Dependable Multiprocessing with the Cell Broadband Engine

Dr. David Bueno - Honeywell Space Electronic Systems, Clearwater, FL
Dr. Matt Clark - Honeywell Space Electronic Systems, Clearwater, FL
Dr. John R. Samson, Jr. - Honeywell Space Electronic Systems, Clearwater, FL
Adam Jacobs - University of Florida, Gainesville, FL

HPEC 2007 Workshop
September 20, 2007
Dependable Multiprocessor Technology

- Desire - -> ‘Fly high performance COTS multiprocessors in space’
  - To satisfy the long-held desire to put the power of today’s PCs and supercomputers in space, three key issues, SEUs, cooling, & power efficiency, need to be overcome

  - Single Event Upset (SEU): Radiation induces transient faults in COTS hardware causing erratic performance and confusing COTS software
    - **DM Solution**
      - robust control of cluster
      - enhanced, SW-based, SEU-tolerance

  - Cooling: Air flow is generally used to cool high performance COTS multiprocessors, but there is no air in space
    - **DM Solution**
      - tapped the airborne-conductively-cooled market

  - Power Efficiency: COTS only employs power efficiency for compact mobile computing, not for scalable multiprocessing
    - **DM Solution**
      - tapped the high performance density mobile market

This work extends DM to the Cell Broadband Engine and PowerPC 970FX cluster in Honeywell’s Payload Processing Lab
DM Technology Advance: Overview

- A high-performance, COTS-based, fault tolerant cluster onboard processing system that can operate in a natural space radiation environment

**NASA Level 1 Requirements (Minimum)**

- high throughput, low power, scalable, & fully programmable >300 MOPS/watt (>100)
- high system availability > 0.995 (>0.95)
- high system reliability for timely and correct delivery of data >0.995 (>0.95)
- technology independent system software that manages cluster of high performance COTS processing elements
- technology independent system software that enhances radiation upset tolerance

---

**Benefits to future users if DM experiment is successful:**

- 10X – 100X more delivered computational throughput in space than currently available
- enables heretofore unrealizable levels of science data and autonomy processing
- faster, more efficient applications software development
  - robust, COTS-derived, fault tolerant cluster processing
  - port applications directly from laboratory to space environment
    - MPI-based middleware
    - compatible with standard cluster processing application software including existing parallel processing libraries
- minimizes non-recurring development time and cost for future missions
- highly efficient, flexible, and portable SW fault tolerant approach applicable to space and other harsh environments
- DM technology directly portable to future advances in hardware and software technology
Cell Broadband Engine (CBE) Processor Overview

- Next-generation, high-performance, heterogeneous processor from Sony, Toshiba, and IBM
- 3.2 GHz, 64-bit multi-core processor
- ~200 GFLOPS peak (single precision)
- 64-bit Power Arch.-compliant PPE
  - Power Processing Element
- 8 128-bit SIMD SPEs
  - Synergistic Processing Elements
- Elements connected via 200+ GB/s EIB
  - Element Interconnect Bus
- 90 and 65nm SOI versions available
- Version with DP SPEs has been announced

- Why Cell?
  - Demonstrates the portability of DM to a modern HPC platform
  - One of the first commercially available, multi-core architectures
  - Provides a vehicle for exploration of next generation architectures
  - Allows exploration of software development considerations for multi-core architectures
  - Sony Playstation3 with Linux and IBM Cell SDK 2.1 provides a powerful, cost-effective platform for product evaluation
    - Key limitations- 256 MB RAM, 6 SPEs rather than 8
  - Expect lower-power versions to emerge for embedded applications
Honeywell CPDS/970FX Cluster

- Four dual-processor SMP PowerPC970 “Jedi” systems
  - 2.0 GHz, 1 GB RAM, Gigabit Ethernet
  - Debian GNU/Linux 4.0

- Four 7-core (PPE + 6 SPE) PS3 “Cell Processor Development Systems” (CPDS)
  - 3.2 GHz, 256 MB RAM, Gigabit Ethernet
  - Fedora Core 6 Linux

- Key benefits of PS3:
  - Performance can approach HPC Cell hardware at fraction of cost

- Key limitations of PS3:
  - 256 MB RAM
  - 6 SPEs instead of 8
  - Gigabit Ethernet
  - Slow hard disk subsystem
DMM Mapping to CPDS/970FX Cluster

DMM – Dependable Multiprocessor Middleware

Scientific Application

Application Programming Interface (API)

Data Processor (Cell or 970FX)

System Controller (970FX)

DMM

OS – WindRiver VxWorks 5.4

Hardware Honeywell RHSBC

Gigabit Ethernet

S/C Interface SW and SOH And Exp. Data Collection

Policies Configuration Parameters

Mission Specific Applications

OS – Fedora Core 6 Linux (Cell)
OS – Debian/GNU Linux 4.0 (970)

Cell Processor PPE or PPC970FX

Cell SPEs

DMM components and agents

Application Specific

Generic Fault Tolerant Framework

Application

Mission Specific Applications

OS/Hardware Specific

SAL
(System Abstraction Layer)

...
CPDS/970FX Cluster DM Configuration

- System Controller node mimics functionality of rad hard SBC in flight system
- Data Processors are heterogeneous mix of 970FX and CPDS
- DMM runs on Cell PPE, doesn’t need to know about Cell SPEs
  - Perfect fit for Cell/PPE, since PPE typically dedicated to management tasks, and usually has compute cycles to spare for tasks related to DMM

(SC)=System Controller  (DS)=Data Store  (DP)=Data Processor
SAR Benchmark on Single Cell BE

- Modified version of University of Florida Synthetic Aperture Radar benchmark to support accelerated processing on Cell
  - IBM Cell SDK 2.1, libspe2
  - No assembly-level performance tuning performed, minimal optimizations such as SPE loop unrolling and branch hinting performed in some instances

- As expected, PPE-only performance of non-accelerated code is much slower than modern Intel processor
  - PPE’s main role in Cell is a management processor, despite its high 3.2 GHz clock speed

- Accelerated version with SPEs achieves 38x speedup over PPE-only version, 10x speedup over Core 2 Duo
  - Range Compression stage exhibited 40x speedup on Cell vs. Core 2 Duo
    - Utilizes optimized IBM FFT libraries

- Relatively linear speedup indicates algorithm is scalable to high-end Cell hardware with 8 or more SPEs

Near Linear Speedup as Number of Active SPEs Increased
SAR Benchmark on Cell Cluster

- Followed with modifications to support MPI parallel processing of patches of a SAR image across multiple Cell-accelerated systems
  - Using Open MPI 1.2.3, supports heterogeneous clusters transparently
- Single 970FX node serves as master, reads patches from file, provides patches to CPDS nodes for processing via MPI, receives processed patches via MPI, writes to file
  - Results include disk I/O time
- Using 970FX as data source mitigates effects of slow PS3 disk access by taking it out of the equation to get a more accurate picture of Cell performance capabilities
  - Master-worker 970FX/single-CPDS combo outperforms single CPDS even though data has to travel over Gigabit Ethernet!
- Scalability of approach limited by Gigabit Ethernet network on PS3 (not a Cell limitation), with excellent speedup obtained at 2 Cell processors but diminishing returns beyond
  - Network connectivity of PS3 is out of balance with theoretical peak performance capability of each node [1]
- Also performed experiments with Core 2 Duo x86-based data source
  - However, network performance greatly suffered—suspect swapping of bytes for endian conversion impacted Cell PPE more significantly than other systems
  - May be a configuration issue

General Cell Development Insights

• Some of these findings have also been documented in the literature, but are worth re-emphasizing as we found them to be very relevant to our work

• PS3 memory limitation of 256MB is a practical constraint on some applications, but is okay for the purposes of technology evaluation

• Impressive speedups possible with relatively little development effort
  - But, need to leverage existing optimized libraries or heavily hand optimize code to really reap the benefits of the architecture [2]
  - SPE programming bugs can be hard to diagnose without appropriate tools
  - SPE won’t let you know if you’ve run out of memory
  - Code can be overwritten with data, etc.
  - Simulator/debugger should be helpful in these cases

Conclusions and Future Work

- DM provides a low-overhead approach for increasing availability and reliability of COTS hardware in space
  - DM easily portable to any Linux-based platform, even on an exotic architecture such as Cell
  - DM well-suited to Cell PPE, which is used primarily as a management processor for most Cell applications
  - Future Cell platforms expected to improve power consumption and will be aided by advances in cooling technology

- Cell provided impressive overall speedups in UF SAR application with low development effort
  - But, much higher speedups for sections of code that primarily leverage existing optimized libraries

- Future Work
  - Complete benchmarking of Cell BE and DM middleware
    - MPI benchmarking, SAR benchmarking, overhead comparison, reliability/availability benchmarking
    - Updates to be included in poster presented at HPEC 2007
  - Augment DM to provide enhanced, Cell-specific functionality
    - Spatial replication across SPEs