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.

RENESAS

H8/300H Super Low Power Series

Differences between Normal and Advanced Modes in Programming Method on Super Low Power Microcontrollers

Introduction

The H8/38099 supports advanced mode, which can go beyond the 64-Kbyte address space, to support 128-Kbyte ROM. This application note describes the differences in programming method between advanced mode and normal mode by showing an example of converting source code for the H8/38076R (supports normal mode) into that for the H8/38099.

Target Device

H8/38099

Contents

1. Specifications

1. The source program of software that runs on the H8/38076R is modified so that it can run on the H8/38099.
2. The source program to be used is the one given in the application note for the H8/38076R, "Flashing Operation of LED Connected to I/O Port" (document number: REJ06B0550-0100).
3. This program is designed to blink the LED connected to a port of the LSI, using the periodic interrupt function of the realtime clock (RTC) module.
4. The method of configuring the High-performance Embedded Workshop (HEW) is also described.

2. Changes to be Made for Use in Advanced Mode

2.1 Address Spaces and Memory Maps

The CPU for the H8/38099 group is the H8/300H CPU, which is internally in a 32-bit configuration having an architecture upward-compatible with the H8/300 CPU. It can handle linear 16-Mbyte address space and enables realtime control. Figure 1 shows the memory maps of the H8/38099 and the H8/38076R.

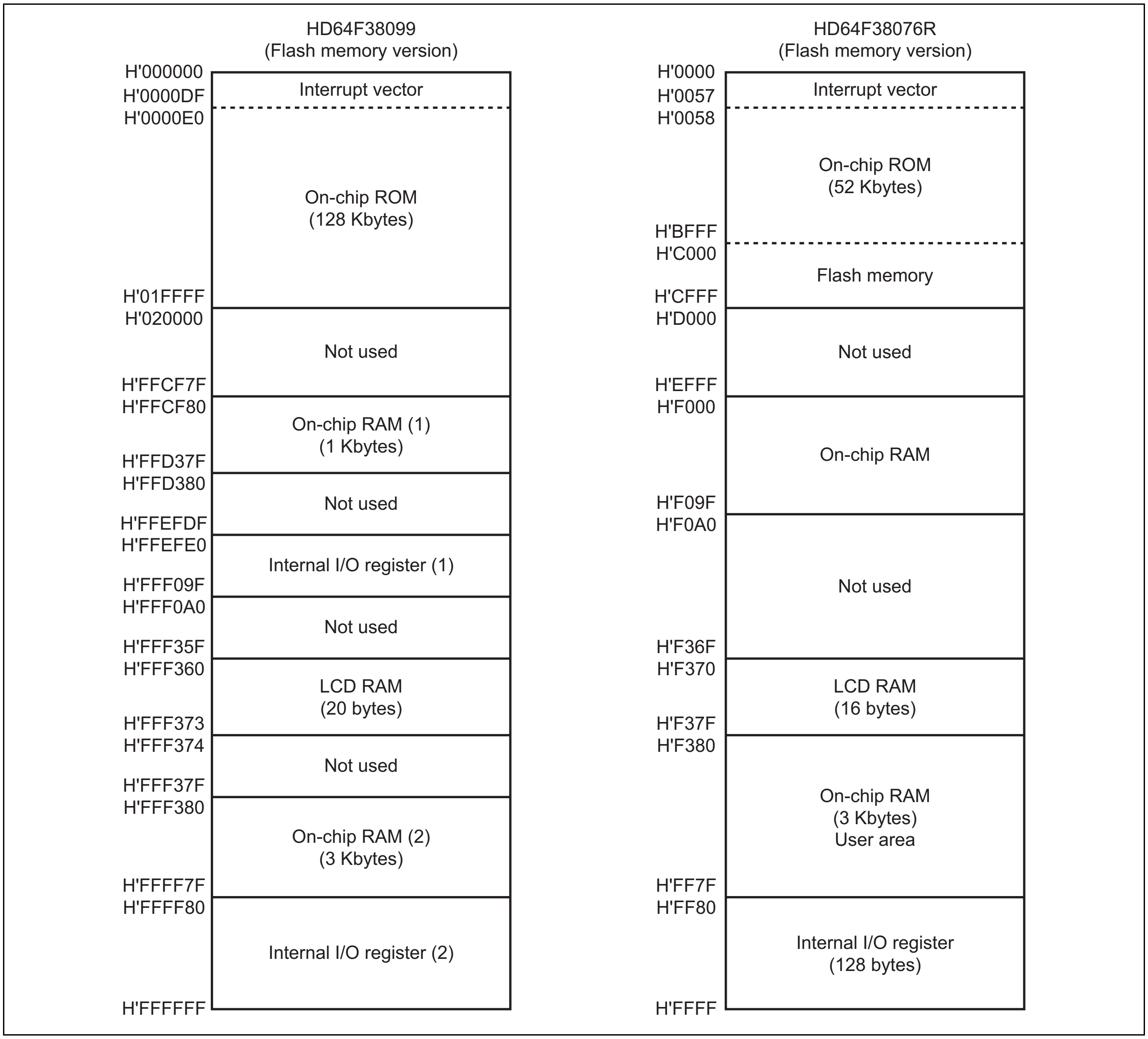


Figure 1 Memory Maps

2.2 Modifying the Source Program

This section describes how the source program should be modified.

2.2.1 Modifying Register Addresses

The registers and register addresses in the source program that are to be modified are given in Table 1.
Modify the addresses in the I/O register symbol definitions in the source program to those of the H8/38099.

Table 1 Registers and Register Addresses

Register name	Abbreviation	H8/38076R Register address	H8/38099 Register address	Line number in the source program to be modified
Port mode register 9	PMR9	H'FFC8	H'FFFFC8	34th line
Port data register 9	PDR9	H'FFDC	H'FFFFDC	37th line
Port control register 9	PCR9	H'FFEC	H'FFFFEC	40th line
RTC control register 1	RTCCR1	H'F06C	H'FFF06C	43rd line
RTC control register 2	RTCCR2	H'F06D	H'FFF06D	47th line
Clock source select register	RTCCSR	H'F06F	H'FFF06F	50th line
RTC interrupt flag register	RTCFLG	H'F067	H'FFF067	51st line
Interrupt enable register 1	IENR1	H'FFF3	H'FFFFF3	54th line

2.2.2 Modifying Vector Addresses

Since the H8/38099 has timer C, timer G, and SCI3_3 interrupt sources added, their vector addresses must be added.
Add the following source code between the 114th and the 115th lines.

```
main,                    /* H'0000B0 : No.44 : System reserve            */
main,                    /* H'0000B4 : No.45 : System reserve            */
main,                    /* H'0000B8 : No.46 : System reserve            */
main,                    /* H'0000BC : No.47 : System reserve            */
main,                    /* H'0000C0 : No.48 : System reserve            */
main,                    /* H'0000C4 : No.49 : System reserve            */
main,                    /* H'0000C8 : No.50 : System reserve            */
main,                    /* H'0000CC : No.51 : System reserve            */
main,                    /* H'0000D0 : No.52 : System reserve            */
main,                    /* H'0000D4 : No.53 : Timer C overflow/underflow */
main,                    /* H'0000D8 : No.54 : Timer G input caputure    */
main,                    /* H'0000DC : No.55 : SIC3_3                    */
```

2.2.3 Modifying the Stack Pointer

Change the stack pointer setting: Change the stack pointer address in the 116th line from H'FF80 to H'FFFF80.

2.3 Configuring the High-performance Embedded Workshop (HEW)

2.3.1 Configuring the High-performance Embedded Workshop at Its Startup

At the startup of the High-performance Embedded Workshop, the window in figure 2 appears;
select "Create New Project Workspace (C)" and click "OK".

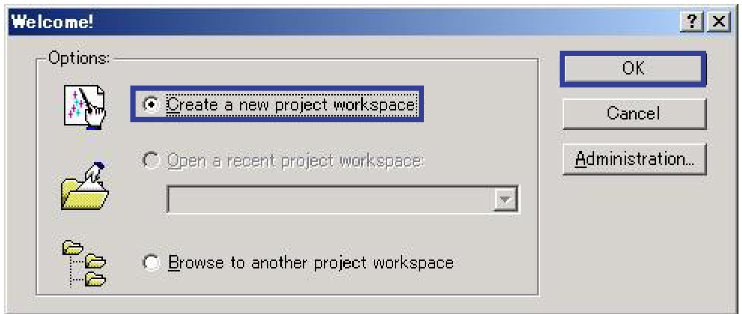


Figure 2 Dialog Displayed when Starting Up the High-performance Embedded Workshop

In the New Project Workspace window shown in figure 3,

1. Select "Empty Application".
2. In the edit box under "Workspace name (W)", enter any desired name. In this application note, "advancedmode" is entered.
3. In the edit box under "Project name (P)", enter any desired name. In this application note, "advancedmode" is entered.
4. From the pull-down menu at "CPU family (C)", select [H8S,H8/300].
5. From the pull-down menu at "Tool chain (T)", select [Hitachi H8S,H8/300 Standard].
6. Click "OK".

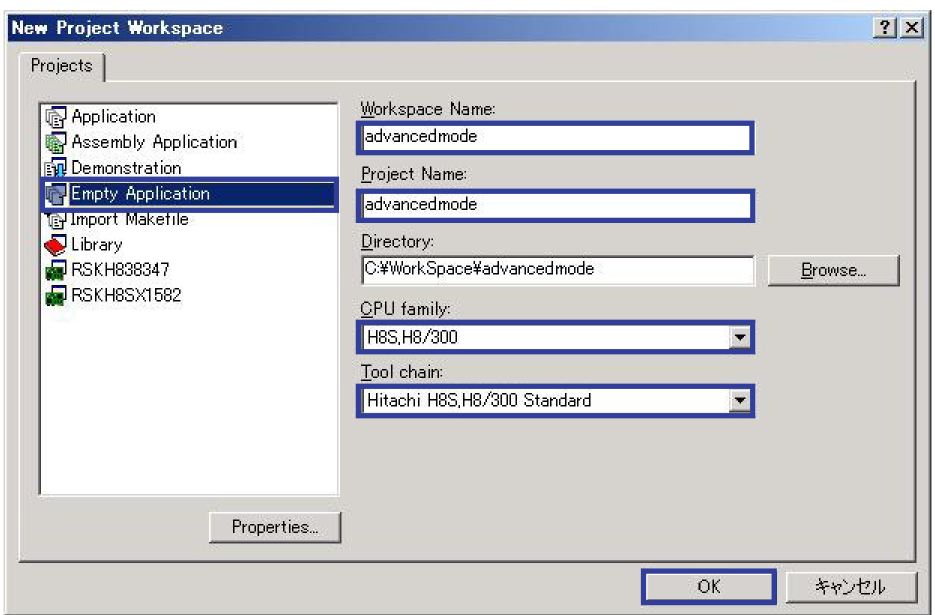


Figure 3 New Project Workspace Window

In the "New Project-1/4-CPU" window, select the tool chain version and the CPU series [300H], and click "Next (N)".

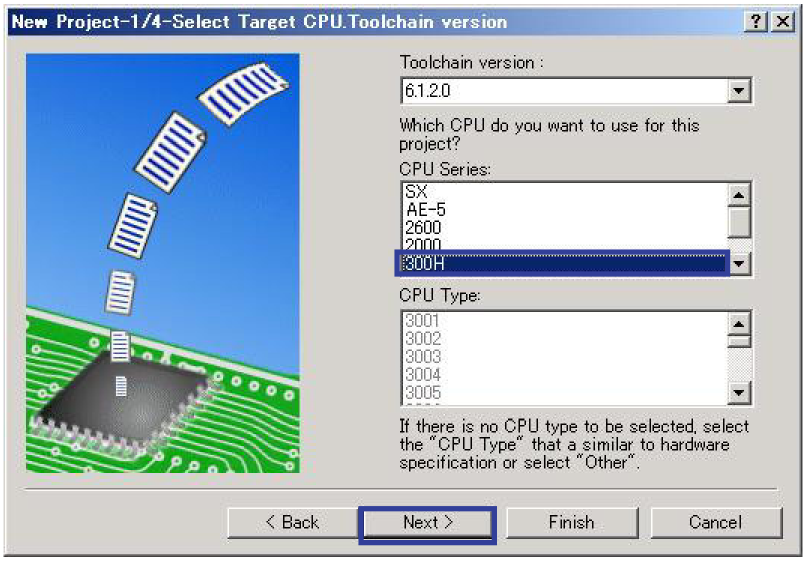


Figure 4 New Project –1/4 — Select Target CPU Toolchain Version

In the "New Project-2/4-Options" window, select [Advanced] for Operating mode and [16M byte] for Address space, and click "Next (N)".

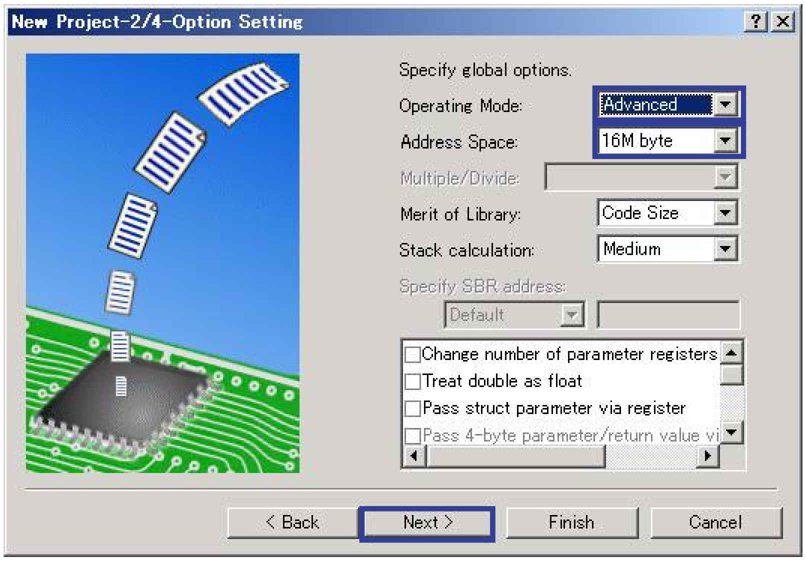


Figure 5 New Project –2/4 — Option Setting

In the "New Project-3/4-Debugger" window, place a check at [H8 Tiny/SLP E8 SYSTEM 300H] under Target, and click "Next (N)".

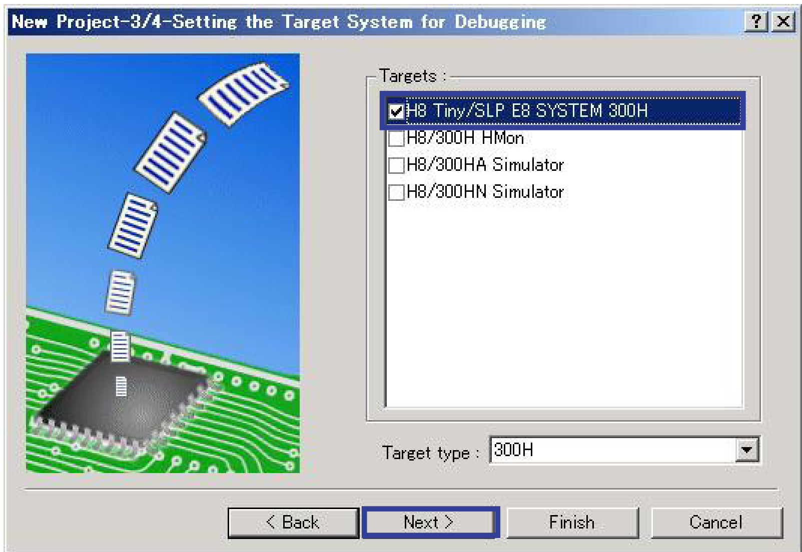


Figure 6 New Project –3/4 — Setting the Target System for Debugging

In the "New Project-4/4-Debugger Options" window, click the "Complete" button.

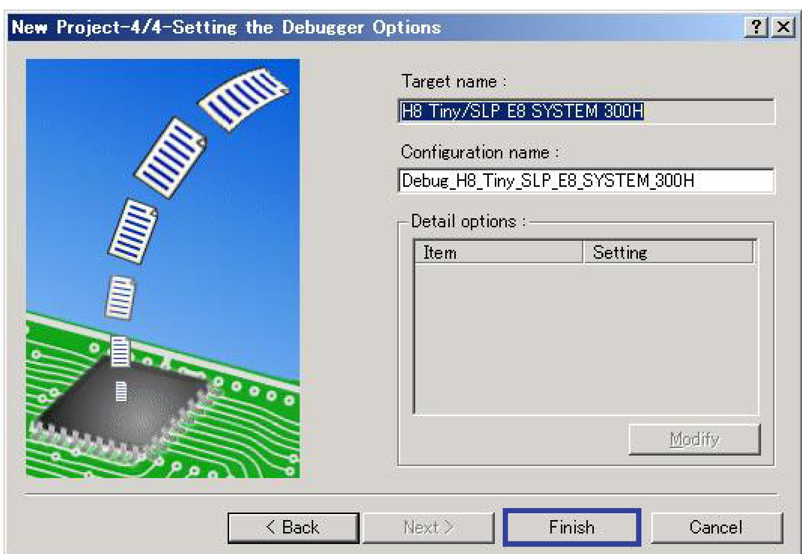


Figure 7 New Project –4/4 — Setting the Debugger Options

In the Overview window that appears, click "OK".

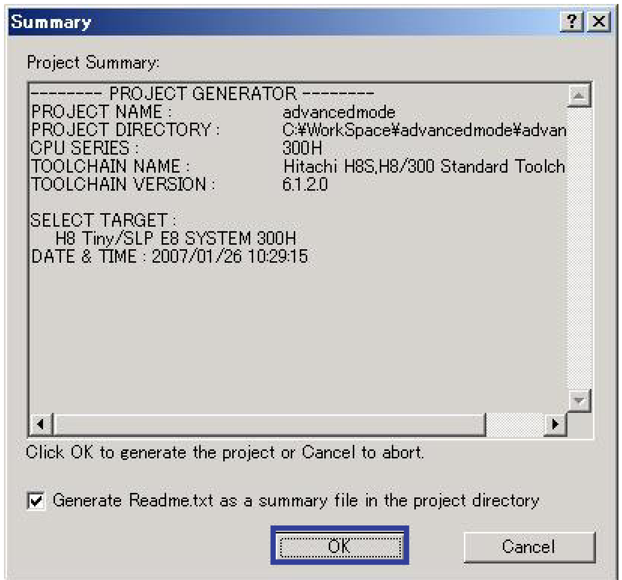


Figure 8 Summary Window

The procedure for configuring the High-performance Embedded Workshop at its startup is now completed.

2.3.2 Configuring a Tool Chain

After the High-performance Embedded Workshop has started up, add the modified source file and then configure a tool chain.

From [Build] in the menu bar, select [H8S,H8/300 Standard Toolchain].

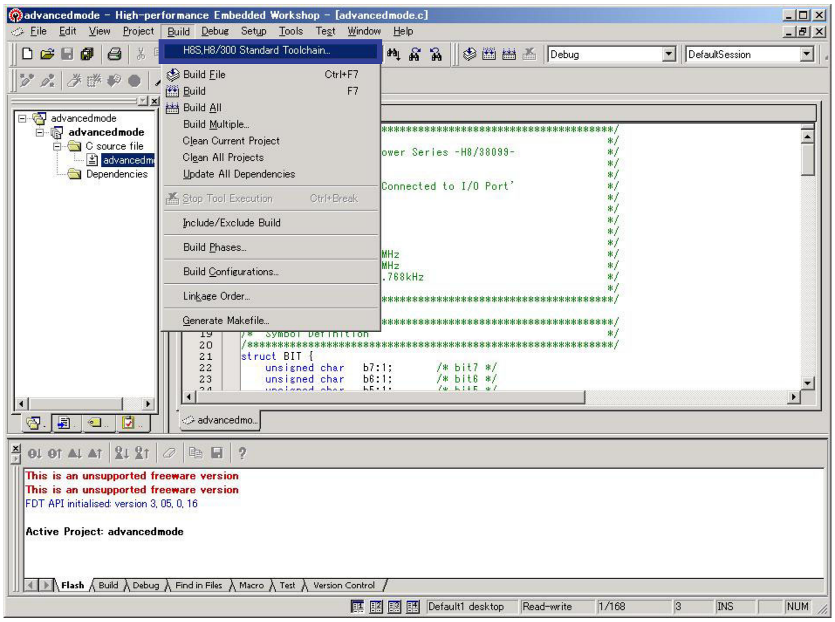


Figure 9 Displaying the Toolchain Configuration Window

Select the "Link/Library" tab, and from the [Category] pull-down menu, select [Section]. Click the "Edit…" button to open the Section Configuration window.

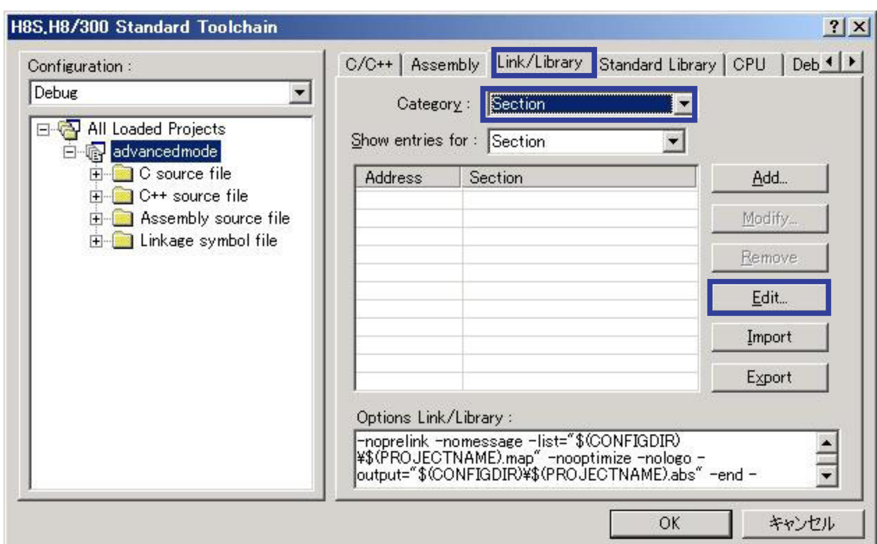


Figure 10 Link/Library Configuration Window

Click the "Add…" button, and enter addresses and section names.

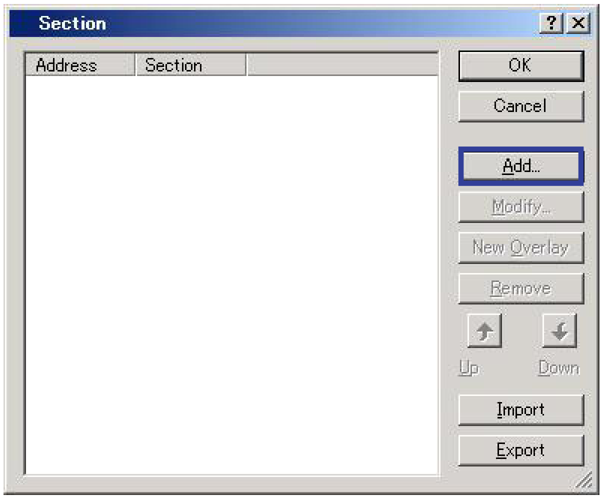


Figure 11 Section Setting Window

The addresses and section names to enter are given in Table 2.

Table 2 Addresses and Section Names

Section Name	Address
CVECT	H'000000
P	H'000800

Click the "OK" button to end configuring a tool chain, and execute a build.

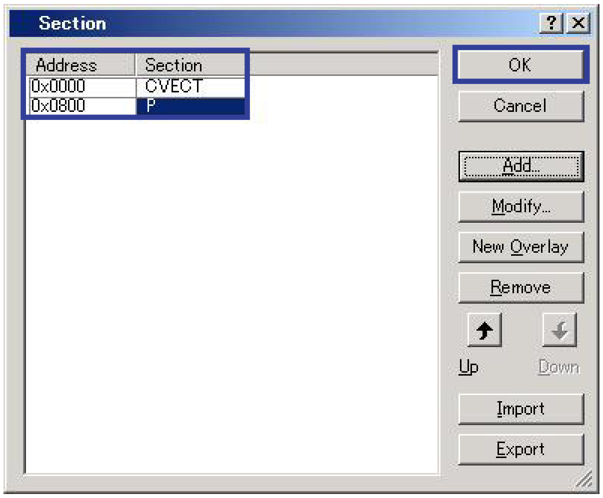


Figure 12 Display After Section Setting is Made

3. Description of Functions Used

3.1 Description of Functions

In this application note, the LED connected to an I/O port is turned on and off, using the periodic interrupt function of the RTC. The following describes the functions used. The details of the bits of each register are given in section 5.3, "Internal Registers Used".

3.1.1 Realtime Clock (RTC) Function

The realtime clock (RTC) function is described below.

- RTC control register 1 (RTCCR1)
  RTCCR1 controls the start/stop and reset of the clock timer.

- RTC control register 2 (RTCCR2)
  RTCCR2 controls 0.5-second RTC periodic interrupts. With the 0.5-second interrupt enabled, the corresponding flag of the RTC interrupt flag register (RTCFLG) is set to 1 when an interrupt is generated.

- Clock source select register (RTCCSR)
  RTCCSR selects a clock source. In this sample task, a 32.768-kHz clock source is selected for RTC operation.

- RTC interrupt flag register (RTCFLG)
  When an interrupt is generated, the corresponding bit in RTCFLG is set. Even after the interrupt is accepted, the corresponding flag will not be automatically cleared. Write 0 to clear the flag.

3.1.2 I/O Port Function

In this sample task, pin 92 of port 9 is configured as an output pin.

- Port data register 9 (PDR9)
  PDR9 is an 8-bit register that stores data for the P93 to P90 pins of port 9. When port 9 is read while PCR9 bits are set to 1, the values stored in PDR9 are read, regardless of the actual pin states.

- Port control register 9 (PCR9)
  PCR9 controls input/output of port 9 pins on a bit-by-bit basis. Setting a bit in PCR9 to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the corresponding pin an input pin. The settings in PCR9 and PDR9 registers are valid when the corresponding pin is set as a general I/O pin. This register is write-only. When it is read, each of its bits is always read as 1.

- Port mode register 9 (PMR9)
  PMR9 controls switching of the pin functions for port 9.

3.1.3 Watchdog Timer Function

The H8/38099 has an internal watchdog timer (WDT). After a reset, the WDT is turned on. The WDT is an 8-bit timer, and the inside of the H8/38099 is reset when the WDT counter overflows because of the CPU being unable to rewrite the counter value due to a system runaway or other reasons. In this sample task, the watchdog timer function is deactivated because it is not used.

- Timer control/status register WD1 (TCSRWD1)
  TCSRWD1 controls writing to TCSRWD1 itself and to TCWD. TCSRWD1 also controls the operation of the watchdog timer and indicates its operating status. To rewrite this register, use the MOV instruction. Bit manipulation instructions cannot be used to change its setting.

3.1.4 Interrupt Controller

This LSI uses the interrupt controller to control interrupts.

- Interrupt enable register 1 (IENR1)
  IENR1 enables RTC, WKP7 to WKP0, IRQ0, IRQ1, IRQ3, IRQ4, and IRQAEC interrupt requests.

3.2 Assignment of Functions

The assignment of functions in this application note is given in table 3. With the functions assigned as shown in table 3, the LED connected to an I/O port is made to blink.

Table 3 Assignment of Functions

Elements	Assignment of functions
PDR9	Stores the output data for P92.
PCR9	Sets P92 as an output pin.
PMR9	Sets P92 as an input/output pin.
RTCCR1	Controls the start/stop of RTC operation, operating mode, resets, and interrupt generation timing.
RTCCR2	Enables 0.5-second periodic interrupts.
RTCCSR	Sets the RTC clock source to 32.768 kHz.
RTCFLG	0.5-second interrupt request flag.
IENR1	Enables RTC interrupt requests.
TCSRWD1	Stops the watchdog timer.

4. Principles of Operation

The principles of operation are shown in Figure 13. Operation of blinking the LED connected to an I/O port is performed with the hardware and software processing shown in Figure 13.

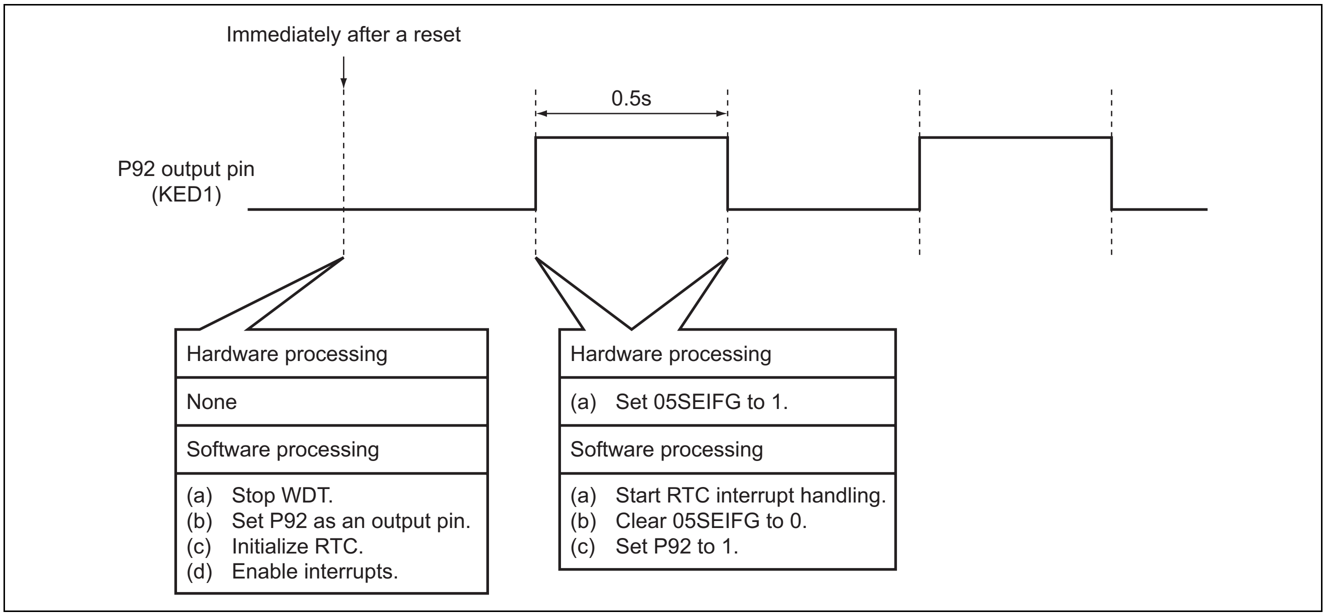


Figure 13 Principles of Operation of Blinking the LED Connected to an I/O Port

5. Description of Software

5.1 Modules

The modules mentioned in this application note are given in Table 4.

Table 4 Description of Modules

Module name	Label name	Description
Main routine	main	Initializes the RTC, sets up the RTC 0.5-second interrupt, initializes port 9, and stop the watchdog timer.
RTC interrupt routine	int_rtc	An interrupt handler for the RTC 0.5-second periodic interrupt that turns on/off the LED on P92.

5.2 Arguments

No arguments are used in this application note.

5.3 Internal Registers Used

The following tables describe the internal registers used in this application note.

- Port Data Register 9 (PDR9) Address: H'FFFFDC

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
2	P92	1	0/1	R/W	When port 9 is read when the corresponding bit of the PCR9 register is 1, the value of PDR9 is directly read. The pin state therefore has no effect on reading. When port 9 is read when PCR9 is 0, the pin state is read.

- Port Control Register 9 (PCR9) Address: H'FFFFEC

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
2	PCR92	0	1	R/W	Setting a PCR9 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR9 and PDR9 registers are valid when the corresponding pin is set as a general I/O pin. This register is write-only. When it is read, each of its bits is always read as 1.

- Port Mode Register 9 (PMR9) Address: H'FFFFC8

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
2	IRQ4	0	0	R/W	P92/IRQ4 Pin Switching Specifies whether to use the P92/IRQ4 pin as the P92 pin or as the IRQ4 pin. 0: The pin functions as the P92 input/output pin. 1: The pin functions as the IRQ4 input pin.

- RTC Control Register 1 (RTCCR1) Address: H'FFF06C

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
7	RUN	—/(0)*	0/1	R/W	RTC Operation Start 0: The RTC stops operation. 1: The RTC starts operation.
4	RST	0	0/1	R/W	Reset 0: Normal operation 1: All registers and the control circuit of the RCT are reset except for RTCCSR and this bit. Note that after setting this bit to 1, be sure to clear it to 0.

Note: * Initial value after the RTC is reset with the RST bit of RTCCR1.

- RTC Control Register 2 (RTCCR2) Address: H'FFF06D

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
1	05SEIE	—/(0)*	1	R/W	0.5-second Periodic Interrupt Enable 0: 0.5-second periodic interrupts are disabled. 1: 0.5-second periodic interrupts are enabled.

Note: * Initial value after the RTC is reset with the RST bit of RTCCR1.

- RTC Clock Source Select Register (RTCCSR) Address: H'FFF06F

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
7	—	—/(0)*	0	R	Reserved This bit cannot be modified.
6 5 4	RCS6 RCS5 SUB32K	0 0 0	0 0 0	R/W R/W R/W	Clock Output Select Selects the clock to be output from the TMOW pin when TMOW of PMR3 is set to 1. 000: ϕ/4 010: ϕ/8 100: ϕ/16 110: ϕ/32 xx1: ϕ_W
3 2 1 0	RCS3 RCS2 RCS1 RCS0	1 0 0 0	1 0 0 0	R/W R/W R/W R/W	Clock Source Select 0000: ϕ/8 (Free-running counter operation) 0001: ϕ/32 (Free-running counter operation) 0010: ϕ/128 (Free-running counter operation) 0011: ϕ/256 (Free-running counter operation) 0100: ϕ/512 (Free-running counter operation) 0101: ϕ/2048 (Free-running counter operation) 0110: ϕ/4096 (Free-running counter operation) 0111: ϕ/8192 (Free-running counter operation) 1000: 32.768 kHz (RTC operation) 1001 to 1111: Setting prohibited

Legend:
x: Don't care
Note: * Initial value after the RTC is reset with the RST bit of RTCCR1.

- RTC Interrupt Flag Register (RTCFLG) Address: H'FFF067

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
1	05SEIFG	—/(0)*2	0/1	R/W*1	0.5-second Periodic Interrupt Enable [Setting condition] • A 0.5-second periodic interrupt is generated. [Clearing condition] • 0 is written to SEIFG when SEIFG is 1.

Notes: *1. Only 0 can be written to clear the flag.
*2. Initial value after the RTC is reset with the RST bit of RTCCR1.

- Interrupt Enable Register 1 (IENR1) Address: H'FFFFF3

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
7	IENRTC	0	1	R/W	RTC Interrupt Request Enable Setting this bit to 1 enables RTC interrupt requests.

- Timer Control/Status Register WD1 (TCSRWD1) Address: H'FFFFB1

Bit	Bit Name	Initial Value	Setting Value	R/W	Description
7	B6WI	1	1	R/W	Bit 6 Write Disable Writing to bit 6 of this register is enabled only when 0 is written to this bit. This bit is always read as 1.
6	TCWE	0	0	R/W	Timer Counter W Write Enable Writing to TCWD is enabled when this bit is set to 1. When writing to this bit, 0 must be written to bit 7.
5	B4WI	1	*	R/W	Bit 4 Write Disable Writing to bit 4 of this register is enabled only when 0 is written to this bit. This bit is always read as 1.
4	TCSRWE	0	*	R/W	Timer Control/Status Register W Write Enable Writing to bits 2 and 0 of this register is enabled when this bit is set to 1. When writing to this bit, 0 must be written to bit 5.
3	B2WI	1	*	R/W	Bit 2 Write Disable Writing to bit 2 of this register is enabled only when 0 is written to this bit. This bit is always read as 1.
2	WDON	1	*	R/W	Watchdog Timer On Setting this bit to 1 causes TCWD to start counting up. Clearing it to 0 causes TCWD to stop counting up. [Clearing condition] • 0 is written to B2WI and WDON while TCSRWE is 1. [Setting conditions] • A reset is made. • 0 is written to B2WI and 1 is written to WDON while TCSRWE is 1.
1	B0WI	1	1	R/W	Bit 0 Write Disable Writing to bit 0 of this register is enabled only when 0 is written to this bit. This bit is always read as 1.
0	WRST	0	0	R/W	Watchdog Timer Reset [Clearing conditions] • A reset is made with the $\overline{RES}$ pin. • 0 is written to B0WI and WRST while TCSRWE is 1. [Setting condition] • TCWD overflows and an internal reset signal is generated.

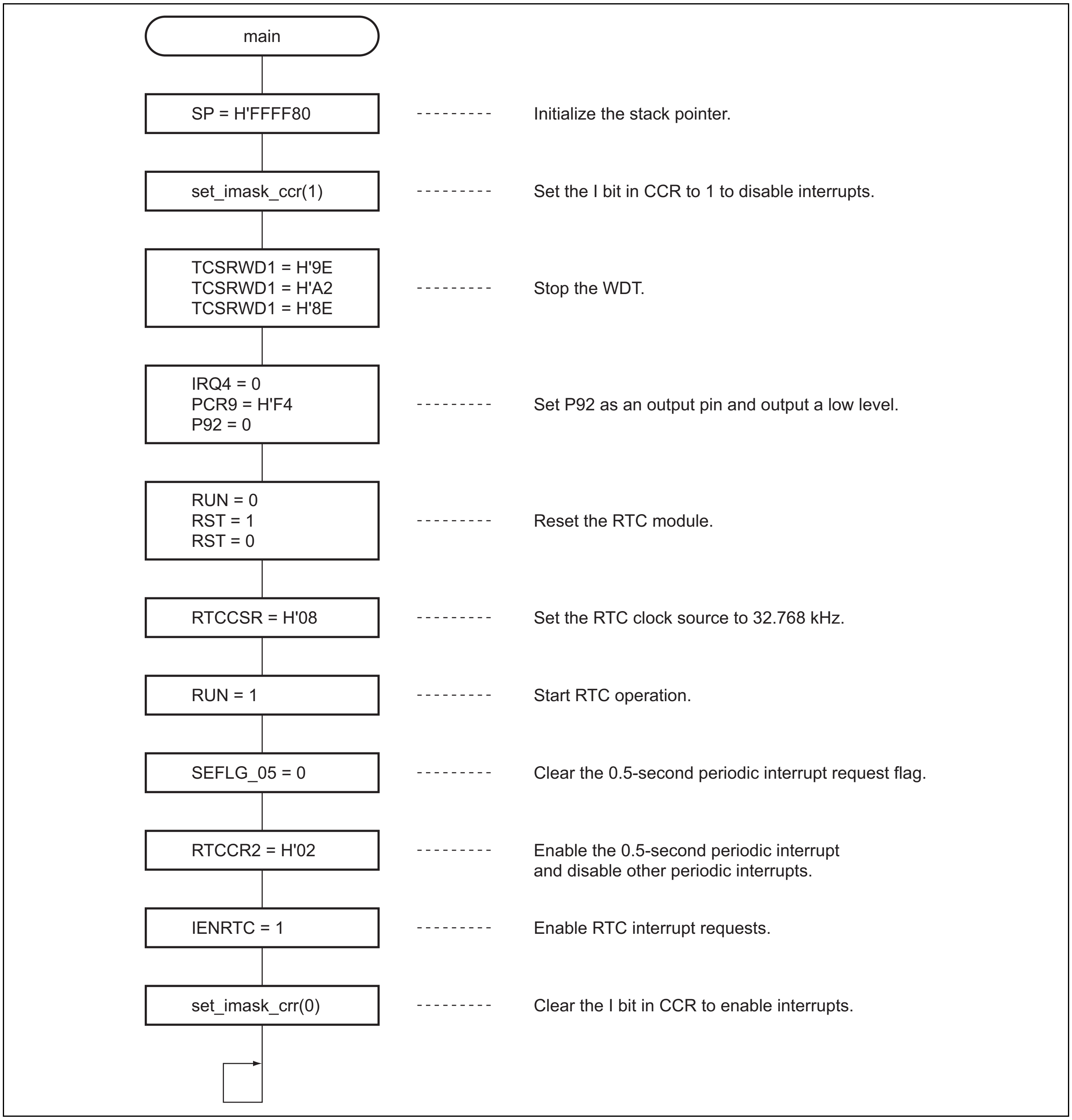
Note: * These bits are manipulated so as to stop the watchdog timer. See the flowchart for the main routine.

5.4 RAM Usage

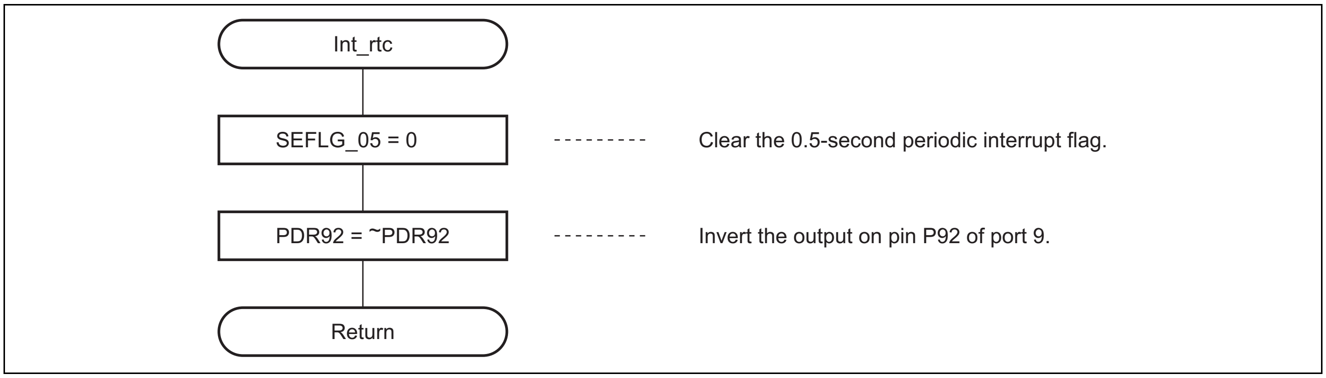
RAM is not used in this application.

6. Flowchart

6.1 main Function



6.2 int_rtc Function



7. Link Address Specifications

Section Name	Address
CVECT	H'000000
P	H'000800

Website and Support

Renesas Technology Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/inquiry
csc@renesas.com

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Mar.15.07	—	First edition issued