Endianness and the TSB12LV41 (MPEG2Lynx) Microprocessor Interface

APPLICATION REPORT: SLLA021

Randy Lawson
1394 Applications Group

Mixed Signal and Logic Products
BuS Solutions
5 December 1997

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.

GPLynx, PCILynx, Lynxsoft, Mpeg2Lynx, 1394 TImes are trademarks 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
US TI 1394 Applications	1394@ti.com

Contents

Figures

Endianness and the TSB12LV41 (MPEG2Lynx) Microprocessor Interface

Abstract

A never-ending source of confusion for users of microprocessors, microcontrollers and their peripherals is the topic of endianness. Endianness refers to the way data is referenced and stored in a processor's memory. TI's TSB12LV41 (MPEG2Lynx) is a 1394-1995 Link Layer controller, designed to support the IEC61883 standard for transmitting and receiving MPEG2 Transport Streams on a 1394 network. The TSB12LV41 supports three types of processors; the ARM7TDMI ("Thumb") processor, as well as Motorola 68000 and Intel 8051 type processors. The Configuration Register space of the TSB12LV41 is Big Endian, however the TSB12LV41's microprocessor interface is designed to support both Big Endian and Little Endian processor types. This document describes how data is stored in both types of endianness domains and gives instructive examples of how both types of processors can read and write to the Configuration Registers inside the TSB12LV41.

Product Support

Related Documentation

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

- *TSB12LV41 (MPEG2Lynx), IEEE 1394-1995 Link-Layer Controller for MPEG2 Transport, Product Preview Data Sheet*, Literature number SLLS276A
- *Interfacing the TSB12LV41 1394 Link Layer Controller to the TMS320AV7100 DSP Embedded ARM Processor*, Literature number SLLA015

World Wide Web

Our World Wide Web site at www.ti.com/sc/1394 contains the most up to date product information, revisions, and additions. On this page, one can subscribe to 1394TImes, which periodically updates subscribers on events, articles, products, and other news regarding 1394 developments.

Email

For technical issues or clarification on 1394 products, please send a detailed email to 1394@ti.com.

TSB12LV41 Endianness

Definitions

- Quadlet = 4 bytes, 32 bits
- Doublet = 1/2 quadlet, 16 bits
- MSB = Most significant bit
- LSB = Least significant bit

General

The TSB12LV41 microprocessor interface is Big Endian. Big Endian means the most significant byte is byte 0 at the left-hand side and the least significant byte is byte 3 at the right hand side. Please refer to Figure 1 and Figure 2 below for an illustration of both endianness types.

Figure 1. Big Endian Illustration chart

Byte#0 (MSByte)	Byte#1	Byte#2	Byte#3 (LSByte)

Figure 2. Little Endian Illustration chart

Byte#3 (MSByte)	Byte#1	Byte#2	Byte#0 (LSByte)

TSB12LV41 registers are 32 bits wide (1 quadlet). When using a 16-bit data bus with Word access mode (word access set by bit 0 at address 1ECh) the least significant bit of the address is ignored. The next consecutive bit determines which doublet, upper or lower, the 16 bit data gets written to or read from.

Figure 3. 12LV41 Internal Register Format

Bit 0 (MSB) . . . . . Bit 31 (LSB)			
Address 000h	Address 001h	Address 002h	Address 003h
Byte#0 (MSByte)	Byte#1	Byte#2	Byte#3 (LSByte)
Bits 0 - 7	Bits 8 - 15	Bits 16 - 23	Bits 24 - 31
Upper Doublet		Lower Doublet	

The TSB12LV41 Microprocessor address bus has the MSB at pin 50 (ADR0) and the LSB at pin 59 (ADR8). The Microprocessor data bus has the MSB at pin 80 (DATA0) and the LSB at pin 61 (DATA15). For the host processor to work correctly with the TSB12LV41, users have to connect their microprocessor's data and address bus LSB to the TSB12LV41's data and address bus LSB, and their microprocessor's data and address bus MSB to the TSB12LV41's data and address bus MSB, regardless of the bit number labeling and which type of endianness their microprocessor uses. For Little Endian processors, the correct byte swapping can be done by setting TSB12LV41's BeCtl bit (Big Endian Control Bit) in the IOCR Register (register1ECh) to 0 (zero).

Important Notes

1) Connect your Microprocessor LSB to 12LV41's LSB for both address and data busses. 12LV41 address LSBit = ADR8, data LSBit = DATA15. THIS MUST BE DONE REGARDLESS OF THE BeCtl BIT SETTING. (NOTE: For Word access the LSB of the TSB12LV41 address bus can be connected to ground since it's value is a "don't care" for word access.)

2) The Big Endian Control bit BeCtl (bit 1 at address 1ECh) should be 0 when connecting to a Little Endian Processor.

3) If the BeCtl bit equals 1, the DataInvarnt bit (bit 6 at address 1ECh) is a "don't care".

4) For a 16 bit data bus (Word Access), set bit 0 at address 1ECh to 0. For an eight bit data bus (Byte Access), set bit 0 to 1.

5) Since the data bus for the 12LV41 is 16 bits wide, but the internal CFRs (Configuration registers) are 32 bits wide, the microprocessor must do two consecutive cycles to do a complete read or write in Word Access mode. For Byte Access mode, the microprocessor must do four consecutive cycles.

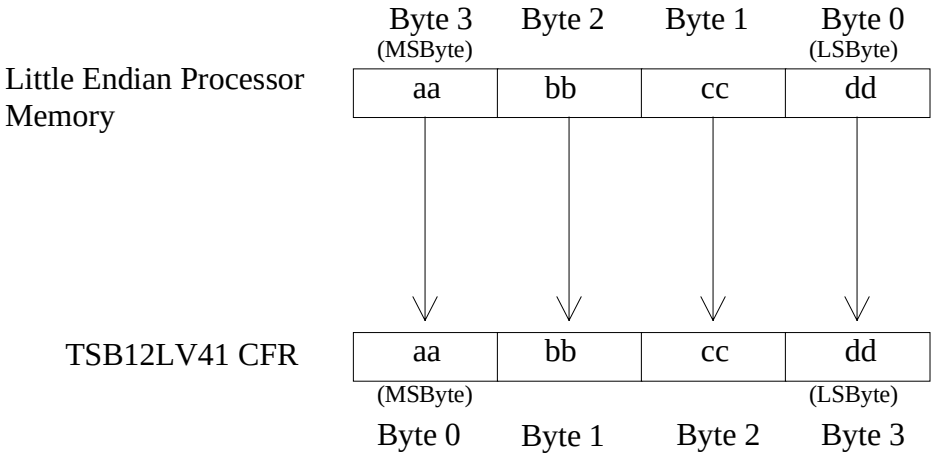
6) When reading from the TSB12LV41 in byte mode, the upper byte (bit 0-7) of the 16-bit data bus contains the byte data, the lower byte (bit 8-15) will be driven to all zeros. When writing to the TSB12LV41 in byte mode, users should put the data in the upper byte (bit 0-7). The data in the lower byte has no effect on the data to be written into the TSB12LV41.

7) When the BeCtl bit is set to the opposite value to that of the actual endianness of the microprocessor the TSB12LV41 is connected to, the data result passed through the two different endianness domains is unpredictable.

Data and Address Invariance for Little Endian Processors

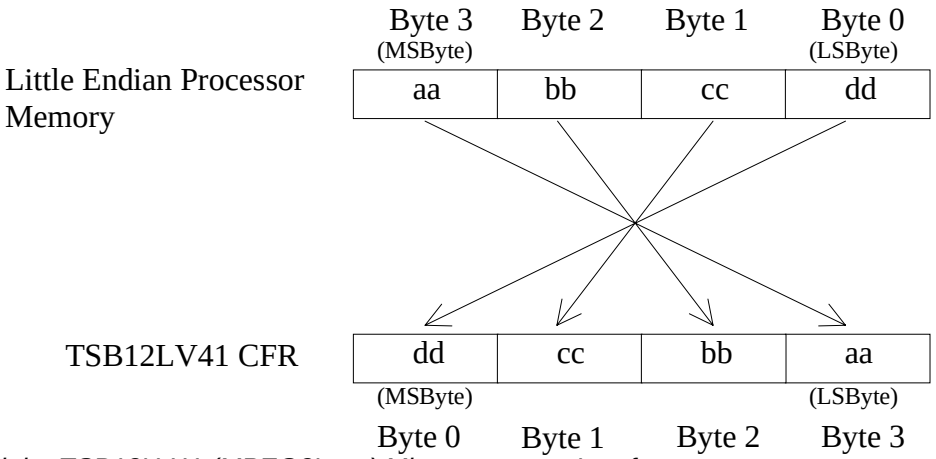
When BeCtl = 0 (Little Endian mode), the setting of DataInvarnt bit (bit 6 at address 1ECh) becomes important. The setting of this bit determines how the received bytes from the Microprocessor are mapped into the TSB12LV41 internal registers. For DataInvarnt = 0 (Address Invariant mode) the byte addresses are preserved. In other words byte 3 from the microprocessor will be mapped to byte 3 of the TSB12LV41 register, byte 2 to byte 2, and so on. For DataInvarnt = 1 (Data Invariant mode) the byte addresses are NOT preserved. See Figure 5 below.

Figure 4. Data Invariance Byte Swap



Data Invariance preserves the actual value of the data as it was stored in the processor's memory. However, note that the byte ordering of the data in both systems is different. Thus if the data represented an integer, then both systems would interpret the data the same. If the data represented an array, or string or any byte-indexed structure, then the data would be interpreted differently by the two systems.

Figure 5. Address Invariance Byte Swap

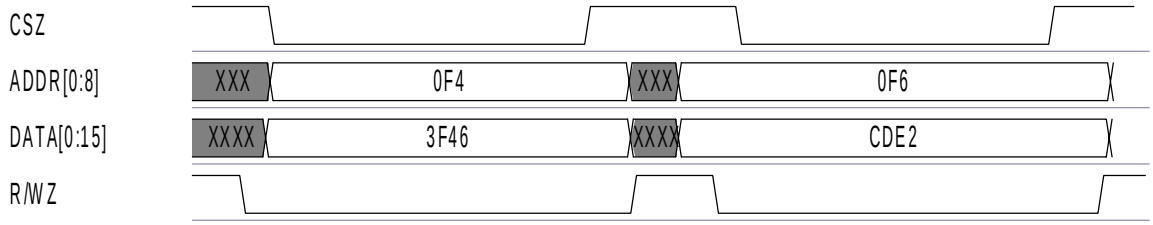


Address Invariance preserves the byte ordering between both systems. For example, byte 3 in the Little Endian processor memory is also byte 3 in the TSB12LV41 CFR space. As Figure 5 shows, this byte ordering is maintained by the TSB12LV41 automatically when in Address Invariance mode by swapping the order in which the incoming bytes on the microprocessor are written to the CFRs.

Big Endian Write Example

A Big Endian processor wants to write the value 3F46CDE2h to address 0F4h (DSS Formatter Control Register) in the TSB12LV41.

Figure 6. Big Endian Write Cycle Functional Timing Example



The data in register 0F4h would be:

Figure 7. Data in register 0F4h

Bit 0 (MSB) . . . . . Bit 31 (LSB)			
Byte#0 (MSByte)	Byte#1	Byte#2	Byte#3 (LSByte)
3F	46	CD	E2

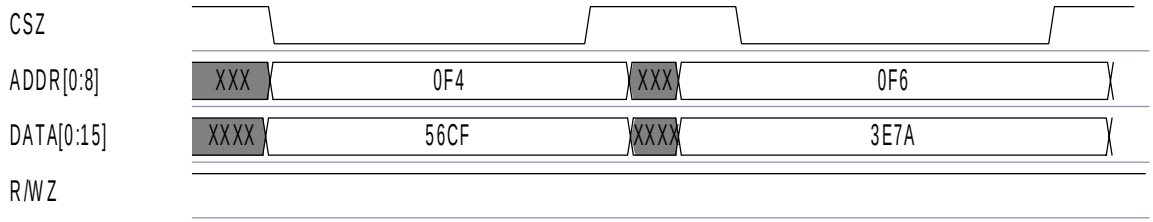
Big Endian Read Example

Figure 8. Data in TSB12LV41 Register 0F4h

Bit 0 (MSB) . . . . . Bit 31 (LSB)			
Address = 0F4	Address = 0F5	Address = 0F6	Address = 0F7
Byte#0 (MSByte)	Byte#1	Byte#2	Byte#3 (LSByte)
56	CF	3E	7A

The read data would look like the following:

Figure 9. Big Endian Read Cycle Functional Timing Example



Little Endian Write Example

Little Endian processor wants to send data 2EA85f6Ch to address 0F4h (DSS Formatter Control register)

Figure 10. Little Endian Write Cycle Functional Timing Example

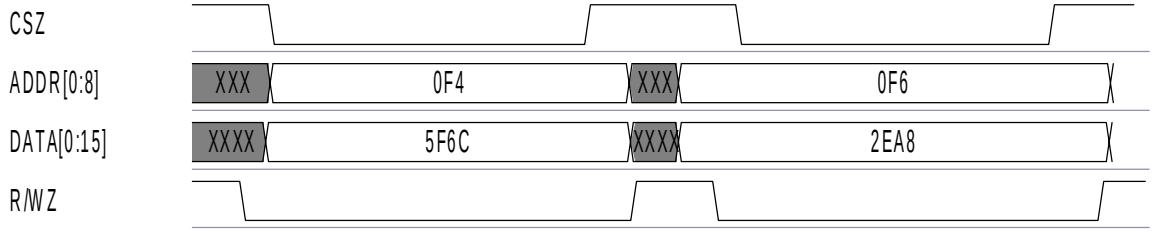


Figure 11. Data in Register 0F4h for ADDRESS INVARIANT Setting

Bit 0 (MSB) . . . . . Bit 31 (LSB)			
Byte#0 (MSByte)	Byte#1	Byte#2	Byte#3 (LSByte)
6C	5F	A8	2E

Figure 12. Data in Register 0F4h For DATA INVARIANT Setting

Bit 0 (MSB) . . . . . Bit 31 (LSB)			
Byte#0 (MSByte)	Byte#1	Byte#2	Byte#3 (LSByte)
2E	A8	5F	6C

Little Endian Read Example

For reads from the 12LV41 see the following example. If the TSB12LV41 has data 3A4F5D6Ch at address 0F4h (32 bit register):

Figure 13. Data in Register 0F4h

Bit 0 (MSB) . . . . . Bit 31 (LSB)			
Address = 0F4	Address = 0F5	Address = 0F6	Address = 0F7
Byte#0 (MSByte)	Byte#1	Byte#2	Byte#3 (LSByte)
3A	4F	5D	6C

Figure 14. Little Endian Read Functional Timing – ADDRESS INVARIANT

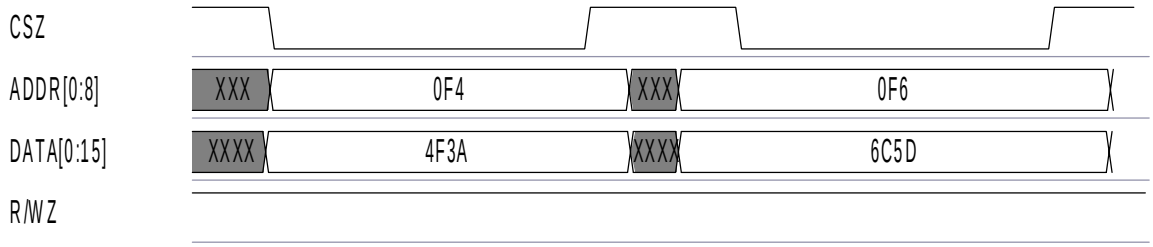


Figure 15. Little Endian Read Functional Timing – DATA INVARIANT