

New Bus meets The Old World

By Kim Vu

CompactPCI is a modern, very high performance industrial bus based on the standard PCI electrical specification and built on a rugged 3U or 6U Eurocard packaging. CompactPCI, for all its advantages and benefits, still lags behind VMEbus because of its limited the number of peripherals available for many standard applications. Many system integrators have a difficult time finding the right CompactPCI interfaces for their CompactPCI CPU systems.

System integrators are offered a very limited selection in choosing CompactPCI interfaces. Moreover, many currently available VME boards contain features that are not replicated in the CompactPCI world. Faced with this dilemma, system integrators can opt for a dual-bus architecture solution. The problem can be simply solved through a combination of the two buses: CompactPCI and VMEbus. Why not CompactPCI and Multibus? The answer is that CompactPCI and VMEbuses share the same 6U form-factor (see Figure 1). Actually, since its birth CompactPCI has paralleled the mechanical attributes of the VMEbus. In addition, the VMEbus has been proven as an outstanding technology since its inception over 15 years ago, and it still has plenty of growth potential. A combination of CompactPCI and VMEbus promises a perfect marriage for industrial applications and provides system integrators a window to a large and diverse system market.



Figure 1

Bridging CompactPCI to VMEbus

The already established similarities in form-factor and mechanical composition provide a firm foundation to start the combination of the two buses. However, issues remain to be

resolved: the PCI-VME bridge hardware design, the chassis with dual backplane construction, and the software driver to support VME peripherals. Of course, the challenge extends beyond merely piecing together two buses. Solflower has addressed these issues and has developed a solution that is low cost, easy to use, and in most cases, plug-and-play. The solution is designed to be compatible with current VME hardware and software peripherals.

PCI-VME bridge hardware

Tundra Semiconductor has designed the Universe II in a single chip. Only a few components such as buffers and oscillators need to be added to build the complete PCI-VMEbus Bridge.

The original architecture and functionality of the Universe is based on Tundra's SCV64 controller. Thus, it makes it possible to consider headache-free utilization of the Universe to develop a dual-bus system. Figure 2 shows the block diagram of the Universe II chip.

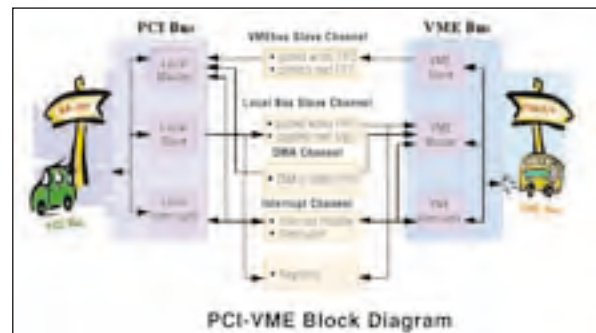


Figure 2

The Universe Bus-Bridging chip provides 64-bit data transfers between the VME and PCI bus. It includes an on-board DMA engine that can move data across the buses at speeds of up to 50 Mbits/sec. The complete Bus-Bridge accompanied by the associated buffer logic are laid out in a single 6U VME card without any cabling or daughter card attached.

Chassis for integration

The dual-bus system consists of a 3U CompactPCI backplane, a 6U VME backplane, and the Solflower single 6U Bus-Bridge card. All of these components are arranged in a 19-inch standard rack mount and provide seven CompactPCI and six 6UVME slots.

The 64 bit CompactPCI bus J1/J2 are positioned on the bottom of the board, as opposed to the VMEbus J1 being located at the top and J2 at the bottom. To accomplish the bus merger, the one feasible option entails the use of the 2-mm connector at the position J4 and J5 on top of the CompactPCI to map the VME signals. The specially designed VME backplane provides six slots for the standard 6U format and one 220 pin 2 mm connector, located on the left, to adapt the J4/J5 signals and to re-map the VME signals on P1 and P2. Finally, the two buses can be mounted together onto a common 19-inch rack-mount chassis, providing seven CompactPCI slots on the left side and six VMEbus slots on the right side.

Figure 3 displays the combination solution for the dual bus system. Partially viewed on the left side is a standard CompactPCI backplane, which begins with slot 1 to the left and ascends in number to slot 6 on the far right, where the CP1006 Bus Bridge resides. Shown in full view on the right side, the VME backplane situates slot 1 to the far left by the Bus Bridge Board, as a system slot, and ends with slot 6 to the far right. This dual-bus architecture permits greater flexibility in mixing and matching the latest CompactPCI technology with popular VME peripherals.



Figure 3

Software support

Without software support, the Bus Bridge does not facilitate many applications. Any system built with the Bus Bridge must consider a degradation of performance or change to the application software. That is, unless the Bus Bridge is supported by a device driver, which is transparent to the applications.

The CP1006 is designed to meet both these goals: no degradation of performance and no application software change. It is capable of supporting Solaris applications by providing plug-and-play for all existing VME Sun4 device drivers.

Many technical issues arise and need to be discussed when an integrator wants to have a Sun4 VME application run on the new Sun PCI-based SPARCengine CP1200, CP1500, or an Intel x86 Single Board Computer. Fundamental differences are addressed through space mapping, interrupt handling, and DMA structures and the ENDIAN of the Intel x86 and the SPARC architectures.

In PCI, there are three types of address spaces: I/O, Configuration, and Memory space. The VMEbus, on other hand, differ-

entiates address spaces by way of decoding AM codes. All of these processes can be covered by hardware, namely the Universe, but management of the size of mapping space involves further consideration. According to the PCI specification, each PCI device should have an ID, base address and the length of occupied spaces. However, if the Universe wants to cover all the VME address spaces, then no more address space would be available for other devices on the PCI bus. Furthermore, if the VMEbus has already been mapped to the PCI space but does not utilize all those spaces, it would represent an enormous waste of resources for the whole system. PCI space is abundant (in the range of 1 Byte to 2 Gbytes), but it is constantly divided with other devices; these include the on-board devices, and the space assignment is dynamically controlled from the Solaris kernel. If the space is not allocated correctly, the system could unpredictably fall into a panic mode.

The CP1006 nexus driver is designed flexibly enough to map all the spaces when (and only when) the application needs, and then adjusts the size to fit the empty space on the PCI bus. The following example demonstrates how the CP1006 driver works.

Once the CP1006 driver is installed, it examines the PCI configuration space to discover which addresses are already occupied and, thus, how much memory is available for accessing VMEbus devices. If other PCI devices are installed in the same PCI space, the amount of PCI space available for VMEbus access is logically reduced. However, the CP1006 driver can, in many cases, work by manipulating holes in the PCI memory space caused by other devices. The CP1006 driver accomplishes this task by using up to 8 slave image windows for mapping address ranges on the VMEbus.

Figure 4 illustrates the CP1006 maps to the free PCI spaces. If a VME device calls the un-map function, the PCI space becomes free again. In this example, the CP1006 PCI-VME Bridge shares PCI bus with a PCI video card and on-board Ethernet/SCSI interfaces.

The CPU I/O assigns the PCI physical addresses for accessing VMEbus address space by using hardware base and bound registers. These base and bound registers permit CP1006 to define space on the PCI bus for its use, space which was not detected when the device was initialized by the Sun Open Boot PROM. The Solflower CP1006 Bus Bridge occupies 4 Kbytes of space in the PCI memory space, and after initialization, the CP1006, in essence, becomes bigger than it used to be. There is no way to tell the Open Boot PROM about this changed condition.

Specific to this circumstance, the CP1006 driver can exclude a certain range of PCI addresses for mapping. This is caused by possible occurrences that another CompactPCI card resides in a certain PCI memory range but does not communicate with the Open Boot PROM via its PCI configuration base register.

After the CP1006 driver is loaded, the VME driver can be attached to the nexus CP1006 as a leaf driver, through configuration file specifications and then setting the parameter "parent=pvme" (pvme is the device driver name of the CP1006 Bus Bridge).

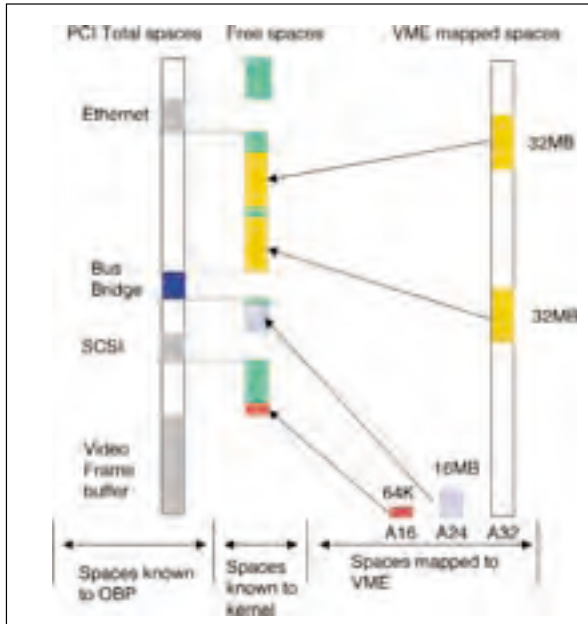


Figure 4

The CP1006 features two types of DMA transfer. One is called slave DMA, where the DMA transfer is controlled from a DMA engine on the VME board. This mode allows the most commonly written Sun4 VME driver to cooperate with the new PCI-based machine – no change, period. The second DMA mode is initialized from an on-board DMA engine, which resides within the Universe chip. The features, such as linked list DMA and programmable transfer length, provide the maximal performance that can be archived through the Bus Bridge.

The byte-ordering issue exists due to the differences in architecture between the VMEbus and the Intel x86 processor. It requires special hardware to avoid byte-ordering conflicts, because the Universe does not cover this Endian conversion. For example, if a Pentium X86 is on the CompactPCI system slot, and a Motorola CPU is on the VME side, during data transfer across the VMEbus, the data at the lowest byte address would be assumed to be the least significant. In actuality, it is the most significant. However, this poses no problem if the 32-bit data is written and read by the same type of processor architecture, since the swapped data will re-swap. For this reason, byte swap logic is available as an option to correct this Endian conversion issue.

Hot swap – a hot topic

Compact PCI specification R2.1 allows hot swap for the CompactPCI interface. It involves both hardware and software to maintain continuous system operation. VME does not have that feature, but the CP1006 allows VME cards in the VME sub-bus to be inserted or removed without shutting down the CPU system. Although this does not equivocate a full hot swap feature as defined in the CompactPCI spec, it allows the system integrator to replace the VME interfaces while Solaris is still running on the CompactPCI bus. This advantage saves a great deal of time, since neither CPU power-down is required, nor is Solaris reboot necessary. This hot swap process is quite simple and requires only that the driver be unloaded. The

power of the VMEbus is turned off in an orderly fashion before it can be removed from the system. When the VMEbus is powered back on, the driver can be reloaded with a single command, and the application can run again. Here is the best part: the user does not pay extra for this hot feature – it is a standard feature and ready to use.

Conclusion

CompactPCI is seen as a complement to VMEbus. Applications such as an industrial control and/or a medical instrument can be solved with CompactPCI or VME components. Integrators prefer to use CompactPCI solutions where the hardware and software is available, but also need VME interfaces where a CompactPCI device is not yet available. Bridging the two worlds produces a very good solution. Further, the creation of such an economical and easy-to-use solution fulfills every Engineer’s dream. Solflower has achieved it with the CP1006, and has answered the question that every integrator asks: “How can I integrate my VME interfaces on the new CompactPCI system without any expensive engineering work?”

The solution we have discussed above can be applied to other platforms, such as Windows NT or VxWorks. The same hardware bus bridge and chassis construction can be used without change. Bus bridge driver software, which is available from several vendors, may be obtained to support the complete system.



Kim Vu is the founder and President of Solflower Computer, Inc. Prior to founding Solflower he worked as design engineer for Sun Microsystems focused on the VME IPI disk controller. He also worked as Hardware designer for System Industries, developing storage add-on subsystems for DEC mini computer. Before moving to the USA (1983) he was a field application engineer at Tewidata in Munich, Germany. Kim has a BSEE from Muenchen Polytechnikum, Germany.

References:

- VME64x/2mm on CompactPCI PICMG 2.2, April 11, 1997
- CompactPCI Specification PICMG 2.5 R1 April 1998
- SPARCengine CP1200/1500 data sheet, Sun Microelectronics 1998
- PCI_VME User’s and Installation Guide, Solflower Computer, Inc.1998

PRODUCT FOCUS AUTHOR

Kim can be reached at:
 Kim Vu
Solflower Computer, Inc.
 3570 Ryder Street
 Santa Clara, CA 95051 USA
 Tel: 408-982-8680, ext. 27
 E-mail: kvu@solflower.com
 www.solflower.com