Combining High-Speed Interconnect with Ethernet Systems

Devashish Paul
Sr. Product Manager
IDT Communications Division
This presentation shows how to architect hardware and software to leverage 20G Serial RapidIO's lowest power & latency, and hardware-controlled fault management when bridging to an Ethernet backhaul. This is a common need for the similar architectures of basestations and servers.

- Serial RapidIO: Brief Intro
- When to use RapidIO/What tool for what job?
- Protocol background and comparison
- Architecture options for bridging Ethernet to RapidIO
- Translating between Ethernet and RapidIO
- Architecture examples
  - Mobile Broadband Wireless Basestation
  - Servers: Blade & Micro
In most systems, data comes from somewhere:
- Camera
- RF Domain for radar or wireless
- Data from Users (ex: Internet)

Typical Interfaces for this:
- Streaming (ex: CPRI/OBSAI)
- Ethernet
- PCIe (ex: graphics)
- Proprietary

Once Data is Received, Processors must perform heavy lifting “inside the box”:
- Could be single processor
- Multi Processor
- A variety of processing elements

Typical Interfaces for this:
- RapidIO
- PCIe
- Hypertransport
- Infiniband
- Interlaken

Almost all systems are networked:
- Need to get to other users
- Need to get to storage
- Need to access applications or processing in another location

Interfaces for this:
- Ethernet
By Application: Interfaces Working Together

**Application**

- **Base Station**
  - Remote Radio Head

- **Military Aerospace**
  - Analog Sensor (ex: Antenna Or FLIR)

- **Image Processing & video**
  - Analog Sensor (ex: Camera, MRI)

- **Server And HPC**
  - Users on Internet

**Data Acquisition**

- CPRI

**Processing**

- **Baseband Processing**
  - RapidIO

- **Payload Processing**
  - RapidIO

- **FPGA or DSP cluster**

**Network Domain**

- **Network**
- **Display Or Network**
- **Display Or Network**
- **Network And Storage**
What is RapidIO?

- A high speed serial switched interconnect for embedded
- Open specification developed by the RapidIO Trade Association
- Existing ecosystem of NPUs, CPUs, DSPs, FPGAs, Switches, Bridges
- Carrier-grade reliability for intra-board, inter-board, backplane, and chassis-to-chassis
- Today: Ecosystem-wide support for 20 Gbps port speed
  - Over 2.5 Million switches shipped
- Tomorrow: 10xN
- Future: 25xN

Key Applications RapidIO 1.2 and beyond

- DSP and Processor Farms
- Wireless 3G, WiMax, 4G and future 5G
- Video servers, IPTV, HDTV, Media Gateways
- microTCA, AMC, PMC
- VME, VSX, VPX systems
- Storage/Server Systems
- High Performance Computing
When To Use RapidIO

RapidIO on board
- as the single, simple interconnect among all board components

RapidIO On the backplane
- Future proof
- High throughput
- Low deterministic latency
- Guaranteed packet delivery
- Prioritized traffic

RapidIO for fault tolerant Systems
- Flexible sparing strategies
- Continued system operation in the event of single faults
- Rapid detection of faults
- Flexible response to faults

Protect your SW investment
- S-RI O logical layer remains the same across different physical layer
- RapidIO scales per port
- Saves system total power
Why RapidIO in the Embedded Systems

- Disruptive Architecture, that changes the overall economics of deployment in all multi-processor application
- **Scalable solutions** for board, backplane and inter-chassis communication
  - Extend overall system solution by aggregating chassis
- Lowest overall system power with S-RIO
- Superior **end to end packet latency**, throughput and fault tolerance
- Flexibility and scalability to evolve system configuration in field
- Large Ecosystem with native endpoint support
- Incumbent interconnect in all top 10 wireless OEM 3G and 4G base station designs

IDT RapidIO over 2.5 million switches shipped
400% year over year revenue growth

Silicon Partners with RapidIO

- Texas Instruments
- Altera
- Freescale
- Lattice Semiconductor
- Wintegra
- Applied Micro
- Cavium Networks
- Octasic
- MDSPEED
- Xilinx
3 Layer Protocol All Terminated in Hardware

Off loads processor from protocol termination stack
Saves Power and CPU cycles
Ultra low end to end system latency
Why RapidIO Inside the Box?

- **RapidIO Gen2**
  - RapidIO ideal for peer multi processor performance
  - Best cut through latency 100 ns, 10 GigE cut through latency of multiple hundreds of ns
  - Guaranteed delivery mechanisms
  - Payload rate 19 Gbps for S-RIO2, 8 Gbps for 10 GigE (256 byte packets)
  - Better per-port economics than 10 GigE switches

---

### Bandwidth and Per Port Economics in Embedded

<table>
<thead>
<tr>
<th>System Requirement</th>
<th>10G Ethernet</th>
<th>RapidIO2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch per-port performance raw data rate</td>
<td>10 G</td>
<td>20 G</td>
</tr>
<tr>
<td>End to end packet termination</td>
<td>&gt;10 ms</td>
<td>~1-2 us</td>
</tr>
<tr>
<td>Messaging performance</td>
<td>Poor</td>
<td>Excellent</td>
</tr>
<tr>
<td>Volume pricing $ per 10G</td>
<td>&gt; $10</td>
<td>&lt; $4.00</td>
</tr>
<tr>
<td>Overall system power</td>
<td>High</td>
<td>Lowest</td>
</tr>
</tbody>
</table>
RapidIO and Ethernet Routing Overview

- RapidIO maps Dest ID to Output Port using an indexing operation.

- Routing can begin after the first 4 bytes of the packet have been received, minimizing latency.

- Ethernet routing requires packet parsing after L2 Header.

- IP routing uses expensive Longest Prefix Match search for 4 (IPv4) or 8 (IPv6) byte addresses, which increases latency and requires additional power i.e external TCAM.

- VLAN/MPLS routing uses indexing operation, but also requires pushing/popping VLAN/MPLS tags from the packet, which requires more latency.

Superior, Low Cost, Low latency routing in Embedded
Low, Deterministic Latency

RapidIO

- High priority packets sent first under network congestion
- Lossless delivery guaranteed

Ethernet

- Packets discarded due to congestion
- Transmission errors cause packet discard

- Error recovery in < 300 nsec

- Recovery from packet discard requires timeout in TX SW Stack/TOE
  - Timeout must be set high to account for congestion in the switch/network (milliseconds to seconds)
RapidIO for Processor Aggregation inside box

- Ethernet has no small switch offering for 4 ports or 8 ports.
- All Ethernet switches are massive, consumer 30-40W, and large real estate
- This means it is hard to aggregate a moderate number of endpoints on one PCB like a line card
- Possible with RapidIO
- Each line card can then be aggregated up to a backplane also with RapidIO switching
RapidI/O and Ethernet Bridging Options

- Ethernet typically bridged to RapidI/O by NPU, microprocessor, FPGA or other component with bulk DRAM for frame buffering
  - NPUs support highest capacities

- System designer’s option of where to bridge: Ethernet or S-RIO on backplane

- RapidI/O on backplane
  - Reduces cost (fewer bridges)
  - Leverages RapidI/O Flow Control
  - Reduces power (fewer components / packet termination)
Seamless RapidIO/Ethernet Networking

- Ethernet
- RapidIO ↔ Ethernet Encapsulation
- Native RapidIO Messaging

- Map IP addresses to/from RapidIO Device IDs
- Map VLAN/MPLS priority to/from RapidIO priority
- Could implement TCP/IP stack for small systems
- Straight forward implementation using Type 9 packets
Applications with RapidIO and Ethernet
Wireless BaseStation Architecture

- Ethernet and RapidIO connect all WCDMA and 4G / LTE base stations on the planet
  - Every call, app download, email, & web page
- Ethernet + 20G RapidIO is the requisite architecture to meet the exponential increase in wireless traffic
- Backhaul: 1GbE or 10GbE
- Backplane: RapidIO
- Baseband Subsystem: RapidIO
- NPU performs Ethernet to Serial RapidIO bridging / translation
  - Freescale, Cavium, LSI, AMCC
PCIe to S-RIO bridge and Gen2 S-RIO Switch brings x86 CPU into RapidIO based Servers

- Multiple links each 20G
- Multiple storage, networking, computing and chassis links
- Network Interface Via Ethernet using NPU
- Micro-server requires low-cost, low power, low-price (aka integrated IO) solution
- Network I/O Module to LAN supported by existing NPUs
Summary

- Use Right Tool for Right Job
- RapidIO ideal for embedded systems with multi processor
- Low latency
- Highly scalable
- Bridge easily to Ethernet for network and Data Acquisition
- Translating between Ethernet and RapidIO is easy
- RapidIO is the standard in 3G and 4G wireless base stations
Backup
Translating between Ethernet and RapidIO

- Mapping Ethernet flow to RapidIO flows is a simple lookup operation
  - Each uses an address and protocol-specific value
    - Ethernet: IP or MAC address, port number
    - RapidIO: device ID, stream ID, and / or Mailbox number

- Strategy for transporting Ethernet over RapidIO
  - For TCP: NPU terminates Ethernet packet & forwards message on RapidIO
    - Today's processors terminate 10Ks of TCP sessions at 20Gbps line rate
    - “Zero CPU cycles”
  - For UDP (or TCP that cannot be efficiently terminated): Encapsulate in RapidIO
    - At 20 Gbps line rate
    - Ethernet encapsulation is native part of RapidIO specification
    - Using Messaging or Data Streaming packets
    - Processing for encapsulation requires only address translation and assignment of priority / Virtual Channel
  - Essentially the converse story in reverse direction

- RapidIO’s comprehensive flow control ensures QoS for Ethernet flows [1]
**RapidIO and Ethernet Quality of Service**

<table>
<thead>
<tr>
<th><strong>RapidIO</strong></th>
<th><strong>Ethernet</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>8 Byte Minimum</td>
<td>64 byte Minimum</td>
</tr>
<tr>
<td>276 Byte Max</td>
<td>9216 byte Max</td>
</tr>
</tbody>
</table>

- Small HOL impact per packet
- Flow control control symbols embedded within packets

- Large HOL impact per packet
- Flow control packets delayed by larger packets

- High priority packets sent first
  - Guaranteed Delivery

- High “priority” packets sent first
  - Packets discarded to clear congestion

- RapidIO Virtual Channels (VC)
  - VC0 – Latency Guarantee
  - VC1-8 – Throughput Guarantee

- TCP protocol slow to adapt, ineffective in DCE
  - VPN complex scheduling
  - Deep packet inspection & separate hardware may be required!
RapidIO and Ethernet Protocols Overview

### RapidIO

- **Physical Layer**: 2 bytes
- **Src & Dest ID**: 2 or 4 bytes
- **Read/Write Header**: 4 to 10 bytes
- **Data Stream Header**: 2 or 4 bytes
- **Data**: 256 bytes
- **CRC**: 2 bytes

### Ethernet

- **Preamble**: 8 bytes
- **L2 Header**: 14 bytes
- **VLAN/MPLS**: 4 bytes
- **IPv6 Header**: 40 bytes
- **TCP**: 20+ bytes
- **RDMA**: 13-17 bytes
- **Data**: ~1480 bytes
- **FCS**: 4 bytes
- **IPG**: 12 bytes

### Transfer Time Comparison

<table>
<thead>
<tr>
<th>Protocol</th>
<th>256 B Message (Bytes)</th>
<th>4K RDMA</th>
</tr>
</thead>
<tbody>
<tr>
<td>RapidIO</td>
<td>276</td>
<td>4372 (16 packets)</td>
</tr>
<tr>
<td>Ethernet</td>
<td>378</td>
<td>4501 (3 packets)</td>
</tr>
</tbody>
</table>
## Comparison of Ethernet & RapidIO

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Ethernet</th>
<th>RapidIO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Layer 2</td>
<td>Layer 3+ (Logical Layer)</td>
</tr>
<tr>
<td>Payload size (Bytes)</td>
<td>46-1500 (802.3) 46-9192 (Jumbo)</td>
<td>26-1460 (802.3) 26-9172 (Jumbo) 1-256 (Jumbo)</td>
</tr>
<tr>
<td>Memory Mapped (Read/Write)</td>
<td>No</td>
<td>RDMA</td>
</tr>
<tr>
<td>Write with Response</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Address Size</td>
<td>N/A</td>
<td>64-bits (RDMA) 34, 50, 66-bits</td>
</tr>
<tr>
<td>Messaging Support</td>
<td>No</td>
<td>TCP (among others) 64 KB User Payloads (Data Streaming)</td>
</tr>
<tr>
<td>Shared Memory</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Deadlock Avoidance</td>
<td>N/A</td>
<td>L3+ Must address Pervasive HW Support</td>
</tr>
<tr>
<td>Topologies</td>
<td>Any</td>
<td>Any</td>
</tr>
<tr>
<td>Delivery Service</td>
<td>Best Effort (std) 'Lossless' (DCB) Guaranteed (TCP, SCTP, others) Guaranteed, with option of Best Effort per VC</td>
<td></td>
</tr>
<tr>
<td>Routing</td>
<td>MAC Address</td>
<td>IP Address</td>
</tr>
<tr>
<td>Maximum Endpoints</td>
<td>2exp48</td>
<td>2exp32 (IPv4) 2exp128 (IPv6) 2exp8 (small) 2exp16 (large)</td>
</tr>
<tr>
<td>Header fields which change link-to-link</td>
<td>None</td>
<td>TTL, MAC Addr, FCS, MPLS TTL AckID (all) Hop Count, CRC (Maint Only)</td>
</tr>
<tr>
<td>Redundant Link Support</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Contact Info & References

- **Contact:**
  - Devashish.Paul@idt.com
  - Senior Product Manager
  - Communications Division

- **Integrated Device Technology (www.IDT.com):**
  - Integrated Device Technology, Inc., the Analog and Digital Company™, develops system-level solutions that optimize its customers’ applications. IDT uses its market leadership in timing, serial switching and interfaces, and adds analog and system expertise to provide complete application-optimized, mixed-signal solutions for the communications, computing and consumer segments. Headquartered in San Jose, Calif., IDT has design, manufacturing and sales facilities throughout the world. IDT stock is traded on the NASDAQ Global Select Stock Market® under the symbol “IDTI.” Additional information about IDT is accessible at [www.IDT.com](http://www.IDT.com). Follow IDT on [Facebook](http://www.facebook.com), [LinkedIn](http://www.linkedin.com), [Twitter](http://www.twitter.com), and [YouTube](http://www.youtube.com).

- **References:**
  - RapidIO Trade Association Education [http://www.rapidio.org/education/technology_overview/](http://www.rapidio.org/education/technology_overview/)