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

Mode Transition

Introduction

The power-down mode switching function switches among the high-speed mode, medium-speed mode, subactive mode, sleep mode, subsleep mode, watch mode, and software standby mode.

Target Device

H8S/2238

Contents

1. Overview

The switching function of H8S/2238 low-power consumption mode switches among the high-speed mode, medium-speed mode, subactive mode, sleep mode, subsleep mode, watch mode, and software standby mode. Besides these modes, the module stop mode and hardware standby mode are available but they are not explained in this application note. For details, see the Hardware Manual.

2. Configuration

Figure 1 shows the confirmed configuration of this application note.

List of Components Used

No.	Component	Specifications	
1	HSB8S2238F H8S/2238 CPU board (HOKUTO DENSHI CO., LTD.)	Board power supply input: 3 to 5 VDC Operating frequency: MCU operating mode:	 Main 12.2880 MHz Sub 37.768 kHz 7 (Single-chip mode)

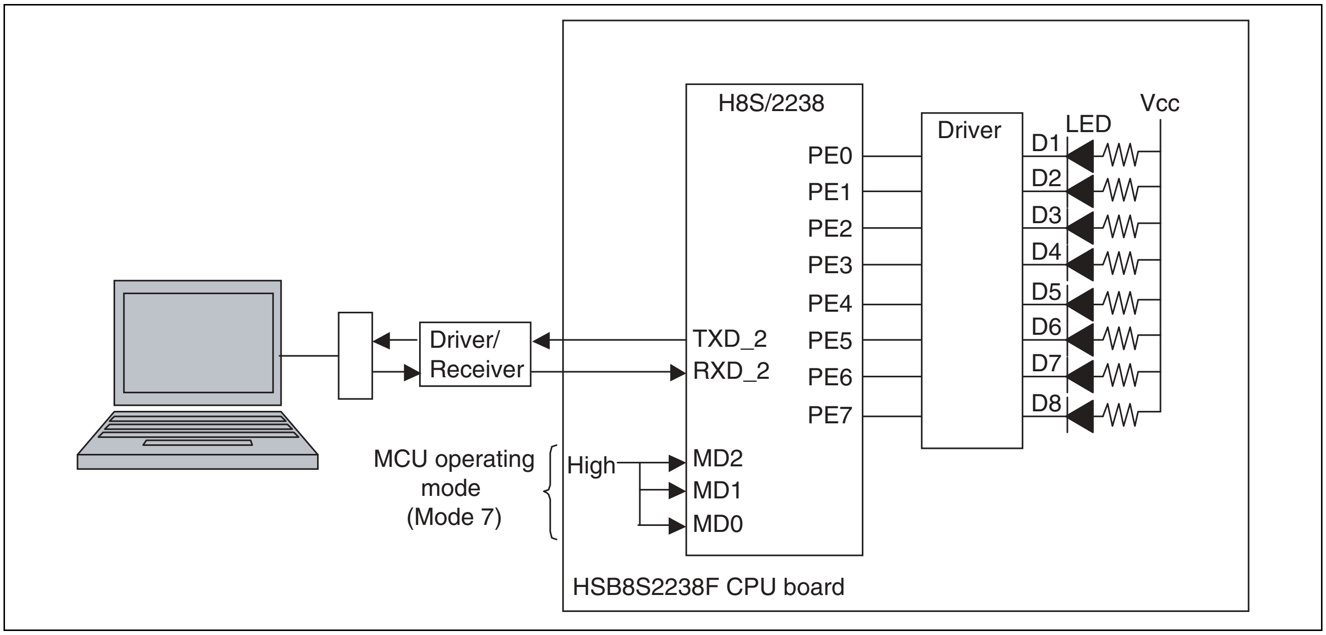


Figure 1 Confirmed Configuration

3. Description of Functions

The transition mode starts by inputting a command from the terminal software connected to the RS-232C interface.

1. Communication specifications
   The terminal software is connected 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

<table>
<tr><th>No.</th><th>Function</th><th>Specifications</th></tr>
<tr><td>1</td><td>High-speed ↔ transition to medium-speed mode</td><td>Format: m1Δ(Transition mode)
<table>
<tr><th>(Transition mode) Settings</th><th>Transition mode</th></tr>
<tr><td>0</td><td>High-speed mode</td></tr>
<tr><td>1</td><td>Medium-speed mode ϕ/2</td></tr>
<tr><td>2</td><td>Medium-speed mode ϕ/4</td></tr>
<tr><td>3</td><td>Medium-speed mode ϕ/8</td></tr>
<tr><td>4</td><td>Medium-speed mode ϕ/16</td></tr>
<tr><td>5</td><td>Medium-speed mode ϕ/32</td></tr>
</table></td></tr>
<tr><td>2</td><td>High-speed/medium-speed mode
→ Transition to subactive mode
→ Return to high-speed/medium-speed mode</td><td>Format: m2Δ(Return mode)
<table>
<tr><th>(Return mode) Settings</th><th>Return mode</th></tr>
<tr><td>0</td><td>High-speed mode</td></tr>
<tr><td>1</td><td>Medium-speed mode ϕ/2</td></tr>
<tr><td>2</td><td>Medium-speed mode ϕ/4</td></tr>
<tr><td>3</td><td>Medium-speed mode ϕ/8</td></tr>
<tr><td>4</td><td>Medium-speed mode ϕ/16</td></tr>
<tr><td>5</td><td>Medium-speed mode ϕ/32</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>No.</th><th>Function</th><th>Specifications</th></tr>
<tr><td>3</td><td>High-speed/medium-speed mode
→ Transition to the specified halt state
→ Release of program halt state by interrupt
→ Return to the specified mode</td><td>Format: m3Δ(Halt mode)Δ(Return mode)
<table>
<tr><th>Halt mode Settings</th><th>Halt transition mode</th></tr>
<tr><td>0</td><td>Sleep mode</td></tr>
<tr><td>1</td><td>Software standby mode</td></tr>
<tr><td>2</td><td>Watch mode</td></tr>
</table>
<table>
<tr><th>(Return mode) Settings</th><th>Return mode</th></tr>
<tr><td>0</td><td>High-speed mode</td></tr>
<tr><td>1</td><td>Medium-speed mode ϕ/2</td></tr>
<tr><td>2</td><td>Medium-speed mode ϕ/4</td></tr>
<tr><td>3</td><td>Medium-speed mode ϕ/8</td></tr>
<tr><td>4</td><td>Medium-speed mode ϕ/16</td></tr>
<tr><td>5</td><td>Medium-speed mode ϕ/32</td></tr>
<tr><td>6</td><td>Subactive mode</td></tr>
</table>
</td></tr>
<tr><td>4</td><td>High-speed/medium-speed mode
→ Transition to subactive mode
→ Transition to the specified halt state
→ Release of program halt state by interrupt
→ Return to the specified mode</td><td>Format: m4Δ(Halt mode)Δ(Return mode)
<table>
<tr><th>Halt mode Settings</th><th>Halt transition mode</th></tr>
<tr><td>2</td><td>Watch mode</td></tr>
<tr><td>3</td><td>Subsleep mode</td></tr>
</table>
<table>
<tr><th>(Return mode) Settings</th><th>Return mode</th></tr>
<tr><td>0</td><td>High-speed mode</td></tr>
<tr><td>1</td><td>Medium-speed mode ϕ/2</td></tr>
<tr><td>2</td><td>Medium-speed mode ϕ/4</td></tr>
<tr><td>3</td><td>Medium-speed mode ϕ/8</td></tr>
<tr><td>4</td><td>Medium-speed mode ϕ/16</td></tr>
<tr><td>5</td><td>Medium-speed mode ϕ/32</td></tr>
<tr><td>6</td><td>Subactive mode</td></tr>
</table>
</td></tr>
</table>

4. Principles of Operation

4.1 Initial Setting

Before exercising DTC control, start up the microcomputer and perform the initial setting of internal resister and other operations as needed.

1. Control of the low-power consumption, initialization of the clock oscillator

Register name ←Set value	Bit	Name	Value	Contents
LPWCR ←0x03	7	DTON	0	Directly transfer on flag
	6	LSON	0	Low-speed on flag
	5	NESEL	0	Sampling frequency (use of ϕ/32)
	4	SUBSTP	0	Operates the subclock oscillator
	3	RFCUT	0	Uses the internal feedback resistance control
	2	----	0	
	1:0	STC[1:0]	11	Bypasses the PLL
MSTPCRA ←0x8C	7	MSTPA7	1	(Reserved)
	6	MSTPA6	0	Operates DTC module
	5	MSTPA5	0	Operates the TPU module
	4	MSTPA4	0	Operates the TMR_0 and 1 modules
	3:2	MSTPA[3:2]	11	(Reserved)
	1	MSTPA1	0	Operates the AD module
	0	MSTPA0	0	Operates the TMR_2 and 3 modules
MSTPCRB ←0x1F	7	MSTPB7	0	Operates the SCI0 module
	6	MSTPB7	0	Operates the SCI1 module
	5	MSTPB7	0	Operates the SCI2 module
	4:0	MSTPB[4:0]	1111	(Reserved)
MSTPCRC ←0x4F	7	MSTPC7	0	Operates the SCI3 module
	6	MSTPC6	1	(Reserved)
	5	MSTPC5	0	Operates the DA module
	4	MSTPC4	0	Operates the PC break controller
	3:0	MSTPC[3:0]	11111	(Reserved)

2. Initialization of IO port

Set the input/output pins of port E and F as below. Set all other pins to the output mode.

Port	Register name ←Set value	Bit	Name	Value	Contents
E	PEDDR ←0xFF	7:0	PE[7:0]DDR	all"1"	Output (LED control)
	PEDR ←0xFF	7:0	PE[7:0]DR	all"1"	Delete All LEDs
F	PEDDR ←0Xf6	7:4	PF[7:4]DDR	all"1"	Output (Not used)
		3	PF3DDR	0	Input (IRQ3)
		2:1	PF[2:1]DDR	all"1"	Output (Not used)
		0	PF0DDR	0	Input (IRQ2)

3. Initialization of the TPU0 timer

Set the timer to be able to interrupt at every 100 ms for monitoring.

Register name ←Set value	Bit	Name	Value	Contents
TCR_0 ←0x23	7:5	CCLR[2:0]	001	Counter clear by the TGRA compare match
	4:3	CKEG[1:0]	00	Count with 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 ←19230	15:0	TGRA_0	19230	TBRA output compare value (To be set to interrupt at every 100 ms)
TCNT_0 ←0x0000	15:0	TCNT_0	0x0000	Counter clear
TSTR ←0x01	7:0	TSTR	0x01	Start TCNT_0 counter

4. Initialization of serial interface (SCI_2)
   Set the interface to connect to the terminal software for starting up DTC.

Register name **←Set value**	**Bit**	**Name**	**Value**	**Contents**
SCR_2 ←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, use internal clock
	7	TIE	0	Disables transmit interrupt.
SMR_2 ←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 = ϕ
SCMR_2 ←0x00	7:4	----	0000	
	3	DIR	0	LSB first
	2	INV	0	No data inversion
	1-0	----	00	
BRR ←9	7:0	BRR	9	Sets the transmission speed to 38400 bps.

— Await for at least one-stop bit. (38400 bps: About 30 µs)

— Enables the receive processing.

Register name **(Address←Set value)**	**Bit**	**Name**	**Value**	**Contents**
SCR_2 ←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, use internal clock

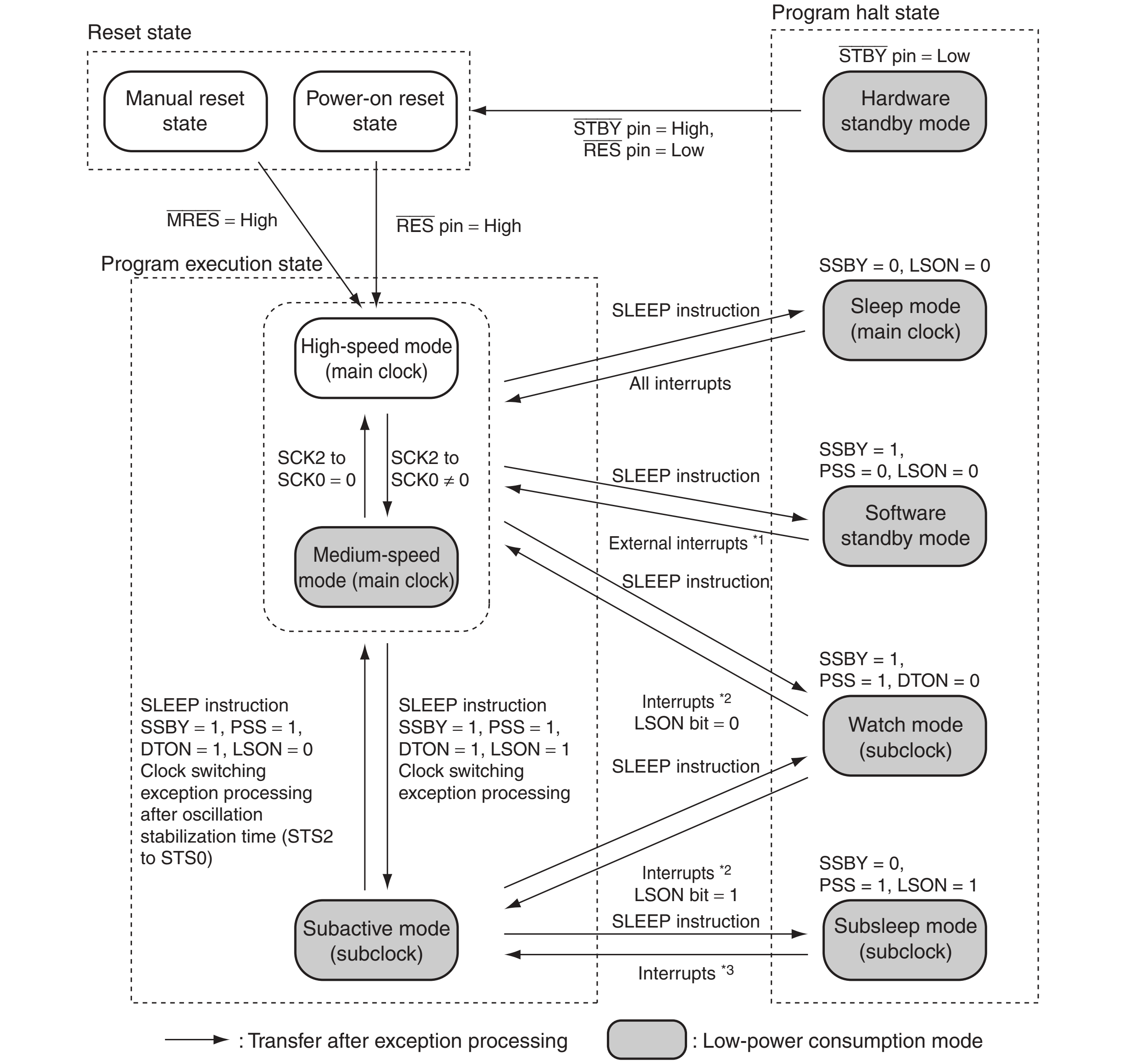
4.2 Mode Transition

Figure 2 shows possible transitions between modes. The SLEEP instruction instructs the transition from the program execution state to the program halt state. An interrupt instructs a return from the program halt state to the program execution state.

You can switch the high-speed mode to the medium-speed mode or vice versa (both modes are the program execution) simply by changing bit 2 (SCK2) to bit 0 (SCK0) in the SCKCR register.

You can also directly switch the high-speed or medium-speed mode to subactive mode without halting program execution.

Table 1 shows transitional conditions for each mode when the SLEEP instruction is executed. The table also shows states of return by interrupts. Table 1 shows the LSI internal states in each operating mode.



Notes: [1] NMI, IRQ0 to IRQ7, and WDT1 interrupt

[2] NMI, IRQ0 to IRQ7, WDT0 interrupt and WDT1 interrupt

[3] NMI and IRQ0 to IRQ7 interrupt

- Only the occurrence of an interrupt source prevents the triggering of transition between modes. Be sure that an interrupt request has been accepted before performing interrupt processing.
- When the RES pin is driven low in all states other than the hardware standby mode, the current state switches to the reset state. When the MRES pin is driven low in all states other than the hardware standby mode and the power-on reset state, the current state switches to the manual reset state.
- When the STBY pin is driven low in all states, the current state switches to the hardware standby mode.
- Be sure to set the high-speed mode when switching to the watch mode and subactive mode.

Figure 2 Diagram of Mode Transition

Table 1 Transitional Conditions in Each Mode when SLEEP Instruction Is Executed and States of Return by Interrupts

Pre-transitional state	Control bit state at transition				Post-transitional state of SLEEP instruction	Post-return state by interruption
	SSBY	PSS	LSON	DTON		
High speed or medium speed	0	×	0	×	Sleep	High speed or medium speed
	0	×	1	×	—	—
	1	0	0	×	Software standby	High speed or medium speed
	1	0	1	×	—	—
	1	1	0	0	Watch	High speed
	1	1	1	0	Watch	Subactive
	1	1	0	1	—	—
	1	1	1	1	Subactive	—
Subactive	0	0	×	×	—	—
	0	1	0	×	—	—
	0	1	1	×	Subsleep	Subactive
	1	0	×	×	—	—
	1	1	0	0	Watch	High speed
	1	1	1	0	Watch	Subactive
	1	1	0	1	High speed	—
	1	1	1	1	—	—

[Note]

— : Do not set these bits.

× : Don't care

Table 4.2 LSI Internal State in Each Operating Mode

Function		High speed	Medium speed	Sleep	Module stop	Watch	Sub-active	Sub-sleep	Software standby	Hard-ware standby
System clock oscillator		Operation	Operation	Operation	Operation	Stop	Stop	Stop	Stop	Stop
Subclock oscillator		Operation/ stop	Operation/ stop	Operation/ stop	Operation/ stop	Operation	Operation	Operation	Operation/ stop	Stop
CPU operation	Instruc-tion	Operation	Medium-speed operation	Stop	Operation	Stop	Subclock operation	Stop	Stop	Stop
	Register			Retention		Retention		Retention	Retention	Undefined
RAM		Operation	Operation	Operation (DTC)	Operation	Retention	Operation	Retention	Retention	Retention
I/O		Operation	Operation	Operation	Operation	Retention	Operation	Operation	Retention	High impedance
External interrupt	NMI	Operation	Operation	Operation	Operation	Operation	Operation	Operation	Operation	Stop
	IRQn									
Peripheral function	PBC	Operation	Medium-speed operation	Operation	Operation/ stop (retention)	Stop (retention)	Subclock operation	Stop (retention)	Stop (retention)	Stop (reset)
	DTC	Operation	Medium-speed operation	Operation	Operation/ stop (retention)	Stop (retention)	Stop (retention)	Stop (retention)	Stop (retention)	Stop (reset)
	DMAC[*1]									
	WDT_1	Operation	Operation	Operation	Operation	Subclock operation	Subclock operation	Subclock operation	Stop (retention)	Stop (reset)
	WDT_0	Operation	Operation	Operation	Operation	Stop (retention)	Subclock operation	Subclock operation	Stop (retention)	Stop (reset)
	TMR	Operation	Operation	Operation	Operation/ stop (retention)	Stop (retention)	Subclock operation	Subclock operation	Stop (retention)	Stop (reset)
	TPU	Operation	Operation	Operation	Operation/ stop (retention)	Stop (retention)	Stop (retention)	Stop (retention)	Stop (retention)	Stop (reset)
	SCI									
	I^{2}C[*2]									
	D/A[*3]									
	A/D	Operation	Operation	Operation	Operation/ stop (reset)	Stop (reset)	Stop (reset)	Stop (reset)	Stop (reset)	Stop (reset)

[Note]
"Stop (retention)" indicates that the internal register values are retained. The internal state is disrupted.
"Stop (reset)" indicates that the internal register values and the internal state are initialized.
In the module stop mode, only the modules which is set to stop is set stop (reset or retention).

[*1] Only the H8S/2239 products provides this function.

[*2] The H8S/2237 and H8S/2227 products do not provide the I^{2}C bus interface.

[*3] The H8S/2227 products does not provide D/A.

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 refer to 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. DTC vector address is also set in this file.
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	h8s2238.c	Main routine of this application note
5	com.c	Main common and interrupt processing routine of this application note
6	2238S.H	Structure definition file of the internal registers of the H8S/2238 products This file can be obtained from [2].
7	cwtbl.h	Variable and constant for this application note are defined.
8	prototypeh	A prototype for this application note is declared.
9	stacksct.h	A stack size is defined.

[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 referred to or set with "Category : section" in the Link/Library tab of the -"Standard Toolchain" of "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, Mode transition can be set and started using one subroutine. This function eases the use of mode transition.

1. Directly performs the transmission between modes.

   Subroutine name: **int** com_active_mode_change (**int** return_mode)

Parameter	Setting		
return_mode	Specifies the transition destination mode.		
	Settings	Value	Transition destination mode
	ACTIVE_MODE_PER_1	0	High-speed mode
	ACTIVE_MODE_PER_2	1	Medium-speed mode ϕ/2
	ACTIVE_MODE_PER_4	2	Medium-speed mode ϕ/4
	ACTIVE_MODE_PER_8	3	Medium-speed mode ϕ/8
	ACTIVE_MODE_PER_16	4	Medium-speed mode ϕ/16
	ACTIVE_MODE_PER_32	5	Medium-speed mode ϕ/32
	SUB_ACTIVE_MODE	6	Subactive mode

Return value	Description
0	Normal end of mode transition
–1	A mode that cannot be switched was specified. • An attempt was made to the switch subactive mode to the subactive mode.

2. This subroutine switches to the program halt state and then to the specified mode.

   Subroutine name: **int** com_sleep_mode_change (**int** stop_mode , **int** return_mode)

Parameter	Setting		
stop_mode	Specifies the program halt state mode.		
	Settings	Value	Transition destination mode
	SLEEP_MODE	0	Sleep mode
	STANDBY_MODE	1	Software standby mode
	WATCH_MODE	2	Watch mode
	SUB_SLEEP_MODE	3	Subsleep mode
return_mode	Specifies the return destination mode.		
	Settings	Value	Transition destination mode
	ACTIVE_MODE_PER_1	0	High-speed mode
	ACTIVE_MODE_PER_2	1	Medium-speed mode ϕ/2
	ACTIVE_MODE_PER_4	2	Medium-speed mode ϕ/4
	ACTIVE_MODE_PER_8	3	Medium-speed mode ϕ/8
	ACTIVE_MODE_PER_16	4	Medium-speed mode ϕ/16
	ACTIVE_MODE_PER_32	5	Medium-speed mode ϕ/32
	SUB_ACTIVE_MODE	6	Subactive mode

Return value	Description of setting
0	Normal end of mode transition
–1	Indicates that A mode disabled to switched was specified. • When the current mode is the subactive mode, the mode transition to a program halt state other than the watch mode and subsleep mode was specified. • When the current mode is the subactive mode, the transition to a state other than the subactive mode was specified after transition to the subsleep mode. • When the current mode is the high-speed/medium-speed mode, the transition to a program halt state (subsleep mode) was specified. • When the current mode is the high-speed/medium-speed mode, the transition to the subactive mode was specified after the transition to a program halt state (sleep mode). • When the current mode is the high-speed/medium-speed mode, the transition to the subactive mode was specified after the transition to a program halt state (software standby mode).

5.4 Notes

To switch from the software standby mode, watch mode, or subactive mode to another mode, you must set the clock stabilization standby time with bits 2 to 0 (STS2 to STS0) in the SBYCR register. In this application note, the clock stabilization standby time is fixed at "STS[2:0] = 100" (standby time = 130172 states). Modify this fixed value appropriately according to the operating frequency of the microcomputer you are using. For information about where the clock stabilization standby time is modified, see the source code of subroutines in Section 5.3.

For information about how to calculate settings, see Section 23.11 in the "Hardware Manual."

5.5 LED Status Display

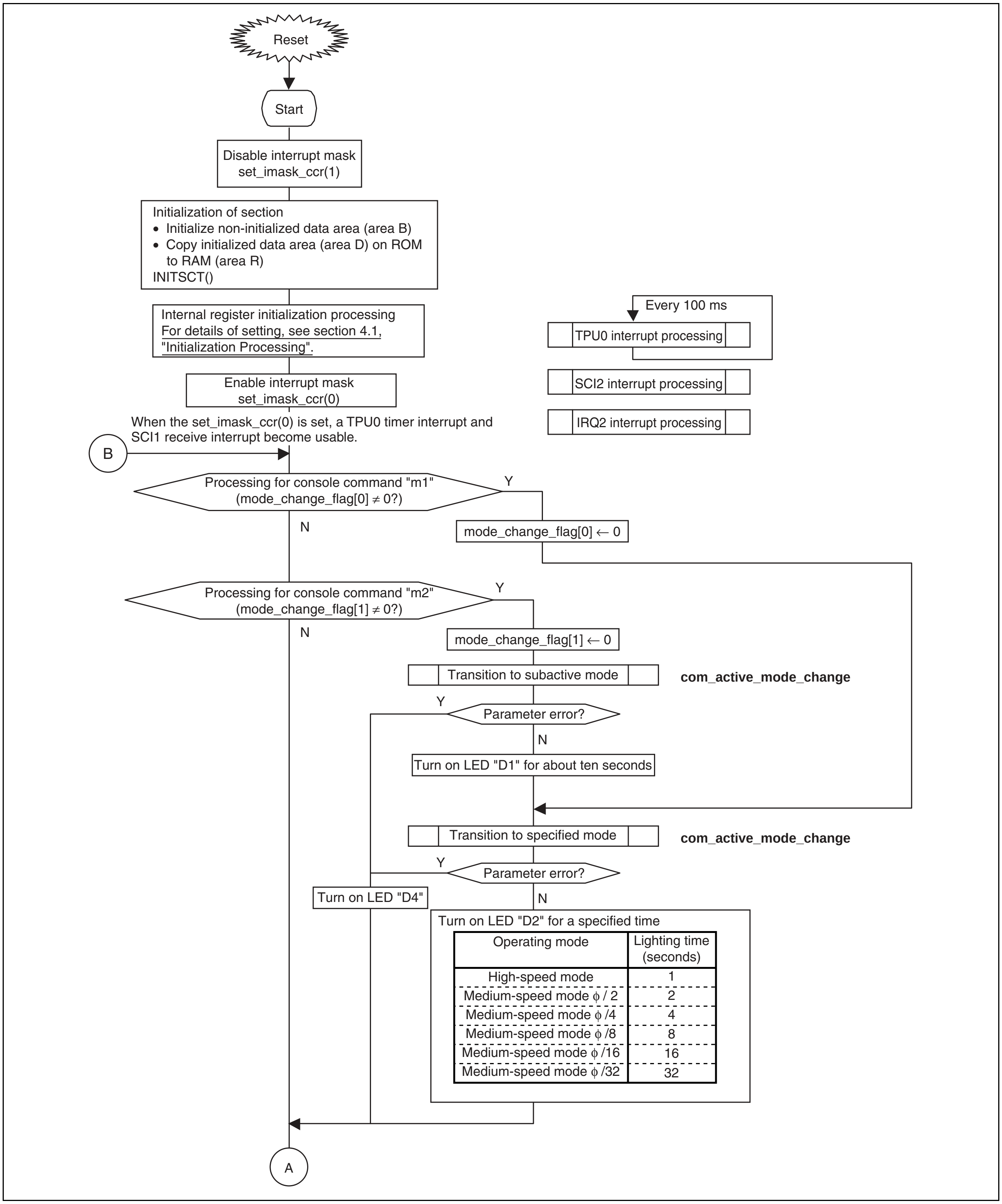
In this application note, LED ON/OFF indicates the mode transition state.

LED name	Description
D1	Illuminates for ten seconds when the current mode switches to the subactive mode.
D2	Illuminates for several seconds according to the CPU operating speed when the current mode switches to the high-speed/medium-speed mode.
D3	Illuminates when the H8S/2238 waits for the IRQ2 interrupt in the program halt state. Extinguishes when the IRQ2 interrupt occurs.
D4	Illuminates when the mode transition specification parameter error occurs.
D5 to D8	(Not used)

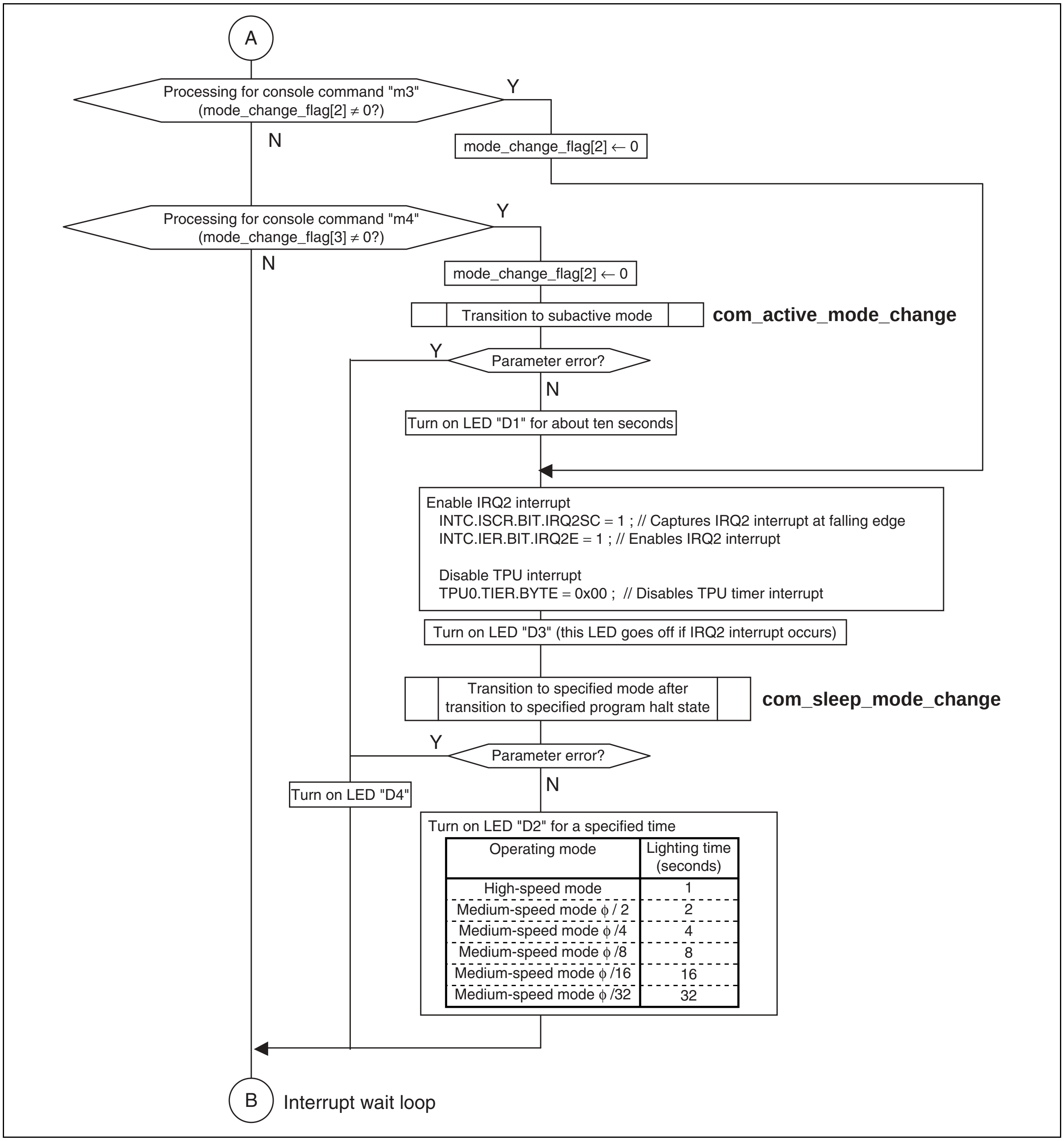
Operating mode	Lighting time (seconds)
High-speed mode	1
Medium-speed mode ϕ/2	2
Medium-speed mode ϕ/4	4
Medium-speed mode ϕ/8	8
Medium-speed mode ϕ/16	16
Medium-speed mode ϕ/32	32

6. Flowchart

6.1 Overall Flow


Reset
Start
Disable interrupt mask
set_imask_ccr(1)
Initialization of section
• Initialize non-initialized data area (area B)
• Copy initialized data area (area D) on ROM to RAM (area R)
INITSCT()
Internal register initialization processing
For details of setting, see section 4.1, "Initialization Processing".
Enable interrupt mask
set_imask_ccr(0)
When the set_imask_ccr(0) is set, a TPU0 timer interrupt and SCI1 receive interrupt become usable.
Every 100 ms
TPU0 interrupt processing
SCI2 interrupt processing
IRQ2 interrupt processing
B
Processing for console command "m1"
(mode_change_flag[0] ≠ 0?)
Y
N
mode_change_flag[0] ← 0
Processing for console command "m2"
(mode_change_flag[1] ≠ 0?)
Y
N
mode_change_flag[1] ← 0
Transition to subactive mode
com_active_mode_change
Y
Parameter error?
N
Turn on LED "D1" for about ten seconds
Transition to specified mode
com_active_mode_change
Y
Parameter error?
Turn on LED "D4"
N
Turn on LED "D2" for a specified time
A

Operating mode	Lighting time (seconds)
High-speed mode	1
Medium-speed mode φ / 2	2
Medium-speed mode φ /4	4
Medium-speed mode φ /8	8
Medium-speed mode φ /16	16
Medium-speed mode φ /32	32

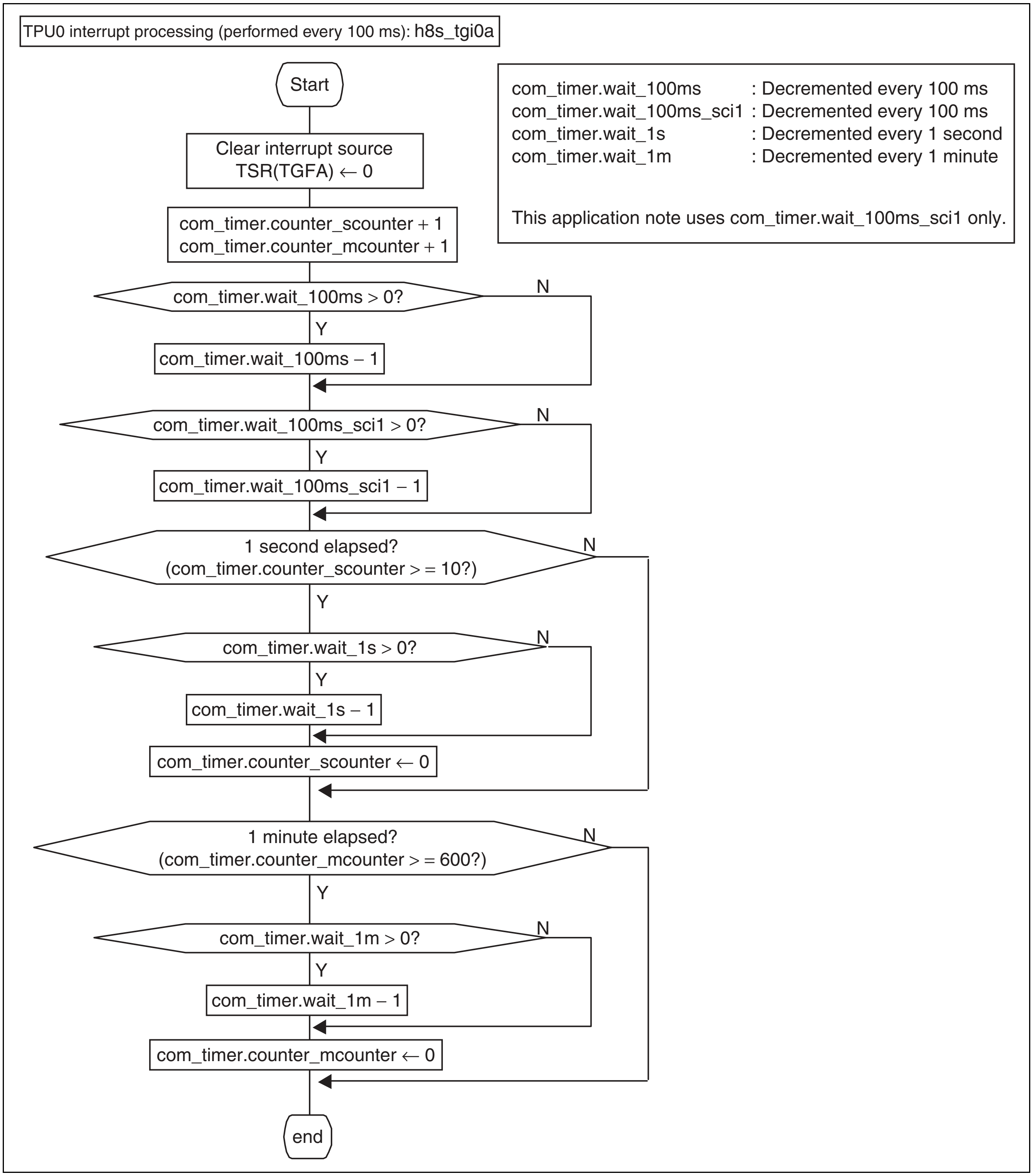

A
Processing for console command "m3" (mode_change_flag[2] ≠ 0?)
Y
N
mode_change_flag[2] ← 0
Processing for console command "m4" (mode_change_flag[3] ≠ 0?)
Y
N
mode_change_flag[2] ← 0
Transition to subactive mode
com_active_mode_change
Y
Parameter error?
N
Turn on LED "D1" for about ten seconds
Turn on LED "D3" (this LED goes off if IRQ2 interrupt occurs)
Transition to specified mode after transition to specified program halt state
com_sleep_mode_change
Y
Parameter error?
N
Turn on LED "D4"
Turn on LED "D2" for a specified time
B
Interrupt wait loop

```
Enable IRQ2 interrupt
  INTC.ISCR.BIT.IRQ2SC = 1 ; // Captures IRQ2 interrupt at falling edge
  INTC.IER.BIT.IRQ2E = 1 ; // Enables IRQ2 interrupt

  Disable TPU interrupt
  TPU0.TIER.BYTE = 0x00 ;  // Disables TPU timer interrupt
```

Operating mode	Lighting time (seconds)
High-speed mode	1
Medium-speed mode ϕ / 2	2
Medium-speed mode ϕ /4	4
Medium-speed mode ϕ /8	8
Medium-speed mode ϕ /16	16
Medium-speed mode ϕ /32	32

6.2 Interrupt Processing

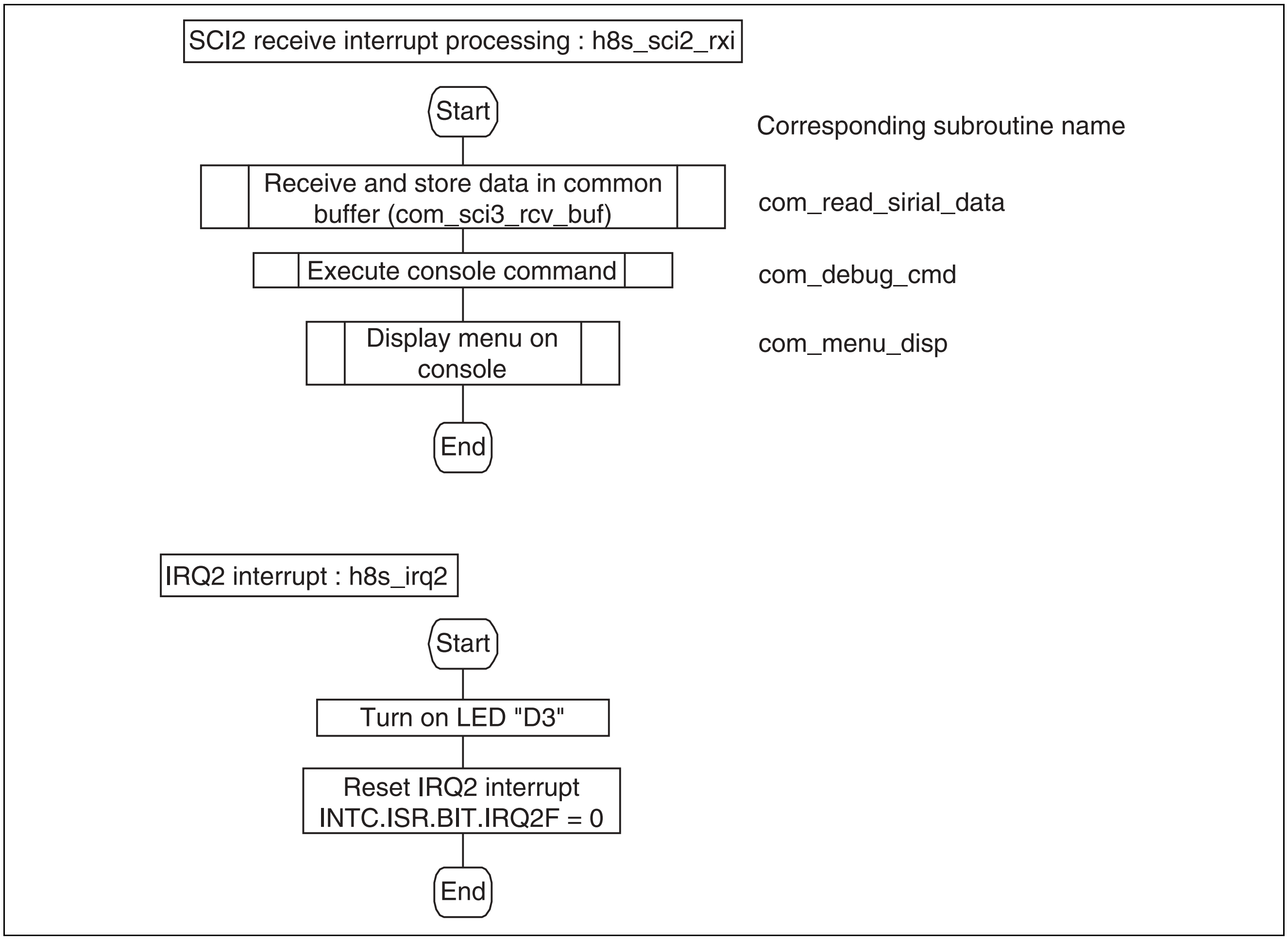


TPU0 interrupt processing (performed every 100 ms): h8s_tgi0a

com_timer.wait_100ms	: Decremented every 100 ms
com_timer.wait_100ms_sci1	: Decremented every 100 ms
com_timer.wait_1s	: Decremented every 1 second
com_timer.wait_1m	: Decremented every 1 minute

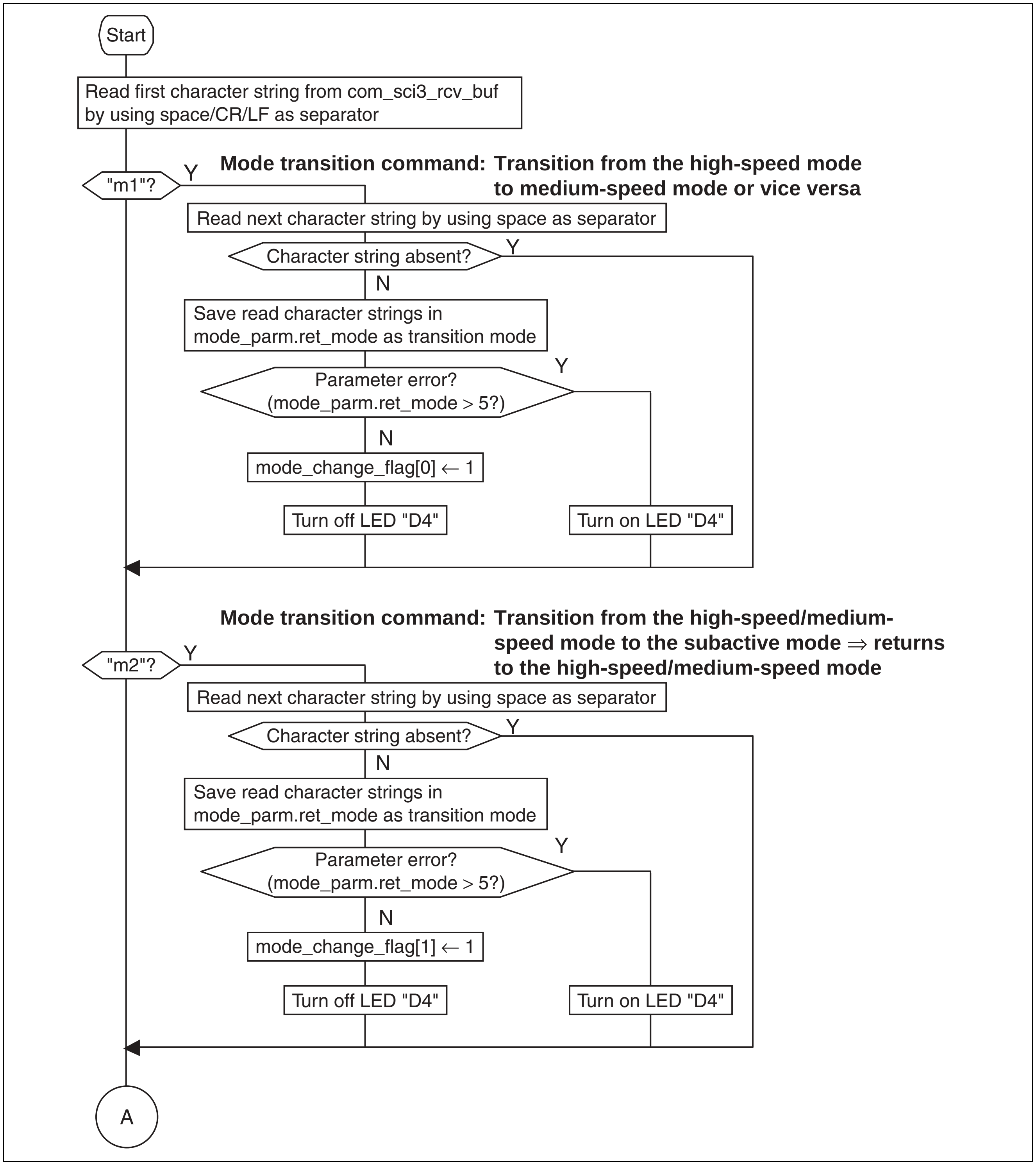
This application note uses com_timer.wait_100ms_sci1 only.

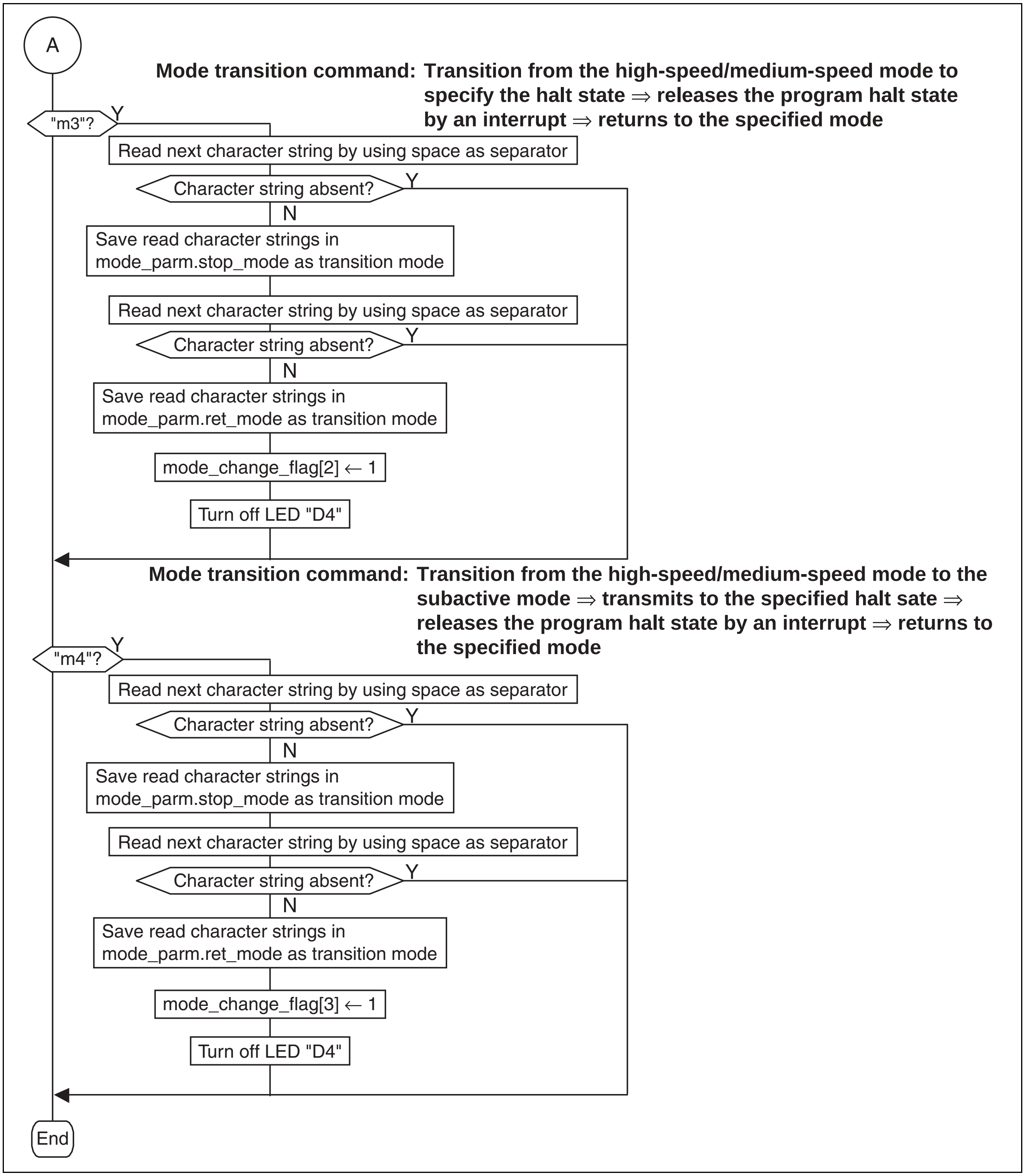
Start
Clear interrupt source
TSR(TGFA) ← 0
com_timer.counter_scounter + 1
com_timer.counter_mcounter + 1
com_timer.wait_100ms > 0?
N
Y
com_timer.wait_100ms – 1
com_timer.wait_100ms_sci1 > 0?
N
Y
com_timer.wait_100ms_sci1 – 1
1 second elapsed?
(com_timer.counter_scounter > = 10?)
N
Y
com_timer.wait_1s > 0?
N
Y
com_timer.wait_1s – 1
com_timer.counter_scounter ← 0
1 minute elapsed?
(com_timer.counter_mcounter > = 600?)
N
Y
com_timer.wait_1m > 0?
N
Y
com_timer.wait_1m – 1
com_timer.counter_mcounter ← 0
end


SCI2 receive interrupt processing : h8s_sci2_rxi
Start
Corresponding subroutine name
Receive and store data in common buffer (com_sci3_rcv_buf)
com_read_sirial_data
Execute console command
com_debug_cmd
Display menu on console
com_menu_disp
End
IRQ2 interrupt : h8s_irq2
Start
Turn on LED "D3"
Reset IRQ2 interrupt INTC.ISR.BIT.IRQ2F = 0
End

6.3 Detailed Processing

com_debug_cmd
: Console command analysis and execution


Start
Read first character string from com_sci3_rcv_buf by using space/CR/LF as separator
"m1"?
Y
Mode transition command: Transition from the high-speed mode to medium-speed mode or vice versa
Read next character string by using space as separator
Character string absent?
Y
N
Save read character strings in mode_parm.ret_mode as transition mode
Parameter error? (mode_parm.ret_mode > 5?)
Y
N
mode_change_flag[0] ← 1
Turn off LED "D4"
Turn on LED "D4"
Mode transition command: Transition from the high-speed/medium-speed mode to the subactive mode ⇒ returns to the high-speed/medium-speed mode
"m2"?
Y
Read next character string by using space as separator
Character string absent?
Y
N
Save read character strings in mode_parm.ret_mode as transition mode
Parameter error? (mode_parm.ret_mode > 5?)
Y
N
mode_change_flag[1] ← 1
Turn off LED "D4"
Turn on LED "D4"
A

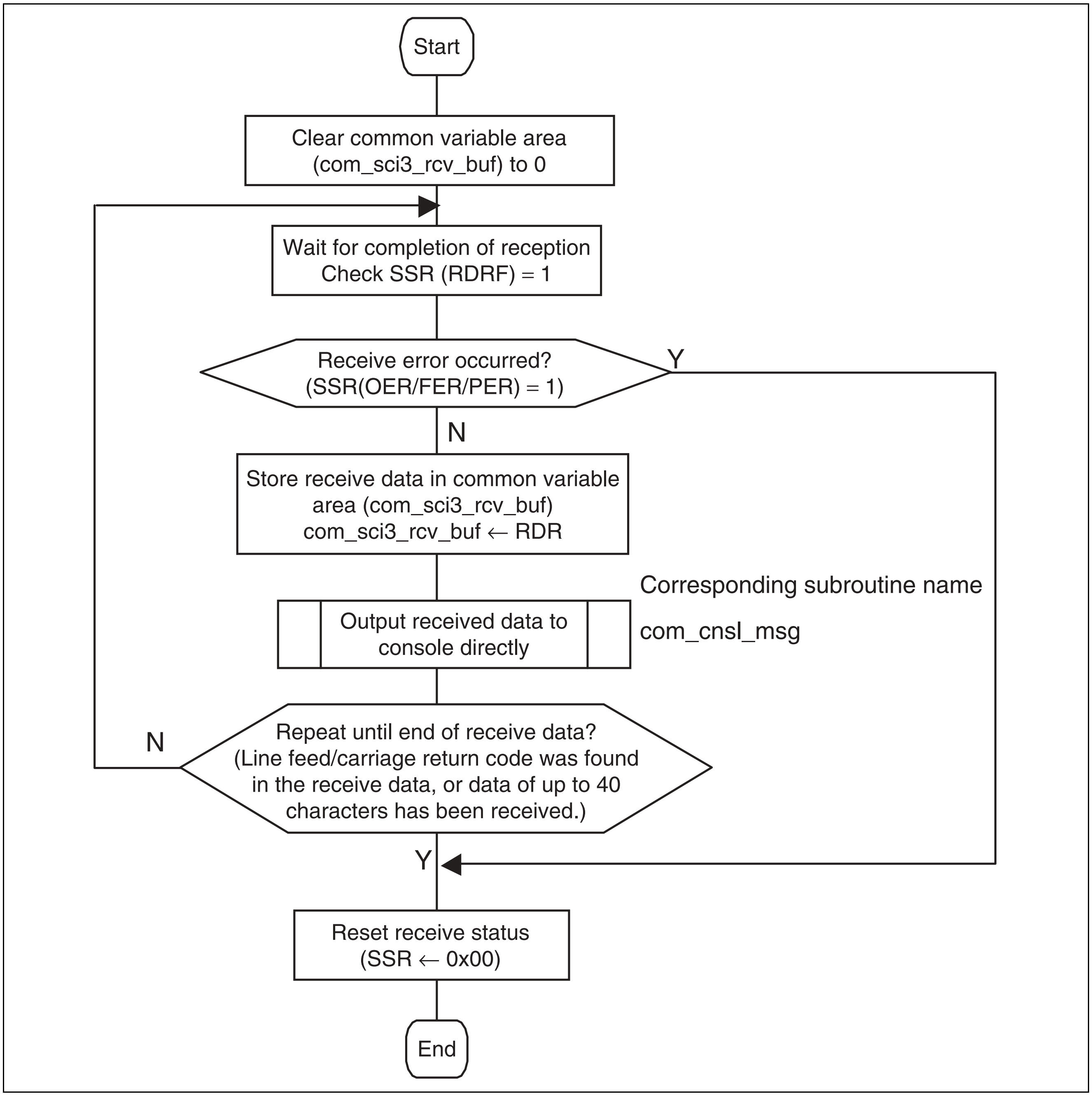


Mode transition command: Transition from the high-speed/medium-speed mode to specify the halt state ⇒ releases the program halt state by an interrupt ⇒ returns to the specified mode

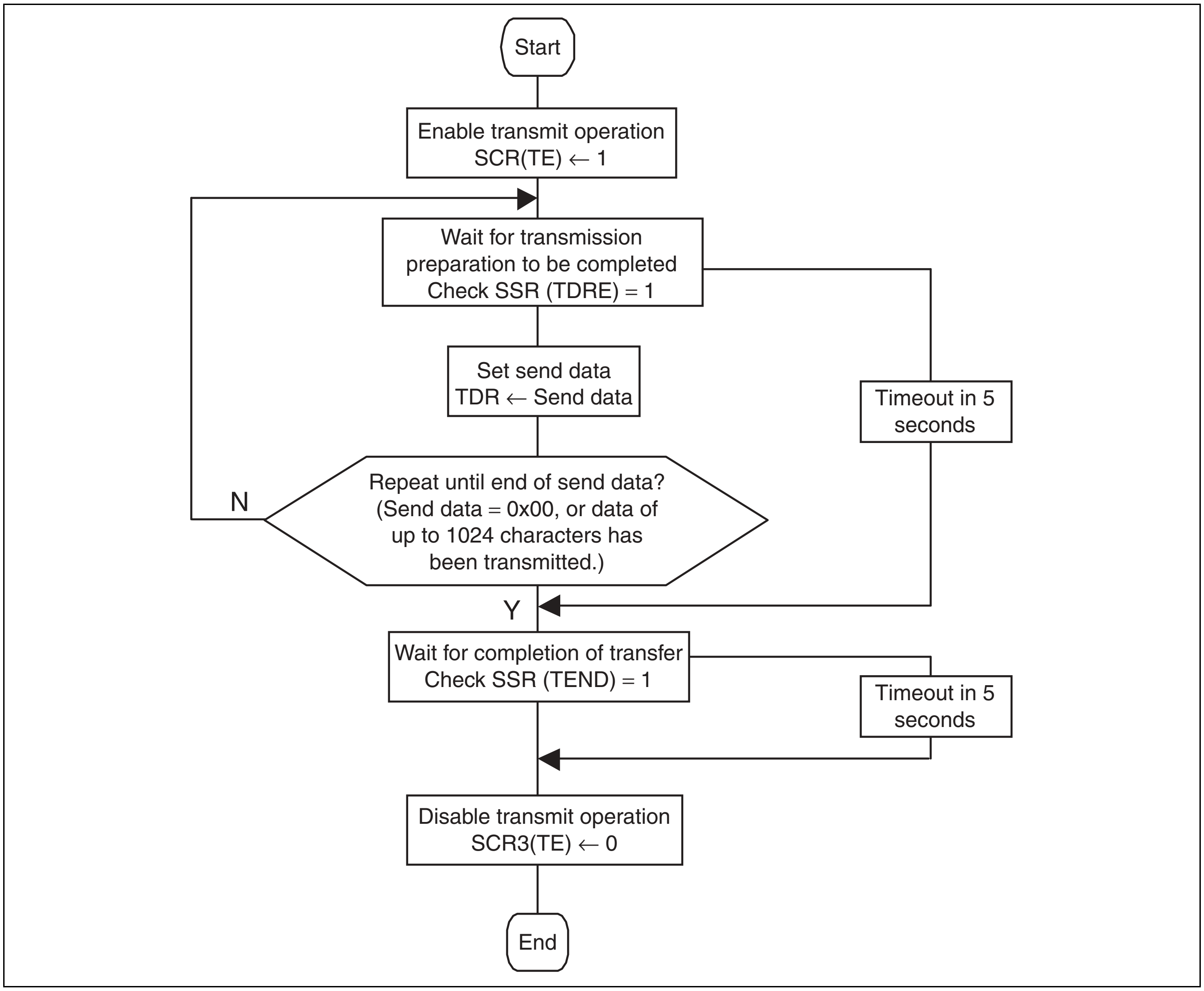
Mode transition command: Transition from the high-speed/medium-speed mode to the subactive mode ⇒ transmits to the specified halt sate ⇒ releases the program halt state by an interrupt ⇒ returns to the specified mode

A
"m3"?
Y
Read next character string by using space as separator
Character string absent?
Y
N
Save read character strings in mode_parm.stop_mode as transition mode
Read next character string by using space as separator
Character string absent?
Y
N
Save read character strings in mode_parm.ret_mode as transition mode
mode_change_flag[2] ← 1
Turn off LED "D4"
"m4"?
Y
Read next character string by using space as separator
Character string absent?
Y
N
Save read character strings in mode_parm.stop_mode as transition mode
Read next character string by using space as separator
Character string absent?
Y
N
Save read character strings in mode_parm.ret_mode as transition mode
mode_change_flag[3] ← 1
Turn off LED "D4"
End

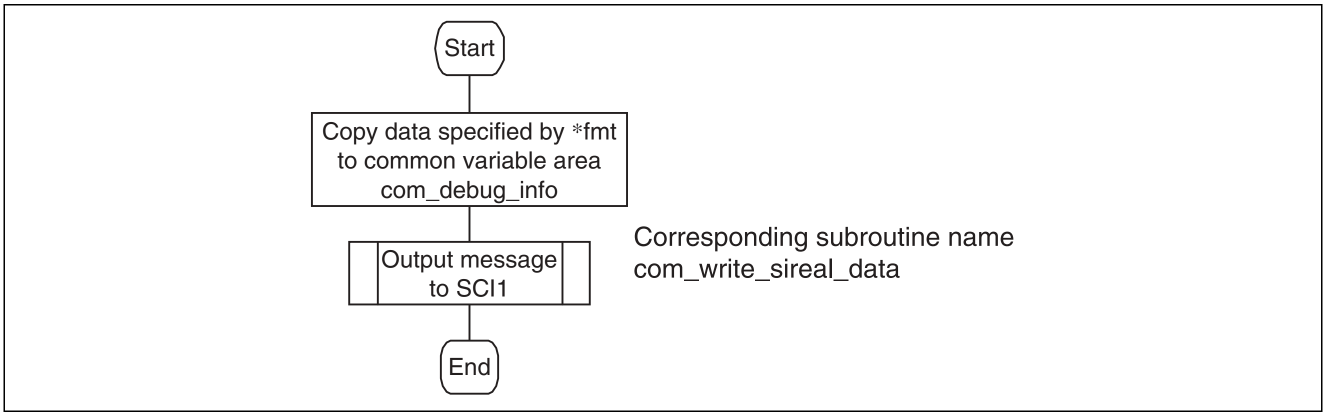
com_read_sirial_data
: Receives a message from the SCI1 interface
Received data with maximum of 40 characters is received
in the common variable area (com_sci3_rcv_buf).


Start
Clear common variable area
(com_sci3_rcv_buf) to 0
Wait for completion of reception
Check SSR (RDRF) = 1
Receive error occurred?
(SSR(OER/FER/PER) = 1)
Y
N
Store receive data in common variable
area (com_sci3_rcv_buf)
com_sci3_rcv_buf ← RDR
Corresponding subroutine name
Output received data to
console directly
com_cnsl_msg
N
Repeat until end of receive data?
(Line feed/carriage return code was found
in the receive data, or data of up to 40
characters has been received.)
Y
Reset receive status
(SSR ← 0x00)
End

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

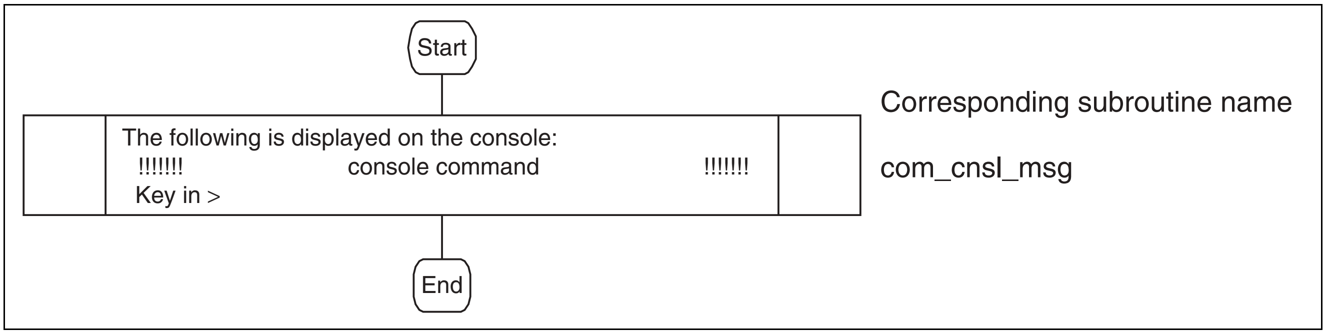

Start
Enable transmit operation
SCR(TE) ← 1
Wait for transmission
preparation to be completed
Check SSR (TDRE) = 1
Set send data
TDR ← Send data
Timeout in 5
seconds
N
Repeat until end of send data?
(Send data = 0x00, or data of
up to 1024 characters has
been transmitted.)
Y
Wait for completion of transfer
Check SSR (TEND) = 1
Timeout in 5
seconds
Disable transmit operation
SCR3(TE) ← 0
End

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


Start
Copy data specified by *fmt to common variable area com_debug_info
Output message to SCI1
End

Corresponding subroutine name
com_write_sireal_data

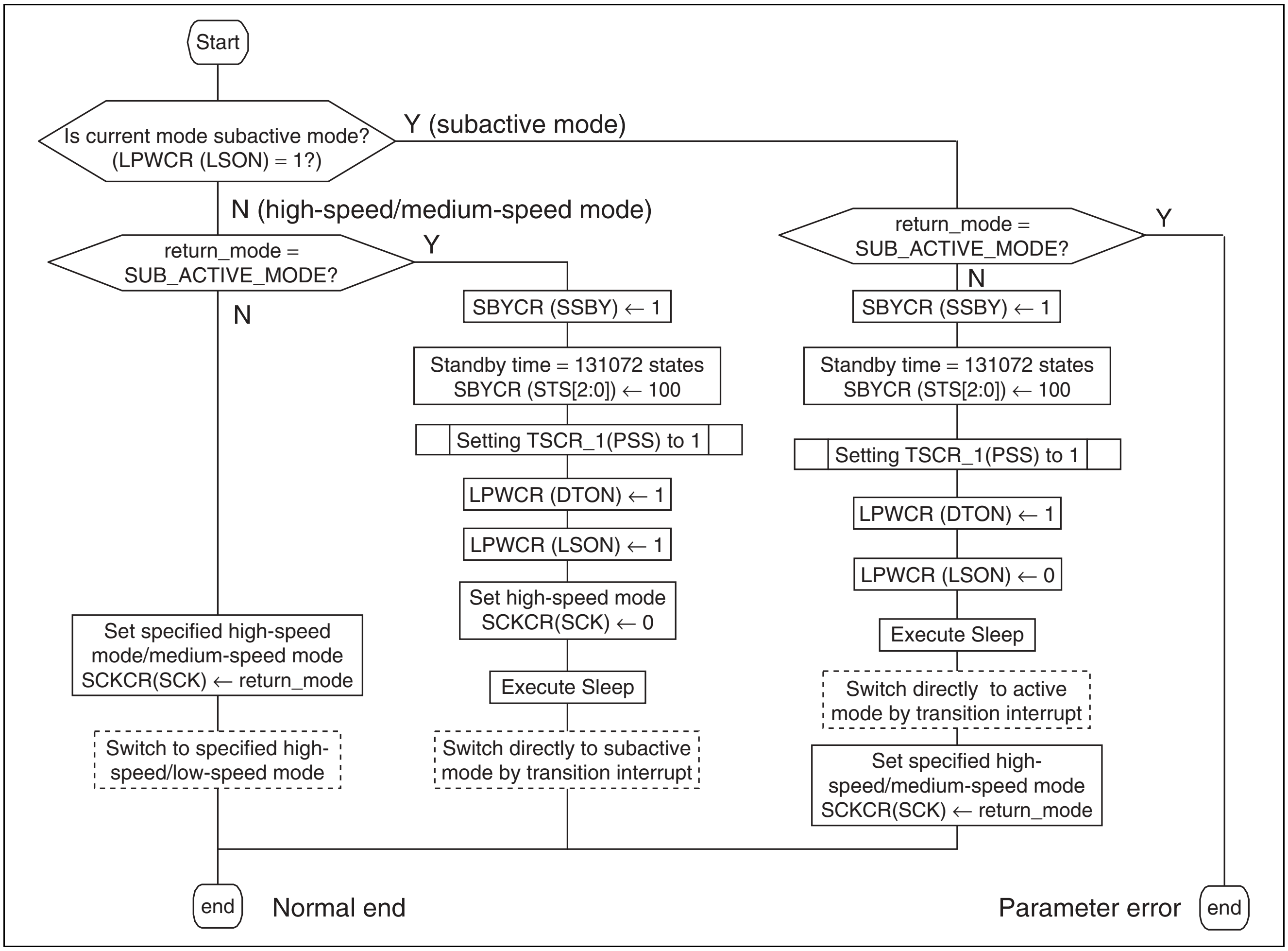
com_menu_disp
: Display of the operation menu on the console


Start
The following is displayed on the console:
!!!!!!! console command !!!!!!!
Key in >
End

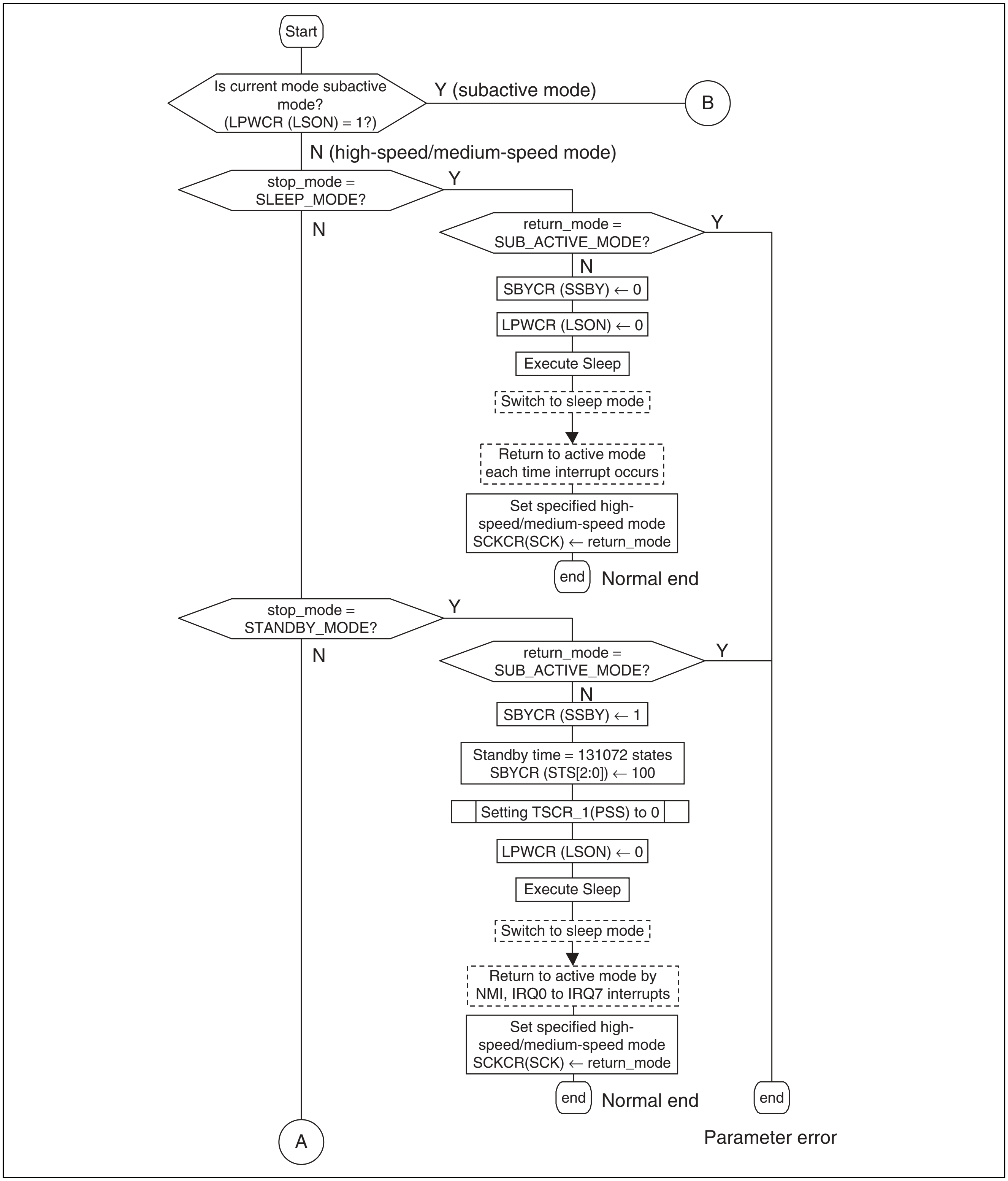
Corresponding subroutine name

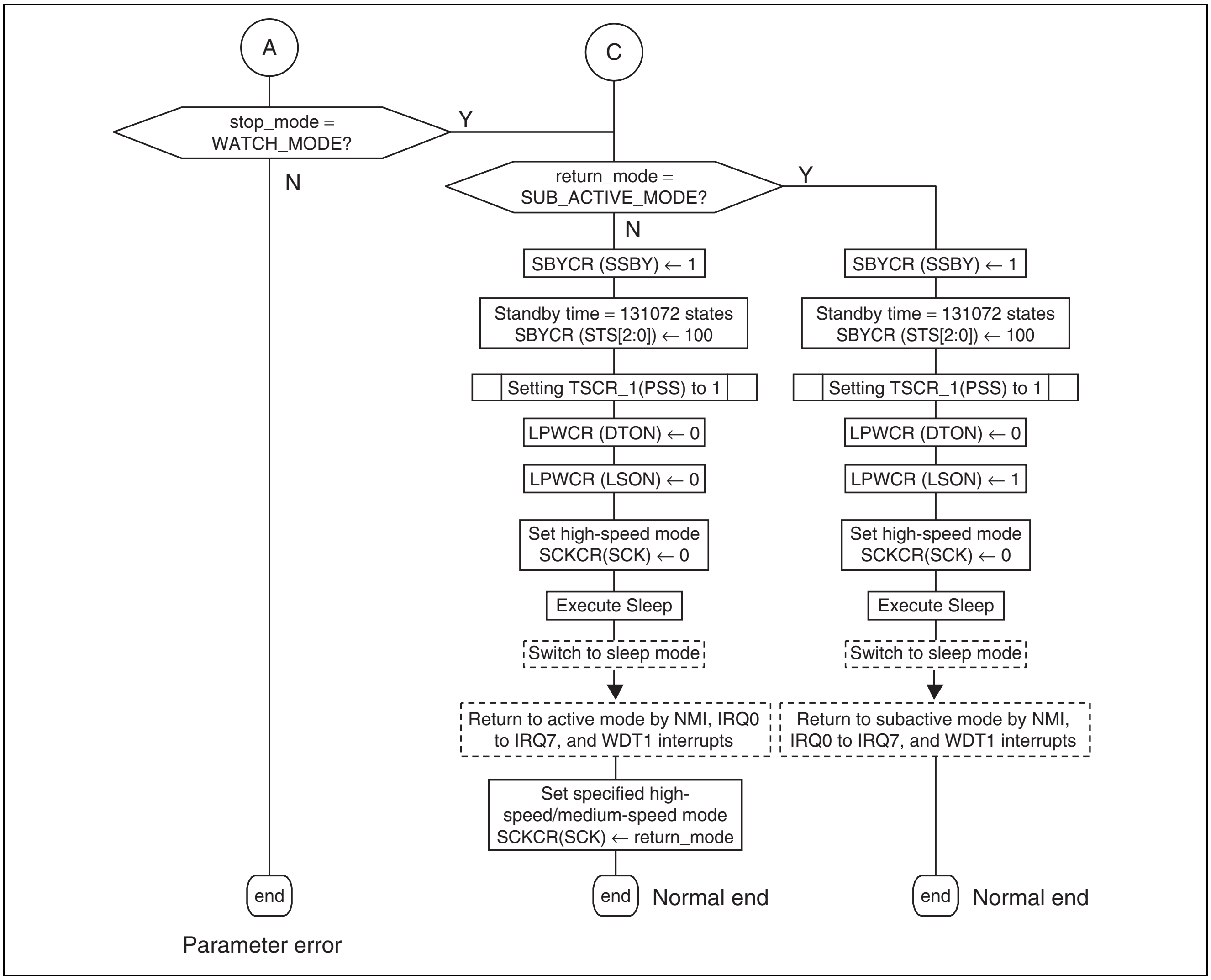
com_cnsl_msg

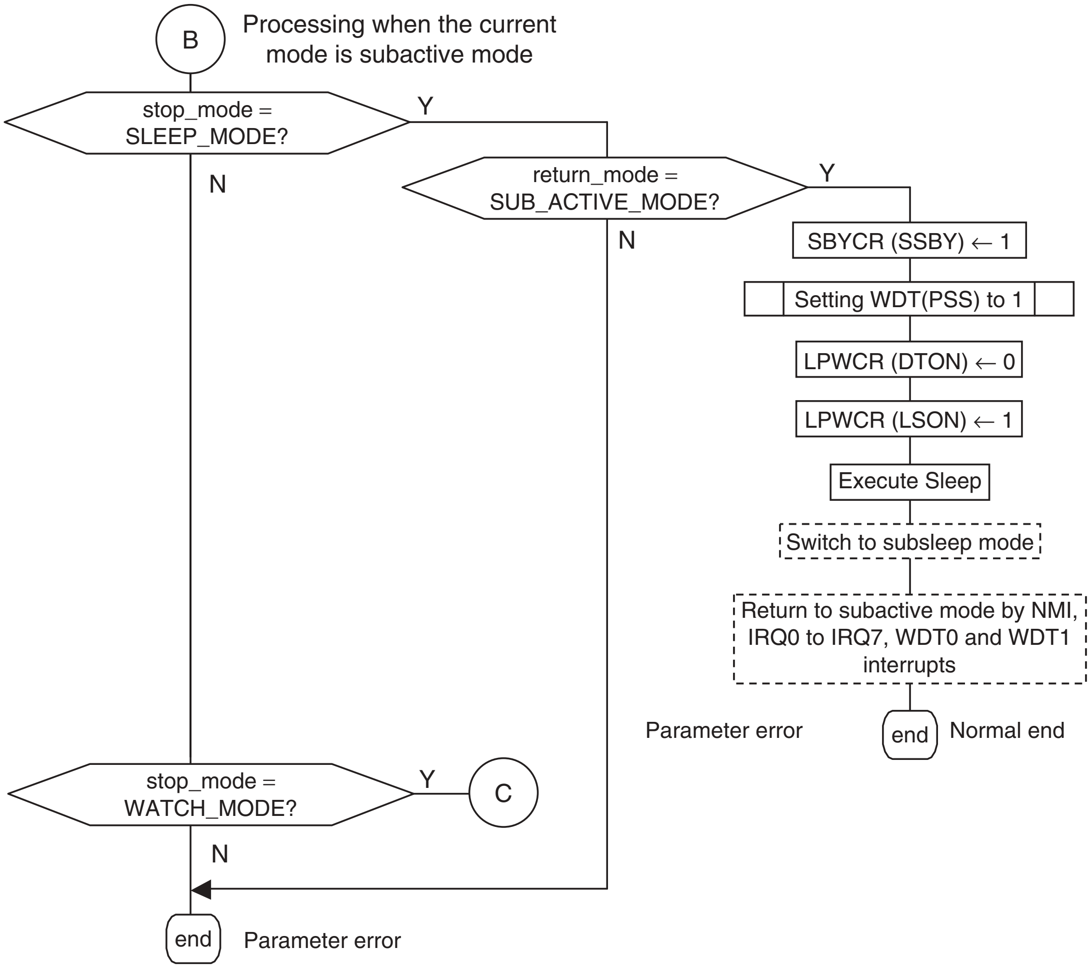
int com_active_mode_change (int return_mode)
: Directly performs the transition between modes.
return_mode : Specifies the transition operating mode.



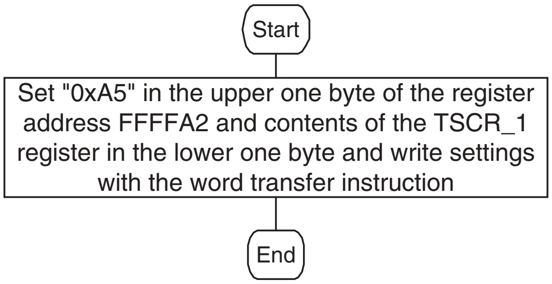
com_sleep_mode_change (int stop_mode , int return_mode)
: Switches to the program halt state and then to the specified mode.
stop_mode : Specifies the program halt state mode.
return_mode : Specifies the return destination mode.


Start
Is current mode subactive mode? (LPWCR (LSON) = 1?)
Y (subactive mode)
B
N (high-speed/medium-speed mode)
stop_mode = SLEEP_MODE?
Y
N
return_mode = SUB_ACTIVE_MODE?
Y
N
SBYCR (SSBY) ← 0
LPWCR (LSON) ← 0
Execute Sleep
Switch to sleep mode
Return to active mode each time interrupt occurs
Set specified high-speed/medium-speed mode SCKCR(SCK) ← return_mode
end
Normal end
stop_mode = STANDBY_MODE?
Y
N
return_mode = SUB_ACTIVE_MODE?
Y
N
SBYCR (SSBY) ← 1
Standby time = 131072 states SBYCR (STS[2:0]) ← 100
Setting TSCR_1(PSS) to 0
LPWCR (LSON) ← 0
Execute Sleep
Switch to sleep mode
Return to active mode by NMI, IRQ0 to IRQ7 interrupts
Set specified high-speed/medium-speed mode SCKCR(SCK) ← return_mode
end
Normal end
end
Parameter error
A


A
C
stop_mode =
WATCH_MODE?
Y
N
return_mode =
SUB_ACTIVE_MODE?
Y
N
SBYCR (SSBY) ← 1
Standby time = 131072 states
SBYCR (STS[2:0]) ← 100
Setting TSCR_1(PSS) to 1
LPWCR (DTON) ← 0
LPWCR (LSON) ← 0
Set high-speed mode
SCKCR(SCK) ← 0
Execute Sleep
Switch to sleep mode
Return to active mode by NMI, IRQ0 to IRQ7, and WDT1 interrupts
Set specified high-speed/medium-speed mode
SCKCR(SCK) ← return_mode
end
Normal end
SBYCR (SSBY) ← 1
Standby time = 131072 states
SBYCR (STS[2:0]) ← 100
Setting TSCR_1(PSS) to 1
LPWCR (DTON) ← 0
LPWCR (LSON) ← 1
Set high-speed mode
SCKCR(SCK) ← 0
Execute Sleep
Switch to sleep mode
Return to subactive mode by NMI, IRQ0 to IRQ7, and WDT1 interrupts
end
Normal end
end
Parameter error



TSCR_1(PSS) change processing : com_srt_wdt_pss(int value)
value :Write value (0 or 1)
This register requires a write procedure for preventing incorrect writing.
For details, see Section 13.6.1, in the "Hardware Manual."


Start
Set "0xA5" in the upper one byte of the register address FFFFA2 and contents of the TSCR_1 register in the lower one byte and write settings with the word transfer instruction
End

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.