SH7786 Group

SH7786 Initial Settings Sample Program

R01AN0519EJ0101
Rev.1.01
Jul 15, 2011

Introduction

This application note presents a sample program for setting the initialization items required at SH7786 startup.

Target Device

SH7786

Contents

1. Introduction

1.1 Specifications

After the power-on reset is cleared, the code presented in this application note initializes the local bus state controller (LBSC) and the DDR3-SDRAM interface (DBSC3).

1.2 Functions Used

- Local bus state controller (LBSC)
- DDR3-SDRAM interface (DBSC3)

1.3 Applicable Conditions

Evaluation board	AP-AH4AD-0A (Alpha Project)*[1]	
	CPU	SH7786
	Operating frequencies	Internal clock: 533 MHz SuperHyway clock: 267 MHz Peripheral clock: 44 MHz DDR3 clock: 533 MHz External bus clock: 89 MHz
	Clock operating mode	Clock mode 3 (MD0 = high, MD1 = high, MD2 = low, MD3 = low)
	Endian mode	Little endian (MD8 = high)
	Addressing mode	29-bit addressing mode (MD10 = low)
	Area 0 bus width	16 bits (MD4 = low, MD5 = High, MD6 = low)
	Memory	NOR flash memory, 16 MB (area 0): Spansion S29GL128P90TFIRI DDR3-SDRAM, 256 MB (areas 2 to 5): Micron MT41J64M16LA-187E (2 chips)
Toolchain	Super-H RISC engine	Standard Toolchain Version 9.3.2.0
	Compiler options	-cpu=sh4a -endian=little -include="$(PROJDIR)\inc\drv","$(PROJDIR)\inc" -object="$(CONFIGDIR)\$(FILELEAF).obj" -debug -gbr=auto -chgincpath -errorpath -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1 –nologo
	Assembler options	-cpu=sh4a -endian=little -round=zero -denormalize=off -include="$(PROJDIR)\inc" -include="$(PROJDIR)\inc\drv" -debug -object="$(CONFIGDIR)\$(FILELEAF).obj" -literal=pool,branch,jump,return -nolist -nologo -chgincpath -errorpath
	Linker options	-noprelink -rom=D=R -nomessage -list= "$(CONFIGDIR)\$(PROJECTNAME).map" -nooptimize -start=INTHandler,VECTTBL,INTTBL,IntPRG/0800, PResetPRG/01000,P,C,C$BSEC,C$DSEC,D/02000, RSTHandler/0A0000000,B,R/0ADF00000, S/0ADFF0000 -nologo

Note: 1. For detailed information on using the AP-SH4AD-0A, refer to *AP-SH4AD-0A Hardware Manual*.

Table 1 lists the section allocations used in this sample program.

Table 1 Section Allocations

Section	Section Usage	Area	Allocation Address (Virtual Address)	
INTHandler	Exception/interrupt handler	ROM	0x00000800	P0 area (Can be cached, MMU address conversion not possible)
VECTTBL	Reset vector table Interrupt vector table	ROM		
INTTBL	Interrupt mask table	ROM		
IntPRG	Interrupt function	ROM		
PResetPRG	Reset program	ROM	0x00001000	
P	Program area	ROM	0x00002000	
C	Constant area	ROM		
C$BSEC	Uninitialized data area address structure	ROM		
C$DSEC	Initialized data area address structure	ROM		
D	Initialized data	ROM		
RSTHandler	Reset handler	ROM	0xA0000000	P2 area (Can not be cached, MMU address conversion not possible)
B	Uninitialized data area	RAM	0xADF00000	
R	Initialized data area	RAM		
S	Stack area	RAM	0xADFF0000	

2. Local Bus State Controller (LBSC)

The functions of the local bus state controller (LBSC) include dividing the external memory space (according to chip select signal settings) and outputting the control signals appropriate for the various types of memory or other devices connected.

2.1 Register Configuration

Table 2 lists the local bus state controller (LBSC) register configuration.

Table 2 LBSC Register Configuration

Register	Symbol	R/W	P4 Address	Access Size	Clock
Memory address map select register	MMSELR	R/W	H'FC40 0020	32	SHck
Bus control register	BCR	R/W	H'FF80 1000	32	Bck
CS0 bus control register	CS0BCR	R/W	H'FF80 2000	32	Bck
CS1 bus control register	CS1BCR	R/W	H'FF80 2010	32	Bck
CS2 bus control register	CS2BCR	R/W	H'FF80 2020	32	Bck
CS3 bus control register	CS3BCR	R/W	H'FF80 2030	32	Bck
CS4 bus control register	CS4BCR	R/W	H'FF80 2040	32	Bck
CS5 bus control register	CS5BCR	R/W	H'FF80 2050	32	Bck
CS6 bus control register	CS6BCR	R/W	H'FF80 2060	32	Bck
CS0 wait control register	CS0WCR	R/W	H'FF80 2008	32	Bck
CS1 wait control register	CS1WCR	R/W	H'FF80 2018	32	Bck
CS2 wait control register	CS2WCR	R/W	H'FF80 2028	32	Bck
CS3 wait control register	CS3WCR	R/W	H'FF80 2038	32	Bck
CS4 wait control register	CS4WCR	R/W	H'FF80 2048	32	Bck
CS5 wait control register	CS5WCR	R/W	H'FF80 2058	32	Bck
CS6 wait control register	CS6WCR	R/W	H'FF80 2068	32	Bck
CS5 PCMCIA control register	CS5PCR	R/W	H'FF80 2070	32	Bck
CS6 PCMCIA control register	CS6PCR	R/W	H'FF80 2080	32	Bck

2.2 The CSnBCR Registers

The CSnBCR registers set the idle cycles, bus widths, and memory types.

Figure 1 shows the timing chart for idle cycles in the same space and figure 2 shows the timing chart for idle cycles in different spaces. Table 3 lists the timing symbols and their descriptions.

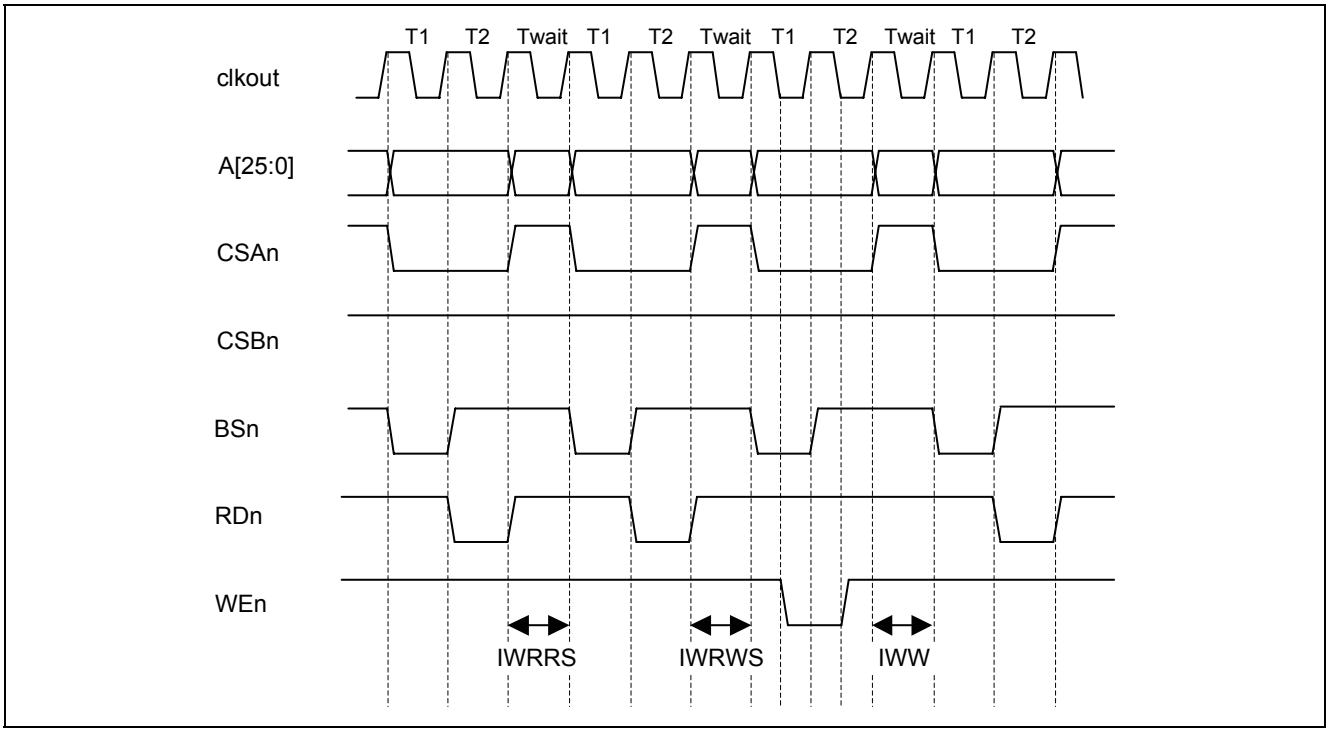

Figure 1 Idle Cycles in the Same Space

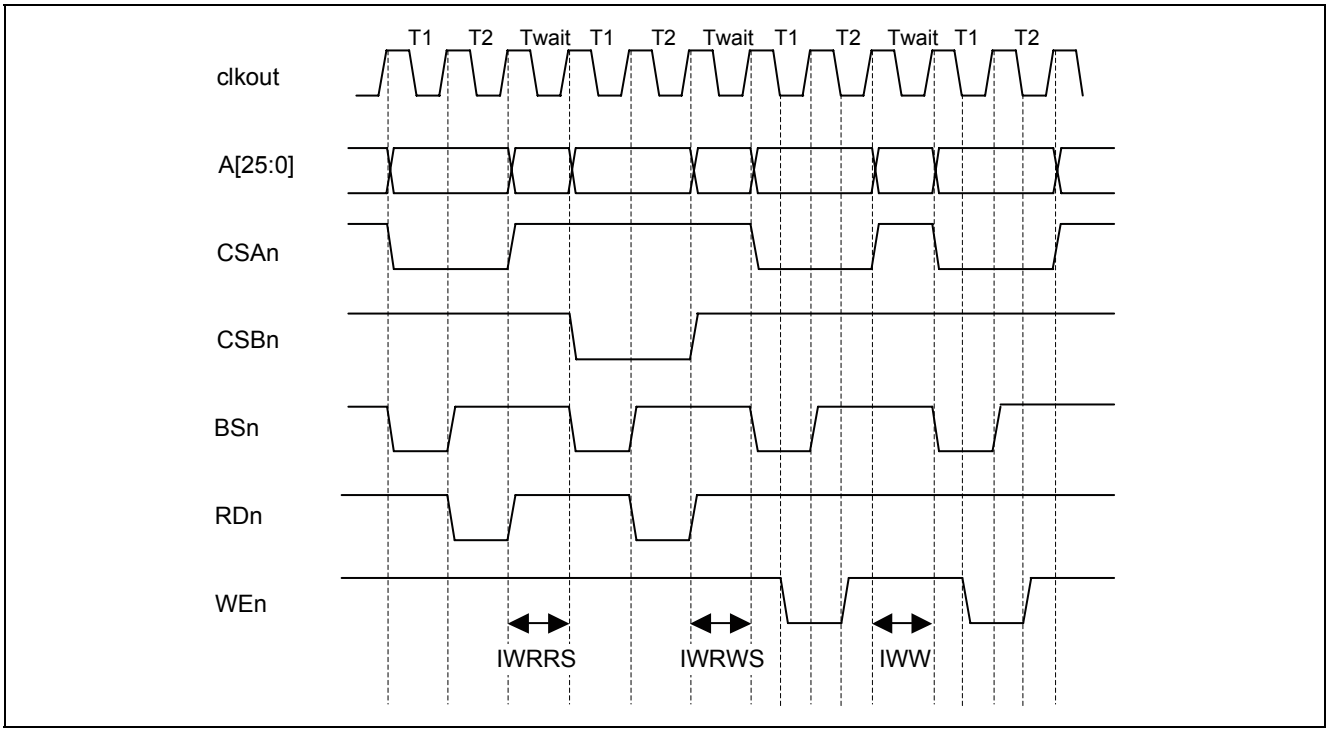

Figure 2 Idle Cycles in Different Spaces

Table 3 Idle Cycle Symbols

Symbol	Description
IWRRS	Idle cycles between read cycles in the same space
IWRWS	Idle cycles between read and write cycles in the same space
IWRRD	Idle cycles between read cycles in different spaces
IWRWD	Idle cycles between read and write cycles in different spaces
IWW	Idle cycles between write and read cycles Idle cycles between write cycles

2.3 The CSnWCR Registers

The CSnWCR registers set the setup and hold times and wait cycles for access cycles.

The data output timing during writes differs with the ADS setting.

Figure 3 shows the access cycle timing chart.

Table 4 lists the access cycle symbols and their descriptions.

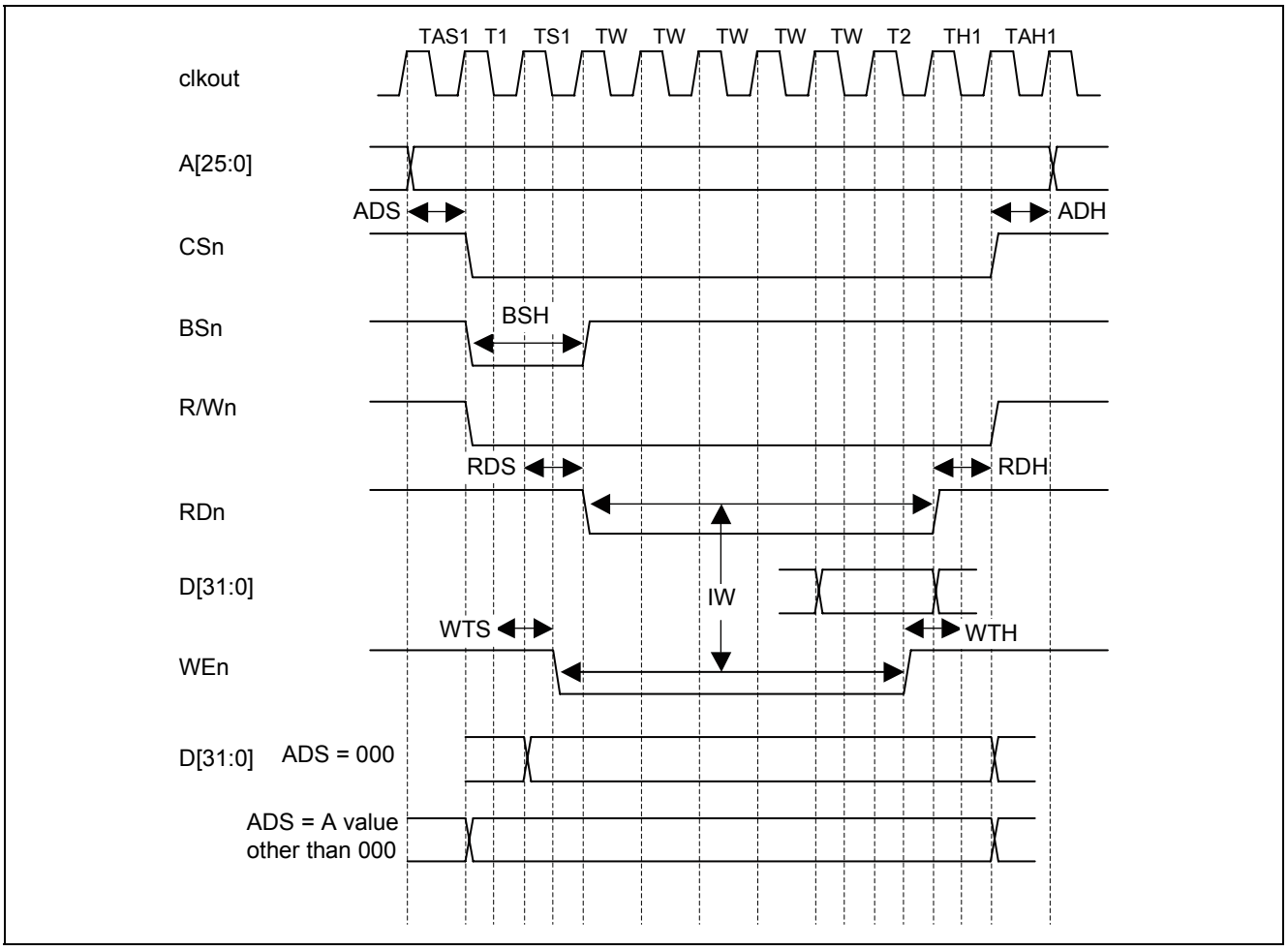

Figure 3 Access Cycle Timing Chart

Table 4 Access Cycle Symbols

Symbol	Description
ADS	Address assert to CS assert delay period
ADH	CS negate to address assert delay period
BSH	BS cycle count
RDS	CS assert to read assert delay period
RDH	Read negate to CS negate delay period
WTS	CS assert to write assert delay period
WTH	Write negate to CS negate delay period
IW	Number of wait cycles in the read and write periods

3. DDR3-SDRAM Interface (DBSC3)

The DDR3-SDRAM interface (DBSC3) arbitrates accesses from the CPU and various modules and outputs control signal required by DDR3-SDRAM allowing DDR3-SDRAM to be connected directly.

- DBSC3 supports a 32-bit bus width
- DBSC3 supports DDR3-1066 (533 MHz) devices
- DBSC3 supports a burst length of 8 and sequential mode

Table 5 lists the memory capacities and configurations that can be used.

Table 5 DBSC3 Memory Capacities and Configurations

External Bus width	Memory Capacity	Configuration	Total Capacity
32 bits	512 Mbits (32 M × 16 bits)	Parallel connection (2 chips)	128 Mbits
	512 Mbits (64 M × 8 bits)	Parallel connection (4 chips)	256 Mbits
	1 Gbit (64 M × 16 bits)	Parallel connection (2 chips)	256 Mbits
	1 Gbit (128 M × 8 bits)	Parallel connection (4 chips)	512 Mbits
	2 Gbits (128 M × 16 bits)	Parallel connection (2 chips)	512 Mbits
	2 Gbits (256 M × 8 bits)	Parallel connection (4 chips)	1 Gbit
	4 Gbits (256 M × 16 bits)	Parallel connection (2 chips)	1 Gbit
	4 Gbits (512 M × 8 bits)	Parallel connection (4 chips)	2 Gbits

3.1 Register Configuration

Table 6 lists the DBSC3 registers.

Table 6 DBSC3 Registers

Register	Symbol	R/W	P4 Address	Access Size	Clock
DBSC3 status register	DBSTATE	R	H'FFA0 000C	32	SHck
SDRAM access enable register	DBACEN	R/W	H'FFA0 0010	32	SHck
Auto refresh enable register	DBRFEN	R/W	H'FFA0 0014	32	SHck
Manual command issuing register	DBCMD	R/W	H'FFA0 0018	32	SHck
Operation completion waiting register	DBWAIT	R/W	H'FFA0 001C	32	SHck
SDRAM kind setting register	DBKIND	R/W	H'FFA0 0020	32	SHck
SDRAM configuration setting register	DBCONF	R/W	H'FFA0 0024	32	SHck
SDRAM timing register 0	DBTR0	R/W	H'FFA0 0040	32	SHck
SDRAM timing register 1	DBTR1	R/W	H'FFA0 0044	32	SHck
SDRAM timing register 2	DBTR2	R/W	H'FFA0 0048	32	SHck
SDRAM timing register 3	DBTR3	R/W	H'FFA0 0050	32	SHck
SDRAM timing register 4	DBTR4	R/W	H'FFA0 0054	32	SHck
SDRAM timing register 5	DBTR5	R/W	H'FFA0 0058	32	SHck
SDRAM timing register 6	DBTR6	R/W	H'FFA0 005C	32	SHck
SDRAM timing register 7	DBTR7	R/W	H'FFA0 0060	32	SHck
SDRAM timing register 8	DBTR8	R/W	H'FFA0 0064	32	SHck
SDRAM timing register 9	DBTR9	R/W	H'FFA0 0068	32	SHck
SDRAM timing register 10	DBTR10	R/W	H'FFA0 006C	32	SHck
SDRAM timing register 11	DBTR11	R/W	H'FFA0 0070	32	SHck
SDRAM timing register 12	DBTR12	R/W	H'FFA0 0074	32	SHck
SDRAM timing register 13	DBTR13	R/W	H'FFA0 0078	32	SHck
SDRAM timing register 14	DBTR14	R/W	H'FFA0 007C	32	SHck
SDRAM timing register 15	DBTR15	R/W	H'FFA0 0080	32	SHck
SDRAM timing register 16	DBTR16	R/W	H'FFA0 0084	32	SHck
SDRAM timing register 17	DBTR17	R/W	H'FFA0 0088	32	SHck
Refresh configuration register 0	DBRFCNF0	R/W	H'FFA0 00E0	32	SHck
Refresh configuration register 1	DBRFCNF1	R/W	H'FFA0 00E4	32	SHck
Refresh configuration register 2	DBRFCNF2	R/W	H'FFA0 00E8	32	SHck
PHY-unit control register 0	DBPDCNT0	R/W	H'FFA0 0200	32	SHck
PHY-unit control register 1	DBPDCNT1	R/W	H'FFA0 0204	32	SHck
PHY-unit control register 2	DBPDCNT2	R/W	H'FFA0 0208	32	SHck
PHY-unit control register 3	DBPDCNT3	R/W	H'FFA0 020C	32	SHck
PHY-unit lock register	DBPDLCK	R/W	H'FFA0 0280	32	SHck
PHY-unit internal register address register	DBPDRGA	R/W	H'FFA0 0290	32	SHck
PHY-unit internal register data register	DBPDRG	R/W	H'FFA0 02A0	32	SHck
Bus control unit 0 control register 0	DBBS0CNT0	R/W	H'FFA0 0300	32	SHck
Bus control unit 0 control register 1	DBBS0CNT1	R/W	H'FFA0 0304	32	SHck

See section 12.4, Register Descriptions, in section 12, DDR3-SDRAM Interface (DBSC3), in the SH7786 Group Hardware Manual (REJ09B0501) for details on these registers.

4. Application Example

4.1 Overview of the AP-SH4AD-0A

4.1.1 Device List

Table 7 lists the devices on the AP-SH4AD-0A board used in this application example.

Table 7 AP-SH4AD-0A Peripheral Devices

Item	Description
NOR FLASH ROM	SPANSION S29GL128P90TFIR20 16-bit bus access
DDR3-SDRAM	Micron MT41J64M16LA-187E 32-bit bus access
Ethernet controller	SMSC LAN9221 16-bit bus access

4.1.2 Memory Map

Table 8 lists the AP-SH4AD-0A memory map.

Table 8 AP-SH4AD-0A Memory Map

Area	Address	Connected device	Bus Width
0	H'0000_0000 to H'00FF_FFFF	S29GL128P90TFIR20 (16 MB)	16 bits
	H'0100_0000 to H'03FF_FFFF	Shadow	
1	H'0400_0000 to H'0400_0FFF	LAN9221 (512 B)	16 bits
	H'0400_1000 to H'07FF_FFFF	Shadow	
2	H'0800_0000 to H'0BFF_FFFF	MT41J64M16LA-187E (256 MB)	32 bits
3	H'0C00_0000 to H'0FFF_FFFF		
4	H'1000_0000 to H'13FF_FFFF		
5	H'1400_0000 to H'17FF_FFFF		
6	H'1800_0000 to H'17FF_FFFF	Available to users	32 bits

4.2 Sample Program Description

The following source program performs the required settings as an initialization program.

Tables, files, and other data for exceptions and interrupts may be added as required.

- vhandler.src
- resetprg.c
- dbsct.c
- vecttbl.src
- vect.inc
- intprg.c
- lowlevelinit.src
- lowlevelinit.inc

1. vhandler.src
   The exception handled in this file (vhandler.src) is run when and exception (reset, general exception, or interrupt) occurs.
   The exception handlers and jumps to the LBSC and DBSC3 initialization routines are coded in the vhandler.src file. The reset handler (_Reset_handler) is run after the power-on reset. The reset handler used in this application note differs from the one generated automatically by the High-performance Embedded Workshop in that LBSC and DBSC3 initialization processing has been added.

2. resetprg.c
   This file (resetprg.c) is based on the file generated automatically by the High-performance Embedded Workshop and includes the code for the PowerON_Reset() function registered in vecttbl.src.
   The PowerON_Reset() function is the first function that is branched to from the reset handler. It sets the VBR (vector base register), calls the _INITSCR() function, which copies sections, and calls the function that enables the cash. After that, it calls the main function. The status register (SR) selects privileged or user mode, specifies the general-purpose register bank, and manages exceptions and interrupts. This register must be set according to the design of the system.

3. dbsct.c
   This file (dbsct.c) codes copying information used to copy section data from ROM to RAM so that programs can be run from the connected external RAM. The _INITSCT() function copies programs immediately after a reset according to the information coded in this file. See the SuperH RISC Engine C/C++ Compiler Application Note (REJ05B0463) for more information.

4. vecttbl.src
   This file (vecttbl.src) codes registration for external handlers used when an exception (reset, general exception, or interrupt) occurs.

5. vect.inc
   This file (vect.inc) codes function to be registered in vecttbl.src.

6. intprg.c
   This file (intprg.c) interrupt functions used when an exception/interrupt occurs and are registered from the exception handler.

7. lowlevelinit.src
   This file (lowlevelinit.src) codes the LBSC and DBSC3 initial settings. The code in lowlevelinit.src is called from the reset handler (_Reset_handler).

8. lowlevelinit.inc
   This file (lowlevelinit.inc) codes setting values used in lowlevelinit.src.

4.3 Items Set by the Sample Program

The sample program sets the LBSC and DBSC3 so they can access the memory and peripheral devices connected to the AP-SH4AD-0A.

Table 9 lists the settings performed by the sample program.

Table 9 Sample Program Settings

Module	Settings
LBSC	CS0 space: Flash memory (memory type: SRAM), bus width: 16 bits CS1 space: Ethernet controller (memory type: SRAM), bus width: 16 bits CS2 to CS5 spaces: DDR3-SDRAM memory CS6 space: Available to users (memory type: SRAM), 32-bit bus width (as the initial value)
DBSC3	Model: MT41J64M16LA-187E Capacity (configuration): 1 Gbit (128 M × 8 bits) Devices used: 2 CAS latency: 7 Refresh cycle: 64 ms, average refresh time: 7.8125 µs Burst length: 8 Row address: A12 to A0 Column address: A11 to A0

4.4 Sample Program Flowchart

Figure 4 shows the processing flow from the power-on reset until the software jumps to the main() function.

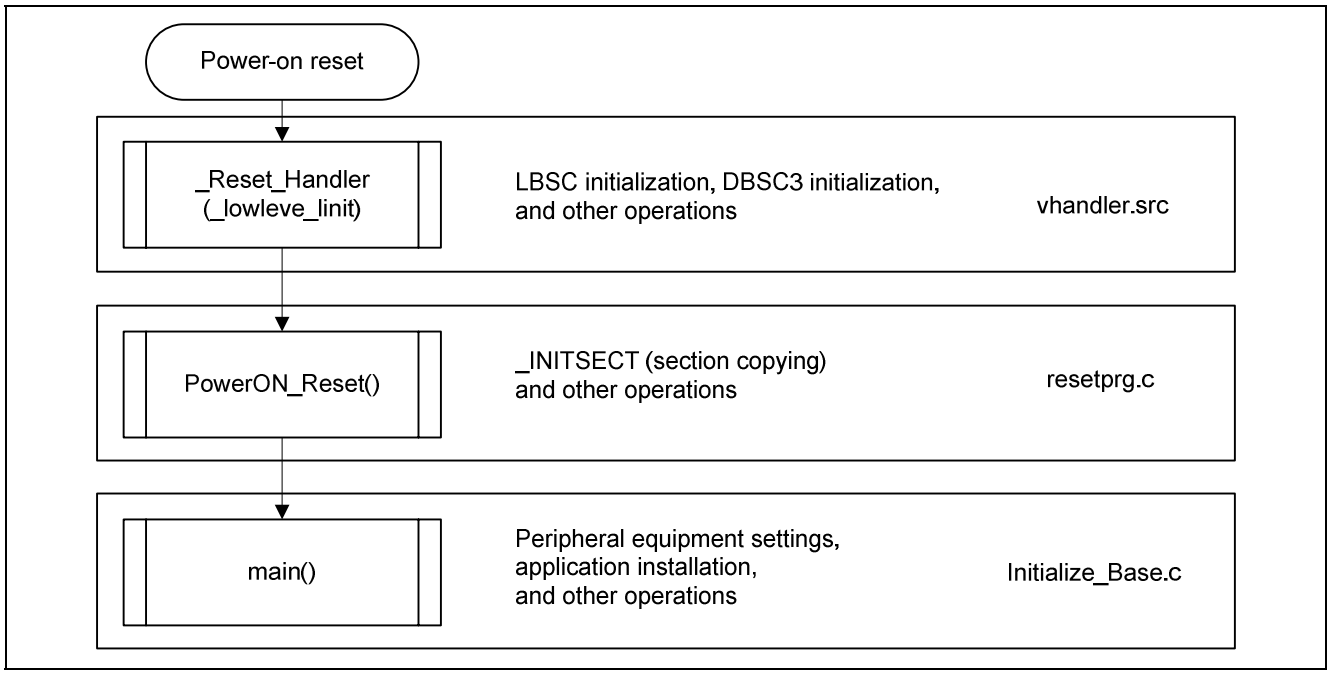

Figure 4 Flowchart from Power-on Reset to the main() Function

Figures 5 to 12 show the processing flow in the _lowlevel_init() function.

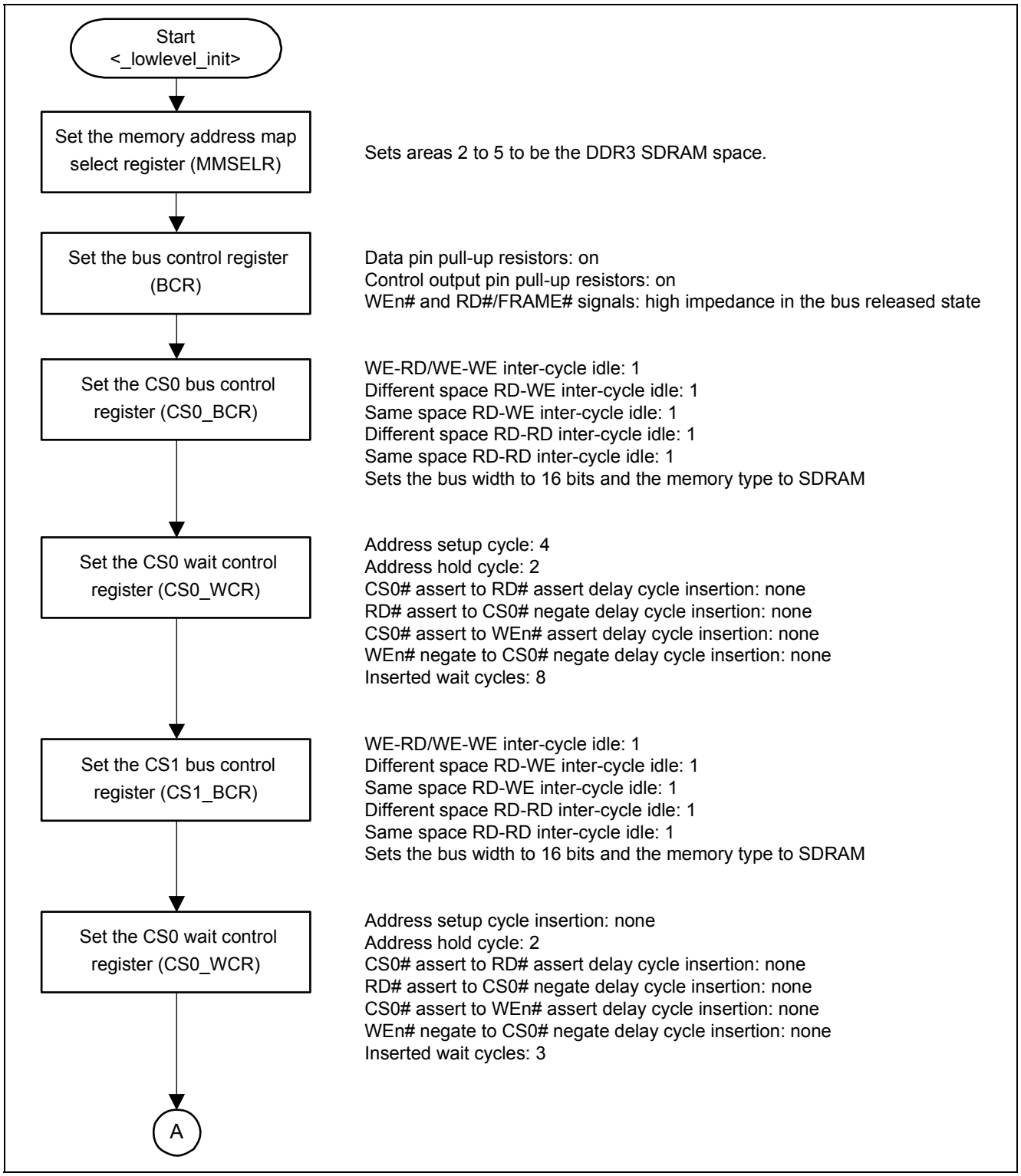

Figure 5 Flowchart for the _lowlevel_init() Function 1

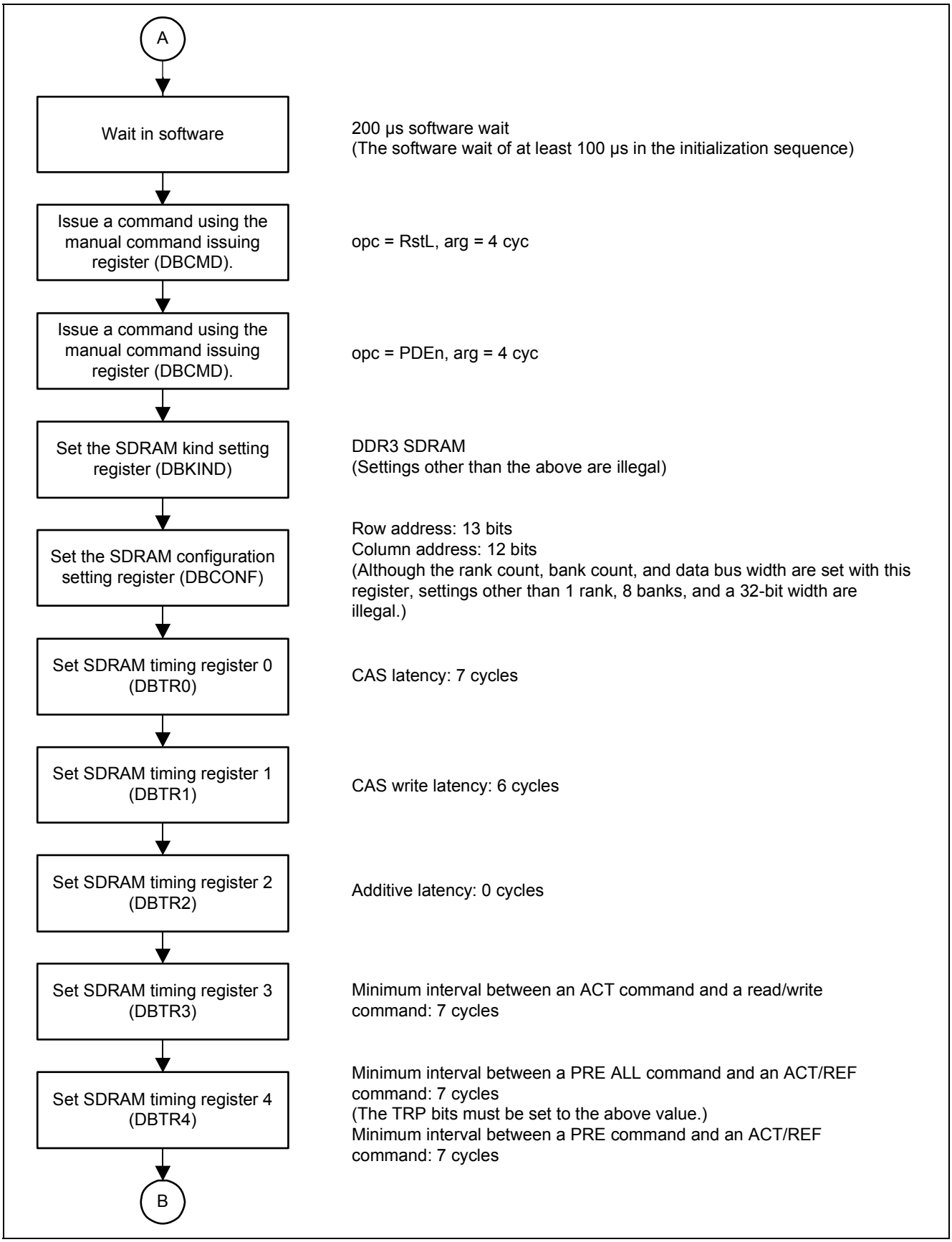

Figure 6 Flowchart for the _lowlevel_init() Function 2

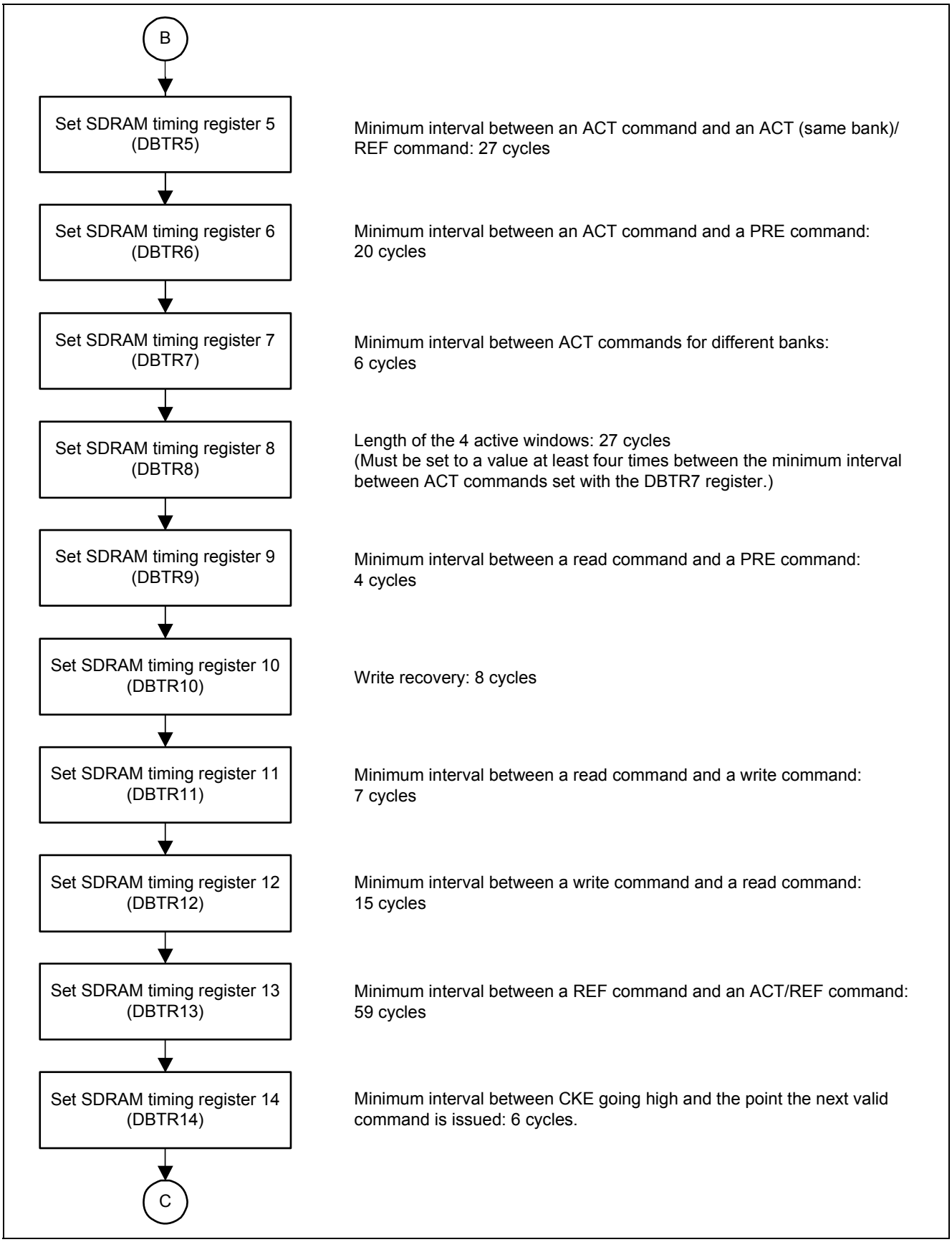

Figure 7 Flowchart for the _lowlevel_init() Function 3

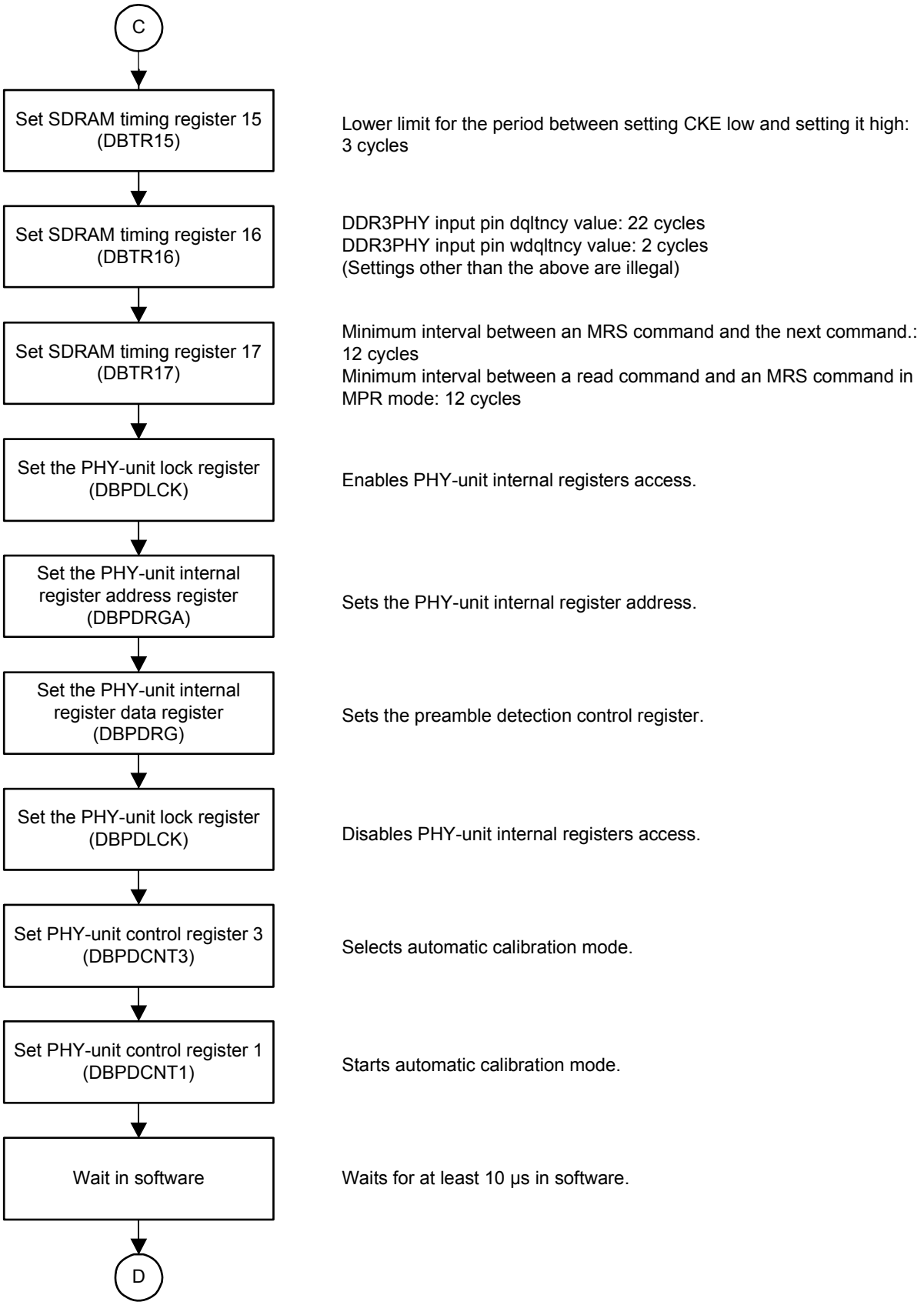

Figure 8 Flowchart for the _lowlevel_init() Function 4

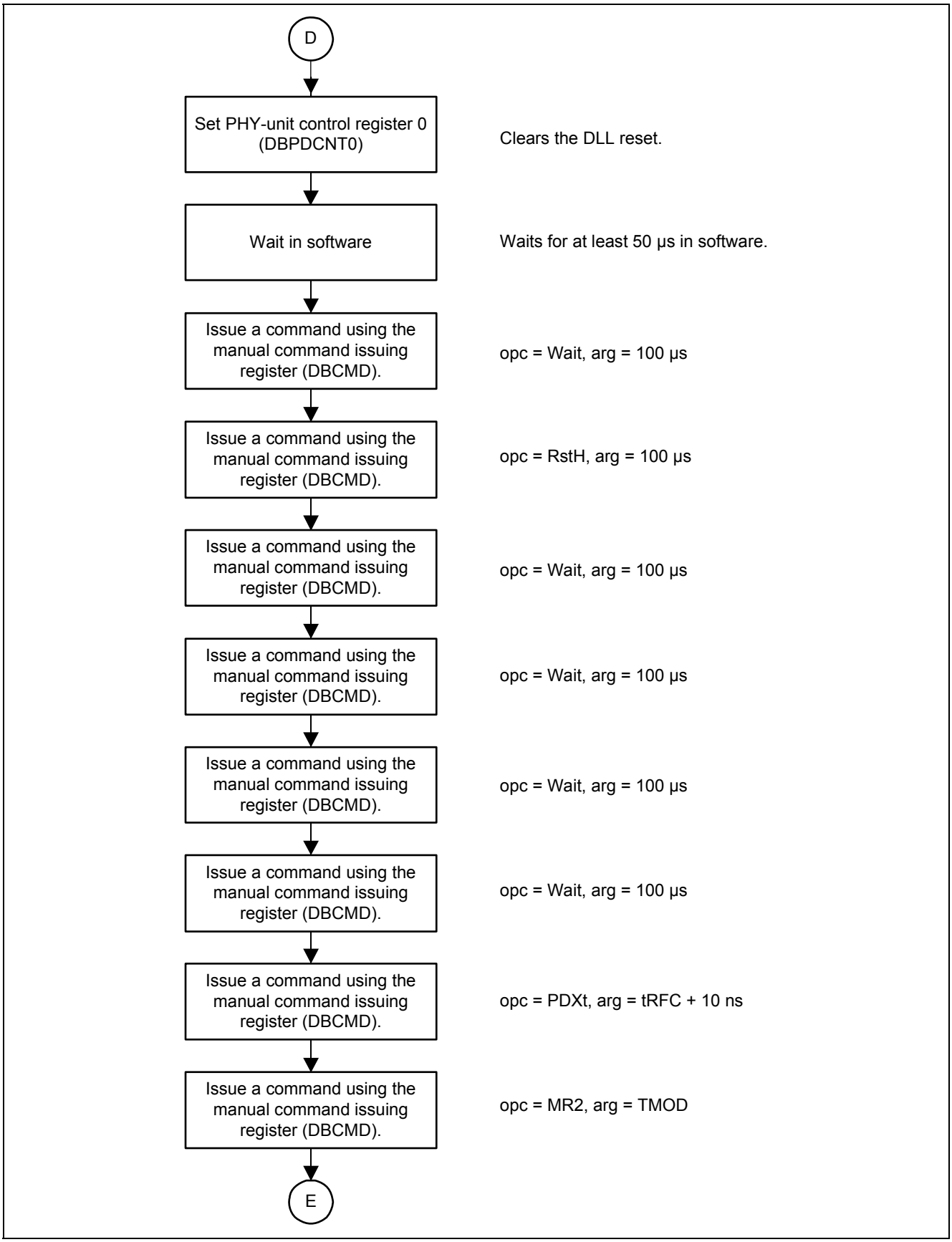

Figure 9 Flowchart for the _lowlevel_init() Function 5

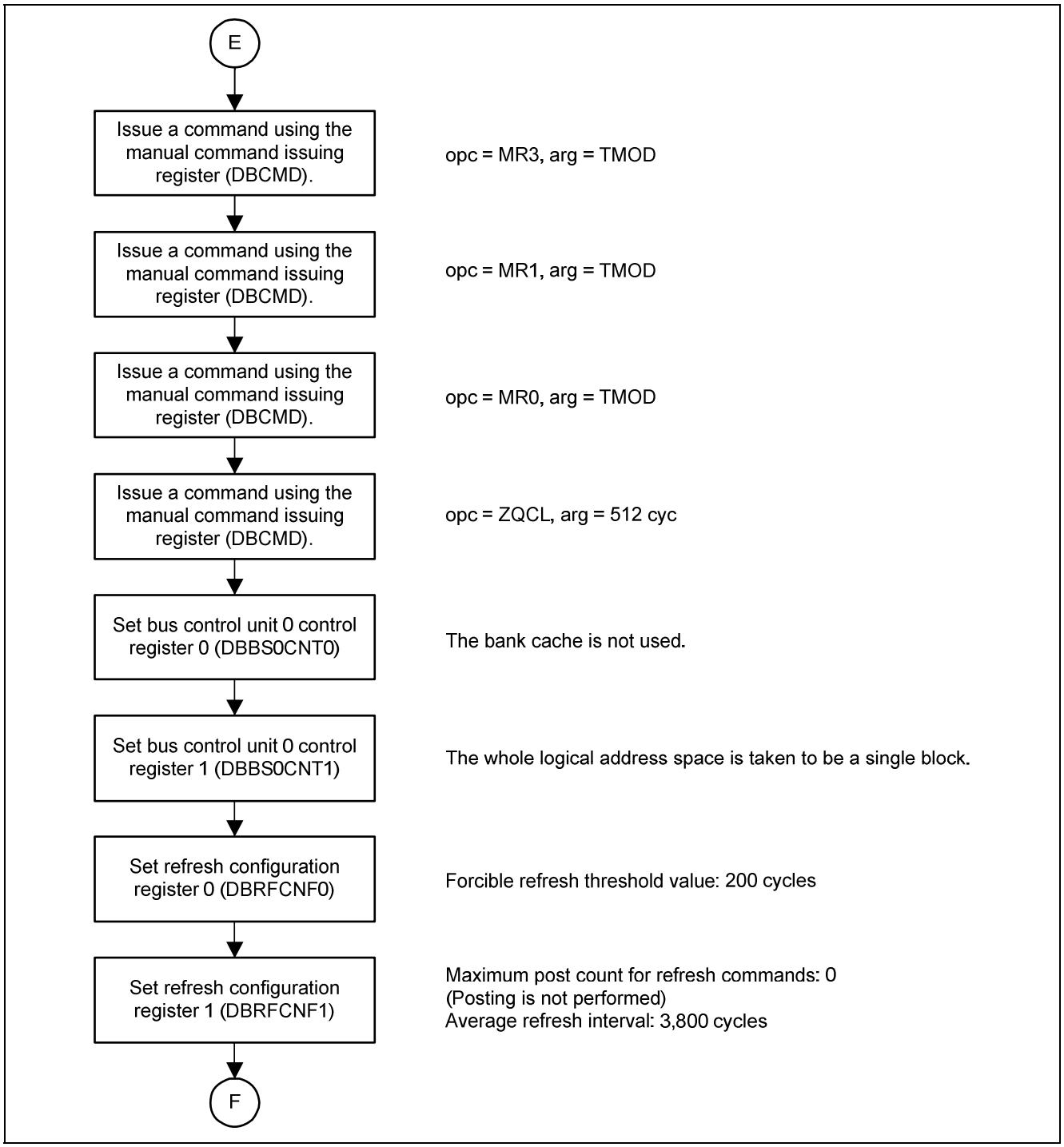

Figure 10 Flowchart for the _lowlevel_init() Function 6

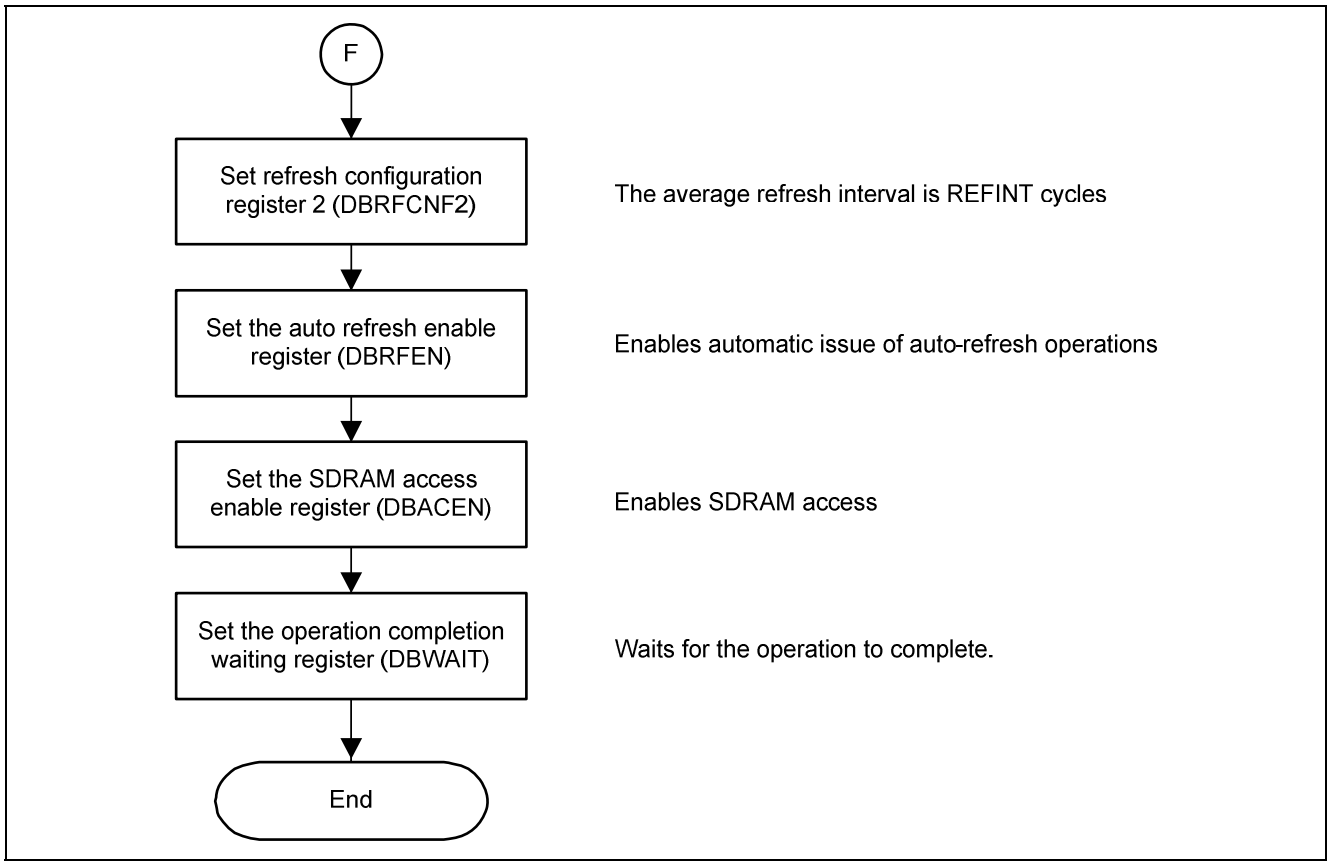

Figure 11 Flowchart for the _lowlevel_init() Function 7

Figure 12 shows the processing flow in the PowerON_Reset() function.

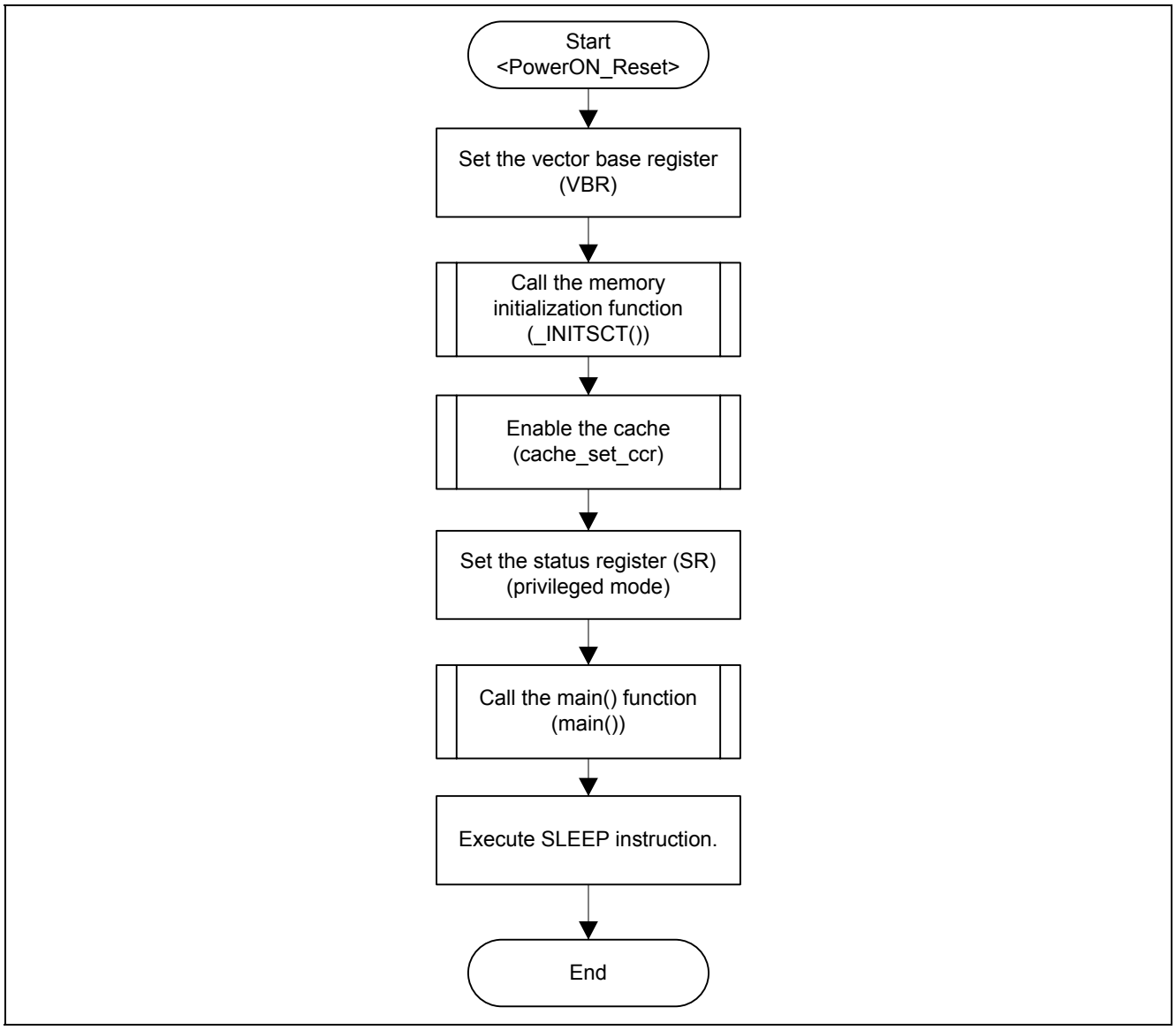

Figure 12 Flowchart for the PowerON_Reset() Function

5. Sample Program

Sample program listing: vhandler.src

The code in this file performs exception handling.

```
001 ;/***********************************************************************
002 ;* DISCLAIMER
003 ;
004 ;* This software is supplied by Renesas Electronics Corporation. and is only
005 ;* intended for use with Renesas products. No other uses are authorized.
006 ;
007 ;* This software is owned by Renesas Electronics Corporation. and is protected under
008 ;* all applicable laws, including copyright laws.
009 ;
010 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT.  ALL SUCH WARRANTIES ARE EXPRESSLY
014 ;* DISCLAIMED.
015 ;
016 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021 ;
022 ;* Renesas reserves the right, without notice, to make changes to this
023 ;* software and to discontinue the availability of this software.
024 ;* By using this software, you agree to the additional terms and
025 ;* conditions found by accessing the following link:
026 ;* http://www.renesas.com/disclaimer
027 ;***********************************************************************/
028 ;/* Copyright (C) 2011. Renesas Electronics Corporation., All Rights Reserved.*/
029 ;/*""FILE COMMENT""************ Technical reference data ***************
030 ;* System Name  : SH7786 Sample Program
031 ;* File Name    : lowlevelinit.src
032 ;* Abstract     : SH7786 Initial Settings Sample Program
033 ;* Version      : Ver 1.00
034 ;* Device       : SH7786
035 ;* Tool-Chain   : High-performance Embedded Workshop (Version 4.07.00.007)
036 ;*              : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
037 ;* OS           : None
038 ;* H/W Platform : This is a sample program with examples of initialization for
039 ;* Description  : the SH7786 on the ALPHA PROJECT SH-4A board, model number AP-SH4AD-3A.
040 ;*              :
041 ;* Operation    :
042 ;* Limitation   :
043 ;*              :
044 ;***********************************************************************
045 ;* History      : 10.Feb.2011 Ver. 1.00 First Release
046 ;*""FILE COMMENT END""*************************************************/
047 ;----------------------------------------------------------------------
048 ;                                                                      |
049 ;   FILE        :vhandler.src                                          |
050 ;   DATE        :Wed, Nov 17, 2010                                     |
051 ;   DESCRIPTION :Reset/Interrupt Handler                                |
052 ;   CPU TYPE    :Other                                                 |
```

```
053 ;                                                                      |
054 ;   This file is generated by Renesas Project Generator (Ver.4.16).    |
055 ;                                                                      |
056 ;-----------------------------------------------------------------------
057
058
059
060             .include "env.inc"
061             .include "vect.inc"
062
063 IMASKclr:   .equ     H'FFFFFF0F
064 RBBLclr:    .equ     H'CFFFFFFF
065 MDRBBLset:  .equ     H'70000000
066
067             .import  _RESET_Vectors
068             .import  _INT_Vectors
069             .import  _INT_MASK
070             .import  _lowlevel_init
071
072 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
073 ;*              macro definition                                       *;
074 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
075                 .macro PUSH_EXP_BASE_REG
076         stc.l   ssr,@-r15               ; save ssr
077         stc.l   spc,@-r15               ; save spc
078         sts.l   pr,@-r15                ; save context registers
079         sts.l   fpscr,@-r15             ; save fpscr registers
080         stc.l   r7_bank,@-r15
081         stc.l   r6_bank,@-r15
082         stc.l   r5_bank,@-r15
083         stc.l   r4_bank,@-r15
084         stc.l   r3_bank,@-r15
085         stc.l   r2_bank,@-r15
086         stc.l   r1_bank,@-r15
087         stc.l   r0_bank,@-r15
088                 .endm
089 ;
090                 .macro POP_EXP_BASE_REG
091         ldc.l   @r15+,r0_bank           ; recover registers
092         ldc.l   @r15+,r1_bank
093         ldc.l   @r15+,r2_bank
094         ldc.l   @r15+,r3_bank
095         ldc.l   @r15+,r4_bank
096         ldc.l   @r15+,r5_bank
097         ldc.l   @r15+,r6_bank
098         ldc.l   @r15+,r7_bank
099         lds.l   @r15+,fpscr
100         lds.l   @r15+,pr
101         ldc.l   @r15+,spc
102         ldc.l   @r15+,ssr
103                 .endm
104 ;
105 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
106 ;    reset                                              ;
107 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
108      .section RSTHandler,code
109 _ResetHandler:
```

```
110             mov.l #_lowlevel_init,r0
111             jsr   @r0
112             nop
113
114             mov.l #EXPEVT,r0
115             mov.l @r0,r0
116             shlr2 r0
117             shlr  r0
118             mov.l #_RESET_Vectors,r1
119             add   r1,r0
120             mov.l @r0,r0
121             jmp   @r0
122             nop
123 ;
124 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
125 ;    exceptional interrupt                                  ;
126 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
127       .section INTHandler,code
128       .export     _INTHandlerPRG
129 _INTHandlerPRG:
130 _ExpHandler:
131             PUSH_EXP_BASE_REG
132 ;
133             mov.l #EXPEVT,r0        ; set event address
134             mov.l @r0,r1            ; set exception code
135             mov.l #_INT_Vectors,r0  ; set vector table address
136             add   #-(h'40),r1       ; exception code - h'40
137             shlr2 r1
138             shlr  r1
139             mov.l @(r0,r1),r3       ; set interrupt function addr
140 ;
141             mov.l #_INT_MASK,r0     ; interrupt mask table addr
142             shlr2 r1
143             mov.b @(r0,r1),r1       ; interrupt mask
144             extu.br1,r1
145 ;
146             stc   sr,r0             ; save sr
147             mov.l #(RBBLclr&IMASKclr),r2     ; RB,BL,mask clear data
148             and   r2,r0             ; clear mask data
149             or    r1,r0             ; set interrupt mask
150             ldc   r0,ssr            ; set current status
151 ;
152             ldc.l r3,spc
153             mov.l #__int_term,r0    ; set interrupt terminate
154             lds   r0,pr
155 ;
156             rte
157             nop
158 ;
159             .pool
160 ;
161 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
162 ;    Interrupt terminate                                    ;
163 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
164             .align4
165 __int_term:
166             mov.l   #MDRBBLset,r0   ; set MD,BL,RB
```

```
167             ldc.l r0,sr             ;
168             POP_EXP_BASE_REG
169             rte                     ; return
170             nop
171 ;
172             .pool
173 ;
174 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
175 ;    TLB miss interrupt                                  ;
176 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
177     .org  H'300
178 _TLBmissHandler:
179             PUSH_EXP_BASE_REG
180 ;
181             mov.l #EXPEVT,r0        ; set event address
182             mov.l @r0,r1            ; set exception code
183             mov.l #_INT_Vectors,r0  ; set vector table address
184             add   #-(h'40),r1       ; exception code - h'40
185             shlr2 r1
186             shlr  r1
187             mov.l @(r0,r1),r3       ; set interrupt function addr
188 ;
189             mov.l #_INT_MASK,r0     ; interrupt mask table addr
190             shlr2 r1
191             mov.b @(r0,r1),r1       ; interrupt mask
192             extu.br1,r1
193 ;
194             stc   sr,r0             ; save sr
195             mov.l #(RBBLclr&IMASKclr),r2     ; RB,BL,mask clear data
196             and   r2,r0             ; clear mask data
197             or    r1,r0             ; set interrupt mask
198             ldc   r0,ssr            ; set current status
199 ;
200             ldc.l r3,spc
201             mov.l #__int_term,r0    ; set interrupt terminate
202             lds   r0,pr
203 ;
204             rte
205             nop
206 ;
207             .pool
208 ;
209 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
210 ;    IRQ                                                ;
211 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
212     .org  H'500
213 _IRQHandler:
214             PUSH_EXP_BASE_REG
215 ;
216             mov.l #INTEVT,r0        ; set event address
217             mov.l @r0,r1            ; set exception code
218             mov.l #_INT_Vectors,r0  ; set vector table address
219             add   #-(h'40),r1       ; exception code - h'40
220             shlr2 r1
221             shlr  r1
222             mov.l @(r0,r1),r3       ; set interrupt function addr
223 ;
```

```
224             mov.l #_INT_MASK,r0     ; interrupt mask table addr
225             shlr2 r1
226             mov.b @(r0,r1),r1       ; interrupt mask
227             extu.br1,r1
228 ;
229             stc   sr,r0             ; save sr
230             mov.l #(RBBLclr&IMASKclr),r2     ; RB,BL,mask clear data
231             and   r2,r0             ; clear mask data
232             or    r1,r0             ; set interrupt mask
233             ldc   r0,ssr            ; set current status
234 ;
235             ldc.l r3,spc
236             mov.l #__int_term,r0    ; set interrupt terminate
237             lds   r0,pr
238 ;
239             rte
240             nop
241 ;
242             .pool
243             .end
```

Sample program listing: resetprg.c

This is the power-on reset function.

```
001 ;/***********************************************************************
002 ;* DISCLAIMER
003 ;
004 ;* This software is supplied by Renesas Electronics Corporation. and is only
005 ;* intended for use with Renesas products. No other uses are authorized.
006 ;
007 ;* This software is owned by Renesas Electronics Corporation. and is protected under
008 ;* all applicable laws, including copyright laws.
009 ;
010 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT.  ALL SUCH WARRANTIES ARE EXPRESSLY
014 ;* DISCLAIMED.
015 ;
016 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021 ;
022 ;* Renesas reserves the right, without notice, to make changes to this
023 ;* software and to discontinue the availability of this software.
024 ;* By using this software, you agree to the additional terms and
025 ;* conditions found by accessing the following link:
026 ;* http://www.renesas.com/disclaimer
027 ;***********************************************************************/
028 ;/* Copyright (C) 2011. Renesas Electronics Corporation., All Rights Reserved.*/
029 ;/*""FILE COMMENT""************ Technical reference data ****************
030 ;* System Name  : SH7786 Sample Program
031 ;* File Name    : lowlevelinit.src
032 ;* Abstract     : SH7786 Initial Settings Sample Program
033 ;* Version      : Ver 1.00
034 ;* Device       : SH7786
035 ;* Tool-Chain   : High-performance Embedded Workshop (Version 4.07.00.007)
036 ;*              : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
037 ;* OS           : None
038 ;* H/W Platform : This is a sample program with examples of initialization for
039 ;* Description  : the SH7786 on the ALPHA PROJECT SH-4A board, model number AP-SH4AD-3A.
040 ;*              :
041 ;* Operation    :
042 ;* Limitation   :
043 ;*              :
044 ;***********************************************************************
045 ;* History      : 10.Feb.2011 Ver. 1.00 First Release
046 ;*""FILE COMMENT END""*************************************************/
047 /***********************************************************************/
048 /*                                                                     */
049 /*  FILE        :resetprg.c                                            */
050 /*  DATE        :Wed, Nov 17, 2010                                     */
051 /*  DESCRIPTION :Reset Program                                         */
052 /*  CPU TYPE    :Other                                                 */
053 /*                                                                     */
054 /*  This file is generated by Renesas Project Generator (Ver.4.16).    */
```

```
055 /*                                                                      */
056 /************************************************************************/
057
058
059
060 #include<machine.h>
061 #include<_h_c_lib.h>
062 //#include <stddef.h>           // Remove the comment when you use errno
063 //#include <stdlib.h>           // Remove the comment when you use rand()
064 #include"typedefine.h"
065 #include"stacksct.h"
066
067 #define SR_Init    0x40000000
068 #ifdef _FPD// when -fpu=double is specified
069 #define FPSCR_Init 0x000C0001
070 #else
071 #define FPSCR_Init 0x00040001
072 #endif
073 #define INT_OFFSET 0x100UL
074
075
076 #ifdef __cplusplus
077 extern "C" {
078 #endif
079 extern void INTHandlerPRG(void);
080 void PowerON_Reset(void);
081 void Manual_Reset(void);
082 void main(void);
083 #ifdef __cplusplus
084 }
085 #endif
086
087 //#ifdef __cplusplus   // Enable I/O in the application(both SIM I/O and hardware I/O)
088 //extern "C" {
089 //#endif
090 //extern void _INIT_IOLIB(void);
091 //extern void _CLOSEALL(void);
092 //#ifdef __cplusplus
093 //}
094 //#endif
095
096 //extern void srand(_UINT);     // Remove the comment when you use rand()
097 //extern _SBYTE *_s1ptr;        // Remove the comment when you use strtok()
098
099 //#ifdef __cplusplus         // Use Hardware Setup
100 //extern "C" {
101 //#endif
102 //extern void HardwareSetup(void);
103 //#ifdef __cplusplus
104 //}
105 //#endif
106
107 //#ifdef __cplusplus      // Remove the comment when you use global class object
108 //extern "C" {            // Sections C$INIT and C$END will be generated
109 //#endif
110 //extern void _CALL_INIT(void);
111 //extern void _CALL_END(void);
```

```
112 //#ifdef __cplusplus
113 //}
114 //#endif
115
116 #pragma section ResetPRG
117
118 #pragma entry PowerON_Reset
119
120 void PowerON_Reset(void)
121 {
122     _UDWORD* ramcr_address;
123
124   set_vbr((void *)((_UINT)INTHandlerPRG - INT_OFFSET));
125
126 //   set_fpscr(FPSCR_Init);
127
128   _INITSCT();
129
130 //   _CALL_INIT();      // Remove the comment when you use global class object
131
132 //   _INIT_IOLIB();     // Enable I/O in the application(both SIM I/O and hardware I/O)
133
134 //   errno=0;              // Remove the comment when you use errno
135 //   srand((_UINT)1);         // Remove the comment when you use rand()
136 //   _s1ptr=NULL;             // Remove the comment when you use strtok()
137
138 //   HardwareSetup();         // Use Hardware Setup
139
140   set_cr(SR_Init);
141
142   main();
143
144 //   _CLOSEALL();    // Close I/O in the application(both SIM I/O and hardware I/O)
145
146 //   _CALL_END();    // Remove the comment when you use global class object
147
148   sleep();
149 }
150
151 void Manual_Reset(void)
152 {
153 }046 * History      : 30.SEP.2010 Ver. 1.00 First Release
047 *""FILE COMMENT END""**************************************************/
048 /*********************************************************************/
049 /*                                                                   */
050 /*  FILE        :resetprg.c                                          */
051 /*  DATE        :Thu, May 13, 2010                                   */
052 /*  DESCRIPTION :Reset Program                                       */
053 /*  CPU TYPE    :Other                                               */
054 /*                                                                   */
055 /*  This file is generated by Renesas Project Generator (Ver.4.16).  */
056 /*                                                                   */
057 /*********************************************************************/
058
059
060
061 #include<machine.h>
```

```
062 #include<_h_c_lib.h>
063 //#include <stddef.h>           // Remove the comment when you use errno
064 //#include <stdlib.h>           // Remove the comment when you use rand()
065 #include"typedefine.h"
066 #include"stacksct.h"
067 #include"cache.h"
068
069
070 #define SR_Init    0x40000000
071 #define INT_OFFSET 0x100UL
072
073 #ifdef __cplusplus
074 extern "C" {
075 #endif
076 extern void INTHandlerPRG(void);
077 void PowerON_Reset(void);
078 void Manual_Reset(void);
079 void main(void);
080 #ifdef __cplusplus
081 }
082 #endif
083
084 #pragma section ResetPRG
085
086
087 #pragma entry PowerON_Reset
088
089 void PowerON_Reset(void)
090 {
091   set_vbr((void *)((_UINT)INTHandlerPRG - INT_OFFSET));
092
093
094   _INITSCT();
095
096 //   errno=0;                 // Remove the comment when you use errno
097 //   srand((_UINT)1);         // Remove the comment when you use rand()
098 //   _s1ptr=NULL;             // Remove the comment when you use strtok()
099   cache_set_ccr(D_CACHE_I_ON | D_CACHE_O_ON);
100
101   set_cr(SR_Init);
102
103   main();
104
105 //   _CLOSEALL();   // Close I/O in the application(both SIM I/O and hardware I/O)
106
107 //   _CALL_END();   // Remove the comment when you use global class object
108
109   sleep();
110 }
111
112 void Manual_Reset(void)
113 {
114 }
```

Sample program listing: dbsct.c

The code in this file performs memory initialization.

```
01 ;/***********************************************************************
02 ;* DISCLAIMER
03 ;
04 ;* This software is supplied by Renesas Electronics Corporation. and is only
05 ;* intended for use with Renesas products. No other uses are authorized.
06 ;
07 ;* This software is owned by Renesas Electronics Corporation. and is protected under
08 ;* all applicable laws, including copyright laws.
09 ;
10 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT.  ALL SUCH WARRANTIES ARE EXPRESSLY
14 ;* DISCLAIMED.
15 ;
16 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 ;
22 ;* Renesas reserves the right, without notice, to make changes to this
23 ;* software and to discontinue the availability of this software.
24 ;* By using this software, you agree to the additional terms and
25 ;* conditions found by accessing the following link:
26 ;* http://www.renesas.com/disclaimer
27 ;***********************************************************************/
28 ;/* Copyright (C) 2011. Renesas Electronics Corporation., All Rights Reserved.*/
29 ;/*""FILE COMMENT""************ Technical reference data ****************
30 ;* System Name  : SH7786 Sample Program
31 ;* File Name    : lowlevelinit.src
32 ;* Abstract     : SH7786 Initial Settings Sample Program
33 ;* Version      : Ver 1.00
34 ;* Device       : SH7786
35 ;* Tool-Chain   : High-performance Embedded Workshop (Version 4.07.00.007)
36 ;*              : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
37 ;* OS           : None
38 ;* H/W Platform : This is a sample program with examples of initialization for
39 ;* Description  : the SH7786 on the ALPHA PROJECT SH-4A board, model number AP-SH4AD-3A.
40 ;*              :
41 ;* Operation    :
42 ;* Limitation   :
43 ;*              :
44 ;***********************************************************************
45 ;* History      : 10.Feb.2011 Ver. 1.00 First Release
46 ;*""FILE COMMENT END""**************************************************/
47 /***********************************************************************/
48 /*                                                                     */
49 /*  FILE        :dbsct.c                                               */
50 /*  DATE        :Wed, Nov 17, 2010                                     */
51 /*  DESCRIPTION :Setting of B,R Section                                */
52 /*  CPU TYPE    :Other                                                 */
53 /*                                                                     */
54 /*  This file is generated by Renesas Project Generator (Ver.4.16).    */
```

```
55 /*                                                                      */
56 /**********************************************************************/
57
58
59
60 #include "typedefine.h"
61
62 #pragma section $DSEC
63 static const struct {
64     _UBYTE *rom_s;      /* Start address in ROM for the initialization data section */
65     _UBYTE *rom_e;      /* End address in ROM for the initialization data section  */
66     _UBYTE *ram_s;      /* Start address in RAM for the initialization data section */
67 }   DTBL[] = {
68     { __sectop("D"), __secend("D"), __sectop("R") }
69 };
70 #pragma section $BSEC
71 static const struct {
72     _UBYTE *b_s;          /* Start address of the uninitialized data section */
73     _UBYTE *b_e;          /* End address of the uninitialized data section */
74 }   BTBL[] = {
75     { __sectop("B"), __secend("B") }
76 };
```

Sample program listing: vecttbl.src

The code in this file sets the interrupt priorities used during interrupt handling.

```
01 ;/***********************************************************************
02 ;* DISCLAIMER
03 ;
04 ;* This software is supplied by Renesas Electronics Corporation. and is only
05 ;* intended for use with Renesas products. No other uses are authorized.
06 ;
07 ;* This software is owned by Renesas Electronics Corporation. and is protected under
08 ;* all applicable laws, including copyright laws.
09 ;
10 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
11 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
12 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
13 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT.  ALL SUCH WARRANTIES ARE EXPRESSLY
14 ;* DISCLAIMED.
15 ;
16 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
17 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
18 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
19 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
20 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
21 ;
22 ;* Renesas reserves the right, without notice, to make changes to this
23 ;* software and to discontinue the availability of this software.
24 ;* By using this software, you agree to the additional terms and
25 ;* conditions found by accessing the following link:
26 ;* http://www.renesas.com/disclaimer
27 ;***********************************************************************/
28 ;/* Copyright (C) 2011. Renesas Electronics Corporation., All Rights Reserved.*/
29 ;/*""FILE COMMENT""************ Technical reference data ****************
30 ;* System Name  : SH7786 Sample Program
31 ;* File Name    : lowlevelinit.src
32 ;* Abstract     : SH7786 Initial Settings Sample Program
33 ;* Version      : Ver 1.00
34 ;* Device       : SH7786
35 ;* Tool-Chain   : High-performance Embedded Workshop (Version 4.07.00.007)
36 ;*              : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
37 ;* OS           : None
38 ;* H/W Platform : This is a sample program with examples of initialization for
39 ;* Description  : the SH7786 on the ALPHA PROJECT SH-4A board, model number AP-SH4AD-3A.
40 ;*              :
41 ;* Operation    :
42 ;* Limitation   :
43 ;*              :
44 ;***********************************************************************
45 ;* History      : 10.Feb.2011 Ver. 1.00 First Release
46 ;*""FILE COMMENT END""**************************************************/
47 ;----------------------------------------------------------------------
48 ;                                                                     |
49 ;   FILE        :vecttbl.src                                          |
50 ;   DATE        :Wed, Nov 17, 2010                                    |
51 ;   DESCRIPTION :Initialize of Vector Table                           |
52 ;   CPU TYPE    :Other                                                |
53 ;                                                                     |
54 ;   This file is generated by Renesas Project Generator (Ver.4.16).   |
```

```
55 ;                                                                      |
56 ;---------------------------------------------------------------------
57
58
59
60          .include        "vect.inc"
61
62          .section        VECTTBL,data
63          .export         _RESET_Vectors
64
65 _RESET_Vectors:
66 ;<<VECTOR DATA START (POWER ON RESET)>>
67          ;H'000 Power On Reset (Hitachi-UDI RESET)
68          .data.l_PowerON_Reset
69 ;<<VECTOR DATA END (POWER ON RESET)>>
70 ;<<VECTOR DATA START (MANUAL RESET)>>
71          ;H'020 Manual Reset
72          .data.l_Manual_Reset
73 ;<<VECTOR DATA END (MANUAL RESET)>>
74 ; Reserved
75          .datab.l        8,H'00000000
76 ;<<VECTOR DATA START (TBL RESET)>>
77          ;H'140 TBL Reset (DATA TBL Reset)
78          .data.l         _TBL_Reset
79 ;<<VECTOR DATA END (TBL RESET)>>
80
81          .section    INTTBL,data
82          .export     _INT_Vectors
83 _INT_Vectors:
84
85          .export     _INT_MASK
86 _INT_MASK:
87
88                          .end
89
90
```

Sample program listing: vect.inc

This file codes the interrupt tables registered for the interrupt handlers.

```
01 ;---------------------------------------------------------------------
02 ;                                                                     |
03 ;   FILE        :vect.inc                                             |
04 ;   DATE        :Mon, Feb 07, 2011                                    |
05 ;   DESCRIPTION :Definition of Vector                                 |
06 ;   CPU TYPE    :Other                                                |
07 ;                                                                     |
08 ;   This file is generated by Renesas Project Generator (Ver.4.16).   |
09 ;                                                                     |
10 ;---------------------------------------------------------------------
11
12
13
14
15 ;<<VECTOR DATA START (POWER ON RESET)>>
16         ;H'000 Power On Reset (Hitachi-UDI RESET)
17         .global_PowerON_Reset
18 ;<<VECTOR DATA END (POWER ON RESET)>>
19 ;<<VECTOR DATA START (MANUAL RESET)>>
20         ;H'020 Manual Reset
21         .global_Manual_Reset
22 ;<<VECTOR DATA END (MANUAL RESET)>>
23 ;<<VECTOR DATA START (TBL RESET)>>
24         ;H'140 TBL Reset (DATA TBL Reset)
25         .global         _TBL_Reset
26 ;<<VECTOR DATA END (TBL RESET)>>
27
```

Sample program listing: intprg.c

The code in this file registers the interrupt processing functions in the interrupt handlers.

```
01 ;---------------------------------------------------------------------
02 ;                                                                    |
03 ;   FILE        :intprg.src                                          |
04 ;   DATE        :Wed, Nov 17, 2010                                   |
05 ;   DESCRIPTION :Interrupt Program                                   |
06 ;   CPU TYPE    :Other                                               |
07 ;                                                                    |
08 ;   This file is generated by Renesas Project Generator (Ver.4.16).  |
09 ;                                                                    |
10 ;---------------------------------------------------------------------
11
12
13
14     .include    "vect.inc"
15     .section    IntPRG, code
16
17 ;H'140 TBL Reset (DATA TBL Reset)
18 _TBL_Reset
19     sleep
20     nop
21     .end
22
```

Sample program listing: lowlevelinit.src

The code in this file initializes the LBSC and DBSC3.

```
001 ;/***********************************************************************
002 ;* DISCLAIMER
003 ;
004 ;* This software is supplied by Renesas Electronics Corporation. and is only
005 ;* intended for use with Renesas products. No other uses are authorized.
006 ;
007 ;* This software is owned by Renesas Electronics Corporation. and is protected under
008 ;* all applicable laws, including copyright laws.
009 ;
010 ;* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES
011 ;* REGARDING THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY,
012 ;* INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
013 ;* PARTICULAR PURPOSE AND NON-INFRINGEMENT.  ALL SUCH WARRANTIES ARE EXPRESSLY
014 ;* DISCLAIMED.
015 ;
016 ;* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
017 ;* ELECTRONICS CORPORATION. NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
018 ;* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES
019 ;* FOR ANY REASON RELATED TO THE THIS SOFTWARE, EVEN IF RENESAS OR ITS
020 ;* AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
021 ;
022 ;* Renesas reserves the right, without notice, to make changes to this
023 ;* software and to discontinue the availability of this software.
024 ;* By using this software, you agree to the additional terms and
025 ;* conditions found by accessing the following link:
026 ;* http://www.renesas.com/disclaimer
027 ;***********************************************************************/
028 ;/* Copyright (C) 2011. Renesas Electronics Corporation., All Rights Reserved.*/
029 ;/*""FILE COMMENT""************ Technical reference data ***************
030 ;* System Name  : SH7786 Sample Program
031 ;* File Name    : lowlevelinit.src
032 ;* Abstract     : Sample Program of SH7786 initialization
033 ;* Version      : Ver 1.00
034 ;* Device       : SH7786
035 ;* Tool-Chain   : High-performance Embedded Workshop (Version 4.07.00.007)
036 ;*              : C/C++ Compiler Package for SuperH Family (V.9.3.2.0)
037 ;* OS           : None
038 ;* H/W Platform : SH-4A Board P/N:AP-SH4AD-3A (Manufacturer:ALPHA PROJECT)
039 ;* Description  : It is an example program of the example of SH7786 initialization.
040 ;*              :
041 ;* Operation    :
042 ;* Limitation   :
043 ;*              :
044 ;***********************************************************************
045 ;* History      : 10.Feb.2011 Ver. 1.00 First Release
046 ;*""FILE COMMENT END""*************************************************/
047
048  .macro  write32  addr, data
049   mov.l \addr, r1
050   mov.l \data, r0
051   mov.l r0, @r1
052  .endm
053
054  .macro  write16  addr, data
```

```
055   mov.l \addr,r1
056   mov.l \data,r0
057   mov.w r0, @r1
058  .endm
059
060  .macro  write8  addr, data
061   mov.l \addr ,r1
062   mov.l \data ,r0
063   mov.b r0, @r1
064  .endm
065
066  .macro  read32  addr
067   mov.l \addr ,r1
068   mov.l @r1 ,r0
069  .endm
070
071  .macro wait_timer  time
072   mov.l \time ,r3
073 loop\@:
074   nop
075   tst r3, r3
076   bf/s  loop\@
077   dt  r3
078  .endm
079
080  .include  "lowlevelinit.inc"
081
082   .export _lowlevel_init
083
084 _lowlevel_init:
085
086 ;  /*------- LBSC -------*/
087   write32 #MMSELR_A, #MMSELR_D
088
089   write32 #BCR_A,    #BCR_D
090
091   write32 #CS0_BCR_A, #CS0_BCR_D
092   write32 #CS0_WCR_A, #CS0_WCR_D
093
094   write32 #CS1_BCR_A, #CS1_BCR_D
095   write32 #CS1_WCR_A, #CS1_WCR_D
096
097 ;  /*------- DBSC3 -------*/
098   wait_timer  #WAIT_200US
099   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D1
100   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D2
101   write32 #DBSC3_DBKIND_A,    #DBSC3_DBKIND_D
102   write32 #DBSC3_DBCONF_A,    #DBSC3_DBCONF_D
103   write32 #DBSC3_DBTR0_A,     #DBSC3_DBTR0_D
104   write32 #DBSC3_DBTR1_A,     #DBSC3_DBTR1_D
105   write32 #DBSC3_DBTR2_A,     #DBSC3_DBTR2_D
106   write32 #DBSC3_DBTR3_A,     #DBSC3_DBTR3_D
107   write32 #DBSC3_DBTR4_A,     #DBSC3_DBTR4_D
108   write32 #DBSC3_DBTR5_A,     #DBSC3_DBTR5_D
109   write32 #DBSC3_DBTR6_A,     #DBSC3_DBTR6_D
110   write32 #DBSC3_DBTR7_A,     #DBSC3_DBTR7_D
111   write32 #DBSC3_DBTR8_A,     #DBSC3_DBTR8_D
```

```
112   write32 #DBSC3_DBTR9_A,     #DBSC3_DBTR9_D
113   write32 #DBSC3_DBTR10_A,    #DBSC3_DBTR10_D
114   write32 #DBSC3_DBTR11_A,    #DBSC3_DBTR11_D
115   write32 #DBSC3_DBTR12_A,    #DBSC3_DBTR12_D
116   write32 #DBSC3_DBTR13_A,    #DBSC3_DBTR13_D
117   write32 #DBSC3_DBTR14_A,    #DBSC3_DBTR14_D
118   write32 #DBSC3_DBTR15_A,    #DBSC3_DBTR15_D
119   write32 #DBSC3_DBTR16_A,    #DBSC3_DBTR16_D
120   write32 #DBSC3_DBTR17_A,    #DBSC3_DBTR17_D
121
122 ;  /*-- Seq 7 --*/
123   write32 #DBSC3_DBPDLCK_A,   #DBSC3_DBPDLCK0_D
124   write32 #DBSC3_DBPDRGA_A,   #DBSC3_DBPDRGA0_D
125   write32 #DBSC3_DBPDRG_A,    #DBSC3_DBPDRG0_D
126   write32 #DBSC3_DBPDLCK_A,   #DBSC3_DBPDLCK1_D
127
128 ;  /*-- Seq 8 --*/
129   write32 #DBSC3_DBPDCNT3_A, #DBSC3_DBPDCNT3_D
130 ;  /*-- Seq 9 --*/
131   write32 #DBSC3_DBPDCNT1_A, #DBSC3_DBPDCNT1_D
132 ;  /*-- Seq 10 --*/
133   wait_timer  #WAIT_200US ;/* 10us */
134 ;  /*-- Seq 11 --*/
135   write32 #DBSC3_DBPDCNT0_A, #DBSC3_DBPDCNT0_D
136 ;  /*-- Seq 12 --*/
137   wait_timer  #WAIT_200US ;/* 50us */
138
139 ; /* DDR_init_14 */
140 ;  /*-- Seq 14 --*/
141   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D3
142 ;  /*-- Seq 15 --*/
143   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D4
144 ;  /*-- Seq 16 --*/
145   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D5
146 ;  /*-- Seq 17 --*/
147   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D6
148 ;  /*-- Seq 18 --*/
149   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D7
150 ;  /*-- Seq 19 --*/
151   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D8
152 ;  /*-- Seq 20 --*/
153   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D9
154 ;  /*-- Seq 21 --*/
155   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D10
156 ;  /*-- Seq 22 --*/
157   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D11
158   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D12
159   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D13
160   write32 #DBSC3_DBCMD_A,     #DBSC3_DBCMD_D14
161
162 ;  /*-- Seq 23 --*/
163   write32 #DBSC3_DBBS0CNT0_A,#DBSC3_DBBS0CNT0_D
164   write32 #DBSC3_DBBS0CNT1_A,#DBSC3_DBBS0CNT1_D
165
166 ;  /*-- Seq 24 --*/
167   write32 #DBSC3_DBRFCNF0_A,  #DBSC3_DBRFCNF0_D
168   write32 #DBSC3_DBRFCNF1_A,  #DBSC3_DBRFCNF1_D
```

```
169   write32 #DBSC3_DBRFCNF2_A,  #DBSC3_DBRFCNF2_D
170
171 ;  /*-- Seq 25 --*/
172   write32 #DBSC3_DBRFEN_A,#DBSC3_DBRFEN_D
173 ;  /*-- Seq 26 --*/
174   write32 #DBSC3_DBACEN_A,#DBSC3_DBACEN_D
175   read32  #DBSC3_DBWAIT_A
176
177   bra lbsc_end
178   nop
179
180 lbsc_end:
181
182   write32 #CCR_A,  #CCR_D
183   rts
184   nop
185
186  .end
```

Sample program listing: lowlevelinit.inc

The code in this file provides an example of LBSC and DBSC3 initialization.

```
001 ;/*********************************************************************
002 ;*
003 ;* Device     : SH-4A/SH7786
004 ;*
005 ;* File Name  : lowlevelinit.inc
006 ;*
007 ;* Abstract   : Definition of lowlevelinit.
008 ;*
009 ;* History    : 1.00  (2011-0x-x)  [Hardware Manual Revision : 1.00]
010 ;*
011 ;* Copyright(c) 2010 Renesas Electronics Corporation.
012 ;*               And Renesas Solutions Corp.,All Rights Reserved.
013 ;*
014 ;*********************************************************************/
015
016 ;/*------- DBSC3 -------*/
017 DBSC3_BASE          .equ  H'ffa00000
018 DBSC3_DBCMD_A:      .equ  DBSC3_BASE + H'18
019 DBSC3_DBKIND_A:     .equ  DBSC3_BASE + H'20
020 DBSC3_DBCONF_A:     .equ  DBSC3_BASE + H'24
021 DBSC3_DBTR0_A:      .equ  DBSC3_BASE + H'40
022 DBSC3_DBTR1_A:      .equ  DBSC3_BASE + H'44
023 DBSC3_DBTR2_A:      .equ  DBSC3_BASE + H'48
024 DBSC3_DBTR3_A:      .equ  DBSC3_BASE + H'50
025 DBSC3_DBTR4_A:      .equ  DBSC3_BASE + H'54
026 DBSC3_DBTR5_A:      .equ  DBSC3_BASE + H'58
027 DBSC3_DBTR6_A:      .equ  DBSC3_BASE + H'5C
028 DBSC3_DBTR7_A:      .equ  DBSC3_BASE + H'60
029 DBSC3_DBTR8_A:      .equ  DBSC3_BASE + H'64
030 DBSC3_DBTR9_A:      .equ  DBSC3_BASE + H'68
031 DBSC3_DBTR10_A:     .equ  DBSC3_BASE + H'6C
032 DBSC3_DBTR11_A:     .equ  DBSC3_BASE + H'70
033 DBSC3_DBTR12_A:     .equ  DBSC3_BASE + H'74
034 DBSC3_DBTR13_A:     .equ  DBSC3_BASE + H'78
035 DBSC3_DBTR14_A:     .equ  DBSC3_BASE + H'7C
036 DBSC3_DBTR15_A:     .equ  DBSC3_BASE + H'80
037 DBSC3_DBTR16_A:     .equ  DBSC3_BASE + H'84
038 DBSC3_DBTR17_A:     .equ  DBSC3_BASE + H'88
039
040 DBSC3_DBPDLCK_A:    .equ  DBSC3_BASE + H'280
041 DBSC3_DBPDRGA_A:    .equ  DBSC3_BASE + H'290
042 DBSC3_DBPDRG_A:     .equ  DBSC3_BASE + H'2A0
043
044 DBSC3_DBPDCNT0_A:   .equ  DBSC3_BASE + H'200
045 DBSC3_DBPDCNT1_A:   .equ  DBSC3_BASE + H'204
046 DBSC3_DBPDCNT3_A:   .equ  DBSC3_BASE + H'20C
047
048 DBSC3_DBBS0CNT0_A:  .equ  DBSC3_BASE + H'300
049 DBSC3_DBBS0CNT1_A:  .equ  DBSC3_BASE + H'304
050
051 DBSC3_DBRFCNF0_A:   .equ  DBSC3_BASE + H'E0
052 DBSC3_DBRFCNF1_A:   .equ  DBSC3_BASE + H'E4
053 DBSC3_DBRFCNF2_A:   .equ  DBSC3_BASE + H'E8
054
```

```
055 DBSC3_DBRFEN_A:      .equ  DBSC3_BASE + H'14
056 DBSC3_DBACEN_A:      .equ  DBSC3_BASE + H'10
057 DBSC3_DBWAIT_A:      .equ  DBSC3_BASE + H'1C
058
059 DBSC3_DBCMD_D1:      .equ  H'20000000
060 DBSC3_DBCMD_D2:      .equ  H'1000d056
061 DBSC3_DBCMD_D3:      .equ  H'0000D056
062 DBSC3_DBCMD_D4:      .equ  H'2100D056
063 DBSC3_DBCMD_D5:      .equ  H'0000D056
064 DBSC3_DBCMD_D6:      .equ  H'0000D056
065 DBSC3_DBCMD_D7:      .equ  H'0000D056
066 DBSC3_DBCMD_D8:      .equ  H'0000D056
067 DBSC3_DBCMD_D9:      .equ  H'11000041
068 DBSC3_DBCMD_D10:     .equ  H'2A000008
069 DBSC3_DBCMD_D11:     .equ  H'2B000000
070 DBSC3_DBCMD_D12:     .equ  H'29000042
071 DBSC3_DBCMD_D13:     .equ  H'28000930
072 DBSC3_DBCMD_D14:     .equ  H'03000200
073
074 DBSC3_DBKIND_D:      .equ  H'00000007
075 DBSC3_DBCONF_D:      .equ  H'0d030a02
076
077 DBSC3_DBTR0_D:       .equ  H'00000007
078 DBSC3_DBTR1_D:       .equ  H'00000006
079 DBSC3_DBTR2_D:       .equ  H'00000000
080 DBSC3_DBTR3_D:       .equ  H'00000007
081 DBSC3_DBTR4_D:       .equ  H'00070007
082 DBSC3_DBTR5_D:       .equ  H'0000001B
083 DBSC3_DBTR6_D:       .equ  H'00000014
084 DBSC3_DBTR7_D:       .equ  H'00000006
085 DBSC3_DBTR8_D:       .equ  H'0000001B
086 DBSC3_DBTR9_D:       .equ  H'00000004
087 DBSC3_DBTR10_D:      .equ  H'00000008
088 DBSC3_DBTR11_D:      .equ  H'00000007
089 DBSC3_DBTR12_D:      .equ  H'0000000F
090 DBSC3_DBTR13_D:      .equ  H'0000003B
091 DBSC3_DBTR14_D:      .equ  H'00000006
092 DBSC3_DBTR15_D:      .equ  H'00000003
093 DBSC3_DBTR16_D:      .equ  H'00160002
094 DBSC3_DBTR17_D:      .equ  H'000C000C
095
096 DBSC3_DBPDLCK0_D:    .equ  H'0000a55a
097 DBSC3_DBPDLCK1_D:    .equ  H'00000000
098 DBSC3_DBPDRGA0_D:    .equ  H'00000028
099 DBSC3_DBPDRG0_D:     .equ  H'00017200
100
101 DBSC3_DBPDCNT0_D:    .equ  H'00000001
102 DBSC3_DBPDCNT1_D:    .equ  H'00000001
103 DBSC3_DBPDCNT3_D:    .equ  H'00004000 ;/* manual will be fixed*/
104
105 DBSC3_DBBS0CNT0_D:   .equ  H'00000000
106 DBSC3_DBBS0CNT1_D:   .equ  H'00000000
107
108 DBSC3_DBRFCNF0_D:    .equ  H'000000C8
109 DBSC3_DBRFCNF1_D:    .equ  H'00000ED8
110 DBSC3_DBRFCNF2_D:    .equ  H'00000000
111 DBSC3_DBRFEN_D:      .equ  H'00000001
```

```
112 DBSC3_DBACEN_D:      .equ  H'00000001
113
114 WAIT_200US:             .equ  33333
115
116 ;/*------- LBSC -------*/
117 MMSELR_A:               .equ  H'fc400020
118 BCR_A:                  .equ  H'FF801000
119 CS0_BCR_A:              .equ  H'FF802000
120 CS1_BCR_A:              .equ  H'FF802010
121 CS2_BCR_A:              .equ  H'FF802020
122 CS3_BCR_A:              .equ  H'FF802030
123 CS4_BCR_A:              .equ  H'FF802040
124 CS5_BCR_A:              .equ  H'FF802050
125 CS6_BCR_A:              .equ  H'FF802060
126 CS0_WCR_A:              .equ  H'FF802008
127 CS1_WCR_A:              .equ  H'FF802018
128 CS2_WCR_A:              .equ  H'FF802028
129 CS3_WCR_A:              .equ  H'FF802038
130 CS4_WCR_A:              .equ  H'FF802048
131 CS5_WCR_A:              .equ  H'FF802058
132 CS6_WCR_A:              .equ  H'FF802068
133 CS5_PCMCIA_A:           .equ  H'FF802070
134 CS6_PCMCIA_A:           .equ  H'FF802080
135
136 MMSELR_D:               .equ  H'a5a50003
137
138 BCR_D:                  .equ  H'00000000
139 CS0_BCR_D:              .equ  H'11111670
140 CS0_WCR_D:              .equ  H'42000008
141 CS1_BCR_D:              .equ  H'11111670
142 CS1_WCR_D:              .equ  H'02000003
143
144 ;/* Chache */
145 CCR_A:                  .equ  H'ff00001c
146 CCR_D:                  .equ  H'0000090b
147
```

6. Reference Documents

- Software Manual
  SH4-A Software Manual (REJ09B0003)
  (The latest version can be downloaded from the Renesas Electronics Web site.)
- Hardware Manual
  SH7786 Group Hardware Manual (REJ09B0501)
  (The latest version can be downloaded from the Renesas Electronics Web site.)

Website and Support

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

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

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Mar.31.11	—	First edition issued
1.01	Jul.15.11	2	Section 1.3 amended

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins
   Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.
   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment when power is supplied.
   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.
   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.
5. Differences between Products
   Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.
   — The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

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.

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

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

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

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

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

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

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.1