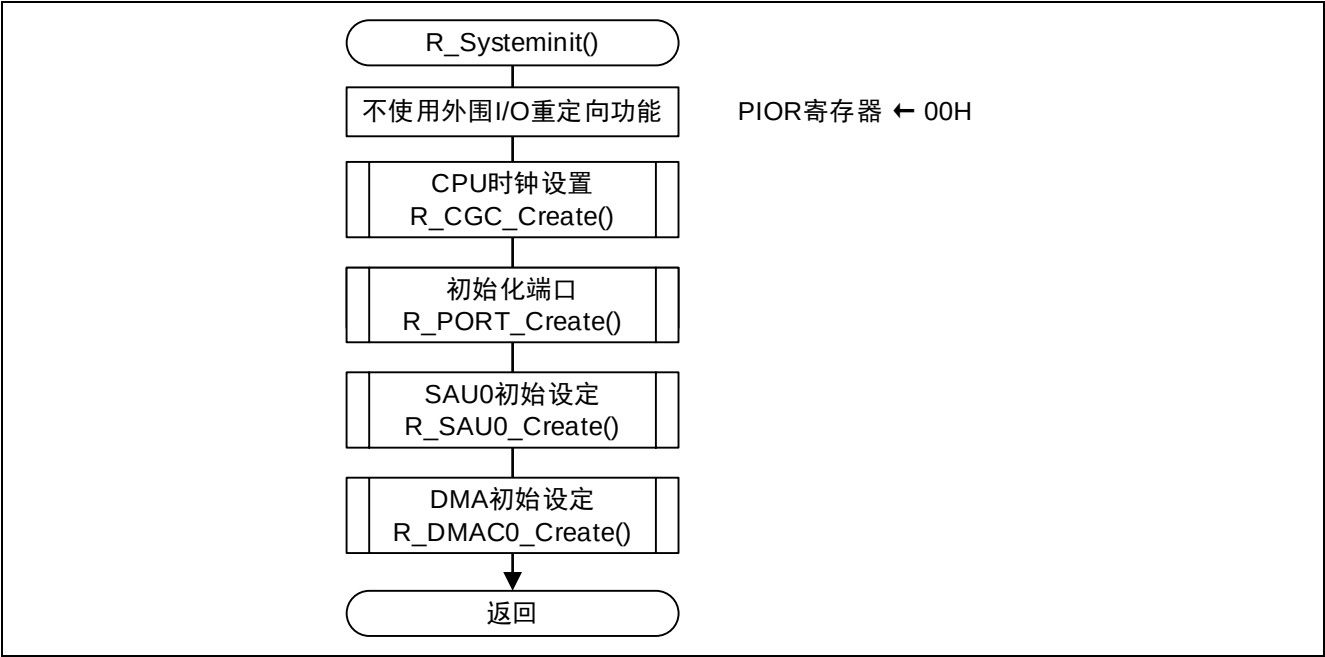


图 4.3 系统函数

4.7.4 初始化端口

初始化端口的流程，请参见“图 4.4”。

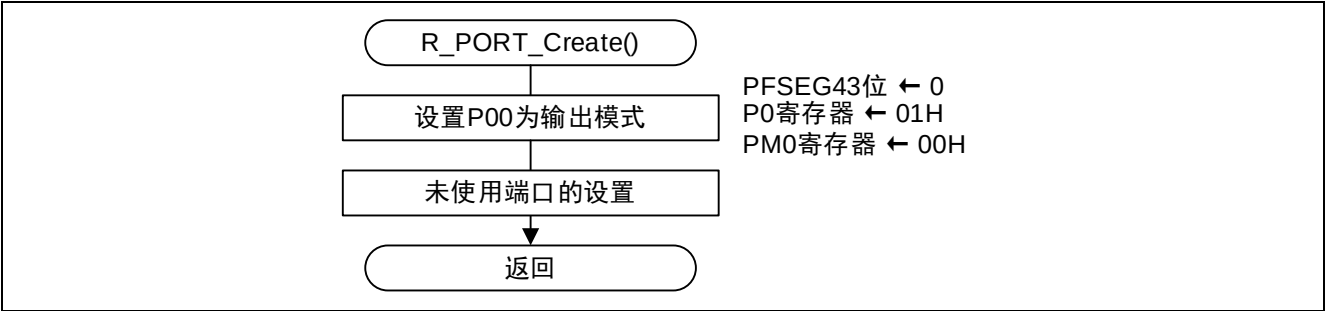


图 4.4 初始化端口

注意：关于未使用端口的设置，请注意根据系统具体要求进行适当的端口处理，并满足电气特性的要求。未使用的输入专用端口，请分别通过电阻上拉到  $V_{DD}$  或是下拉到  $V_{SS}$ 。



## LED 端口的设置

- LCD 端口功能寄存器 5（PFSEG5）

设置 P00 用作端口。

| 符号     | 7           | 6           | 5           | 4           | 3           | 2           | 1           | 0           |
|--------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| PFSEG5 | PFSEG<br>46 | PFSEG<br>45 | PFSEG<br>44 | PFSEG<br>43 | PFSEG<br>42 | PFSEG<br>41 | PFSEG<br>40 | PFSEG<br>39 |
| 设定值    | x           | x           | x           | 0           |             | x           | x           | x           |

## 位 4

| PFSEG43 | P00 引脚的端口（非段输出）/段输出设定 |
|---------|-----------------------|
| 0       | 用作端口（非段输出）            |
| 1       | 用作段输出                 |

- 端口寄存器 0（P0）

设置端口寄存器 0。

| 符号  | 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| P0  | P07 | P06 | P05 | P04 | P03 | P02 | P01 | P00 |
| 设定值 | x   | x   | x   | x   | x   | x   | x   | 1   |

## 位 0

| P00 | P00 引脚输出数据的控制 |
|-----|---------------|
| 0   | 输出“0”         |
| 1   | 输出“1”         |

- 端口模式寄存器 0（PM0）

设置 P00 引脚为输出模式。

| 符号  | 7    | 6    | 5    | 4    | 3    | 2    | 1    | 0    |
|-----|------|------|------|------|------|------|------|------|
| PM0 | PM07 | PM06 | PM05 | PM04 | PM03 | PM02 | PM01 | PM00 |
| 设定值 | x    | x    | x    | x    | x    | x    | x    | 0    |

## 位 0

| PM00 | P00 引脚 I/O 模式的选择 |
|------|------------------|
| 0    | 输出模式（输出缓冲器 ON）   |
| 1    | 输入模式（输出缓冲器 OFF）  |

注意：关于寄存器设置的详细方法，请参考 R7F0C019 用户手册硬件篇。

寄存器图中的设定值说明：

x：未使用位；空白：未变更位；—：预留位或者是什么都不配置的位

4.7.5 CPU 时钟设置

CPU 时钟设置的流程，请参见“图 4.5”。

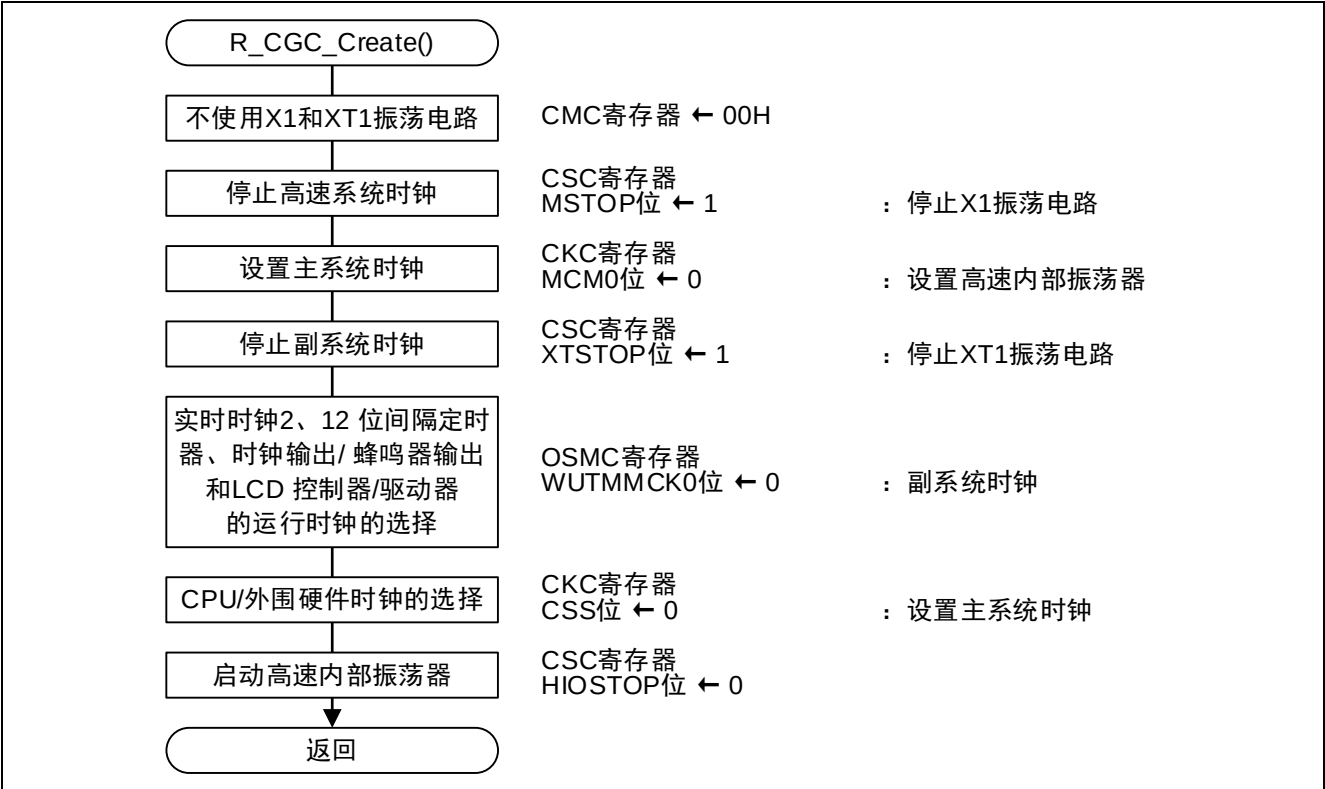


图 4.5 CPU 时钟设置

4.7.6 SAU0 初始化设置

SAU0 初始化设置流程，请参见“图 4.6”。

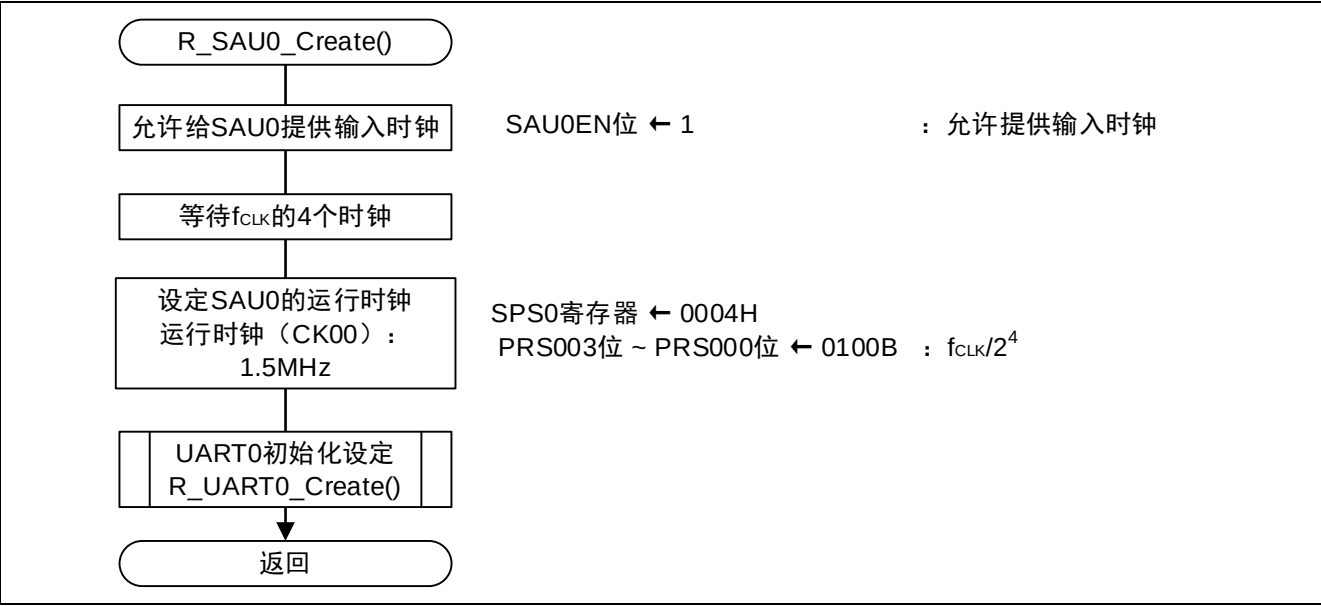


图 4.6 SAU0 初始化设置

4.7.7 UART0 初始化设置

UART0 初始化设置的流程，请参见“图 4.7”。

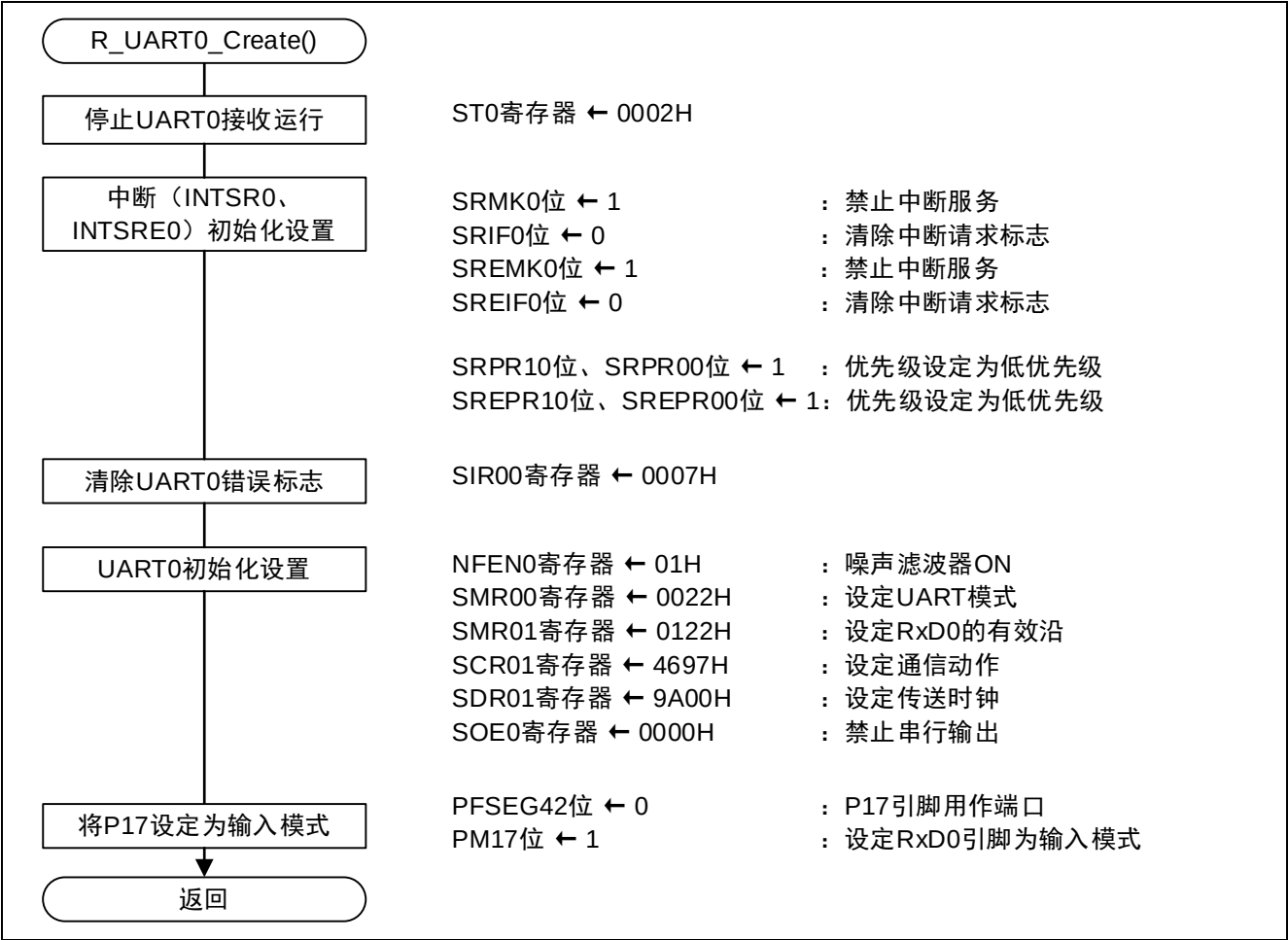


图 4.7 UART0 初始化设置

4.7.8 DMA 控制器的初始化设置

DMA 控制器的初始化设置的流程，请参见“图 4.8”。

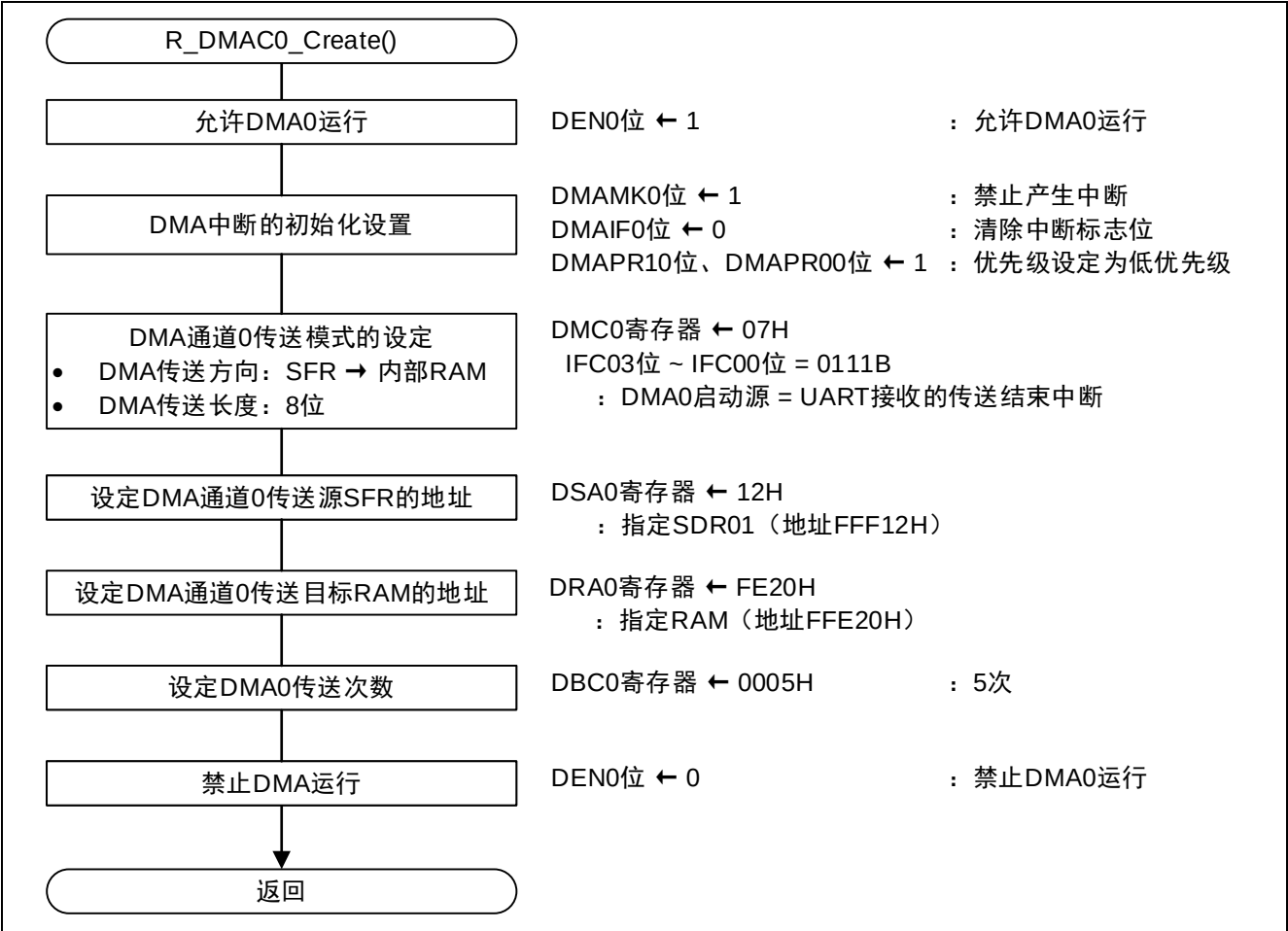


图 4.8 DMA 控制器的初始化设置

设定允许/禁止 DMA 通道 0 运行

- DMA 运行控制寄存器 0（DRC0）

| 符号   | 7    | 6 | 5 | 4 | 3 | 2 | 1 | 0    |
|------|------|---|---|---|---|---|---|------|
| DRC0 | DEN0 | 0 | 0 | 0 | 0 | 0 | 0 | DST0 |
| 设定值  | 0/1  | — | — | — | — | — | — | 0    |

位 7

| DEN0 | DMA 运行允许标志                    |
|------|-------------------------------|
| 0    | 禁止 DMA 通道 0 的运行（停止 DMA 的运行时钟） |
| 1    | 允许 DMA 通道 0 的运行               |

位 0

| DST0 | DMA 传送模式标志                  |
|------|-----------------------------|
| 0    | DMA 通道 0 的 DMA 传送结束         |
| 1    | DMA 通道 0 的 DMA 传送没有结束（正在传送） |

DMA 传送结束中断的初始化设定

- 中断请求标志寄存器（IF0H）  
清除中断请求标志
- 中断屏蔽标志寄存器（MK0H）  
禁止中断处理
- 优先级指定寄存器（PR00H、PR10H）  
中断优先级：优先级 3（低优先级）

| 符号   | 7     | 6      | 5                           | 4      | 3      | 2      | 1     | 0     |
|------|-------|--------|-----------------------------|--------|--------|--------|-------|-------|
| IF0H | SRIF0 | TMIF00 | STIF0<br>CSIIF00<br>IICIF00 | DMAIF1 | DMAIF0 | SREIF2 | SRIF2 | STIF2 |
| 设定值  |       | x      | x                           | x      | 0      | x      | x     | x     |

位 3

| DMAIF0 | 中断请求标志          |
|--------|-----------------|
| 0      | 不产生中断请求信号       |
| 1      | 产生中断请求，处于中断请求状态 |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位

| 符号   | 7     | 6      | 5                           | 4      | 3      | 2      | 1     | 0     |
|------|-------|--------|-----------------------------|--------|--------|--------|-------|-------|
| MK0H | SRMK0 | TMMK00 | STMK0<br>CSIMK00<br>IICMK00 | DMAMK1 | DMAMK0 | SREMK2 | SRMK2 | STMK2 |
| 设定值  |       | x      | x                           | x      | 1      | x      | x     | x     |

位 3

| DMAMK0 | 中断处理的控制 |
|--------|---------|
| 0      | 允许中断处理  |
| 1      | 禁止中断处理  |

| 符号    | 7      | 6       | 5                              | 4       | 3       | 2       | 1      | 0      |
|-------|--------|---------|--------------------------------|---------|---------|---------|--------|--------|
| PR10H | SRPR10 | TMPR100 | STPR10<br>CSIPR100<br>IICPR100 | DMAPR11 | DMAPR10 | SREPR12 | SRPR12 | STPR12 |
| 设定值   |        | x       | x                              | x       | 1       | x       | x      | x      |

| 符号    | 7      | 6       | 5                              | 4       | 3       | 2       | 1      | 0      |
|-------|--------|---------|--------------------------------|---------|---------|---------|--------|--------|
| PR00H | SRPR00 | TMPR000 | STPR00<br>CSIPR000<br>IICPR000 | DMAPR01 | DMAPR00 | SREPR02 | SRPR02 | STPR02 |
| 设定值   |        | x       | x                              | x       | 1       | x       | x      | x      |

位 3

| DMAPR10 | DMAPR00 | 选择优先等级         |
|---------|---------|----------------|
| 0       | 0       | 指定优先级 0 (高优先级) |
| 0       | 1       | 指定优先级 1        |
| 1       | 0       | 指定优先级 2        |
| 1       | 1       | 指定优先级 3 (低优先级) |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位

设置 DMA 通道 0 传送模式

• DMA 模式控制寄存器 0 (DMC0)

软件触发不运行

DMA 传送方向: SFR 传送到内部 RAM

数据长度: 8 位

通过 DMA 启动请求进行 DMA 传送

启动源: 选择 UART0 传送结束中断

| 符号   | 7    | 6    | 5   | 4      | 3     | 2     | 1     | 0     |
|------|------|------|-----|--------|-------|-------|-------|-------|
| DMC0 | STG0 | DRS0 | DS0 | DWAIT0 | IFC03 | IFC02 | IFC01 | IFC00 |
| 设定值  | 0    | 0    | 0   | 0      | 0     | 1     | 1     | 1     |

位 7

|      |                                  |
|------|----------------------------------|
| STG0 | DMA 传送开始的软件触发                    |
| 0    | 软件触发不运行                          |
| 1    | 在允许 DMA 运行 (DEN0=1) 时, 开始 DMA 传送 |

位 6

|      |             |
|------|-------------|
| DRS0 | DMA 传送方向的选择 |
| 0    | SFR→内部 RAM  |
| 1    | 内部 RAM→SFR  |

位 5

|     |               |
|-----|---------------|
| DS0 | DMA 传送数据长度的指定 |
| 0   | 8 位           |
| 1   | 16 位          |

位 4

|        |                            |
|--------|----------------------------|
| DWAIT0 | DMA 传送的保留                  |
| 0      | 通过 DMA 启动请求进行 DMA 传送 (不保留) |
| 1      | 即使接收到 DMA 启动请求也保留 DMA 传送   |

注意: 关于寄存器设置的详细方法, 请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明:

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位



位 3 ~ 0

| IFC03    | IFC02    | IFC01    | IFC00    | DMA 启动源的选择          |                                              |
|----------|----------|----------|----------|---------------------|----------------------------------------------|
|          |          |          |          | 触发信号                | 触发内容                                         |
| 0        | 0        | 0        | 0        | —                   | 通过中断禁止 DMA 传送。<br>(只能允许软件触发)                 |
| 0        | 0        | 0        | 1        | INTAD               | A/D 转换结束中断                                   |
| 0        | 0        | 1        | 0        | INTTM00             | 定时器通道 00 的计数结束或者捕捉中断                         |
| 0        | 0        | 1        | 1        | INTTM01             | 定时器通道 01 的计数结束或者捕捉中断                         |
| 0        | 1        | 0        | 0        | INTTM02             | 定时器通道 02 的计数结束或者捕捉中断                         |
| 0        | 1        | 0        | 1        | INTTM03             | 定时器通道 03 的计数结束或者捕捉中断                         |
| 0        | 1        | 1        | 0        | INTST0/<br>INTCSI00 | UART0 发送的传送结束或者缓冲器空中断/<br>CSI00 传送结束或者缓冲器空中断 |
| <b>0</b> | <b>1</b> | <b>1</b> | <b>1</b> | <b>INTSR0</b>       | <b>UART0 接收的传送结束中断</b>                       |
| 1        | 0        | 0        | 0        | INTST1              | UART1 发送的传送结束或者缓冲器空中断                        |
| 1        | 0        | 0        | 1        | INTSR1              | UART1 接收的传送结束中断                              |
| 1        | 0        | 1        | 0        | INTST2              | UART2 发送的传送结束或者缓冲器空中断                        |
| 1        | 0        | 1        | 1        | INTSR2              | UART2 接收的传送结束中断                              |
| 其他       |          |          |          | 禁止设置                |                                              |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位

设置 DMA 通道 0 传送的 SFR 地址

- DMA SFR 地址寄存器 0 (DSA0)  
指定 DMA 传送源 SFR

| 符号   | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|---|---|---|---|---|---|---|---|
| DSA0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |

设置 RxD0 (SFR 地址: 0x00FFF12) 的低 8 位 (0x12)

设置 DMA 使用的 RAM 地址

- DMA RAM 地址寄存器 0 (DRA0)  
指定 DMA 传送目标 RAM 地址

| 符号   | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| DRA0 |    |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |

设置 RAM 地址 (0xFFE20) 的低 16 位 (0xFE20)

设置 DMA 通道 0 传送次数

- DMA 字节计数寄存器 0 (DBC0)  
设置 DMA 传送次数

| 符号   | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| DBC0 | 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

位 9~0

| DBC0[9:0] | 传送次数的设定 (写 DBC0) | 剩余的传送次数 (读 DBC0)        |
|-----------|------------------|-------------------------|
| 000H      | 1024 次           | 传送结束或者等待 1024 次的 DMA 传送 |
| 001H      | 1 次              | 等待剩余的 1 次 DMA 传送        |
| •         | •                | •                       |
| 005H      | 5 次              | 等待剩余的 5 次 DMA 传送        |
| •         | •                | •                       |
| 3FEH      | 1022 次           | 等待剩余的 1022 次 DMA 传送     |
| 3FFH      | 1023 次           | 等待剩余的 1023 次 DMA 传送     |

将传送次数设置为 5 次

注意: 关于寄存器设置的详细方法, 请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明:

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位



4.7.10 主函数初始化

主函数初始化的流程，请参见“图 4.10”。

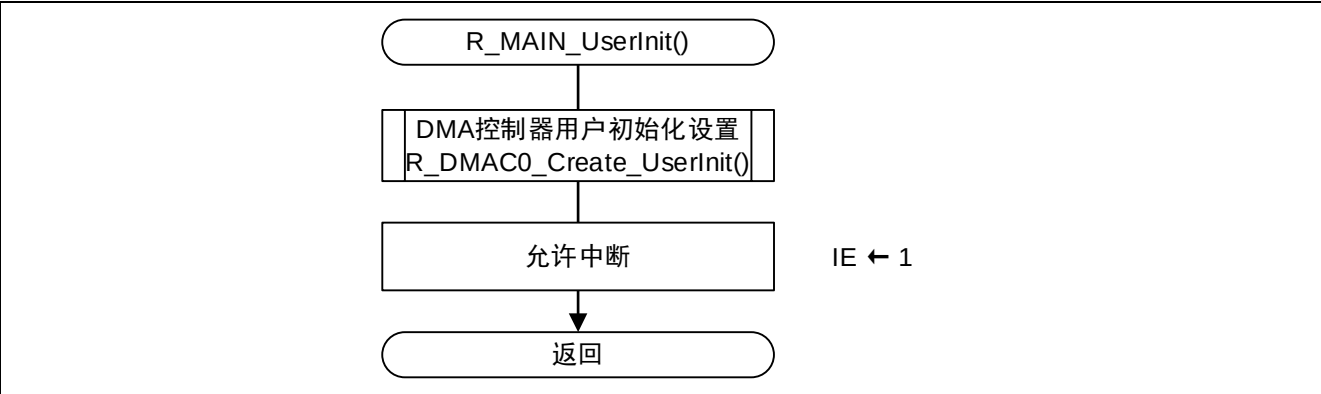


图 4.10 主函数初始化

4.7.11 DMA 控制器用户初始化设置

DMA 控制器用户初始化设置的流程，请参见“图 4.11”。

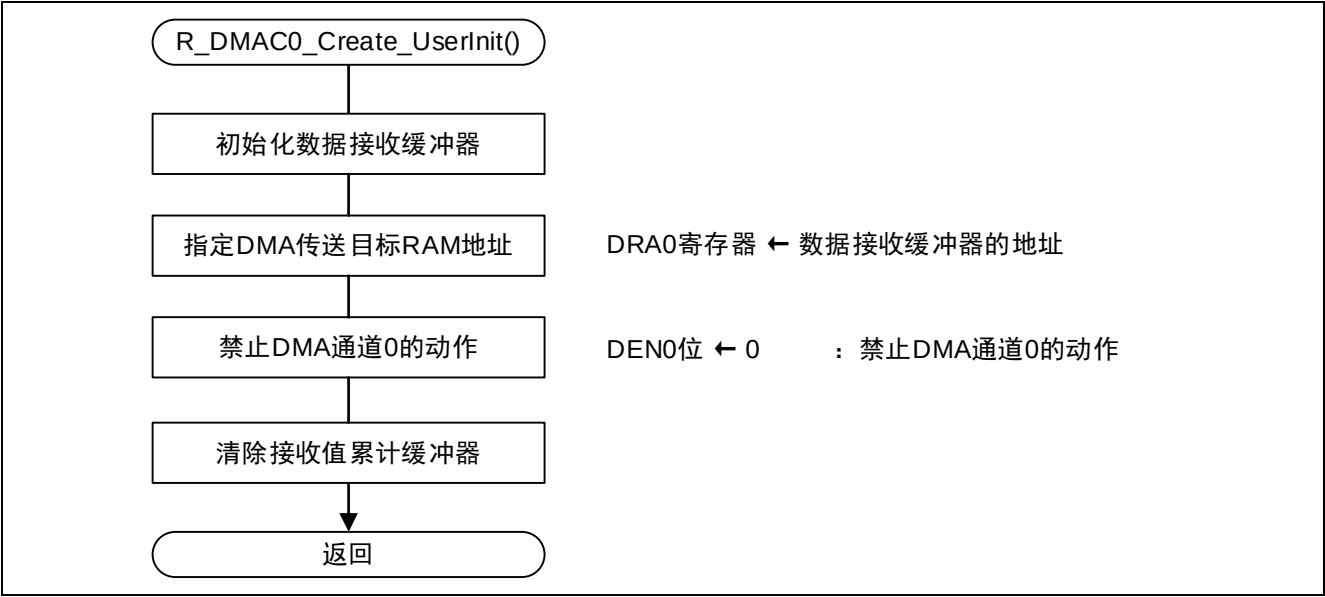


图 4.11 DMA 控制器用户初始化设置

4.7.12    UART0 通信开始处理（屏蔽中断）

UART0 通信开始处理（屏蔽中断）的流程，请参见“图 4.12”。

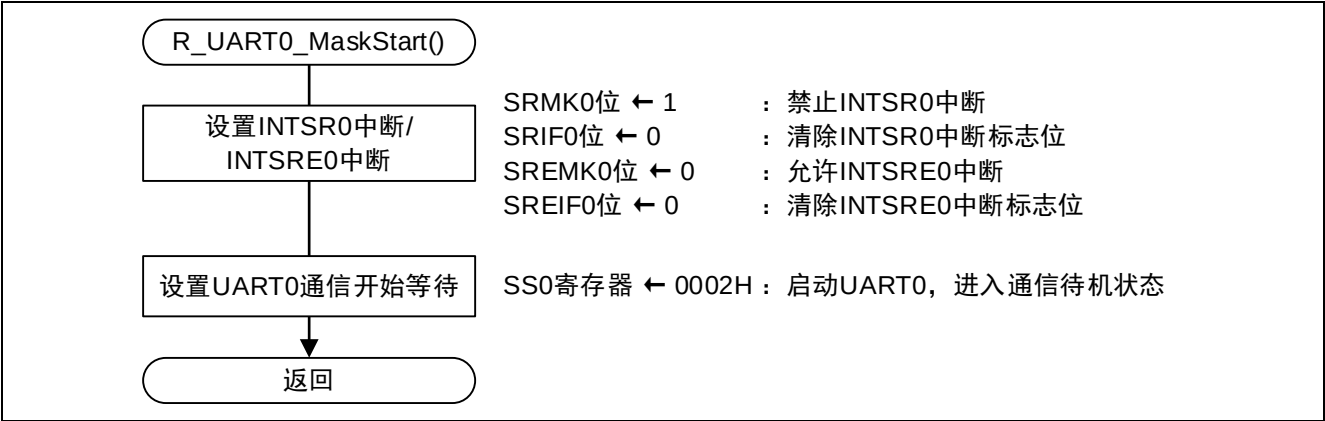


图 4.12 UART0 通信开始处理（屏蔽中断）

UART0 中断的初始化设定

- 中断请求标志寄存器（IF1L、IF0H）

清除中断请求标志

| 符号   | 7      | 6    | 5      | 4       | 3                 | 2     | 1                | 0                 |
|------|--------|------|--------|---------|-------------------|-------|------------------|-------------------|
| IF1L | TMIF01 | FMIF | RTITIF | IICAIF0 | SREIF1<br>TMIF03H | SRIF1 | STIF1<br>IICIF10 | SREIF0<br>TMIF01H |
| 设定值  | x      | x    | x      | x       | x                 | x     | x                | 0                 |

位 0

| SREIF0 | 中断请求标志          |
|--------|-----------------|
| 0      | 不产生中断请求信号       |
| 1      | 产生中断请求，处于中断请求状态 |

| 符号   | 7     | 6      | 5                          | 4      | 3      | 2      | 1     | 0     |
|------|-------|--------|----------------------------|--------|--------|--------|-------|-------|
| IF0H | SRIF0 | TMIF00 | STIF0<br>CSIF00<br>IICIF00 | DMAIF1 | DMAIF0 | SREIF2 | SRIF2 | STIF2 |
| 设定值  | 0     | x      | x                          | x      |        | x      | x     | x     |

位 7

| SRIF0 | 中断请求标志          |
|-------|-----------------|
| 0     | 不产生中断请求信号       |
| 1     | 产生中断请求，处于中断请求状态 |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位

- 中断屏蔽标志寄存器 (MK1L、MK0H)  
清除中断屏蔽

| 符号   | 7      | 6    | 5      | 4       | 3                 | 2     | 1                | 0                 |
|------|--------|------|--------|---------|-------------------|-------|------------------|-------------------|
| MK1L | TMMK01 | FMMK | RTITMK | IICAMK0 | SREMK1<br>TMMK03H | SRMK1 | STMK1<br>IICMK10 | SREMK0<br>TMMK01H |
| 设定值  | x      | x    | x      | x       | x                 | x     | x                | 0                 |

位 0

| SREMK0 | 中断处理的控制 |
|--------|---------|
| 0      | 允许中断处理  |
| 1      | 禁止中断处理  |

| 符号   | 7     | 6      | 5                           | 4      | 3      | 2      | 1     | 0     |
|------|-------|--------|-----------------------------|--------|--------|--------|-------|-------|
| MK0H | SRMK0 | TMMK00 | STMK0<br>CSIMK00<br>IICMK00 | DMAMK1 | DMAMK0 | SREMK2 | SRMK2 | STMK2 |
| 设定值  | 0     | x      | x                           | x      |        | x      | x     | x     |

位 7

| SRMK0 | 中断处理的控制 |
|-------|---------|
| 0     | 允许中断处理  |
| 1     | 禁止中断处理  |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位



启动 UART0

- 串行通道开始寄存器 0（SS0）  
开始 UART0 的通信

|     |    |    |    |    |    |    |   |   |   |   |   |   |      |      |      |      |
|-----|----|----|----|----|----|----|---|---|---|---|---|---|------|------|------|------|
| 符号  | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3    | 2    | 1    | 0    |
| SS0 | 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | SS03 | SS02 | SS01 | SS00 |
| 设定值 | —  | —  | —  | —  | —  | —  | — | — | — | — | — | — | x    | x    | 1    | x    |

位 1

|      |                         |
|------|-------------------------|
| SS01 | 通道 1 运行开始的触发            |
| 0    | 没有触发                    |
| 1    | 将 SE01 位置 “1”，转移到通信待机状态 |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位

4.7.13 DMA0 动作开始

DMA0 动作开始的流程，请参见“图 4.13”。

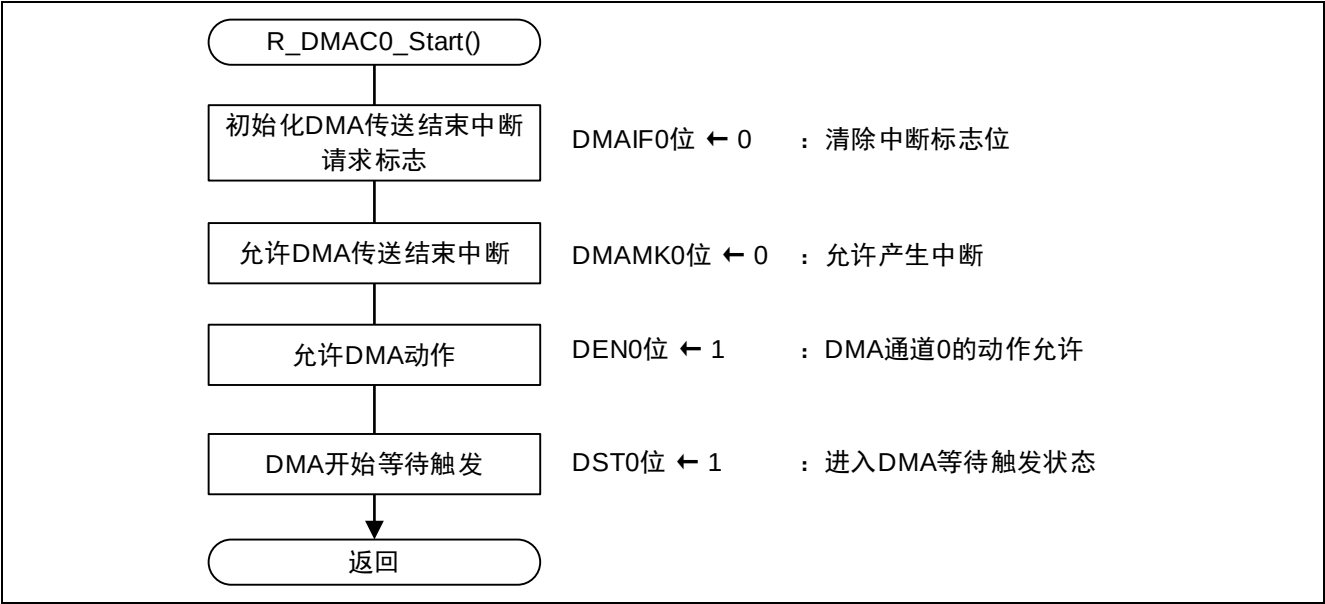


图 4.13 DMA0 动作开始

DMA 传送结束中断的允许准备

- 中断请求标志寄存器 (IF0H)  
清除中断请求标志
- 中断屏蔽标志寄存器 (MK0H)  
清除中断屏蔽

| 符号   | 7     | 6      | 5                          | 4      | 3      | 2      | 1     | 0     |
|------|-------|--------|----------------------------|--------|--------|--------|-------|-------|
| IF0H | SRIF0 | TMIF00 | STIF0<br>CSIF00<br>IICIF00 | DMAIF1 | DMAIF0 | SREIF2 | SRIF2 | STIF2 |
| 设定值  |       | x      | x                          | x      | 0      | x      | x     | x     |

位 3

| DMAIF0 | 中断请求标志           |
|--------|------------------|
| 0      | 不产生中断请求信号        |
| 1      | 产生中断请求, 处于中断请求状态 |

| 符号   | 7     | 6      | 5                           | 4      | 3      | 2      | 1     | 0     |
|------|-------|--------|-----------------------------|--------|--------|--------|-------|-------|
| MK0H | SRMK0 | TMMK00 | STMK0<br>CSIMK00<br>IICMK00 | DMAMK1 | DMAMK0 | SREMK2 | SRMK2 | STMK2 |
| 设定值  |       | x      | x                           | x      | 0      | x      | x     | x     |

位 3

| DMAMK0 | 中断处理的控制 |
|--------|---------|
| 0      | 允许中断处理  |
| 1      | 禁止中断处理  |

注意: 关于寄存器设置的详细方法, 请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明:

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位

设定允许/禁止 DMA 通道 0 运行

- DMA 运行控制寄存器 0（DRC0）

| 符号   | 7    | 6 | 5 | 4 | 3 | 2 | 1 | 0    |
|------|------|---|---|---|---|---|---|------|
| DRC0 | DEN0 | 0 | 0 | 0 | 0 | 0 | 0 | DST0 |
| 设定值  | 1    | — | — | — | — | — | — | 1    |

位 7

| DEN0 | DMA 运行允许标志                    |
|------|-------------------------------|
| 0    | 禁止 DMA 通道 0 的运行（停止 DMA 的运行时钟） |
| 1    | 允许 DMA 通道 0 的运行               |

位 0

| DST0 | DMA 传送模式标志                  |
|------|-----------------------------|
| 0    | DMA 通道 0 的 DMA 传送结束         |
| 1    | DMA 通道 0 的 DMA 传送没有结束（正在传送） |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白: 未变更位、—: 预留位或者是什么都不配置的位

## 4.7.14 DMA0 传送结束中断

DMA0 传送结束中断的流程，请参见“图 4.14”。

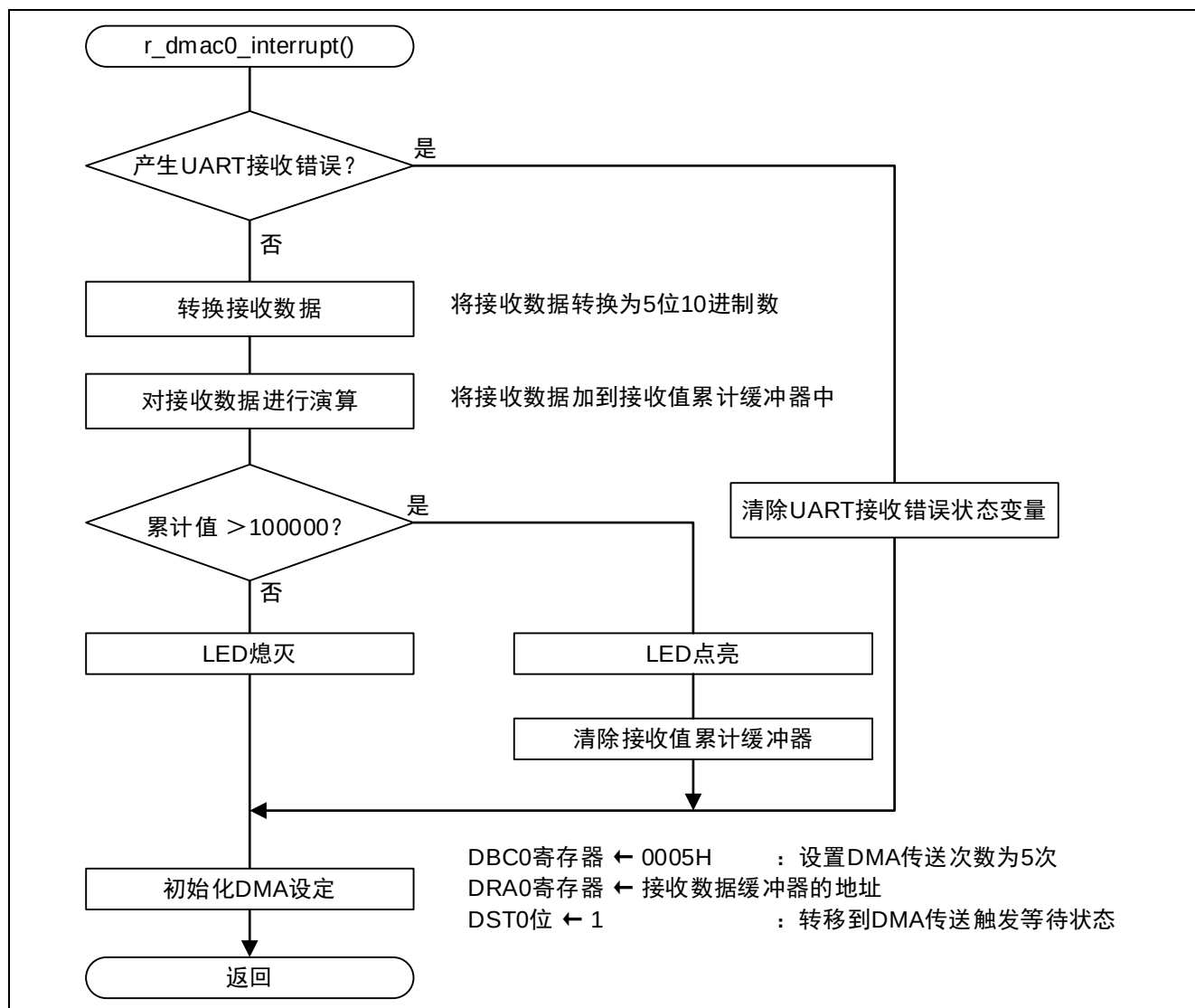


图 4.14 DMA0 传送结束中断

4.7.15 UART0 接收错误中断

UART 接收错误中断的流程，请参见“图 4.15”。

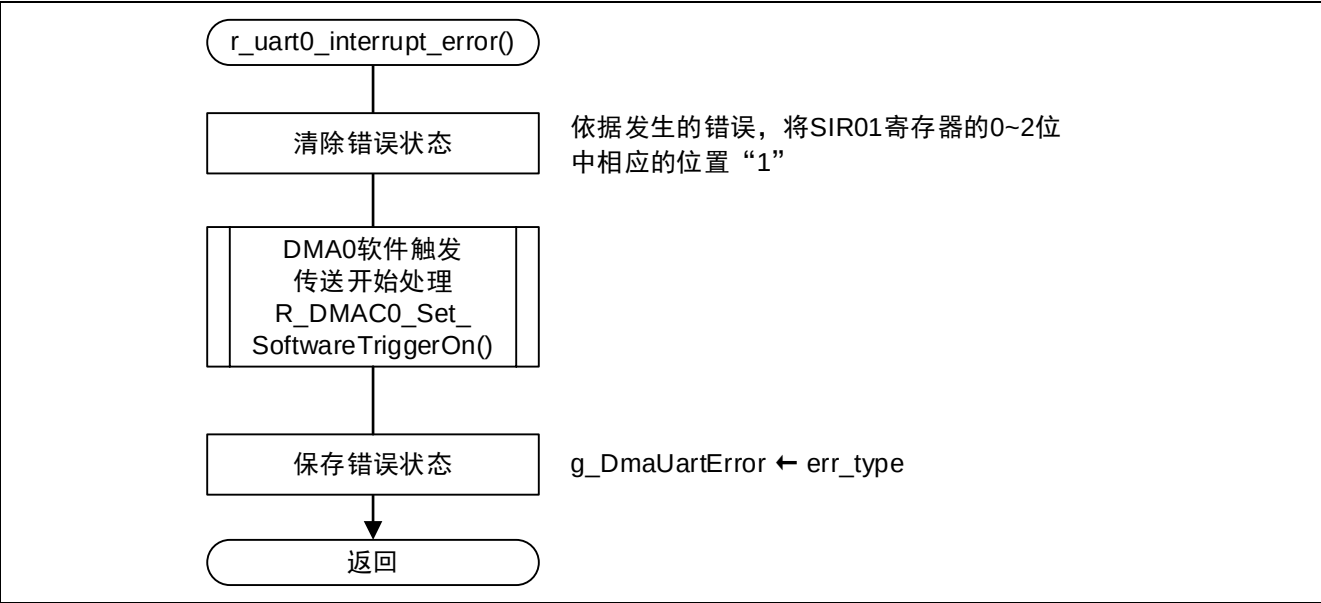


图 4.15 UART0 接收错误中断

4.7.16 DMA0 软件触发传送开始处理

DMA0 软件触发传送开始处理的流程，请参见“图 4.16”。

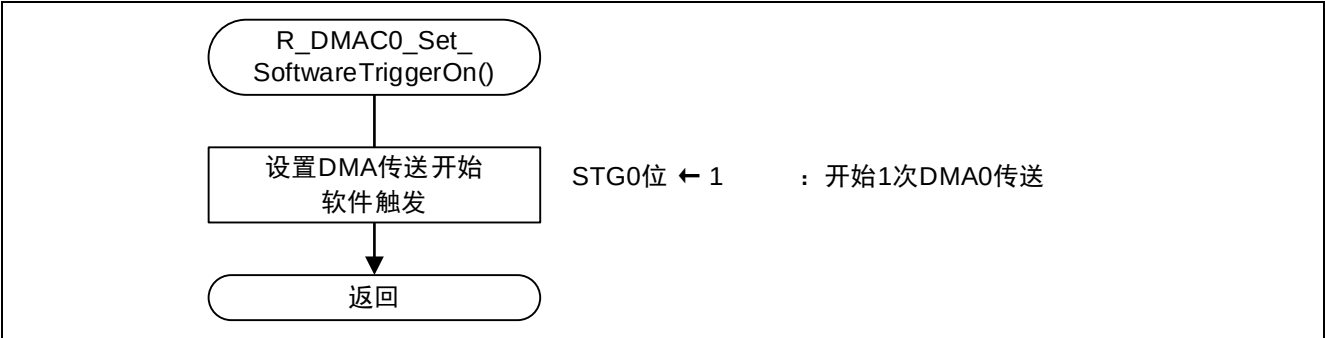


图 4.16 DMA0 软件触发传送开始处理

设置 DMA 通道 0 传送开始软件触发

- DMA 模式控制寄存器 0（DMC0）  
设置 DMA 传送开始软件触发

| 符号   | 7    | 6    | 5   | 4      | 3     | 2     | 1     | 0     |
|------|------|------|-----|--------|-------|-------|-------|-------|
| DMC0 | STG0 | DRS0 | DS0 | DWAIT0 | IFC03 | IFC02 | IFC01 | IFC00 |
| 设定值  | 1    |      |     |        |       |       |       |       |

位 7

| STG0 | DMA 传送开始的软件触发                 |
|------|-------------------------------|
| 0    | 软件触发不运行                       |
| 1    | 在允许 DMA 运行（DEN0=1）时，开始 DMA 传送 |

注意： 关于寄存器设置的详细方法，请参考 R7F0C019 用户手册 硬件篇。

寄存器图中的设定值说明：

x: 未使用位、空白：未变更位、—：预留位或者是什么都不配置的位

## 5. 参考例程

参考例程请从瑞萨电子网页上取得。

## 6. 参考文献

R7F0C019L2DFB 用户手册 硬件篇（R01UH0465C）

RL78 family User's Manual: Software (R01US0015E)

（最新版本请从瑞萨电子网页上取得）

技术信息/技术更新

（最新信息请从瑞萨电子网页上取得）

## 公司主页和咨询窗口

瑞萨电子主页

- <http://cn.renesas.com/>

咨询

- <http://cn.renesas.com/contact/>
- [contact.china@renesas.com](mailto:contact.china@renesas.com)



修订记录

| Rev. | 发行日     | 修订内容 |      |
|------|---------|------|------|
|      |         | 页    | 要点   |
| 1.00 | 2015.12 | 一    | 初版发行 |
|      |         |      |      |
|      |         |      |      |

所有商标及注册商标均归其各自拥有者所有。

## 产品使用时的注意事项

本文对适用于单片机所有产品的“使用时的注意事项”进行说明。有关个别的使用时的注意事项请参照正文。此外，如果在记载上有与本手册的正文有差异之处，请以正文为准。

### 1. 未使用的引脚的处理

**【注意】**将未使用的引脚按照正文的“未使用引脚的处理”进行处理。

CMOS产品的输入引脚的阻抗一般为高阻抗。如果在开路的状态下运行未使用的引脚，由于感应现象，外加LSI周围的噪声，在LSI内部产生穿透电流，有可能被误认为是输入信号而引起误动作。未使用的引脚，请按照正文的“未使用引脚的处理”中的指示进行处理。

### 2. 通电时的处理

**【注意】**通电时产品处于不定状态。

通电时，LSI内部电路处于不确定状态，寄存器的设定和各引脚的状态不定。通过外部复位引脚对产品进行复位时，从通电到复位有效之前的期间，不能保证引脚的状态。

同样，使用内部上电复位功能对产品进行复位时，从通电到达到复位产生的一定电压的期间，不能保证引脚的状态。

### 3. 禁止存取保留地址（保留区）

**【注意】**禁止存取保留地址（保留区）

在地址区域中，有被分配将来用作功能扩展的保留地址（保留区）。因为无法保证存取这些地址时的运行，所以不能对保留地址（保留区）进行存取。

### 4. 关于时钟

**【注意】**复位时，请在时钟稳定后解除复位。

在程序运行中切换时钟时，请在要切换成的时钟稳定之后进行。复位时，在通过使用外部振荡器（或者外部振荡电路）的时钟开始运行的系统中，必须在时钟充分稳定后解除复位。另外，在程序运行中，切换成使用外部振荡器（或者外部振荡电路）的时钟时，在要切换成的时钟充分稳定后再进行切换。

### 5. 关于产品间的差异

**【注意】**在变更不同型号的产品时，请对每一个产品型号进行系统评价测试。

即使是同一个群的单片机，如果产品型号不同，由于内部ROM、版本模式等不同，在电特性范围内有时特性值、动作容限、噪声耐量、噪声辐射量等不同。因此，在变更不同型号的产品时，请对每一个型号的产品进行系统评价测试。

## Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation 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.
  2. 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.
  3. 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.
  4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
  5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.  
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.  
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.  
Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantation etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). 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 for which it is not intended. 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 which the product is not intended by Renesas Electronics.
  6. 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.
  7. 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 systems manufactured by you.
  8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. 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.
  9. 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. You should not use Renesas Electronics products or 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. When exporting the Renesas Electronics 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.
  10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
  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 majority-owned subsidiaries.  
(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

以下"注意事项"为从英语原稿翻译的中文译文，仅作参考译文，英文版的"Notice"具有正式效力。

## 注意事项

1. 本文件中所记载的关于电路、软件和其他相关信息仅用于说明半导体产品的操作和应用实例。用户如在设备设计中应用本文件中的电路、软件 and 相关信息，请自行负责。对于用户或第三方因使用上述电路、软件或信息而遭受的任何损失，瑞萨电子不承担任何责任。
  2. 在准备本文件所记载的信息的过程中，瑞萨电子已尽量做到合理注意，但是，瑞萨电子并不保证这些信息都是准确无误的。用户因本文件中所记载的信息的错误或遗漏而遭受的任何损失，瑞萨电子不承担任何责任。
  3. 对于因使用本文件中的瑞萨电子产品或技术信息而造成的侵权行为或因此而侵犯第三方的专利、版权或其他知识产权的行为，瑞萨电子不承担任何责任。本文件所记载的内容不应视为对瑞萨电子或其他人所有的专利、版权或其他知识产权作出任何明示、默示或其它方式的许可及授权。
  4. 用户不得更改、修改、复制或者以其他方式部分或全部地非法使用瑞萨电子的任何产品。对于用户或第三方因上述更改、修改、复制或以其他方式非法使用瑞萨电子产品的行为而遭受的任何损失，瑞萨电子不承担任何责任。
  5. 瑞萨电子产品根据其质量等级分为两个等级：“标准等级”和“高质量等级”。每种瑞萨电子产品的推荐用途均取决于产品的质量等级，如下所示：  
标准等级：计算机、办公设备、通讯设备、测试和测量设备、视听设备、家用电器、机械工具、个人电子设备以及工业机器人等。  
高质量等级：运输设备（汽车、火车、轮船等）、交通控制系统、防灾系统、预防犯罪系统以及安全设备等。  
瑞萨电子产品无意用于且未被授权用于可能对人类生命造成直接威胁的产品或系统及可能造成人身伤害的产品或系统（人工生命维持装置或系统、植埋于体内的装置等）中，或者可能造成重大财产损失的产品或系统（核反应堆控制系统、军用设备等）中。在将每种瑞萨电子产品用于某种特定应用之前，用户应先确认其质量等级。不得将瑞萨电子产品用于超出其设计用途之外的任何应用。对于用户或第三方因将瑞萨电子产品用于其设计用途之外而遭受的任何损害或损失，瑞萨电子不承担任何责任。
  6. 使用本文件中记载的瑞萨电子产品时，应在瑞萨电子指定的范围内，特别是在最大额定值、电源工作电压范围、移动电源电压范围、热辐射特性、安装条件以及其他产品特性的范围内使用。对于在上述指定范围之外使用瑞萨电子产品而产生的故障或损失，瑞萨电子不承担任何责任。
  7. 虽然瑞萨电子一直致力于提高瑞萨电子产品的质量和可靠性，但是，半导体产品有其自身的具体特性，如一定的故障发生率以及在某些使用条件下会发生故障等。此外，瑞萨电子产品均未进行防辐射设计。所以请采取安全保护措施，以避免当瑞萨电子产品在发生故障而造成火灾时导致人身事故、伤害或损害的事故。例如进行软硬件安全设计（包括但不限于冗余设计、防火控制以及故障预防等）、适当的老化处理或其他适当的措施等。由于难于对微机电系统单独进行评估，所以请用户自行对最终产品或系统进行安全评估。
  8. 关于环境保护方面的详细内容，例如每种瑞萨电子产品的环境兼容性等，请与瑞萨电子的营业部门联系。使用瑞萨电子产品时，请遵守对管制物质的使用或含量进行管理的所有相应法律法规（包括但不限于《欧盟RoHS指令》）。对于因用户未遵守相应法律法规而导致的损害或损失，瑞萨电子不承担任何责任。
  9. 不可将瑞萨电子产品和技术用于或者嵌入日本国内或海外相应的法律法规所禁止生产、使用及销售的任何产品或系统中。也不可将本文件中记载的瑞萨电子产品或技术用于与军事应用或者军事用途有关的任何目的（如大规模杀伤性武器的开发等）。在将本文件中记载的瑞萨电子产品或技术进行出口时，应当遵守相应的出口管制法律法规，并按照上述法律法规所规定的程序进行。
  10. 向第三方分销或处分产品或者以其他方式将产品置于第三方控制之下的瑞萨电子产品买方或分销商，有责任事先向上述第三方通知本文件规定的内容和条件；对于用户或第三方因非法使用瑞萨电子产品而遭受的任何损失，瑞萨电子不承担任何责任。
  11. 在事先未得到瑞萨电子书面认可的情况下，不得以任何形式部分或全部转载或复制本文件。
  12. 如果对本文件所记载的信息或瑞萨电子产品有任何疑问，或者用户有任何其他疑问，请向瑞萨电子的营业部门咨询。
- (注1) 瑞萨电子：在本文件中指瑞萨电子株式会社及其控股子公司。  
(注2) 瑞萨电子产品：指瑞萨电子开发或生产的任何产品。



### SALES OFFICES

### Renesas Electronics Corporation

<http://www.renesas.com>

Refer to "http://www.renesas.com/" for the latest and detailed information.

**Renesas Electronics America Inc.**  
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.  
Tel: +1-408-588-6000, Fax: +1-408-588-6130

**Renesas Electronics Canada Limited**  
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3  
Tel: +1-905-237-2004

**Renesas Electronics Europe Limited**  
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K  
Tel: +44-1628-585-100, Fax: +44-1628-585-900

**Renesas Electronics Europe GmbH**  
Arcadiastrasse 10, 40472 Düsseldorf, Germany  
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

**Renesas Electronics (China) Co., Ltd.**  
Room 1709, Quantum Plaza, No.27 ZhichunLu Haidian District, Beijing 100191, P.R.China  
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

**Renesas Electronics (Shanghai) Co., Ltd.**  
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333  
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

**Renesas Electronics Hong Kong Limited**  
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong  
Tel: +852-2265-6688, Fax: +852-2886-9022

**Renesas Electronics Taiwan Co., Ltd.**  
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan  
Tel: +886-2-8175-9600, Fax: +886-2-8175-9670

**Renesas Electronics Singapore Pte. Ltd.**  
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949  
Tel: +65-6213-0200, Fax: +65-6213-0300

**Renesas Electronics Malaysia Sdn.Bhd.**  
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jin Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia  
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

**Renesas Electronics India Pvt. Ltd.**  
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India  
Tel: +91-80-67208700, Fax: +91-80-67208777

**Renesas Electronics Korea Co., Ltd.**  
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea  
Tel: +82-2-558-5737, Fax: +82-2-558-5141