SH7262/SH7264 Group

Serial Sound Interface in Master Transmitter Mode

R01AN0545EJ0101
Rev. 1.01
Feb. 23, 2011

Summary

This application note describes an example of setting the SH7262/SH7264 Microcomputers (MCUs) Serial Sound Interface (SSI) in master transmitter mode.

Target Device

SH7262/SH7264 MCU (In this document, SH7262/SH7264 are described as "SH7264".)

Contents

1. Introduction

1.1 Specifications

Sets the SH7264 Serial Sound Interface (SSI) in master transmitter mode to transmit the PCM data. To transfer data to SSI, use the Direct Memory Access Controller (DMAC) to transfer data to SSI.

1.2 Modules Used

- Serial Sound Interface (SSI)
- Direct Memory Access Controller (DMAC)
- General-purpose I/O Ports
- Interrupt Controller

1.3 Applicable Conditions

MCU	SH7262/SH7264
Operating Frequency	Internal clock: 144 MHz Bus clock: 72 MHz Peripheral clock: 36 MHz
Integrated Development Environment	Renesas Electronics Corporation High-performance Embedded Workshop Ver.4.07.00
C Compiler	Renesas Electronics SuperH RISC engine Family C/C++ compiler package Ver.9.03 Release 00
Compiler Options	Default setting in the High-performance Embedded Workshop (-cpu=sh2afpu -fpu=single -debug -gbr=auto -global_volatile=0 -opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1)

1.4 Related Application Note

Refer to the related application notes as follows:

- SH7262/SH7264 Group Example of Initialization
- SH7262/SH7264 Group Serial Sound Interface in Slave Receiver Mode
- SH7262/SH7264 Group Serial Sound Interface in Master Transceiver Mode

1.5 About Active-low Pins (Signals)

The symbol "#" suffixed to the pin (or signal) names indicates that the pins (or signals) are active-low.

2. Applications

This application sets the sampling frequency of the SSI to 44.1 kHz to operate as the master transmitter.

2.1 SSI Operation

The SSI has the following features

- Number of channels: 4
- Operating mode: Non-compressed mode

Non-compressed mode supports the serial audio streams divided by channels.

- Operates both as the transmitter and the receiver
- Channel 0 supports full-duplex communication
- Supports the serial bus format
- Asynchronous transfer between the data buffer and the shift register
- Clock divide ratio used in the serial bus interface selectable
- Controls the data transmission/reception by the DMAC or interrupts
- Oversampling clock options as follows:
  AUDIO_CLK pin
  AUDIO_X1, AUDIO_X2 pins
- Eight deep FIFO buffer included both in the transmitter and receiver

Figure 1 shows the SSI block diagram.

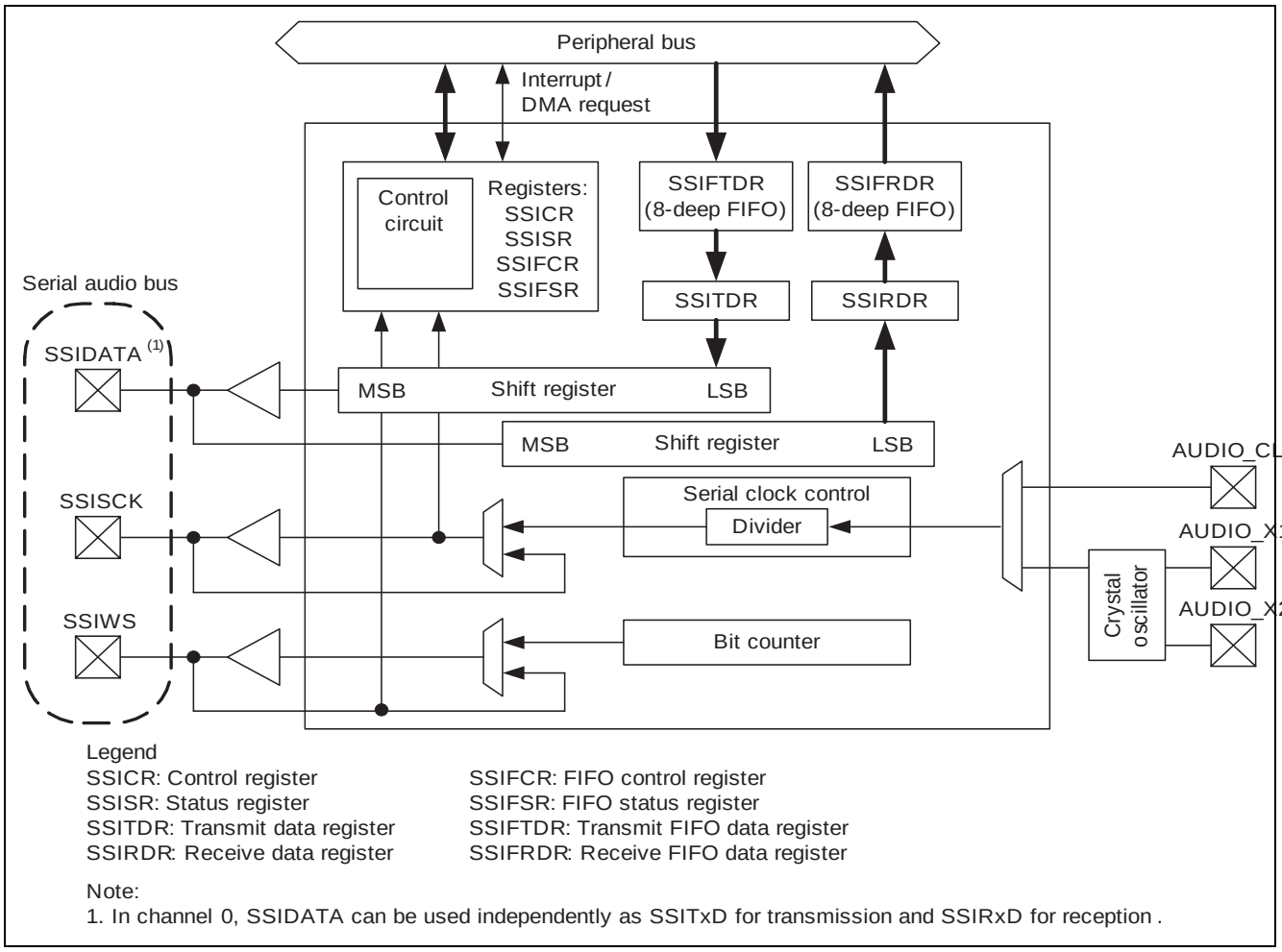

Figure 1 SSI Block Diagram

2.2 SSI Setting Procedure

Figure 2 shows the flow chart of setting the SSI. Figure 3 shows the flow chart of setting the DMAC.

Refer to the SH7262 Group, SH7264 Group Hardware Manual for details on registers.

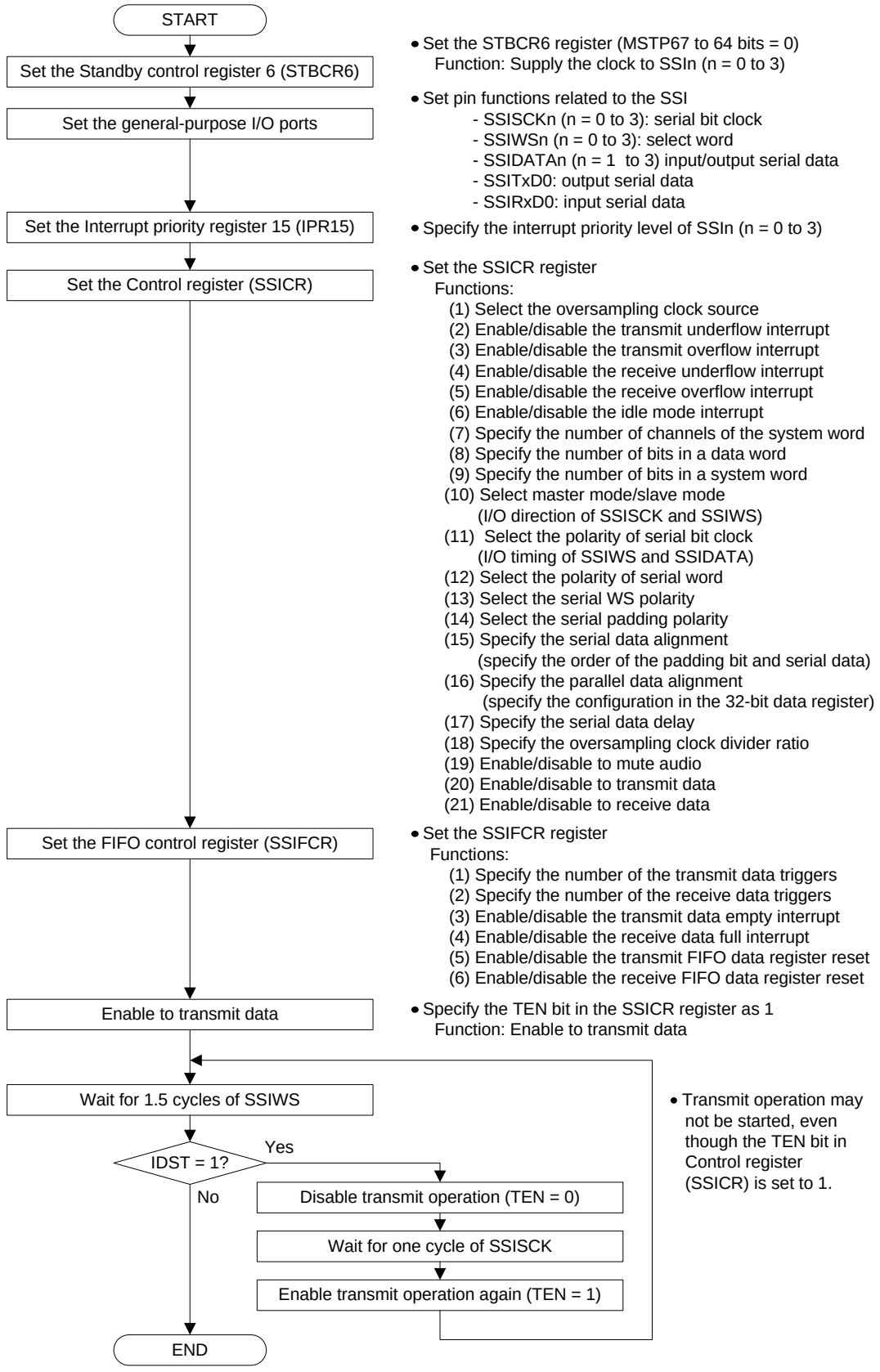

Figure 2 SSI Setup Flow Chart

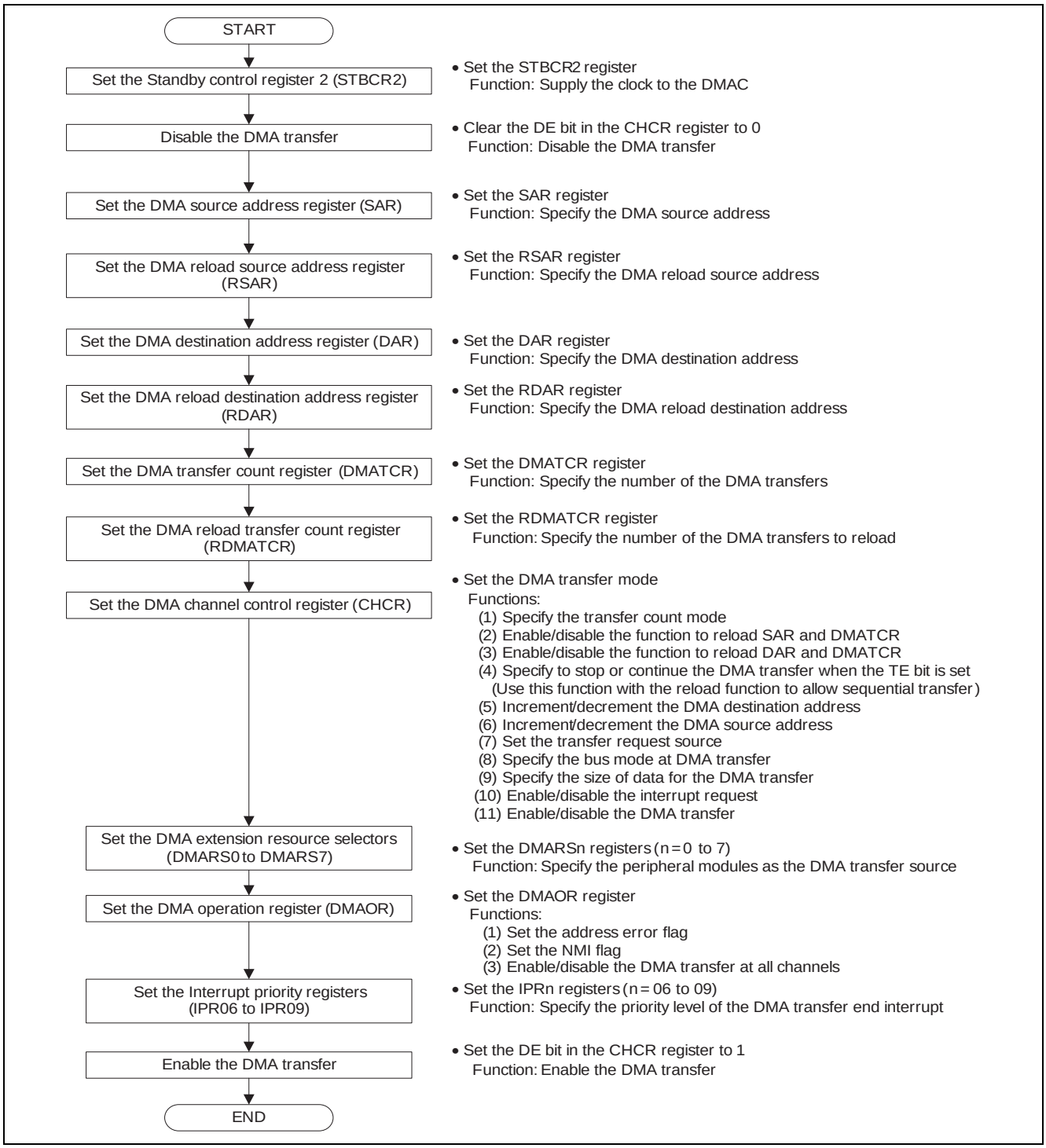

Figure 3 DMAC Setup Flow Chart

2.3 Sample Program Operation

The SSI activates the channel 1 of the DMAC by the DMA transfer request (transmit data empty interrupt) in the sample program. The DMAC transfers data from an external memory to the Transmit FIFO data register (SSIFTDR) in the SSI channel 1. When the SSI detects a space in the Transmit data register (SSITDR), it transfers the data from the SSIFTDR register to the SSITDR register. When a transmission request occurs, the SSIDATA1 pin outputs the data in the SSITDR register via the shift register.

The sample program continues to transfer ten samples (40 bytes) of PCM data for four times, and SSI is muted after the transfer is complete.

SSI setting in the sample program is as follows:

- Channel used: channel 1
- Operating mode: master transmitter
- Data transmission controlled by: DMAC
- Oversampling clock: AUDIO_X1 input (11.2896 MHz)
- Serial oversampling clock frequency: One quarter the oversampling clock frequency (2.811 MHz)
- Data word length: 16 bits
- System word length: 32 bits
- Padding bit: Low level
- No delay between the SSIWS and SSIDATA signals
- Outputs the SSIWS and SSIDATA signals at the falling edge of the SSISCK signal
- Sampling frequency: 44.1 kHz (354 ns or 2.8224 MHz x 32 bits x 2)
- Outputs "H'FFFF" at the data word 1 (channel L) of the 1st channel, and "H'0000" at the data word 2 (channel R) of the 2nd channel.

Figure 4 shows the signal waveform in the sample program. Figure 5 shows the sample program block diagram.

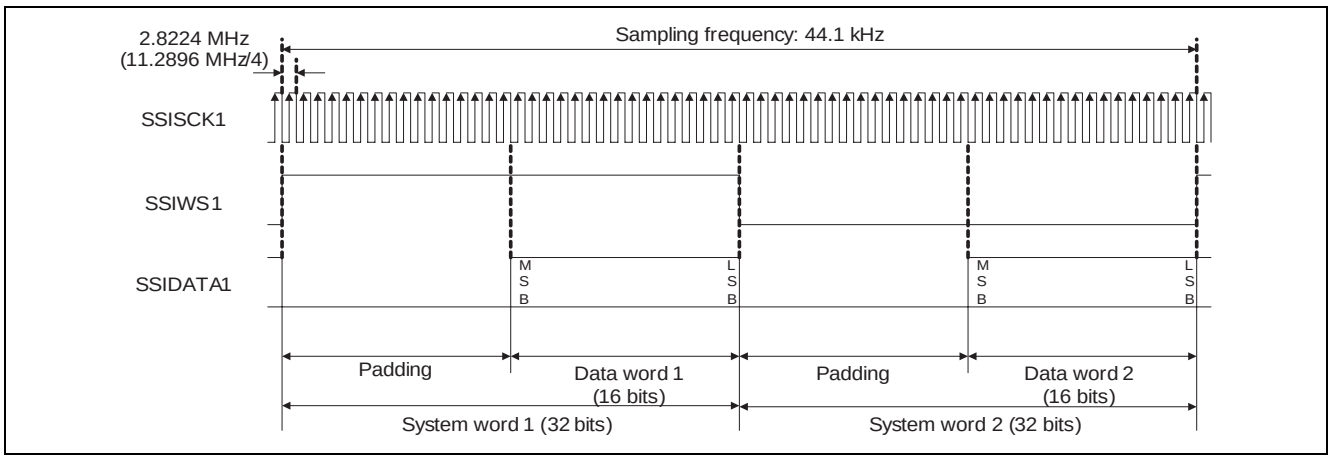

Figure 4 Signal Output Waveform in the Sample Program

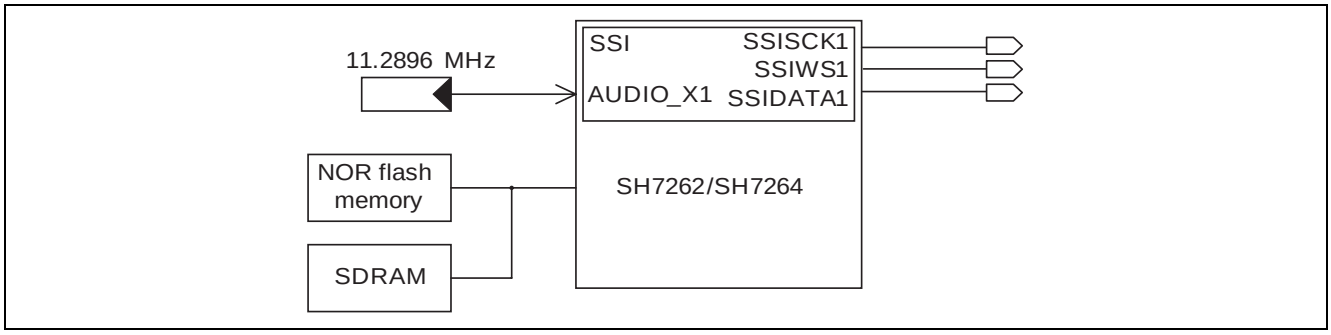

Figure 5 Sample Program Block Diagram

2.4 Sample Program Procedure

The table below lists the SSI registers setting in the sample program.

Table 2 lists the DMAC registers setting in the sample program. Figure 6 shows the flow chart of the sample program.

Table 1 SSI Registers Setting

Register Name	Address	Value	Description
Control register 1 (SSICR_1)	H'FFFF 0800	H'300B D520	• CKS bit = B'0 (Specifies the oversampling clock as AUDIO_X1 input) • TUIEN bit = B'1 (Enables the Transmit underflow interrupt) • TOIEN bit = B'1 (Enables the Transmit overflow interrupt) • RUIEN bit = B'0 (Disables the Receive underflow interrupt) • ROIEN bit = B'0 (Disables the Receive overflow interrupt) • IIEN bit = B'0 (Disables the idle mode interrupt) • CHNL [1:0] bits = B'00 (System words have one channel) • DWL [2:0] bits = B'001 (Data word length: 16 bits) • SWL [2:0] bits = B'011 (System word length: 32 bits) • SCKD bit = B'1 (Serial bit clock output, master mode) • SWSD bit = B'1 (Serial word select output, master mode) • SCKP bit = B'0 (Outputs SSIWS and SSIDATA signals at the falling edge of the SSISCK signal) • SWSP bit = B'1 (SSIWS is High for the 1st channel, and is Low for the 2nd channel) • SPDP bit = B'0 (Padding bits are low) • SDTA bit = B'1 (Transmits and receives in the order of padding bits, and serial data) • PDTA bit = B'0 (Transmits and receives lower bits of parallel data) • DEL bit = B'1 (No delay between the SSIWS and SSIDATA) • CKDV bits [3:0]= B'0010 (Specifies the oversampling clock as audio Φ4) • MUEN bit = B'0 (Not muted) • TEN bit = B'0 (Disables to transmit data) • REN bit = B'0 (Disables to receive data)
		H'300B D522	• TEN bit = B'1 (Enables to transmit data)
FIFO control register 1 (SSIFCR_1)	H'FFFF 0810	H'0000 0008	• TTRG[1:0] bits = B'00 (Number of transmit data triggers: 7) • RTRG [1:0] bits = B'00 (Number of receive data triggers: 1) • TIE bit = B'1 (Enables the transmit data empty interrupt) • RIE bit = B'0 (Disables the receive data full interrupt) • TFRST bit = B'0 (Disables to reset the transmit FIFO) • RFRST bit = B'0 (Disables to reset the receive FIFO)

Table 2 DMAC Registers Setting

Register Name	Address	Value	Description
DMA channel control register 1 (CHCR_1)	H'FFFE 101C	H'0000 0000	• DE bit = B'0 (Disables the DMA transfer)
		H'2010 1814	• TC bit = B'0 (Transmits data once by one transfer request) • RLDSAR bit = B'1 (Enables the SAR reload function) • RLDDAR bit = B'0 (Disables the DAR reload function) • DAF bit, SAF bit = B'00 (Not used) • DO bit = B'0 (Not used) • TL bit = B'0 (Not used) • TEMASK bit = B'1 (Continues the DMA transfer when the TE bit is set) • HE bit, HIE bit = B'00 (Not used) • AM bit, AL bit = B'00 (Not used) • DM [1:0] bits = B'00 (Destination address fixed) • SM [1:0] bits = B'01 (Increments the source address) • RS [3:0] bits = B'1000 (Specifies the DMA extension resource) • DL bit, DS bit = B'00 (Not used) • TB bit = B'0 (Specifies the cycle steal mode) • TS [1:0] bits = B'10 (Specifies the longword transfer) • IE bit = B'1 (Enables an interrupt request) • DE bit = B'0 (Disables the DMA transfer)
		H'2010 1815	• DE bit = B'1 (Enables the DMA transfer)
DMA source address register 1 (SAR_1)	H'FFFE 1010	Internal RAM	• Specifies the internal RAM as the DMA transfer source start address
DMA reload source address register 1 (RSAR_1)	H'FFFE 1110	Internal RAM	• Specifies the internal RAM as the DMA reload transfer source start address
DMA destination address register 1 (DAR_1)	H'FFFE 1014	H'FFFF 0818	• Specifies the SSIFTDR register 1 as the DMA transfer destination start address
DMA transfer count register 1 (DMATCR_1)	H'FFFE 1018	H'0000 000A	• Number of transfers: 10
DMA reload transfer count register 1 (RDMATCR_1)	H'FFFE 1118	H'0000 000A	• Number of transfers: 10
DMA operation register (DMAOR)	H'FFFE 1200	H'0001	• CMS [1:0] bits = B'00 (Normal mode) • PR [1:0] bits = B'00 (Channel priority level: Fixed mode 1) • AE bit = B'0 (Clears the address error flag) • NMIF bit = B'0 (Clears the NMI interrupt) • DME bit = B'1 (Enables the DMA transfer on all channels)
DMA extension resource selector 0 (DMARS0)	H'FFFE 1300	H'2700	• Specifies the SSI channel 1 as the transfer request source of the DMA channel 1

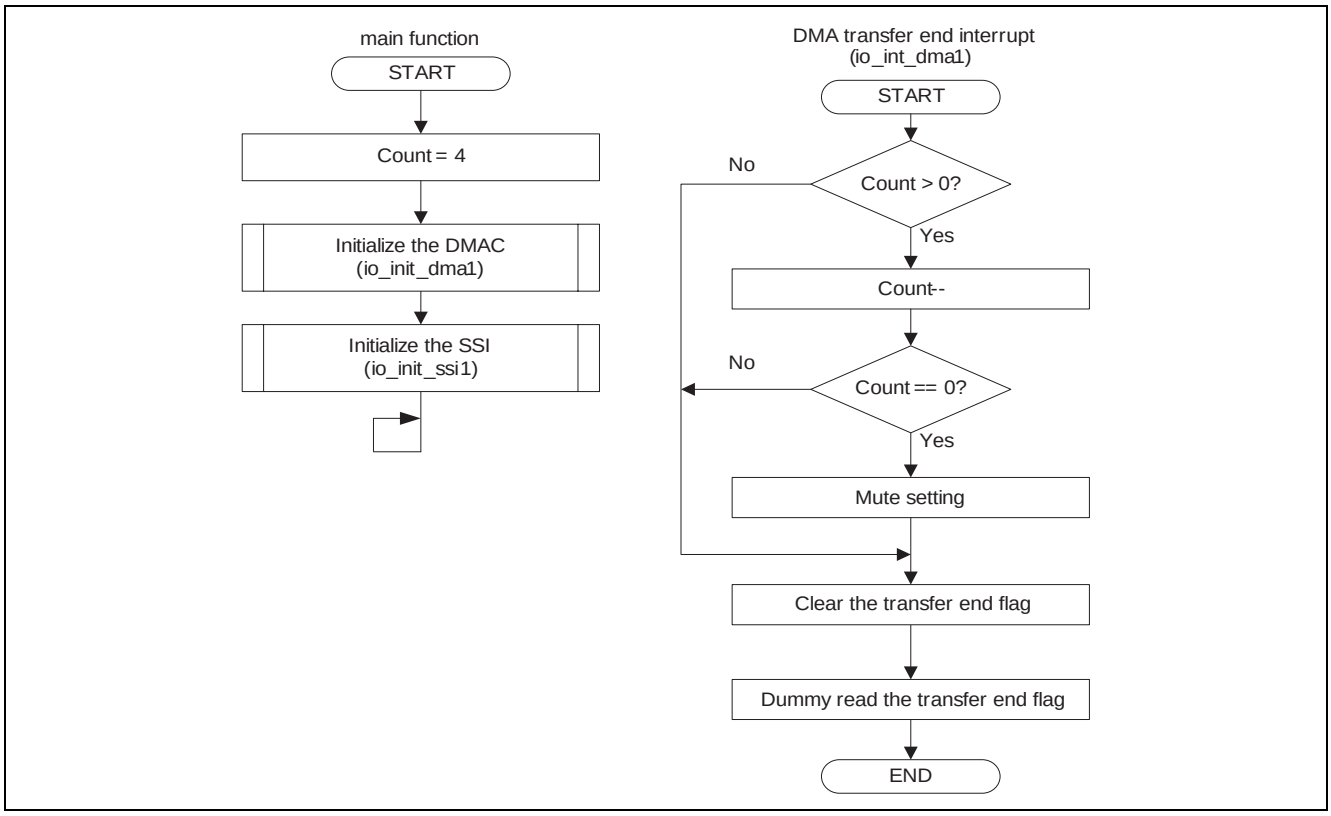

Figure 6 Sample Program Flow Chart

3. Sample Program Listing

3.1 Sample Program Listing "main.c" (1/8)

```
1       /***********************************************************************
2       *   DISCLAIMER
3       *
4       *   This software is supplied by Renesas Electronics Corporation and is only
5       *   intended for use with Renesas products.  No other uses are authorized.
6       *
7       *   This software is owned by Renesas Electronics Corporation and is protected under
8       *   all applicable laws, including copyright laws.
9       *
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 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) 2009(2011) Renesas Electronics Corporation. All rights reserved.
29      *""FILE COMMENT""*********** Technical reference data *************************
30      *   System Name : SH7264 Sample Program
31      *   File Name   : main.c
32      *   Abstract    : SSI in Master Transmitter Mode
33      *   Version     : 1.01.00
34      *   Device      : SH7262/SH7264
35      *   Tool-Chain  : High-performance Embedded Workshop (Ver.4.07.00).
36      *               : C/C++ compiler package for the SuperH RISC engine family
37      *               :                             (Ver.9.03 Release00).
38      *   OS          : None
39      *   H/W Platform: M3A-HS64G50(CPU board)
40      *   Description :
41      ***********************************************************************
42      *   History     : Feb.24,2009 Ver.1.00.00
43      *               : Feb.23,2011 Ver.1.01.00
44      *""FILE COMMENT END""*****************************************************/
45      #include <string.h>
46      #include "iodefine.h"     /* This file is automatically generated by the
47                                   High-performance Embedded Workshop. */
48
```

3.2 Sample Program Listing "main.c" (2/8)

```
49    /* ==== Macro declaration ==== */
50    #define SSI_DATASIZE 40u
51    #define SSI_MUTEDATA 0x00000000ul
52
53    /* ==== Prototype declaration ==== */
54    void main(void);
55    void io_init_ssi1(void);
56    void io_init_dma1(void *src, void *dst, size_t size);
57
58    /* ==== Variable declaration ==== */
59    unsigned long Data[SSI_DATASIZE/sizeof(unsigned long)] = {
60                  0x0000FFFFul,0x0000FFFFul,
61                  0x0000FFFFul,0x0000FFFFul,
62                  0x0000FFFFul,0x0000FFFFul,
63                  0x0000FFFFul,0x0000FFFFul,
64                  0x0000FFFFul,0x0000FFFFul};
65    unsigned int Count;
66
67    /*""FUNC COMMENT""*****************************************************************
68     * ID          :
69     * Outline     : main
70     *-------------------------------------------------------------------------------
71     * Include     : #include "iodefine.h"
72     *-------------------------------------------------------------------------------
73     * Declaration : void main(void);
74     *-------------------------------------------------------------------------------
75     * Description : Initializes the SSI, and transmits data.
76     *-------------------------------------------------------------------------------
77     * Argument    : void
78     *-------------------------------------------------------------------------------
79     * Return Value: void
80     *""FUNC COMMENT END""*************************************************************/
81    void main(void)
82    {
83      Count = 4u;                                       /* Number of the DMA transfers */
84
85      /* ==== Initializes the DMAC/enable the DMA transfer ==== */
86      io_init_dma1(    Data,                            /* Source address */
87                   (void *)&SSIF1.SSIFTDR,              /* Destination address */
88                   SSI_DATASIZE);                    /* Number of bytes */
89
90      /* ==== Initializes the SSI1 ==== */
91      io_init_ssi1();
92
93      while(1){
94        /* Program end */
95      }
96    }
```

3.3 Sample Program Listing "main.c" (3/8)

```
97    /*""FUNC COMMENT""*****************************************************************
98     * ID          :
99     * Outline     : Initialize the SSI
100    *-------------------------------------------------------------------------------
101    * Include     : #include "iodefine.h"
102    *-------------------------------------------------------------------------------
103    * Declaration : void io_init_ssi1(void);
104    *-------------------------------------------------------------------------------
105    * Description : Transfers data in master transmitter mode.
106    *             : The sampling frequency is at 44.1 kHz.
107    *-------------------------------------------------------------------------------
108    * Argument    : void
109    *-------------------------------------------------------------------------------
110    * Return Value: void
111    *""FUNC COMMENT END""*************************************************************/
112   void io_init_ssi1(void)
113   {
114     volatile int w;
115
116     /* ---- Supplies the clock to the SSI ---- */
117     CPG.STBCR6.BIT.MSTP66 = 0u;         /* SSIF1 */
118
119     /* ----Selects the SSI pin functions ---- */
120     PORT.PFCR0.BIT.PF0MD = 2u;          /* SSISCK1 */
121     PORT.PFCR0.BIT.PF1MD = 2u;          /* SSIWS1 */
122     PORT.PFCR0.BIT.PF2MD = 2u;          /* SSIDATA1 */
123
124     /* ---- Specifies the SSI interrupt level ---- */
125     INTC.IPR15.BIT._SSI1 = 1u;          /* SSI1 */
126
127     /* ---- Sets the Control register (SSICR) ---- */
128     SSIF1.SSICR.LONG = 0x300BD520ul;
129       /*
130            bit31       : reserve 0
131            bit30       : CKS : 0-----------------  AUDIO_X1 input
132            bit29       : TUIEN : 1---------------  Enables the transmit underflow interrupt
133            bit28       : TOIEN : 1---------------  Enables the transmit overflow interrupt
134            bit27       : RUIEN : 0---------------  Disables the receive underflow interrupt
135            bit26       : ROIEN : 0---------------  Disables the receive overflow interrupt
136            bit25       : IIEN : 0----------------  Disables the idle mode interrupt
137            bit24       : reserve 0
138            bit23 to 22 : CHNL : B'00-------------  System words have one channel
139            bit21 to 19 : DWL : B'001-------------  Data word length: 16 bits
140            bit18 to 16 : SWL : B'011-------------  System word length: 32 bits
141            bit15       : SCKD : 1----------------  Serial bit clock output, master mode
142            bit14       : SWSD : 1----------------  Serial word WS output, master mode
143            bit13       : SCKP : 0----------------  Outputs at the falling edge of the SSISCK
144            bit12       : SWSP : 1----------------  High level at 1st channel,
145                                                    low level at 2nd channel
146            bit11       : SPDP : 0----------------  Padding bits are low level
147            bit10       : SDTA : 1----------------  Transmits and receives in the order of
148                                                    padding bits, and serial data
```

3.4 Sample Program Listing "main.c" (4/8)

```
149           bit9        : PDTA : 0-----------------  Transmits and receives lower bits of
150                                                    parallel data
151           bit8        : DEL : 1------------------  No delay between the SSIWS and SSIDATA
152           bit7 to 4   : CKDV : B'0010------------  Specifies the oversampling clock as
153                                                    the AUDIO clock/4 (44.1 kHz)
154           bit3        : MUEN : 0-----------------  Not muted
155           bit2        : reserve 0
156           bit1        : TEN : 0------------------  Disables to transmit data
157           bit0        : REN : 0------------------  Disables to receive data
158       */
159     /* ---- Sets the FIFO control register (SSIFCR) ---- */
160     SSIF1.SSIFCR.LONG = 0x00000008ul;
161       /*
162           bit31 to 8  : reserve 0
163           bit7 to 6   : TTRG : B'00----------- Number of transmit data triggers: 7
164           bit5 to 4   : RTRG : B'00----------- Number of receive data triggers: 1
165           bit3        : TIE : 1--------------- Enables the transmit data empty interrupt
166           bit2        : RIE : 0--------------- Disables the receive data full interrupt
167           bit1        : TFRST : 0------------- Disables to reset the transmit FIFO
168                                                data register
169           bit0        : RFRST : 0------------- Disables to reset the receive FIFO
170                                                data register
171       */
172     /* ---- Enables to transmit data ---- */
173     SSIF1.SSICR.BIT.TEN = 1u;
174
175     /* ---- Checks the transmission begins ---- */
176     while(1){
177       /* Wait for 1.5 cycles of SSIWS */
178       for( w = 16000 ; w > 0 ; w--){/* 1.1 ms = (1/44.1 kHz) * 32 bit * 1.5 cyc */
179           /* wait */
180       }
181       /* If the serial bus is running */
182       if( SSIF1.SSISR.BIT.IDST == 0 ){
183           break;
184       }
185       /* Disables to transmit data */
186       SSIF1.SSICR.BIT.TEN = 0u;
187
188       /* Wait for one cycle of SSISCK */
189       for( w = 400 ; w > 0 ; w--){      /* 23 us = 1/44.1 kHz */
190           /* wait */
191       }
192       /* Enables to transmit data */
193       SSIF1.SSICR.BIT.TEN = 1u;
194     }
195
196   }
```

3.5 Sample Program Listing "main.c" (5/8)

```
197   /*""FUNC COMMENT""*****************************************************************
198    * ID          :
199    * Outline     : SSI interrupt
200    *-------------------------------------------------------------------------------
201    * Include     : #include "iodefine.h"
202    *-------------------------------------------------------------------------------
203    * Declaration : void io_int_ssi1(void);
204    *-------------------------------------------------------------------------------
205    * Description : Handles the SSI interrupts.
206    *-------------------------------------------------------------------------------
207    * Argument    : void
208    *-------------------------------------------------------------------------------
209    * Return Value: void
210    *""FUNC COMMENT END""*************************************************************/
211   void io_int_ssi1(void)
212   {
213     /* Transmit underflow error */
214     if(SSIF1.SSISR.BIT.TUIRQ == 1u){
215       SSIF1.SSISR.BIT.TUIRQ = 0u;
216       while(1){
217           /* dead loop */
218       }
219     }
220     /* Transmit overflow error */
221     if(SSIF1.SSISR.BIT.TOIRQ == 1u){
222       SSIF1.SSISR.BIT.TOIRQ = 0u;
223       while(1){
224           /* dead loop */
225       }
226     }
227   }
228
```

3.6 Sample Program Listing "main.c" (6/8)

```
/*""FUNC COMMENT""*************************************************************
 * ID          :
 * Outline     : DMA transfer initialization
 *-----------------------------------------------------------------------------
 * Include     : #include "iodefine.h"
 *-----------------------------------------------------------------------------
 * Declaration : void io_init_dma1(void *src, void *dst, size_t size);
 *-----------------------------------------------------------------------------
 * Description : Transfers the "size" bytes of data from the source address "src" to
 *             : the destination address "dst" by the DMAC.
 *             : When the DMA transfer is complete, it continues to transmit the
 *             : same data. Enables the DMA transfer end interrupt.
 *             : Specifies the transfer size in units of longword, and the SSI1 as
 *             : the transfer destinaion.
 *             : When the transfer size, and source or destination address alignment
 *             : does not match, the operation will not be guaranteed.
 *-----------------------------------------------------------------------------
 * Argument    : void *src   : source address
 *             : void *dst   : destination address
 *             : size_t size : transfer size (in bytes).
 *-----------------------------------------------------------------------------
 * Return Value: void
 *""FUNC COMMENT END""*********************************************************/
void io_init_dma1(void *src, void *dst, size_t size)
{
  /* ---- Sets the Standby control register 2 ---- */
  CPG.STBCR2.BIT.MSTP8 = 0u;             /* DMAC operates */

  /* ---- Disables the DMA transfer ---- */
  DMAC.CHCR1.BIT.DE = 0u;

  /* ---- Sets the DMA source address register ---- */
  /* ---- Sets the DMA reload source address register ---- */
  DMAC.SAR1.LONG = (unsigned long)src;
  DMAC.RSAR1.LONG= (unsigned long)src;

  /* ---- Sets the DMA destination address register ---- */
  DMAC.DAR1.LONG = (unsigned long)dst;

  /* ---- Sets the DMA transfer count register ---- */
  /* ---- Sets the DMA reload transfer count register ---- */
  DMAC.DMATCR1.LONG = size >> 2u;
  DMAC.RDMATCR1.LONG= size >> 2u;

```

3.7 Sample Program Listing "main.c" (7/8)

```
273     /* ---- Sets the DMA channel control register ---- */
274     DMAC.CHCR1.LONG = 0x20101814ul;
275       /*
276           bit31      : TC : 0--------------------Transmits data once by one request
277           bit30      : reserve 0
278           bit29      : RLDSAR : 1----------------Enables the SAR reload function
279           bit28      : RLDDAR : 0----------------Disables the DAR reload function
280           bit27      : reserve 0
281           bit26      : DAF : 0-------------------Not used
282           bit25      : SAF : 0-------------------Not used
283           bit24      : reserve 0
284           bit23      : DO : 0--------------------Not used
285           bit22      : TL : 0--------------------Not used
286           bit21      : reserve 0
287           bit20      : TEMASK : 1----------------Continues the DMA transfer when
288                                                     the TE bit is set
289           bit19      : HE : 0--------------------Not used
290           bit18      : HIE : 0-------------------Not used
291           bit17      : AM : 0--------------------Not used
292           bit16      : AL : 0--------------------Not used
293           bit15 to 14: DM[1:0] : B'00-------------Destination address fixed
294           bit13 to 12: SM[1:0] : B'01-------------Increments the source address
295           bit11 to 8 : RS[3:0] : B'1000------- Specifies the DMA extension resource selector
296           bit7       : DL : 0--------------------Not used
297           bit6       : DS : 0--------------------Not used
298           bit5       : TB : 0--------------------Specifies the cycle steal mode
299           bit4 to 3  : TS : B'10-----------------Specicies the longword transfer
300           bit2       : IE : 1--------------------Enables an interrupt request
301           bit1       : TE : 0--------------------Transfer end flag
302           bit0       : DE : 0--------------------Disables the DMA transfer
303       */
304     /* ----Sets the DMA extension resource selector 0 ---- */
305     DMAC.DMARS0.BIT.CH1MID = 0x09u;         /* MID = SSI1 */
306     DMAC.DMARS0.BIT.CH1RID = 0x03u;         /* RID  */
307
308     /* ----Sets the DMA operation register ---- */
309     DMAC.DMAOR.WORD &= 0xFFF9u;             /* Clears the AE, NMIF bits */
310     DMAC.DMAOR.BIT.DME = 1u;                /* Enables the DMA transfer on all channels */
311
312     /* ---- Sets the interrupt priority level register ---- */
313     INTC.IPR06.BIT._DMAC1 = 1u;
314
315     /* ---- Enables the DMA transfer ---- */
316     DMAC.CHCR1.BIT.DE = 1ul;
317   }
```

3.8 Sample Program Listing "main.c" (8/8)

```
/*""FUNC COMMENT""*****************************************************************
 * ID          :
 * Outline     : DMA transfer end interrupt
 *-----------------------------------------------------------------------------
 * Include     : #include "iodefine.h"
 *-----------------------------------------------------------------------------
 * Declaration : void io_int_dma1(void);
 *-----------------------------------------------------------------------------
 * Description : When the specified number of the DMA transfers are complete,
 *             : the SSI transitions to the mute status.
 *-----------------------------------------------------------------------------
 * Argument    : void
 *-----------------------------------------------------------------------------
 * Return Value: void
 *""FUNC COMMENT END""*************************************************************/
void io_int_dma1(void)
{
  volatile unsigned long dummy;

  if( Count > 0 ){
    /* ---- Counts the number of transfers ---- */
    Count--;

    if( Count == 0 ){
        /* ---- When the specified number of transfers are complete,
                it transitions to mute status (Continues the DMA transfer) ---- */
        SSIF1.SSICR.BIT.MUEN = 1u;     /* Starts to mute */
                                       /*  Replaces the data in FIFO with the muted data */
    }
  }
  /* ---- Clears the transfer end flag ---- */
  DMAC.CHCR1.BIT.TE = 0u;
  dummy = DMAC.CHCR1.BIT.TE;      /* Dummy read */
}
/* End of File */
```

4. References

- Software Manual
  SH-2A/SH2A-FPU Software Manual Rev.3.00
  The latest version of the software manual can be downloaded from the Renesas Electronics website.

- Hardware Manual
  SH7262 Group, SH7264 Group Hardware manual Rev.2.00
  The latest version of the hardware manual can be downloaded from the Renesas Electronics website.

Website and Support

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

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

All trademarks and registered trademarks are the property of their respective owners.

Revision Record

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Apr.14.09	—	First edition issued
1.01	Feb.23.11	5,15	Corresponds to Technical Update (TN-SH7-A799A/E)

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: +49-211-6503-0, 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.
7F, No. 363 Fu Shing North Road Taipei, Taiwan, R.O.C.
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

© 2010 Renesas Electronics Corporation. All rights reserved.

Colophon 1.0