Niagara2: A Highly Threaded Server-on-a-Chip

Robert Golla
Principal Architect
Sun Microsystems
Contributors

- Jama Barreh
- Jeff Brooks
- William Bryg
- Bruce Chang
- Robert Golla
- Greg Grohoski
- Rick Hetherington
- Paul Jordan

- Mark Luttrell
- Mark Mcpherson
- Shimon Muller
- Chris Olson
- Bikram Saha
- Manish Shah
- Michael Wong
Agenda

• Chip Overview
• Throughput Computing
• Sparc core
• Crossbar
• L2 cache
• Networking
• PCI-Express
• Power
• Status
• Summary
Niagara2 Chip Overview

- 8 Sparc cores, 8 threads each
- Shared 4MB L2, 8-banks, 16-way associative
- Four dual-channel FBDIMM memory controllers
- Two 10/1 Gb Enet ports
- One PCI-Express x8 1.0A port
- 342 mm^2 die size in 65 nm
- 711 signal I/O, 1831 total
Niagara2 Chip Overview

- Full 8x9 crossbar switch
- Connects every core to every L2 bank and vice-versa
- Supports 8 byte writes from a core to a bank
- Supports 16 byte reads from a bank to core
- One port for core to read/write IO
- System interface unit connects networking and IO to memory
For a single thread
- Memory is THE bottleneck to improving performance
  - Commercial server workloads exhibit poor memory locality
- Only a modest throughput speedup is possible by reducing compute time
- Conventional single-thread processors optimized for ILP have low utilizations

With many threads
- It’s possible to find something to execute every cycle
- Significant throughput speedups are possible
- Processor utilization is much higher
Engineering Solutions

• Design Problem
  > Double UltraSparc T1's throughput and throughput/watt
  > Improve UltraSparc T1's FP single-thread and throughput performance
  > Minimize required area for these improvements

• Considered doubling number of UltraSparc T1 cores
  > 16 cores of 4 threads each
  > Takes too much die area
    > No area left for improving FP performance
Engineering Solutions

- Probabilistic Modelling
  - Generate synthetic traces for each thread with an instruction/miss profile that matches TPC-C
  - Schedule ready threads to run on some number of execution units
  - End simulation once simulated distributions are close to actual distributions

- Works very well for simple scalar cores running lots of threads on transactional workloads
  - Within 10 percent of a detailed cycle accurate simulator
  - Detailed cycle accurate simulator not available at beginning of the project
Engineering Solutions

• Decided to increase the number of threads per core and increase execution bandwidth
  > 8 threads per core x 8 cores = 64 threads total
  > 2 EXUs per core
  > More than doubles UltraSparc T1’s throughput
  > Doubling threads is more area efficient than doubling cores
    > Integrate FGU into core pipeline
      – 6 cycle FP latency
      – Threads running FP are non-blocking
    > Enhance Niagara2’s cryptography
      – Added more ciphers
      – Enhanced existing public key support
Throughput Changes

- Niagara2 throughput changes vs. UltraSparc T1
  > Add instruction buffers after L1 instruction cache for each thread
  > Add new pipe stage “pick”
    > Choose 2 threads out of 8 to execute each cycle
  > Increase execution units from 1 to 2
  > Increase set associativity of L1 instruction cache to 8
  > Increase size of fully associative DTLB from 64 to 128 entries
  > Increase L2 banks from 4 to 8
    > 15 percent performance loss with only 4 banks and 64 threads
  > Increase threads from 4 to 8
Sparc Core Block Diagram

- IFU – Instruction Fetch Unit
  - 16 KB I$, 32B lines, 8-way SA
  - 64-entry fully-associative ITLB
- EXU0/1 – Integer Execution Units
  - 4 threads share each unit
  - 8 register windows/thread
  - 160 IRF entries/thread
- LSU – Load/Store Unit
  - 8 threads share LSU
  - 8KB D$, 16B lines, 4-way SA
  - 128-entry fully-associative DTLB
- FGU – Floating-Point/Graphics Unit
  - 8 threads share FGU
  - 32 FRF entries/thread
- SPU – Stream Processing Unit
  - Cryptographic coprocessor
- TLU – Trap Logic Unit
  - Updates machine state, handles exceptions and interrupts
- MMU – Memory Management Unit
  - Hardware tablewalk (HWTW)
  - 8KB, 64KB, 4MB, 256MB pages
Core Pipeline

- 8 stage integer pipeline

- 3-cycle load-use penalty
  - Memory (data translation, access tag/data array)
  - Bypass (late way select, data formatting, data forwarding)

- 12 stage floating-point pipeline

- 6-cycle latency for dependent FP ops
- Longer pipeline for divide/sqrt
Integer/LSU Pipeline

- Instruction cache is shared by all 8 threads
  - Least-recently-fetched algorithm used to select next thread to fetch
  - Each thread is written into thread-specific instruction buffer
    - Decouples fetch from pick
  - Each thread statically assigned to one of 2 thread groups
  - Pick chooses 1 ready thread each cycle within each thread group
    - Picking within each thread group is independent of the other
    - Least-recently-picked algorithm used to select next thread to execute
  - Decode resolves resource hazards not handled during pick
Threads are interleaved between pipeline stages with very few restrictions
- Any thread can be at fetch or cache stage
- Threads are split into 2 thread groups before pick stage

Load/store and floating-point units are shared between all 8 threads
- Up to 1 thread from either thread group can be scheduled on a shared unit
Stream Processing Unit

- Cryptographic coprocessor
  - One per core
  - Runs in parallel w/core at same frequency
- Two independent sub-units
  - Modular Arithmetic Unit
    - RSA, binary and integer polynomial elliptic curve (ECC)
    - Shares FGU multiplier
    - Cipher/Hash Unit
      - RC4, DES/3DES, AES-128/192/256
      - MD5, SHA-1, SHA-256
      - Designed to achieve wire-speed on both 10Gb Ethernet ports
        - Facilitates wire-speed encryption and decryption
- DMA engine shares core’s crossbar port
Crossbar

- Connects 8 cores to 8 L2 Banks and I/O
- Non-blocking, pipelined switch
- 8 load/store requests and 8 data returns can be done at the same time
- Divided into 2 parts
  - PCX – processor to cache
  - CPX – cache to processor
- Arbitration for a target is required
- Priority given to oldest requestor to maintain fairness and order
- Three cycle arbitration protocol
  - Request, arbitrate and then grant
L2 Cache

- 4 MB L2 cache
- 16 way set associative
- 8 L2 banks
- 64 byte line size
- L2 cache is write-back, write-allocate
  - L1 data cache is write-thru
- Support for partial stores
- Coherency is managed by the L2 cache
  - Directories maintained for all 16 L1 caches
- Data transfers between the L2 and a core are done in 16 byte packets
Integrated Networking

- Integrate networking for better overall performance
  - All network data is sourced from and destined to main memory
- Integration minimizes impact of memory
  - Get networking closer to memory to reduce latency
  - Able to take full advantage of higher memory bandwidth
- Eliminates inherent inefficiencies of I/O protocol translation

FBDIMMs

42 GB/s read
21 GB/s write

Pipelined memory accesses tolerate relaxed ordering

L2 L2 L2 L2 L2 L2 L2

Crossbar

C0 C1 C2 C3 C4 C5 C6 C7

NIU (Ethernet)

System Interface Unit

PCI-Ex

10 GE Ethernet

X8 @ 2.5 GHz
Networking Features

- Line Rate Packet Classification (~30M pkt/s)
  > Based on Layer 1/2/3/4 of the protocol stack
- Multiple DMA Engines
  > Matches DMAs to threads
    > Binding flexibility between DMAs and ports
  > 16 transmit + 16 receive DMA channels
- Virtualization Support
  > Supports up to 8 partitions
  > Interrupts may be bound to different hardware threads
- Dual Ethernet ports
  > 2 dual-speed MACs (10G/1G) with integrated serdes
PCI-Express operates at 2.5 Gb/s per lane per direction

- Point-to-point, dual-simplex chip interconnect
- Transfers are in packets with headers and max data payloads from 128B to 512B
- IOMMU supports I/O virtualization and process device isolation by using PCIE’s BDF#

- MSI Support
  - Event queue accumulates MSIs
  - Allows many MSIs to be serviced upon an interrupt

- Total I/O bandwidth is 3-4 GB/s with max payload sizes of 128B to 512B
Power Management

- Limit speculation
  - Sequential prefetch of instruction cache lines
  - Predict conditional branches as not-taken
  - Predict loads hit in the data cache
  - Hardware tablewalk search control

- Extensive clock gating
  - Datapath
  - Control blocks
  - Arrays

- Power throttling
  - 3 external power throttle pins
  - Inject stall cycles into the decode stage based on state of these pins
    - If power_throttle_pins[2:0] == n then n stalls in window of 8, n is 0-7
    - Affects all threads
Niagara2 System Status

- First silicon arrived at the end of May
- Booted Solaris in 5 days
- Current systems are fully operational
- Expect systems to ship in 2H2007
Summary

• Niagara2 combines all major server functions on one chip
  > Integrated networking
  > Integrated PCI-Express
  > Embedded wire-speed cryptography
• Niagara2 has improved performance vs. UltraSparc T1
  > Better integer throughput and throughput/watt (>2x)
  > Improved integer single-thread performance (>1.4x)
  > Better floating-point throughput (>10x)
  > Better floating-point single-thread performance (>5x)
• Enables new generation of power-efficient, fully-secure datacenters
Thank you ...
robert.golla@sun.com