Linking PCI based processor platforms

Martin Weymann

Creative Electronic Systems, 70 route du Pont-Butin, CH 1213 Petit-Lancy, http://www.ces.ch/CES_info/Welcome.html, email: ces@lancy.ces.ch

PCI bus is widely used as a local bus for both VME processor boards and desktop workstations. The PVIC concept described below has been developed by CES to allow PCI based processors to be connected into clusters of up to 15 nodes spanning up to 200 meters. While preserving the full PCI throughput (128 MByte/s at 32-bit,32 MHz) for block accesses, the transparent mapping of remote PCI adresses minimizes latencies and drastically reduces the software overhead. An integrated DMA controller allows to perform complex data transactions with minimal CPU load. Broadcast and multicast cycles, interrupt dispatching, mailboxes, a reflective memory and global semaphores provide the hardware support for efficient inter-processor communication. The PVIC complements network oriented interconnections such as ATM and FDDI which may cover a larger numer of nodes and larger distances at the cost of increased software complexity and larger latencies.

Key words: PowerPC; PCI; VME; PMC; PVIC; ATM; FDDI

1 Introduction

The data acquisition system of a HEP experiment typically contains a large number of processor nodes working in parallel in order to collect, format and analyse event data. Processors may be embedded, such as VME processor boards, or they may be desktop workstations. For high end processors in both domains PCI bus[1] is becoming the predominant choice for the local bus. PCI bus, while providing a fairly high bandwith locally (128 MByte/s at 32-bit,32 MHz), is confined to the spatial dimensions of a processor board or PC-motherboard.

For interconnecting processors on the distance scale of a HEP experiment, two different types of connections have been used: *networks* such as Ethernet that route packets of data between nodes or ports and *transparent mapping mechanisms* such as VIC bus (ISO/IEC 26.11458) that allow the local CPU to access part of a remote address space directly on a cycle-by-cycle basis.

PCI based processor boards, such as the CES PowerPC based VME processors RTPC 8067 or RIO2 806x[3] can be connected to high bandwith networks via PCI Mezzanine Cards (PMC[2]). CES offers PMCs for ATM and FDDI, a PMC for 100 MBit Ethernet is forthcoming (8468,8466,8478[3]). Integrated with an RIO2 processor and drivers these network connections form the PowerLink[3] family of products. TCP/IP support allows to migrate traditional network applications smoothly to a new performance scale. Lower-level protocols can coexist with TCP/IP where low latency and high efficiency for small packets is required.

In order to reduce latencies even further and to increase bandwith, CES has, in collaboration with the Electronic System Design group of the Rutherford Appelton Laboratories Technology Department, developed the PVIC concept.

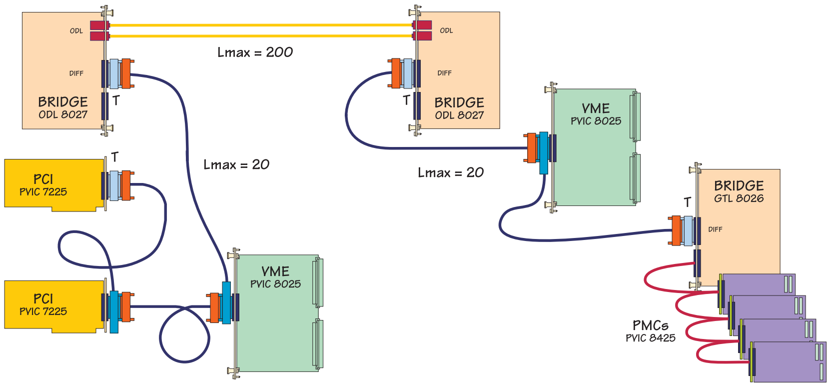

Fig. 1. PVIC system example

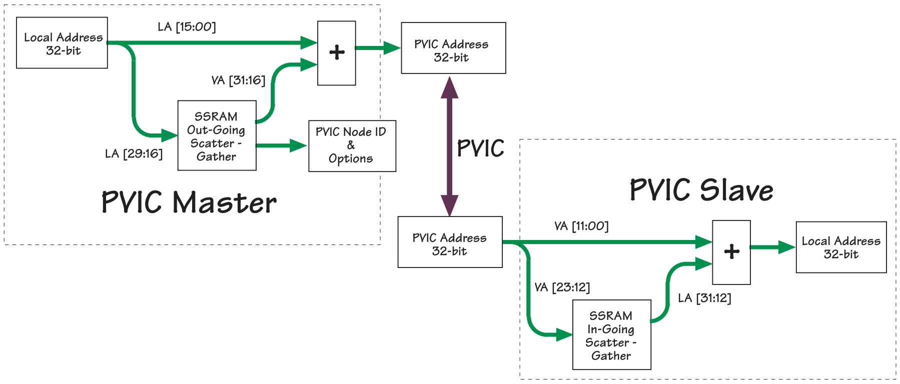

Fig. 2. PVIC mapping mechanisms

2 The PVIC concept

The PVIC implements a transparent mapping mechanism between PCI bus segments in much the same way than VIC bus offers transparent mapping between different VME crates. It can connect PCI based processor boards to each other and to PCI based desktop workstations on three different distance scales: below 1 meter (e.g. within a VME crate, flat cable, GTL+), up to 20 meters (PECL differential copper connection), and up to 200 meters (850 nm multimode fiber). Figure 1 shows a system using all three connection types to interconnect the different members of the PVIC family of boards: a PMC (8425), a short format PC-type PCI card (7225) and a group of 6U VME cards implementing bridges between the different PVIC transmission media (8026) and between PVIC and VME (8025) [3].

As shown in Figure 2, *Address translation* is handeled on two levels: the PVIC master first translates a local PCI address into a PVIC address, which the PVIC slave then, after an optional second mapping stage, uses to generate a remote PCI address. The first mapping stage uses a page size of 64 KBytes. The second stage, which is particularly useful for broadcast operations where it allows to re-map a common PVIC address individually for each node, uses 4 KByte pages. Selective broadcast (multicast) can be implemented by selecting on a per-page basis wether or not a broadcast cycle is accepted. The mapping tables for the address translation are kept in a synchronous SRAM which is part of each node.

Each node also contains a fast dual ported SRAM (DPRAM), part of which is used to implement *write and read buffers* for transparent access. This allows to interleave local PCI cycles with ongoing PVIC transactions and helps avoid deadlock conditions when a node is accessing the PVIC while another PVIC cycle is already pending to be translated into a local PCI cycle. PVIC read cycles are implemeted as split transactions, a read request generated by the local node followed by a read response generated by the remote node.

Table 1
PVIC cycle times (ns)

PVIC cycle type	GTL+	DIFF (1m)	DIFF (10m)	FIBER (100m)
Single WRITE	200	200	300	1200
Single WRITE RFM	200	200	200	200
Single WRITE Broadcast	300	300	400	1300
Single READ	2000	2000	2100	3000
Burst WRITE 256 bytes	2200	2200	2300	3200
Burst WRITE 256 bytes Broadcast	2300	2300	2400	3400
Burst READ 128 bytes	3000	3000	3100	4000
Semaphore Test&Set	500	500	600	1600

The remaining part of the DPRAM is used to implement a *Reflected Memory (RFM)*. RFM read cycles are handled locally while write cycles to the RFM are converted into PVIC cycles that update the RFMs of all nodes coherently. A message FIFO implemented in parallel with the RFM can be used to log the lower address part of a RFM access and to generate an interrupt. This mechanism can be used to implement *mailbox signalling*. Each location of the RFM, if accessed in a special mode, can be used to implement a *global semaphore*. If the semaphore reservation was successful, the access will return bit#31 as 0 and at the same time set the bit to 1 in all RFMs.

A *DMA controller* capable of handling *chains*, linked lists of data blocks to be transfered, is part of each PVIC node. Each DMA controller handles local PCI reads and PVIC write operations only but can be controlled from any node in the system. Thus in order to read data from a remote node via DMA the remote nodes DMA controller is used. DMA chains are kept in the nodes local SRAM. A few entries in that SRAM are reserved for DMA transfers triggered by special conditions such as local PCI interrupts. Used together with mailbox signalling, these special chains can be used for *interrupt dispatching*.

Table 1 shows PVIC cycle times for a range of distances. Due to buffering, write cycles are much faster than read cycles. Burst cycles must be used to obtain the full bandwidth: a read cycle has a latency of 2 μs to transfer the first byte while the remaining 127 bytes can be transfered in 1 μs.

The PVIC architecture is designed to minimize software overheads associated with data transfers. Most PVIC operations can be implemented by an application level library. The use of drivers can be restricted to interrupt handling and low-level PCI address assignment. This helps to keep PVIC software portable between different platforms. After an initialization phase access to a remote nodes PCI address space is fully transparent and thus requires no overhead at all. DMA transfers can be triggered with a single write instruction if the chain has been prepared in advance.

References

[1] *PCI Local Bus Specification*, ver. 2.0, April 1993 (PCI Special Interest Group, M/S HF3-15A, 5200 N.E. Elam Young Parkway, Hillsboro, Oregon 97124-6497).

[2] IEEE, *Draft Standard Physical and Environmental Layers for PCI Mezzanine Cards: PMC*, P1386.1/Draft2.0, April 1995 (The Institute of Electrical and Electronic Engineers, Inc., 345 East 47th Street, New York, NY 10017, USA).

[3] CES, *CATALOGUE, Edition 97-98* (Creative Electronic Systems, 70 route du Pont-Butin, CH 1213 Petit-Lancy, http://www.ces.ch/CES_info/Welcome.html, email: ces@lancy.ces.ch, Tel: +41-22-792 57 45 Fax: +41-22-792 57 48).