

# Building onto Baseband's Serial RapidIO Foundation: Enhancing the IQ Data Pipe with Pre-Processing

Second- and third-generation serial RapidIO devices enhance 3G+ baseband performance and function.

## Serial RapidIO Devices

This is the second of a three part series on how enhancements in second- and third-generation serial RapidIO devices can improve 3G+ baseband processing. Part one demonstrated how serial RapidIO serves as the foundation of the baseband system. (The link can be found at the end of this article.) The third and final article in the series will focus on a real prototype solution using off-the-shelf hardware to implement the baseband and switch boards and their functions.

By Trevor Hiatt, IDT

This article is the second of three articles focusing on how enhancements in second- and third-generation serial RapidIO devices can improve 3G+ baseband processing. The first article in this series demonstrated how serial RapidIO serves as the foundation of the baseband system. This article shows how second- and third-generation devices provide additional proprietary features to further enhance 3G+ baseband performance and function.

## Switch Enhancements

The switch is central to all endpoints and stands at a topological "high ground," as shown in Figure 1. It seems intuitive then that any generic data transforms that must be applied on In-Phase and Quadrature (IQ) data streams can be performed just once in the switch. This is especially effective in the uplink where a low number of high-bandwidth streams must be reformatted according to baseband requirements and often multicast to multiple endpoints. A single switch board might multicast to multiple baseband boards, and similarly, those baseband board switches will likely multicast this data to multiple signal processors.

If the switch can perform this data formatting first, the receiving endpoints will not have to perform the same transforms. Thus, precious digital signal processing (DSP) cycles and power may be saved.

Conversely, on the downlink, the switch serves as a central

aggregation point on the baseband and the switch board(s). Architecturally, this is another key location to provide enhancements over standard serial RapidIO switching with data manipulation capabilities.

## Enhancing the Data Pipe with Pre-Processing Capability

For our example, the packet payload processing functions in the uplink and downlink are for a generic UMTS/WCDMA/TD-SCDMA system, assuming a single IQ data stream supports all potential Antenna Carriers (AxCs) in a given packet and a given number of baseband boards. The amount of baseband boards will each support a unique subset of the potential AxCs, with possible overlap of AxCs among baseband boards. We will partition functions between switch and baseband boards, as shown in Figure 2.

Figure 2 highlights specific operations to be implemented. These functions will be a subset of all possible pre-processing capabilities. Given this specific implementation, actual system performance figures will be presented in the third article in this series. Where appropriate, other possible processing capabilities may be cited, but these will not be reflected in the final system performance figures.

## Uplink: Switch Board

For our implementation on the uplink, IQ data formatting will occur on every packet and will include de-interleaving of I and Q sample data. The resultant IQ streams will then be multicast to all baseband boards. Native support should allow this to be applied to I-only or I and Q streams.

Any control and timestamp information should be protected in a user-defined field within a portion of the packet payload so that it is not corrupted by data processing operations on the remainder of the packet payload.

Since the switch board is central to all baseband boards, it is a key location to provide global data formatting of the IQ streams. Alternative sample manipulation functions might include endianness change, IQ reordering, and/or sample resizing with or without sign extension and deletion.

An alternative implementation might allow the specific subset of AxCs supported by a given baseband board to be extracted from the complete IQ data stream. Thus, any number of unique pre-processing scenarios would be required in support of



Figure 1. This generic baseband implementation shows switch card and baseband cards. In this configuration, serial RapidIO is used on the backplane. A fabric interface chip (FIC) may be used to bridge RapidIO to an air interface protocol, such as CPRI, OBSAI or perhaps a proprietary interface.

the number of baseband boards. The resulting packet payload size will be reduced since it is a subset of the AxCs from the originating payload. The system benefit is a reduction in bandwidth to the baseband cards. Ultimately, this can reduce the number or width of the serial RapidIO ports to the baseband cards, reducing hardware cost and power consumption. For our example, we will take advantage of this AxC truncation on the baseband card.

### Uplink: Baseband Board

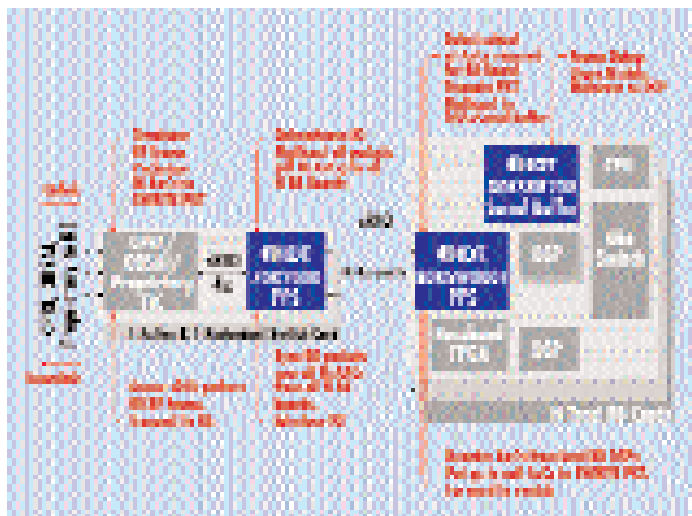
As an optional implementation on the switch card, we will use the AxC extraction capability to select just the subset of AxCs supported on the given baseband card. Since each baseband board will support its own unique subset of AxCs, the switch's pre-processor will be uniquely programmed on each baseband board to reorder just that baseband board's AxCs to the front of the payload, and then truncate the remainder of the packet (AxCs deletion). The resulting smaller packets will then be multicast to the DSPs at a fraction of the incoming data rate. In doing so, the hardware cost and power consumption can be reduced.

The subset of carriers supported on a given baseband card may change dynamically over time. The pre-processor supports dynamic reprogramming of all processing requirements, including AxC selection/truncation, without dropping a single packet during the changeover.

Serial RapidIO natively supports the direct transfer of data into the target endpoint's memory. The packet header maintains the target address so that packets may be delivered directly to a given location in the target's memory. Pre-processing features enhance this by providing multiple output packet generation in tandem with target address generation. Thus, a given resultant packet may be partitioned into several smaller packets, each with its own unique start, stop and increment address values. This allows blocks of data within a packet's payload to be placed in several unique ranges of addresses within the DSP's memory. This enhanced direct memory access (DMA) support provides a level of flexibility to the DSP software engineer.

### Downlink: Baseband Board

Making the roundtrip back, we assume the baseband signal processors for a given baseband board



**Figure 2. WCDMA pre-processing requirements. Upper and lower callouts highlight required pre-processing and frame delay functions on uplink and downlink, respectively.**

will again be forwarding a subset of AxCs to the switch board. We will assume that the packet's AxC arrangement may be out of order with respect to radio card requirements. We will also assume that AxCs will ultimately be summed for all baseband boards upon reaching the switch board.

The subset of AxCs supported on a given baseband card may be out of sequence, with some AxCs not supported at all. Before summing at the switch card, it is critical that AxCs are re-ordered in proper sequence and null carriers are padded with sufficient zeros for the summing AxCs to be correct. Thus, the pre-processing requirements for a given baseband card are to re-order the AxCs within the packet payload and zero any null carriers. For example, the following sequence of AxCs from the DSP memory {AxC12, AxC15, AxC13} must become {0, 0, AxC12, AxC13, 0, AxC15, ... , AxC1M}.

On the baseband card, control information may be embedded in the protected user-defined field within the payload as previously described. Again, the requirement to the pre-processor is to guarantee the user-defined field is protected during payload processing operations.

### Downlink: Switch Board

At the switch board, packets from all available baseband boards are summed. AxC re-ordering and unused AxC zeroing ensures summation across antenna samples will be correct. For example, if the data from baseband board one is

$$\{0, 0, \text{AxC12}, \text{AxC13}, 0, \text{AxC15}, \dots, \text{AxC1M}\},$$

and from baseband board two is

$$\{\text{AxC20}, 0, \text{AxC22}, 0, 0, \text{AxC25}, \dots, \text{AxC2M}\},$$

then the sum result for the two will be

$$\{\text{AxC20}, 0, (\text{AxC12} + \text{AxC22}), \text{AxC13}, 0, (\text{AxC15} + \text{AxC25}), \dots, (\text{AxC1M} + \text{AxC2M})\}.$$

The resultant sum may saturate, be truncated or be shifted bitwise in accordance with system requirements. Finally, IQ is interleaved and the resultant packet is forwarded to the RF card.

A unique challenge within packet-based systems is the synchronizing of packets before summation. Some finite skew will inevitably exist among packets received from the separate baseband boards given the lack of explicit system synchronicity. Unavoidable skew may be incurred by standard serial RapidIO physical layer effects, such as link synchronization or packet retries and synchronization skew among baseband boards. The pre-processor has separate buffer space for each input port and arrival windows to effectively de-skew packets from the baseband boards. Further, if a packet from a given baseband board is simply not received within the user-defined arrival period, the packet may optionally be replaced with zeros (or ones) and the summation will proceed without the packet to avoid stalling the data stream.

Lost packets may be the result of several system issues. One possible cause might be related to bad signaling on the serial RapidIO link or even a link that has failed and must be re-initialized. The pre-processor is able to identify and report a missing packet by sending a Port-Write maintenance packet with error information to the host processor or offending DSP so the situation can be corrected. If physical layer errors exist on the switch ports themselves, similar action may be taken.

The third and final article in this series will focus on a real prototype solution using off-the-shelf hardware to implement the baseband and switch boards and their functions. Latency and throughput performance values for uplink and downlink, as well as power figures, will be provided for each of the components comprising the solution.

## WDD

### About the Author

Trevor Hiatt is the applications manager of the IDT Flow-Control Management (FCM) division. For more information, contact Trevor Hiatt at [trevor.hiatt@idt.com](mailto:trevor.hiatt@idt.com).

### Editor's Note

This is the second article of a three-part series. The first article appeared in the December 2007 issue. Use the following link to read Part I. <http://www.wirelessdesignmag.com/PDFs/2007/1207/wd712f2ol.pdf>.