By Colin Davies,
Product Manager Systems Business Unit,
Radstone Technology.

PPzero - Multiprocessing PCI over the VMEbus P0 connector

This article reviews existing "first generation" VME/PCI expansion techniques and considers the limitations which prevent peer multiprocessing. PPzero a new VME/PCI architectural concept from Radstone which provides a genuine multiprocessing PCI upgrade capability will then be introduced. Crucially this implementation of PCI through the VMEbus P0 connector represents a logical development and an upgrade path for existing boards and systems. The technology provides a second system data path which has the additional benefit of providing access to state of the art PCI/PMC innovations.

FOREWORD

At the VMEbus board level PCI is now an established interconnection technology both as an on card local bus and as an expansion mechanism in the form of PMC mezzanine cards.

At the system level, though, the picture is not quite so clear. The versatility and cost effectiveness of PMC solutions has resulted in a number of expansion methodologies both proprietary and open. Each has its merits but none, so far, has attained the capabilities of the on card solutions. They lack the vital software and RTOS support necessary for a system level solution.

This article reviews existing "first generation" VME/PCI expansion techniques and considers the limitations which prevent peer multiprocessing. PPzero a new VME/PCI architectural concept from Radstone which provides a genuine multiprocessing PCI upgrade capability will then be introduced. Crucially this implementation of PCI through the VMEbus P0 connector represents a logical development and an upgrade path for existing boards and systems. The technology provides a second system data path which has the additional benefit of providing access to state of the art PCI/PMC innovations.

FIRST GENERATION PMC EXPANSION CONCEPTS

The "plug and play" nature of PMC solutions coupled to the cost effectiveness of the PMC cards themselves fuelled the development of PMC expansion schemes. A VME processor card could support perhaps 1 or 2 PMC sites depending upon the degree of ruggedization. By using "carrier cards" systems were able to support additional PMCs, located in adjacent VME slots, connected by PCI to the host processor's PCI local bus.

Several carrier card schemes evolved e.g. Motorola (PMCspan) and Radstone (PCC1, PMCC2). In each case PMC cards are mounted on 6U carrier cards which are literally bolted onto the host processor card. PCI interconnect is by connectors between the host and the carrier card.

Although simple this scheme does work and can support 2 to 4 extra PMCs mounted on up to two carrier cards. The stack of cards structure does have one significant disadvantage, even with depopulated 5 row VME connectors the backplane insertion force may be significant! Injector/ejector handles can mask this problem from the user to some degree but electrical interconnects between the boards have to be a concern if all boards do not "seat" at exactly the same moment. In this respect rugged products, equipped with metalwork to stiffen the boards, tend to avoid the sequenced insertion problem better and prevent stresses being translated through delicate PCI connectors.

BASIC PCI EXPANSION ARCHITECTURE

The carrier card scheme introduces a fundamental PCI expansion architecture. A host processor card and a single carrier card are sufficient to understand the mechanisms and limitations and this is illustrated in Figure 1.

PCI loading considerations dictate the use of a PCI to PCI transparent bridge such as the Intel 21150 device. The host processor is able to initialise the PCI expansion bus and allocate memory space as appropriate for each PCI site. The nature of the transparent bridge is that the primary (local) and secondary (expansion) busses must be clocked together with the primary and secondary addresses being the same. Hence the 21150 bridge must provide a clock source for all "downstream" PCI devices and this will be derived from the host's PCI bus clock.

Functionally the host can become master of the secondary PCI bus in order to access PMC cards on the carrier. Similarly each PMC may become bus master in order to support DMA into the host's memory.

ALTERNATIVE PHYSICAL ARRANGEMENTS AND PPZERO

PCI is a well defined interconnect technology and within the bounds of the electrical interface specification there is scope for a number of physical implementations. The carrier scheme described above is just one example but the PCI connection route could be achieved by an alternative means. Connection through a backplane connector, with PCI routed by a backplane tracking scheme, avoids the insertion force prob-

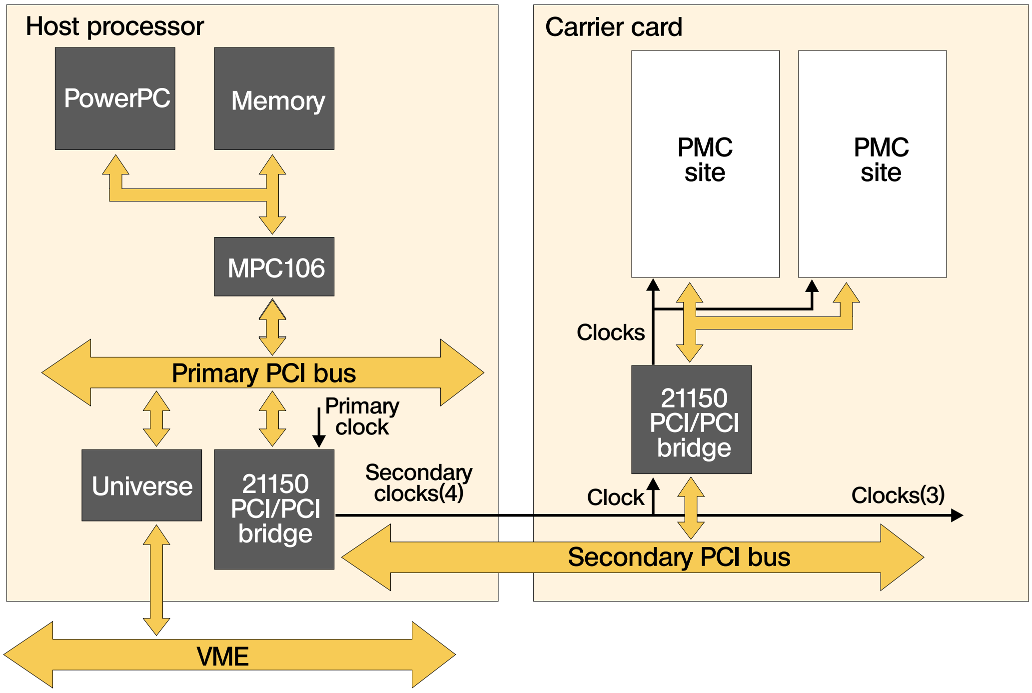

Figure 1. Fundamental PMC carrier architecture.

lems of multi-card schemes.

The VMEbus P2 user defined pins could be used at the expense of pins traditionally used for I/O and would therefore not be a simple upgrade route applicable to the majority of implementations. In addition efforts to standardise this (VITA 27, PCI on P2) have now been discontinued as cross talk issues dictated non standard 47 ohm termination resistors which limited the PCI bus speed to 25 MHz.

PPzero recognises that there is a defined mapping between the PMC I/O connector (P4) and the more recently introduced VME P0 connector (ref 2, ref 3). If an appropriate bridge device is mounted on a PMC card with its I/O through the P4 connector then PCI will be routed to P0 and the backplane. This can then be utilised on any card equipped with the PMC/P4 to VME/P0 connection although track lengths must be both short and of equal length to prevent signal skew. This is illustrated in Figure 2. In practice a half length PMC module is large enough to accommodate the circuits and Figure 3 shows a practical implementation, the Radstone PMCPCI.

Options exist for the connection of the secondary PCI bus between adjacent VME/P0 connectors. For development an off the shelf "tracking PCB" may be conveniently soldered to the rear of the VME/P0 connectors across the required slots (Figure 4). For higher volume and production requirements a backplane incorporating dedicated tracking for the secondary PCI bus is normally more cost effective and a number of configurations are already available as COTS items.

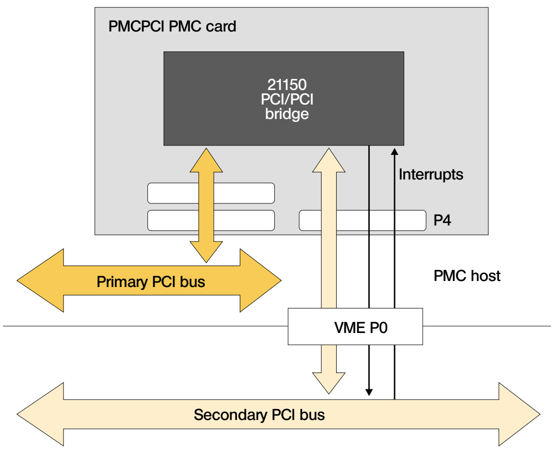

Figure 2. PCI through VME PO

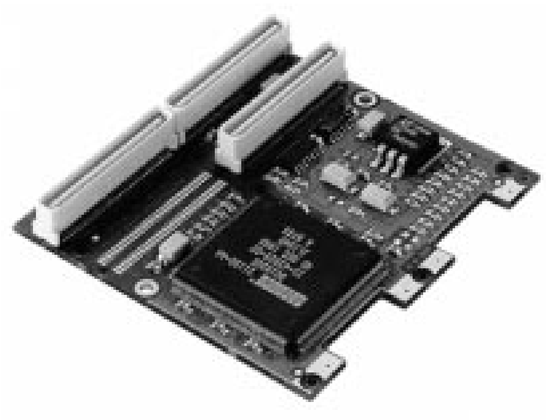

Figure 3. Radstone PMCPCI.

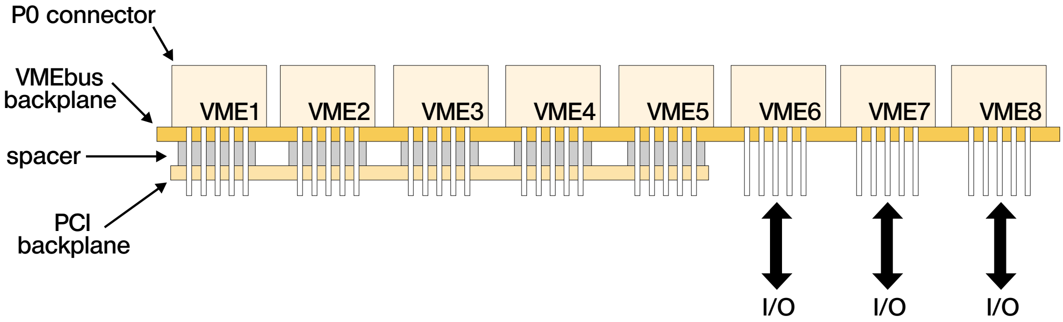

Figure 4. VME/PO tracking PCB

MULTIPLE PROCESSOR IMPLEMENTATIONS

Multiple processors connected using transparent bridges such as the 21150 are not an option, today's processor bus architectures do not support the required clock synchronisation. The recent introduction of non-transparent PCI to PCI bridge devices, such as the Intel 21554, provide a partial solution through support for different primary/secondary clocks and address mapping between the two busses. Figure 5 illustrates the resultant architecture.

Processor 1 utilises a 21150 transparent bridge to provide clocks, bus control and interrupt handling. Processor 1 initialises the secondary PCI bus by scanning PCI nodes and allocating memory space to processors 2 and 3 which each utilise 21554 non transparent bridges. With this architecture processor 1 may access processors 2 and 3. Similarly processors 2 and 3 may access each other. Unfortunately neither processor 2 nor processor 3 may access processor 1 (1). We have multiple processors and NOT multiprocessing because each processor can not access a peer processors main memory. Processor 1 requires the bus control functionality of a transparent bridge combined with the address mapping and multiple bus clock capabilities of the non transparent bridges used on processors 2 and 3.

PPZERO PROVIDES PEER MULTIPROCESSING SUPPORT

Figure 6 highlights a Radstone PMCPCIM module based upon the 21554 non transparent bridge and with the features required to Master the secondary PCI bus. When fitted to processor 1 (Figure 5) this gives a genuine multiprocessing environment with each processor having the ability to access memory on any of the other cards. In practice the "processor" cards

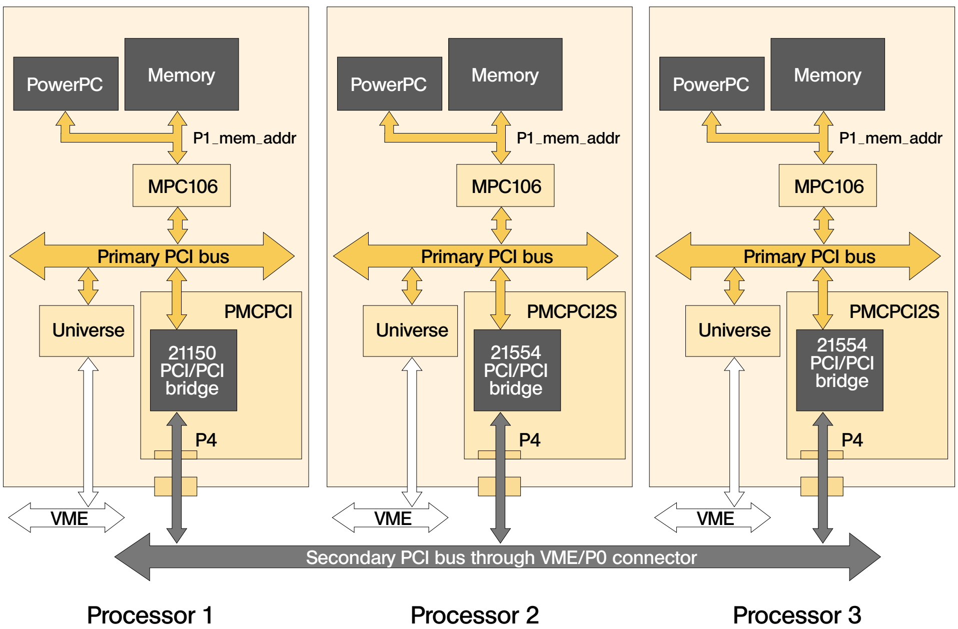

Figure 5. Multiple processor architecture.

AD VMIC

To understand the problem with Processor 2 or Processor 3 (Figure 5) access to Processor 1 main memory it must be understood that the 21150 transparent bridge does not support address translation. Primary and secondary PCI addresses must be the same. Hence Processor 2 or 3 must use an address P1_mem_addr on the secondary PCI bus to gain access to Processor 1. Registers within the 21150 must be set up to decode this address.

The problem comes when a VME cycle attempts to access memory at P1_mem_addr. Naturally the memory controller should decode this and respond. Unfortunately the 21150 transparent bridge will also decode the address and respond creating a conflict.

A non transparent bridge supports the concept of different slave and master addresses. Hence if Processor 1 employed a 21554 non transparent bridge then Processor 2 could access memory at P1_mem_addr but the memory map of Processor 1 could be configured to include a different access address for the secondary PCI bus.

may be DSP or graphics modules creating a wide range of system architecture options. The PMCPCIM also provides the PMC expansion capability of the previous PMCPCI.

Radstone's PPzero architecture is supported by a range of three PMC modules, enabling the full range of architectural options for a secondary PCI bus realised through the VME/P0 connector.

PPZERO SOFTWARE MODEL

In the model using just the PMCPCI adaptor, Radstone initialisation code performs a full PCI bus scan and dynamically allocates PCI resources for all onboard devices and all PMCs acessable via PPzero. This goes down to every leaf-node of every bus in any generic heirarchy of bridges, and deals with memory space, I/O space, plus interrupt routing and multi-function devices. A key benefit of this method is that BSP support does not require re-coding when extra PMCs are plugged in, (many BSPs on the market have 'hard-wired' PCI addresses and interrupt routings).

This same initialisation software operates with a PMCPCIM adaptor, but further software deals with the multi-processing aspects of this product. In a mixed system containing PPzero enabled peer processor boards, only the (single) board fitted with a PMCPCIM can control any expansion PMCs. This is because the PCI interrupts are routed via this adaptor.

For peer multi-processing via PPzero boards are configured to know whether they are connected to P0 via a PMCPCIM or a PMCPCIS adaptor. The board fitted with a PMCPCIM (the Master) performs a full PCI scan and allocation at initialisation as described above. Boards fitted with a PMCPCIS (Slaves) only allocate the resources of their internal PCIbus. The Master board will then also configure the Slaves such that they each have bi-directional windows onto the PPzero bus, and also have correct PCI interrupt routings back to the Master itself. The precise order of initialisation and the resulting window setups vary depending on which firmware or O/S is being supported.

On the standard real-time O/Ss such as VxWorks and LynxOS, the windows which are setup will correspond to those required by the standard 'VME backplane driver' protocols of these O/Ss. These driver protocols are mapped onto PPzero hardware so that existing standard O/S interfaces can continue to be used, but now also on the PCIbus. Both higher levels in the O/S (e.g. VxMP for VxWorks) and customer applications operate over PPzero in the same manner as they currently do over VME.

This model is greatly assisted by full hardware support for snooping of PCI masters on all Radstone PowerPC boards. This allows straightforward support of shared multi-processing data areas in any manner defined by the O/S. Some boards on the market do not support this feature. The normal VxWorks model for backplane protocol data movement is to use processor transfers. However Radstone also provide a PCI to PCI DMA capability which can be used for greater performance and efficiency in transfering bulk data. The LynxOS backplane protocol model intrinsically supports DMA transfers.

APPLICATIONS

Two multiprocessing bus structures provide a candidate solution for many system architecture problems. Figure 7 provides an illustrative system configuration which can be used to consider how the technology can be used effectively.

Module	Bridge	Use
PMCPCI	21150 transparent bridge	Entry level PMC expansion schemes only
PMCPCIM	21554 non transparent bridge + Master circuits	Master PCI controller for multiprocessing architecture used on most "upstream" board, plus PMC expansion schemes
PMCPCIS	21554 non transparent bridge	Slave PCI controller for multiprocessing architecture

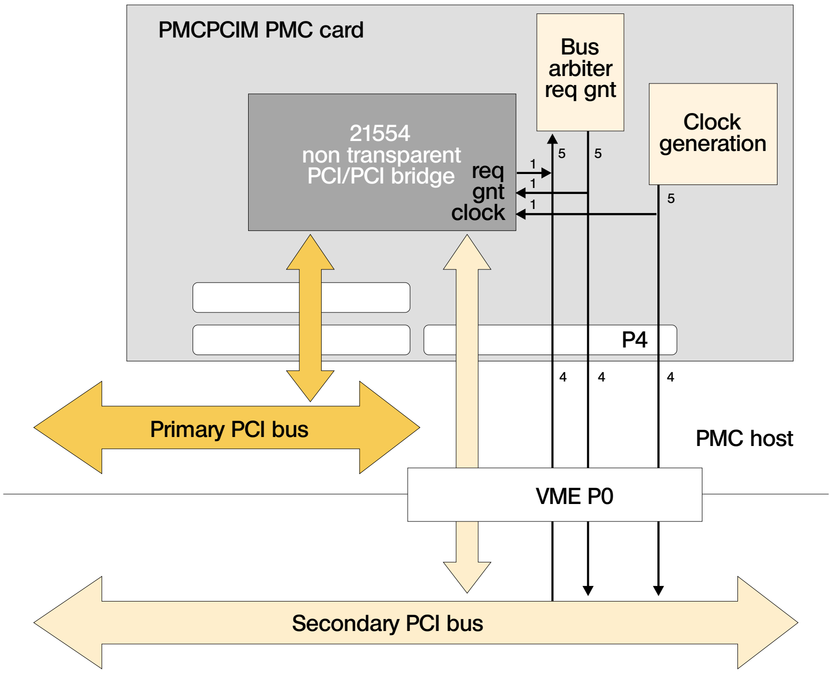

Figure 6. Multiprocessing PCI controller.

Figure 7 also introduces a PMC carrier card the P0CC1. The P0CC1 utilises a 21150 transparent bridge as introduced in Figure 1 except that the PCI connection is now through VME/P0 with control from the "upstream" PPC4(1) which handles interrupts via the PMCPCIM module. Loading and bus characteristics limit the secondary PCI bus to 5 slots, one "Master" and four processing slots.

Performance considerations

Many data intensive graphics or DSP applications suffer from the performance limitations of a single bus structure. This could be a raw bandwidth issue or the side effect of a requirement for determinism, which forces the DMA block size to be reduced sufficiently to allow an acceptable processor response time and in so doing lowers the effective DMA rate.

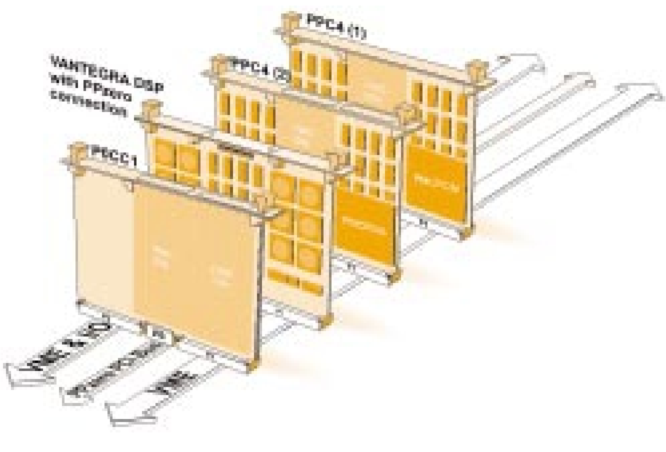

Figure 7. PPzero multiprocessing PCI system configuration.

Considering Figure 7, the DSP processor could utilise the PCI bus on P0 to DMA data directly into the main memory of the controlling processor PPC4(1). Parameters such as block size can be adjusted to realise optimum performance in the knowledge that other tasks or sub-systems may use VME bandwidth and resources at the same time.

In practice it is possible to add more system wide bandwidth than a single 33MHz PCI bus. Within a 21 slot VMEbus backplane several sub-systems may coexist, each with its own smaller PCI segment across VME/P0 connectors.

Determinism

Legacy VME cards tend not to exhibit the performance or functionality of today's implementations and as a result often dominate VMEbus response time considerations. A PCI secondary bus provides a high performance data path with quantifiable buffering capabilities and latency.

Second data path

Fault tolerance is a complex subject involving multiple disciplines. A vital architectural concept is that of a secondary bus structure able to probe, isolate or re-initialise a suspect node. In this respect a PCI secondary bus provides a solution at the task level by providing a mechanism by which an active processor can be made to re-initialise a locked out VMEbus controller for example.

Program risk

PPzero is a genuine upgrade path for existing VME/PMC hardware. Similarly, the controlling software drivers layer directly onto existing RTOS implementations. As a result there is a low incremental component cost to the addition of extra bandwidth or PMC expansion during a program. The technology effectively de-risks a development by providing insurance against creeping complexity, the Achilles heel of any program.

CONCLUSIONS

Commercially successful products are often the result of an evolutionary development program and not a revolutionary one. The PPzero multiprocessing PCI expansion scheme described in this article provides a genuine upgrade path for existing environments and an architecturally sound option for new developments. All concepts introduced are entirely in keeping with mainstream VMEbus hardware and software developments creating options for future technology insertion and other measures utilised to minimise through life costs in COTS applications.

All system components required to implement PPzero multiprocessing PCI through the VME/P0 connector are commercially available today. This includes not only the PMC controllers but also the development backplane components and a range of tracked backplane configurations. To be successful though this can not become a proprietary development. The technology is robust and of benefit to the entire VMEbus community. As a result it is the objective to standardise P0 pin outs and other issues which relate to interoperability. A logical way to do this is by means of an extension to the VITA 35-199x standard (PMC-P4 pin out mapping to VME-P0) ■

By Colin Davies, Product Manager Systems Business Unit, Radstone Technology.

REFERENCES

1. IEEE P1386.1, Draft Standard Physical and Environmental Layers for PCI Mezzanine Cards: PMC, Draft 2.0 4-Apr-95.
2. VITA 35-199x, Draft 0.4, September 19 1999. PMC-P4 pin out mapping to VME-P0 and VME64x-P2.
3. IEEE P1386; Draft Standard for a Common Mezzanine Card Family: CMC, Draft 2.0 4-Apr-95.
4. Draft Standard for Conduction Cooled PMC Modules, VITA 20 - 199x Draft 1.10 October 1999.
5. PCI Local Bus Specification, Revision 2.2.