OMAP5910 Dual-Core Processor Multichannel Serial Interface (MCSI) Reference Guide

Literature Number: SPRU685A
March 2004

OMAP™
TEXAS INSTRUMENTS TECHNOLOGY

TEXAS INSTRUMENTS

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Preface

Read This First

About This Manual

Multichannel serial interfaces (MCSIs) have multichannel transmission capability and expand the parallel interface of a DSP to connect to external devices such as codecs and GSM system simulators.

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.

Related Documentation From Texas Instruments

The following documents describe the OMAP5910 device and related peripherals. Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

OMAP5910 Dual-Core Processor Data Manual (literature number SPRS197)

OMAP5910 Dual-Core Processor Silicon Errata (literature number SPRZ016)

OMAP5910 Dual-Core Processor MPU Subsystem Reference Guide (literature number SPRU671)

OMAP5910 Dual-Core Processor DSP Subsystem Reference Guide (literature number SPRU672)

OMAP5910 Dual-Core Processor Memory Interface Traffic Controller Reference Guide (literature number SPRU673)

OMAP5910 Dual-Core Processor System DMA Controller Reference Guide (literature number SPRU674)

OMAP5910 Dual-Core Processor LCD Controller Reference Guide (literature number SPRU675)

OMAP5910 Dual-Core Processor Universal Asynchronous Receiver/Transmitter (UART) Devices Reference Guide (literature number SPRU676)

OMAP5910 Dual-Core Processor Universal Serial Bus (USB) and Frame Adjustment Counter (FAC) Reference Guide (literature number SPRU677)

OMAP5910 Dual-Core Processor Clock Generation and System Reset Management Reference Guide (literature number SPRU678)

OMAP5910 Dual-Core Processor General-Purpose Input/Output (GPIO) Reference Guide (literature number SPRU679)

OMAP5910 Dual-Core Processor MMC/SD Reference Guide (literature number SPRU680)

OMAP5910 Dual-Core Processor Inter-Integrated Circuit (I2C) Controller Reference Guide (literature number SPRU681)

OMAP5910 Dual-Core Processor Timer Reference Guide (literature number SPRU682)

OMAP5910 Dual-Core Processor Inter-Processor Communication Reference Guide (literature number SPRU683)

OMAP5910 Dual-Core Processor Camera Interface Reference Guide (literature number SPRU684)

OMAP5905 Dual-Core Processor Multichannel Serial Interface (MCSI) Reference Guide (literature number SPRU685)

OMAP5910 Dual-Core Processor Micro-Wire Interface Reference Guide (literature number SPRU686)

OMAP5910 Dual-Core Processor Real-Time Clock (RTC) Reference Guide (literature number SPRU687)

OMAP5910 Dual-Core Processor HDQ/1-Wire Interface Reference Guide (literature number SPRU688)

OMAP5910 Dual-Core Processor PWL, PWT, and LED Peripheral Reference Guide (literature number SPRU689)

OMAP5910 Dual-Core Processor Multichannel Buffered Serial Port (McBSP) Reference Guide (literature number SPRU708)

Trademarks

Contents

Figures

Tables

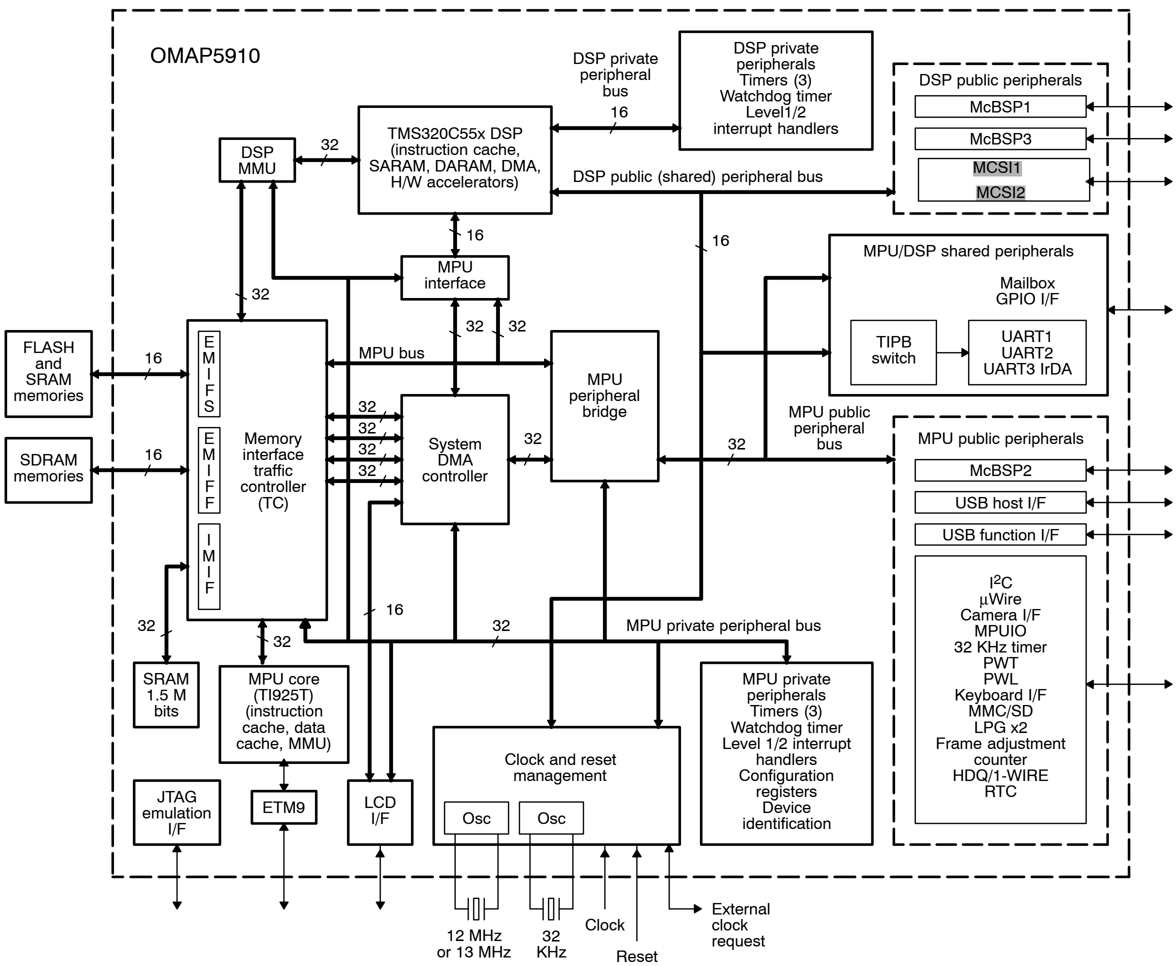
(This page has been left blank intentionally.)

Multichannel Serial Interface (MCSI)

1 Introduction and Overview

Multichannel serial interfaces (MCSIs) have multichannel transmission capability and expand the parallel interface of a DSP to connect to external devices such as codecs and GSM system simulators.

Figure 1. OMAP5910 Functional Overview



The two public MCSIs shown at the top right-corner on Figure 1 under DSP public peripherals provide full duplex transmission and master/slave clock control. All transmission parameters are configurable to cover the maximum number of operating conditions:

- Master/slave clock control (transmission clock and frame synchronization pulse)
- Programmable transmission clock frequency
- Single-channel or multichannel (x16) frame structure
- Programmable word length: 3 to 16 bits
- Full-duplex transmission
- Programmable frame configuration
  - Continuous or burst transmission
  - Normal or alternate framing
  - Normal or inverted frame polarity
  - Short or long frame pulse
  - Programmable oversize frame length
  - Programmable frame length
- Programmable interrupt occurrence time (TX and RX)
- Error detection with interrupt generation on wrong frame length
- DMA support for both TX and RX data transfers

1.1 Communication Protocol

1.1.1 Configuration Parameters

The configuration parameters can be modified only if the MCSI is disabled (control_reg[0] = 0).

1.1.1.1 Slave/Master Control

Using the control bit, the interface can be configured in one of two ways:

- In the master mode with the transmission clock and the frame synchronization pulse generated by the interface
- In the slave mode with the transmission clock and the frame synchronization pulse generated from an external device

Control bit:
MAIN_ PARAMETERS _REG(6) = MCSI_MODE
1: Master
0: Slave

1.1.1.2 Single-Channel/Multichannel

The frame structure can be single-channel-based (one-channel-per-frame) or multichannel, with the number of channels fixed at 16.

Control bit:
MAIN_ PARAMETERS _REG(7) = MULTI
1: Multichannel
0: Single-channel

1.1.1.3 Short/Long Framing

The frame-synchronization pulse duration can be either short (with a pulse duration equal to the bit duration) or long (with a pulse duration equal to the channel duration).

The long frame is active only during transmission on channel 0.

Control bit:
MAIN_ PARAMETERS _REG(8) = FRAME_SIZE
1: Long
0: Short

1.1.1.4 Normal/Alternate Frame Synchronization

The frame-synchronization pulse position is normal with the frame-synchronization pulse starting one bit before channel 0 or it alternates with the frame-synchronization pulse starting with the first bit of channel 0.

Control bit:
MAIN_ PARAMETERS _REG(9) = FRAME_POSITION
1: Alternate
0: Normal

1.1.1.5 Continuous/Burst Mode

The frame mode is continuous with one frame-synchronization pulse at the first frame or operates in bursts with one frame-synchronization pulse at each frame.

Control bit:
MAIN_ PARAMETERS _REG(5) = FRAME_MODE
1: Continuous
0: Burst

1.1.1.6 Normal/Inverted Clock

The polarity of the clock can be normal when writing on positive edge clock and reading on negative edge clock or inverted when writing on negative edge clock and reading on positive edge clock.

Control bit:
MAIN_ PARAMETERS _REG(4) = CLOCK_POLARITY
1: Inverted
0: Normal

1.1.1.7 Normal/Inverted Frame Synchronization

The polarity of the frame-synchronization pulse can be normal with a positive pulse or inverted with a negative pulse.

Control bit:
MAIN_ PARAMETERS _REG(10) = FRAME_POLARITY
1: Inverted
0: Normal

1.1.1.8 Channel Used

To enable a channel in multimode, set bit n for the desired channel n.

1.1.1.9 Word Size

To choose the size of the word, it's size (minus one) parameter must be set in the main parameter registers.

Control bit:
MAIN_PARAMETERS_REG(3:0) = WORD_ SIZE
(2 <= WORD_ SIZE <= 15)

The MCSI transmits and receives the most significant bit first. For example, if the word_size equals 11, the upper 12 bits of the TX registers are transmitted, the upper 12 bits of the RX registers contain the received data, and the lower 4 bits are zeros.

1.1.1.10 Frame Size

To add any overhead bits at the end of each frame, the number of desired overhead bits in the over_size_register must be set.

Control bit:
OVER_CLOCK_REG(9:0) = OVER_CLK (0<=OVER_CLK <=1023)

1.1.1.11 Transmission Clock Frequency

In master mode, the clock frequency is derived from the 12-MHz master clock and can be programmed from 5.8 kHz to 6 MHz in increments of 83 ns.

Control bit:
CLOCK_FREQUENCY_REG(10:0) = CLK_FREQ
(2<=CLK_FREQ <= 2047)

with
($t_{CLK} = t_{12MHz}$ * CLK_FREQ)

1.1.2 Sample Setup for Communication μ-Law Interface Using Interrupts

1.1.2.1 MCSI Configuration

The following is an example of communication μ-law interface setup using interrupts:

- DSP_Write(0x0000) = CONTROL_REG (disa/ble MCSI for setup)
- DSP_Write(0x0007) = MAIN_PARAMETERS_REG (set up MCSI per configuration below)
  - Bit 15-14 (00b): No DMA
  - Bit 10 (0b): Positive polarity for frame
  - Bit 9 (0b): Normal synchronization mode
  - Bit 8 (0b): Short framing
  - Bit 7 (0b): Single channel
  - Bit 6 (0b): Slave mode
  - Bit 5 (0b): Burst mode
  - Bit 4 (0b): Positive edge for clock
  - Bit 3–0 (0111b): 8-bit data
- DSP_Write(0x0700) = INTERRUPTS_REG (all interrupts are enabled)
- DSP_Write(0x0000) = OVER_CLOCK_REG
- DSP_Write(0x0001) = CONTROL_REG (start MCSI)

1.1.2.2 Transmit Data Loading (TX_INT ISR)

- DSP_Write = TX_REG

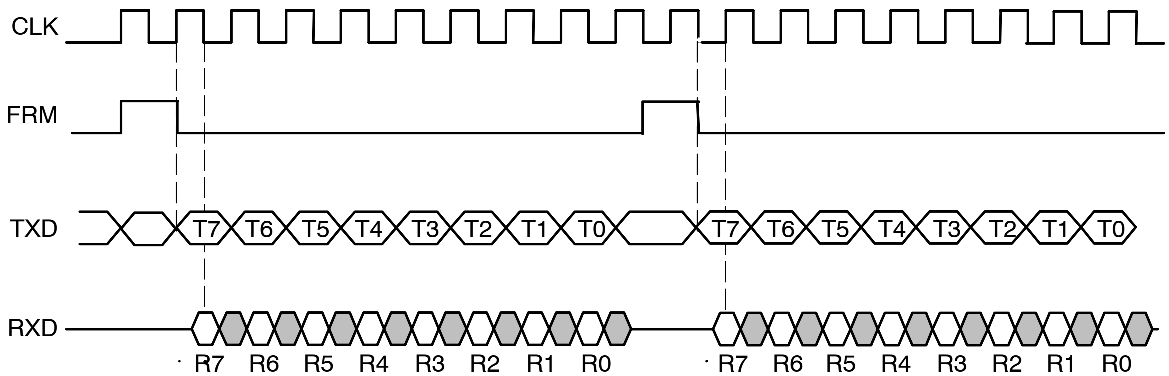
1.1.2.3 Received Data Loading (RX_INT ISR)

- DSP_Read = RX_REG

1.1.2.4 Stop MCSI

- DSP_Write(0x0000) = CONTROL_REG (disable MCSI clock)
- DSP_Write(0x0002) = CONTROL_REG (reset MCSI registers)

Figure 2. Communication μ-Law Interface Interrupts Waveform Example



1.1.3 Interface Management

1.1.3.1 Interrupt Generation

Three physical interrupts are available for real-time management of the MCSI by the DSP:

- RX_INT (data receive interrupt)
- TX_INT (data transmit interrupt)
- FERR_INT (frame duration error interrupt)

RX_INT, TX_INT, and FERR_INT are maskable with dedicated programmable control bits in the interrupt register INTERRUPTS_REG.

- RX_INT is masked when MASK_IT_RX = 0.
- TX_INT is masked when MASK_IT_TX = 0.
- FERR_INT is masked when MASK_IT_ERROR = 0.

Each interrupt is associated with a flag bit in the STATUS_REG register that is set to 1 when the interrupt is generated. To acknowledge the interrupt and release the corresponding physical signal, the DSP must write a 1 at the bit location in the status register. The following list provides interrupt/flag bit associations:

- RX_INT (RX_READY flag and acknowledge bit)
- TX_INT (TX_READY flag and acknowledge bit)
- FERR_INT (FRAME _ERROR flag and acknowledge bit)

1.1.3.2 Receive Interrupt

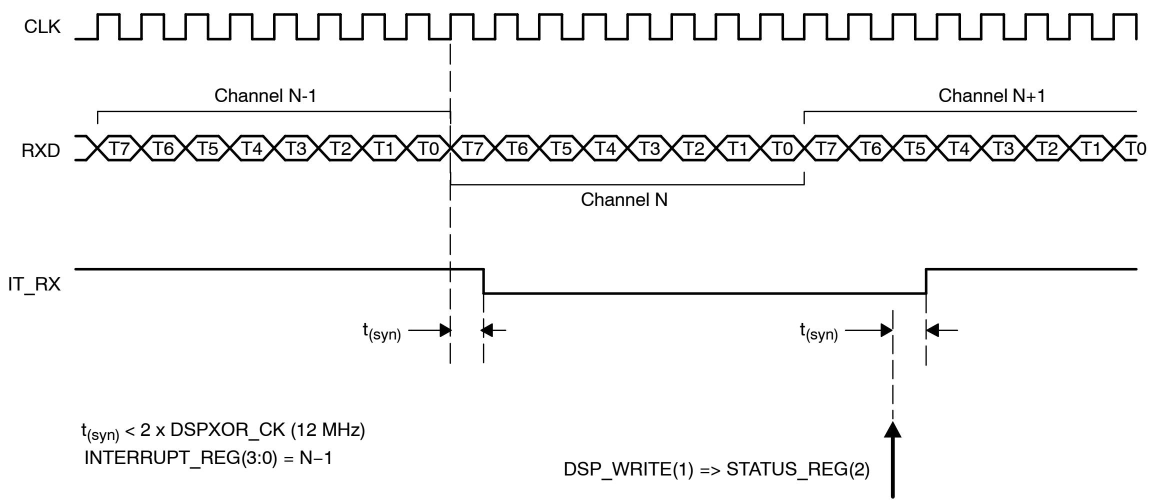
The receive interrupt is generated every frame after the completion of the reception of a data word:

- In single-channel mode, the interrupt is generated one half-clock period (plus a synchronization delay) after the reception of the word.
- In multichannel mode, the interrupt is generated one half-clock period (plus a synchronization delay) after the reception of the word of the channel whose number is defined by the NB_CHAN_IT_RX parameter of INTERRUPTS_REG register.

Note:

If MCSI is in slave mode, the clock must be driven after valid data reception until the interrupt is generated and must not be gated before then, because the interrupt is generated on the MCSI interface clock.

Figure 3. Receive Interrupt Timing Diagram

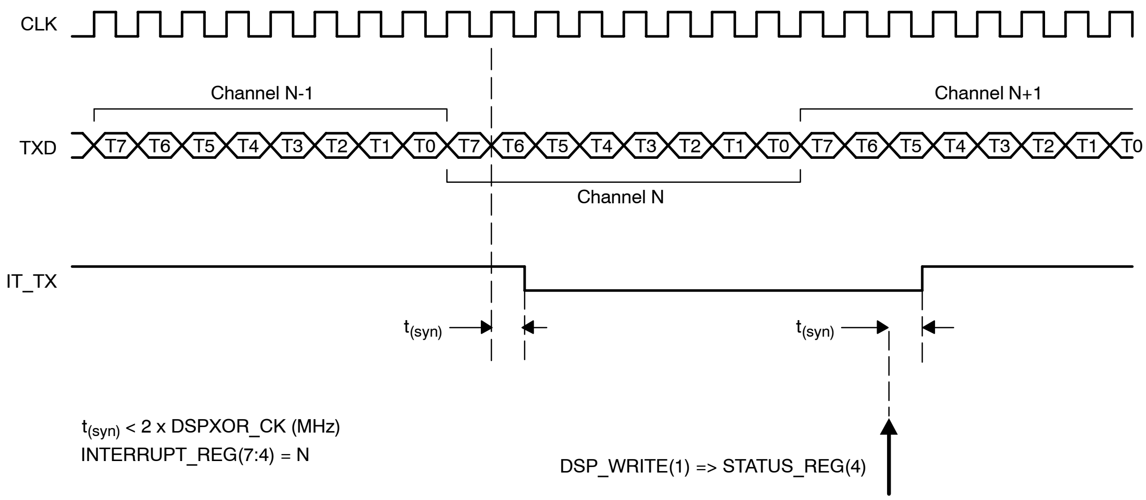


1.1.3.3 Transmit Interrupt

The transmit interrupt is generated every frame after the start of the transmission of a data word.

- In the single-channel mode, the interrupt is generated one clock period after the beginning of the transmission of the word.
- In the multichannel mode, the interrupt is generated one clock period after the transmission of the word of the channel whose number is defined by the NB_CHAN_IT_RX parameter of INTERRUPTS_REG register.

Figure 4. Transmit Interrupt Timing Diagram



1.1.3.4 Frame Duration Error Interrupt

The frame duration error interrupt is only generated when:

- ❑ The interface is configured in burst mode (CONTINUOUS = 0).
- ❑ The frame duration is smaller or longer than the expected value.

Namely, the expected frame duration = [(channels number) * (word size)] + (over-size number) is in clock period units with over-size numbers defined in OVER_SIZE_REG register.

If the frame duration is longer than the expected value, then an interrupt is generated one clock period after the number of the over_size clock periods, as defined in OVER_CLOCK parameter.

If the frame duration is smaller than the expected value, the interrupt is generated one clock period after the occurrence of the next frame pulse (first active edge).

Figure 5. Frame Duration Error—Too Many (Long)

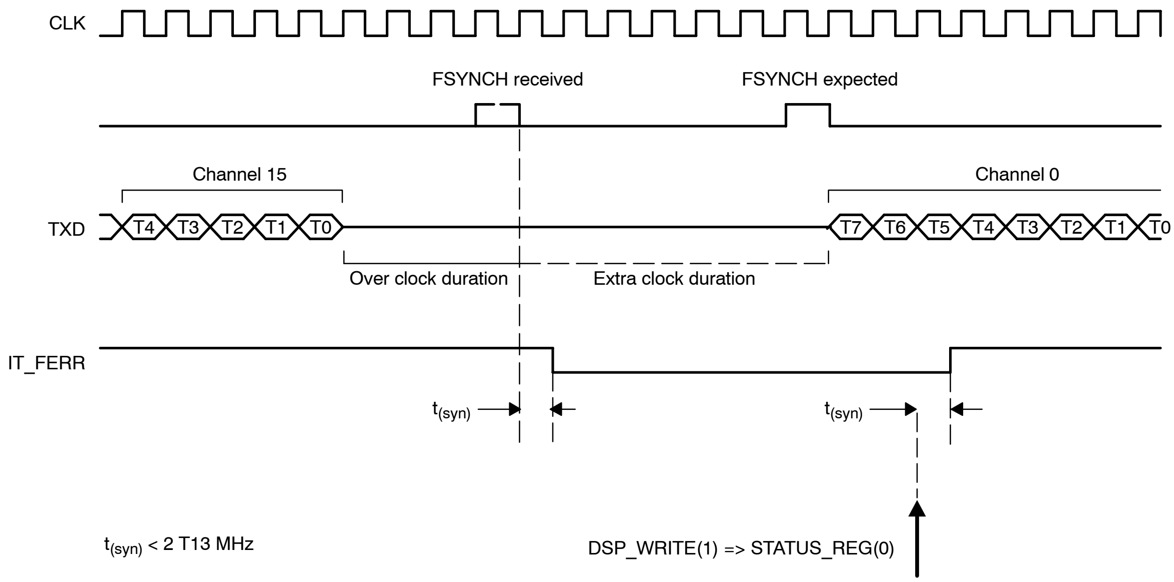
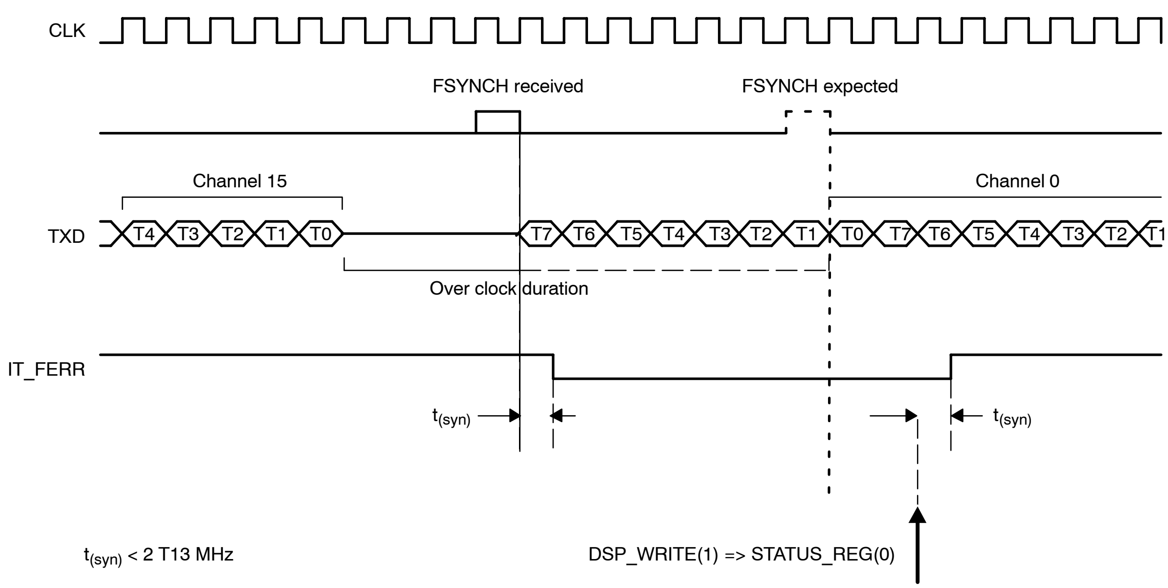


Figure 6. Frame Duration Error—Too Few (Short)



1.1.4 Interrupt Programming

At module reset, RX_INT, TX_INT, and FERR_INT are masked.

To validate an interrupt:

If in the multichannel mode, the RX and TX interrupts can be configured to occur in a dedicated channel of the frame [1-16].

- DSP_WRITE(channel_nb) = INTERRUPTS_REG(3:0) for RX_INT
- INTERRUPTS_REG(7:4) for TX_INT

Unmask the interrupt:

- DSP_WRITE(1) =
  - INTERRUPTS_REG(8) for RX_INT
  - INTERRUPTS_REG(9) for TX_INT
  - INTERRUPTS_REG(10) for FERR_INT

On interrupt occurrence:

- DSP_READ =
  - STATUS_REG(1) for FERR_INT occurrence
  - STATUS_REG(2) for RX_INT occurrence
  - STATUS_REG(3) for RX character overflow
  - STATUS_REG(4) for TX_INT occurrence
  - STATUS_REG(5) for TX character underflow

To release the interrupt signal and reset the corresponding status bits:

- DSP_WRITE(1) =
  - STATUS_REG(1) for FERR_INT release
  - STATUS_REG(2) for RX_INT release
  - STATUS_REG(4) for TX_INT release

1.1.5 DMA Channel Operation

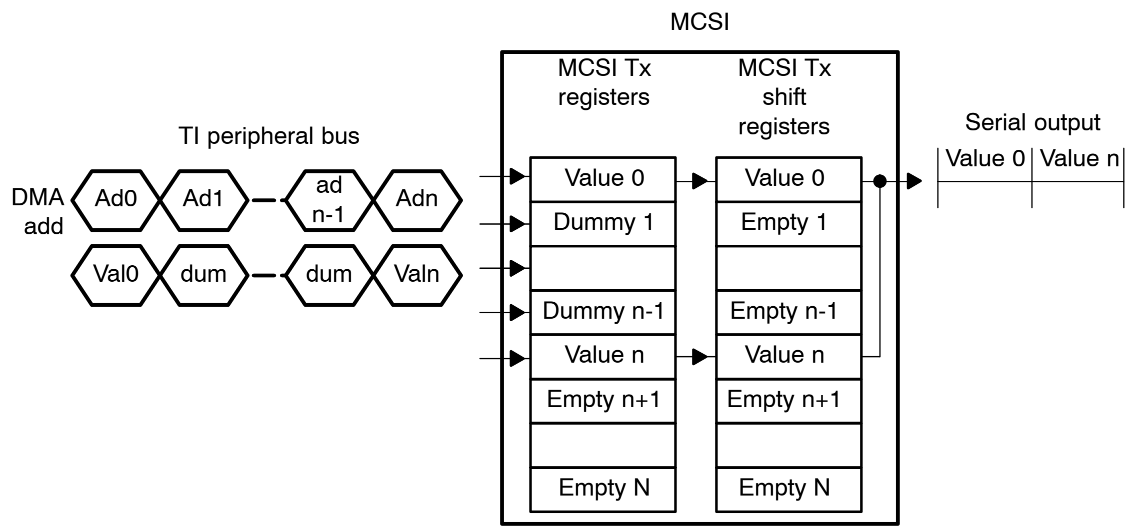
Both transmit and receive operations can be supported by DMA. DMA support is enabled by control bits in the MAIN_PARAMETERS_REG:

- MAIN_PARAMETERS_REG(15:14) = DMA_ENABLE(1:0)
  - TX_DMA_REQ enabled when DMA_ENABLE(0) = 1
  - TX_DMA_REQ disabled when DMA_ENABLE(0) = 0
  - RX_DMA_REQ enabled when DMA_ENABLE(1) = 1
  - RX_DMA_REQ disabled when DMA_ENABLE(1) = 0

1.1.5.1 Transmit DMA Transfers

A new transmit DMA transfer is initiated during the transmission of the last channel of a frame, at which time all data in the transmit registers (TX_REGs) has been moved to shift registers; the TX_REGs are now ready to be rewritten. If N channels are used, the DMA controller successively accesses all consecutive registers between TX_REG(0) and TX_REG(N-1). If some channels between TX_REG(0) and TX_REG(N-1) are not used, the DMA controller writes dummy values when addressing these unused registers (see Figure 7).

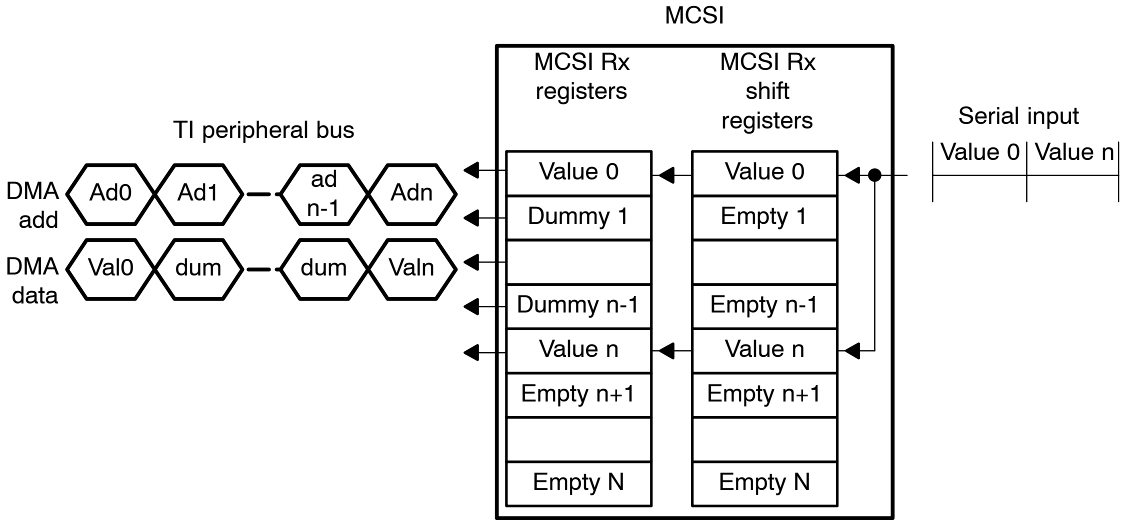
Figure 7. Transmit DMA Transfers



1.1.5.2 Receive DMA Transfers

A receive DMA transfer is initiated after the reception of the last channel of a frame, at which time all receive registers RX_REG have been updated and are ready to be read. If N channels are used, the DMA controller successively accesses all consecutive registers between RX_REG(0) and RX_REG(N-1). If some channels between RX_REG(0) and RX_REG(N-1) are not used, the DMA controller reads dummy values when addressing these unused registers (see Figure 8).

Figure 8. Receive DMA Transfers



A multichannel application cannot use DMA for some channels and interrupt servicing for others. RX/TX interrupts are not generated when DMA RX/TX transfers are enabled.

1.1.6 Interface Activation

1.1.6.1 Start Sequence

A typical sequence to start the interface is:

1) MCSI configuration:

   a) DSP_WRITE(0x0000)= CONTROL_REG in order to remove the write protection on the control registers

   b) DSP_WRITE(0x….)= MAIN_PARAMETERS_REG

   c) DSP_WRITE(0x….)= INTERRUPTS_REG

   d) DSP_WRITE(0x….)= CHANNEL_USED_REG

   e) DSP_WRITE(0x….)= CLOCK_FREQUENCY_REG

   f) DSP_WRITE(0x….)= OVER_CLOCK_REG

2) Transmit data loading for selected channels:

   a) DSP_WRITE(0x….)= TX_REG[channel index]

3) Enable the MCSI clock:

   a) DSP_WRITE(0x0001)= CONTROL_REG

1.1.6.2 Stop Sequence

A typical sequence to stop the interface is:

1) Disable the MCSI clock: DSP_WRITE(0x0000) = CONTROL_REG

   The status register keeps its content even after the stop of the transmission. The control registers can now be modified.

2) Software reset: DSP_WRITE(0x0002) = CONTROL_REG

   The software reset initializes the status register.

1.1.6.3 Software Reset

The MCSI software reset is activated with the SW_RESET bit of the control register (CONTROL_REG) (see Table 6, *Activity Control Register*).

This reset is limited to the control and status registers, the internal state machine, and the PISO and SIPO logic. The parameters registers are not affected by this software reset.

On the software reset, the MCSI reference clock is disabled, halting the execution of any current operating mode.

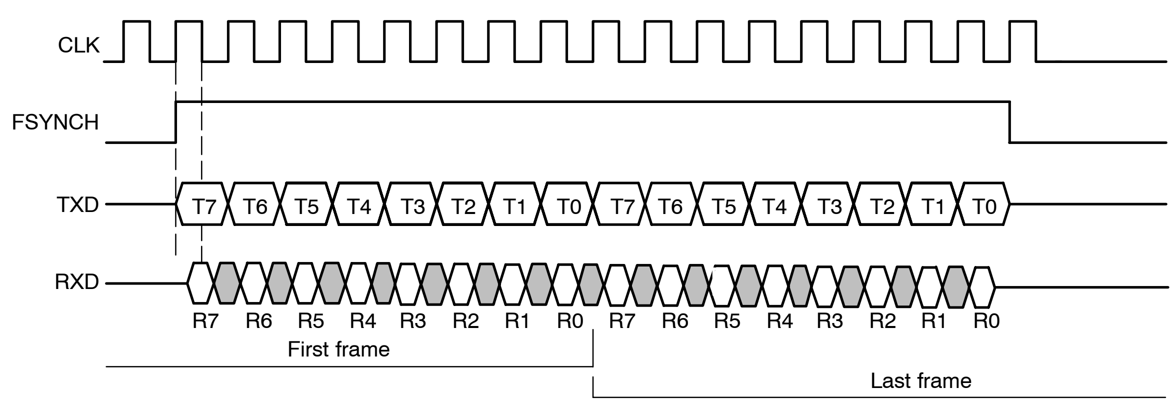
1.1.7 Functional Mode Timing Diagrams

The following timing diagrams are based on a positive clock polarity with the parameter CLOCK_POL = 0.

(Transmit on rising edge/receive on falling edge)

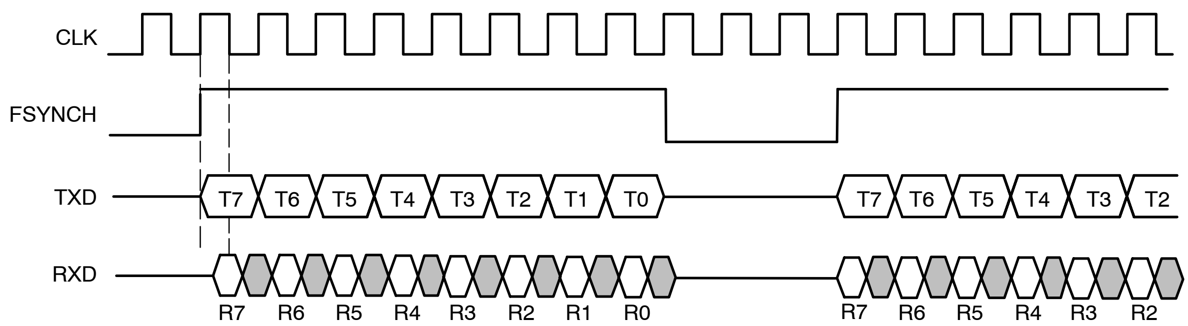
1.1.7.1 Single-Channel/Alternate Long Framing

Figure 9. Single-Channel/Alternate Long Framing



1.1.7.2 Single-Channel/Alternate Long Framing/Burst

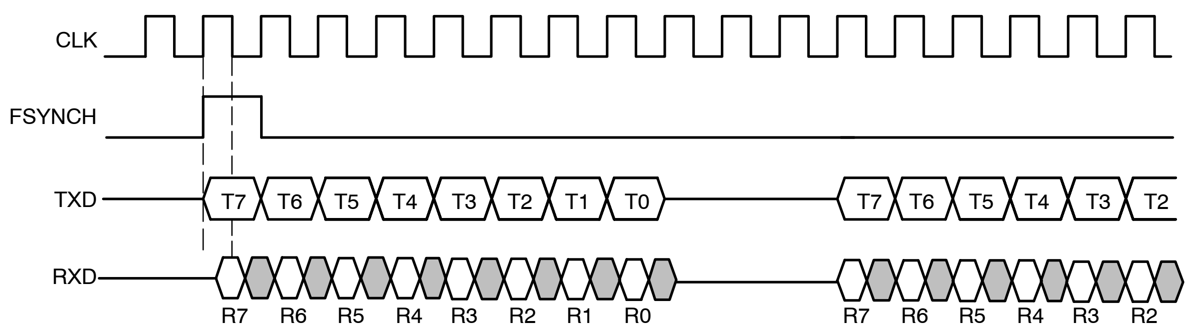
Figure 10. Single-Channel/Alternate Long Framing/Burst



OVER_CLOCK_REG = 0x0003

1.1.7.3 Single-Channel/Alternate Short Framing/Continuous/Burst

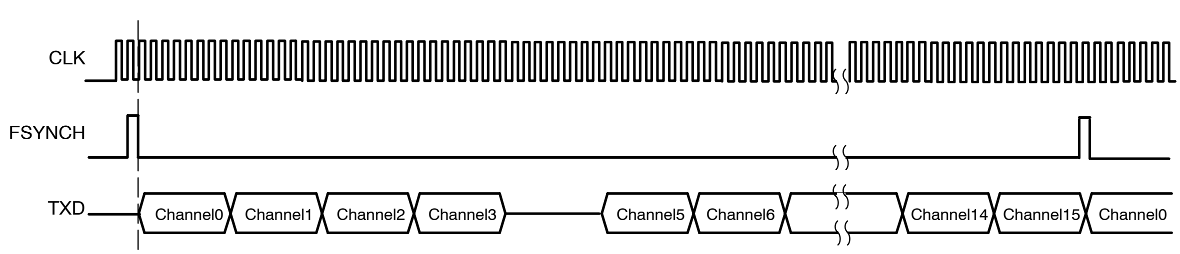
Figure 11. Single-Channel/Alternate Short Framing/Continuous/Burst



OVER_CLOCK_REG = 0x0003

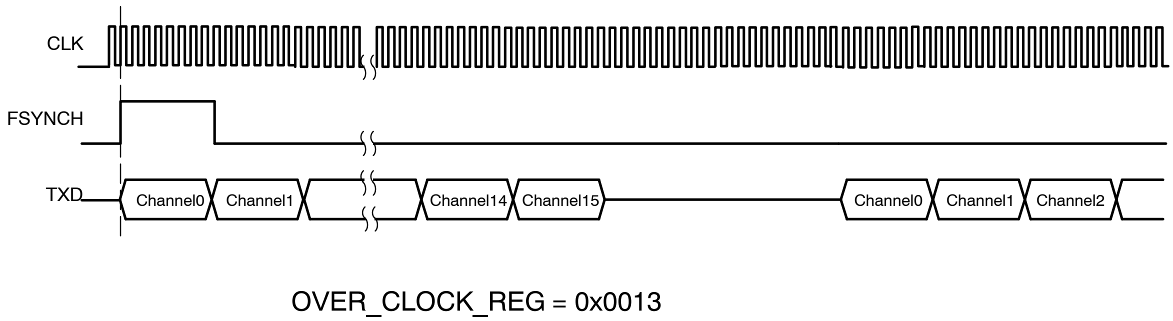
1.1.7.4 Multichannel/Normal Short Framing/Channel Disable

Figure 12. Multichannel/Normal Short Framing/Channel Disable



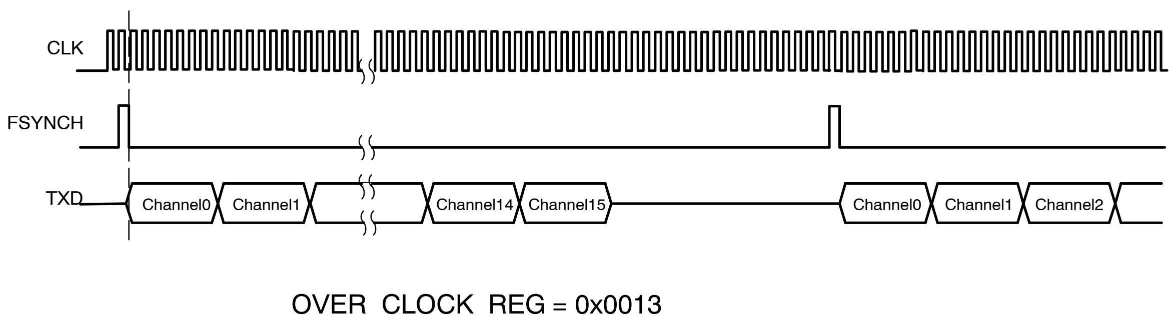
1.1.7.5 Multichannel/Alternate Long Framing/Continuous/Burst

Figure 13. Multichannel/Alternate Long Framing/Continuous/Burst



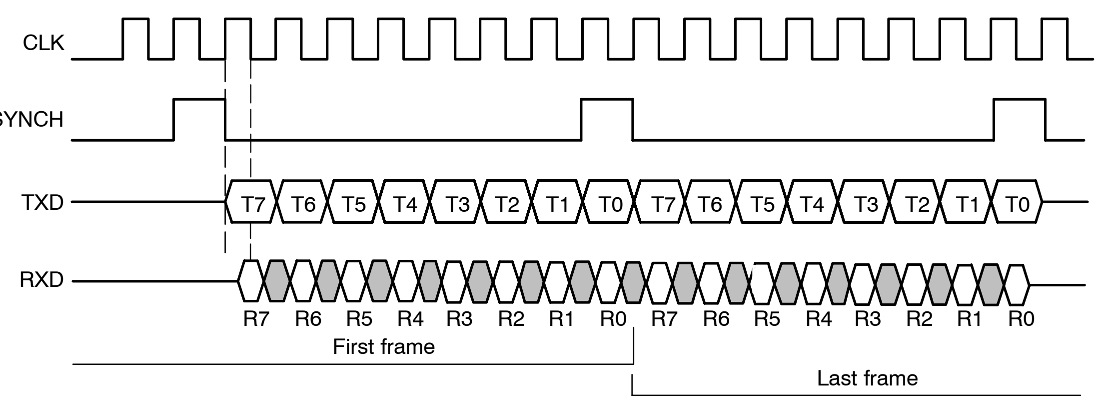
1.1.7.6 Multichannel/Normal Short Framing/Burst

Figure 14. Multichannel/Normal Short Framing/Burst



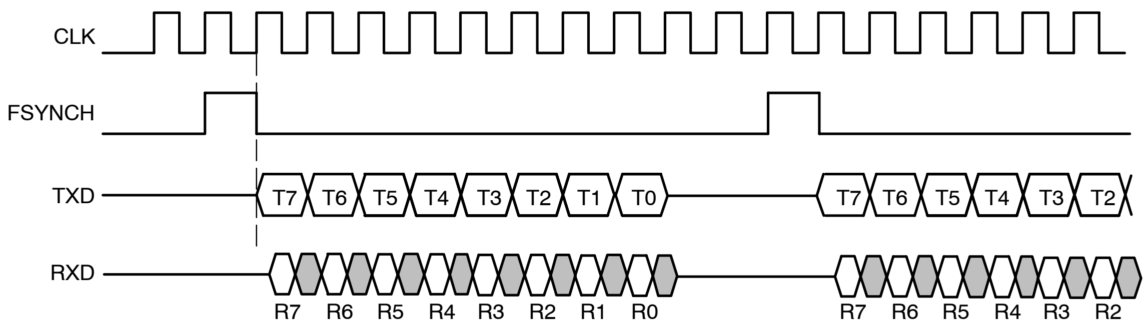
1.1.7.7 Single-Channel/Normal Short Framing

Figure 15. Single-Channel/Normal Short Framing



1.1.7.8 Single-Channel/Normal Short Framing/Burst

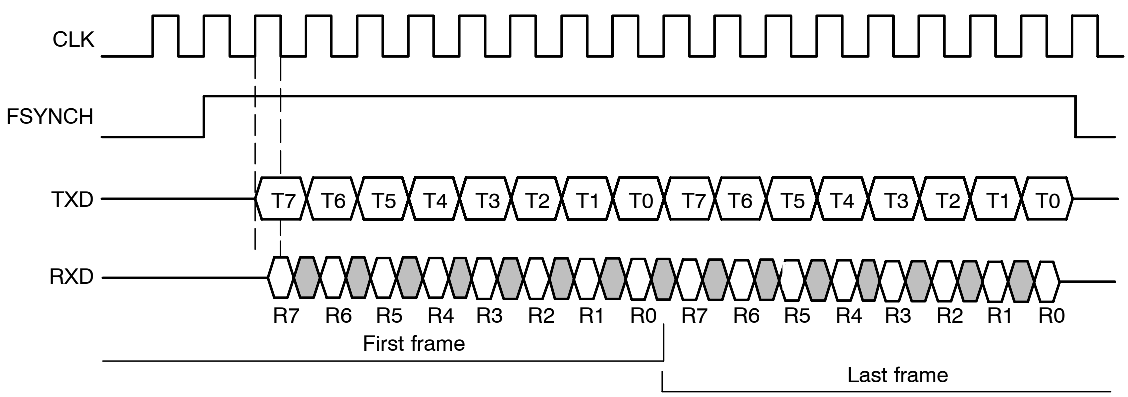
Figure 16. Single-Channel/Normal Short Framing/Burst



OVER_CLOCK_REG = 0x0003

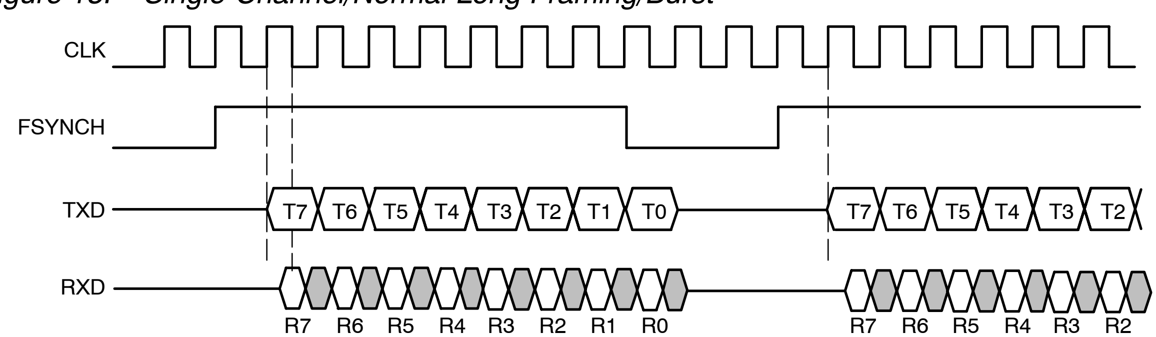
1.1.7.9 Single-Channel/Normal Long Framing

Figure 17. Single-Channel/Normal Long Framing



1.1.7.10 Single-Channel/Normal Long Framing/Burst

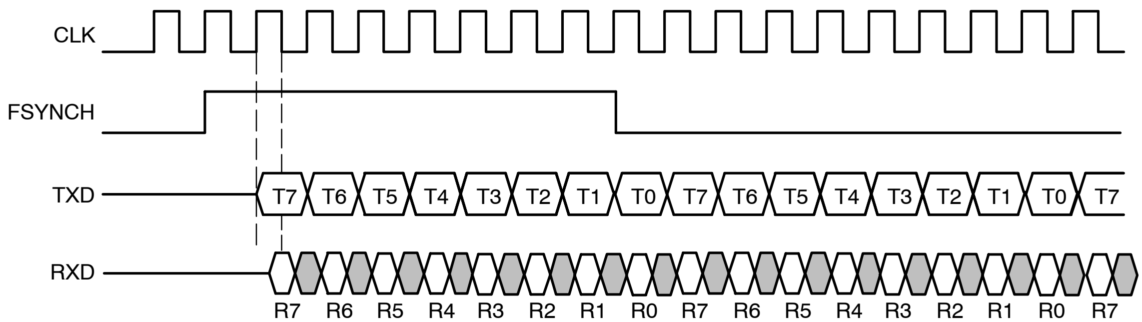
Figure 18. Single-Channel/Normal Long Framing/Burst



OVER_CLOCK_REG = 0x0003

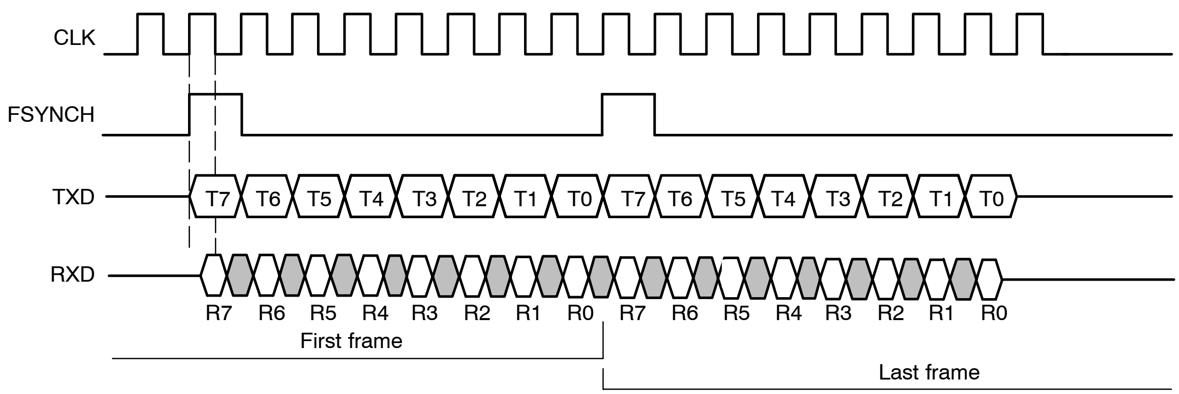
1.1.7.11 Single-Channel/Normal Long Framing/Continuous

Figure 19. Single-Channel/Normal Long/Continuous



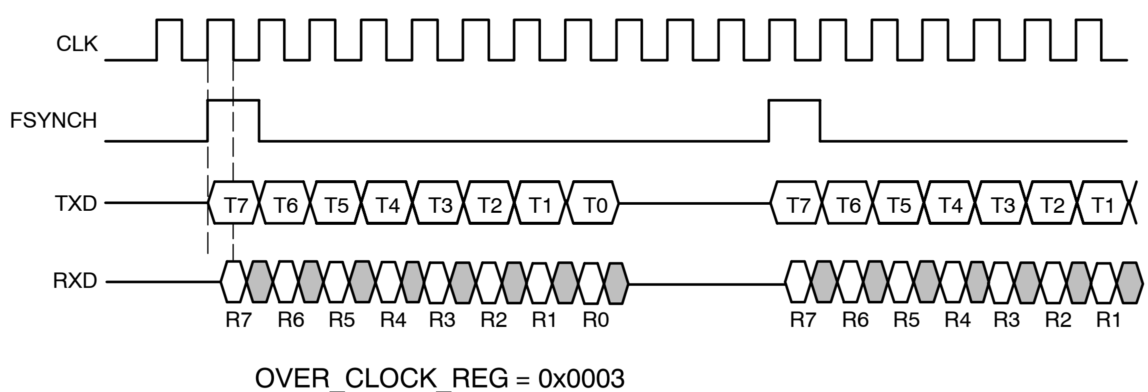
1.1.7.12 Single-Channel/Alternate Short Framing

Figure 20. Single-Channel/Alternate Short Framing



1.1.7.13 Single-Channel/Alternate Short Framing/Burst

Figure 21. Single-Channel/Alternate Short Framing/Burst



1.2 MCSI Register Descriptions

Table 1 through Table 9 describe the MCSI registers. The CHANNEL_USED_REG, CLOCK_FREQUENCY_REG, OVER_CLOCK_REG, INTERRUPTS_REG, and MAIN_PARAMETERS_REG are write protected if the MCSI is enabled (control_reg[0] = 1).

The channel selection register is only used in multichannel mode (see Table 1).

Table 1. Channel Selection Register (CHANNEL_USED_REG)

Bits	Field	Access	Hardware Reset
15	use_ch15	R/W	0
14	use_ch14	R/W	0
13	use_ch13	R/W	0
12	use_ch12	R/W	0
11	use_ch11	R/W	0
10	use_ch10	R/W	0
9	use_ch9	R/W	0
8	use_ch8	R/W	0
7	use_ch7	R/W	0
6	use_ch6	R/W	0
5	use_ch5	R/W	0
4	use_ch4	R/W	0
3	use_ch3	R/W	0
2	use_ch2	R/W	0
1	use_ch1	R/W	0
0	use_ch0	R/W	0

USE_CH[i] selects channel [i] for data transmission (active high).

The clock frequency register is used only in the master mode when the interface generates the serial clock (see Table 2).

Table 2. Clock Frequency Register (CLOCK_FREQUENCY_REG)

Bits	Field	Description	Access	Hardware Reset
15−11	Unused		R	0000 0
10−0	clk_freq	Division factor of 12 MHz reference clock (2<=clk_freq<= 2047) In the master mode, this register defines the transmission baud rate from a frequency ratio based on a 12 MHz reference clock. The transmission clock frequency can be programmed from 5.8 kHz to 6 MHz in steps or increments of 83 ns. Clock frequency = 12 MHz / clk_freq with 2 <= clk_freq <= 2047.	R/W	000 0000 0000

CLK_FREQ: division factor of 12 MHz reference clock (2<=clk_freq<= 2047)

In the master mode, this register defines the transmission baud rate from a frequency ratio based on a 12 MHz reference clock. The transmission clock frequency can be programmed from 5.8 kHz to 6 MHz in steps or increments of 83 ns.

Clock frequency = 12 MHz / clk_freq with 2 <= clk_freq <= 2047.

Table 3. Oversized Frame Dimension Register (OVER_CLOCK_REG)

Bits	Field	Description	Access	Hardware Reset
15−10	Unused		R	0000 00
9−0	over_clock	Overhead clock periods in frame duration (0 = OVER_CLOCK = 1023)	R/W	00 0000 0000

Table 4. Interrupt Masks Register (INTERRUPTS_REG)

Bits	Field	Description	Access	Hardware Reset
15−11	Unused		R	0000 0
10	mask_it_error	Mask of frame duration error interrupt (active at 0)	R/W	0
9	mask_it_tx	Mask of transmit interrupt (active at 0)	R/W	0
8	mask_it_rx	Mask of receive interrupt (active at 0)	R/W	0
7−4	Number channel for it_tx	Channel number for transmit interrupt generation (0 <=Nb_chan <= 15)	R/W	0000
3−0	Number channel for it_rx	Channel number for receive interrupt generation (0 <= Nb_chan <=15)	R/W	0000

Table 5. Main Parameters Register (MAIN_PARAMETERS_REG)

Bits	Field	Value	Description	Access	Hardware Reset
15−14	DMA enable		Enable bits for DMA:	R/W	00
		00	Normal mode (No DMA)		
		01	DMA transmit mode, normal receive mode		
		10	Normal transmit mode, DMA receive mode		
		11	DMA transmit and receive mode		
13−11	Reserved		Reserved bits. These bits should always be written as 0.	R/W	000
10	fsynch_polarity		Frame-synchronization pulse polarity	R/W	0
		0	Positive		
		1	Negative		
9	fsynch_mode		Frame-synchronization pulse position	R/W	0
		0	Normal		
		1	Alternate		

Table 5. Main Parameters Register (MAIN_PARAMETERS_REG) (Continued)

Bits	Field	Value	Description	Access	Hardware Reset
8	fsynch_size		Frame-synchronization pulse shape	R/W	0
		0	Short		
		1	Long		
7	Multi/single		Frame structure	R/W	0
		0	Single		
		1	Multi		
6	MCSI mode		Interface transmission mode	R/W	0
		0	Slave		
		1	Master		
5	Continuous/burst		Frame mode	R/W	0
		0	Burst		
		1	Continuous		
4	clock_polarity		Clock edge selection	R/W	0
		0	Positive		
		1	Negative		
3–0	Word size		Word size in bit numbers (2 <= size <= 15) with 2 for 3 bits and 15 for 16 bits.	R/W	0000

Table 6. Activity Control Register (CONTROL_REG)

Bits	Field	Value	Description	Access	Hardware Reset	Software Reset
15−3	Reserved		Reserved bits. These bits should always be written as 0.	R	0000 0000 0000 0	0000 0000 0000 0
2	Reserved		Reserved bits. These bits should always be written as 0.	R/W	0	0
1	MCSI software reset		Asynchronous reset of MCSI module	R/W	0	1
		0	Disable			
		1	Enable			
0	MCSI clk enable		Enable clock of MCSI module	R/W	0	0
		0	Disable			
		1	Enable			

Note:

The software reset is applied as long as the MCSI software reset bit is set to 1. A software reset disables the MSCI (the MCSI clk enable bit is cleared) and initializes the status register. It does not modify the other registers.

To clear an interrupt on the MCSI, the DSP must write to the MCSI status register with the bit corresponding to the interrupt set to 1. The MCSI status register has a two-cycle latency when written to, so the interrupt line is cleared two cycles after a write. In order to prevent clearing the interrupt handler before the interrupt line is cleared, the interrupt routine must be at least two cycles long.

Table 7. Interface Status Register (STATUS_REG)

Bits	Field	Value	Description	Access	Hardware Reset	Software Reset
15–7	Reserved		Reserved bits. These bits should always be written as 0.	R	0000 0000 0	0000 0000 0
6	Reserved		Reserved bits. These bits should always be written as 0.	R/W	0	0
5	TX underflow		Transmit underflow	R	0	0
		0	No under flow			
		1	Under flow			
4	TX ready		Flag for transmit interrupt occurrence	R/W	0	0
		0	No interrupt			
		1	Interrupt			
3	RX overflow		Receive overflow	R	0	0
		0	No overflow			
		1	Overflow			
2	RX ready		Flag for receive interrupt occurrence	R/W	0	0
		0	No interrupt			
		1	Interrupt			
1	Error type few/many		Too short (few) or too long frame (many) status	R	0	0
		0	Short			
		1	Long			
0	Frame error		Error flag when wrong frame duration	R/W	0	0
		0	Correct			
		1	Bad			

This register is cleared by a software reset.

Table 8. Receive Word Register (RX_REG[15:0])

Bits	Field	Access	Hardware Reset
15	b15	R	U
14	b14	R	U
13	b13	R	U
12	b12	R	U
11	b11	R	U
10	b10	R	U
9	b9	R	U
8	b8	R	U
7	b7	R	U
6	b6	R	U
5	b5	R	U
4	b4	R	U
3	b3	R	U
2	b2	R	U
1	b1	R	U
0	b0	R	U

Note: The MCSI receives the most significant bit first. For example, if the word_size equals 11, the upper 12 bits of the RX registers contain the received data, and the lower 4 bits are zeroes.

Table 9. Transmit Word Register (TX_REG[15:0])

Bits	Field	Access	Hardware Reset
15	b15	R/W	U
14	b14	R/W	U
13	b13	R/W	U
12	b12	R/W	U
11	b11	R/W	U
10	b10	R/W	U
9	b9	R/W	U
8	b8	R/W	U
7	b7	R/W	U
6	b6	R/W	U
5	b5	R/W	U
4	b4	R/W	U
3	b3	R/W	U
2	b2	R/W	U
1	b1	R/W	U
0	b0	R/W	U

Note: The MCSI transmits the most significant bit first. For example, if the word_size equals 11, the upper 12 bits of the TX registers are transmitted.

2 MCSI1

This section provides information specific to MCSI1 (Figure 22) on the OMAP5910 device.

2.1 MCSI1 Pin Description

Table 10 identifies the MCSI1 I/O pins.

Table 10. MCSI1 Pin Descriptions

Pin	I/O Direction	Description
MCSI1.DIN	In	Data input
MCSI1.DOUT	Out	Data output
MCSI1.CLK	In/out	Bit clock
MCSI1.SYNC	In/out	Frame synchronization

2.2 MCSI1 Interrupt Mapping

Table 11 identifies the MCSI1 interrupt mappings. MCSI1 generates level 2 interrupts for both the DSP and the MPU. Only one MPU MCSI1 interrupt covers TX, RX, and frame error conditions. Software must check the MCSI1 status register to determine the interrupt source.

Table 11. MCSI1 Interrupt Mapping

Incoming Interrupts	Level 2 DSP Interrupt	Level 2 MPU Interrupt
MCSI1 TX interrupt	IRQ_06	IRQ_16
MCSI1 RX interrupt	IRQ_07	IRQ_16
MCSI1 Frame Error	IRQ_10	IRQ_16

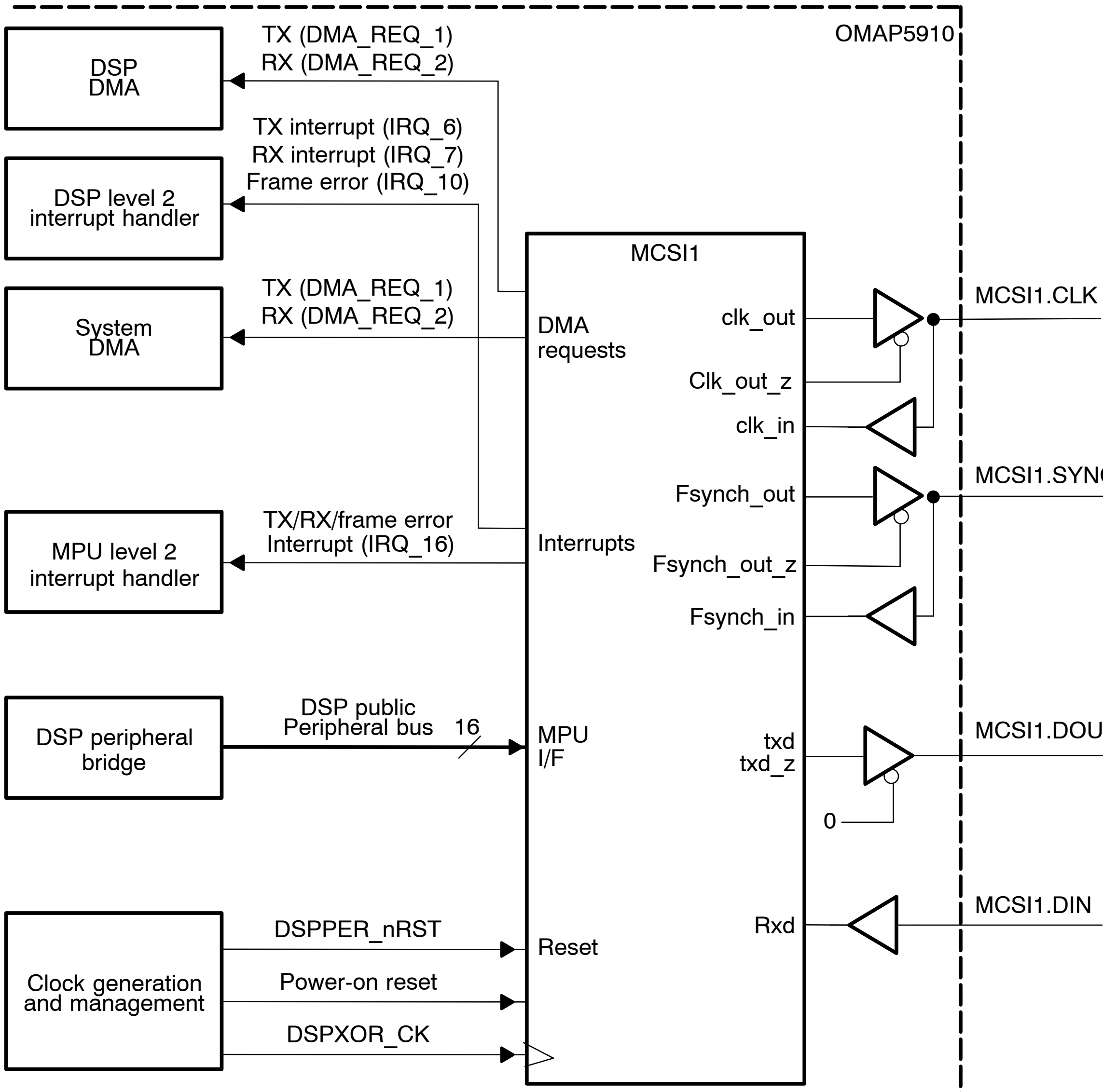
2.3 MCSI1 DMA Request Mapping

Table 12 identifies MCSI1 DMA request lines.

Table 12. TDMA Request Mapping—MCSI1

DMA Request Source	DMA Request Line—DSP	DMA Request Line—MPU
MCSI1 TX	DMA_REQ_01	DMA_REQ_01
MCSI1 RX	DMA_REQ_02	DMA_REQ_02

Figure 22. MCSI1 Interface Diagram



3 MCSI2

This section provides information specific to MCSI2 (Figure 23) on the OMAP5910 device.

3.1 MCSI2 Pin Description

Table 13 identifies the MCSI2 I/O pins.

Table 13. MCSI2 Pin Descriptions

Pin	I/O Direction	Description
MCSI2.DIN	In	Data input
MCSI2.DOUT	Out	Data output
MCSI2.CLK	In/out	Bit clock
MCSI2.SYNC	In/out	Frame synchronization

3.2 MCSI2 Interrupt Mapping

Table 14 identifies the MCSI2 interrupts. MCSI2 generates level 2 interrupts for both the DSP and the MPU. Only one MPU MCSI2 interrupt covers TX, RX, and frame error conditions; software must check the MCSI2 status register to determine the interrupt source.

Table 14. MCSI2 Interrupt Mapping

Incoming Interrupts	Level 2 DSP Interrupt	Level 2 MPU Interrupt
MCSI2 TX interrupt	IRQ_08	IRQ_17
MCSI2 RX interrupt	IRQ_09	IRQ_17
MCSI2 Frame Error	IRQ_11	IRQ_17

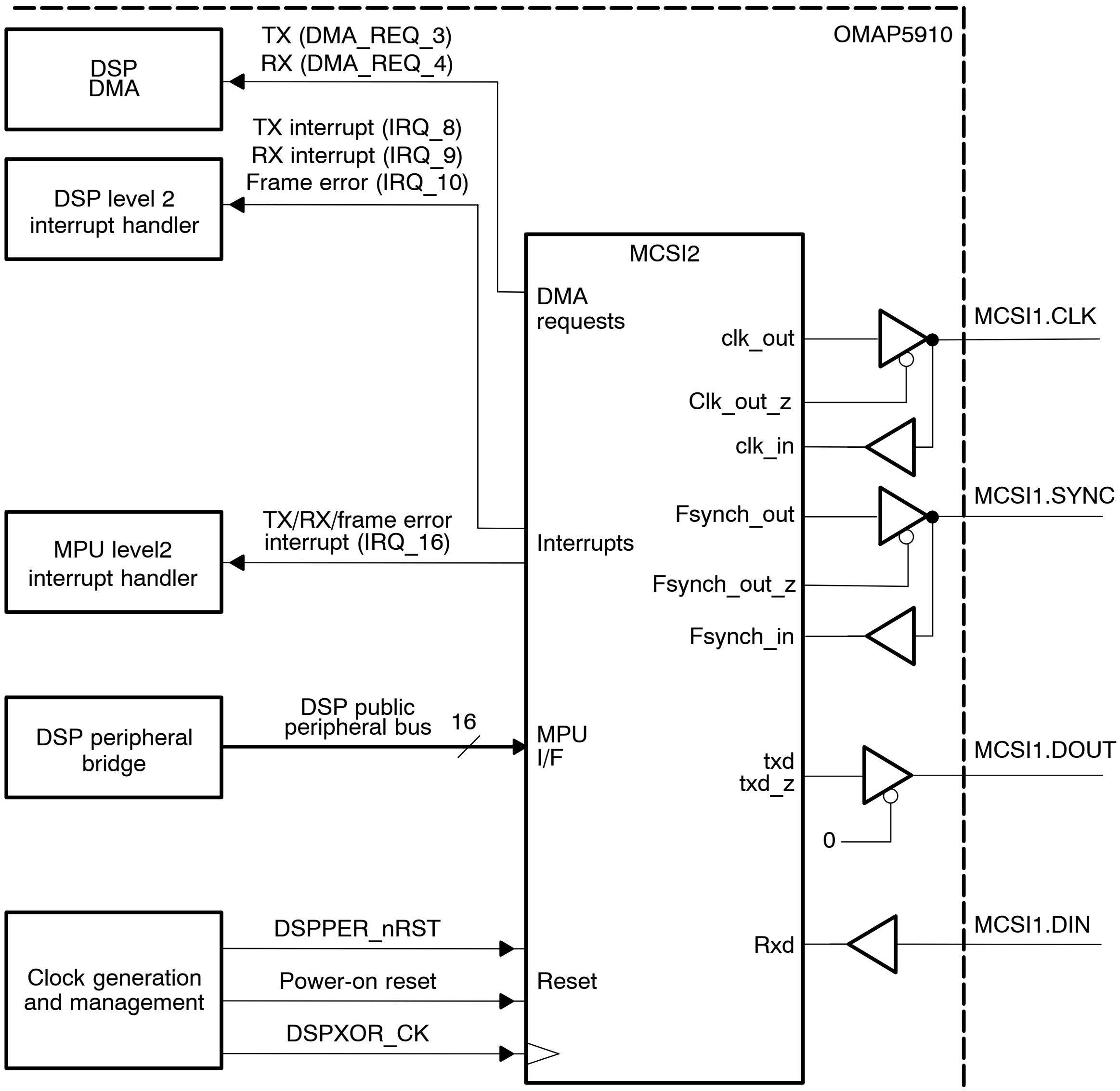
3.3 MCSI2 DMA Request Mapping

Table 15 identifies MCSI2 DMA request lines. Only the DSP DMA controller can transfer MCSI2 data; there is no MPU DMA capability.

Table 15. DMA Request Mapping—MCSI2

DMA Request Source	DMA Request Line—DSP	DMA Request Line—MPU
MCSI2 TX	DMA_REQ_03	–
MCSI2 RX	DMA_REQ_04	–

Figure 23. MCSI2 Interface Diagram



3.4 MCSI Addresses and Mapping

The base address for each MCSI register map is:

- MCSI1 (Bluetooth MCSI):
  - 0x09400 (DSP memory map word address)
  - E101:2800 (MPU memory map byte address)

- MCSI2 (Modem MCSI):
  - 0x09000 (DSP memory map word address)
  - E101:2000 (MPU memory map byte address)

Table 16 shows the MCSI registers and their offset addresses.

Table 16. MCSI Register Mapping

Register Name	Offset In Bytes	Register Name	Offset In Bytes
RX15	0x7E	TX10	0x54
RX14	0x7C	TX9	0x52
RX13	0x7A	TX8	0x50
RX12	0x78	TX7	0x4E
RX11	0x76	TX6	0x4C
RX10	0x74	TX5	0x4A
RX9	0x72	TX4	0x48
RX8	0x70	TX3	0x46
RX7	0x6E	TX2	0x44
RX6	0x6C	TX1	0x42
RX5	0x6A	TX0	0x40
RX4	0x68	Unused	0x3F
RX3	0x66	/	/
RX2	0x64	Unused	0x0E
RX1	0x62	Status	0x0C
RX0	0x60	Clock frequency	0x0A
TX15	0x5E	Over-clock	0x08
TX14	0x5C	Channel used	0x06
TX13	0x5A	Interrupts	0x04
TX12	0x58	Main parameters	0x02
TX11	0x56	Control	0x00

Revision History

Table 17 lists the changes made since the previous version of this document.

Table 17. Document Revision History

Page	Additions/Modifications/Deletions
	This document has been reviewed for accuracy and there are no changes since the previous version (October 2003) of this document.

This page is intentionally left blank.

Index