

Heterogeneous Computing Clusters for HPC

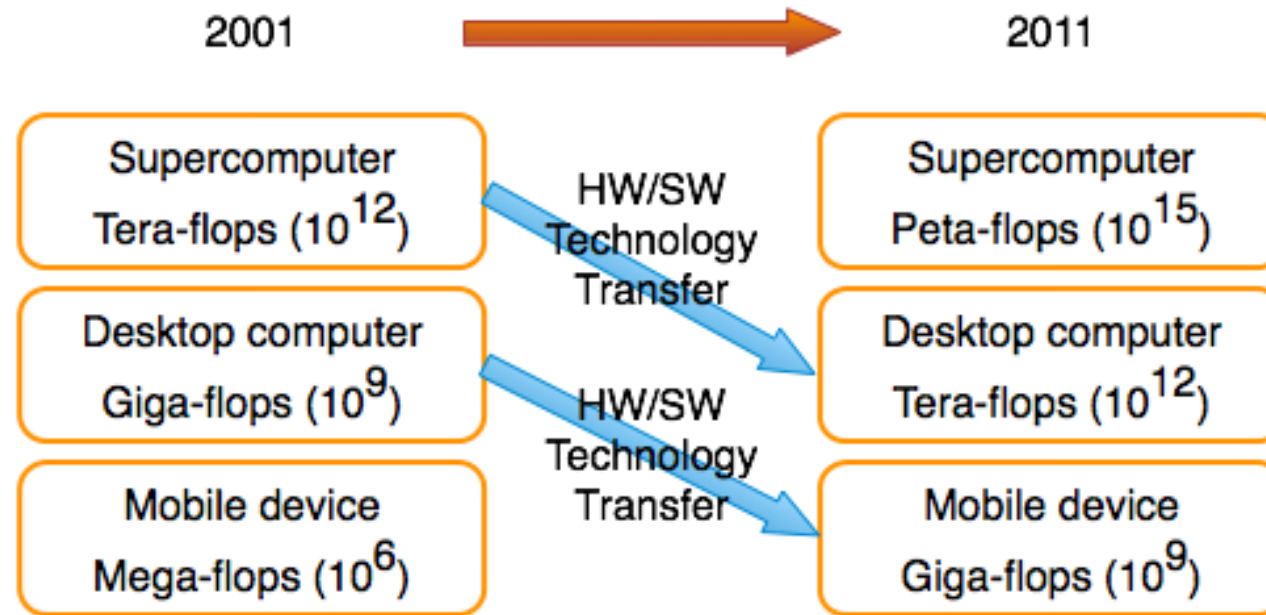
Jaejin Lee

Center for Manycore Programming

Seoul National University

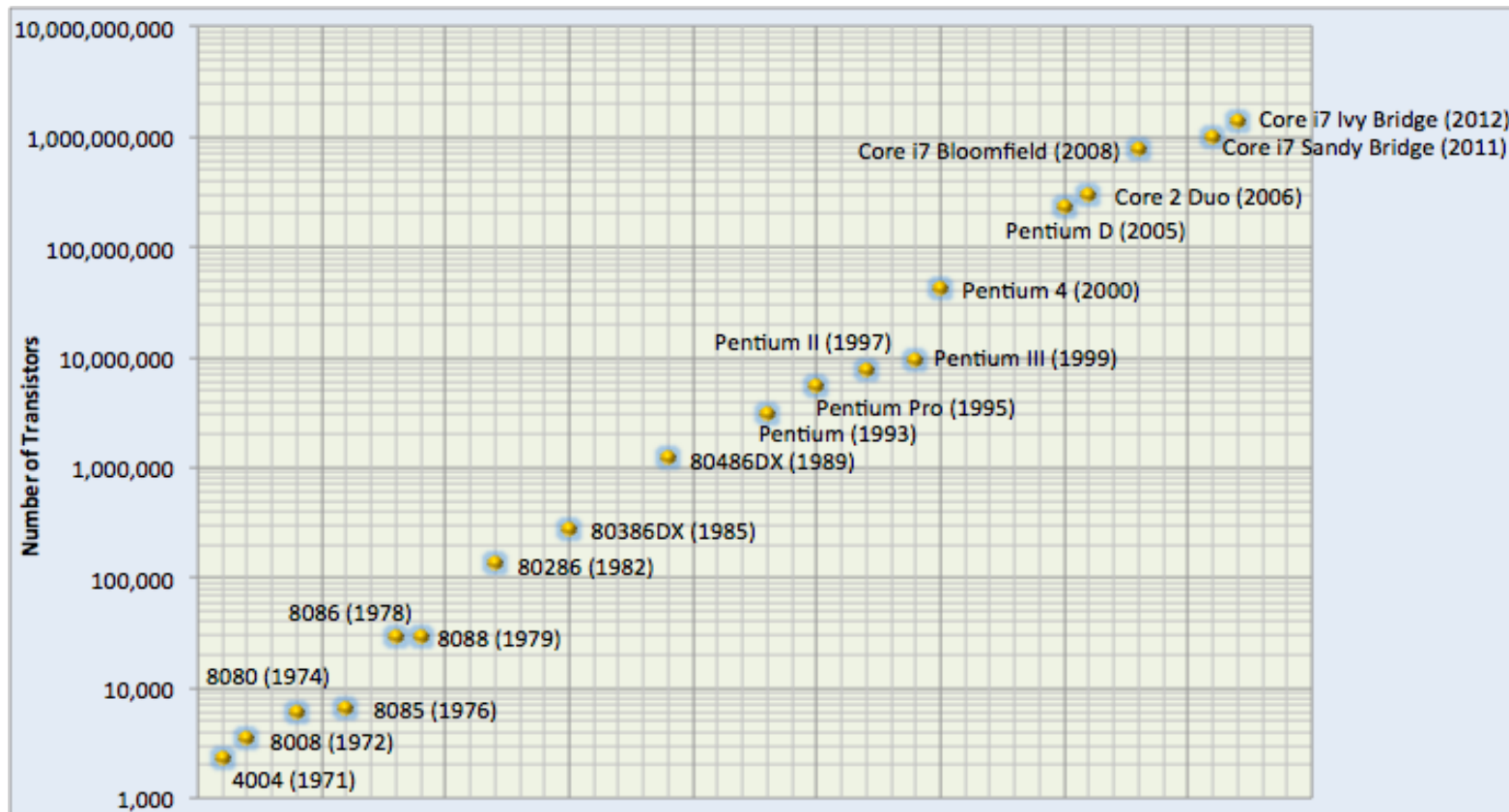
Trends in Performance

- HW/SW technology transfer pattern
 - Duration for technology transfer will be much shorter (3 ~ 5 years)
 - Expecting exa-scale (10^{18} flops) computing by 2020



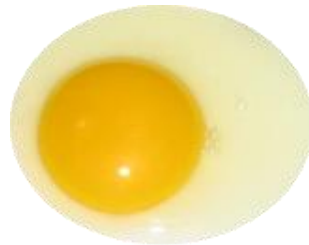
Moore's Law

- The number of transistors on a single die doubles approximately every two years



Power Wall

- CPU's computing power \propto CPU clock frequency
- Power consumption \propto CPU clock frequency
 - Cannot increase the CPU clock frequency indefinitely
 - Frequency increase stopped at 3GHz ~ 4GHz
- Heat dissipation (for servers) \propto power consumption
- Battery life for mobile devices
 - Inversely proportional to power consumption



Issue Width of Superscalar Processors

- The goal of the instruction pipeline
 - To issue an instruction on every clock cycle
- Issuing an instruction
 - The instruction proceeds into the execution unit (in general)
- Issue-width is the maximum number of instructions that can be issued by a processor
 - When the hardware can issue up to n instructions on every cycle:
 - The processor has n issue slots
 - The processor is an n -issue processor

ILP Wall

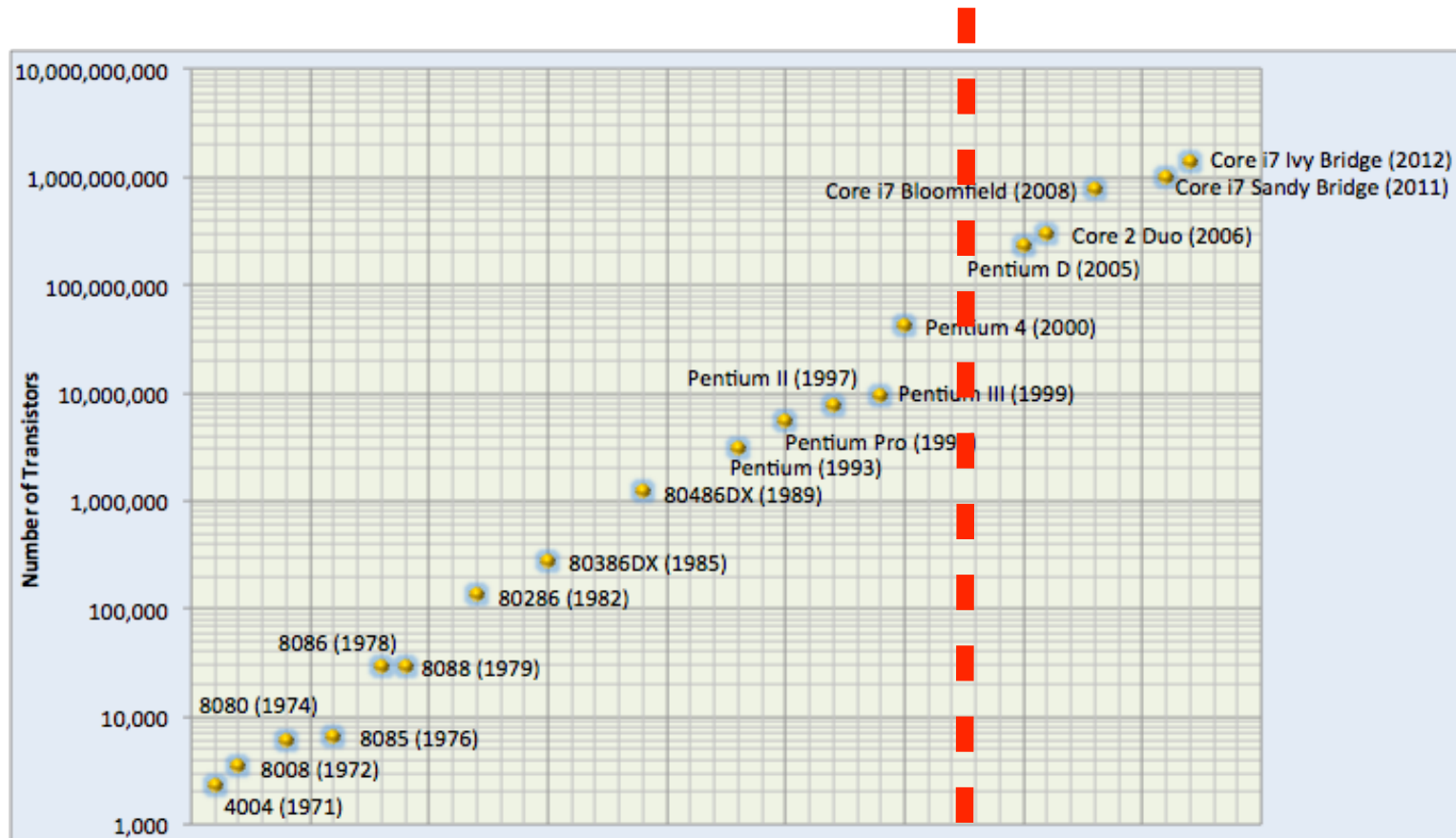
- Instruction Level Parallelism
 - How many instructions can be issued at the same time?
- The lack of ILP in a single thread
 - The ILP in an application is limited
 - Cannot increase the issue width indefinitely
- However, ILP has enabled the rapid increase in processor speed so far

Multicores

- A multicore is a single chip that contains two or more independent processors, called cores
- Manycore
 - A multicore with more than 8 or 16 cores
- A solution to the power wall and ILP wall

New Moore's Law

- The number of cores in a single chip doubles approximately every two years



Mobile Systems vs. General-purpose Systems

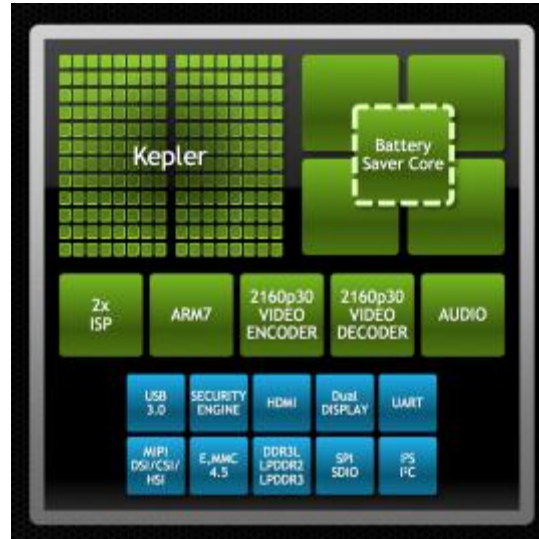
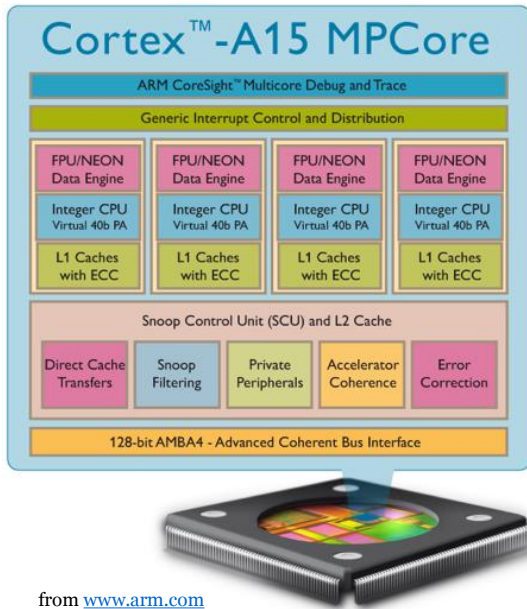
- Different in terms of
 - Compute power
 - Memory size
 - Memory bandwidth
 - Power consumption
 - Physical size
 - Cost
- The principle behind is the same

Demands in Mobile Processing

- Mobile devices will be facing the same level of performance and power demands as that of PCs
 - Web browsing, HD video, 3D gaming, multitasking, etc.

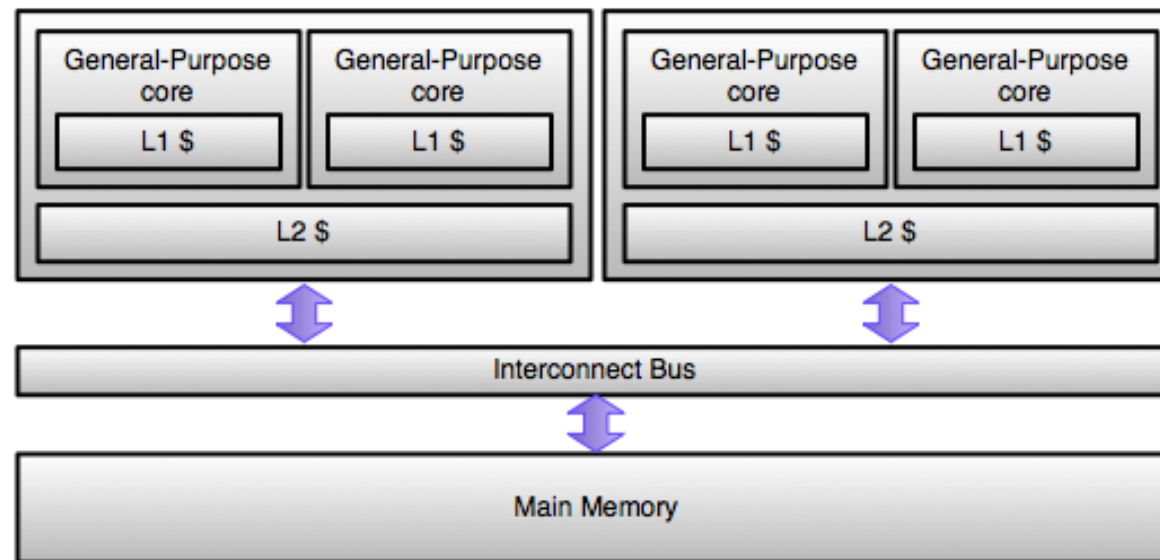
The Era of Multicores

- To overcome the Power Wall and ILP Wall

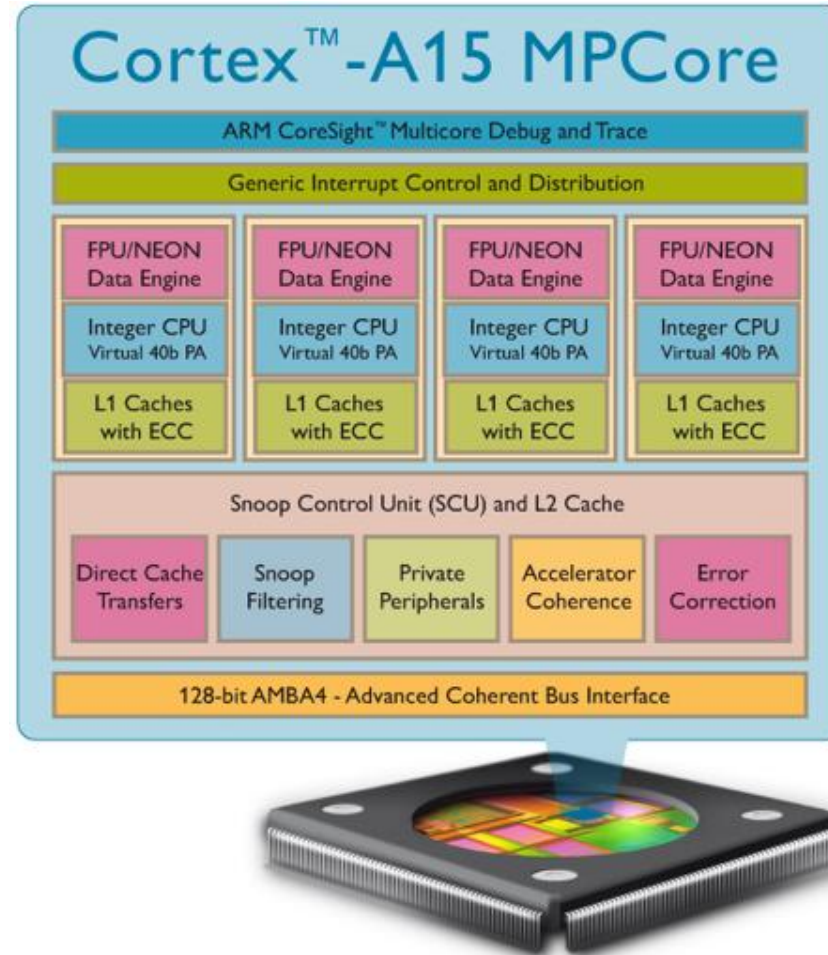


Homogeneous Multicore Architectures

- Multiple homogeneous cores in a single chip
- Intel Xeon, AMD Opteron, ARM Cortex A15 MPCore, IBM Power7, Oracle UltraSPARC T4, etc.



ARM Cortex A15 MPCore



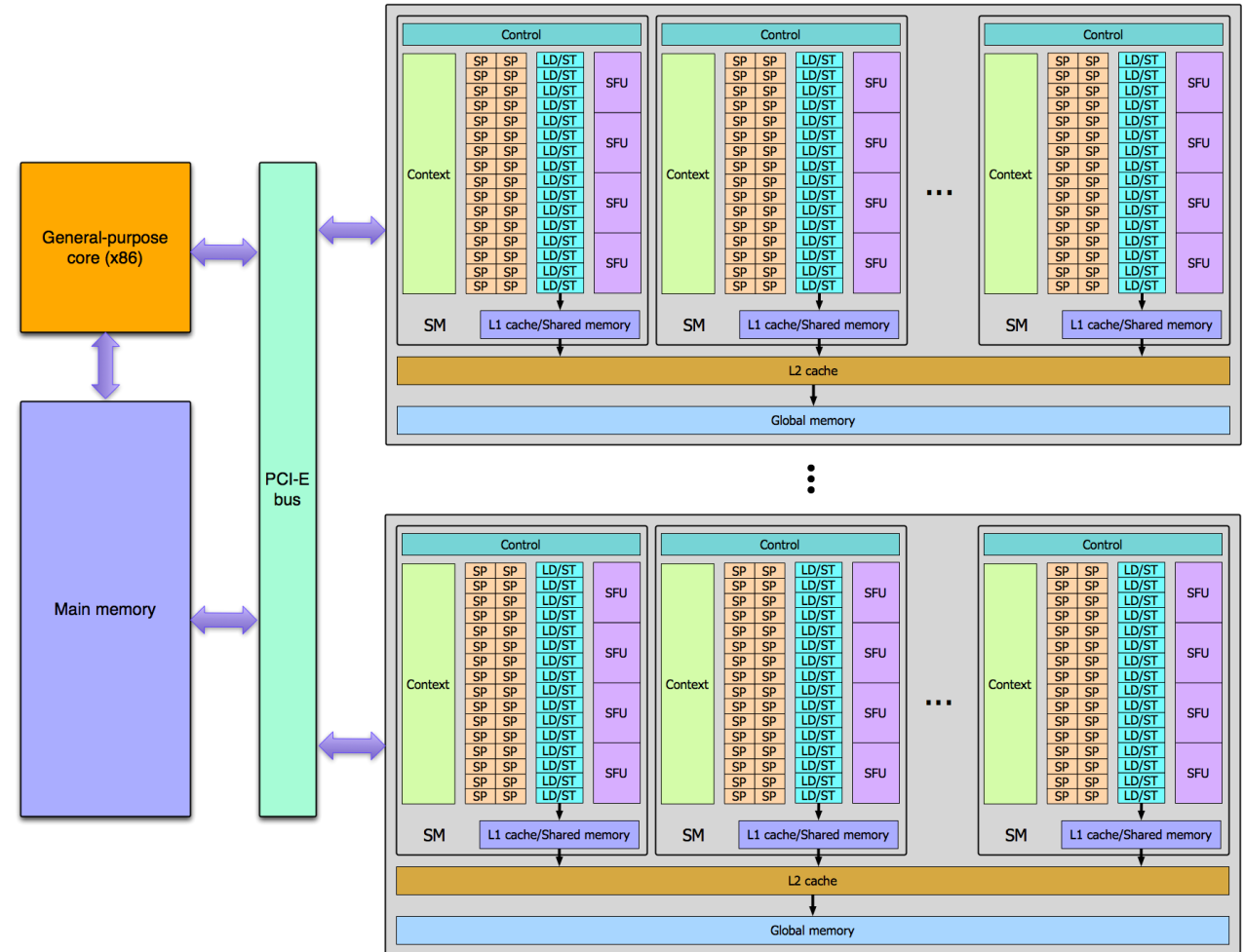
from www.arm.com

Heterogeneous Computing Systems

- Contain different types of processors
 - Processors: CPUs, DSPs, GPUs, FPGAs, or ASICs
 - For extra performance and power efficiency
- General-purpose processors (resource management) + accelerator processors (compute intensive)
- Heterogeneity in
 - ISAs, processing power, power consumption, memory hierarchies, micro-architectures, etc.

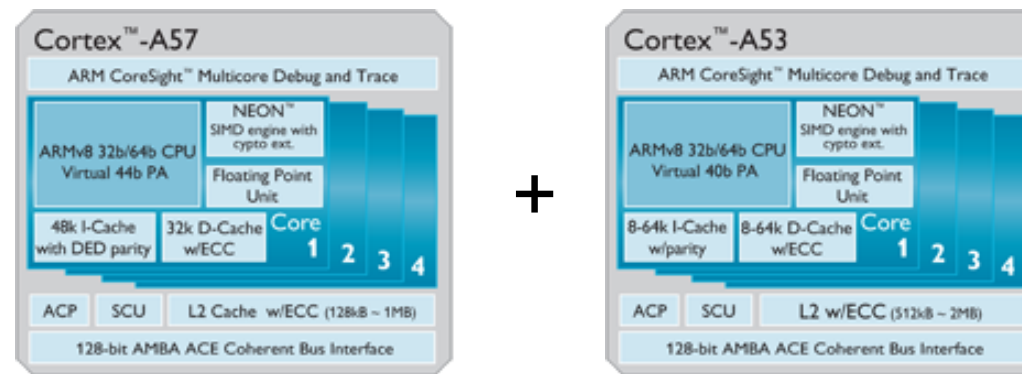
GPGPU

- General-Purpose computing on Graphics Processing Units



Heterogeneous Multicore Architectures

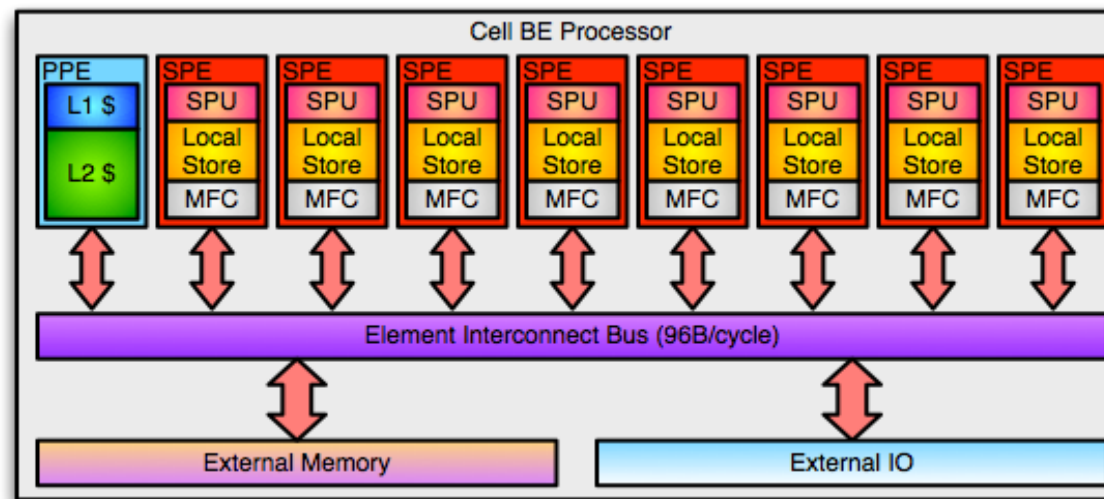
- Asymmetric multiprocessing (ASMP)
- Asymmetric chip-multiprocessor (ACMP)
- AMD fusion, Intel i7, AMD Fusion, IBM Cell BE, TI OMAP, ARM big.LITTLE, Nvidia Tegra, etc.



from www.arm.com

Cell Broadband Engine

- Used in Sony Playstation 3
- IBM Roadrunner supercomputer
 - 12,960 IBM PowerXCell 8i processors + 6,480 AMD Opteron dual core processors
 - The first 1.0 Pflops system
 - Ranked the first in Top500 in June 2008



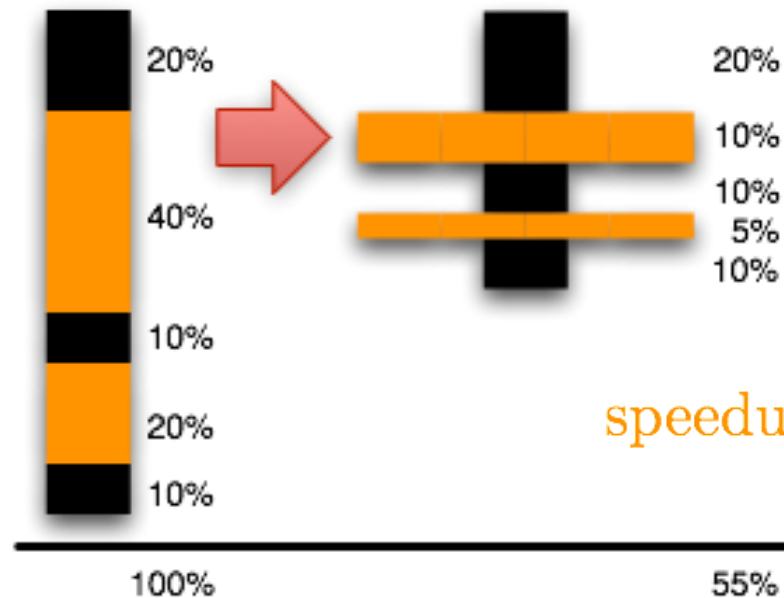
FPGA

- Altera supports OpenCL(not hardware specific) for FPGAs
- FPGA as an accelerator



Amdahl's Law

- p : the proportion of a program that can be parallelized
- $1 - p$: the proportion of a program that cannot be parallelized
- n : the number of processors

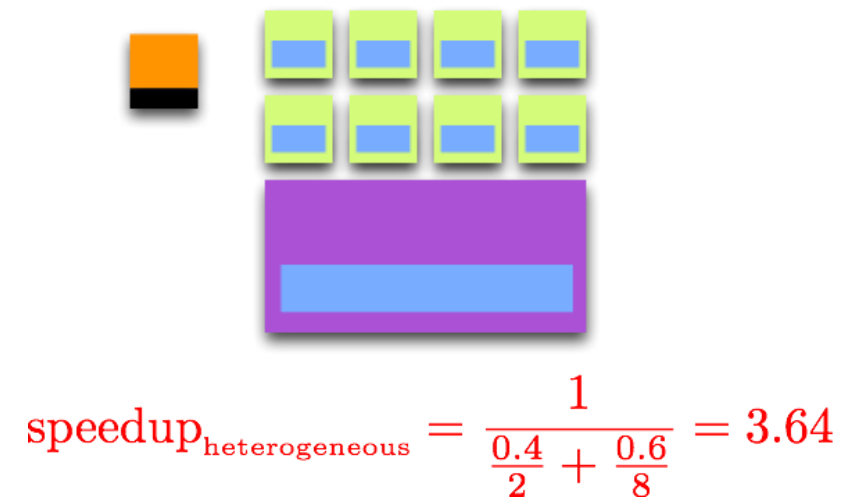
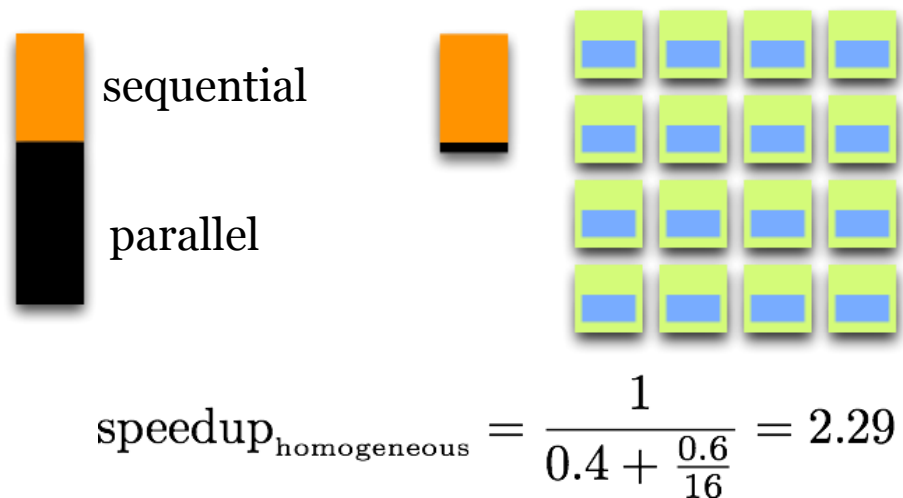


$$\text{speedup} = \frac{1}{(1 - p) + \frac{p}{n}}$$

$$\text{speedup} = \frac{1}{(1 - 0.6) + \frac{0.6}{4}} = \frac{1}{0.55} = 1.82$$

Why Heterogeneous Systems?

- Assume,
 - A sequential code fragment that takes 40% of the sequential execution time can be accelerated by an accelerator core
 - A single large core runs the sequential code twice as fast
 - The rest of the program (60%) can be parallelizable



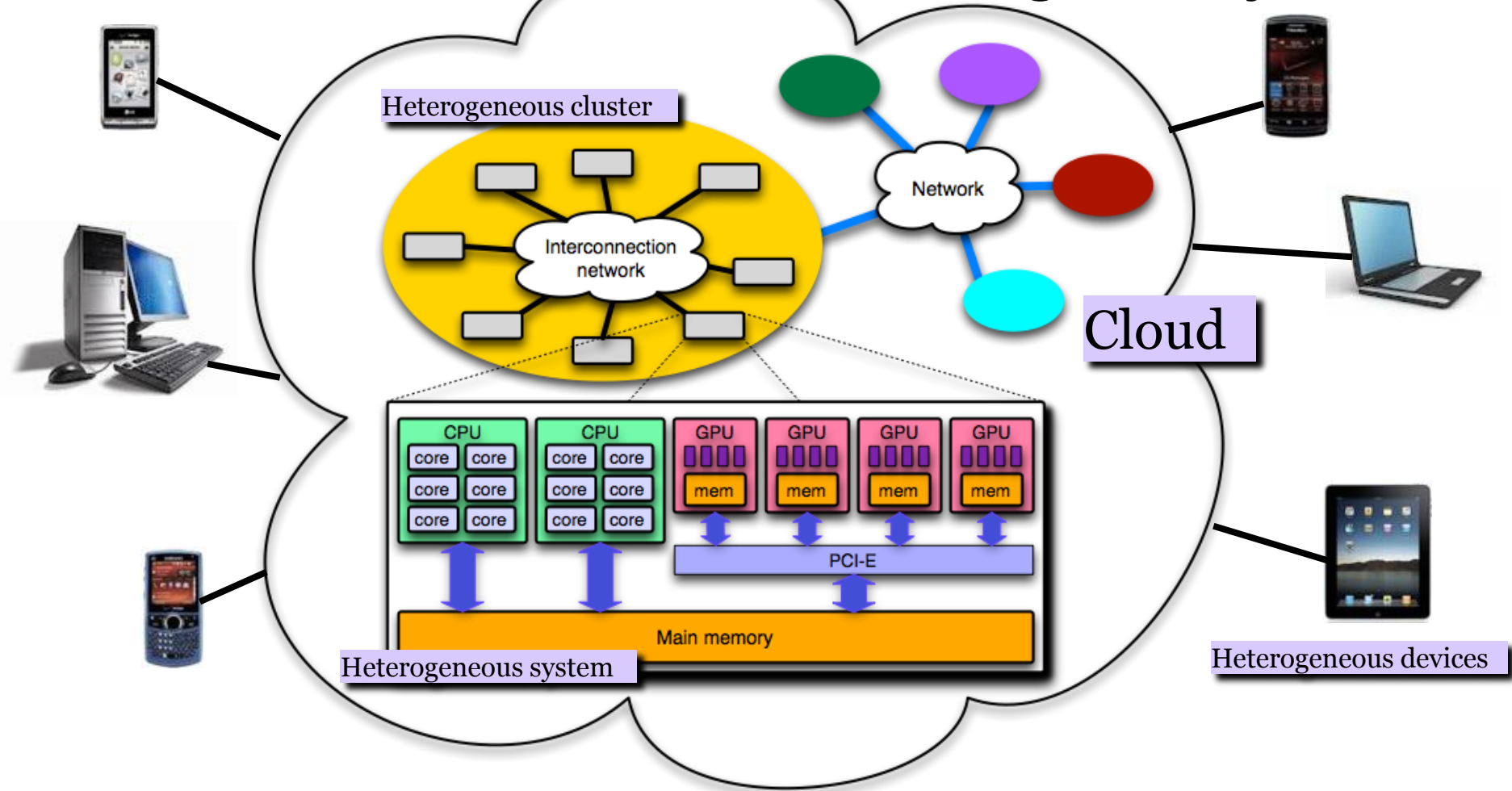
The Trend in TOP500

- The number of heterogeneous supercomputers is increasing

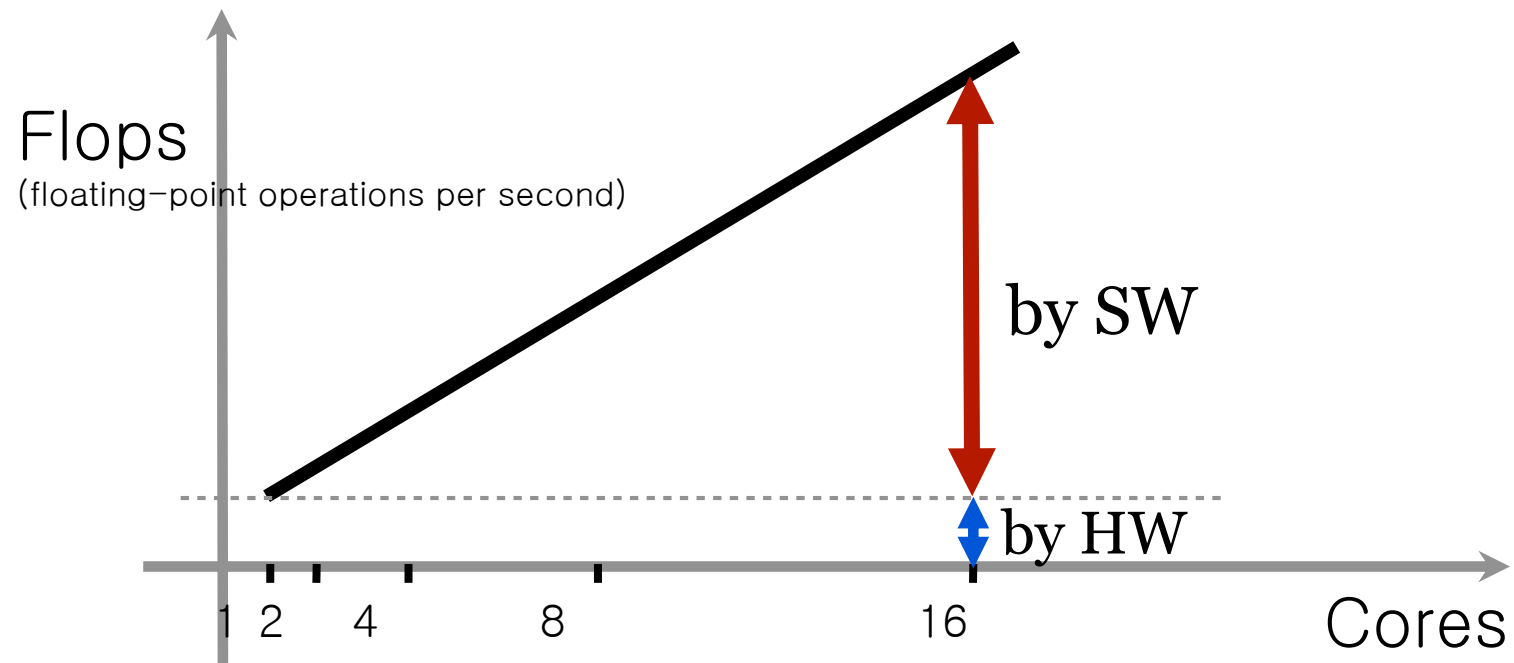
Top500	Jun 2009	Nov 2009	Jun 2010	Nov 2010	Jun 2011	Nov 2011	Jun 2012	Nov 2012	Jun 2013	Nov 2013	Jun 2014
Homogeneous	495	493	491	483	481	461	442	438	446	447	436
Heterogeneous	5	7	9	16	19	39	58	62	54	53	64 (12.8%)

Heterogeneous Parallel Computing

- How to deal with such heterogeneity



Multicore Programming



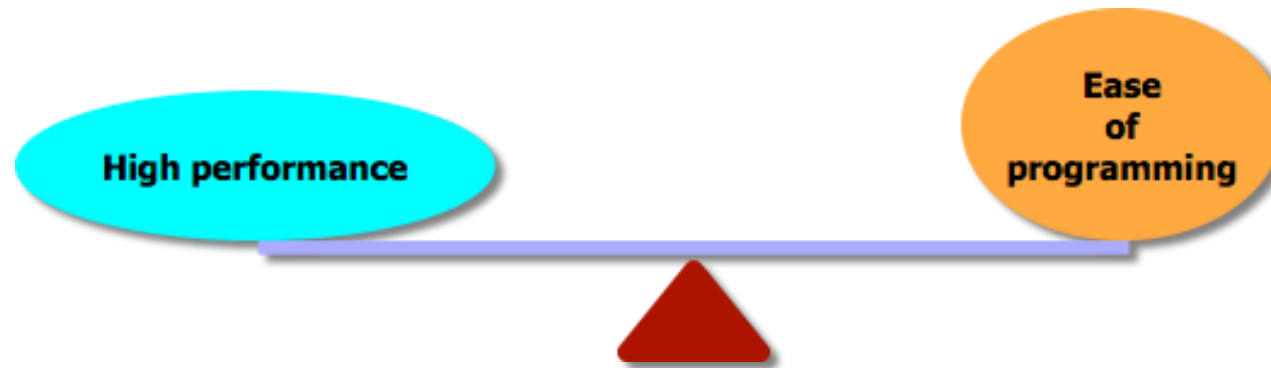
Programming Wall

- How to easily create software that efficiently exploit the parallelism of manycore hardware
- Has not been solved for last 30 years
 - For traditional multiprocessor systems



Parallel Programming Models

- An interface between the programmer and the parallel machine when developing an application
 - Languages, libraries, language extensions, compiler directives, etc.
- Important to have balance between delivering high performance and ease of programming



Parallel Programming Models

- Pthreads (POSIX threads)
- Message Passing Interface (MPI)
- OpenMP
- OpenCL
- SnuCL
- CUDA
- OpenACC
- Cilk
- ...

OpenCL

- Open Computing Language
- A framework (parallel programming model) for heterogeneous parallel computing
 - A language, API, libraries, and a runtime system
 - From mobile devices to supercomputers
 - License free
- The specification of OpenCL 1.0 was released in late 2008
 - Now, OpenCL 2.0, but no implementation available yet
- Portable code across different architectures
 - CPUs, GPUs, Cell BE processors, Xeon Phi, FPGAs etc.
- Based on ANSI/ISO C99 standard
- Supported by many vendors, such as Apple, AMD, ARM, IBM, Intel, NVIDIA, Samsung, TI, Qualcomm, etc.

Conclusions

- Heterogeneous computing will be popular
- Many opportunities
 - R&D activities has begun recently (3~4 years ago)
- Software is very important
- New programming models