

PRODUCT GUIDE:

Processors

A high-performance SDR implementation for CompactPCI

By Rob Kraft

SDR refers to Software Defined Radio, the technology of implementing the processing required for radio reception and transmission in software-reconfigurable platforms based on devices such as DSPs, FPGAs, and general purpose processors. Compared to traditional implementations of radios completely in analog or using a combination of analog and fixed function digital devices, the SDR approach has advantages in flexibility and cost savings. However, the implementation of high-performance SDR in communications base stations and complex applications requires systems with multiple processing elements, high internal data flow, and system-level high availability. Hot-swap CompactPCI is an ideal foundation upon which to build such systems. The augmentation of hot-swap CompactPCI with open-standards-based switched fabric interconnects over the backplane is ideal for an SDR implementation.

INSIDE THIS SECTION

A high-performance SDR implementation for CompactPCI

Low power embedded Intel architecture - an excellent fit for CompactPCI systems

PICMG 2.16 compliant dual processor single slot CompactPCI SBC

Processor table

This article briefly discusses SDR and then describes details of Spectrum's high-performance CompactPCI SDR platform - the SDR-3000. The SDR-3000 distinguishes itself from other SDR platforms on the market because it was specifically architected to meet the synchronization, processing, and data flow topology and bandwidth requirements of demanding SDR applications. The system is made up of three component CompactPCI cards providing a balanced combination of I/O and reconfigurable processing. The hot-swappable cards are integrated using the cPSB standard and other passive-backplane switched fabric interconnects. Software design for the system is simplified by the use of abstracting software drivers, and CORBA; firmware design, by the support of FPGA cores along with a unique hardware abstracting "wrapper." This synthesis of hardware and software technologies has produced a reconfigurable SDR platform capable of extremely high-channel densities, high bandwidths, and previously unrealizable implementations. It provides OEMs the ability to reduce cost-per-channel and communication engineers with the means to create applications that in the past they could only dream about.

Software defined radio

The idea behind Software Defined Radio (SDR) is to perform radio signal processing (receiving only, or receiving and transmitting) in a combination of software and firmware, rather than employing fixed-function digital or analog components for that purpose. The processing of higher-level communication protocols in software is not new, but processing of the air-interface layer in reconfigurable devices is, because programmable device technology has only recently reached the required performance level. Examples of the types of processing functions are:

- Up and down conversion of signals between baseband and an intermediate frequency (IF)

- Various levels of synchronization
- Multiple access processing
- Modulation and demodulation of the baseband signals
- Forward error correction (e.g., Viterbi, Turbo code/decoders) of the baseband signals
- Spreading and despreading

But what motivates embracing SDR instead of using fixed-function ASICs? After all, SDR does not offer a pure performance benefit, and per air interface, an SDR implementation is typically more expensive. So, if you are only going to manufacture a single type of radio forever, are never planning to upgrade or expand it, and are not expecting any changes in the air-interface standard or signals it is using, then SDR is not the ticket.

However, to an OEM that is producing products with a mandate of scalability or upgradeability, where more than one air-interface standard or an evolving standard is being supported, or where there are several different product lines, each addressing a different standard or application, SDR offers:

- Flexibility to improve or migrate and evolve algorithms and protocols by uploading new software onto the same hardware
- On-the-fly adaptability to different air-interface standards in multi-standard environments
- Economy of scale and reduction in inventory diversity by using a common software-reconfigurable hardware platform across various product lines
- Reduction in development effort for future products (need to develop software only, rather than both hardware and software)

In other words, SDRs offers an economic benefit because of its flexibility. The flexibility allows you to reuse hardware and amortize SDRs cost over multiple air interfaces, revisions of a standard, or products and results in the economic benefit.

Of course, an inherent assumption in SDR is that the underlying reconfigurable hardware will in fact enable the software to meet all of the performance requirements

of the SDR applications. This is by no means a shoe-in and requires that careful consideration be given to the hardware architecture. Some of the key requirements of a hardware platform that will be used for physical layer processing are:

- A/D and D/A converters with sufficient speed and resolution to cover the band of interest with sufficient dynamic range
- FPGAs large and fast enough to handle the lowest-latency, most demanding processing functions that occur on the high-speed data coming from/going to the A/Ds and D/As (e.g., functions such as down and upconversion, de/spreading)
- Processors powerful enough to handle the remaining baseband modulation/demodulation functions
- Inter- and intraboard communication paths of sufficient bandwidth to handle the data flows to and from the various devices
- Communication paths that have the flexibility to permit the required data flow topologies required in current and future applications (e.g., many-to-many topology required by beam forming/steering and direction/location applications)

In addition to these hardware requirements imposed by SDR in general, the implementation of SDR in infrastructure gateways and complex applications imposes some additional requirements like:

- The capacity to handle numerous channels of data, thereby requiring multiple processing elements and correspondingly higher bandwidth communication paths
- Scalability of the platform to enable expansion of the system as user load increases and for use in both low and high performance product lines
- High availability at the system level (particularly for infrastructure equipment), implying an absence of single points of failure

Hot-swap CompactPCI along with switched fabric interconnects over the backplane provide a base upon which to build an SDR platform that accommodates all of these requirements. Spectrum Signal Processing's SDR-3000 is an example of just such a platform.

The SDR platform

The SDR-3000 consists, at a minimum, of two hot-swappable CompactPCI blades and a backplane transition module. The

transition module, known as the TM1-3300 holds the A/Ds and D/As and connects directly through the backplane J5 connector to the PRO-3100, the blade containing FPGAs. The PRO-3100 is connected to the PRO-3500, a blade holding the processors, via flexFabric, a switched-fabric interconnect running over the backplane J3 and J4 connectors. flexFabric is a passive backplane interconnect using Serial RapidIO protocol and associated LVDS signaling. The blades can also communicate over the PCI bus running on J1 and J2, or over the redundant PICMG 2.16 compliant CompactPCI Packet Switching Backplane (cPSB) implemented as 100 Mbit Ethernet over the backplane J3 connector. The PCI bus would typically be used for control plane functions, with the flexFabric acting as the high-speed data plane. The cPSB would typically be used for transmitting payload data between the SDR-3000 blades running physical layer algorithms and other

blades running higher protocol layers, user interfaces, etc.

The overall system diagram is shown in Figure 1. A typical SDR system partitioning would use the PRO-3100's FPGAs as the channelizer, implementing wide or narrowband downconverter and upconverter cores to break out the individual frequency channels in the IF connection with the A/Ds and D/As. Depending on the complexity of the application, some or all of the modulation/demodulation would also be implemented in these FPGAs. The PRO-3500's processors would be used for the baseband processing. In the following sections we will discuss the individual boards' hardware and software in greater detail and then look at some of the advantages of the entire platform.

The transport module

Figure 2 is an illustration of the TM1-3300 transition module that holds four

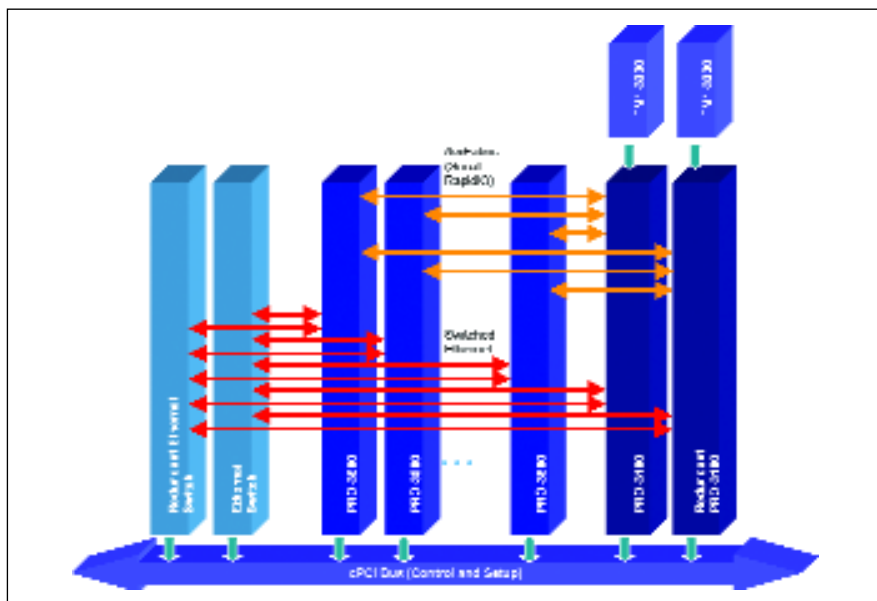


Figure 1. Typical SDR-3000 platform showing interboard communication linkage

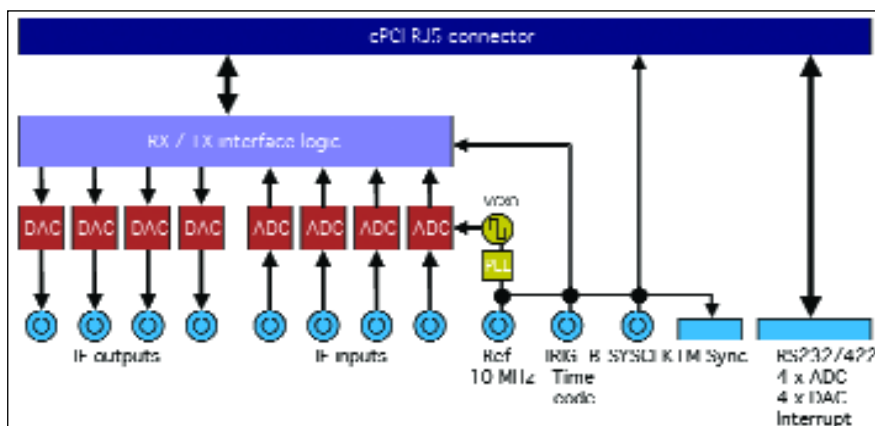


Figure 2. TM1-3300 Transition Module containing A/D and D/A converters

AD6645 14-bit A/Ds and four AD9772 14-bit D/As from Analog Devices. The A/D and D/A are capable of being clocked at up to 80 Msamples/sec, although the D/A has an internal 2x interpolation filter that up-samples the data to a rate of up to 160 Msamples/sec. The converters can be clocked internally or using an external reference. In addition, for location (direction finding) and beam forming applications, facility is provided both to timestamp the data using an external IRIG-B reference and to synchronize multiple TM1-3300 modules. Data flows between the converters and the PRO-3100 through a 1.6-Gbyte/sec serial RapidIO-like interface through the backplane J5 connector, called the TM1 interface. The converter module is the only component of the SDR system made up primarily of fixed-function hardware; this is necessitated by the fact that it is the interface between the analog IF signals and the digital domain.

The channelizer board

The PRO-3100 channelizer board is illustrated in Figure 3. It contains four user-programmable Xilinx Virtex-II FPGAs (XC2V1000s or XC2V6000s; the latter having 6 million gates) each having up to 256 Mbytes of SDRAM. The IBM PPC-405GP (405) embedded controller and flash allows the board to be self-hosting; it can act as the PCI bus controller, removing any dependency on using a (single point-failure) SBC in the chassis as the controller. The 405 also handles control and setup functionality, and offloads the FPGAs from low-speed data movement and Ethernet control.

The focus of the board is the tremendous reconfigurable processing power provided by the FPGAs. The Virtex-IIs are each connected to two neighboring Virtex-IIs, to the *flexFabric* interface, and to the TM1 interface. The TM1 interface is capable of broadcasting data to all four Virtex-IIs, a data flow required by some beamforming and direction-finding applications. The 256 Mbyte SDRAM can be used for applications requiring a delay buffer for incoming receiver data (e.g., surveillance, test and measurement) or as a staging area for outgoing transmitter data. The PRO-3100 also has provisions for accepting synchronization signals from the TM1-3300 as well as for synchronization with other boards in the system.

Programming this behemoth is simplified through several software and firmware features. Board-level functions such as

booting and accessing resources in the FPGAs are supported using Spectrum's C-language API and library called *quicComm*. Higher-level programming is supported through the VxWorks operating system and the industry standard CORBA interface running on the 405. An important simplification for firmware designers is the existence of an application "wrapper" that shields the FPGA designer from having to understand the hardware details of the various interfaces to the FPGA. The wrapper provides a uniform look to the *flexFabric*, TM1, SDRAM, and inter-FPGA interfaces.

from the *flexFabric* can be sent to all four G4s in a single transaction, an important factor for beamforming applications. Mechanisms have also been employed to facilitate the coordination of time-critical events with the other boards in the system.

The PRO-3500's 405 Embedded Controller is used for set-up, control, and Ethernet. Some applications may take advantage of the 405 to run higher protocol layers. The G4's themselves could be used for this purpose in applications where their signal processing ability is in less demand. As with the PRO-3100, *quicComm*,

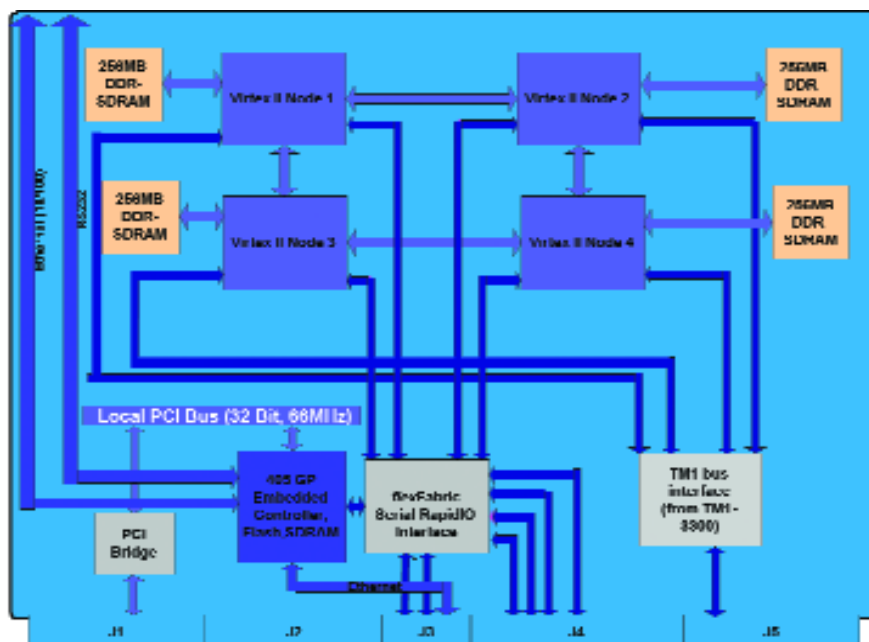


Figure 3. PRO-3100 FPGA board with Virtex II FPGAs

The baseband processing engine

The PRO-3500 baseband processing engine, shown in Figure 4, has up to four identical nodes, each containing a G4 PowerPC 7410 processor (with AltiVec) and memory. Two of the nodes are located on the board, while the other two can be added in the form of two enhanced PMC (ePMC) mezzanine modules called ePMC-PPCs. ePMC augments the PMC standard with four 200 Mbytes/sec full-duplex Solano~links (based on Spectrum's Solano Communications IC), providing high-speed low-latency links between the G4s and the *flexFabric*, as well as directly between the G4s; in both cases DMA engines are used to move the data to/from SDRAM, reducing the processor's burden to that of initiating the DMA and responding to an interrupt upon completion. Note that Solano ICs have a broadcast mode, so that packets arriving

VxWorks, and CORBA are all supported, with the addition that this applies to all of the G4s, as well as the 405.

The fabric

The *flexFabric* serial RapidIO backplane fabric uses point-to-point connections. The fabric uses the 4x serial link protocol, where four 1.25 Gbit/sec links (each link is full-duplex, giving 1.25 Gbit/sec transmit and receive concurrently) are ganged together to give a payload data rate 400 Mbytes/sec full-duplex (after accounting for 8B/10B encoding and other packet overhead). Since the protocol, including error detection and recovery schemes) is implemented in hardware, the connections are low latency and present no software overhead to the destination processing elements. Unlike some existing interconnects that use active backplanes, the *flexFabric* backplane is passive. This

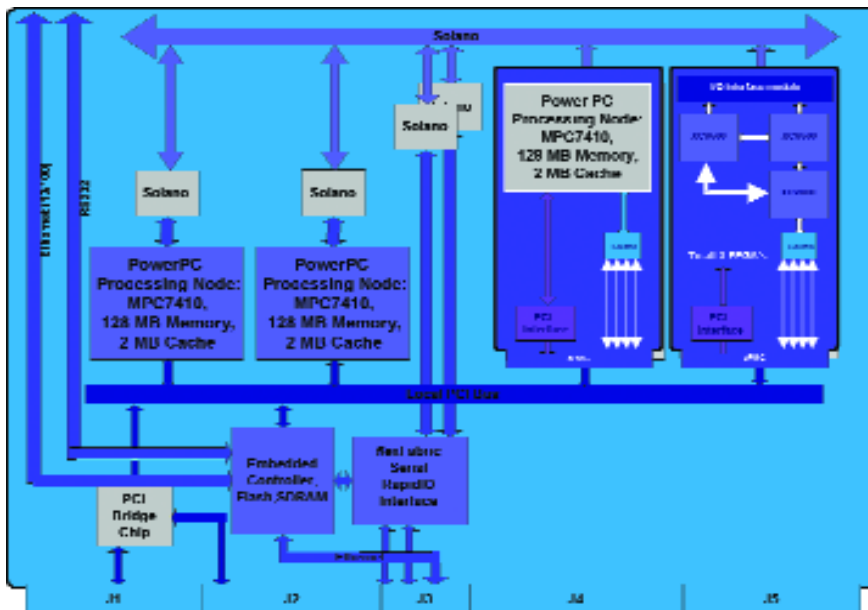


Figure 4. PRO-3500 processor board with PowerPC G4 7410 processors

enables the support of hot swap and avoids making the backplane a source of single-point failure. Since the connections are all point to point, a diversity of connectivity topologies are realized by sending packets through switch cards called fabric cards in the system (simple daisy chain topologies without *fabric cards* are also supported). The PRO-3100, with its six flexFabric links serves as the fabric card. Each PRO-3500 has two links, allowing for fully redundant systems, where the PRO-3500's two links each connect to a different PRO-3100 fabric card. Altogether, *flexFabric* provides the flexibility, performance, and high-reliability required for the variety of communications topologies in various multiprocessor infrastructure SDR applications.

Advantages

The SDR-3000 platform meets all of the requirements that a reconfigurable platform must have in order for an SDR solution to realize its full potential benefits. The TM1-3300's leading edge A/Ds and D/As have the speed and resolution required to interface to the analog IF signals in use today and in upcoming generations. The PRO-3100's reconfigurable VirtexII FPGAs that interface to the converters have the capacity to run multiple channels of the most demanding up- and downconverter processing. The PRO-3500's AltiVec-equipped give it the base-band processing horsepower required to execute multiple instances of complex demodulation algorithms on the data arriving from the PRO-3100. The commu-

nications paths delivered by the *flexFabric* provide both the bandwidth and flexibility of data flow topology to permit the same platform to fit in a diverse variety of radio applications and to be scaled according to their needs. Finally, the boards' hot-swap capability, embedded controllers, and the redundancy built into the fabric eliminate single points of failure and enable the construction of high-availability systems required for infrastructure and mission-critical applications.

The key advantage of the SDR-3000 architecture, and the area where other off-the-shelf systems typically break down, is the juxtaposition, using open standards, of a flexible and high-speed communications interface with the sheer processing power of the FPGAs and G4 PowerPCs. The processing power is a necessary precursor to implementing complex algorithms or processing multiple channels in less space. This is crucial in providing a lower cost-per-channel than was previously available in off-the-shelf systems. But it is only when that power is combined with the high-speed switched-fabric that the flexibility and scalability of the system is realized. It is the fabric's packet-switched nature that enables the same hardware to be applied in different connection topologies for channelized and nonchannelized (e.g., beamforming's many-to-many data flow) applications, and with different ratios of FPGAs to processors. It is the fabric's high-bandwidth, determinism, and low latency that truly enables the data to get where it's going in time to permit

partitioning of the algorithms across multiple boards. And it's the use of abstracting software and firmware, and open-standards like hot swap CompactPCI, serial RapidIO, and CORBA that make sure that going forward a variety of off-the-shelf solutions can be integrated together to form solutions.

Conclusions

SDR-3000 is a platform designed to meet the requirements of infrastructure and complex software defined radio systems. The advent of the SDR-3000 platform brings a change in paradigm for the architecture of multi-channel SDR applications. Rather than the traditional consideration of the application in terms of channel cards and densities of channels per slot the paradigm now shifts to channels per chassis. This shift occurs because of the capability to partition the processing of channels across several cards. And this capability is in turn realized by a platform based on diverse high-performance processing elements linked with a high-availability, high-performance switched-fabric interconnect. The SDR-3000 platform provides the canvas upon which designers and manufacturers of wireless infrastructure, test, and surveillance equipment can creatively paint their next-generation prototypes and production SDR systems.



Rob Kraft received his Bachelor of Applied Science in the Engineering Science program (electrical option) and Master of Applied Science

in Systems Control at the University of Toronto. Rob began his career as a Systems Engineer before joining Spectrum Signal Processing's Wireless Field Applications Engineering department. He spends his days assisting customers map their applications to Spectrum's systems and providing feedback for product development.

For more information, contact Rob at:

Rob Kraft
Spectrum Signal Processing
 200-2700 Production Way
 Burnaby, B.C.
 V5A 4X1 Canada
 Tel: 604-421-5422
 E-mail: rob_kraft@spectrumsignal.com
 Web site: www.spectrumsignal.com