

Integrating FPGAs in High Performance Computing

A System, Architecture, and Implementation Perspective

Nathan Woods
XtremeData

FPGA 2007

Outline

- Background
- Problem Statement
- Possible Solutions
- Description of one commercial solution
 - XtremeData XD1000 FPGA Co-processor
 - University Program

Commodity x86 Server Background

- Intel Xeon, AMD Opteron dominate
- Two-socket most popular, 4-socket growing
- Popular form factors
 - 1U and 2U rack-mount (U=1.75 in)
 - Blades
- The rise of the blade
 - Increased compute density (2x or more)
 - Easier to maintain (fewer cables, hot-pluggable)
 - Improved reliability (fewer cables, improved management s/w)
 - Growing rapidly
 - 2004, 7% of servers sold were blades
 - 2008, 30% of servers sold will be blades (IDC estimate)
 - Proprietary form factors ☹
 - Very little room for I/O cards on CPU blade

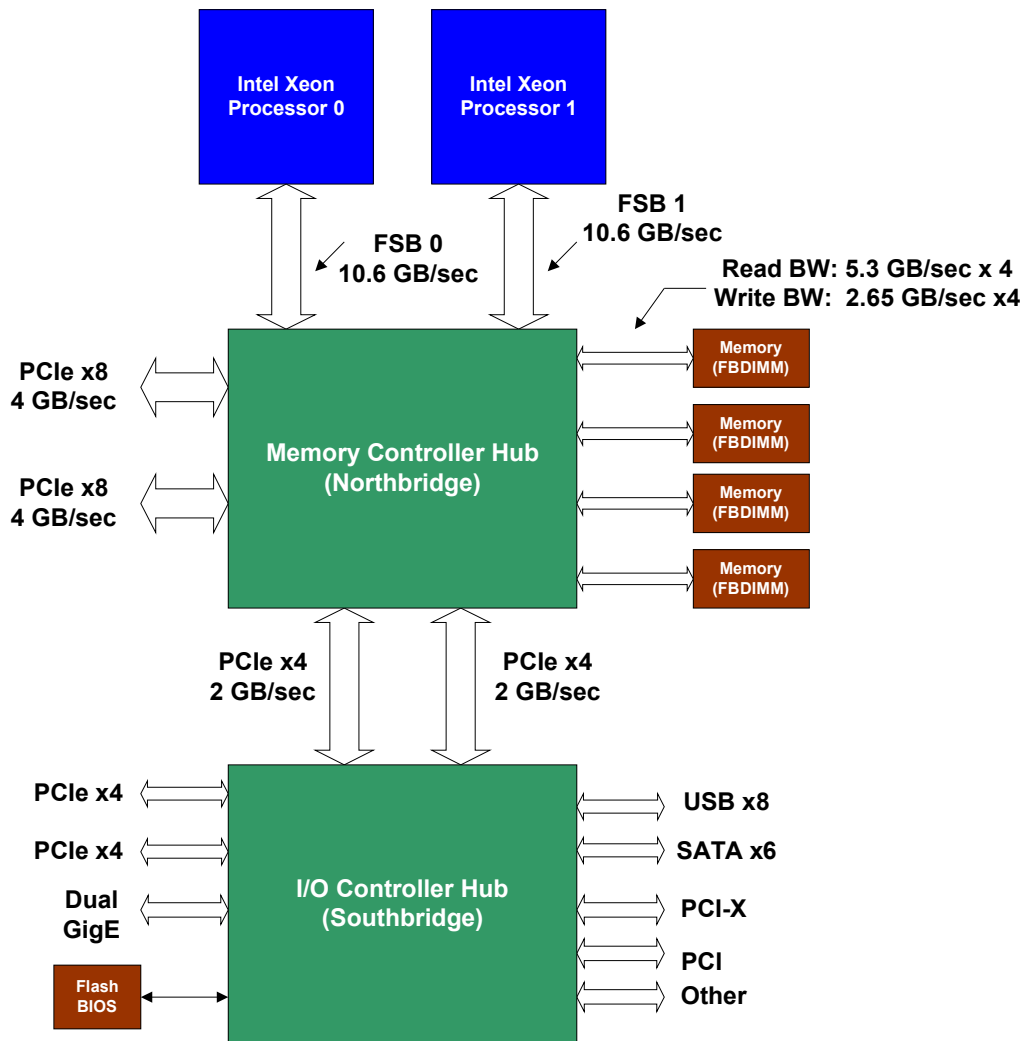


IBM 1U server



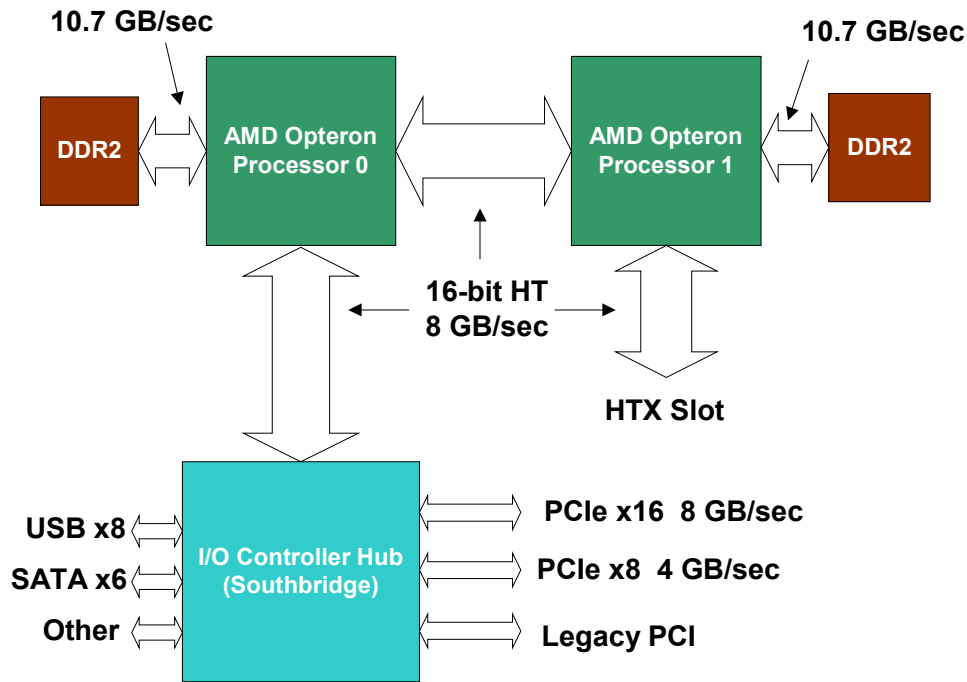
7U rack with 14 IBM LS21 Blades

Inside the Server: Intel Xeon Architecture



- Northbridge + Southbridge
- Large Front-side bus B/W
- FBDIMMs: asymmetric B/W
- PCI Express available at northbridge (lower latency)
- Observations:
 - x16 PCIe ports in Intel-based servers are still rare
 - One PCIe x8 slot often used by Infiniband HBA card

Inside the Server: AMD Opteron Architecture



- Opteron's integrated mem controller eliminates northbridge
- 3 HyperTransport (HT) links per Opteron
- HT
 - High-speed serial LVDS, point-to-point
 - Not 8/10b encoded (DC coupled)
 - Max theoretical bandwidth for x16 link @1 GHz DDR: $\sim 0.85 * 4 \text{ GB} = 3.4 \text{ GB/sec}$ in each direction
 - Latency $\sim 300 \text{ ns}$ (estimate)

FPGA Hardware Integration Issues

- Mechanical
 - FPGA hardware has to fit
 - Variety of commodity server form-factors
- Power Supply
 - Operate within supply limits of server
 - Ex: PCI slot: 25 W
 - Ex: PCI Express: 25 W initially, 75 W after config
- Thermal
 - **Must guarantee server + FPGA h/w thermal solution**
 - Requires detailed thermodynamic simulation of server to guide design
 - Experimental validation of solution with actual hardware
 - Effort: 1 to 2 man/months per server platform
 - **Problem: tier-1 vendors don't do this for FPGA cards today**
- Connectivity
 - Must provide communication link between FPGA and rest of system

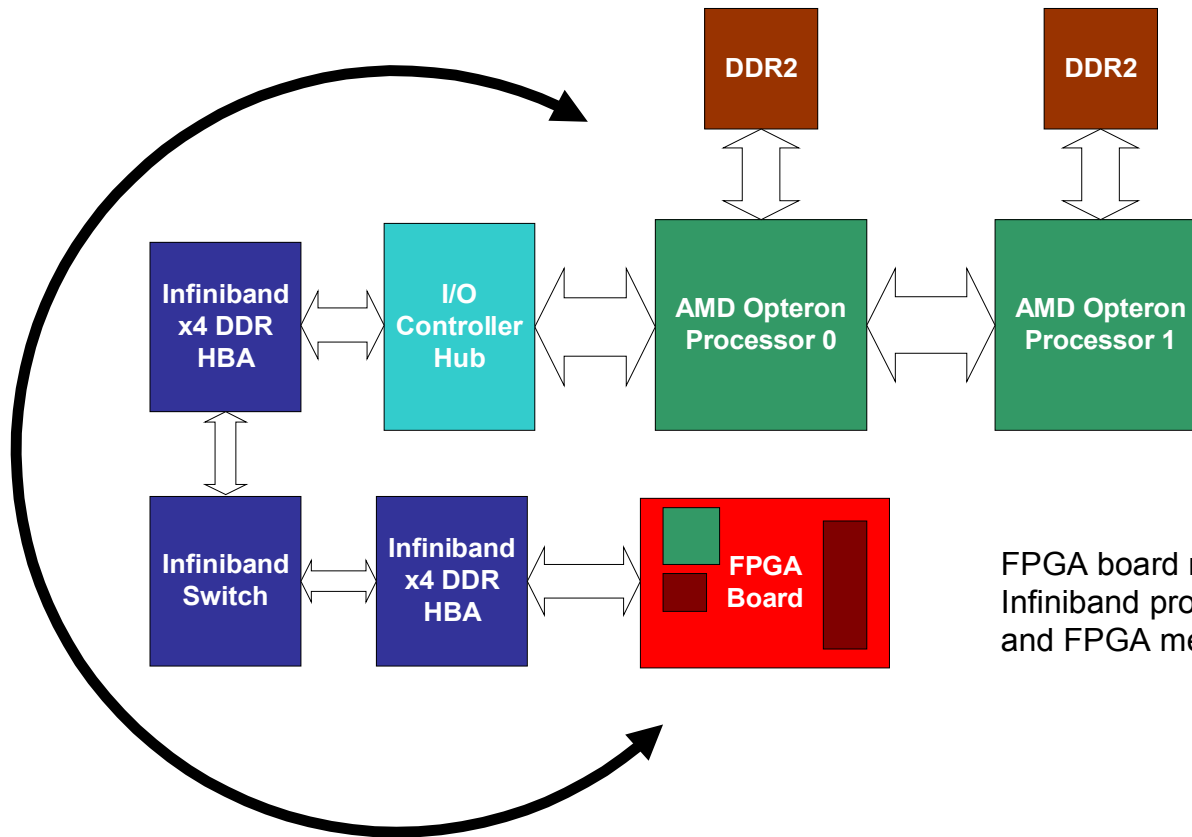
Problem Statement

- Design a piece of hardware that integrates an FPGA with a commodity x86 server subject to the following constraints:
 - Maximize bandwidth between CPU and FPGA board
 - Minimize latency between CPU and FPGA board
 - Maximize FPGA memory resources (b/w & space)
 - Maximize number of compatible server form factors
 - Maximize reliability of overall system
 - Minimize time to market
 - Minimize NRE costs
 - Minimize unit cost

Integration Options

- List of all relevant sockets/plugs on an x86 motherboard:
 - Network socket
 - Disk socket
 - I/O expansion socket
 - Memory DIMM socket
 - Processor socket
- Consider each option in terms of bandwidth and latency to keep things simple

Integration Option: Network Attached



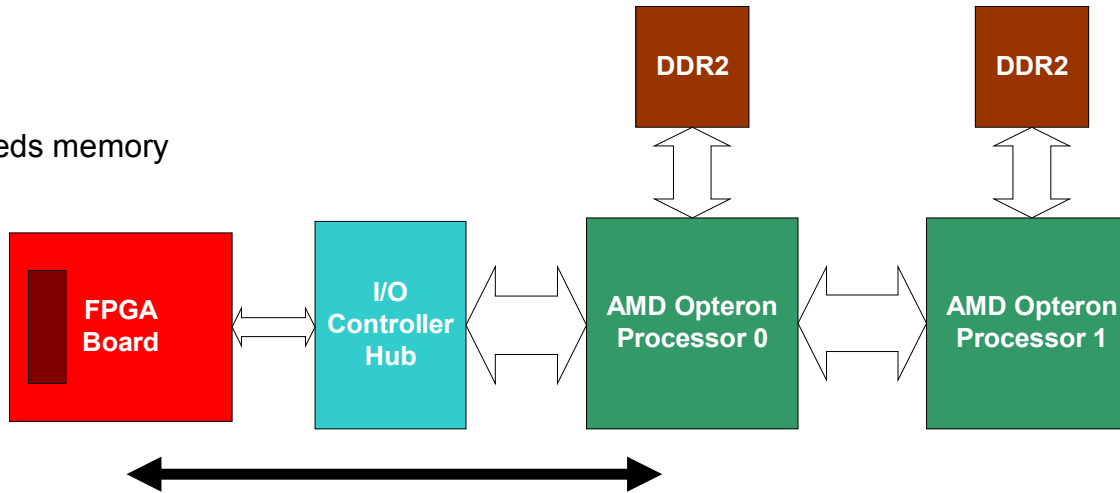
FPGA board needs CPU to run Infiniband protocol stack, CPU memory, and FPGA memory

***Bandwidth: 2.7 GB/sec**

***Latency: 5 to 10 μ s**

Integration Option: Disk Attached

FPGA board needs memory



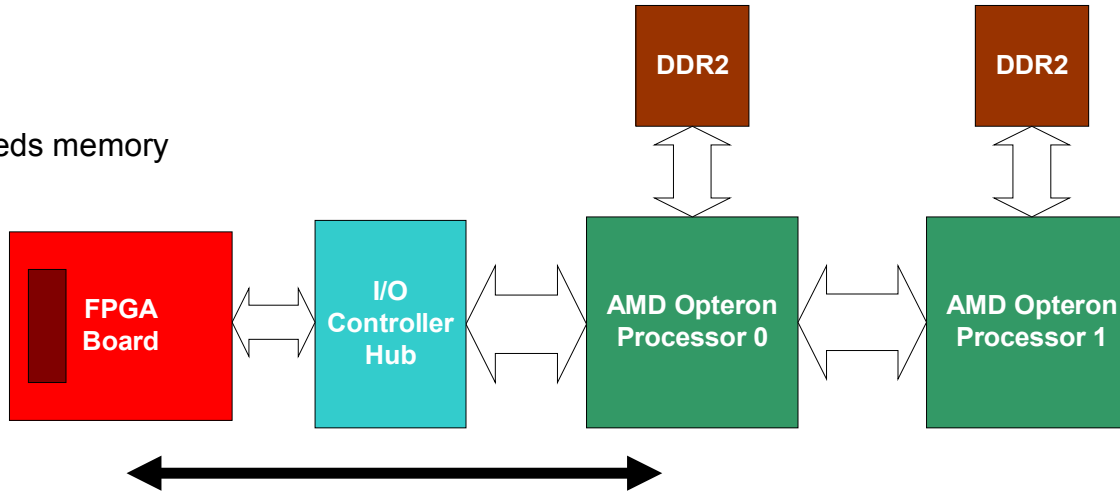
Assuming SATA 2:

Bandwidth: 300 MB/sec (if lucky)

Latency: (who cares)

Integration Option: PCI Express Card

FPGA board needs memory



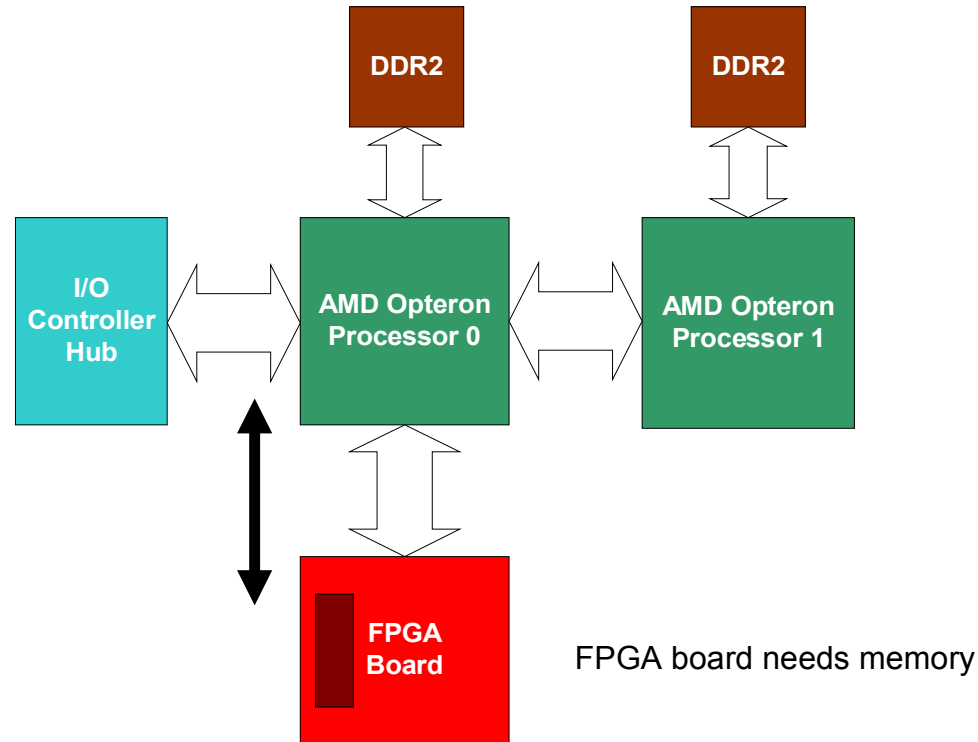
Assuming PCIe 1.0a x8:

Bandwidth: $0.9 * 0.8 * 2.5 \text{ GB/sec} \times 2 =$

3.6 GB/sec (peak theoretical)

Latency: 1 us (est.)

Integration Option: HTX Card



Assume 16-bit HT @ 1 GHz DDR:

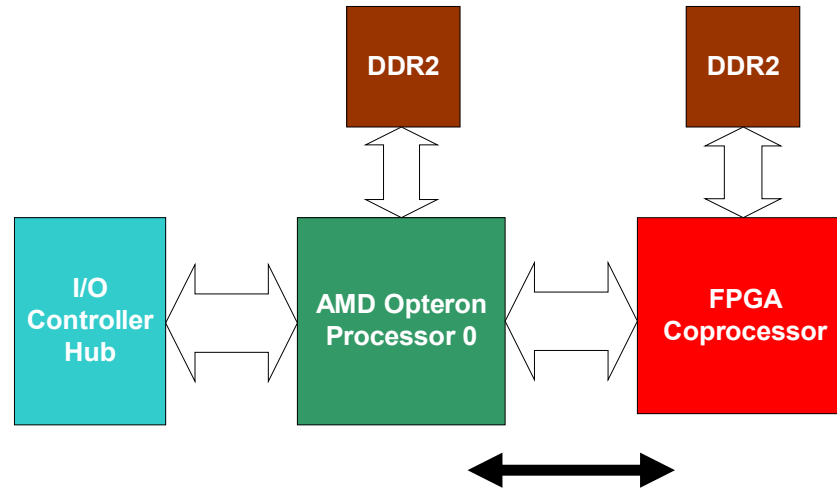
Bandwidth: $0.82 * 4 \text{ GB/sec} * 2 =$

6.5 GB/sec (peak theoretical)

Latency: 300 ns (est.)

FPGA 2007

Integration Option: External Coprocessor



FPGA board can use DDR2 SDRAM on the motherboard.

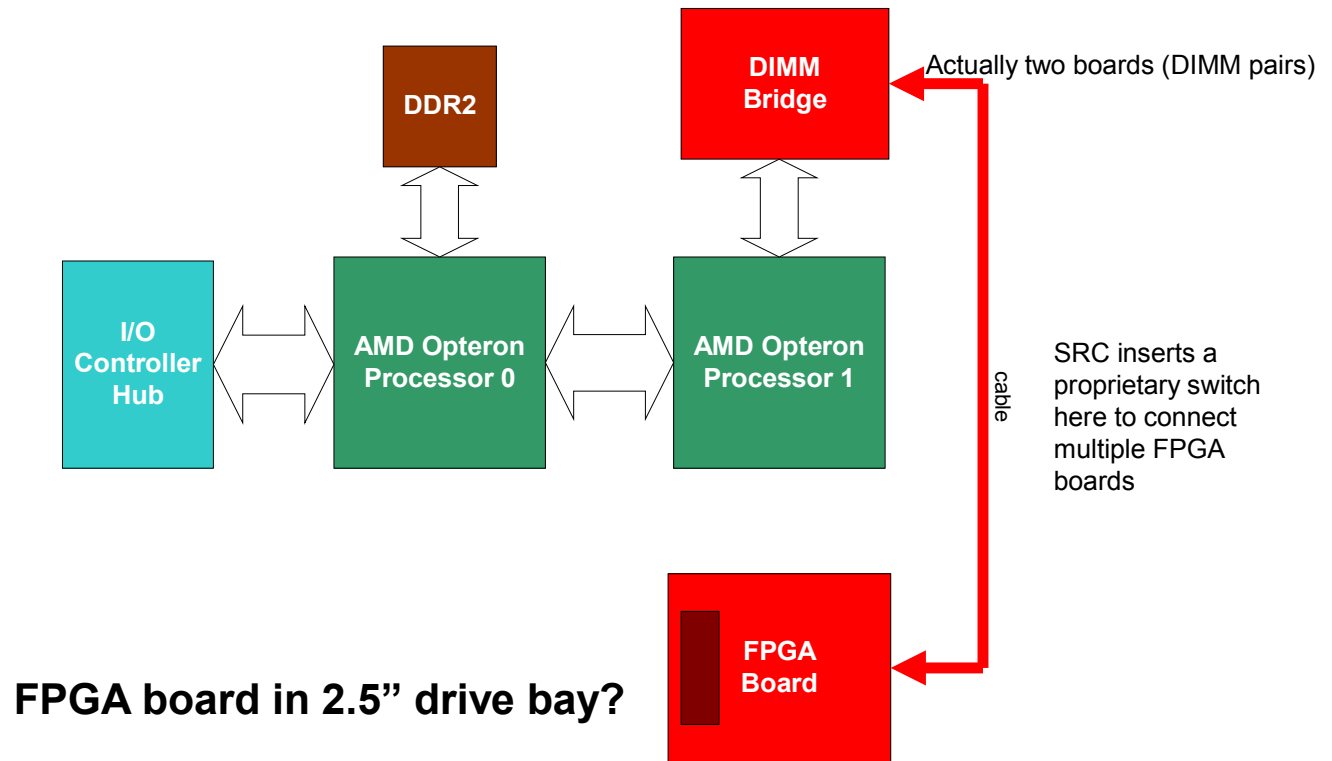
Can also leverage motherboard power supply and CPU cooling solution.

Assume 16-bit HT @ 1 GHz DDR:

Bandwidth: 6.5 GB/sec (peak theoretical)

Latency: 300 ns (est.)

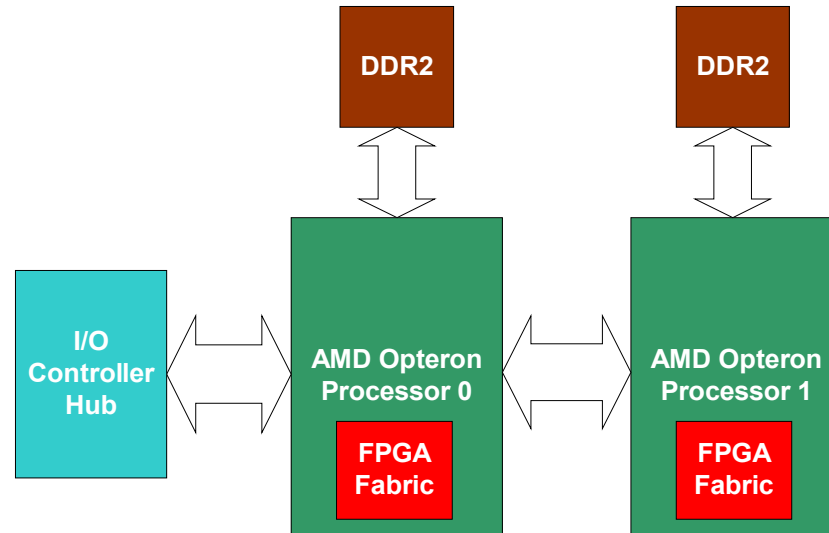
Integration Option: Memory DIMM Bridge + FPGA Card



***Bandwidth: 7.2 GB/sec aggregate (SRC)**

***Latency: < 500 ns**

Integration Option: Internal Coprocessor



Already some commercial examples, but not x86

- 1) ASIC: Stretch, Tensilica RISC core + FPGA fabric on a chip
- 2) FPGA: Nios II + accelerator, PPC + accelerator, etc.

Bandwidth: Comparable to L1 cache B/W?

Latency: A few ns?

Solution Space Pruning

- Disk-port-attached has low BW
- Network-attached doesn't make sense
 - High latency: ~5 to 10 us for Infiniband
 - **High cost**: enclosure, power supply, cooling, FPGA, memory, network I/F
- Memory port (DIMM) attached
 - Great performance, but...
 - Patented (SRC Computer)
 - **High cost**: 3 boards (at least) + cable + license fee
 - High integration complexity (mechanical, thermal)
 - Reliability?
- Internal FPGA coprocessor
 - **Extremely high cost** (65 nm: masks alone are \$2M)
 - Intel and/or AMD in the future?

Remaining Options: I/O Card and External FPGA Co-processor

PCI Express Card Pros and Cons

- **Advantages**
 - Processor independent
 - Plug and play
 - Widely supported
 - Free to design memory architecture around FPGA

- **Disadvantages**
 - High latency
 - Roughly 25% of board area devoted to power circuitry
 - Space limitations => limited memory capacity
 - Short form factor card
 - 2 SODIMMs?
 - OR solder chips directly on board => not commodity \$ anymore
 - Relatively complex board design
 - PCB design alone = 6 man months?
 - Thickness of board limited by slot => limits the number of PCB layers
 - Thermal solution is **COSTLY AND PAINFUL**
 - Must perform air-flow thermal analysis and testing in every target server platform!
 - Server configuration affects the analysis!
 - Might not fit (blades and some 1U servers have a half-height board constraint)

FPGA External Coprocessor Pros and Cons

- **Advantages**

- Low latency
- Fits virtually everywhere, including blades
- Leverage commodity DIMMs
 - Large capacity
 - Large bandwidth
 - Low cost
- **FPGA power < CPU power: thermal solution guaranteed independent of server or server configuration**
- **Simple board**
 - Fast time to market
 - Low NRE and low risk
- **Low cost: completely dominated by cost of FPGA**

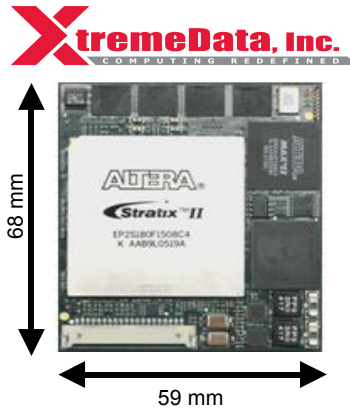
- **Disadvantages**

- **Not plug and play!**
 - BIOS must be modified
 - Open source LinuxBIOS helps
- **Tied to processor socket (AMD or Intel)**
- **Lose a processor chip**

I/O Card vs. FPGA External Coprocessor

	Fictitious PCIe 1.0a x8 short card	In-socket Co-Processor
Bandwidth	3.6 GB/sec	2 HT400 links: 5.2 GB/sec Intel FSB: 10 GB/sec
Latency	1000 ns	300 ns
Memory space	1 to 2 GB	8 to 16 GB
Memory bandwidth (Assuming DDR1)	2x6 GB/s + 2x800 MB/s = 13.6 GB/s	6 GB/s + 800 MB/s = 6.8 GB/sec
Server form factors	4U, 2U, some 1U	4U, 2U, 1U, and most blades
Plug-and-play?	Yes	No
Thermal solution	Heat sink + I/O card bay air flow	CPU heat sink, CPU air flow
Thermal engineering	Complex	Trivial
Power solution	May require supplemental power and customization depending on the motherboard	Trivial: use solution for CPU + memory
Reliability	High	Higher?
Time-to-market	7 months	3 months
NRE cost	~10 man months	~6 man months
BOM Unit cost	\$2,110	\$1,975

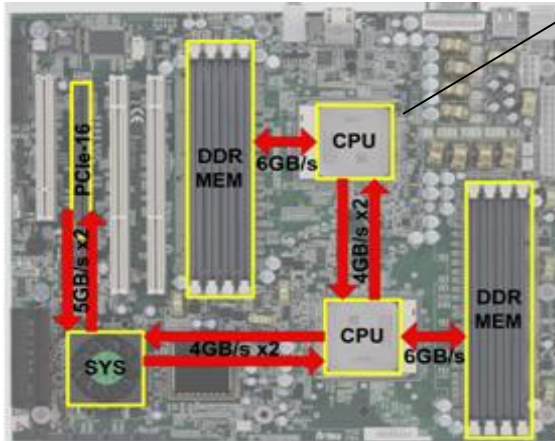
Commercial FPGA Coprocessor Example



- Plugs into AMD Opteron 940 Socket
- Features Altera's largest FPGA, the Stratix II 2s180
- All 3 16-bit HT links available to FPGA
- On-board ZBT SRAM and Flash Memory

XD1000 FPGA Coprocessor Module

Replace one CPU with XD1000

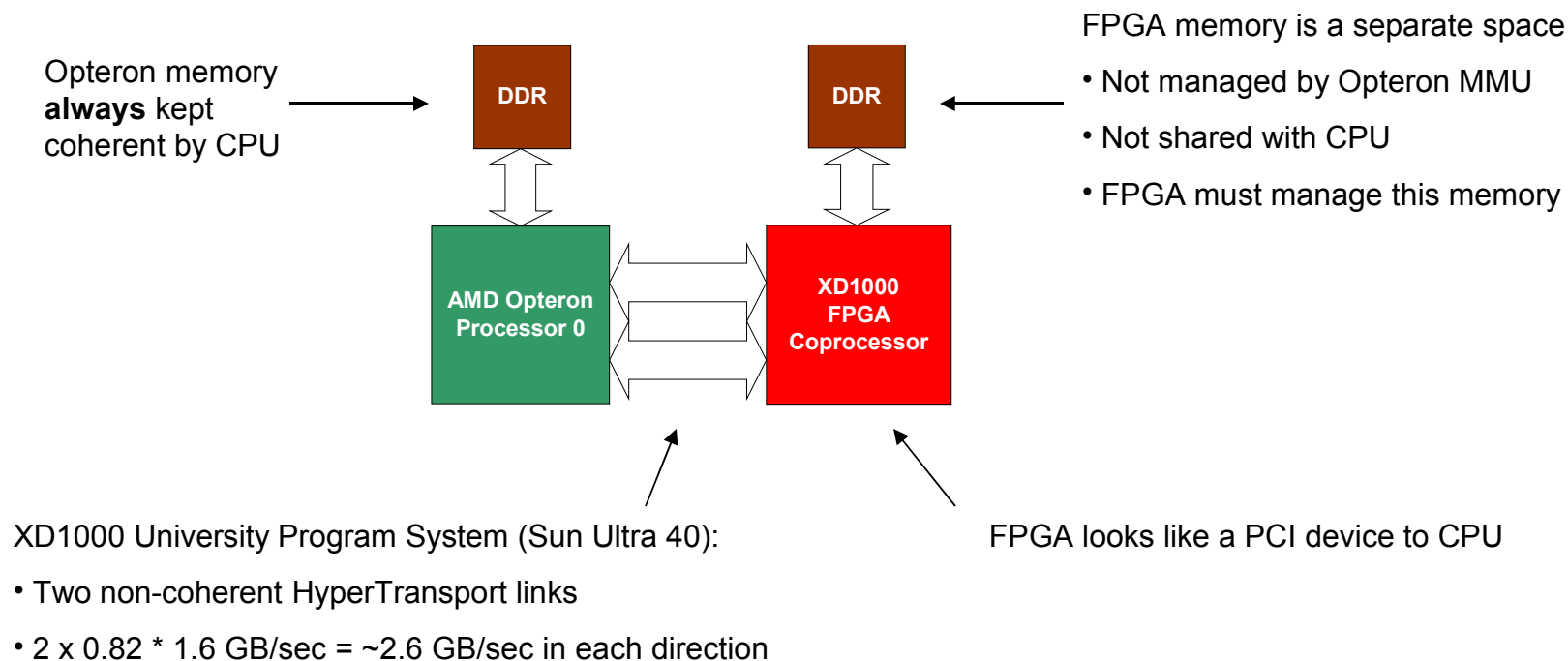


Dual Opteron motherboard

Leverages all resources on the motherboard:

- CPU power supply
- CPU heat sink
- DDR SDRAM (8 GB @ ~5.3 GB/sec)
- CPU <-> FPGA HT links (~1.6 GB/sec each)
- www.XtremeDataInc.com/products.html

XD1000 Coprocessor Memory Model



- XD1000 behaves exactly like a PCI card

XD1000 Coprocessor Communication

- XD1000 Linux driver
 - Map/unmap PCI device BAR into Opteron memory space
 - Enables memory-mapped access of FPGA registers
 - Lock/unlock Opteron memory for transfer
 - Supply user-space buffer pointer, size
 - Driver locks memory and returns list of corresponding physical page addresses (gaping security hole)
 - Enables FPGA DMA to/from Opteron memory
 - Wait on FPGA MSI interrupt
- XD1000 low-level user-space I/O routines
 - Programmed I/O access of FPGA registers
 - Initiate DMA transfer between CPU memory and FPGA memory
 - Register FPGA MSI interrupt service routine

XD1000 FPGA Coprocessor University Program



- Sponsored by AMD, Sun, Altera, and XtremeData
- \$1M program announced July 2006
- XtremeData XD1000 development systems awarded to qualifying university research programs
 - Dual-Opteron PC with integrated XD1000 Coprocessor
 - Reference design
- All 20 systems already awarded ☹
- All parties have strong interest in renewing program for 2007 ☺
- Stay-tuned: www.xtremedatainc.com/UniversityProgram.php

Research Challenges

- Hardware integration is the easy part
- High-level language (HLL) design flow is the tough part
- Goals:
 - Write once, run anywhere (Multi-core CPUs, GPUs, FPGAs)
 - “High level” of abstraction
 - Ease of use
- Challenges:
 - Many fundamental compiler problems
 - Hardware abstraction / FPGA virtualization
 - Programming model
 - Memory model: global vs. distributed?
 - Communication: Message-passing vs. shared memory?
 - Parallelism: Explicit coarse-grain + implicit fine-grain?
 - Concurrency and consistency: locks vs. transactional memory?
- Interesting commercial tool examples:
 - Impulse C (explicit + implicit parallelism model), Mitrion C (SSA)
 - RapidMind, PeakStream (h/w agnostic: GPUs, Multicore CPUs)

Thank you!