‘C6x McBSP as a TDM Highway

APPLICATION REPORT: PRELIMINARY

Shaku Anjanaiah

Digital Signal Processing Solutions
July 1998

TEXAS INSTRUMENTS

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain application using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

TRADEMARKS

TI is a trademark of Texas Instruments Incorporated.

Other brands and names are the property of their respective owners.

CONTACT INFORMATION

US TMS320 HOTLINE	(281) 274-2320
US TMS320 FAX	(281) 274-2324
US TMS320 BBS	(281) 274-2323
US TMS320 email	dsph@ti.com

Contents

Figures

Tables

‘C6x McBSP as a TDM Highway

Abstract

This document describes how the multi-channel buffered serial ports (McBSP) in the Texas Instruments (TI™) TMS320C6201 digital signal processor (DSP) is used to communicate on a time division multiplexed data highway.

TDM provides multiple devices a time slot for performing data transfer. Thus multiple users use multiple channels, but each user has a set of channel(s) assigned to it for transmission and reception. The McBSP can support up to 128 channels in multi-channel mode. Each of these 128 channels can be enabled or disabled via software in order to communicate only in those time slots that it is supposed to.

When multiple users are connected to the same data lines, bus contention can be an issue during transmission. Although each device connected to this TDM highway has its own channel assignment, it is possible that the next device getting on to the bus can contend with the last bit(s) of the previous channel. The workaround for this data contention problem is addressed in this document by considering multiple McBSPs connected to a single data line.

Product Support

Related Documentation

The following list specifies product names, part numbers, and literature numbers of related TI documentation.

- *TMS320C6201 Digital Signal Processor* data sheet, March 1998, Literature number SPRS051C, and its errata.
- *TMS320C6201/C6701 Peripherals* Reference Guide, March 1998, Literature number SPRU190B

World Wide Web

Our World Wide Web site at **www.ti.com** contains the most up to date product information, revisions, and additions. Users registering with TI&ME can build custom information pages and receive new product updates automatically via email.

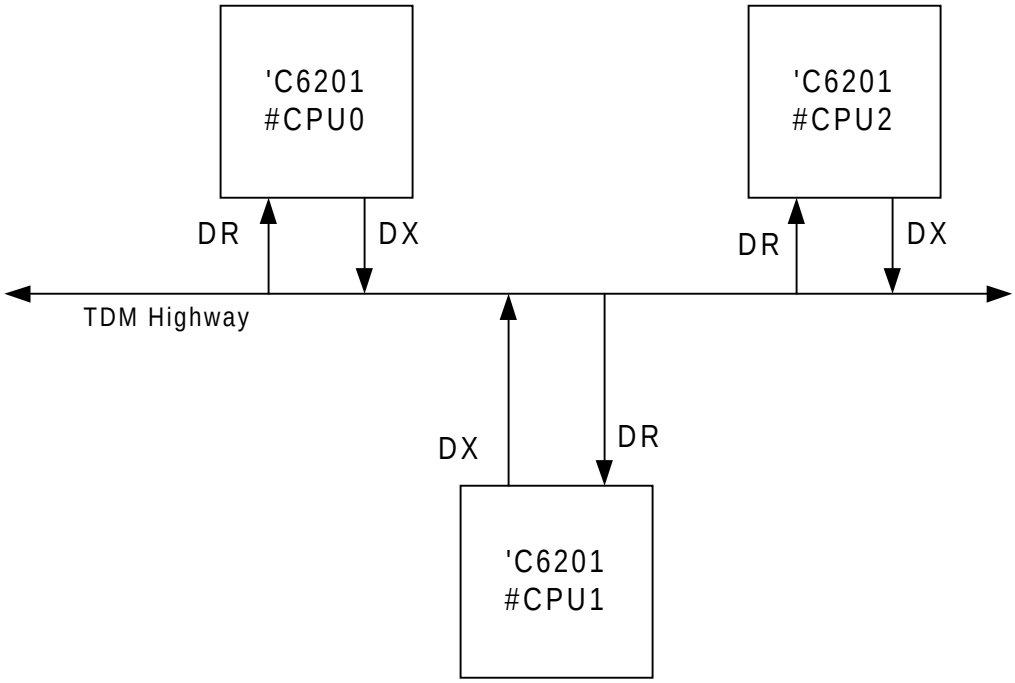
Email

For technical issues or clarification on switching products, please send a detailed email to **dsph@ti.com**. Questions receive prompt attention and are usually answered within one business day.

Design Problem

How do I use the TMS320C6201 multi-channel buffered serial port (McBSP) to communicate over a Time Division Multiplexed (TDM) data highway without bus contention?

Figure 1. Multiple 'C6201 on a Time-Division Multiplexed Bus



Multi-channel Operation Overview

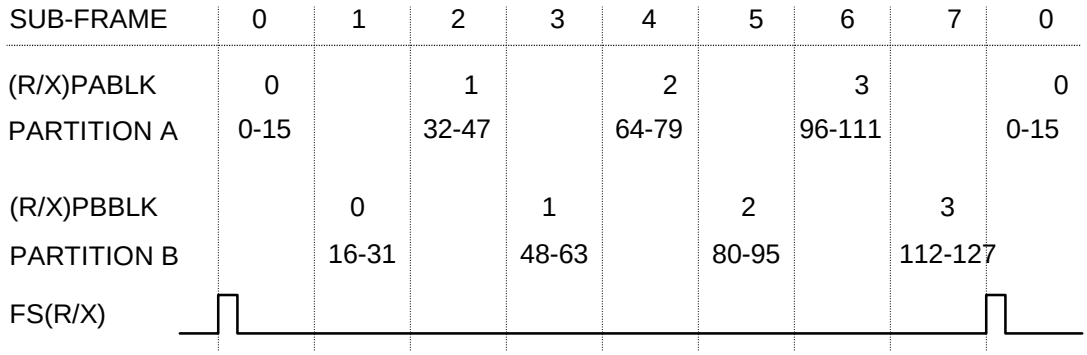
The McBSP can perform multi-channel operation for single-phase frames. Single-phase frames are characterized by a group of elements that have same element size. The maximum number of elements per frame is the same as the number of channels, which is equal to 128. Therefore each frame represents a time-division multiplexed data stream with up to 128 channels.

A set of programmable control registers in the McBSP specific for multi-channel operation makes communication on a TDM highway possible. They are Multi-channel Control Register (MCR), the Transmit Channel Enable Register (XCER), and Receive Channel Enable Register (RCER). These registers are explained in detail in the *TMS320C6201/C6701 Peripherals* Reference Guide.

The 128 channels in a frame are divided into eight 16-channel sub-frames as shown in Figure 2. Each channel can have programmable data sizes (8-, 12-, 16-, 20-, or 32-bits). But all channels in a frame have to be of same data size. Odd-numbered sub-frames constitute Partition A (represented by (R/X)PABLK), and even-numbered belong to Partition B (represented by (R/X)PBBLK). Each of the 128 channels can be enabled or disabled for both transmit and receive. This is achieved by the 32-bit (R/X)CER register. Therefore, up to 32 channels can be enabled in two consecutive sub-frames, 16 in Partition A and 16 in Partition B.

Channels can be enabled or disabled at any time as long as it does not belong to the current or active sub-frame. An active sub-frame can be viewed as that time slot where transmission or reception is taking place. For example, if data transfer is ongoing in sub-frame 2 (channels 32-47 in (R/X)PABLK=1), the lower 16-bits of the channel enable registers (which correspond to Partition A) should not be changed since they would affect the enabling of the current channels. This can be ensured by using the current block status bits (R/X)CBLK in the MCR.

Figure 2. Sub-frames, Partitions, and Channels in a Multi-channel Frame



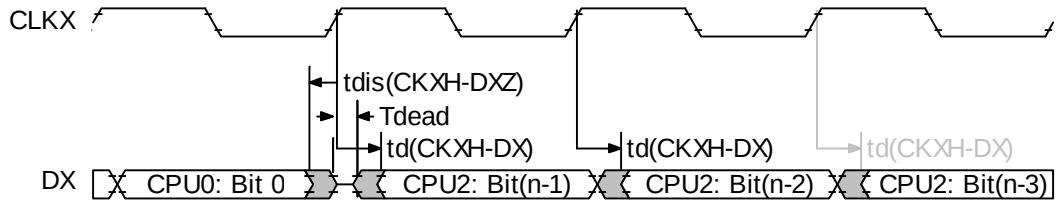
Problem Description

When multiple devices transmit over the same DX line, care should be taken to avoid bus jamming due to simultaneous or overlapped write accesses by two or more devices. This bus contention during transmission can be avoided by ensuring enough dead time between the last write of one device and the first write access of the next device.

Basically, the disable time (tdis(CKXH-DXZ)) of the DX output should be much smaller than the enable or delay time (td(CKXH-DX)) of the DX output of the next device to avoid data collision. This is shown as Tdead in Figure 3 and is defined as:

$$Tdead = \{ td(CKXH\text{-}DX) - tdis(CKXH\text{-}DXZ) \} >> 0$$

Figure 3. DX Timing for Multichannel Operation



The timing parameters and their values for the ‘C6201 McBSP DX output as per the datasheet is shown in Table 1. It can be inferred from the timing numbers of the ‘C6x McBSP that the delay time and disable time for the DX pin is the same. Therefore, if two ‘C6201s transmitted on successive channels, there *will* be bus contention. The following sections will describe the workaround for this problem.

Table 1. Switching characteristics of DX Output of 'C6201 McBSP

NO.	PARAMETER			Min	Max	UNIT
12	tdis(CKXH-DXZ)	Disable time, CLKX high to DX high impedance following last data bit	CLKX int	0	4	ns
			CLKX ext	3	16	ns
13	td(CKXH-DX)	Delay time, DX valid after CLKX high. For the first data bit, this is assured by design to be the delay time for data to become low impedance.	CLKX int	0	4	ns
			CLKX ext	3	16	ns

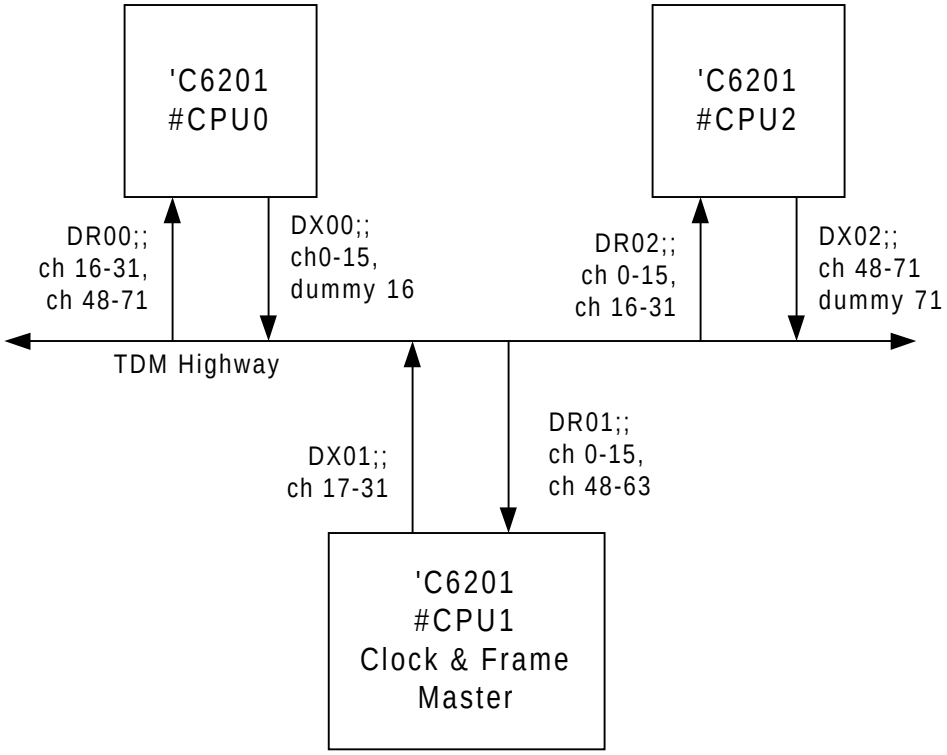
Software Solution

The bus contention problem is due to lack of control on the DX output buffer’s turn-on (delay) and turn-off (disable) time. This can be overcome by programming the McBSP to transfer one extra (dummy) element or channel than what is necessary. This additional (dummy) channel should be disabled via the XCER register since it does not represent data of interest. A disabled channel will have its DX in high-impedance state. The dummy channel will provide the necessary dead time between transfers from two ‘C6201s and thus prevent contention.

Consider the example where one McBSP in each of the three ‘C6201s are connected to the TDM bus as shown in Figure 4. Assume CPU1 is the clock and frame master. Therefore CPU1 generates transmit and receive clocks, and transmit and receive frame syncs.

In Figure 4, D(R/X)00 represents D(R/X) pins of McBSP0 in CPU0. Similarly D(R/X)01 and D(R/X)02 correspond to McBSP0 of CPU1 and CPU2. For this example, DX00 occupies sub-frame 0 and 1, DX01 occupies sub-frame 1, and DX02 in sub-frames 3, and 4. On the receive side, DR00 listens to sub-frame 1, 3, and 4; DR01 receives channels in sub-frame 0, and 3; and lastly DR02 receives sub-frame 0, and 1.

Figure 4. Three 'C6201s on a TDM bus



The actual channels (or time slots) on which each of the McBSPs transmit and receive are listed in Table 2 and Table 3 respectively. In this TDM set up, up to 71 out of the available 128 channels are considered for data transfer. Among these 71 channels, only some are enabled for transmission and reception. In this example, some/all channels in sub-frame 0, 1, 3, and 4 are enabled whereas sub-frame 2 corresponding to channels 32-47 is not used and therefore disabled. In order to selectively choose channels, the appropriate multi-channel mode has to be chosen. For this example, XMCM=RMCM=1 will suit the application. The (R/X)MCM=1 mode disables all 128 channels by default, and the required channels are enabled via the 16-channel sub-frames (R/X)P(A/B)BLK and the channel enable registers (R/X)CER.

TDM Transmission

Since CPU1 is the clock and frame master, it will be programmed to generate frame sync for every 71 channels. The frame period (FPER) depends on the serial element size in a frame and the number of elements or channels in a frame. CPU0 and CPU2 are slaves, and therefore will start their respective data transfer upon receiving the frame and clock from the master CPU1.

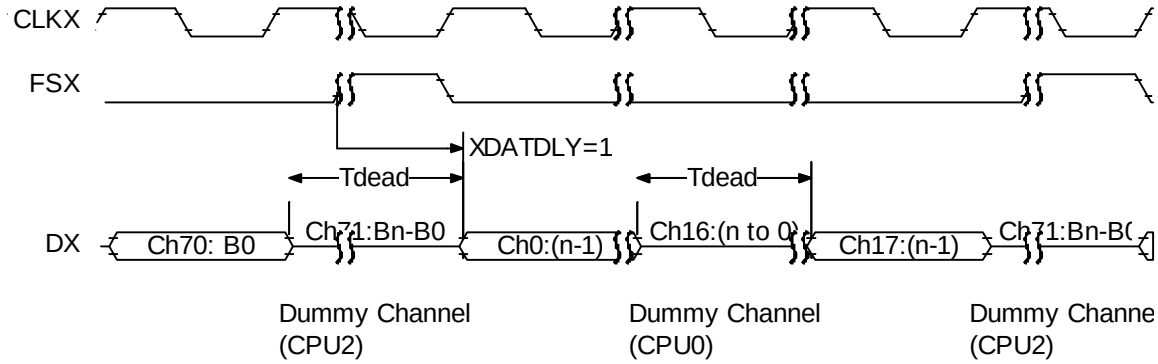
Table 2. TDM Data Transmission Setup

CPU#	Sub-frame/ XP(A/B)BLK	Channels Transmitted	Register Value
CPU#0 DX00	Sub-frame 0: XPABLK = 0; XPBBLK = 0;	Sub-frame channels: 0-15;; Enabled channels (XPABLK): 0-15;; Dummy channel: 16;; Disabled channels: all others;;	XCER = 0x0000FFFF
CPU#1 DX01	Sub-frame 1: XPBBLK = 0;	Sub-frame channels: 16-31;; Enabled channels (XPBBLK): 17, 19; 20-23; 25,27; 28-31;; Disabled channels: all others;;	XCER = 0xFAFA0000
CPU#2 DX02	Sub-frames 3, and 4: XPBBLK = 1; XPABLK = 2;	Sub-frame channels: 48-63 and 64-79;; Enabled channels (XPBBLK): 48, 49, 51; 52-54; 56, 58; 60, 61, 63;; Enabled channels (XPABLK): 64-70;; Dummy channel: 71;; Disabled channels: all others;;	XCER = 0xB55B007F

Consider the transmit and receive data delay to be 1. The frame sync from the master initiates data transmission on CPU0 since CPU0 is enabled for transmission on channels 0-15. The first data bit on DX is available 1 clock after the frame sync is active. At the same time, CPU1 and CPU2 are enabled to receive some channels on sub-frame 0. Note that more than one device can receive the same channel(s), and this will not cause any contention problem.

Now, in the next sub-frame 1 (channels 16-31), CPU1 is programmed to start transmitting from channel 17. CPU0 is programmed for 17 (channels 0-16) serial elements and the last element will be disabled so that DX will be driven to high-impedance state. This is because channel 16 has to be a dummy channel in order to prevent bus contention between CPU0 and 1.

Figure 5. Dummy channel on TDM bus to prevent Bus-contention



The transmission and reception continues on enabled channels up to channel 71. Soon after channel 71 is transmitted on the TDM bus by CPU2, the next frame starts at channel 0. This is because FPER is programmed for ((data-size in bits * 71) –1) and also the data delay is 1. This provides for no gaps between two frames, which is known as maximum packet frequency. Since channel 0 is enabled, this again leads to successive channels (channel 71 and channel 0) being driven and resulting in bus contention. Therefore channel 71 has to be a dummy channel and therefore should be disabled.

TDM Reception

The receive section in CPU0, 1, and 2 can receive any channel of interest without any restrictions on overlapping of channels as shown in Table 3. However, note that CPU0 receives channels from sub-frames 1, 3, and 4. Sub-frame 1 and 3 belong to partition B (RPBBLK) and sub-frame 4 to RPABLK. Note that the channels enabled for one sub-frame cannot be changed when it is the current one in transfer. Therefore, when CPU0 receiver is in sub-frame 0 with a particular set of channels enabled (via RCER in RPBBLK 0), the RCER cannot be changed in order to be ready for the next sub-frame 3 which falls under the same partition. But, before sub-frame 3 arrives, the receiver should be enabled for the appropriate channels (56, 58 in this case). To do this, the current block status bits RCBLK in the Multi-channel Control Register (MCR) can be probed to find out the current partition in progress. If the RCBLK does not point to block 0, then RCER can be programmed for channel enabling for sub-frame 3. This can also be done for the transmit side using the XCBLK, if required.

Table 3. TDM Data Reception Setup

CPU#	Sub-frame/ RP(A/B)BLK	Channels Received	Register Value
CPU#0 DR00	Sub-frames 1, 3, 4: RPBBLK = 0;	Sub-frame channels: 16-31, 48-63, and 64-71;; Enabled channels (RPBBLK): 25, 27;;	RCER = 0x0A000000
	RPBBLK = 1; RPABLK = 1;	Enabled channels (RPBBLK): 56, 58;; Enabled channels (RPABLK): 64-70;; Disabled channels: All others ;;	RCER = 0x05000000 RCER = 0x0000007F
CPU#1 DR01	Sub-frames 0, 3: RPABLK = 0;	Sub-frame channels: 0-15, 48-63;; Enabled channels (RPABLK): 0-7;;	RCER = 0xB05B00FF
	RPBBLK = 1;	Enabled channels (RPBBLK): 48, 49, 51; 52-54; 60, 61, 63;; Disabled channels: all others;;	
CPU#2	Sub-frames 0, 1:	Sub-frame channels: 0-15, 16-31;;	

DR02	RPABLK = 0;	Enabled channels (RPABLK): 0-15;;	RCER = 0xF0FAFFFF
	RPBBLK = 0;	Enabled channels (RPBBLK): 17, 19; 20-23; 28-31;;	
		Disabled channels: all others;;	

McBSP Initialization

As it is in typical applications, consider that the DMA services the McBSP. The following steps describe the setup of interrupts, DMA, and the McBSP in the required order for the TDM example above.

1. For CPU0, CPU1, and CPU2 McBSP0: Program the Sample Rate Generator Register (SRGR), Serial Port Control Register (SPCR), Pin Control Register (PCR), Receive Control Register (RCR), Multi-channel Control Register (MCR), Receive/Transmit Channel Enable Registers ((R/X)CER) to the required values.

   Caution: Do not set the /GRST bit in SPCR in this step.

2. Take CPU1's sample rate generator out of reset by setting /GRST=1 in the SPCR. /GRST is not required to be set for CPU0 and 2 since the clocks and frames are inputs, and therefore the sample rate generator is not used for McBSP0.

3. Enabling Interrupts: To use interrupts, you have to set the Global Interrupt Enable (GIE), and Non-Maskable Interrupt Enable (NMIE) bits in the IER. For receiver of CPU0, an end-of-subframe interrupt (RINTM=01b) can be used to determine end of a 16-channel sub-frame and thereby change the channel enabling for odd or even numbered sub-frame (see first row in Table 3).

4. Select the DMA channel you want to use. Enable CPU interrupts that correspond to the DMA channel that will be used to service the McBSP. The default mapping of DMA channel-complete interrupts to CPU is as follows:

   DMA channel 0 ➔ CPU interrupt 8

   DMA channel 1 ➔ CPU interrupt 9

   DMA channel 2 ➔ CPU interrupt 11

   DMA channel 3 ➔ CPU interrupt 12

   DMA initialization: Program the DMA source/destination registers, primary control register for required operation.

5. Instruct the DMA to run. For example, set START=01b in the DMA channel's primary control register to start the DMA without auto-initialization.

6. Take the transmitter and receiver of slave CPUs (0 and 2) out of reset. They will now await the clock (CLKR/X) and frame sync (FSR/X) from the master CPU1.

7. Take the transmitter and receiver of the master (CPU0) out of reset. Set /FRST=1 in CPU1. This causes the first frame sync output on FS(R/X) after 8 bit-clocks.

Conclusion

Multiple 'C6x devices can be connected in a multi-processor environment with one of them as a clock and frame master. This is achieved using time-division multiplexing via the McBSP interface using the multi-channel mode. The programmable features in the multi-channel mode of the McBSP such as channel enabling, detection of current block so that the next similar-numbered (odd or even) sub-frame channels can be enabled, and the various ways of enabling and unmasking the channels via (R/X)MCM bits make complex multi-channel applications simple.