Connecting the Atmel ARM-based Serial Synchronous Controller (SSC) to an I^2S-compatible Serial Bus

Introduction

This Application Note describes the configuration required to connect the Atmel ARM-based Synchronous Serial Controller (SSC) to a device with an I^2S-compatible serial bus, such as a stereo audio digital-to-analog converter (DAC) or a stereo audio Codec.

The digital interface of these audio devices is generally compliant with the I^2S standard. An I^2S-standard compliant device has a word length of 16 bits, as does the SSC peripheral embedded in the AT91RM9200 series microcontrollers.

I^2S (Inter-IC Sound) is a serial bus designed for digital audio devices and technologies, such as compact disc (CD) players, digital sound processors, and digital TV (DTV) sound. One of the characteristics of the I^2S protocol is the separate handling of audio data and clock signals. Separating the data and clock signals eliminates the need for anti-jitter devices by removing time-related errors.

The Application Note includes a dedicated software package for the AT91RM9200, but is applicable to all Atmel ARM-based products that embed the Synchronous Serial Controller (SSC).

This application note takes into account the SSC warnings as described in the AT91RM9200 Errata Sheet, literature number 6015.

Warranty

All delivered sources are free of charge and can be copied or modified without authorization.

The software is delivered "AS IS" without warranty or condition of any kind, either express, implied or statutory. This includes without limitation any warranty or condition with respect to merchantability or fitness for any particular purpose, or against the infringements of intellectual property rights of others.

AT91 ARM® Thumb® Microcontrollers

Application Note

6020A–ATARM–11/03

I²S Audio Bus

The I²S (Inter-IC Sound) standard is based on a three-wire bus architecture. This standard defines a serial link dedicated to data transfer between integrated circuits in digital audio systems. This three-wire link provides additional information to audio data, such as subcoding and control, transferred separately. The three lines defined by the I²S protocol are:

- a Serial Data (SD) line containing two time-division multiplexed channels
- a left/right channel Word Select (WS)
- a continuous Serial Clock (SCK)

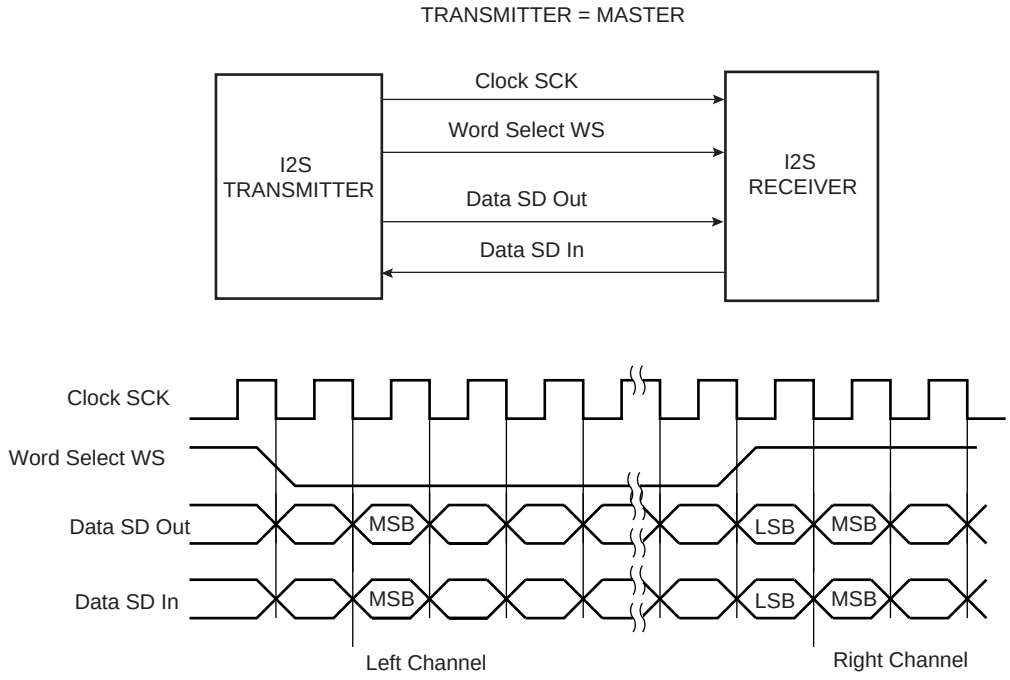
The I²S link is used primarily to send audio data from a processor(master) to an Audio DAC (slave). The three lines driven by an I²S master transmitter are

- a Serial Data Out (SD Out) for two time-division multiplexed channels (from master to slave)
- a left/right channel Word Select (WS)
- a continuous Serial Clock (SCK); the I²S master (transmitter) and slave (receiver) share the same clock signal for data transmission

An additional line can be used to connect an I²S slave input signal (such as an audio Codec):

- a Serial Data In (SD In) for two time-division multiplexed channels (from slave to master)

Figure 1. I²S Block Diagram Example



I²S Word Length Considerations

The I²S standard defines several possible data (word) lengths from 16 to 32 bits. For most audio applications, the data length is 16 bits, corresponding to a dynamic range of 96 dB.

Note: The dynamic range is given by the following formula:
Dynamic Range = 20 log(Bit Range) where bit range = 2^{bit}

Table 1. Bit Range versus Dynamic Range

Bit Range	Dynamic Range
2^{16}	96 dB
2^{24}	144 dB
2^{32}	196 dB

Only 16-bit words are managed by the SSC due to the maximum value of the field FSLEN of the Frame Mode Register.

I²S Clock Considerations

The sampling frequency of audio devices can vary from 8 to 48 kHz for 16-bit data.

To generate data with the correct bit rate, the SSC peripheral divides its internal peripheral clock (MCK) by an integer factor. Table 2 indicates the error on audio clock signal frequency (related to the peripheral main clock) compared to the ideal value. Due to this uncertainty, the peripheral clock value and thus the crystal frequency must be chosen carefully with respect to the limitations of the components.

The I²S bit rate determines data flow on the I²S bus and I²S clock signal frequency:

I2S Bit Rate = Number Bits per Channel × Number of Channels × Sampling Frequency

For 16-bit audio, left/right, the I²S bit rate is calculated as follows:

I2S Bit Rate = 16 × 2 × Sampling Frequency

The MCK divider factor value is set in the SSC_CMR register and equals half of the peripheral clock frequency (MCK) divided by the required bit rate. If this factor is not an integer, the real bit rate (generated by the SSC) is different from the theoretical one.

Table 2 gives the error between theoretical and real values of audio sample frequency. It is important to note that the difference between two frequency levels may result in distortion on the audio output signal.

Table 2. Using a 60 MHz MCK Clock

Theoretical Audio Sample Frequency (Hz)	Theoretical Bit Rate	Theoretical MCK Divider Factor	SSC_CMR (Real MCK Divider Factor)	Audio Sample Frequency (Hz)
48 000	1 536 000	39.06250	20 (40)	46 875
44 100	1 411 200	42.51701	21 (42)	44 643
22 050	705 600	85.03401	43 (86)	21 802
16 000	512 000	117.18750	59 (118)	15 890
8 000	256 000	234.37500	117 (234)	8 013

Connection to a Stereo Audio DAC

The following implementation using the I^2S bus illustrates the DAC connection as audio output of an ARM-based microcontroller embedding the SSC and TWI peripherals.

The SSC is used and connected as an I^2S processor, sending 16-bit words, a word-select signal and the serial clock. The standard audio DAC, only used as audio output, is connected as an I^2S slave.

Figure 2. SSC Audio DAC Connection

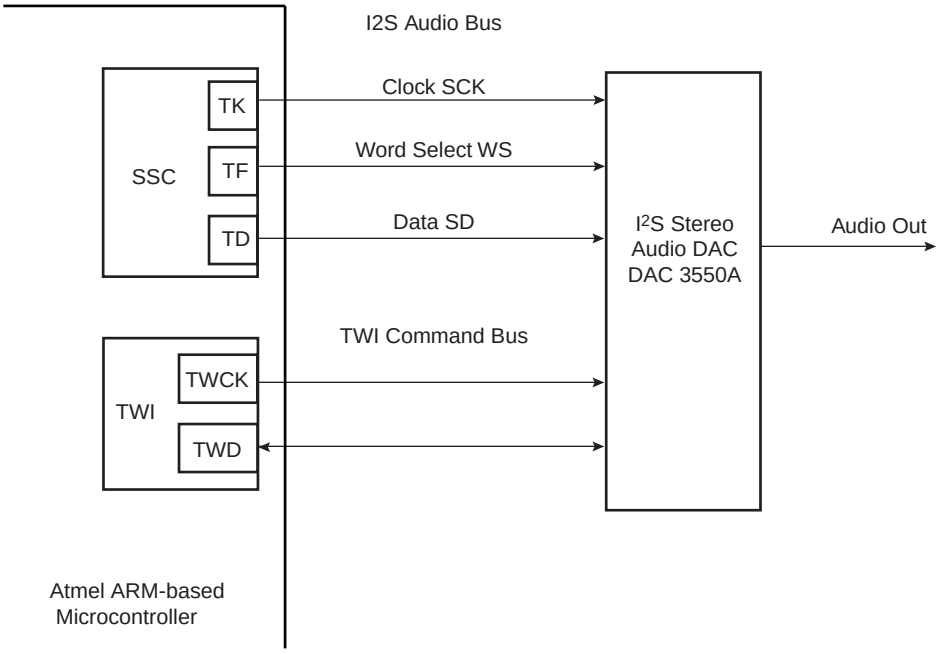


Table 3 gives the hardware connections between the microcontroller and a Micronas DAC 3550A:

Table 3. Hardware Connections

Microcontroller	DAC 3550A	Bus Name
SSC (TFx)	WSI	WS (I^2S)
SSC (TKx)	CLI	SCK (I^2S)
SSC (TDx)	DAI	SD (I^2S)
TWI (TWCK)	SCL	SCL
TWI (TWD)	SDA	SDA

Configuring the SSC

The following example illustrates the use of one of the SSC peripherals embedded in the AT91RM92000, the SSC1.

The related SSC I/O lines of the Parallel Input/Output Controller (PIO) are configured in peripheral mode. When using a DAC, only the three lines of the SSC emitter are used because the SSC is used as an I^2S master only for emission.

The software is configured according to the following general characteristics:

- Sample audio frequency (FILE_SAMPLING_FREQ): 44.1 kHz
- Number of slots by frame (SLOT_BY_FRAME): 2 (left and right channels)
- Number of bits by slot (BITS_BY_SLOT): 16 (data length is 16 bits)
- SSC peripheral clock (MCK): 60 MHz

Note: All parameters and functions are compliant with the ARM-based Software Packages Application Note, lit. no. 6016.

Standard Initialization

The following configuration steps are common to all serial peripherals:

1. Configure the corresponding Parallel Input/Output Controller to work in peripheral mode, i.e., the three PIO lines TF, TK and TD related to the SSC1 and multiplexed with PIOB must not be in PIO management mode.

```
*AT91C_PIOB_PDR= ((unsigned int) AT91C_PB7_TK1 ) |
((unsigned int) AT91C_PB8_TD1 ) | ((unsigned int) AT91C_PB6_TF1 );
```

2. Configure the Power Management Controller to enable the current peripheral and set the PMC by enabling the SSC1 clock.

```
AT91F_SSC1_CfgPMC();
```

3. Reset the SSC1:

```
pSSC->SSC_CR = AT91C_SSC_SWRST ;
```

4. Clear Transmit and Receive PDC Counters:

```
AT91F_PDC_Close((AT91PS_PDC) &(pSSC->SSC_RPR));
```

Configuring the Clock Mode Register (SSC_CMR)

The definition of the Clock Mode Register is done for the characteristics described above (sample audio frequency, number of slots by frame, data length and peripheral clock frequency).

This setting is managed by the standard software package.

```
function AT91F_SSC_SetBaudrate:
  Bit rate= SLOT_BY_FRAME*BITS_BY_SLOT*FILE_SAMPLE_FREQ
       = 2*16*44100
       = 1.4112 MHz
  AT91F_SSC_SetBaudrate(
    pSSC,MCK, FILE_SAMPLING_FREQ*(BITS_BY_SLOT*SLOT_BY_FRAME));
```

Configuring the Transmit Frame Mode Register (SSC_TFMR)

The Transmit Frame Mode register is used to manage both TF (corresponding to the Word Select line) and TD (corresponding to the data line) signals.

Table 4. Transmit Frame Mode Register (SSC_TFMR) Settings

Field Name	Value	Comments
DATLEN	BITS_BY_SLOT-1	Programs Data Length
DATDEF	0	Default value for data. For compatibility with the I^2S protocol, all bits are set to zero when not sending data.
MSBF	1	Most Significant Bit First. MSB is sent first.
DATNB	SLOT_BY_FRAME - 1	Data Number per frame. Programs the number of data contained in one frame.
FSLEN	BITS_BY_SLOT - 1	Frame Sync LENgth. Programs the duration of the active level on TF in number of serial clock cycles. The Word Select signal must be as long as one data word.
FSOS	Negative Pulse	Frame Sync Output Selection
FSDEN	Disabled	Frame Sync Data ENable. No Sync Data (SSC_TSHR) to emit.
FSEDGE	Positive Edge Detection	Frame Sync Edge Detection Not used in this case.

```
 *AT91C_SSC1_TFMR = (AT91C_SSC_FSOS_NEGATIVE |
(((BITS_BY_SLOT-1)<<16) & AT91C_SSC_FSLEN)  |
(((SLOT_BY_FRAME-1)<<8) & AT91C_SSC_DATNB) |
 AT91C_SSC_MSBF | (BITS_BY_SLOT-1) ;
```

Configuring Interrupt Mode

In this application, the Peripheral Data Controller (PDC) interrupt sources are handled to loop the sending of the wave file to the DAC.

1. Configure the Advanced Interrupt Controller (AIC) to handle SSC interrupts:

```
AT91F_AIC_ConfigureIt (
   AT91C_BASE_AIC,                         // AIC base address
   AT91C_ID_SSC1,                          // System peripheral ID
   IRQ_LEVEL_I2S,                          // Max priority
   AT91C_AIC_SRCTYPE_INT_LEVEL_SENSITIVE, // Level sensitive
   AT91F_ASM_I2S_Handler );
```

2. Enable the SSC interrupt in AIC:

```
AT91F_AIC_EnableIt(AT91C_BASE_AIC, AT91C_ID_SSC1);
```

3. Enable SSC End Of Transmit interrupt in the SSC Interrupt Enable Register:

```
AT91F_SSC_EnableIt (pSSC, AT91C_SSC_ENDTX);
```

Configuring the PDC

1. Configure the PDC:

```
AT91F_PDC_SetTx ((AT91PS_PDC) &(pSSC->SSC_RPR),
        (char *)wav_file, AT91C_WAV_FILE_SIZE/2);
AT91F_PDC_SetNextTx ((AT91PS_PDC) &(pSSC->SSC_RPR),
        (char *)wav_file, AT91C_WAV_FILE_SIZE/2);
```

2. Enable the PDC feature:

```
AT91F_PDC_EnableTx ((AT91PS_PDC) &(pSSC->SSC_RPR));
```

Enabling the Peripheral and Starting Transmission

The Transmit Clock Mode Register must be configured before enabling and starting transmission. Then, when transmission is enabled, it starts automatically.

Table 5. Transmit Clock Mode Register (SSC_TFMR) Settings

Field Name	Value	Comments
CKS	Divided Clock	Transmit Clock Selection. Internally generated divided clock (from peripheral clock MCK) is selected.
CKO	Continuous Transmit Clock	Clock Output Mode
CKI	Shifting out on serial clock falling edge	Transmit Clock Inversion. The data and the frame sync signals are shifted out on transmit clock falling edge.
START	Falling edge on TF signal	Transmit Start Selection. The first word to be transferred is left channel.
STTDLY	1	Transmit Start Delay. First bit after falling edge of TF signal is the last bit of right channel.
PERIOD	((SLOT_BY_FRAME *BITS_BY_SLOT)/2) - 1	Transmit Period Divider Selection. Frame Length, including left and right channel

```
*AT91C_SSC1_TCMR =  ((((BITS_BY_SLOT*SLOT_BY_FRAME)/2) -1) <<24) |
      ((1<<16) & AT91C_SSC_STTDLY) | AT91C_SSC_START_FALL_RF |
      AT91C_SSC_CKO_CONTINOUS | AT91C_SSC_CKS_DIV);
```

1. Enable the TX transmitter:

```
AT91F_SSC_EnableTx (pSSC);
```

Connection to a Stereo Audio Codec

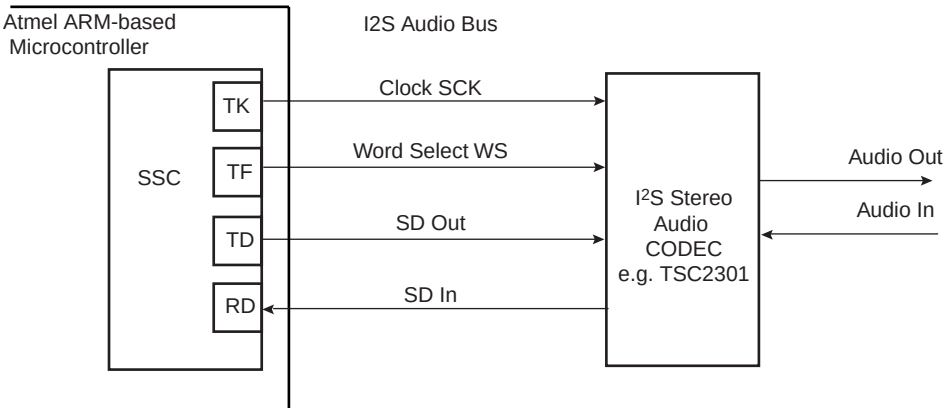
The following examples illustrate the use of one or two SSC peripherals in the AT91RM9200 connecting to a stereo audio Codec. The SSC peripheral in the AT91RM9200 is able to manage two channels in output mode and one channel in input mode. This is sufficient for most applications using a Codec as generally only one channel is required in input mode. Hardware and software configurations related to this type of application are described in “Configuration for Stereo Output and Mono Input” on page 8.

When the two channels are managed in input mode (i.e., all received frame bits), the AT91RM9200 requires two SSC channels. Hardware and software implementation related to this type of application is described in “Configuration for Stereo Input and Output” on page 10.

Configuration for Stereo Output and Mono Input

To connect the Codec as stereo output and mono channel input (only left channel), the following implementation using I^2S with a 16-bit data format can be used. The standard audio Codec is used in both output and input modes. The SSC, used in master mode, manages all I^2S transmit signals.

Figure 3. Stereo Audio Codec with One Input Channel



Configuring the SSC

The following configuration is an example using the AT91RM9200 and one of its SSC peripherals (SSC1).

The Parallel Input/Output Controller (PIO) is configured in peripheral mode for the corresponding SSC I/O lines.

Configuring the Receive Frame Mode Register

The Receive Frame Mode Register (SSC_RFMR) is configured to manage data reception.

Table 6. Receive Frame Mode Register (SSC_RFMR) Settings

Field Name	Value	Comments
DATLEN	BITS_BY_SLOT - 1	Programs Data Length
LOOP	NO LOOP	Loop Mode. For compatibility with I^2S protocol, all bits are set to zero when not sending data.
MSBF	1	Most Significant Bit First. MSB is received first.

Table 6. Receive Frame Mode Register (SSC_RFMR) Settings (Continued)

Field Name	Value	Comments
DATNB	0 (1 slot per frame)	Data Number per frame. Programs the number of data contained in one frame.
FSLEN	0 (Default value because not used)	Frame Sync LENgth.
FSOS	Not used	Frame Sync Output Selection
FSDEN	Disabled	Frame Sync Data ENable. No Sync Data (SSC_RSHR) to receive.
FSEDGE	Positive Edge Detection (Default value)	Frame Sync Edge Detection Not used in this case.

```
*AT91C_SSC1_RFMR = AT91C_SSC_MSBF | (BITS_BY_SLOT-1) ;
```

Configuring the Receive Clock Mode Register

Table 7. SSC1 Receive Clock Mode Register (SSC1_RCMR) Settings

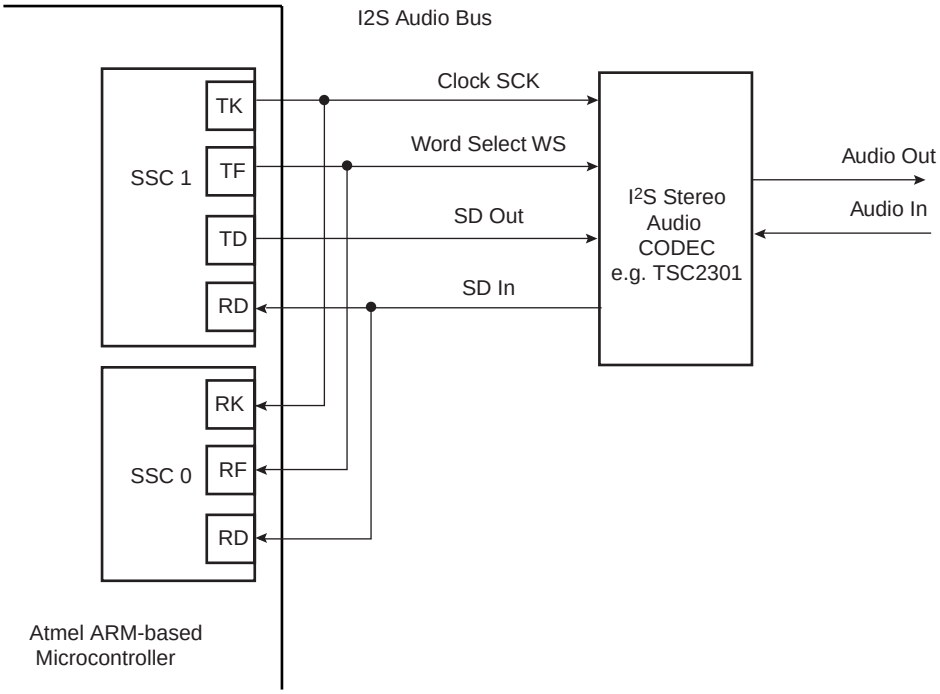
Field Name	Value	Comments
CKS	TK Clock Signal	Receive Clock Selection
CKO	0 (Default value)	Clock Output Mode Selection. Not Used.
CKI	Sampling on rising edge of serial clock	Receive Clock Inversion
START	Transmit Start	Receive Start Selection
STTDLY	1	Receive Start Delay. First bit after the falling edge is the last bit of right channel.
PERIOD	0	Receive Period Divider Selection. Not used

```
*AT91C_SSC1_RCMR = AT91C_SSC_CKS_TK | AT91C_SSC_START_TX |
((1<<16) & AT91C_SSC_STTDLY) | 0x1 << 8 | AT91C_SSC_CKI );
```

Configuration for Stereo Input and Output

To connect the Codec as stereo input and output (both channels received), the following implementation using I^2S with a 16-bit data format can be used. The standard audio Codec is used in input and output modes. The first SSC peripheral, used in master mode, manages all I^2S transmit signals and left-channel input. The second SSC peripheral manages right-channel input only and is synchronized with the first SSC peripheral.

Figure 4. Stereo Audio Codec with Two Input Channels



Configuring the SSC1

The configuration and settings for SSC1 are the same as those described in “Configuration for Stereo Output and Mono Input” on page 8.

Configuring the SSC0

The SSC0 must be configured to receive the second stereo input channel.

1. Configure the PMC by enabling the SSC0 clock:

```
    *AT91C_PMC_PCER |= 1<< AT91C_ID_SSC0;    /* enable the SSC0
peripheral clock*/
```

2. Reset the SSC0 Peripheral:

```
    *AT91C_SSC0_CR = AT91C_SSC_SWRST ;
```

3. Configure the SSC0 Receive Frame Mode Register. The Receive Frame Mode Register (RFMR) is configured to manage data reception.

Table 8. SSC0 Receive Frame Mode Register (SSC0_RFMR) Settings [1]

Field Name	Value	Comments
DATLEN	BITS_BY_SLOT - 1	Programs Data Length
LOOP	NO LOOP	Loop Mode. For compatibility with I^2S protocol, all bits are set to zero when not sending data.
MSBF	1	Most Significant Bit First. MSB is received first.
DATNB	0 (1 slot per frame)	Data Number per frame. Programs the number of data contained in one frame.
FSLEN	0 (Default value because not used)	Frame Sync LENgth.
FSOS	Not used	Frame Sync Output Selection
FSDEN	Disabled	Frame Sync Data ENable. No Sync Data (SSC_RSHR) to receive.
FSEDGE	Positive Edge Detection (Default value)	Frame Sync Edge Detection Not used in this case.

1. Settings are identical to those of SSC1.

```
    *AT91C_SSC0_RFMR = AT91C_SSC_FSOS_NONE  | AT91C_SSC_MSBF | | (16-
1);
```

Table 9. SSC0 Receive Clock Mode Register (SSC0_RCMR) Settings

Field Name	Value	Comments
CKS	RK Clock Signal	Receive Clock Selection
CKO	0 (Default value)	Clock Output Mode Selection. Not Used.
CKI	Sampling on rising edge of serial clock	Receive Clock Inversion
START	On rising edge of RF	Receive Start Selection. Receiver of SSC0 must get the right channel while SSC1 is receiving the left channel. Start edge condition is inverted with respect to SSC1.
STTDLY	1	Receive Start Delay. First bit after the falling edge is the last bit of right channel.
PERIOD	0	Receive Period Divider Selection. Not used

```
*AT91C_SSC0_RCMR = AT91C_SSC_CKS_RK |
AT91C_SSC_CKO_NONE | AT91C_SSC_CKI |
AT91C_SSC_START_RISE_RF | ((1<<16) & AT91C_SSC_STTDLY) ;
```

4. Enable SSC0 receiver RX:

```
*AT91C_SSC0_CR = AT91C_SSC_RXEN;    /* Enable Tx */
```

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

Regional Headquarters

Europe
Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
Tel: (41) 26-426-5555
Fax: (41) 26-426-5500

Asia
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimshatsui
East Kowloon
Hong Kong
Tel: (852) 2721-9778
Fax: (852) 2722-1369

Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Atmel Operations

Memory
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

Microcontrollers
2325 Orchard Parkway
San Jose, CA 95131, USA
Tel: 1(408) 441-0311
Fax: 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
Tel: (33) 2-40-18-18-18
Fax: (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards
Zone Industrielle
13106 Rousset Cedex, France
Tel: (33) 4-42-53-60-00
Fax: (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR, Scotland
Tel: (44) 1355-803-000
Fax: (44) 1355-242-743

RF/Automotive
Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
Tel: (49) 71-31-67-0
Fax: (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
Tel: 1(719) 576-3300
Fax: 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/
High Speed Converters/RF Datacom
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
Tel: (33) 4-76-58-30-00
Fax: (33) 4-76-58-34-80

Literature Requests
www.atmel.com/literature

Disclaimer: Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

6020A–ATARM–11/03