3G+ Baseband Prototype System Implementation Part 3 of 3

Second- and third-generation serial RapidIO devices provide additional proprietary features to further enhance 3G+ baseband performance and function.

Trevor Hiatt, IDT Flow-Control Management (FCM) Division

This is the third and final article in a series focusing on how enhancements in second- and third-generation serial RapidIO devices can improve 3G+ baseband

processing. The first article in this series discussed how serial RapidIO serves as the foundation of the baseband system. The second article detailed how second- and third-generation devices provide additional proprietary features to further enhance 3G+ baseband performance and function.

This article shows how to achieve an actual 3G+ prototype development platform using off-the-shelf Advanced Telecom Computing Architecture (ATCA)-based hardware to meet the example baseband system implementation described in the second article, with the associated per-component power and latency, latency jitter and throughput for uplink and downlink provided. It also describes how next-generation serial RapidIO device features meet system-level requirements, such as error management, hot-swap and bit error rate (BER) measurement.

Off-the-Shelf Prototype Hardware Selection

For our example, we will utilize off-the-shelf components and hardware to create a prototype baseband system and review associated performance information. Several reasonable switches and associated evaluation boards can handle our switch board’s pure serial RapidIO port count "plumbing." To additionally meet the pre-processing requirements, we will use the EVC70K2000 Evaluation Carrier from IDT.

For the baseband board, a few suitable off-the-shelf AMC cards can minimally meet prototyping requirements. Silicon Turnkey Express' (STx) AMC80KSW0001 board features four Texas Instruments' TMS320C6482 digital signal processors (DSPs) and the IDT80KSW0001 Pre-Processing Switch device. This switch features the pre-processing requirements described for the baseband card.

The 10G Serial Buffer Module Provides UMTS Frame Delay on Uplink

In our particular system, we elected to place a serial buffer local to the baseband board rather than the switch board. Since the pre-processor on each baseband board will truncate the packets, the bandwidth can be reduced and, conversely, a greater incoming In-Phase and Quadrature (IQ) data stream throughput may be supported.

Additionally, it is expected that, to optimize the system throughput, the switch board to baseband board links will run at about 60 percent of throughput just to support the real-time data stream. This allows a 40 percent margin for any eventualities of traffic congestion and subsequent packet retries. Had the serial buffer been placed on the switch board, this would effectively double the required switch board-baseboard link throughput to 120 percent (real time plus delayed traffic), which is not physically possible.

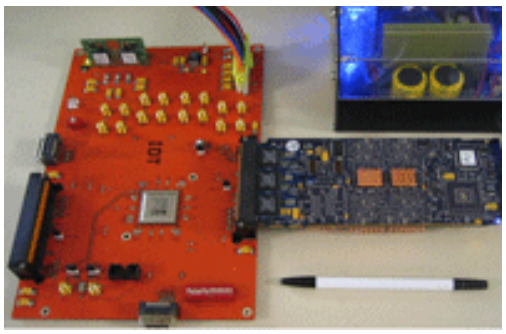
The serial buffer may store a full wireless message transfer system (UMTS) frame or a variable Z number of slots within a frame, user-adjustable via configuration registers.

As the delayed data is re-sent, the packets may have user-adjustable interpacket gap timing. The user may also enable the serial buffer to generate a doorbell interrupt to the receiving processor immediately after a user-defined number of packets (block of IQ data) have been transferred. This minimizes any delay in the processor between data receipt and the start of processing, and improves overall system performance.

The aforementioned AMC80KSW0001 baseband processing board offers an expansion port supporting STx's 80KSBR200 10G serial buffer module to meet the UMTS/CDMA frame delay buffer requirement.

Putting It All Together: The Final Word Is Performance

Figure 1 reflects the component performance figures (power, latency and latency jitter) for our particular system example. The pre-processors were used exclusively on the uplink and



downlink data pipe, and the figures reflect the performance of the pre-processors for the data operations given in our example. The pre-processors are dedicated, provisioned paths and there is no latency jitter in the pre-processing data path &#151 the 35 ns latency jitter is purely related to the standard serial RapidIO physical layer effects. The pre-processor's best-in-class latency jitter saves overall system latency and thus the jitter budget to meet system synchronization requirements.

Though the data represents the uplink and downlink data paths to the baseband board, it should be expected that, during the primary baseband processing, depending on the algorithms required and associated partitioning, multiple hops among DSPs and the FPGA through the switch block itself will be required. It is critical then that switching latency is kept to a minimum because the total jitter will increase by multiples of the number of hops required. Typical latency for pure switching is 160ns and associated jitter is 35 ns (assuming no blocking at the output port by other packets and priorities).

The pre-processors offload the DSP so that they may be used toward their primary algorithmic functions rather than data formatting requirements. This saves available cycles in the DSP. Consider, for example, that each pre-processor operation (de-interleaving) requires at least one processor cycle per sample (some operations require two or more). Consider that the DSP must perform this operation on each sample within the payload, real time on the data stream. For a typical 1GHz processor, depending on wireless system parameters, 20 percent of available processor cycles or more may be consumed by performing even a single function on all IQ samples.

Error Handling

The completed system needs to be robust enough to detect, report and even correct systemic failures. Thus, error-handling capability cannot be taken lightly in a baseband infrastructure, which has stringent reliability and availability requirements, complicated by the distance of remotely deployed equipment. Serial RapidIO provides a robust set of error-reporting capabilities through the error management extension specification and software assist error recovery registers. Thus, many error conditions may be detected and even reported for those devices in which these are implemented. However, both of these are optional implementations. For example, if a field replaceable unit has a bad connection with the backplane, a classic symptom is loss of lane sync or bad characters seen at the

receiver. Robust error handling support would identify this connectivity issue, log the error and report this back to a host processor where corrective action can be taken.

It is particularly critical for switches to implement robust error-handling capabilities since they connect with every endpoint in a system. System designers should expect current generation devices to offer serial RapidIO-based features, and even enhancements beyond these. Hardware may be enhanced to detect &#151 with resolution beyond the serial RapidIO specifications &#151 the types of errors seen not only on the serial RapidIO ports themselves, but also on any other functional blocks. This includes proprietary functional blocks on the devices. They must not only provide error detection, but also hardware-based error responses. These responses should minimally include the ability to report errors back to a host processor. Ultimately, the action taken is best served by software interrupt service routines, allowing the most flexibility to the error management and controls software developer.

Maintaining System Bit Error Rate

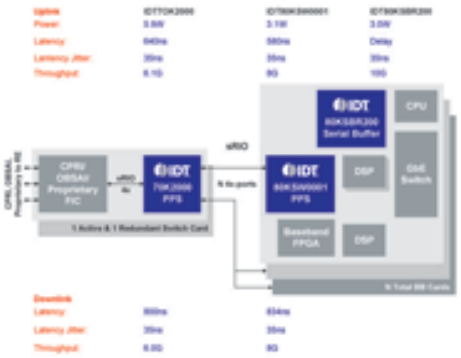
The serial RapidIO physical layer specifications provide explicit requirements to maintain 10E-12 BER or better. Component providers meet the requirements by providing transmitters with drive strength and pre-emphasis controls to meet serial RapidIO-specified minimum transmission and receive requirements. Many vendors offer transmission capabilities beyond the minimums required by the serial RapidIO specifications.

Enhancements to the standard include proprietary features that support measurement of system BER. This is especially useful in fine tuning transmission characteristics to optimize BER on prototype boards prior to production. Features may also support "run time" measurement of BER of production boards so that the BER outside of system requirements may be reported, and transmission characteristics (drive strength and pre-emphasis) may be adjusted to resolve the issue.

Though serial RapidIO does not currently specify these features, there are industry standard SerDes built-in self-test capabilities, such as Pseudo Random Bit Sequence (PRBS) generators and corresponding error checkers, and loopback capabilities. Further enhancements include stressed eye capabilities to allow added confidence by testing at tighter margins. The system designer should take into account the enhanced proprietary feature sets of these devices to ensure BER may be tested, monitored and adjusted &#151 especially between devices from different vendors, which may have unique implementations.

Hot Swap Support

The serial RapidIO standard provides facilities to support hot-swapping of boards. This complements the ATCA management functions built into boards, such as the AMC70K2000 baseband prototype board. The



aforementioned optional software assist error recovery registers allow resynchronizing transaction AckIDs in link partners when the link has been disrupted and there may be a mismatch between AckID values between ports on separate boards. This is particularly useful to support field replaceable switch and/or baseband boards as described in our example system.

Per serial RapidIO specification, reset control symbols may be issued by a host to reset the entire device, and thus, the ports themselves. This will undoubtedly reset any AckIDs but may not be appropriate for a run-time system &#151 since this will reset the device back to power-up values and reset route tables.

For run-time systems, configuration and routing information must stay intact or traffic will fail to be routed through the system. A non-serial RapidIO standard override to the full device reset that will only reset the port that receives the reset control symbol is implemented to support this process. Thus, even if the link partner’s have their AckID’s out of sync and packets can no longer be transmitted on that link, one of the partners may still issue a reset control symbol at the physical layer to reset just that port &#151 and not the entire device.

Configurable Ports to Support Modular Hardware

Serial RapidIO specification defines a standard 1x or 4x port. The 4x port can downgraded to a 1x port on lane 0 or lane 2 for redundancy. This 1x/4x port mapping has been adopted for ATCA’s and microTCA’s subsidiary serial RapidIO fabric specifications.

Although most baseband systems have explicit bandwidth and, therefore, port rate and width requirements, some are extensible to allow the same hardware to bridge femto, micro, pico or macro system borders. To facilitate this, look for serial RapidIO components, especially switches, to allow enhanced levels of configurability beyond the serial RapidIO specification. Enhanced ports may be software reconfigured to support multiple 1x ports from a given 4x port. This optimizes pin count utilization and ultimately saves cost for the hardware designer.

Conclusion

Serial RapidIO&#153 has become the embedded interconnect of choice and serves as the foundation of the baseband system. Second- and third-generation serial RapidIO devices provide additional proprietary features to further enhance 3G+ baseband performance and function. About the Author

Trevor Hiatt is the applications manager of the IDT Flow-Control Management (FCM) division. He is responsible for product definition, product documentation, system-board and software development, customer support, and field and sales training for the division's wireless networking products. Hiatt has patents pending for the company's statistics engine device. For more information, contact Trevor Hiatt at trevor.hiatt@idt.com.

Editor's Note:
Parts 1 and 2 of this article can be found at www.wirelessdesignmag.com [1] and www.wirelessdesignmag.com [2]respectively.

Source URL (retrieved on *03/12/2014 - 7:27am*):
http://www.wirelessdesignmag.com/product-releases/2008/05/3g-baseband-prototype-system-implementation-part-3-3?qt-blogs=0

Links:
[1] http://www.wirelessdesignmag.com/PDFs/2007/1207/wd712f2ol.pdf
[2] http://www.wirelessdesignmag.com/PDFs/2008/wd84f2.pdf