Developers creating large and complex embedded systems can often achieve their goals by building on the framework of a standard backplane bus. Two standards have emerged as the dominant choices for such systems: VME and CompactPCI. Both have similar physical and performance specifications. Both have the support of numerous manufacturers that offer a range of processors, memory, and peripheral functions. On the surface they appear to be enough alike to compete for the same market. Yet significant differences exist, and understanding them will assist in choosing the right bus for a given application.

Understanding these buses starts with knowing their history. The VMEbus (Versa Module Europe) was first introduced in 1981 and within a few years became the dominant bus in the backplane industry. Its contemporary competitors, such as FutureBus and Multibus, have since all but disappeared. Since 1994 VMEbus has held nearly half the market share for embedded computer boards.

VMEbus, under the control of the VME Standards Organization (VSO), has been a stable platform. Its first significant change was the definition of a 64-bit version in 1995, nearly 14 years after the initial specification was released. The first mechanical change was the introduction of the VME-64X specifications in 1998. Even then, the VSO defined connectors so that the same board could be used on either legacy VMEbus or VME-64X backplanes (although a VME-64X board on a legacy VMEbus backplane loses some I/O functionality). Such concern for backward compatibility along with VME’s historically long mean time between significant changes has allowed users to migrate their existing systems relatively easily from one specification to another.

CompactPCI grew out of the personal computer’s internal PCI (Peripheral Computers Interface) bus. In 1992, the PCI Special Interest Group (PCI-SIG) formed to support PCI’s push into industrial applications. Because of its computer origins, however, the original PCI bus used edge connectors and had limited room for user defined I/O lines. The establishment, in 1994, of the PCI Industrial Computer Manufacturers Group (PICMG) led to specifications that adapted PCI technology for use in industrial and telecommunications applications. PICMG’s first specifications still used the personal computer’s PCI/ISA form factor (PICMG 1.x), but then along came PICMG 2.x, which includes the definition of CompactPCI for Eurocard-based, rackmount applications.

In its early history, CompactPCI was unstable and went through major changes as it tried to adapt PCI to industrial needs, while trying to differentiate itself from VME. Because of these changes, users of PICMG 1.0 had to redesign whole systems when PICMG 2.0 was released. Although the pace is now slower, CompactPCI specifications still vary significantly with every new revision. The last major change was the release of PICMG 3.0, which was introduced in January 2003.

**Competitors in different arenas**

The creators of CompactPCI sought to meld the low-cost components of the personal computer bus with the high-reliability of passive-backplane systems such as VME. In most systems, the card-cage is buried inside the system in such a way that it is difficult to access. Therefore, repairing any failure of this board requires removal of the entire assembly, which typically causes the system to be down for a relatively long time. With a passive backplane, one that has no or few active components, the assumption is that this part’s Mean Time Between Failure (MTBF) is one of the highest in the system, so difficult access is not a concern. CompactPCI mimicked VME in its choice of connector types, physical structure, and card sizes, prompting many users to see the two as direct competitors for the same applications.

Despite this apparent competition, the two buses found their place in different market segments. The telecommunications industry embraced CompactPCI while defense and automation markets used VME. This has led to a divergence in product offerings. For instance, one can find many more motion controllers and image processors on VME than on CompactPCI. On the other hand, CompactPCI offers many more E1/T1 and switch applications than VME.

However, there are other differences between CompactPCI and VME beyond the availability of different board types. Mechanical differences in the backplane connectors have implications for system reliability. The connector differences also affect the availability of user-definable I/O on the systems.

Comparing 6U boards (the most popular size) of both VME and CompactPCI quickly reveals the mechanical differences between the two buses. VME-64 defines two onboard I/O connectors (J1 and J2) and the VME-64X extension defines a third (J0). Both J1 and J2 are DIN connectors, each of have three rows of 32 male pins and two rows of 32 edge connector pads for a total of 320 lines. J0, the I/O connector added by the VME-64X extension, is a high-density connector with 5 rows of 19 female pins, adding 95 pins for a total of 415 lines.

CompactPCI uses five high-density connectors (J1 through J5) with a total count of 535 lines. The connectors on the board are female, with the male pins located on the backplane. CompactPCI thus has more I/O signals available on a board, but they come at the cost of lower overall system reliability.

This reliability assessment stems from the fact that the connection of a board into a backplane is blind, i.e., you do not see the connectors mate when you push the board into its slot. If a board or the connectors are not fully aligned, or if the male pins are not straight, inserting the board could bend the pins without the user knowing. These pins are fragile and if they bend, then in most cases they will break if you try to straighten them. On VME boards, most of the male pins are on the board, which is relatively easy to replace. As a result, the system suffers little downtime for connector failures.

On CompactPCI boards the male pins are all on the backplane. An error means that the backplane will have to be replaced—a much more time-consuming task. PICMG has responded to this potential problem by defining another type of connector, type AB, which has additional keys to help
align the board and backplane connectors. Using boards and backplanes with this type of connector will help ensure higher system reliability.

I/O pins a factor
With bent-pin concerns addressed, the CompactPCI connector scheme clearly has an overwhelming advantage if user I/O signals are needed. VMEbus definitions occupy all but 223 of the available I/O lines, while CompactPCI has more than 400 user-definable pins available. This difference becomes especially important if one of the optional, application-specific bus structures is used on the system. For instance, the H.110 bus supports telephony applications on CompactPCI and uses most of the I/O pins of J4, but still leaves 300 lines for user definition. While not formally defined for VME, the H.110 bus would occupy most of the available lines, leaving almost nothing for users. Similarly, adding PCI to the VME backplane would use all of J0, leaving only 128 user-defined lines.

The number of user-definable pins each bus offers is a useful insight into the overall system flexibility allowed by the buses. A second insight comes from looking at the total number of cards a system can contain. In this regard, VME comes out on top.

VME is an asynchronous bus, and the limiting factor on card count is signal integrity alone. The number of boards in the system is simply a function of the backplane and how well the signals can propagate through it. The data transfer rate used by the bus will determine the type of backplane, but not the total number of boards. That said, physical constraints of a rackmounted card cage limit the maximum number of slots in a VME system to 21.

Unlike VME, PCI is a synchronous bus. In order to assure proper signal timing characteristics, CompactPCI has strict limitations on the total load (mainly capacitance) a given board may drive. This translates to an upper limit on the number of boards allowed on one backplane driven by a single master bridge. It also means that the upper limit depends on the speed of the signals. The higher the bandwidth, the less boards can be used in the system.

Bus bandwidth considerations
Bus bandwidth is an important point in the competition between the two buses. Many applications require the highest transfer rate possible and here CompactPCI has a significant advantage, at least when looking at off-the-shelf available technology. CompactPCI using 66 MHz PCI and a 64-bit bus is available off the shelf with a data transfer rate of 528 Mbytes/sec.

This transfer rate is only achievable, however, when performing a write-only block transfer with data moving in one direction. To calculate a transfer rate more representative of actual system use, one needs to consider the overhead. For example, manipulating data in memory, involves two operations – read and write – and not always in large blocks. Thus, the average expected transfer rate is one half of the maximum possible.

Traditional VME is available off the shelf with a transfer rate of only 80 Mbytes/sec. The VSO has defined a high transfer rate protocol – 2eSST (2 edge Synchronous Source Transfer) – that theoretically supports unlimited transfer rates when moving data in one direction (write). The unlimited attribute stems from the fact that the VME is an asynchronous bus and thus, the data rate is limited only by the hardware capabilities. In practice, extremely high transfer rates have been achieved. At the 2000 Bus & Board Conference, General Micro Systems (GMS) and Arizona Digital demonstrated a transfer rate of 768 Mbytes/sec over a VME-320 backplane using the prototype of the GMS OmniVME PCI-to-VME Bridge.

Electrical and functional differences also exist, such as the number of cards a system can contain, data transfer rates, interrupt structure, and special system attributes such as multiprocessing, high-availability, and plug-and-play capability. Each area of difference needs to be evaluated in light of the final application.

Interrupts for real-time operation
Along with data bandwidth, a key performance element of real-time systems is their response to interrupts. Here, the VMEbus has some advantages. On CompactPCI there are only four physical Interrupt Request (IRQ) lines. This does not mean that only four interrupting devices may be connected in a given PCI system (there may be up to 32), but all 32 devices’ IRQs will be connected through the four physical lines.

A CompactPCI system assigns the IRQ numbers during a plug-and-play initialization process. The basic plug-and-play procedure is a three-step process that first checks what resources each device needs. Next, it coordinates the assignment of resources, including IRQs, to avoid conflicts. Finally, the process tells the system and software which choices it has made.

In theory, the plug-and-play process prevents interrupt conflicts from occurring. But the underlying PCI bus supports legacy devices that will always have the same IRQ number assigned to them. Having two such legacy devices in a system may cause conflict. Further, some interrupt numbers may be shared if a system has more than 32 interrupting devices, also raising the possibility for conflict.

VME doesn’t support a plug-and-play operation yet. However, the VME interrupt structure is simpler and richer than CompactPCI’s. There are seven IRQ lines on the VME bus supporting a virtually unlimited number of possible interrupting devices. Each of the interrupting devices has a vector number that the system designer pre-assigns. This vector number is unique to the interrupting device, eliminating the chance of having conflicts. The weakness of the system is that system configuration, resource assignment, and avoiding conflict is the system engineer’s responsibility. This means that VME developers must be thorough and accurate in their documentation, in case someone else should add to or change the system.

Multiprocessing on VME
Along with these bandwidth and interrupt performance differences, the two buses differ in their behaviors. One difference is in multiprocessing. To achieve multiprocessing, the computer boards in a system must be able to communicate with each other and with the other cards in the system. The backplane provides a convenient
communications path, but to use it each of
the computer boards should be able to
become the system master or bus master.

Both buses support this approach. However,
since VME was designed from the begin-
nning to fully support multiprocessing, this
feature is built-in and readily implemented
by each of the computer boards. In VME,
business arbitration is done by hardware. The
Bus Request (BR) line goes through the
backplane while the Bus Grant (BG) is
daisy-chained. When a computer board gets
the bus, it can immediately start working
with any other board on the bus and access
any address in the VME address space.
Because all the addressing is predefined, the
application software knows the addresses
for all the different boards and/or functions
that the computer board may need to access.
The software overhead is nil.

CompactPCI was not really designed for
multiprocessing—it is an added feature that
suffers some overhead. CompactPCI
enables each board to communicate with
each other, but because of the plug and play
mechanism, the addressing and interrupt
structures are dynamic. The only board that
knows the addressing and interrupt assign-
ment is the System Master (SM). Other
boards that may become bus masters are
Peripheral Masters (PMs). When a PM
needs to communicate with another PM, it
must not only arbitrate for control of the
bus, it must get the information about the
other PM from the SM before it can start
communicating. This added overhead
makes a hard real-time multiprocessing
scheme on CompactPCI impractical.

High-availability for telecom
While multiprocessing is awkward
on CompactPCI, the bus handles high-
availability computing readily. More and
more mission critical applications require
6-sigma (99.9999 percent) system avail-
ability. In order to get such high-avail-
ability performance, the system needs to
have mechanisms for:

- Detecting malfunction
- Communicating the malfunction to the
  right system components
- Allowing hot-swap of system cards
- Offering redundant functions

Telecommunications has adopted and
driven high availability because such avail-
ability is a major requirement for telecom
systems. Because telecommunication pro-
viders also adopted CompactPCI as their
standard system bus, all of the features
needed to implement high availability are
available off the shelf for CompactPCI.

VME, on the other hand, has never had
any real need for this mechanism to be
implemented. It is available. In fact, VME
was the pioneer in defining the hot-swap
mechanism for backplane cards. Lack of
interest, however, left VME with a stan-
ard but no off-the-shelf implementation.

This situation is repeated in other ways in
the CompactPCI and VME buses. Many
potential users adopted the CompactPCI bus
to cash in on its anticipated cost reductions
(stemming from the use of personal com-
puter components). The VME community
responded to the potential market loss by
trying to match CompactPCI features, defin-
ing standards that were never implemented.
Fortunately, the bus wars seem to be over
and each community is settling down and
focusing on what their bus does best.

That leaves system
designers with a
much simpler choice.
With the two com-
munities no longer
trying to be all things
to all users, develop-
ers can evaluate the
buses on what they do off-the-shelf, not
what they might do with the right support.

Udi Levin joined General Micro
Systems (GMS) in March of 1998 as Chief
Technical Officer and Product Manager.
Udi received his BSEE degree from the
Technion Institute of Technology in Israel
and is close to completing his MCS in
Electrical Engineering.

For more information, Udi can be
contacted at:
Tel: 1-800-307-4863
E-mail: ulevin@gms4sbc.com

Go to www.vmebus-systems.com/
etseminarppt.mov for GMS’s
Net Seminar “VME vs CompactPCI”