By Remy David,
Director of Marketing and Technology,
GESPAC S.A.

Why CompactPCI will replace VME

In just a year, the CompactPCI bus has risen to prominence in the board industry worldwide, prompting users to compare it with VME and to wonder if it could replace VME in many applications. There are two ways to compare buses. The first is to look at the intricate details of the bus signals, bus transceiver technology and timing diagrams. From this point of view, VME (and VME64x) fares as an outstanding technology with plenty of growth potential. The other way is to see what silicon components, system and software facilities a given architecture offers to simplify system design. From this perspective, PCI and CompactPCI offers unique possibilities thanks mostly to the broad silicon support and "system-oriented" concepts like Plug and Play. While the installed base of applications and customers ensures VME many more years of existence, these powerful force are already at work to make CompactPCI the dominant industrial OEM bus at the end of the decade.

Historically, new industrial bus standards typically have originated from a number of sources. As semiconductor companies introduced new microprocessors, they also built a series of board and system products designed mainly to simplify the use and speed the acceptance of their chips. These buses often were a direct extension of the microprocessor local bus onto a backplane. This is how Multibus and Versabus - which eventually evolved into Versa Modules Europe (VME) - came to be. Another source for buses is the computer industry. When IBM introduced the PC in 1981, it needed a scheme for allowing add-on functions to be easily added and removed. The PC/ISA bus was not an end in itself but merely a needed feature in an open system design.

History includes a number of smaller, innovative companies who independently defined bus architecture. For example GESPAC defined the G-64 bus in 1979, Pro-Log the STD bus in the mid 70's, and more recently Ampro defined the PC/104 bus. These buses have done well in the market because they were aimed at specific needs and were relatively independent from any given chip or computer maker.

An active, but commercially less successful source of buses has been institutions like the IEEE. Notable ventures include the STE bus and more recently Futurebus. While none of these buses was ever endorsed by a large company or gained market acceptance, they have contributed to the development of new technologies that eventually made their way in many of the buses that are in use today.

It is fair to say, however, that the most important contribution to the bus/board market have been done by the Computer and the Semiconductor industries.

BOARD LEVEL VS. SYSTEM LEVEL BUSES

Buses in system hierarchy

Buses are elaborate technologies and specialists have dedicated careers to refine them. They can also be emotional subjects; as recent "battle of the buses" have shown. But one must not lose sight of the big picture. Buses are essentially communication schemes designed to build systems. Figure 1 shows a layered model of a typical system. The Physical layer (1) defines the board format and the connector type and positions. The Electrical layer (2) defines the transceiver technology, pin assignment, clock speeds, etc. ... The Protocol layer (3) describes how elementary transfers, arbitration and responses to interrupts are made. Layer 4 deals with the way boards are mapped in the address space and how they are detected and configured. This and the following layers are no longer specific to the bus but deal with system design issues. For example, layer (5) provides recommended architecture models for selecting and interconnecting peripherals in a manner that ensure compliance with a Reference Platform Standard. The Hardware Abstraction Layer (6) is provided to allow different hardware implementations while maintaining compatibility at the higher layers. BIOS or OpenFirmware software reside on this layer. Next the Operating System layer (7) rests as the foundation for what the system is ultimately intended for: run the user Application software. Eventually all layers have to be addressed for a system to work.

Difference between computer and semiconductor maker's buses.

Buses developed by chipmakers, like VME, tend to focus on the elementary functions of the system. In the

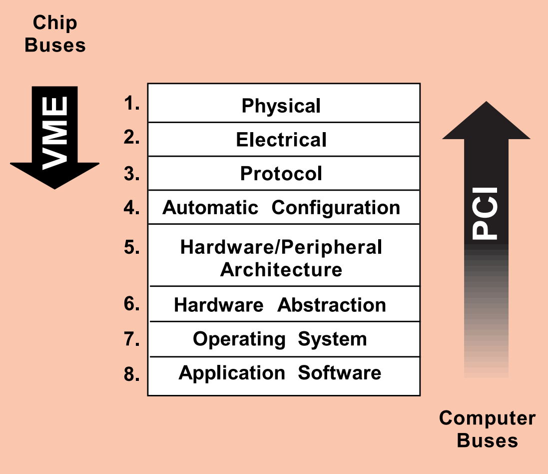

Figure 1. Layered model of the elements of a typical open system

early days of the microcomputing industry, in fact, it was enough for a bus to ensure that boards plugged in the same backplane merely communicate. Most VME board users can feel the limitations of this ground-up approach: Today, there are no two VME systems from different vendors that can run the same off-the-shelf software. Most user application software requires significant adaptation when moving it from one VME system model to another. VME board vendors too must spend a disproportionate amount of energy porting and maintaining operating systems on each of their product.

Buses originating from the Computer industry, on the other hand, have taken a much more comprehensive approach: They are more concerned with how users will be able to easily add boards to a system and how they will be integrated in the system logically as well as physically. For example, since the beginning. ISA boards have featured on-board BIOS ROMs that allowed them to integrate seamlessly the Operating System as soon as the card was plugged into its backplane. The bus itself is typically not the driving element of the architecture.

Today, chip, board and computer makers and users alike understand the importance of looking at the ”system” picture. A system must be built so that it can run standard software and not the other way around. The PCI bus was born out of this comprehensive view and is an extraordinary result of collaboration of chip, software and computer companies.

Indeed, never before has an entire industry laid aside its differences to embrace a new technology as in the case of PCI. Today, PCI is at the core of all Pentium, PowerPC and Alpha systems. Millions of PCI systems are sold yearly, leading to a flurry of low cost silicon.

PCI is now the only practical way to build systems. Chip sets invariably include bridges to PCI and peripheral devices for graphics or networking, for example, are now only available in PCI version.

The PCI system reference model

As discussed earlier, PCI is a ”system level” bus. It not only specifies how data are to be exchanged between devices, but it also suggests an architecture model for assembling CPU, cache, local memory and peripherals. Although this model, shown in figure 2, is not firmly set in the PCI specification, all available silicon is designed around it.

As a result, not only do Pentium and PowerPC VME boards today integrate PCI, but in fact PCI is the main internal bus of these boards. The interface to VME bus is done via a PCI to VME bridge chip.

The diminishing role of VME as a backplane bus

Another interesting trend to analyze is the regressing use over time of the VME bus itself in VME systems (see figure 3). When the VME bus was first defined, its speed was as high or higher than this of fastest 68000 or 68010 CPU it could support. So memory or peripherals could invariably reside on the CPU board or anywhere on the backplane with no difference in performance. As CPU clocks increased, external memory boards were less used in favor for on-CPU-board memory. More recently, even faster CPU speed have forced, mostly PCI based peripherals to move on mezzanine modules on the CPU board.

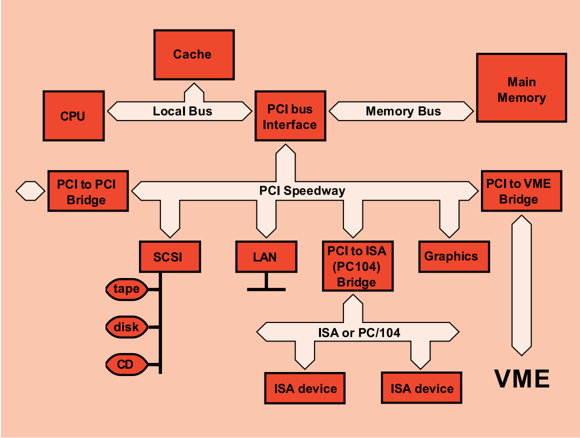

Figure 2. Recommended internal architecture model of a PCI based computer

Fortunately, high-density chips have made it possible to squeeze this additional functionality on the CPU board and on piggyback modules. However, this has weakened the nature of VME systems, whose modularity was a key asset.

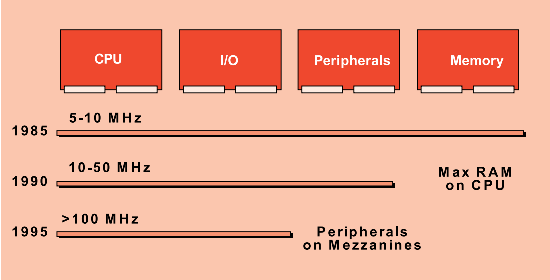

Figure 3. Use of the VME backplane bus with time as processor speeds increase

The 68000 legacy

While faster CPU speed has been the driving force to move function off the VME backplane, another factor has recently come into play: As a chip maker's bus, VME is an extension of the 68000 microprocessor signals which is growing increasingly incompatible with modern peripheral silicon, particularly PCI: Byte ordering and interrupt structure of the 68000/VME bus greatly differs from the native features of Pentium, PowerPC and other RISC processors. As a result, the conversion from local bus to VME, and from VME to the peripheral bus that would be needed to talk to a device on a separate VME board, adds significant overhead which almost always introduces software incompatibility. This overhead and resulting incompatibilities can be eliminated by moving the function directly on the CPU board, or by using a backplane bus, like PCI and CompactPCI, which is directly compatible with the native processor bus and its supporting chip sets.

The VME64x specification, currently under development at the VSO, boosts the bus's speed to 160 Mbytes/s (using 64-bit transfers). Not only is this speed little more than half of PCI but because of its 68000

Copyright 1998 by Real-Time Magazine 98-4 (http://www.realtime-info.be)

legacy, VME64x does not support the system and software features that would ease the design of peripherals that automatically map themselves into the system. Consequently, VME64x will not reverses this trend away from the backplane bus to mezzanine buses and it will never grow to become a major system-level backplane bus.

THE COMPACTPCI BUS

For all its power, cost and software benefits, many industrial users have resisted using the mechanically inferior PC architecture in their application. The CompactPCI architecture finally brings the mechanical benefits of VME with the power and software support of the PC these users. CompactPCI is a variation of the standard PCI bus where PCI is the backplane bus for 3U or 6U board level microcomputer products (figure 4). The CompactPCI specification is the result of the collaborative work of members of the PCI Industrial Computer Manufacturer's Group (PICMG). PICMG executive membership includes GESPAC, Compaq, Intel, Force Computers, Motorola, ProLog, DEC, IBM, Hewlett-Packard, and Ziatech to name only a few.

CompactPCI is an adaptation of the PCI Specification for industrial and embedded applications requiring a more robust mechanical form factor than desktop PCI. CompactPCI uses standard Eurocard mechanical components and metric connector technologies to provide a system optimized for rugged applications.

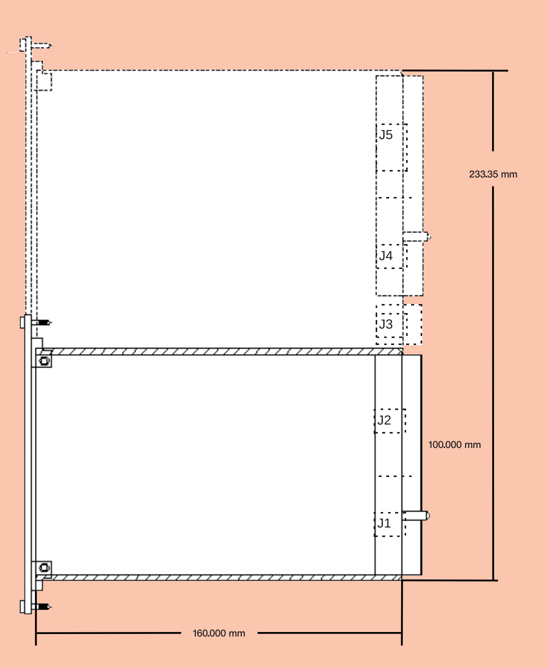

Figure 4. Physical configuration of a 3U or 6U CompactPCI boards and its connectors

CompactPCI is electrically compatible with the PCI Specification 2.0 (33 MHz), allowing low-cost PCI chip sets to be utilized in a mechanical form factor suited for rugged environments.

Ad Trenew

COMPACTPCI COMPARED TO VME BUS

Using CompactPCI for connecting to Peripherals and Memory.

As its name implies, PCI (Peripheral Component Interconnect) is best suited for interfacing peripheral devices that typically require configuration and the exchange of large blocs of data. PCI is a synchronous bus clocked at 33 ns. All exchanges are burst mode transfer according to the diagram in figure 5, leading to a maximum transfer rate of 133 Mbytes/s in 32-bit mode and 266 Mbytes/s in 64-bit mode. This deterministic, high throughput benefits such functions as Graphics, networking (fast Ethernet, ATM) and any peripheral that needs to move large blocs of data such as image acquisition and processing, DSP, analog converters with large buffer memory, or intelligent slave processor. By contrast, VME is an asynchronous bus capable of data transfers at 40 Mbytes/s in 32-bit mode.

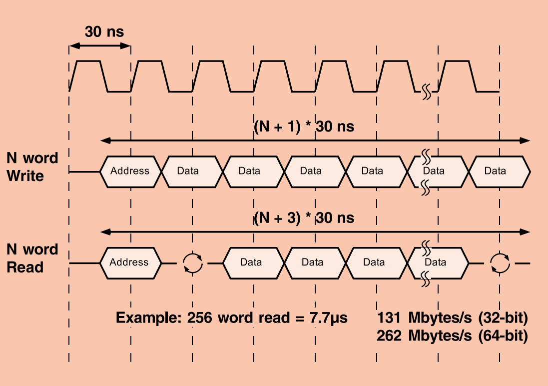

Figure 5. Maximum data transfer rates on PCI

Using CompactPCI for Simple I/O

One of the main objections to the generalization of CompactPCI is that PCI is a synchronous bus optimized for burst mode transfers and that it would perform miserably for single word accesses. The timing diagram of figure 6 shows the theoretical access times for single word exchanges.

While it is true that single word reads are as much as four time slower than top speed burst transfers, the resulting speed is in the range of the current VME specification. In all cases, this speed is far in excess of what is typically required for simple, single register devices such as Digital or Analog I/O interfaces.

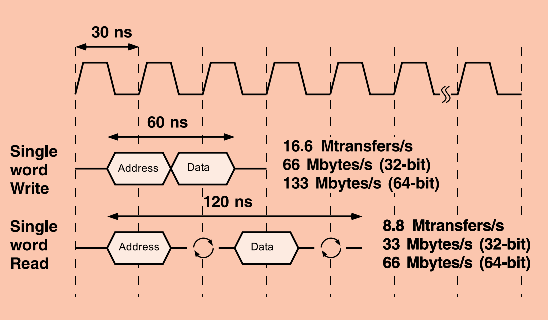

Figure 6. Timing diagram for single-word PCI read and write transfers

CompactPCI is limited to 8 slots in a system. However, the bus can be expanded using PCI to PCI bridges and GESPAC will soon demonstrate 14 slot and 19 slot backplanes. While VME is capable of connecting up to 21 slots without bridges, CompactPCI makes it possible to build I/O board on 6U cards where the top connector is used for routing I/O lines. By contrast 32-bit VME boards need both connectors of the 6U board for the bus.

Interrupting devices on CompactPCI

Interrupts are a very important element in real-time systems - the principal application of VME systems. For best performance, the interrupt scheme used on the bus should closely match this of the CPU chip. Unlike memory access cycles which differ little from one CPU type to another, interrupts handling vary significantly from one processor to another, and, more significantly, from one "system" architecture to the next.

On a 68xx0 processor, VME's 7 interrupt levels are mapped directly into the CPU which contains built-in hardware to resolve priorities and special instructions to quickly determine the source of interrupt, save register context and jump to the appropriate interrupt service routine.

When a non 68xx0 processor is used on VME, the VME interrupt mechanism needs to be translated to the native Intel model (in the case of a Pentium CPU), adding extra latency. For instance, interrupts on Intel processors are edge sensitive and byte ordering is reversed (little endian).

RISC processors like the PowerPC, lack the multiple hardware interrupt lines and special instruction that were present in the 68xx0. On a RISC processor, like the PowerPC 603/4, there are only two hardware interrupt lines and the user (or OS developer) must take care of acknowledging and prioritizing interrupts, detect the interrupting device and jump to the proper service routine. Custom hardware is often added to a RISC system to assist the CPU in these steps.

PCI and CompactPCI does not bring a miracle solution to the problem of interrupts on non-68xx0 processor, however, it establishes a processor independent scheme for interconnecting peripheral devices. PCI and CompactPCI specifies 4 level-sensitive interrupt levels and an interrupt acknowledge bus transaction (figure 7).

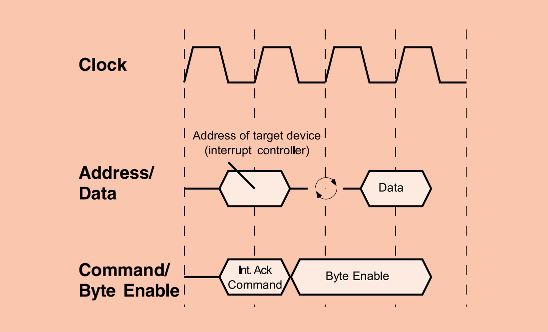

Figure 7. Interrupt acknowledge transaction on PCI

Copyright 1998 by Real-Time Magazine 98-4 (http://www.realtime-info.be)

PCI does not specify interrupt priorities or how interrupt should be routed to the CPU. While it would theoretically be possible to build custom hardware to handle interrupts in any specific way, interrupts on PCI systems are fully defined in reference platform specifications such as the PC/AT standard or the PowerPC Reference Platform standards.

Multiprocessing on CompactPCI

CompactPCI does support multiprocessing. The multiprocessing scheme is different from VME in that PCI specifies that the main CPU board hosts the arbitration of the bus and that it has absolute control of the system. In that sense, it is not possible to add a second processor of equal priority in an adjacent slot. The bus, however, can be equipped with ”bus masters” which are typically Intelligent I/O functions, but which may also be standalone coprocessors. Figure 8 shows the routing of the bus arbitration lines between the arbiter and the peripherals. A separate set of bus request and bus grant lines are routed from the arbiter and the each bus master.

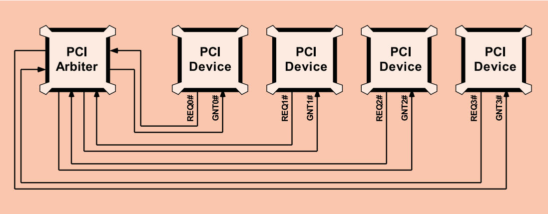

Figure 8. Multiprocessing arbitration on CompactPCI

When a master wants the bus it requests it from the arbiter, which grants it if available. If more than one device wants the bus at the same time, a priority algorithm is implemented in the arbiter to ensure fair access to the bus. Once a master has the bus, it can initiate Direct Memory Address transfer from and to any other device on the bus, including the host CPU.

Most Pentium and PowerPC chip set are capable of supporting Symmetric Multi Processing (SMP). In SMP systems, several identical processors share the same local bus (before the PCI bridge) and can share a computing load transparently. SMP requires that the operating system knows how to distribute the execution of threads among the available processor, as is the case of WindowsNT. SMP is not an architecture proper to PCI and CompactPCI, however, chip sets that support SMP typically also support PCI.

Hot Swap Peripherals and I/O on CompactPCI

One of the goals of the CompactPCI is to include Hot Swap as part of the standard. This feature is critical to several PICMG members involved in the telecommunication industry and a working group has been setup in April 1996 to write a specification. In parallel, a group of hardware and software companies has formed within the PCI SIG to define a Hot-Plug architecture, mostly for PCs used as servers.

The CompactPCI Hot-Swap scheme that serves as foundation to the current effort calls for staged backplane connector pins that cause the board to be powered, with the PCI signals in high impedance before the PCI pins make contact with the backplane. When the board is fully inserted, the PCI bus is activated and the board initialized.

The combined effort of the PICMG and PCI SIG will result in a specification that permits not only the exchange of board without powering down the system but also to ensure that a board is properly recognized and initialized after it has been replaced.

The CompactPCI connectors and bus interface

PCI is a multiplexed bus that requires only 47 signals for its implementation. As a result 32-bit peripheral boards can be built using a smaller connector (J1). 64-bit peripherals can be built on a 3U board using the full size connector. 64-bit VME requires 6U-board size. Figure 9 shows the CompactPCI connectors.

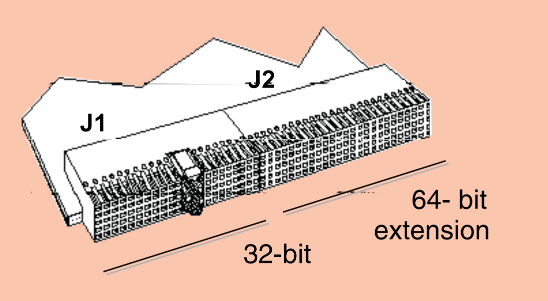

Figure 9. the two parts of the CompactPCI connector

Because of the very large volumes of the PC market, PCI is built directly into the peripheral chips at practically no added cost. Furthermore, PCI peripherals don’t require external buffers and the connection between the chip and the connector is direct. These factors combine to make CompactPCI products less expensive than their VME equivalents.

MIGRATING FROM VME TO COMPACTPCI

Migrating from 3U VME

CompactPCI is expected to precipitate the decline of 3U VME first. This is because the 3U-form factor is particularly ineffective from a cost and performance point of view. Since they are only using one connector, 3U VME boards only support 16-bit transactions. To implement 32 or 64-bit processors, both the P1 and P2 connectors of a 6U board must be used. Furthermore, the small 100 x 160 mm board format is very small and the glue logic and buffers needed to interface to the VME bus take away precious real estate from useful system functions, thus making the 3U boards less cost effective. CompactPCI, by contrast, support full 64-bit PCI transfers off a 3U board and no special logic or buffers are necessary to connect PCI peripheral chips on to the CompactPCI backplane.

The 3U form factor is very compact making it inherently more robust and vibration proof than 6U boards. With CompactPCI, current users of 3U VME will find an effective way to upgrade their system to modern PowerPC or Pentium processors with room to grow to

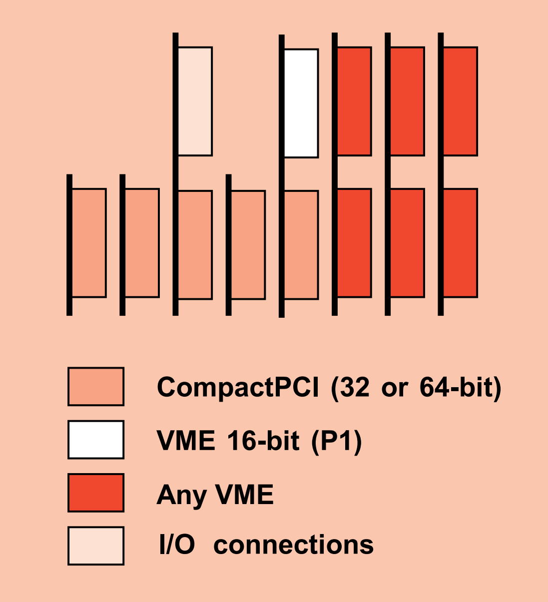

Figure 10. Combined CompactPCI/VME systems

full 64-bit systems.

Migrating from 6U VME

Since 6U is supporting 32 and 64-bit transfers, the most important benefit of CompactPCI is its superior software integration with the Intel-Microsoft model which is essential for CPU and system level peripherals (video, networking, mass storage...). On the other hand VME still enjoys a significant lead in the availability of I/O functions. Because it shares VME's mechanical attributes, CompactPCI is well suited for building boards that combine both buses. When the CompactPCI specification was being drafted, the bus connector was placed on the bottom part of the 6U board (where VME P2 stands) to ensure that boards with VME on P1 and CompactPCI on P2 could be built. Dual bus board could easily designed and several companies are working on such designs. This scheme is the one that offers the best openness to the existing base of VME products. However, because only the VME P1 connector is implemented, only 16-bit boards may be used. Figure 10 shows possible system configurations using this combined CompactPCI and traditional VME architecture.

In February 1996, Force Computers, the acknowledged #2 VME vendor announced its support of the CompactPCI architecture and proposed a scheme for combining CompactPCI and VME64x on a single board. To achieve this, the entire VME64x bus, which is normally mapped on two 160-pin connectors has been re-mapped on a 220-pin, 2mm metric connector identical to the CompactPCI connector, located on the top part of the board (J4 an J5 on figure 4). Using this scheme, connecting to traditional VME boards requires a special backplane to re-map the VME64x signals on P1 and P2. However, this method makes for systems that offer additional possibilities. Figure 11 shows possible system configurations using combined VME64x and CompactPCI.

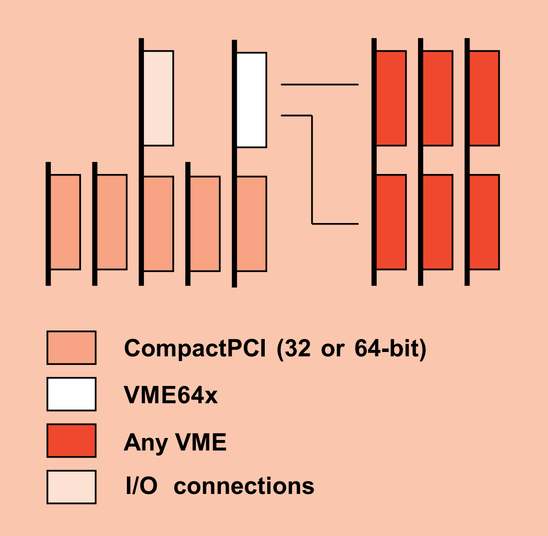

Figure 11. Combined CompactPCI/VME64x systems

CONCLUSION

PCI has now so much momentum and acceptance that it is practically impossible to build a modern computer without it. Pentium and PowerPC VME boards all have an internal PCI bus. PMC modules offer a limited way to take advantage of PCI in VME boards. Unlike PMC, CompactPCI is a backplane bus with the same modularity as typical plug-in VME boards. Although CompactPCI is designed to build complete systems using CompactPCI-only boards, it can easily be implemented on the P2 of a 6U VME board to expand the capabilities of VME. This hybrid dual-bus architecture offers a migration path to "CompactPCI-only" systems, which are more powerful, less expensive, have more room for growth and run more off-the-shelf software than VME. ■

REFERENCES

[1] PICMG, CompactPCI specification V.1.0.

[2] PCI SIG, PCI local bus specification 2.0, April 1993

[3] Apple, IBM, Motorola, PowerPC Microprocessor Common Hardware Reference Platform

[4] Motorola Semiconductor, PowerPC™ 603 RISC Microprocessor User's Manual.

[5] Wendell Smith, Paul Nelson, and Amy Dyson, Motorola, AN1267 PowerPC 603™ Hardware Interrupt Latency In Embedded Applications

---

Rémy DAVID has been graduated with a Master of Science in Electronic Systems Design at the college of Aeronautic in Cranfield (U.K.). Joining GESPAC in 1993, he has been the driving force inside the company for selecting the CompactPCI bus standard as the strategic architecture for the new generation of embedded controllers. Thanks to this choice, the company has achieved a significant growth in the board level market. Today, he acts as the marketing & technology manager and he is an active member of both PICMG Exec (USA) and PICMG Europe.

Copyright 1998 by Real-Time Magazine 98-4 (http://www.realtime-info.be)