To our customers,

# Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

# 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; anti-crime 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 majority-owned subsidiaries.

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

# H8S/2200 Series

## DMA Transfer of SCI Receive Data to SRAM

### Introduction

Data received from the serial interface (SCI) is transferred to the SRAM by using the DMAC.

### Target Device

H8S/2215

### Contents

## 1. Overview

Data received from the serial interface (SCI1) is transferred to the SRAM by using the DMAC.

## 2. Configuration

Figure 1 shows the confirmed operation of this application note.

List of Components Used

| No. | Component | Specifications |
|---|---|---|
| 1 | Solution Engine<br>H8S/2215 CPU board<br>(Manufactured by Hitachi ULSI Systems) | Board power supply input: 5 VDC<br>Operating frequency: 16 MHz<br>MCU operating mode: 6<br>SRAM (128k × 16 bits) |

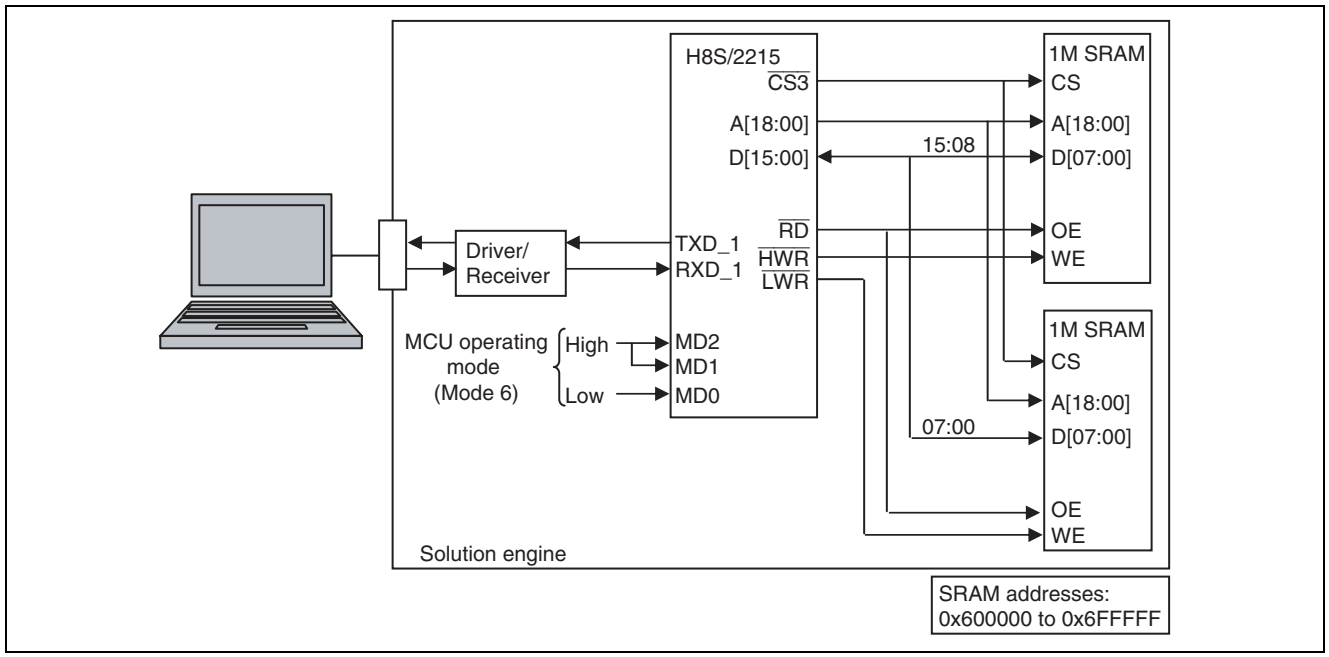



**Figure 1 Confirmed Operation**

## 3. Description of Functions

The DMAC is started by inputting a command from the terminal software connected to the RS-232C interface. Moreover, a function for reading from and writing to memory and a function for reading from and writing to registers internal to the microcomputer are provided for debugging.

1. Communication specifications
   The terminal software is connected according to the specifications below.

| Communication method | Asynchronous |
|---|---|
| Bit rate | 38400 bps |
| Data size | 8 bits |
| Parity | None |
| Stop bit | 1 bit |
| Terminating code | Line feed |

2. Supported functions

| No. | Function | Specifications |
|---|---|---|
| 1 | From the serial interface (SCI1), data as long as specified in length is transferred to a specified memory location by DMA. | Format: dmΔaddressΔlength<br>(Specify a 4-byte value in address, and a 2-byte value in length.) |
| 2 | Data of 64 bytes is read from a specified memory address for display. | Format: mrΔaddress<br>(Specify a 4-byte value in address.) |
| 3 | Data of 4 bytes is written starting at a specified memory address. | Format: mwΔaddressΔdata<br>(Specify a 4-byte value in address and data.) |
| 4 | A register internal to the H8S/2215 is read for display. | Format: hrΔaddress<br>(Specify a 2-byte value in address.)<br>All registers are displayed if no value is specified in address. |
| 5 | Specified data is written to a register internal to the H8S/2215. | Format: hwΔaddressΔdata<br>(Specify a 4-byte value in address, and a 1-byte value in data.) |

# 4. Principles of Operation

## 4.1 Initialization Processing

Before exercising DMAC control, start up the microcomputer and perform operations such as internal register initialization.

1. Control for low power consumption, clock oscillator initialization.

| Register name ←Set value | Bit | Name | Value | Contents |
|---|---|---|---|---|
| LPWCR ←0x03 | 7-4 | ---- | 0000 | |
| | 3 | RFCUT | 0 | Uses internal feedback resistance control. |
| | 2 | ---- | 0 | |
| | 1:0 | STC[1:0] | 11 | Bypasses PLL. |
| MSTPCRA ←0x0D | 7 | MSTPA7 | 0 | Operates the DMAC module. |
| | 6 | MSTPA6 | 0 | Operates the DTC module. |
| | 5 | MSTPA5 | 0 | Operates the TPU module. |
| | 4 | MSTPA4 | 0 | Operates the TMR module. |
| | 3:2 | MSTPA[3:2] | 11 | (Reserved) |
| | 1 | MSTPA1 | 0 | Operates the AD module. |
| | 0 | MSTPA0 | 1 | (Reserved) |
| MSTPCRB ←0x1F | 7 | MSTPB7 | 0 | Operates the SCI0 module. |
| | 6 | MSTPB7 | 0 | Operates the SCI1 module. |
| | 5 | MSTPB7 | 0 | Operates the SCI2 module. |
| | 4:1 | MSTPB[4:1] | 1111 | (Reserved) |
| | 0 | MSTPB0 | 1 | Stops the USB module. |
| MSTPCRC ←0xDF | 7:6 | MSTPC[7:6] | 11 | (Reserved) |
| | 5 | MSTPC5 | 0 | Operates the DA module. |
| | 4:0 | MSTPC[4:0] | 11111 | (Reserved) |

2. I/O port initialization
   Set the input/output pins of port G as indicated below. Set all other pins to the output mode.

| Port | Register name ←Set value | Bit | Name | Value | Contents |
|---|---|---|---|---|---|
| G | PGDDR ←0xFF | 7:5 | --- | 11 | |
| | | 4:2 | PG[4:2]DDR | 111 | Output (Not used) |
| | | 1 | PG1DDR | 1 | Output (CS3 enabled) |
| | | 0 | PG0DDR | 1 | Output (Not used) |

3. Bus controller initialization
   Set the bus so that the externally connected SRAM (128k × 16 bits) can be accessed.

| Register name ←Set value | Bit | Name | Value | Contents |
|---|---|---|---|---|
| ABWCR ←0x77 | 7 | ABW7 | 0 | Area 7 Bus width 16 bits (Not used) |
| | 6:4 | ABW[6:4] | 111 | Areas 6 to 4 Bus width 8 bits (Not used) |
| | 3 | ABW3 | 0 | Area 3 Bus width 16 bits (SRAM) |
| | 2:0 | ABW[2:0] | 111 | Areas 2 to 0 Bus width 8 bits (Not used) |
| PFCR ←0x0F | 7:4 | ---- | 0 | |
| | 3:0 | AE[3:0] | 1111 | Enables A23:00 output. |

4. TPU0 timer initialization
   Make settings so that a timer interrupt is generated at intervals of 100 ms for timer monitoring.

| Register name ←Set value | Bit | Name | Value | Contents |
|---|---|---|---|---|
| TCR_0 ←0x23 | 7:5 | CCLR[2:0] | 001 | Counter clear by TGRA compare match |
| | 4:3 | CKEG[1:0] | 00 | Count on rising edge |
| | 2:0 | TPSC[2:0] | 011 | Count with φ/64 |
| TMDR_0 ←0x00 | 7:6 | ---- | 00 | |
| | 5 | BFB | 0 | Normal TGRB operation |
| | 4 | BFA | 0 | Normal TGRA operation |
| | 3 | ---- | 00 | |
| | 2:0 | MD[2:0] | 000 | Normal operation |
| TIORH_0 ←0x00 | 7:4 | IOB[3:0] | 0000 | TBRB output compare (Not used) |
| | 3:0 | IOA[3:0] | 0000 | TBRA output compare |
| TIORL_0 ←0x00 | 7:4 | IOD[3:0] | 0000 | TBRD output compare (Not used) |
| | 3:0 | IOC[3:0] | 0000 | TBRC output compare (Not used) |
| TIER_0 ←0x00 | 7 | TTGE | 0 | Disables AD conversion start request. (Not used) |
| | 6 | ---- | 0 | |
| | 5 | TCIEU | 0 | Disables underflow interrupt. (Not used) |
| | 4 | TCIEV | 0 | Disables overflow interrupt. (Not used) |
| | 3 | TGIED | 0 | Disables TGRD interrupt. (Not used) |
| | 2 | TGIEC | 0 | Disables TGRC interrupt. (Not used) |
| | 1 | TGIEB | 0 | Disables TGRB interrupt. (Not used) |
| | 0 | TGIEA | 0 | Enables TGRA interrupt. |
| TGRA_0 ←25000 | 15:0 | TGRA_0 | 25000 | TBRA output compare value (To be set to generate an interrupt at intervals of 100 ms) |
| TCNT_0 ←0x0000 | 15:0 | TCNT_0 | 0x0000 | Counter clear |
| TSTR ←0x01 | 7:0 | TSTR | 0x01 | TCNT_0 counter start |

5. Serial interface (SCI_1) initialization
   Make settings to connect the terminal software for starting the DMAC.

| Register name ←Set value | Bit | Name | Value | Contents |
|---|---|---|---|---|
| SCR_1 ←0x00 | 7 | TIE | 0 | Disables transmit interrupt. |
| | 6 | RIE | 0 | Disables receive interrupt. |
| | 5 | TE | 0 | Disables transmit operation. |
| | 4 | RE | 0 | Disables receive operation. |
| | 3 | MPIE | 0 | Disables multi-processor interrupt. |
| | 2 | TEIE | 0 | Disables transmit end interrupt. |
| | 1:0 | CKE[1:0] | 00 | Asynchronous, internal clock used |
| | 7 | TIE | 0 | Disables transmit interrupt. |
| SMR_1 ←0x00 | 7 | C/A | 0 | Asynchronous mode |
| | 6 | CHR | 0 | 8-bit length |
| | 5 | PE | 0 | No parity check. |
| | 4 | O/E | 0 | Even parity (Not used) |
| | 3 | STOP | 0 | One stop bit |
| | 2 | MP | 0 | Disables the multi-processor communication function. |
| | 1:0 | CKS[1:0] | 00 | Clock source = $\phi$ |
| SCMR_1 ←0x00 | 7:4 | ---- | 0000 | |
| | 3 | DIR | 0 | LSB first |
| | 2 | INV | 0 | No data inversion |
| | 1-0 | ---- | 00 | |
| BRR ←12 | 7:0 | BRR | 12 | Sets the transmission speed to 38400 bps. |

— The time for at least one stop bit is awaited. (38400 bps: About 30 μs)
— Receive processing is enabled.

| Register name (Address←Set value) | Bit | Name | Value | Contents |
|---|---|---|---|---|
| SCR_1 ←0x50 | 7 | TIE | 0 | Disables transmit interrupt. |
| | 6 | RIE | 1 | Enables receive interrupt. |
| | 5 | TE | 0 | Disables transmit operation. |
| | 4 | RE | 1 | Enables receive operation. |
| | 3 | MPIE | 0 | Disables multi-processor interrupt. |
| | 2 | TEIE | 0 | Disables transmit end interrupt |
| | 1:0 | CKE[1:0] | 00 | Asynchronous, internal clock used |

## 4.2 DMAC Control

1. Overview

The DMAC is started by a command from the terminal software connected to SCI_1.

Format: dmΔaddressΔlength (Specify a 4-byte value in address, and 2-byte value in length.)

As many data characters as specified in length received from the serial interface (SCI1) are transferred by DMA to the memory specified in address. In this application note, a setting is made so that no normal receive interrupt is generated during DMA transfer by resetting a receive interrupt source with the DMAC (depending on the setting of DMABCRH (DTA0A)). Upon completion of transfer of data characters as many as specified in length, a DMAC end interrupt is generated to end DMA transfer processing.

The DMAC of the H8S/2215 has four channels (0A, 0B, 1A, and 1B) and operates in the operating modes indicated below. In this application note, channel 0A is used in the sequential mode of the short address mode. For the detailed specifications of each operating mode, refer to the "H8S/2215 Series Hardware Manual."

| Transfer mode | | | Transfer source | Remarks |
|---|---|---|---|---|
| Short address mode | Dual address mode | (1) Sequential mode<br>(2) Idle mode<br>(3) Repeat mode | • Compare match/input capture A interrupt of the TPU channels 0 to 2<br>• SCI send data empty interrupt<br>• SCI receive data full interrupt<br>• A/D converter conversion end interrupt | • Up to 4 channels can be operated independently of each other. |
| Full address mode | (4) Normal mode | | • USB request<br>• Auto-request | • Up to 2 channels combining channels A and B can be operated.<br>• In the case of an auto-request, a choice can be made between burst mode transfer and cycle steal transfer. |
| | (5) Block transfer mode | | • Compare match/input capture A interrupt of the TPU channels 0 to 2<br>• SCI send data empty interrupt<br>• SCI receive data full interrupt<br>• A/D converter conversion end interrupt<br>• USB request | |

2. DMAC setting

Before starting the DMAC, set each register of channel 0A according to the procedure below.

This application program enables this setting to be made using one subroutine (dma_rxd_to_ram) at a time. For details, see section 5, "Description of Sample Program."

| No. | Setting | Register to be set |
|---|---|---|
| 1 | Sets the operating mode to the short address mode | DMABCRH (FAE0) ←0 |
| 2 | Sets for resetting a receive interrupt source with the DMAC during DMA transfer | DMABCRH (DTA0A) ←1 |
| 3 | Sets of a transfer source address (RDR_1: Receive data storage area of SCI1) in IOAR0A | IOAR0A←FF85 |
| 4 | Sets of a transfer destination address in MAR0A | MAR0A←address<br>(Lower 3 bytes of address received with a command) |
| 5 | Sets of the number of transfers in ETCR0A | ETCR0A←length<br>(Lower 2 bytes of length received with a command) |
| 6 | Sets of 1 byte as the unit of DMA transfer | DMCRA (DTSZ) ←0 |
| | Incrementing MAR0A by 1 each time a DMA transfer is made | DMCRA (DTID) ←0 |
| | Sets of the sequential mode | DMCRA (RPE) ←0 |
| | Sets of IOAR as a transfer source, and MAR as a transfer destination | DMCRA (DTDIR) ←0 |
| | Sets for starting DMA with a receive data full interrupt of SCI channel 1 | DMCRA (DTF[3:0]) ←7 |
| 7 | Sets for enabling DMA data transfer<br>(Actual transfer processing is performed when an SCI1 receive interrupt is generated.) | DMABCRH (DTE0A) ←1<br>(Unless the state of DTE0A=0 is read beforehand, 1 cannot be set.) |
| 8 | Sets for generating a DMA end interrupt of channel 0A when DMA transfer ends | DMABCRH (DTIE0A) ←1 |

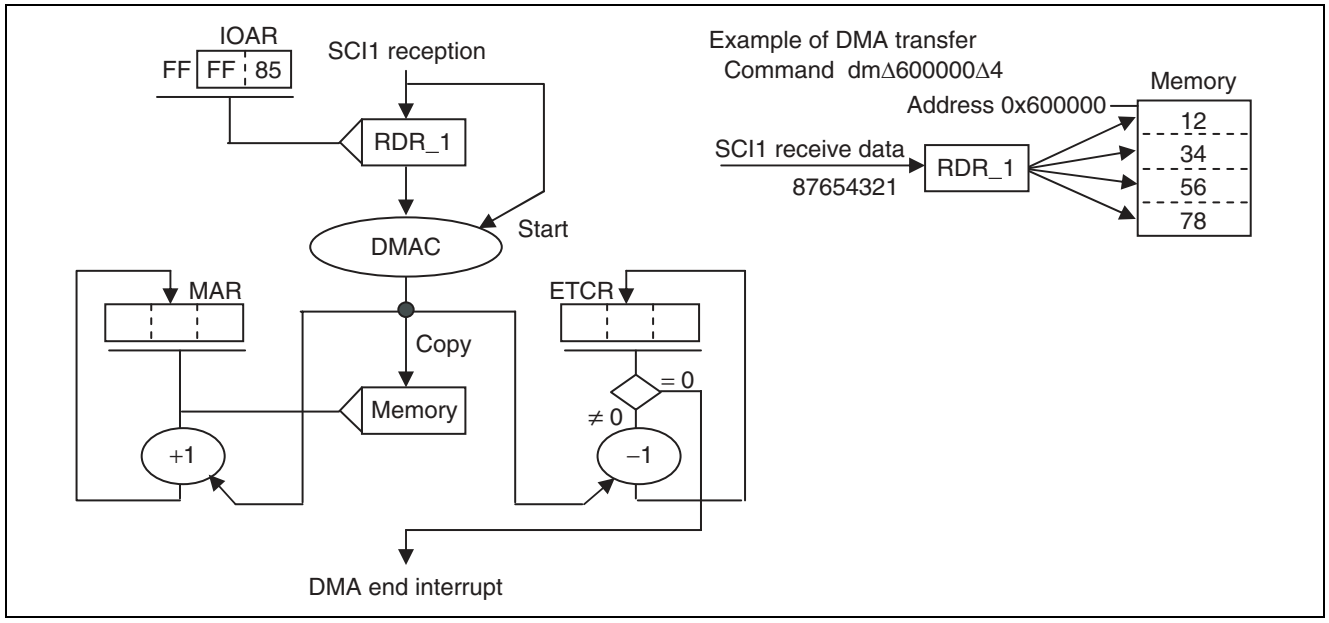
3. DMAC operation

If the settings above have been made, a DMA transfer is made each time SCI1 data is received.

DMA transfers are made as many times as the number of transfers set in the ETCR0A register. An end interrupt of DMAC channel 0A is finally generated to end the processing.

The processing flow of this operation is described below.

A. When SCI1 receive data is stored in the RDR_1 register, the DMAC is started.
B. The DMAC copies the data of the transfer source address (RDR_1 address) indicated by the IOAR0A register to the transfer destination address indicated by the MAR0A register. At this time, an SCI1 receive interrupt source is reset by the DMAC.
C. The DMAC increments the transfer destination address indicated by the MAR0A register by 1, and decrements the number of transfers indicated by the ETCR0A register by 1.
D. Steps A. to C. are repeated until ERCT0A is cleared to 0.
E. When ERCT0A is cleared to 0, a DMA end interrupt of DMAC channel 0A is generated, and the DMAC resets DMABCRH (DTE0A) to 0.
F. During DMA end interrupt processing, the software clears DMABCRH (DTIE0A) to 0 to end DMAC processing.



# 5. Description of Sample Program

## 5.1 File Configuration

A sample program is provided as a project of HEW (High-performance Embedded Workshop). When h8s.hws is executed, HEW starts up to enable source program referencing and updating. If you do not have HEW, directly reference the following source files with an editor:

| No. | File name | Application |
|---|---|---|
| 1 | resetprg.c | Executed starting at reset vector address 0 when the microcomputer is reset |
| 2 | intprg.c | Executed when an interrupt source other than a reset is generated |
| 3 | dbsct.c | Processing for setting the start and end addresses of a section used by the _INITSCT function of resetprg.c in the section initialization table. For details of the processing, refer to sections 9 and 10 of the "H8S and H8/300 Series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual." |
| 4 | h8s.c | Main routine of this application note |
| 5 | com.c | Main common and interrupt processing routine of this application note |
| 6 | 2215S.H | Structure definition file of the internal registers of the H8S/2215<br>This file can be obtained from *2. However, modifications are made to the DMAC-related definition. For the modifications, check the source code. |
| 7 | cwtbl.h | Variable and constant definitions are made for this application note. |
| 8 | prototypeh | A prototype declaration is made for this application note. |
| 9 | stacksct.h | A stack size definition is made. |

*1 http://www.renesas.com/
*2 http://www.renesas.com/

## 5.2 Linkage

The linkage address of each section is indicated below.

In a HEW project file, a section can be referenced or set with Category : section in the Link/Librarq tab of the -Standard Toolchain option.

| Section | Start address |
|---|---|
| PResetPRG | 0x000400 |
| PIntPRG | |
| P | 0x000800 |
| C | |
| C$DSEC | |
| C$BSEC | |
| D | |
| B | 0xFFB000 |
| R | |
| S | 0xFFEDB0 |

## 5.3 Subroutine Specifications

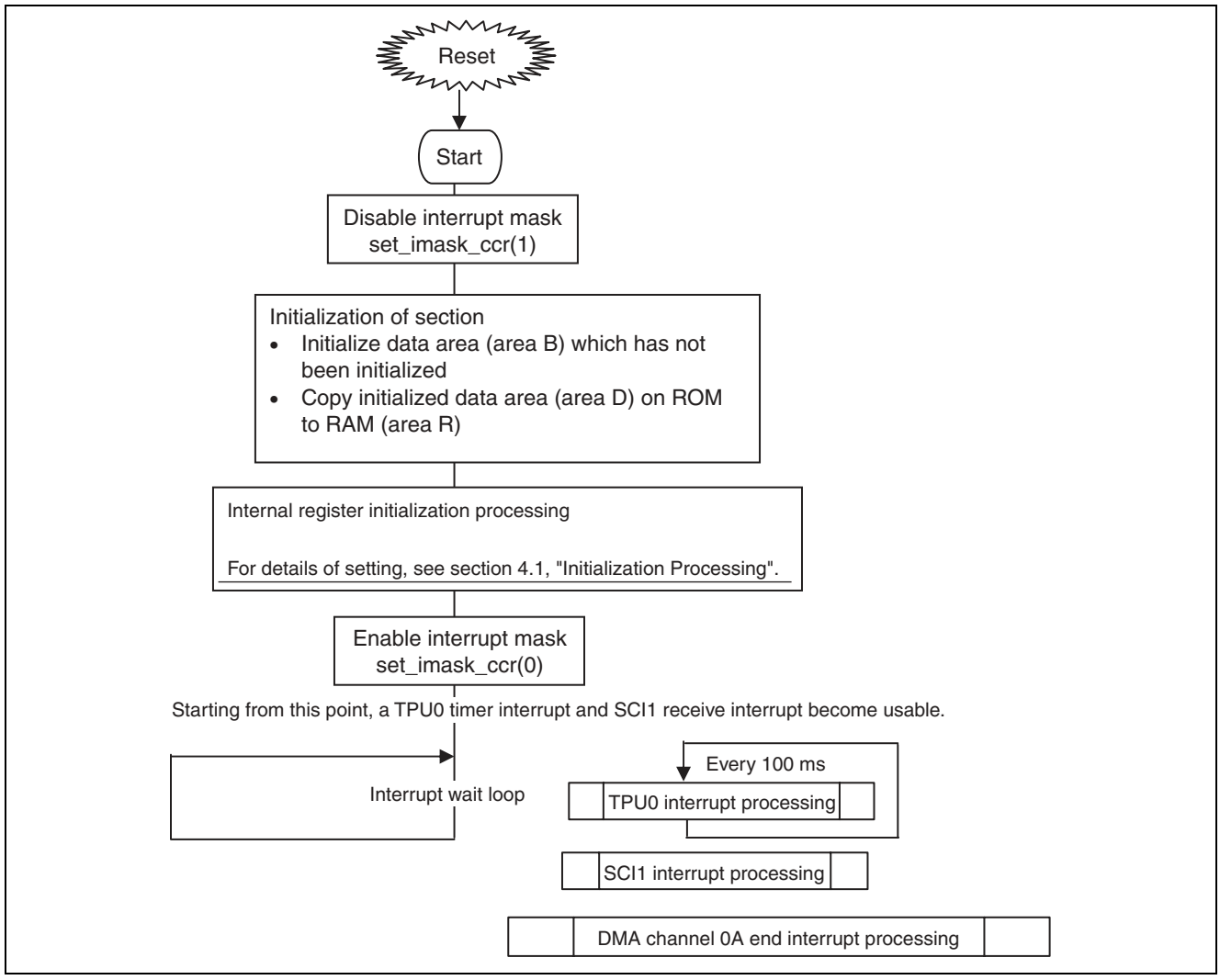
With this application note, DMA parameters can be set and started using one subroutine. This function eases the use of DMA.

1. Data received from SCI is transferred by DMA to a specified address.
   Subroutine name: `void dma_rxd_to_ram ( int sci_no , unsigned long ram_addr , unsigned int length)`

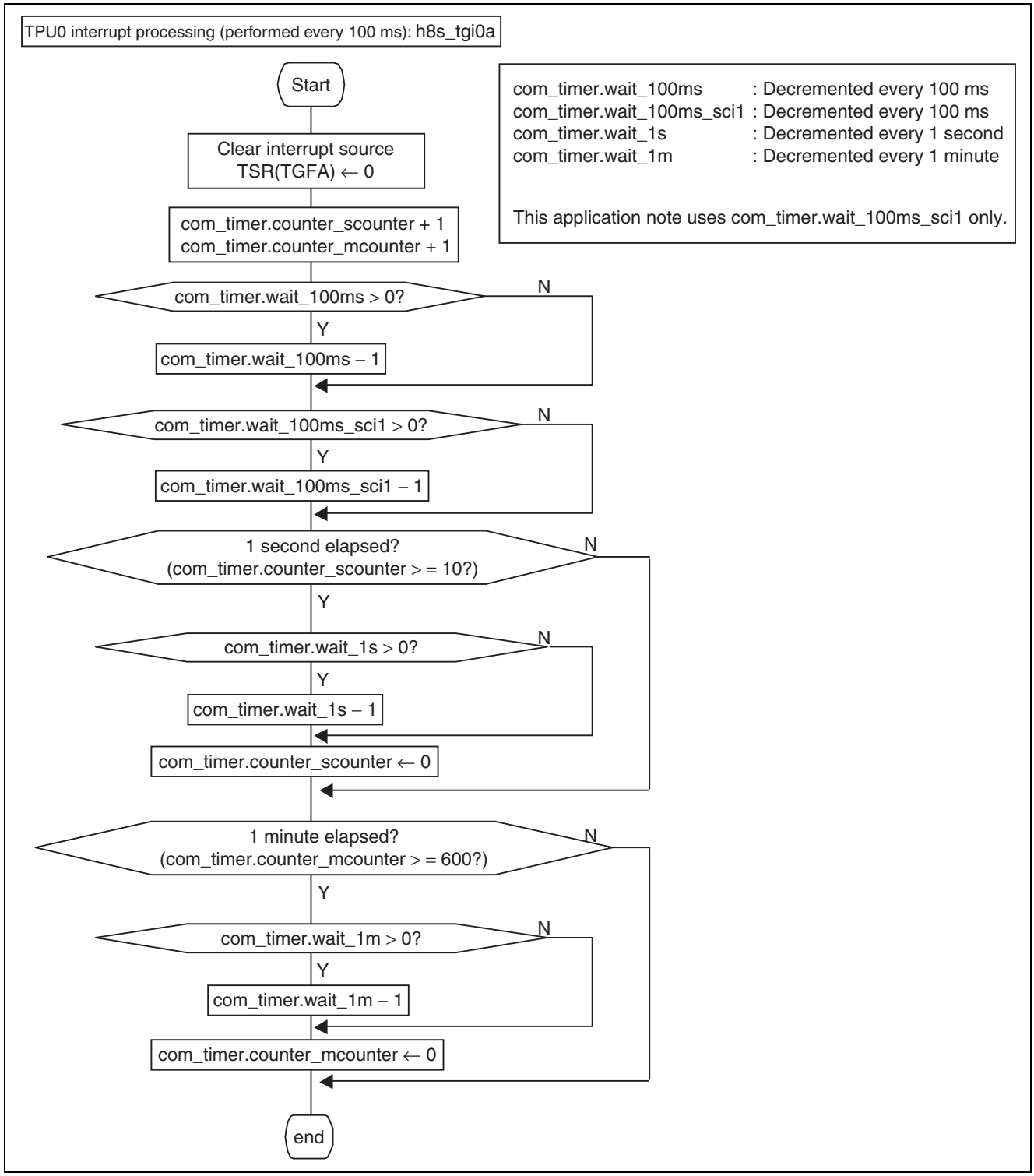
| Argument | Setting |
|---|---|
| sci_no | Specifies an SCI number.<br>**SCI_0** (0) : SCI_0<br>**SCI_1** (1) : SCI_1<br>With the H8S/2215, SCI_2 cannot be used as a DMA start source, so that SCI_2 may not be specified here. |
| ram_address | Specifies the start address of an area where data received from the SCI is to be stored. |
| length | Specifies the size of data to be transferred by DMA.<br>When data of a specified size has been transferred, a DMA end interrupt is generated. |

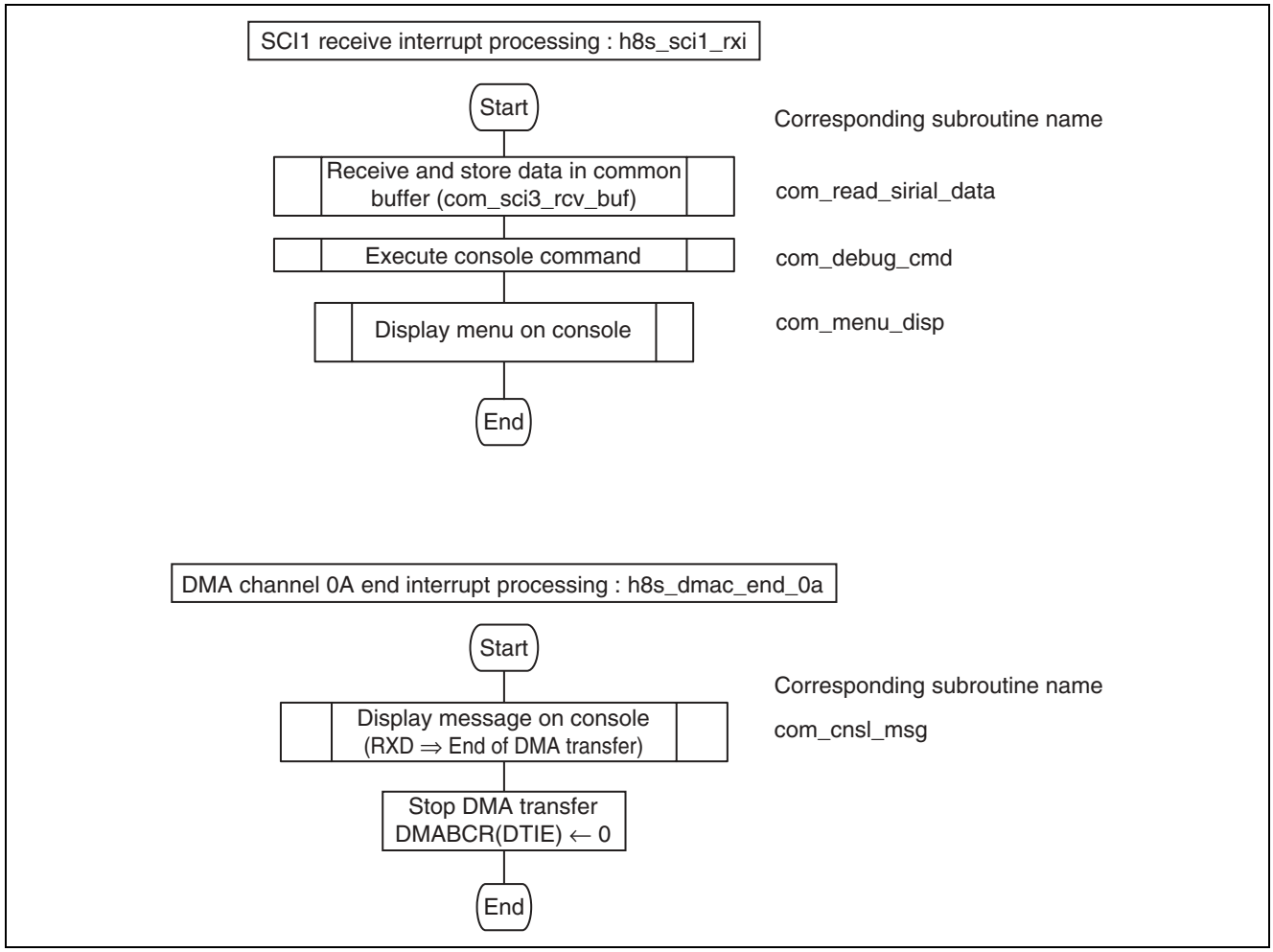
# 6. Flowchart

## 6.1 Overall flow



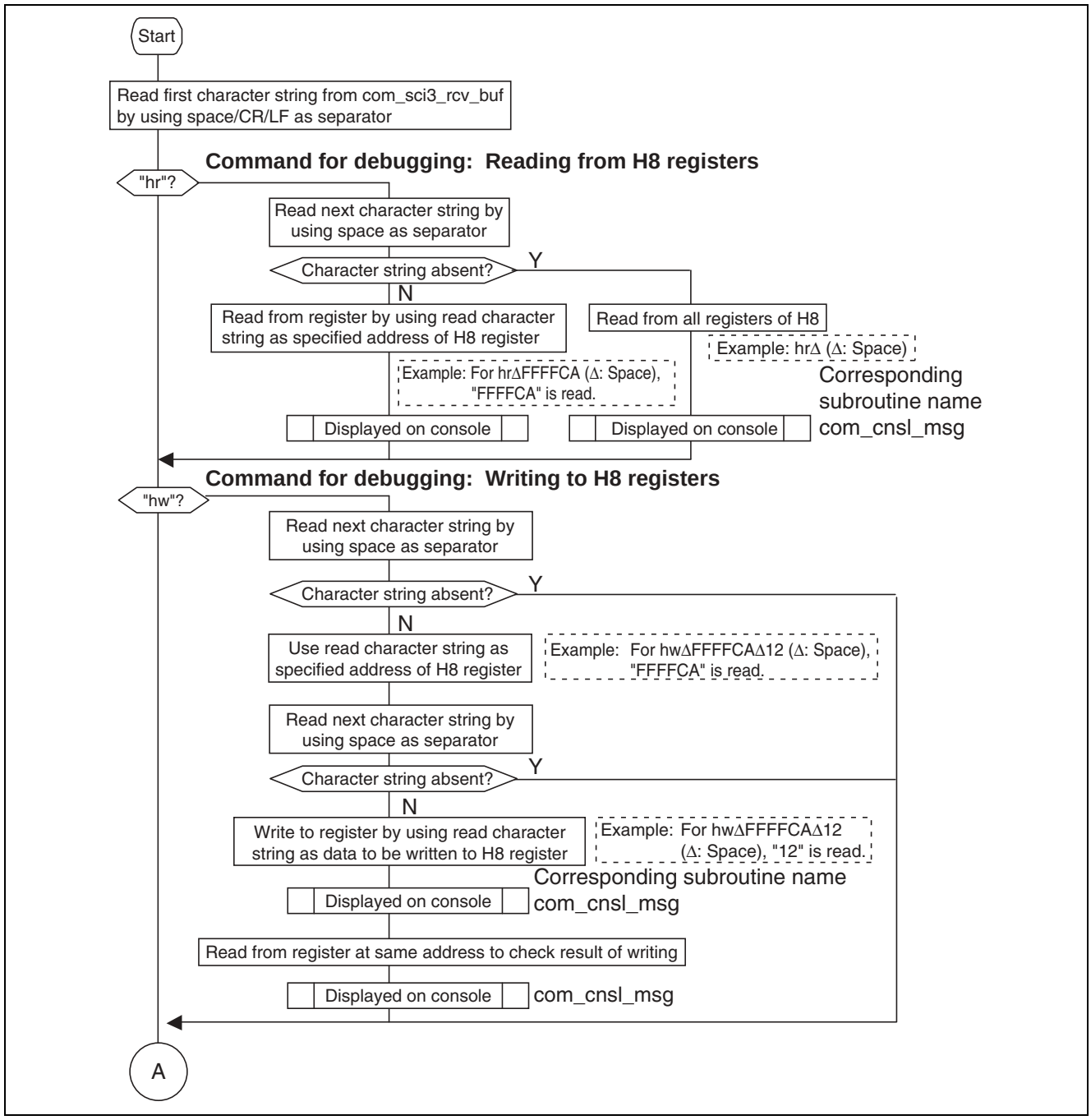
## 6.2 Interrupt processing

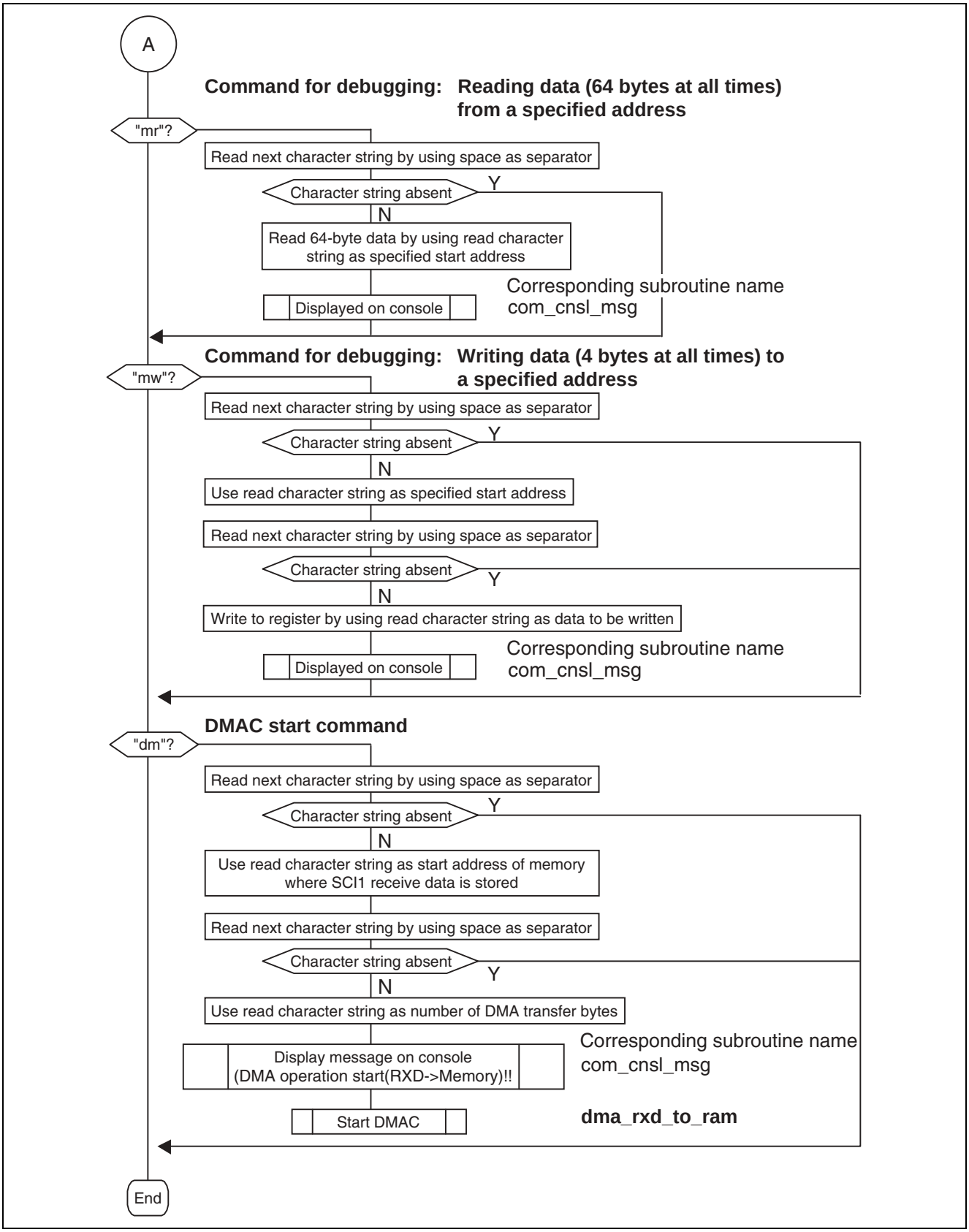



SCI1 receive interrupt processing : h8s_sci1_rxi
Start
Receive and store data in common buffer (com_sci3_rcv_buf)
Execute console command
Display menu on console
End
Corresponding subroutine name
com_read_sirial_data
com_debug_cmd
com_menu_disp
DMA channel 0A end interrupt processing : h8s_dmac_end_0a
Start
Display message on console (RXD ⇒ End of DMA transfer)
Stop DMA transfer DMABCR(DTIE) ← 0
End
Corresponding subroutine name
com_cnsl_msg

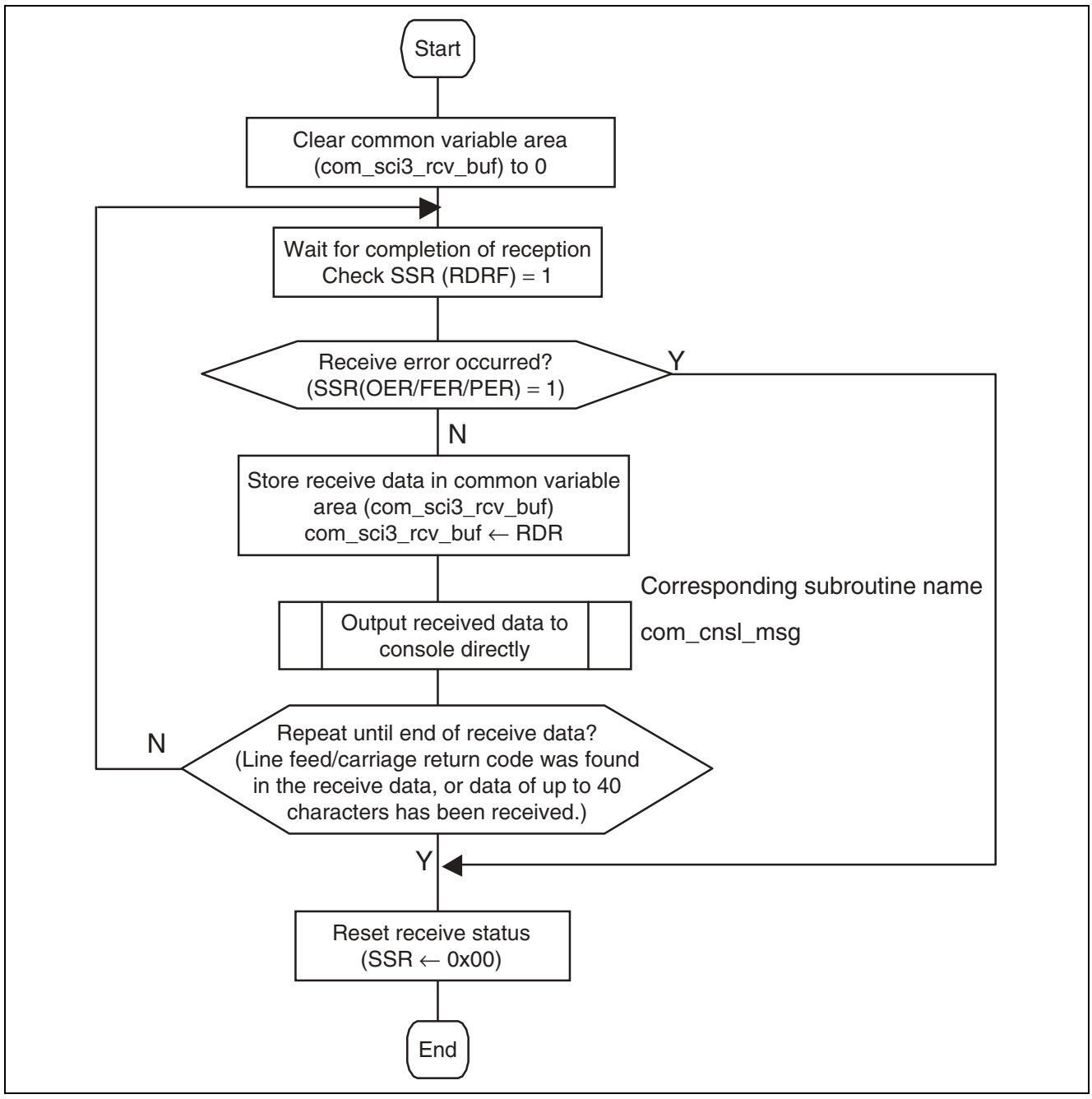
## 6.3 Detailed processing

```
com_debug_cmd
: Console command analysis and execution
```

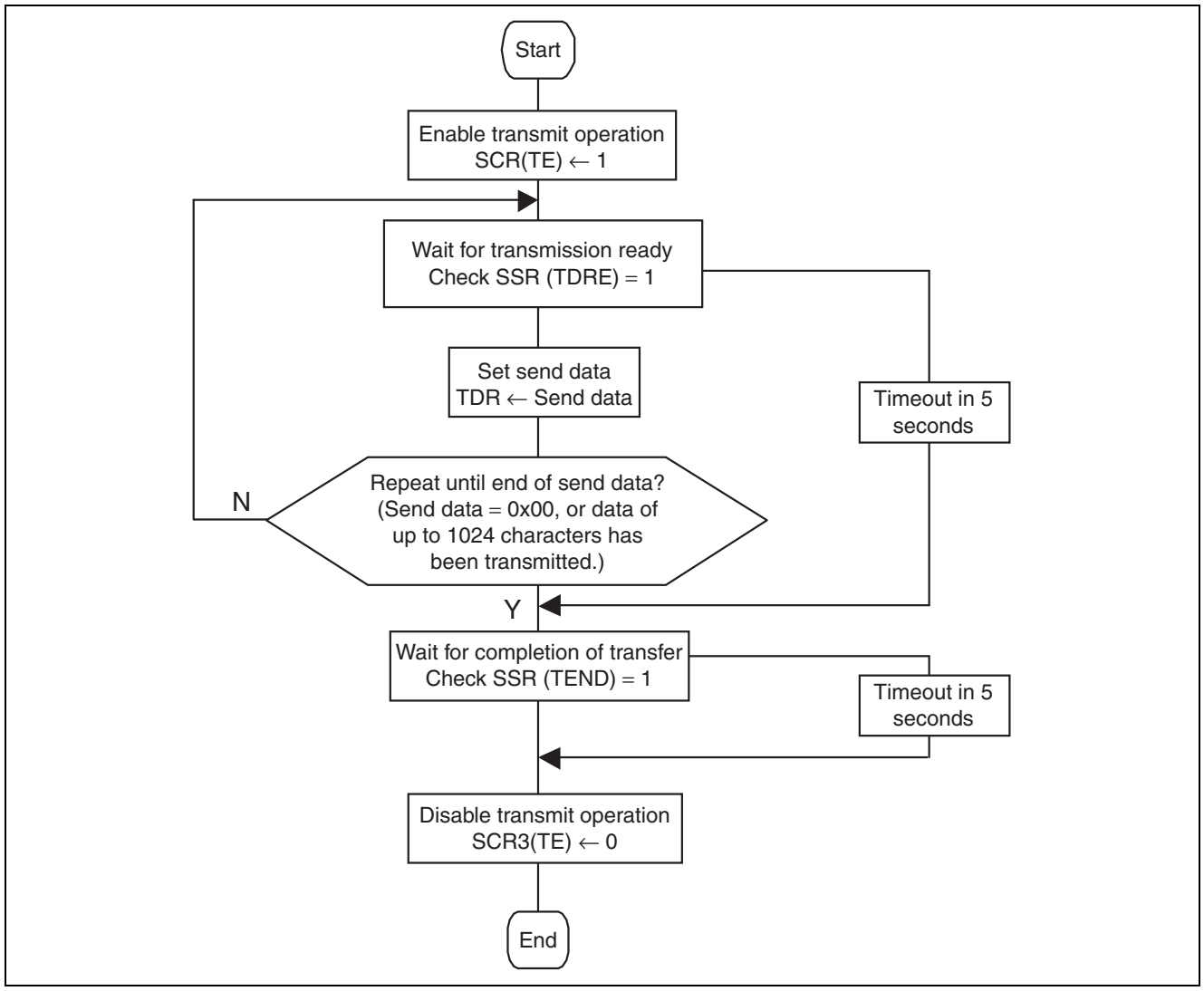



A
Command for debugging: Reading data (64 bytes at all times) from a specified address
"mr"?
Read next character string by using space as separator
Character string absent
Y
N
Read 64-byte data by using read character string as specified start address
Displayed on console
Corresponding subroutine name com_cnsl_msg
Command for debugging: Writing data (4 bytes at all times) to a specified address
"mw"?
Read next character string by using space as separator
Character string absent
Y
N
Use read character string as specified start address
Read next character string by using space as separator
Character string absent
Y
N
Write to register by using read character string as data to be written
Displayed on console
Corresponding subroutine name com_cnsl_msg
DMAC start command
"dm"?
Read next character string by using space as separator
Character string absent
Y
N
Use read character string as start address of memory where SCI1 receive data is stored
Read next character string by using space as separator
Character string absent
Y
N
Use read character string as number of DMA transfer bytes
Display message on console (DMA operation start(RXD->Memory)!!
Corresponding subroutine name com_cnsl_msg
Start DMAC
dma_rxd_to_ram
End

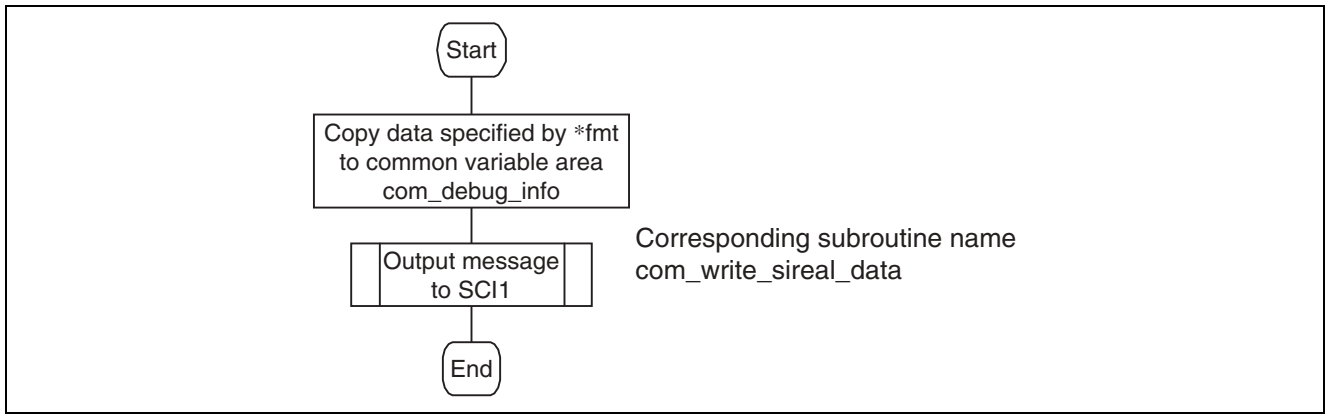
```
com_read_sirial_data
: Reception of a message from the SCI1 interface
   Receive data not longer than 40 characters is received
   in the common variable area (com_sci3_rcv_buf).
```



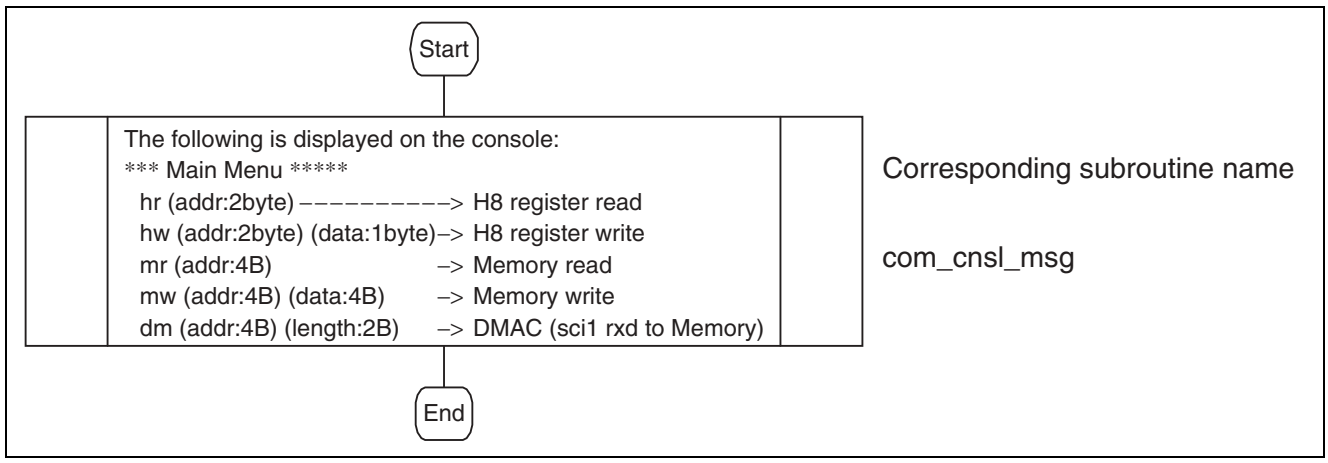
```
com_write_sireal_data ( char *p )
: Transmission of a message to the SCI3 interface
   *p : Address where message data is stored
```



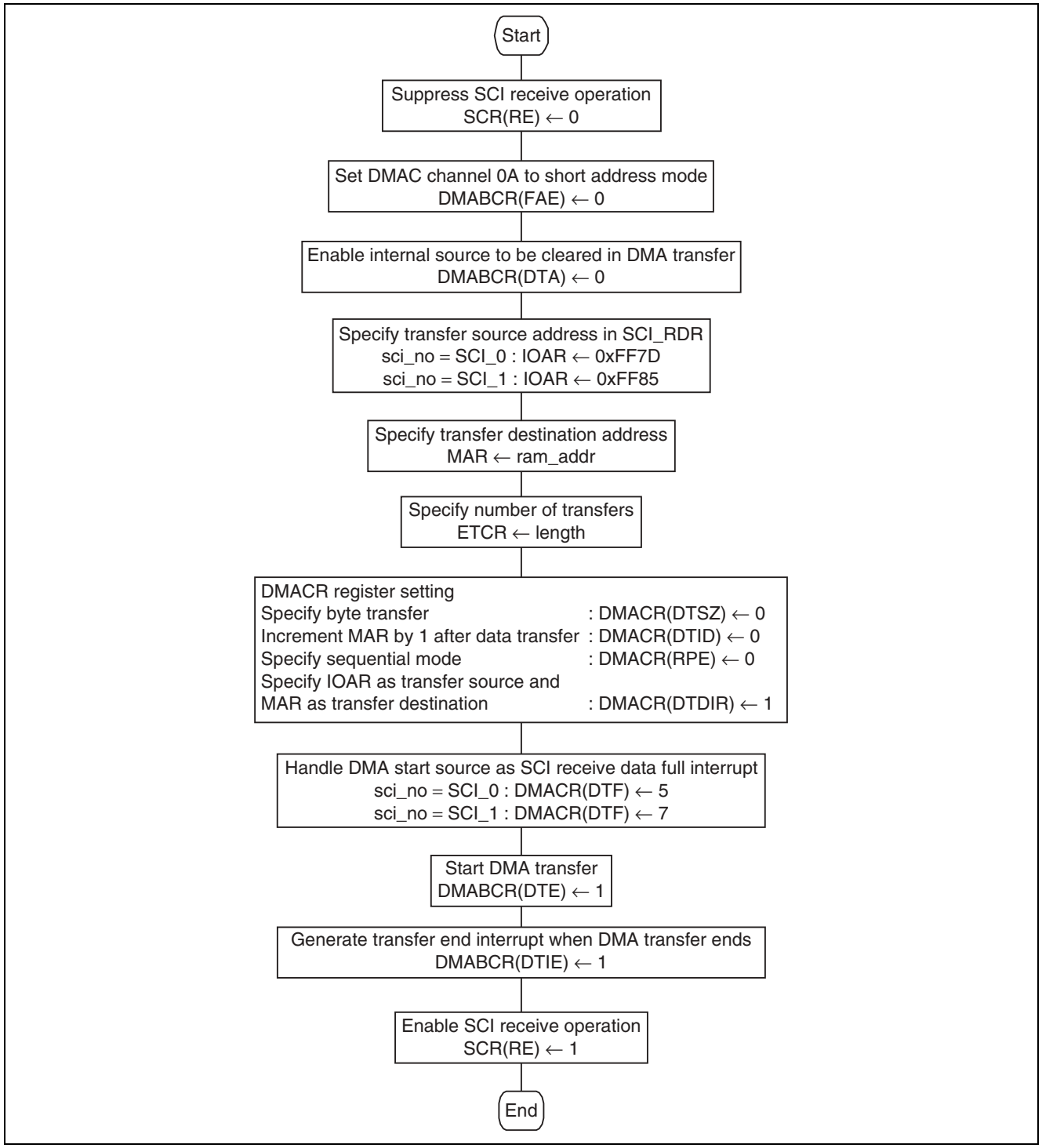
```
com_cnsl_msg(char *fmt, ...)
: Transmission of a message to the console
   *fmt : Address where variable-length message data is stored
```




```
com_menu_disp
: Display of the operation menu on the console
```



```
dma_rxd_to_ram ( int sci_no , unsigned long ram_addr , unsigned int length)
: DMA transfer of SCI1 receive data to a specified address
   sci_no : Specifies an SCI number (SCI_0 or SCI_1).
   ram_addr : Specifies the start address of an area where SCI receive data is
stored.
   length : Specifies the size of data to be transferred by DMA.
```



## Revision Record

| | | Description | |
|---|---|---|---|
| Rev. | Date | Page | Summary |
| 1.00 | Mar.16.04 | — | First edition issued |
| | | | |
| | | | |
| | | | |
| | | | |

## Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

## Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.