intel®

Implementing a Low-Cost PCI Bridge and Memory Controller

Application Note (AP-416)

intel®

Application Note (AP-416)

Contents

Revision History

Revision	Revision Date	Description
0.3	Feb. 2000	Initial release.
0.4	June 2000	Intel Confidential status removed.

Application Note (AP-416)

1.0   Introduction

The Intel® 82559ER Fast Ethernet PCI Controller is an integrated Media Access Controller (MAC) and Physical Layer (PHY) solution for 10/100 Mbps Ethernet designs. It succeeds the Intel 82558 integrated MAC/PHY controller for embedded designs.

1.1   Scope

This application note addresses embedded designs using the 82559ER Fast Ethernet PCI Controller without the need to add a PCI bus. It is assumed that the designer has absolute programming control over the operating system and knows what devices will be placed on the PCI bus (in other words, no bus connectors). Therefore, strict PCI compliance is not necessary.

The addition of a PCI bridge and memory controller to an embedded processor system may seem like a difficult task. However, designers will find that it is possible to minimize the PCI local bus functionality and still adequately support the 82559ER.

1.2   Reference Documents

Is is assumed that the designer is familiar with the PCI Local Bus Specification. An understanding of the PCI specifications is valuable prior to integrating the 82559ER in any platform. The following documents were referenced in the development of this document:

- PCI Bus Applications in Altera Devices Application Note 41, Version 1. Altera Corp., 1995.
- Using FPGAs as a Flexible PCI Interface Solution, Version 2.0. Jim McManus, Xilinx, Inc., 1998.
- Optimizing Network Interface Cards for Operation in a Standard High Volume Server. Jato Technologies, 1998.
- PCI Local Bus Specification, Revision 2.2. PCI Special Interest Group, 1999.

The following documents provide background material for an 82559ER embedded solution:

- 82559ER Fast Ethernet PCI Controller Datasheet. Intel Corporation.
- PCI Local Bus Specification, Revision 2.2. PCI Special Interest Group.

2.0 Required PCI Command Support

Table 1 lists all possible PCI bus command types and denotes the required commands for the 82559ER interface. The bus commands are described in the following subsections.

Table 1. PCI Bus Commands for Systems Using the 82559ER

PCI Command	Bridge/Memory Controller[a]	82559ER Controller[b]
Interrupt Acknowledge		
Special Cycle		
I/O Read	Optional	
I/O Write	Optional	
Reserved		
Memory Read	X	X
Memory Write	X	X
Configuration Read	X	
Configuration Write	X	
Memory Read Multiple		X
Dual Address Cycle		
Memory Read Line		X
Memory Write and Invalidate		Optional

a. Column denotes commands initiated by the bridge/controller as a PCI Bus Master.
b. Column denotes commands initiated by the 82559ER as a PCI Bus Master.

PCI bus masters, including the 82559ER, issue bus commands based on the assumption that data in memory is prefetchable. However, data prefetching is not a required implementation in a PCI target.

2.1 I/O Read and Write Commands

I/O Read and Write cycles are used to read from and write to the 82559ER's Control and Status Register (CSR) and its EEPROM space. I/O mapping is optional for the 82559ER. However, since many embedded systems do not use separate memory and I/O spaces, the use of I/O mapping is not recommended.

2.2 Memory Read and Write Commands

Memory Read and Write commands are fundamental PCI memory commands and contain burst accesses of varying lengths. The 82559ER issues memory read cycles to transmit control blocks in memory and memory write cycles to command blocks, receive frame descriptors, receive buffer descriptors, receive memory data buffers and for statistical counters. The 82559ER has the ability to perform long burst length transfers, but TCP/IP segmentation leads to relatively small data transfers in many networks.

2.3   Memory Read Multiple Command

Memory Read Multiple cycles are similar to memory read cycles. However, memory read multiple cycles are encoded indicating that the 82559ER device expects to retrieve data beyond the end of the first cache line in a data cache. The 82559ER memory read multiple is a performance enhancing command used in accessing receive data buffers. Caching is too complex for low-cost PCI bridge and memory controller implementation. This command can be used as an alias to the memory read cycle.

2.4   Memory Read Line Command

Memory Read Line command is a performance enhancement to the 82559ER and enables it to access an entire cache line. The 82559ER issues this cycle to access command blocks, transmit block descriptors, receive frame descriptors and receive buffer descriptors. Again, caching should be avoided in a low-cost design. The PCI specification only requires that the target accept the command and behave as if it is a Memory Read.

2.5   Memory Write and Invalidate Command

The Memory Write and Invalidate (MWI) command also improves performance when a write-back caching bridge/memory controller is present. When this command is enabled, it is used to write receive packet data into buffers. The MWI command can be enabled or disabled in the 82559ER by setting the cache line size bits in the cache line size register to any value other than 8 or 16. When the feature is disabled, the 82559ER device uses the memory write command instead.

The 82559ER has the ability to terminate read accesses on a cache line boundary when possible. This feature also relies on the programmed value of the cache line size bits. For best performance in a non-caching system, this capability should be disabled by setting the Read Align Enable bit in the 82559ER to 0.

3.0   Performance versus Simplicity

The basic 82559ER embedded design includes address translation blocks, buffer FIFOs, Direct Memory Access (DMA), and arbitration logic. There are a couple of factors to consider, such as memory requirements and system complexity (based on the devices sharing the PCI bus with the 82559ER) with regards to system performance. Figure 1 shows a generalized embedded design using the 82559ER.

Figure 1. Generalized Embedded Design with an 82559ER
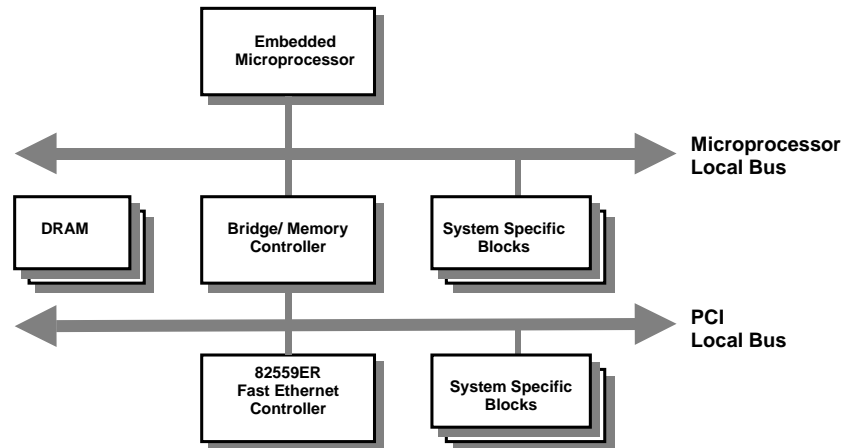
One reasonable solution may include a controller that is comparable in cost to PCI devices supported.

Recent PC chip sets do not include a cached memory bridge/controller. However, most low-cost designs do not require cached memory unless several PCI masters access memory and high performance is needed.

When multiple PCI bus masters exist on the bus, master latency timers are used to guarantee fairness. The 82559ER device has a latency timer. In a system where the bridge/memory controller and the 82559ER are the only PCI masters, the vast majority of the data flow will be initiated by the 82559ER device and it may not be necessary to implement a master latency timer in the bridge. If it is needed, a leftover timer may be available in another part of the system.

One way to reduce cost and complexity in a bridge/memory controller is to avoid implementing parity generation and checking on the PCI bus. Without parity, the system does not comply with the PCI specification, but compliance may not be required in a "controlled" design. Reporting of parity errors on the Parity Error (PERR#) and System (SERR#) signals can be disabled in the 82559ER. However, if data parity is missing, the Ethernet controller constantly reports a data parity error in its PCI Configuration Status Register. The 82559ER does not terminate any bus cycles due to parity errors and software can ignore the detected parity error bit.

Other design recommendations for implementing a low-cost bridge/memory controller include:

- Use the 3 KByte transmit and receive FIFOs in the 82559ER instead of the buffers in the bridge/memory controller.

- Keep the DRAM bus width to 32 bits.

- Minimize the number of DMA channels.

- Do not provide concurrent main processor and PCI device access to memory.
- Minimize the number of devices on the PCI bus.
- Avoid the use of PCI expansion card slots.
- Minimize power-down logic in the bridge/memory controller.

4.0   Off-the-Shelf Solutions

If the system PCI bus control requirements exceed a low-cost design implementation, an off-the-shelf PCI controller can be used. PCI bridge/memory controller devices are available to support members of the following microprocessor families:

- i960
- QED* RM52xx (MIPS)
- IDT* 4xxx/5xxx (MIPS)
- NEC VR4300* (MIPS)
- IBM PowerPC*
- Motorola PowerPC
- Motorola 68000*
- Hitachi SuperH*

A second alternative is to use a microprocessor with a built-in PCI bridge/memory controller, such as the Intel® 80960VH (i960VH) Embedded-PCI Processor. The i960VH processor is ideal for many embedded computing applications such as Virtual Private Network (VPN) appliances and Ethernet expansion boards.