

9800511-9

TAG

This Sheet Has Been Replaced

THIS SHEET HAS BEEN
REPLACED

A METHOD AND APPARATUS FOR DATA ORDERING OF I/O TRANSFERS IN BI-MODAL ENDIAN POWERPC SYSTEMS

TECHNICAL FIELD

The present invention relates in general to data processing systems, and in particular, to supporting data communications between input/output devices and a processor.

BACKGROUND INFORMATION

In most computer systems today, the smallest addressable unit of storage is an 8-bit byte. Data is typically transferred between the various components of a computer system in groups of bytes, such as words, doublewords and quadwords to meet bandwidth needs. If the data is to be used or evaluated correctly, it is necessary to know the correlation of the byte addresses to the value of the data, *i.e.*, the most significant byte ("MSB").

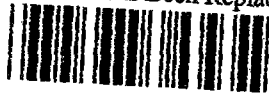
Processor architectures that order data such that the lowest byte address corresponds to the highest order byte (leftmost) are called Big Endian.



Architectures that assign the lowest byte address to the lowest order (rightmost) byte are referred to as Little Endian. Some processor architectures, such as the PowerPC processor manufactured by IBM corporation support both byte ordering schemes.

To illustrate the difference, consider the format of the word of data shown in FIGURE 1. Data structure "n" consists of 4 bytes of data represented as hexadecimal characters 1A, 2B, 3C, 4D. When a processor, in Big Endian mode, stores the word in memory, the resulting storage map appears as shown in FIGURE 2. If the processor was set up in Little Endian mode, the storage mapping would look as shown in FIGURE 3.

When configured to operate in Little Endian mode, processors based on the PowerPC architecture present a slightly different storage map to the true Little Endian map as shown in FIGURE 3. When presenting 1, 2 or 4-byte data transfers to the memory subsystem, PowerPC processors such as the 601, 604 and 620 chips will modify the lower 3 bits of the effective storage address creating the "pseudo-Little Endian" storage map as shown in FIGURE 4. The address translation results in reversing the byte ordering within doublewords from the true Little Endian mapping. This creates a problem for input/output (I/O) devices that need access to architected storage areas. Therefore, there is a need in the art to permit I/O devices, which require a true Little Endian image of these storage facilities to be able to operate with PowerPC processors.

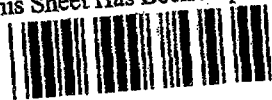


SUMMARY OF THE INVENTION

The foregoing need is addressed by the present invention, which provides circuitry for reordering data structures so that input/output devices, which require a true Little Endian data structure order, can efficiently communicate with a PowerPC processor and its particular manner for ordering data structures. More particularly, the present invention provides for byte ordering multiplexers implemented between the input/output devices and the remainder of the PowerPC system architecture.

In one embodiment, the present invention includes a data processing system that comprises a PowerPC processor utilizing PowerPC Little Endian byte ordering, and an I/O device utilizing true Little Endian byte ordering, and a memory controller operable for communicating data between the PowerPC processor and the I/O device, wherein the memory controller includes circuitry operable for converting data so that it can be transferred between the I/O device and the PowerPC processor despite the differences between their respective Little Endian byte ordering. The PowerPC Little Endian byte ordering within a doubleword is reversed with respect to the true Little Endian byte ordering. In true little Endian byte ordering, bytes 0:3 are associated with addresses 3:0 of the doubleword respectively. Within PowerPC Little Endian byte ordering, bytes 0:3 are associated with addresses 4:7 of the doubleword.

This Sheet Has Been Replaced



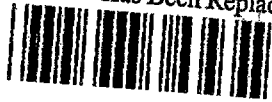
4

THIS SHEET HAS BEEN
REPLACED

The present invention can be configured to operate to convert data in the manner set forth above between any two data processing system devices, wherein the first device is configured to operate with a Little Endian byte ordering that is the reverse of the Little Endian byte ordering of the second device.

The foregoing has outlined rather broadly the features and technical advantages of the present invention in order that the detailed description of the invention that follows may be better understood. Additional features and advantages of the invention will be described hereinafter which form the subject of the claims of the invention.

This Sheet Has Been Replaced



5

THIS SHEET HAS BEEN
REPLACED

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention, and the advantages thereof, reference is now made to the following descriptions taken in conjunction with the accompanying drawings, in which:

FIGURE 1 illustrates the format of a word of data;

FIGURE 2 illustrates a Big Endian storage map;

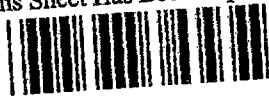
FIGURE 3 illustrates a Little Endian storage map;

FIGURE 4 illustrates a Little Endian storage map within a PowerPC architecture;

FIGURE 5 illustrates an implementation of the present invention;

FIGURE 6 illustrates byte ordering multiplexers in accordance with the present invention; and

FIGURE 7 illustrates a data processing system configured in accordance with the present invention.



DETAILED DESCRIPTION

In the following description, numerous specific details are set forth such as specific word or byte lengths, etc. to provide a thorough understanding of the present invention. However, it will be obvious to those skilled in the art that the present invention may be practiced without such specific details. In other instances, well-known circuits have been shown in block diagram form in order not to obscure the present invention in unnecessary detail. For the most part, details concerning timing considerations and the like have been omitted inasmuch as such details are not necessary to obtain a complete understanding of the present invention and are within the skills of persons of ordinary skill in the relevant art.

Refer now to the drawings wherein depicted elements are not necessarily shown to scale and wherein like or similar elements are designated by the same reference numeral through the several views.

A representative hardware environment for practicing the present invention is depicted in FIGURE 7, which illustrates a typical hardware configuration of workstation 700 in accordance with the subject invention having central processing unit (CPU) 710, such as a conventional microprocessor, and a number of other units interconnected via system bus 712. Workstation 700 includes random access memory (RAM) 714, read only memory (ROM) 716, and input/output (I/O) adapter 718 for connecting peripheral devices such as disk units 720 and tape drives 740 to bus 712, user interface adapter 722 for connecting keyboard 724,



mouse 726, speaker 728, microphone 732, and/or other user interface devices such as a touch screen device (not shown) to bus 712, communication adapter 734 for connecting workstation 700 to a data processing network, and display adapter 736 for connecting bus 712 to display device 738. CPU 710 may include other circuitry not shown herein, which will include circuitry commonly found within a microprocessor, e.g., execution unit, bus interface unit, arithmetic logic unit, etc. CPU 710 may also reside on a single integrated circuit.

In one embodiment of the present invention, CPU 710 is a PowerPC processor manufactured by IBM Corporation.

Referring to FIGURE 5, the invention describes a method to support true Little-Endian data transfers between I/O devices 718 and PowerPC architected processor chips 710 and memory 714 in Little Endian systems. Please refer to Chapter 23: Big vs. Little Endian in PowerPC System Architecture, copyright 1995 by MindShare, Inc., pp. 277-308, which is hereby incorporated by reference herein. In the example herein, I/O device 718 refers to I/O adapter 718 in FIGURE 7; however, any device operating in a true Little Endian mode may take advantage of the features of the present invention. Byte reordering stations 500 are incorporated into the data paths between the I/O and processor ports of the memory controller subsystem 501. Byte reordering will occur on data transfers originating from devices on the CPU bus 712 targeted for I/O devices 718, for data transfers from I/O devices 718 targeted for the memory controller 501, and for memory accesses from I/O devices 718.

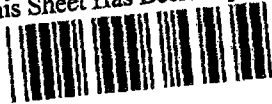


FIGURE 6 illustrates how the byte reordering task is accomplished. Each BE/LE station 500 consists of two 32-bit 2:1 multiplexors 601, 602, which are data doubleword wired. Data is presented to the leftmost 4-byte input [0:3] of each mux 601, 602 in the order it was received from the external buses. The rightmost 4-byte data input [4:7] is wired to the mux 601, 602 in reversed order. When the system is operating in Big Endian mode, the multiplexor select signal LE_ENB is disabled ("0"), the leftmost 4 bytes of each multiplexor 601, 602 (bytes [0:3] of mux 601 and bytes [4:7] of mux 602) are gated to the output and no reordering of bytes takes place. The significance of each byte remains unchanged.

In Little Endian mode, the LE_ENB select is enabled ("1"). The rightmost data mux inputs (bytes [4:7] of mux 601 and bytes [0:3] of mux 602) are gated to the output. The resulting data doubleword will have byte 0 swapped with byte 7, 1 with 6, 2 with 5, and 3 with 4 from the order it was received from the external bus. The data doubleword now reflects a true Little Endian storage image with byte 7 as the MSB and byte 0 as the LSB. Similar mux structures (not shown) are used to adjust the parity associated with each byte in the doubleword. The multiplexor select (LE_ENB) is sourced from a flipflop (not shown) that is either set or reset via software prior to run-time.

The table below further illustrates the above-described operation of multiplexor 601 and 602:

This Sheet Has Been Replaced



THIS SHEET HAS BEEN
REPLACED

LE_ENB	MUX DATA-OUT BYTE ORDER							
'0'	0	1	2	3	4	5	6	7 (Big Endian)
'1'	7	6	5	4	3	2	1	0 (Little Endian)

Although the present invention and its advantages have been described in detail, it should be understood that various changes, substitutions and alterations can be made herein without departing from the spirit and scope of the invention as defined by the appended claims.



THIS SHEET HAS BEEN
REPLACED

CLAIMS

1. A data processing system comprising:
 - a PowerPC processor utilizing PowerPC Little Endian byte ordering;
 - an I/O device utilizing true Little Endian byte ordering; and
 - a memory controller operable for communicating data between said PowerPC processor and said I/O device, wherein said memory controller further comprises:
 - circuitry operable for converting data to be transferred from said I/O device to said PowerPC processor or memory from said true Little Endian byte ordering to said PowerPC Little Endian byte ordering; and
 - circuitry operable for converting data to be transferred from said PowerPC processor or memory to said I/O device from said PowerPC Little Endian byte ordering to said true Little Endian byte ordering.
2. The system as recited in claim 1, wherein said PowerPC Little Endian byte ordering within a doubleword is reversed with respect to true Little Endian byte ordering.

This Sheet Has Been Replaced





THIS SHEET HAS BEEN
REPLACED

3. The system as recited in claim 2, wherein in said true Little Endian byte ordering bytes 0:1 are associated with addresses 1:0 of said doubleword, respectively, and wherein in said PowerPC Little Endian byte ordering bytes 0:1 are associated with addresses 6:7 of said doubleword, respectively.

4. The system as recited in claim 1, wherein said converting circuitry each further comprises:

first and second 2:1 multiplexers, wherein a leftmost 4 byte input of each multiplexer receives said data in a same order as presented on a bus coupling said memory controller to either said processor or said I/O device, and wherein a rightmost 4 byte input of each multiplexer receives said data in a reverse order as presented on said bus coupling said memory controller to either said processor or said I/O device.

5. The system as recited in claim 1, wherein said memory controller may be configured to operate in Big Endian mode.

This Sheet Has Been Replaced



THIS SHEET HAS BEEN
REPLACED



6. A memory controller operable for coupling a first bus device to a second bus device, wherein said first bus device is configured to operate with a first Little Endian byte ordering that is a reverse of a second Little Endian byte ordering of said second bus device, said memory controller comprising:

circuitry operable for converting data to be transferred from said second bus device to said first bus device from said second Little Endian byte ordering to said first Little Endian byte ordering; and

circuitry operable for converting data to be transferred from said first bus device to said second bus device from said first Little Endian byte ordering to said second Little Endian byte ordering.

7. The memory controller as recited in claim 6, wherein said byte ordering is with respect to a doubleword.

8. The memory controller as recited in claim 7, wherein in said second Little Endian byte ordering bytes 0:1 are associated with addresses 1:0 of said doubleword, respectively, and wherein in said first Little Endian byte ordering bytes 0:1 are associated with addresses 6:7 of said doubleword, respectively.

This Sheet Has Been Replaced





THIS SHEET HAS BEEN
REPLACED

9. The memory controller as recited in claim 8, wherein said converting circuitry each further comprises:

first and second 2:1 multiplexers, wherein a leftmost 4 byte input of each multiplexer receives said data in a same order as presented on a bus coupling said memory controller to either said first bus device or said second bus device, and wherein a rightmost 4 byte input of each multiplexer receives said data in a reverse order as presented on said bus coupling said memory controller to either said first bus device or said second bus device.

10. The memory controller as recited in claim 9, wherein said memory controller may be configured to operate in Big Endian mode.

11. The memory controller as recited in claim 6, wherein said first bus device is a PowerPC processor.

12. The memory controller as recited in claim 11, wherein said second bus device is an I/O device.

This Sheet Has Been Replaced





THIS SHEET HAS BEEN
REPLACED

13. In a data processing system, a method comprising the steps of:
providing circuitry operable for coupling a first bus device to a second bus device, wherein said first bus device is configured to operate with a first Little Endian byte ordering that is a reverse of a second Little Endian byte ordering of said second bus device; and
converting, in said circuitry, data to be transferred from said second bus device to said first bus device from said second Little Endian byte ordering to said first Little Endian byte ordering.
14. The method as recited in claim 13, further comprising the step of:
converting, in said circuitry, data to be transferred from said first bus device to said second bus device from said first Little Endian byte ordering to said second Little Endian byte ordering.
15. The method as recited in claim 13, wherein said byte ordering is with respect to a doubleword.
16. The method as recited in claim 14, wherein in said second Little Endian byte ordering bytes 0:1 are associated with addresses 1:0 of a doubleword, respectively, and wherein in said first Little Endian byte ordering bytes 0:1 are associated with addresses 6:7 of said doubleword, respectively.

This Sheet Has Been Replaced





THIS SHEET HAS BEEN
REPLACED

17. The method as recited in claim 14, wherein said coupling circuitry each further comprises:

first and second 2:1 multiplexers, wherein a leftmost 4 byte input of each multiplexer receives said data in a same order as presented on a bus coupling said coupling circuitry to either said first bus device or said second bus device, and wherein a rightmost 4 byte input of each multiplexer receives said data in a reverse order as presented on said bus coupling said coupling circuitry to either said first bus device or said second bus device.

18. The method as recited in claim 14, wherein said coupling circuitry is a memory controller and said first bus device is a processor.

19. The method as recited in claim 14, wherein said first bus device is a PowerPC processor.

20. The method as recited in claim 19, wherein said second bus device is an I/O device.

This Sheet Has Been Replaced

