

Extending and Enhancing DSP Designs

Through the Use of CompactPCI

*How to Leverage Standards Based PCI Communications and Software
Without Compromising Internal DSP Data Flow*

By

Tim Marchant

Spectrum Signal Processing Inc.

Requirements for Signal Processing vs. Data Processing	2
Real-time Responsiveness	3
Deterministic Behavior.....	3
Adaptive to External Requirements	3
Lean, Fast & Scaleable	3
“Black Box” Subsystems vs. Host Processing.....	4
Proprietary Internal Buses	4
Shared Memory Architectures	4
New Opportunities with CompactPCI	4
What is CompactPCI?	4
Rugged Capabilities & Flexible Configurations.....	5
Processor Independence	5
Extending Standards-based Environments to Embedded Subsystems.....	5
Architectural Issues for CompactPCI-based DSP Systems.....	6
Unchanged Architectural Requirements	6
The Need to Isolate DSP Systems from Host Data Traffic	6
The Need for Embedded Real-Time Operating Systems	6
New Architectural Opportunities	7
Implementing Local DSP Buses with PCI.....	7
Eliminating “Double-copying” by Using Distributed Shared-Memory.....	7
Extending DSP Systems to Provide Global Addressing Capabilities	8
The Need for Efficient, High-Density DSP/PCI Bridge Chips.....	8
Small Form Factor.....	8
Highly Efficient Memory Interfaces.....	8
Full PCI Bus Utilization.	8
Augment Global Addressing.....	8
Putting It All Together	9
“Snugly Coupled” DSP Processors with Improved Data Movement	9
The Leverage of a Flexible & Extensible Architecture.....	10

Requirements for Signal Processing vs. Data Processing

Digital Signal Processing (DSP) applications are a ubiquitous fact of life that have made the digital computer revolution possible. In general, digital data processing is able to maximize its internal performance by isolating itself from the impacts of the outside world's "messy" analog nature. On the other hand, many computing systems are specifically designed to help us analyze, monitor, control or otherwise deal with these real world phenomena. This is where DSP comes into the picture. By living "at the edge" where the analog world becomes digitally represented, DSP systems provide data processors with their vital, real-time linkages to "natural data types".

Because data processing and signal processing have very different requirements and objectives, the architectures used to implement them have developed along inherently different paths. In order to provide an interface to constantly changing analog phenomena, a DSP system must be much more time-oriented than a data processing system. In signal processing, concurrent events and dynamic changes must be handled gracefully in a real-time mode, where time is of the essence. The data path and processor utilization must be optimized for continuous flow, while avoiding unnecessary queue time or processor interrupts. On the other hand, data processing systems typically are designed to optimize flexible access by users and applications, while timely response is treated as a variable rather than a constant.

DSP Lives "At The Edge" of our analog world



In order to deal with massive signal input streams, such as seismic analysis, medical imaging, telephone switching, etc., DSP processing has evolved over the past two decades into highly specialized systems that are uniquely designed to handle real-world phenomena in real-time situations. Some of the critical characteristics that distinguish these specialized DSP architectures from host-based data processing include the following:

Real-time Responsiveness

Signal Processors must be designed for lean performance oriented responsiveness to specific external events. Unlike a data processing systems, where the user can simply re-enter missed data, the DSP system has to continuously move data within the hard time-deadlines of real-world events with constantly changing signal characteristics.

Deterministic Behavior

DSP systems are not designed to flexibly perform many different tasks, but rather to perform a specific task with consistently predictable results. In order to convert natural data into consistently reliable digital data, the DSP's internal data flow and processing has to handle every event in a deterministically consistent fashion. Put another way, actions within the DSP system have to happen the same way every time to get consistent results.

Adaptive to External Requirements

In order to handle real world data, DSP systems have to be tailored to external events, rather than the other way around. With a data processing system, the user or application is typically required to do most of the adapting, for instance entering data in a structured fashion that is optimized for the system's requirements. In a signal processing system, the design has to be adapted to meet real-world events on their own terms by conforming to the full natural range of external data frequencies, values, variations, etc.

Lean, Fast & Scaleable

Today's sophisticated digital signal processing systems typically incorporate very high-performance, high-density architectures, consisting of multiple DSPs, tightly linked for both data flow and inter-processor synchronization. Extraneous functions are held to a minimum while core tasks are highly optimized. Storage space (RAM, ROM, etc.) is often very limited and must be carefully allocated only where required. Software operating environments invariably consist of lean, high-performance Real Time OS (RTOS) kernels designed to deliver maximum responsiveness within a tightly focused environment. On the other hand, data processing systems offer full-featured, ease-of-use oriented environments, designed to deliver user-friendly, error-forgiving performance in the much slower timeframes common to human operation. By leveraging today's large RAM and disk storage capacities, data processing systems can support huge, multi-faceted operating systems, which would be completely untenable in a focused DSP environment.

“Black Box” Subsystems vs. Host Processing

Because DSP functions are so inherently different from the data processing activities of host processors, DSPs are often treated as completely separate “black box” subsystems. Host-centered systems typically require a dedicated backplane that is mastered by the host processor at all times, thereby creating a built-in incompatibility with the DSP subsystem’s own requirements for unrestricted bandwidth. Traditionally this has been resolved by completely de-coupling the DSP subsystem’s internal processing and communications activities from the host-driven traffic on the system bus.

Proprietary Internal Buses

As the DSP industry has migrated toward a common practice of de-coupling the signal processing subsystem from the host bus, a wide range of options have become possible for optimizing internal performance. For instance, today’s multi-processor DSP systems make use of high-speed internal buses to maximize data flow through the system. Usually, these on-board private bus structures consist of direct extensions of the DSP processor buses. For instance, Texas Instruments’ TMS320C6X family and the Analog Devices 2106X family both offer the ability to extend their internal proprietary address and data buses to encompass shared memory with other processors.

Shared Memory Architectures

Such shared memory structures have now become a primary mechanism to optimize internal data flow on today’s DSP boards by acting as a “data dump” for inter-processor communication. However, as will be detailed later, the use of shared memory architectures often requires each processor to perform multiple copy and read actions because they cannot directly write and read each other’s local memory. This overhead requirement for “double copying” is one of the key DSP design challenges, which potentially could be mitigated through use of a more standardized on-board address and data bus, such as PCI.

New Opportunities with CompactPCI

Growing industry acceptance of the CompactPCI bus is now opening new opportunities for more closely integrating the DSP subsystem with PCI-based host systems and also for bringing standards-based flexibility to internal DSP system bus structures.

What is CompactPCI?

CompactPCI is a very high performance industrial bus based on the standard PCI electrical specification but packaged in a rugged VME-like 3U or 6U Eurocard format. The CompactPCI standard is controlled by a broadly-based industry consortium called the PCI Industrial Computer s Manufacturer’s Group (PICMG). PICMG currently has over 360 member companies and is open to all organizations and individuals interested in furthering the use of the PCI standard in the industrial marketplace.

Rugged Capabilities & Flexible Configurations

Instead of the commercially-oriented card edge connectors used in desktop PCI systems, CompactPCI makes use of high-quality pin and socket connectors. These 2mm metric connectors conform to IEC and Bellcore standards and provide a much higher level of shock and vibration resistance than card edge connectors. The CompactPCI 3U form factor supports two connectors, while the 6U form factor adds three additional connectors, which can be used to implement secondary buses (such as SCSI or MVIP telephony buses) on the same backplane or for bridging to other buses (such as VME). PICMG has also developed recommended pinouts for supporting IndustryPack and PMC on the upper three 6U connectors. To provide maximum configuration flexibility, CompactPCI boards are inserted from the front of the Eurocard chassis with options for I/O connections to either the front and/or rear of the card. Extensibility for “hot swap” operation has also been built into CompactPCI through the specification of staged, multi-length pins to allow power connection prior to signal connections.

Each CompactPCI bus is limited to a maximum of eight card slots for electrical loading reasons, however the bus can easily be expanded with readily available PCI-PCI bridge chips. All interrupts, plug-and-play information, and data are automatically transferred across each bridge, with a minimal penalty of about 1 clock per transaction. Large burst mode transactions can therefore move from bus to bus with a relatively small bridging overhead.

Processor Independence

CompactPCI also provides an open opportunity for building systems based around a wide range of host processors. Even though PCI’s primary recognition has come as the local bus for Pentium class PCs, PCI also lies at the core of all modern microprocessor designs. For instance, DEC’s Alpha and the PowerPC processors both support PCI interfaces and can easily be implemented on the CompactPCI bus.

Extending Standards-based Environments to Embedded Subsystems

One of the greatest potential benefits of the migration to CompactPCI is the ability to leverage some of the benefits of standard mainstream PCI-based computing systems from within embedded subsystems. In addition to bringing the whole range of Windows-based software to PC hosted systems along with flexible, extensible backplanes, CompactPCI also brings the concepts of plug-and-play configuration and hardware abstraction layers into the realm of embedded systems for the first time. CompactPCI’s role as a true “systems level” bus sets the stage for a much greater degree of software portability and standards-based integration to specialized subsystems, such as signal processing devices.

Architectural Issues for CompactPCI-based DSP Systems

Given the growing usage of CompactPCI and the significant opportunity for increased high-level system integration between host and subsystem processing, what are the architectural possibilities for implementing PCI within high-performance DSP systems?

Unchanged Architectural Requirements

Basically, while it opens new doors for bus standardization and improved inter-processor communications, we must keep in mind that PCI is by no means a panacea that portends a complete merging of DSP activities into the host system. Some critical aspects of existing DSP architectures will need to remain unchanged in these new systems, while other aspects can potentially be radically improved. Regardless of the bus structures being used, the key architectural premise of separating, streamlining and optimizing the DSP subsystem will continue to be critical for achieving adequate performance.

The Need to Isolate DSP Systems from Host Data Traffic

As previously described, the DSP system has to retain its independent capacity for real-time response to real-world data. Even if PCI and/or CompactPCI buses are implemented for both the host and the DSP subsystem, they must run completely independent of one another. The host processor still has to respond to external interrupts from the Human-Computer Interface (HCI) and balance multiple data processing issues, without adversely impacting the DSP subsystem. Today's 33 MHz 32-bit PCI bus can handle up to 132 Mbytes/s sustained transfer rates, which can be adequate for internal DSP subsystems, providing there is no bandwidth loss to host processor overhead activities. Even though emerging 64-bit PCI buses, running at 33 MHz, will be able to support sustained transfer bandwidths up to 264 Mbytes/s, a real-time oriented DSP system will still need to be sheltered from the intermittently available nature of the host-mastered bus.

The Need for Embedded Real-Time Operating Systems

In addition to needing local control over its own bus, a DSP system invariably must incorporate a lean, performance-optimized run-time operating system. Given the continuing growth, some would say "bloat", of most mainstream operating environments, this means that DSP systems must continue to rely on small Real-Time Operating System (RTOS) kernels that are tailored to embedded system requirements. Because RTOS kernels must be small enough to run on individual DSPs, with 2 MB or less of on-board memory, DSP board designers are constantly making tradeoffs between desired RTOS features and the overall size limitations of the DSP processors' RAM and/or ROM.

In addition to running within limited space on the individual DSPs, optimal system performance requires that the RTOS support both distributed operations and parallel programming of DSP tasks. To ensure extensibility, the RTOS kernels must also be able to support seamless addressing and data across the PCI bus.

New Architectural Opportunities

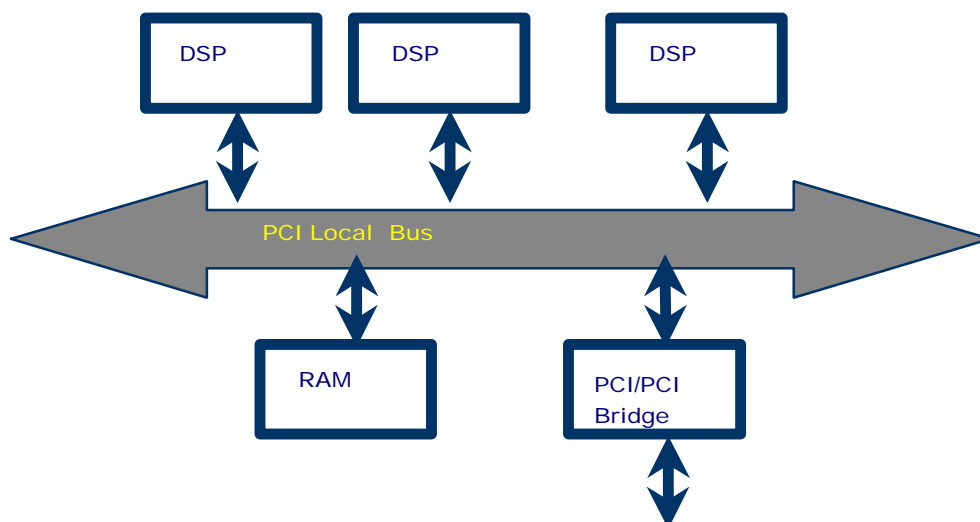
Implementing Local DSP Buses with PCI

The raw speeds achievable by dedicating the PCI bus to the DSP subsystem can be fully sufficient to meet the data flow needs of many signal processing requirements. Today's widely used 33 MHz 32-bit PCI bus offers a nominal sustained bandwidth of 132 Mbytes/s and the rapidly emerging 33 MHz 64-bit version doubles that to 264 Mbytes/s. The major issue that must be resolved involves building a highly efficient interface between each DSP and the PCI bus to enable full utilization of the bus bandwidth while providing the processor densities required for large scale signal processing.

Eliminating "Double-copying" by Using Distributed Shared-Memory

As previously described, traditional multi-processor DSP designs have tended to use shared memory as a convenient "data dump" for inter-processor communication. Because the processors cannot write/read each other's local memory the data flow is inherently a multi-step process. For data to move between processors, it must be written into the shared memory by the first processor and then read by the receiving processor. Essentially every data movement between processors requires significant time to be expended in "double-copying". Further exacerbating the problem in multi-board systems, is the need to move data between boards, which typically requires an additional step, resulting in "triple-copying". With steady increases in both the use of multi-processors and the raw speeds of individual DSPs, these inherent limitations of shared memory write/read overheads are now becoming the limiting factor in the overall performance of many DSP systems.

On the other hand, the use of a PCI local bus, with its standards-based 32-bit addressing capabilities, opens up the internal DSP architecture to allow multiple processors to write and read to the same memory areas. By enabling all of the DSP processors to write/read each other's local memory, they can directly move data in a "single-copying" process. Coupling this concept of "distributed shared memory" with the use of PCI as the local DSP bus can effectively double the data flow bandwidth between processors, thereby making full use of every processor's raw speed capabilities and greatly improving overall system performance.



Extending DSP Systems to Provide Global Addressing Capabilities

In addition to significantly improving on-board data flow, the use of PCI to PCI bridge chips can further optimize data flow between on-board and off-board DSP processors in a multi-board system. However, today's typical DSP addressing ranges of 16 to 24 bits will need to be extended in order to take full advantage of the PCI bus' 32-bit address range. From a systems standpoint, full PCI bus addressability can open the door to a whole new level of DSP system scalability by giving DSPs direct access to data and resources throughout the system.

The Need for Efficient, High-Density DSP/PCI Bridge Chips

Perhaps the most critical factor in bringing the inherent speed and flexibility benefits of PCI to the DSP world is the need for an efficient and real-estate conserving method for bridging each DSP to the PCI bus. Because of the high processor densities used in today's DSP systems, the design cannot afford to give excess board space to PCI bridge circuitry. At the same time, the throughput demands of DSP systems require that the bridge functions be highly optimized. Therefore the key to overall system performance lies in the ability to inexpensively and efficiently "PCI enable" each DSP.

Critical requirements for the DSP/PCI bridge chip must include:

Small Form Factor.

The DSP/PCI bridge chip will need to be used many times on an already tightly constrained PCB. Board real-estate limitations will necessitate that the bridge devices be available in Thin BGA type packages, which can be mounted on either side of the PCB for maximum flexibility.

Highly Efficient Memory Interfaces.

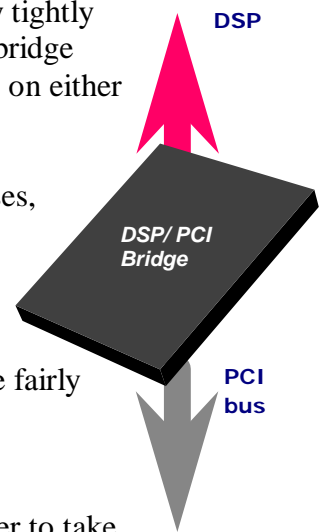
The bridge chip must provide a very clean interface to DSP memory buses, with no-wait-state support for fast static RAM.

Full PCI Bus Utilization.

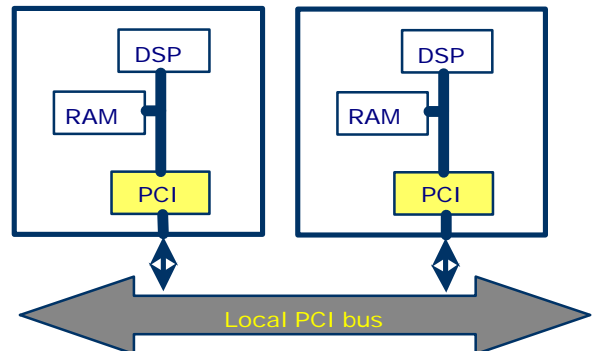
It must also deliver full speed access to the PCI bus' sustained transfer rates (132 Mbytes/s for 32-bit PCI and 264 Mbytes/s for 64-bit PCI), with no bottlenecks or data flow disruptions. This will likely necessitate fairly large FIFO buffers for adequate elasticity and to avoid stalls.

Augment Global Addressing.

The DSP/PCI bridge chip should also provide a built-in mechanism for augmenting the limited address ranges of today's DSP processors in order to take advantage of the PCI bus' full 32-bit addressing range.



By providing an efficient and uniformly replicable interface between the DSP, its local memory and full bandwidth access to the local PCI bus, the DSP/PCI bridge chips will provide the vital link to PCI enabling the entire DSP subsystem. In addition, it will open the door for giving each DSP direct addressing access to global resources throughout the host system.

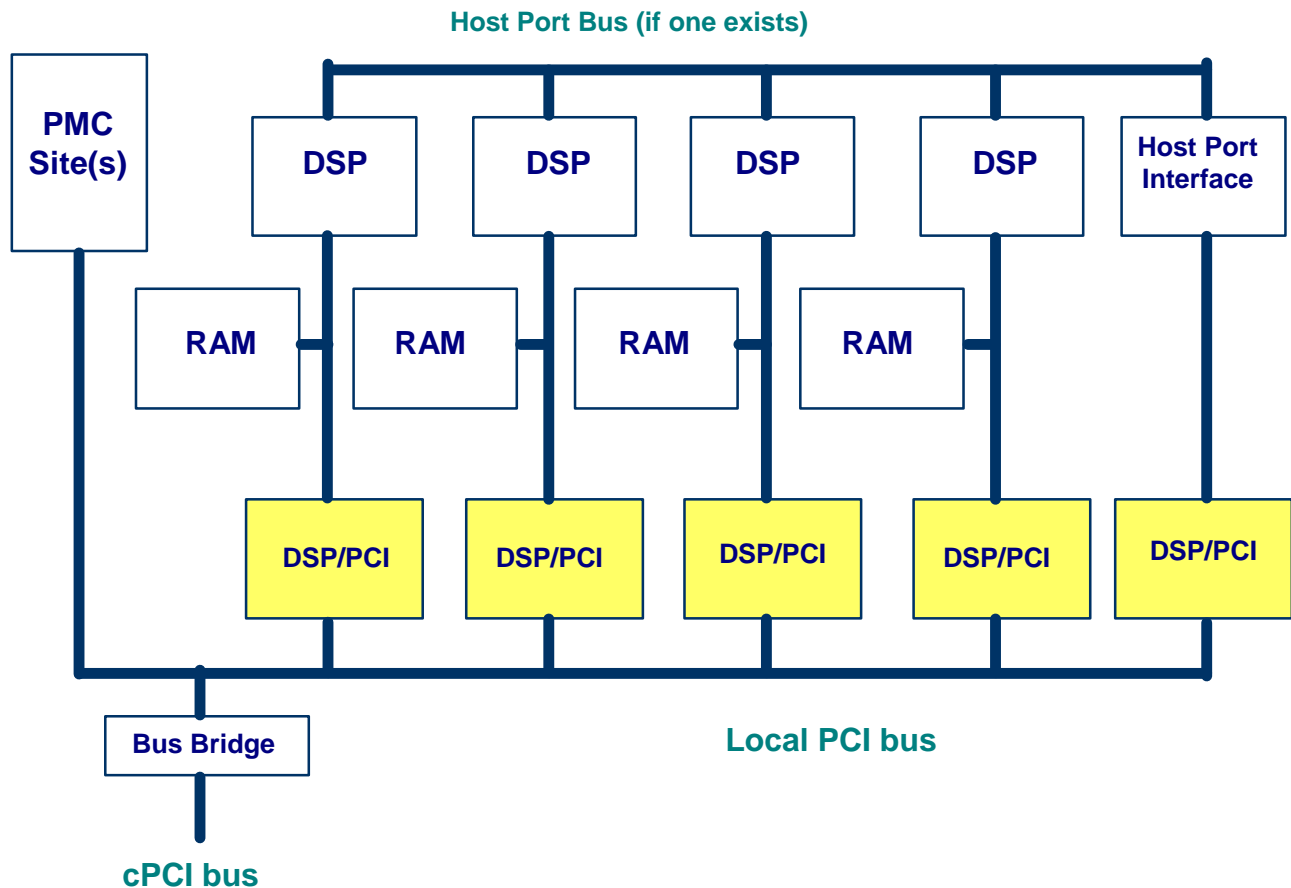


Putting It All Together

By PCI-enabling the DSP subsystem, the system designer can accomplish two major break-throughs at the same time. The first is essentially doubling the DSP system's internal data flow by substituting direct addressing in a Distributed Shared Memory structure to alleviate the double-copying bottlenecks of previous shared memory designs. Secondly, the provision of direct DSP access to an internal PCI bus breaks through the negative aspects of DSP isolation and opens up possibilities for maximum leverage of external system resources and software flexibility.

“Snugly Coupled” DSP Processors with Improved Data Movement

Instituting Distributed Shared Memory essentially creates a “snugly coupled” relationship between the DSPs in a multi-processor system. They are neither loosely-coupled (e.g. relying only on message passing) nor are they tightly-coupled (e.g. constrained to an arbitrated shared memory scheme). By making a small investment in new hardware, the highly replicable DSP/PCI bridge chip, the data flow options have been dramatically enhanced while performance has been simultaneously improved. In addition, the configuration possibilities for efficiently bridging the DSP system to the CompactPCI bus and the host processor have also been expanded.



The Leverage of a Flexible & Extensible Architecture

Although the primary original objective was the improvement of DSP performance in high-density systems, perhaps the more important result of PCI-enabling the DSP system is the leverage that comes from PCI's flexible architecture. For a number of years, mainstream host-side Pentium designers have been enjoying the benefits of PCI's systems-oriented capabilities, in which any processor can directly address any PCI resource. Now, with PCI-enabled DSP, the same degree of programming flexibility, "flat architecture", and orthogonal inter-processor communications can be extended to designers of signal processing systems.

This flexibility greatly simplifies the task of integrating the DSP subsystems into the overall system design. In essence PCI is providing the mechanism for signal processing to break out of its "black box". PCI's uniform inter-processor communications scheme means that the DSP can interface directly with other resources throughout the system across the CompactPCI bus, without requiring host intervention or incurring host overheads. For increasingly large DSP challenges, this flexibility also opens the door for next-generation, highly scalable, multi-master, distributed DSP systems, which can independently handle massive signal processing tasks.

By clustering all of the DSP system's time-sensitive, real-world dominated activities around the same system-level bus that forms the heart of modern data processing systems, the stage is set for continuous processing and seamless movement of data, both within and between the system components. Although DSP systems will continue to "live at the edge" between our natural world of messy analog phenomena and the orderly, highly structured world of digital data processing, the PCI enabling of DSP will dramatically help "shorten the distance" between those worlds.