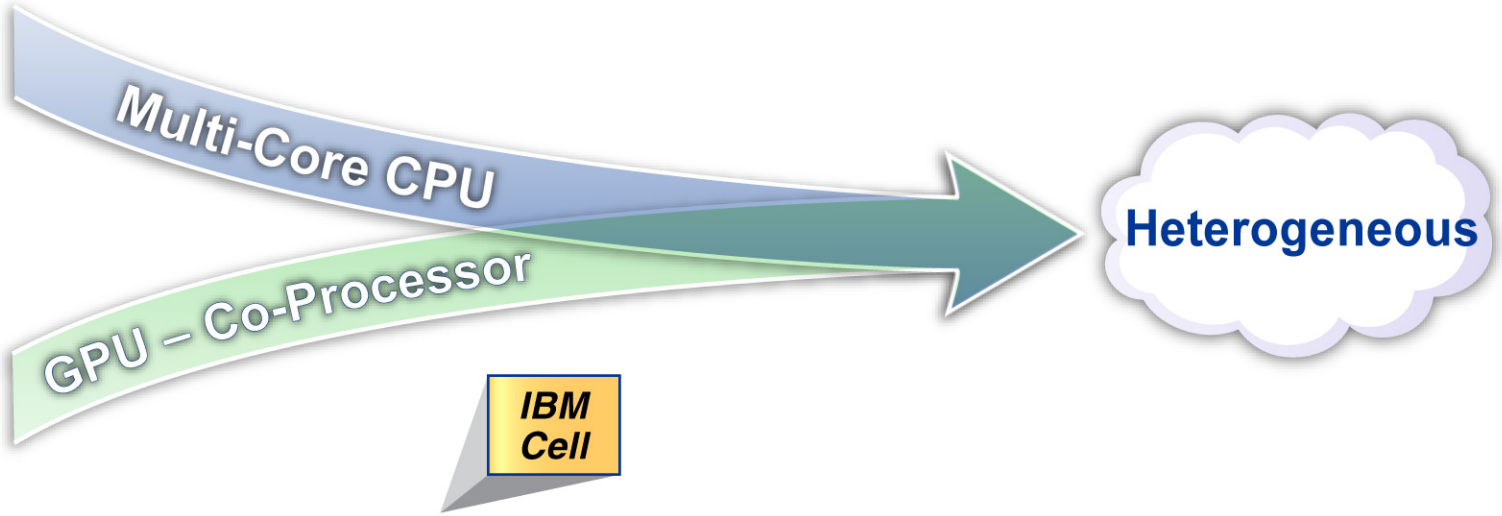
GENERAL DYNAMICS
Advanced Information Systems

Implementation of Parallel Processing Techniques on Graphical Processing Units

Brad Baker, Wayne Haney, Dr. Charles Choi

Industry Direction

- High performance COTS computing is moving to multi-core and heterogeneous silicon



MULTI-CORE CPU	CURRENT	FUTURE
▪ Multi-core CPU with smaller individual cores ▪ GPU co-processor	▪ Multi-core CPU with 1-3 GPU co-processors ▪ Heterogeneous multi-core (IBM Cell)	▪ Smaller, heterogeneous cores on same silicon

Objectives – Methods

- Investigate challenges associated with supporting signal processing applications on GPUs
  - Identify cost-effective alternative approaches
- Trade studies performed
  - Multi-core, Cell, and GPU Hardware
    - GFLOP / Watt
    - GFLOP / $
  - GPU (Stream) Software
    - Stream processing allows easier parallelization using GPU hardware

Stream Processing

CPU/GPU Integration
GPU Used as a Math Co-Processor

Trade Study - Hardware

- Multi-Core CPU
  - 2 x 265HE 1.8Ghz AMD Opterons
- GPU
  - Assumes single GPU on defined hardware baseline*
- Cell system
  - Single CAB from Mercury Systems on defined hardware baseline**

Evaluation Criteria	Metric			Units
	GPU *	Cell **	CPU	
Price (Complete System)	$7500	$14500	$6500	$
Power Consumption	395	435	225	Watts
Memory Capacity	0.768	5	16	Gb
Cache Memory	0.1	0.512	2	Mb
Memory Bandwidth	86.4	22.4	6.4	Gb/s
Platform Maturity	0.5	1	3	years
Software Composite Score	2.5	4	8	Subj.
Theoretical Performance	554	215	36	Gflop/s

8800GTX Gflop Calculation: MADD (2 FLOPs) + MUL (1 FLOP)) × 1350MHz × 128 SPs □ 518.4 GFlop/s

Hardware Baseline	Power (watts)	Gflop/s (theor)	Gflop/s (obs)	Cost	Size in U
AMD Opteron 265HE – 1.8 Ghz	225	36	32	$ 6,500	1

Trade Study - Hardware

- Score weighting system based on theoretical performance with one subjective score
  - 1-10 performance scale rates each category as a percentage of the maximum performance
  - Ratio scale relates specific performance scores to the highest performance to highlight the large differences between platforms
- Scenario titles indicate weighting of specific metrics in the comparison

Scenario	1 to 10			Ratio		
	GPU	Cell	CPU	GPU	Cell	CPU
Perf	141.59	52.62	161.00	318.22	89.45	82.60
Perf, $	194.23	52.62	161.00	325.71	89.45	85.27
Perf, Power, $	212.01	61.62	251.00	335.62	98.45	102.67
Perf, Power, $, Software	221.01	92.71	341.00	344.62	112.85	131.47

Trade Study for GPU Platform

- OS used – Centos Linux v4.4

	Math Support	Library Functionality
PeakStream	1D & 2D Arrays, Single and Double Precision, standard C/C++ math library, BLAS, Matrix Solver, Random Number Generators, 2K complex to complex FFTs.	Runtime Virtual Machine that installs on top of OS. gcc 3.4.5, gcc 4.0.3, or Intel compiler 9.0 gdb 6.3
RapidMind	Standard C++ libraries for stream processing types. Matrix support. 2K complex to complex FFTs	Offers a transparency layer on top of the parallel processor platform.
Brook	Standard C Library. FFT and Matrix support.	Brook extends C to include parallel data constructs. Offers a high level language that is platform independent using openGL, DX, or CTM.
CUDA	Standard FFT and BLAS libraries. 1D, 2D, and 3D transforms of complex and real□valued data up to 16k.	C, C++ not fully supported (no classes definitions but supports function templates). Supports thread communication.
CTM	No mathematic libraries provided. Examples provided by the vendor for FFTs and Matrix multiply.	Program in the native instruction set and memory AMD Assembler.

Experiment Description

- Sonar Passive Narrowband Processing:
  - Multiple FFT operations and spectral processing of beamformed data
- Implementation
  - OOP design written in C++
  - 4k complex 1D FFT over 100 beams, 1 aperture
  - Substituted Nvidia's CUDA FFT library routines in place of Intel's MKL FFT routines

Math Benchmark and Signal Processing String Results

Fundamental Math Benchmarks		
Software Platform (GPU)	1k SGEMM Gflops	1k 1d Complex FFT
Peakstream (AMD r520)	80.13	8.7
CUDA (Nvidia g80)	95	43.4
RapidMind (Nvidia g80)	24	7.5
RapidMind (AMD r520)	26	4.9
Intel Core 2 Quad QX6700	12	14.2
AMD Opteron 265HE	8.8	4.8

Nvidia Cuda Platform
- Utilizes most powerful GPU on market
- Most extensive pre-built math library

Approx. 50% Performance Gain Using CUDA's FFT

Application Results	
Architecture	Execution Time
VSIPL++ PNB Algorithm on Intel Core 2 Quad QX6700 CPU	735.78 msec
CUDA Batch Style PNB Algorithm on Nvidia g80 GPU	367.23 msec

Conclusion

- Rudimentary math results on the GPU show improvements over traditional hardware
- Application results impressive with minimal effort
- GPU performance requires multiple math operations in sequence to form a larger kernel with minimal I/O transfers
  - Automatically operates in parallel on large data structures
  - I/O via PCI-E bus is the main bottleneck limiting the GPU
- Multi-Core CPUs perform well on smaller data sets where I/O speed is critical
- Tools needed to alleviate the burden of porting to vendor specific hardware