

# DATA COMMUNICATION TECHNOLOGIES & ARCHITECTURES FOR DISTRIBUTED SDR TRANSCEIVER SYSTEMS

Frank Van Hooft (Spectrum Signal Processing, Burnaby, BC, Canada;  
Frank\_Van\_Hooft@spectrumsignal.com)

## 1. INTRODUCTION

The increasing visibility of SDR as a viable communications technology is driving equipment consumers to demand ever-greater performance. Today even the highest bandwidth & datarate waveforms are candidates for SDR implementations. Coupled with a desire for the maximum possible number of simultaneous channels, the high-end SDR implementations can absorb all of the processing power that can physically be applied to them.

In concert with high processing power comes a high data throughput requirement. Wide RF bandwidths and multi-channel implementations generate massive amounts of data that must be routed in real-time between various elements of the SDR system. Without reliable data paths the SDR system could not function. Yet the datarates required, which can easily extend into the many hundreds of megabytes per second, are far beyond that which traditional busses, such as PCI and VME, can readily accommodate. New data-movement technologies are essential to make real the promise of these high-capacity SDR systems.

This paper describes the different types of data that must be moved within an SDR system, including their varying requirements for parameters such as latency and datarate. A comparative examination of numerous candidate bus and fabric architectures is performed, highlighting the strengths and weaknesses of each. From this analysis some fabrics are selected for use. To conclude an example architecture is presented, showing how these fabrics can be applied to a Compact-PCI based SDR system to essentially off-load the default legacy PCI bus. The end result demonstrates the ability to create a high-performance SDR infrastructure within a standards-based environment..

## 2. DISTRIBUTED SDR TRANSCEIVER OVERVIEW

To begin, it is necessary to review an overview of a distributed transceiver. As a generalized statement this transceiver will have some number of RF inputs, some number of RF outputs, some number of channelizers, and

some number of modem / codec / baseband processing instances. Assume greater than one for all of these. In addition there is a control plane to manage the system. This architecture is illustrated in Figure 1.

On the receive side, this subsystem receives either digitized IF or baseband signals, extracts multiple user channels from these signals in the channelizer, then forwards these channels to channel processing for demodulation and decoding. This process is reversed on the transmit side, with payload data being encoded and modulated in the channel processor and then inserted into the output signal by the channelizer for transmission. In a distributed transceiver architecture, the channelization and channel processing functions are distributed across multiple signal processing elements, with a single channelizer often supporting multiple channel processors. From a practical perspective the channelizer and channel processors may be located on physically separate cards; there is no implicit assumption made about the physical configuration of any of the elements shown above.

## 3. DISTRIBUTED SDR TRANSCEIVER DATAFLOW REQUIREMENTS

Using Figure 1 it is possible to identify five distinct dataflows, each with their own set of requirements. There is significant commonality between some of them, which will be taken advantage of later.

1. A high-speed input/output channel between the processing elements in the channelizer and the digital interface to the RF transceivers. Interfacing with the high-speed A/D and D/A converters this communications channel typically represents the highest overall bandwidth in the system, and must guarantee low latency and deterministic performance. Often each digitized RF channel must be distributed to multiple processing elements in the channelizer, especially for applications supporting adaptive beam forming, and so the communications infrastructure must support broadcast capabilities.

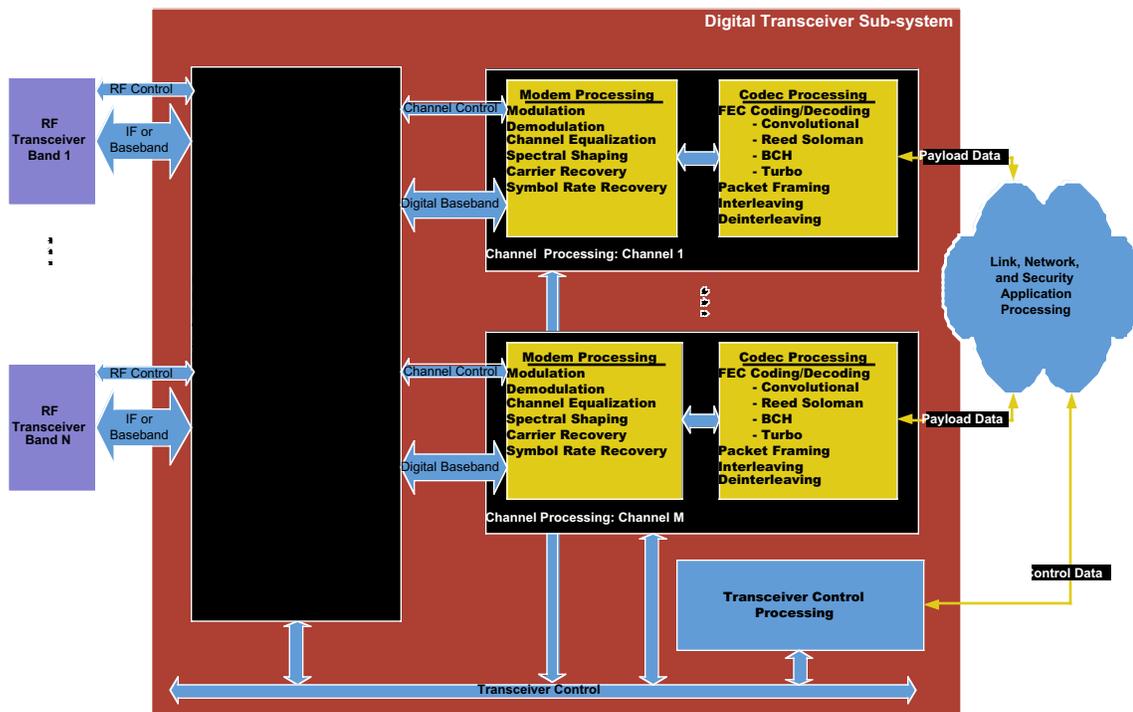


Figure 1 - Distributed Transceiver Overview

2. An inter-processor communications structure must exist to support the dataflow requirements between the processing elements of the distributed transceiver architecture. Since a single processing element will often host multiple simultaneous digital radio functions, each with independent data and control channels, the inter-processor communications structure must provide support for multiple logical channels over the physical communications path.
3. In conjunction with this, a third communications requirement can be identified as an inter-board communications channel. Ideally this would be the same as the inter-processor structure described above, so that the physical separation of processors within a board and across multiple boards is completely hidden from the application. But in many systems inter-board communications is a very distinct, or separate, communications path and so is identified separately here. This communications channel shares the same requirements as the inter-processor requirements listed earlier, but also must be capable of passing over a backplane.
4. A separate input/output payload data path must be provided for each of the channel processors as an interface to the rest of the SDR platform. Typically, the payload bandwidth per user channel is relatively

low when compared to the dataflow requirements of the rest of the system, so multiple user channels often share a common physical communication structure through the use of logical channels. The composite data rate of the physical link supporting these logical channels must be sufficient for not only the aggregate data rate required for the combined data paths, but also for any protocol overhead. Additionally, the protocol stack must guarantee that latency and determinism requirements of these components are maintained.

5. A control data path must be provided to each processing element with an independent logical channel often required for each digital radio function. This path can exist as a separate communications infrastructure, or it can share an existing structure, so long as it does not degrade the required data flow between the processing elements. Depending upon the exact application this path may also have some low-latency requirements, and hence if it shares with another channel there must be a mechanism to guarantee this control path some level of service.

#### 4. DATA MOVEMENT TECHNOLOGIES REVIEW

A variety of communications standards are available to address the above requirements. In general, these standards

break down into two categories: Legacy Architectures, and Packet Switched Architectures.

#### 4.1 Legacy Architectures

Legacy architectures represent communications standards that have been in use for some years. These include bussed communications architectures such as PCI, and circuit switched architectures such as Raceway.

##### 4.1.1 PCI

PCI is a bussed communications architecture. The basic standard is a 32 bit multiplexed address & data bus running at 33 MHz, for a theoretical aggregate bandwidth of 132 MB/s. Extensions to 64 bits and 66 MHz allow a maximum bandwidth of 528 MB/s. PCI-X provides further extensions to the PCI bus specification, allowing the bus to run at up to 133 MHz, at 64 bits, for a peak data rate of just over 1 GB/s peak. In addition, PCI-X provides new transaction types that allow packet-like operations.

As a shared bus PCI can limit system reliability and latency determinism, since a single faulty card can capture the bus, resulting in all cards on the bus being unable to function until the faulty card is removed. These issues limit the usefulness PCI bus to primarily payload and control data paths. PCI-X can be architected as a star, assisting with this problem, but other legacy limitations remain.

##### 4.1.2 VME

VME shares many of the basic characteristics of PCI, being a shared parallel bus structure. Like PCI, VME has seen significant evolution over its lifetime. The “VME Renaissance” and speed-up made possible by the new 1eSST signaling scheme has provided performance increases. However at its heart it shares the same limitations as those discussed for PCI.

##### 4.1.3 Raceway and Race++

Raceway is a circuit switched architecture that assumes a classic “star” topology. Each Raceway endpoint connects directly to a Raceway crossbar, with the crossbars routing data from endpoint to endpoint. Race++ provides a 32-bit bus operating at 66.667 MHz for a maximum of 267 MB/s per link. It has support for priorities and routing capabilities in the crossbars, effectively allowing limited logical channels through the fabric. Raceway typically uses an active backplane architecture consisting of one or more crossbar devices. The failure of a crossbar may shut down those cards connected to it, but the ensuing replacement of the active backplane will often require shutting down the entire system. This is a fundamental concern in the use of Raceway in high availability systems.

#### 4.2 Packet Switched Architectures

The emerging requirement to support large numbers of logical channels within the communications infrastructure leads away from legacy bussed and circuit switched protocols to packet-based switch fabrics. Packet switched fabrics contain the common features of a transport layer capable of end-to-end routing and multiple high-bandwidth links. The protocols may be divided into parallel and serial physical layers, resulting in sets of fabrics with relatively similar sets of strengths and weaknesses.

#### 4.3 Parallel Packet Switched Technologies

Parallel packet switched architectures are distinguished by multiple data lines running in parallel with a separate data clock line. Parallel packet switched architectures typically support high link bandwidth. In addition, they often have scalable bus widths and clock rates that allow them to efficiently meet both current and future system requirements.

##### 4.3.1 RAPIDIO

Also known as parallel RapidIO, this is a packet switched protocol developed specifically for embedded systems by the RapidIO Trade Association to emphasize high-bandwidth, low latency communications. RapidIO utilizes low voltage differential signaling (LVDS) with a source synchronous protocol to support data rates per pin ranging from 500 Mb/s to 2 Gb/s. A RapidIO link is full-duplex, using data widths of either 8 or 16 bits.

RapidIO was designed with a relatively small protocol stack to allow for simple endpoint devices. The low protocol overhead (much packet-handling takes place in the silicon) and high bandwidth of RapidIO make it very appealing for inter processor communications on a board, but the relatively high pin count makes parallel RapidIO less useful for board-to-board links.

##### 4.3.2 HYPERTRANSPORT

HyperTransport is a bus-like fabric defined by AMD. This is a source-synchronous protocol that uses differential signaling to support data widths of 4, 8, 16 and 32 bits at signal rates of 800 Mb/s. AMD is now positioning HyperTransport less as an embedded fabric and more as a front-side bus. As such, it is still valuable to embedded systems for interprocessor communications, although it is unlikely to be used for I/O and inter board communications. Hypertransport can support 1.6 GB/s in each direction with an 8-bit data width.

#### 4.4 The Deficiencies of Parallel Packet Switched Technologies for Inter-Board Communications

For interboard data paths, limited backplane connector pins can greatly restrict the number of high-pincount parallel busses that may be routed. Additionally, parallel protocols are susceptible to differential skew between data signals. This is addressed in RapidIO and HyperTransport by including a clock signal for every eight

data signals, but while minimizing such skews across a PCB is a reasonable task, it becomes much more difficult when routing through a backplane in a board-to-board connection. Ultimately, high bandwidth per pin across interboard connections is best obtained by obviating the need for clock/data skew control, as is done with the serial packet switched protocols.

#### **4.5 Serial Packet Switched Technologies**

These have a physical layer that is specifically designed to cope with inter-board skew issues. The clock is embedded into the data signal, typically utilizing an encoding scheme such as 8B/10B. Serial packet switched technologies also usually support transmission over a longer range, making them ideal not only for board-to-board communications, but also for chassis-to-chassis.

##### *4.5.1 SWITCHED ETHERNET*

With the development of the CompactPCI Packet Switching Backplane (cPSB) concept through the PICMG 2.16 standard, Switched Ethernet is readily available for SDR systems. The primary benefit of Switched Ethernet is that the physical connections and protocol are widely used and understood. Also, the cPSB structure allows redundancy to be added to embedded Ethernet. With speeds to 1 Gb/s, Switched Ethernet is an excellent choice for control and payload data as it can connect seamlessly to many other systems.

##### *4.5.2 SERIAL RAPIDIO*

Serial RapidIO is a serial adaptation of the parallel RapidIO protocol. Serial RapidIO supports serial links (1x) at 1.25, 2.5 and 3.125 Gb/s. Higher data rates are supported by combining four 1x links together into a single 4x link. The protocol overlap with parallel RapidIO allows bridge chips to easily connect both parallel and serial links. This factor validates the approach of using parallel RapidIO for intraboard links and serial RapidIO for interboard links, extracting the benefits of each approach in the appropriate domain.

##### *4.5.3 StarFabric*

Originally conceived as an upgrade path to the venerable H.110 bus standard, it uses point-to-point signaling pairs running at a lower speed of 622 Mb/s each. This maps well to certain telecommunication requirements but is rather slow for general-purpose use. Hence StarFabric can also be run in 4X mode, giving the 4 links an aggregate of 2.488 Mb/s. A mapping between StarFabric and PCI has also been developed to facilitate bridging between these standards.

##### *4.5.4 INFINIBAND*

Originally driven by Intel, it was intended to replace Ethernet and Fibre Channel as a high-bandwidth system interconnect for computing clusters and server I/O. Infiniband uses differential signaling at 2.5 Gb/s with 1x, 4x, or 12x data signals to provide raw bandwidths of up to 3 GB/s. Infiniband was designed to communicate over fairly long distances, which at 2.5 Gb/s requires a significant amount of power, making it unsuitable for interprocessor and interboard communication fabrics. The protocol stack requires significant resources, making it, in general, too cumbersome for use as an embedded fabric. Infiniband, therefore, is limited in a distributed transceiver architecture primarily to high speed I/O paths that require communications outside of the chassis.

#### **4.6 Future**

Several communications technologies are under development that could have significant impact on future architectures. Chief among these is PCI Express. Heavily promoted by Intel as PCI's successor, PCI Express uses multiple LVDS transmit and receive pairs. Each pair is run at an initial 2.5 Gb/s with 8B/10B encoding, with the resulting full-duplex link called a "lane". Up to 32 lanes can be combined to increase the throughput of the resulting PCI Express link.

**Table 1 – Interconnects Summary**

<b>Standard</b>	<b>Pros</b>	<b>Cons</b>	<b>Ideal Usage</b>
PCI / PCI-X	Low cost. Simple. Well understood.	Poor QoS. Low BW / pin.	Control and perhaps payload data, within the chassis.
Raceway/ Race++	Simple. Scalable.	Moderate QoS. Low BW/pin.	Interprocessor and board to board communications.
Parallel RapidIO	High BW/pin. Simple protocol.	Higher pin count.	High speed I/O. Interprocessor communications. Short-range board to board communications.
Serial RapidIO	High BW / pin. Low pin count. Simple & common protocol.	Power.	Digitized IF, payload and control data for board to board communications. Suited to passing through, and along, backplanes.
StarFabric	Maps well to H.110 data. Moderate BW/pin.	More complex protocol. Lower speed per pin than alternatives.	H.110 upgrade for higher-capacity systems. Possible PCI bridging extensions.
Hyper-Transport	High BW/pin	Higher pin count. Market positioning (PC focus).	Interprocessor and inter-board communications, likely within a PC environment.
Infiniband	High BW/pin. Powerful protocol sets.	Large / complex protocol set. Power.	High-speed I/O, payload and control data outside the chassis.
Switched Ethernet	Common, well understood.	Limited bandwidth. Protocol overhead.	Payload and control data, inter-board and inter-chassis. Interface to control station.

## 5. TECHNOLOGY SELECTION & EXAMPLE APPLICATION

Table 1 provides a summary of the various technologies. In general, the combination of (Parallel) RapidIO and Serial RapidIO provides the best mix of support for digitized IF, inter-board and inter-processor communications, primarily because these two high-speed technologies share a common, efficient, protocol stack. The fact that several CPU silicon vendors have announced devices with embedded RapidIO ports makes that decision much easier. Gigabit switched ethernet is a strong contender for payload and control data within the chassis and particularly going outside the chassis. Infiniband may be substituted outside the chassis if ethernet's speeds are insufficient.

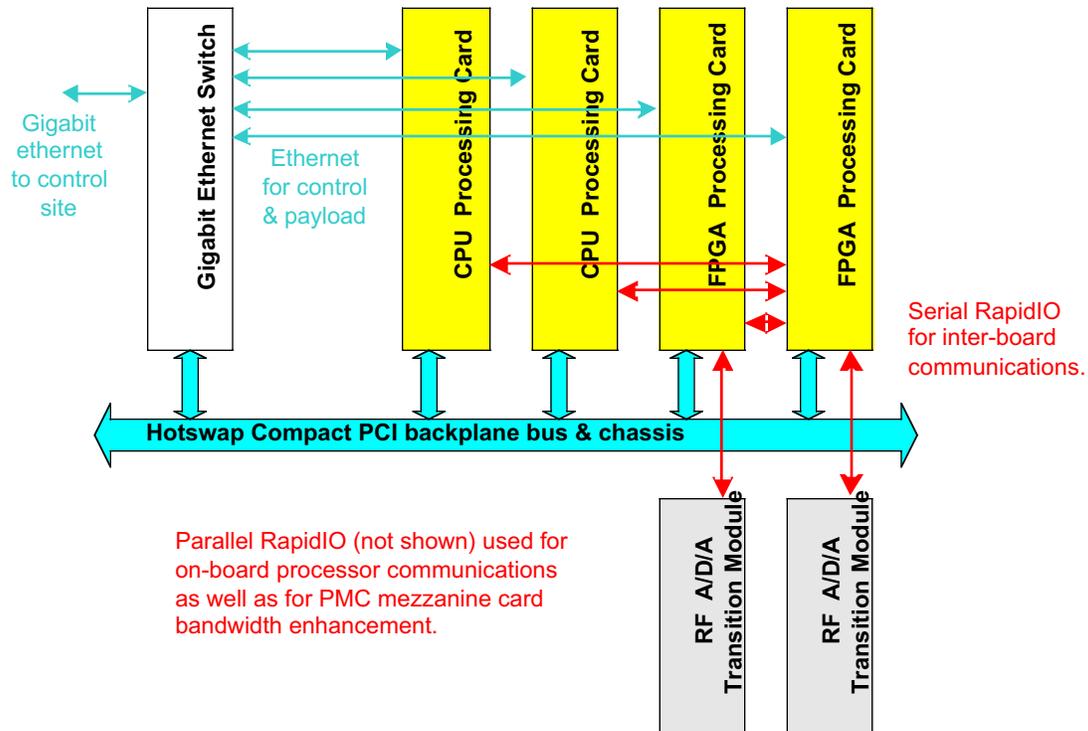
Having selected the data interconnects, an example architecture is shown in Figure 2.

Physically this architecture makes use of the 2.x CompactPCI formfactor to provide a chassis, cooling, power&ground, and a hotswap-capable PCI bus usable

for system initialization and control. Beyond that, the PCI bus need play no other function and may in fact be phased out completely in the future. Within a single processing card, be it FPGA or CPU, Parallel RapidIO is used for on-board communications between processing elements. Parallel RapidIO is also brought to any "enhanced" PMC sites to alleviate the PCI bandwidth bottleneck that would otherwise occur.

Serial RapidIO is used to interconnect all of the cards. This provides the major software advantage that the application need not know whether the processor it's communicating with is on its board or on another; the RapidIO fabric takes care of routing the data, irrespective of whether it's parallel or serial. Serial RapidIO is also used to communicate with the transition modules. Also known as the "Digital IF bus", this is the path containing the raw A/D and D/A data, with aggregate data rates able to exceed 1 GB/s.

All processing cards may communicate via TCP/IP to an ethernet switch sitting in the rack. This switch has a main gigabit feed connected to the outside world, typically a control station or switching centre.



**Figure 2 – Top Level Architecture for SDR Data Interconnects**

This example architecture is easily scaled, both to add redundancy (achieving five-9s availability) as well as to increase processing power and data throughput. By taking advantage of a purely passive backplane any potential active single-points-of-failure are also eliminated.

## 6. CONCLUSION

A high-performance SDR system is only as good as the data it can move. Even the best systems can be throttled by a bottleneck in a critical area. By careful analysis of the different data types that must be moved around an SDR system, followed by careful selection of the technology used to move that data, it is possible to build a scalable, high-performance Software Digital

Radio system to support even the highest bandwidth applications.

This paper has examined the various datatypes inherent in an SDR system, and identified each one. It then performed a significant review of both current and next-generation data busses and fabrics, before selecting RapidIO and switched ethernet as being the most applicable for SDR. Finally it presented an example architecture, showing how these fabrics can be used in conjunction with off-the-shelf standard hardware to build very high performance SDR systems. These types of architectures have the flexibility and power to meet the demands of SDR today, as well as providing a platform for driving SDR well into the future.