

Computer Performance Evaluation and Benchmarking

EE 382M

Dr. Lizy Kurian John

Evolution of Single-Chip Microprocessors

	1970' s	1980' s	1990' s	2010s
Transistor Count	10K- 100K	100K-1M	1M-100M	100M- 10 B
Clock Frequency	0.2- 2MHz	2-20MHz	20M- 1GHz	0.1- 4GHz
Instruction/Cycle	< 0.1	0.1-0.9	0.9- 2.0	1-100
MIPS/MFLOPS	< 0.2	0.2-20	20-2,000	100- 10,000

Hot Chips 2014 (August 2014)



**Applying AMD's "Kaveri" APU for
Heterogeneous Computing**

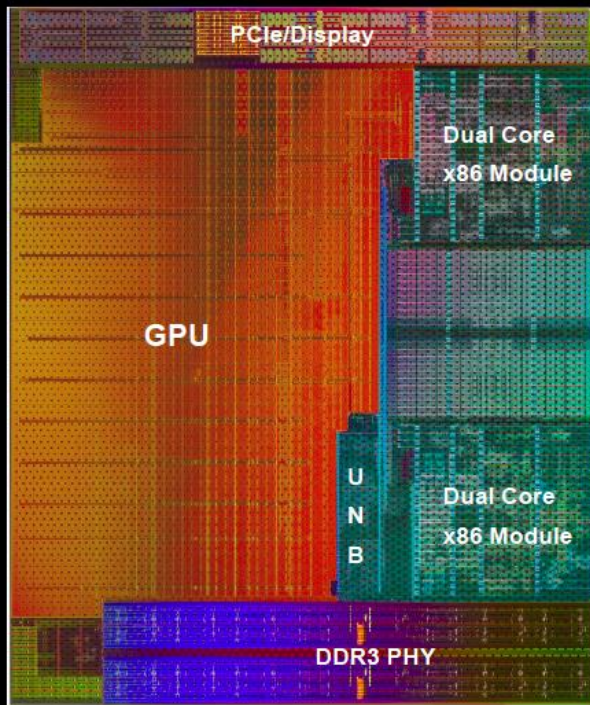
DAN BOUVIER, BEN SANDER
AUGUST 2014

AMD KAVERI HOT CHIPS

2014

“KAVERI”

AMD



Die Size: 245mm²

Transistor count: 2.41 Billion

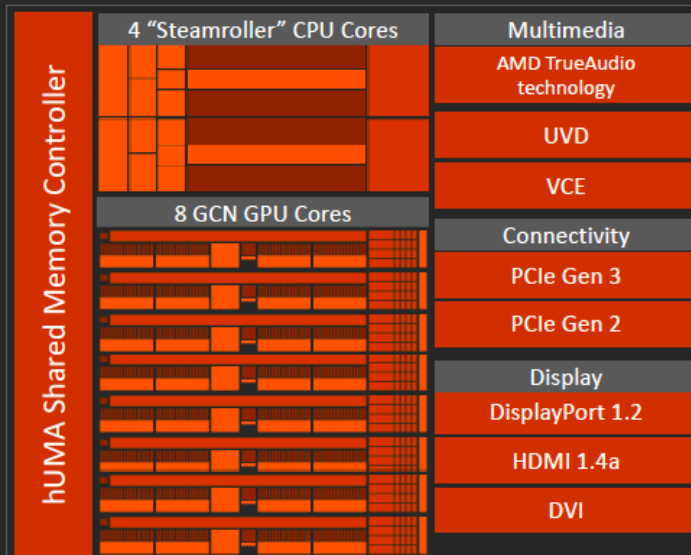
Process: 28nm

AMD KAVERI

A-SERIES REDEFINES COMPUTE



Kaveri



MAXIMUM COMPUTE PERFORMANCE

- Up to 12 compute cores*
 - 4 "Steamroller" CPU cores
 - 8 GCN GPU cores
 - HSA enabled

ENHANCED USER EXPERIENCES

- Video acceleration
- AMD TrueAudio technology
- 4 display heads

HIGH PERFORMANCE CONNECTIVITY

- 128bits DDR3 up to 2133
- PCI-Express® Gen3 x16 for discrete graphics upgrade
- PCI-Express® for direct attach NVMe SSD

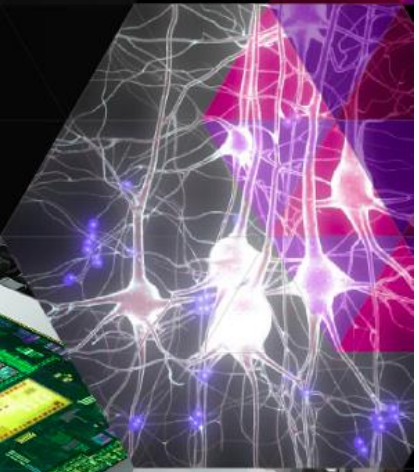
HOTCHIPS 2014



HOT CHIPS 2014 NVIDIA'S DENVER PROCESSOR

Darrell Boggs, CPU Architecture

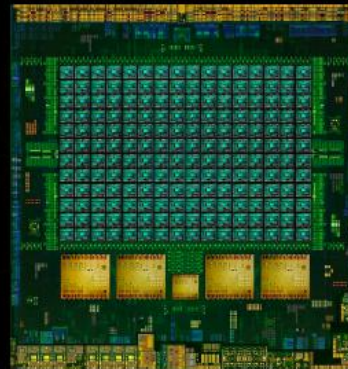
Co-authors: Gary Brown, Bill Rozas,
Nathan Tuck, K S Venkatraman



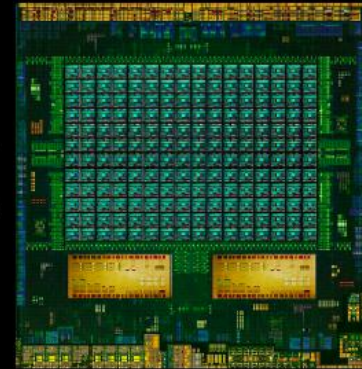
Hotchips 2014

TEGRA K1
192-core
Kepler-Class Chip

One Chip – Two Versions



Pin
Compatible



Quad A15 CPUs

32-bit

3-way Superscalar

Up to 2.3GHz

32K+32K L1\$

Dual Denver CPUs

64-bit

7-way Superscalar

Up to 2.5GHz

128K+64K L1\$

Hotchips 2014 - NVIDIA

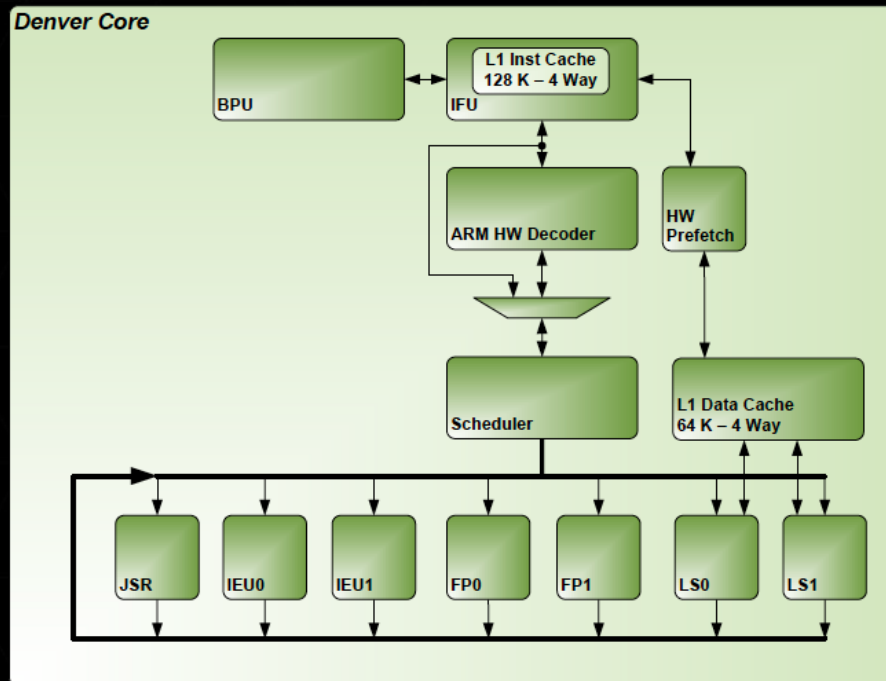
DENVER CPU

Highest Perf ARMv8 CPU

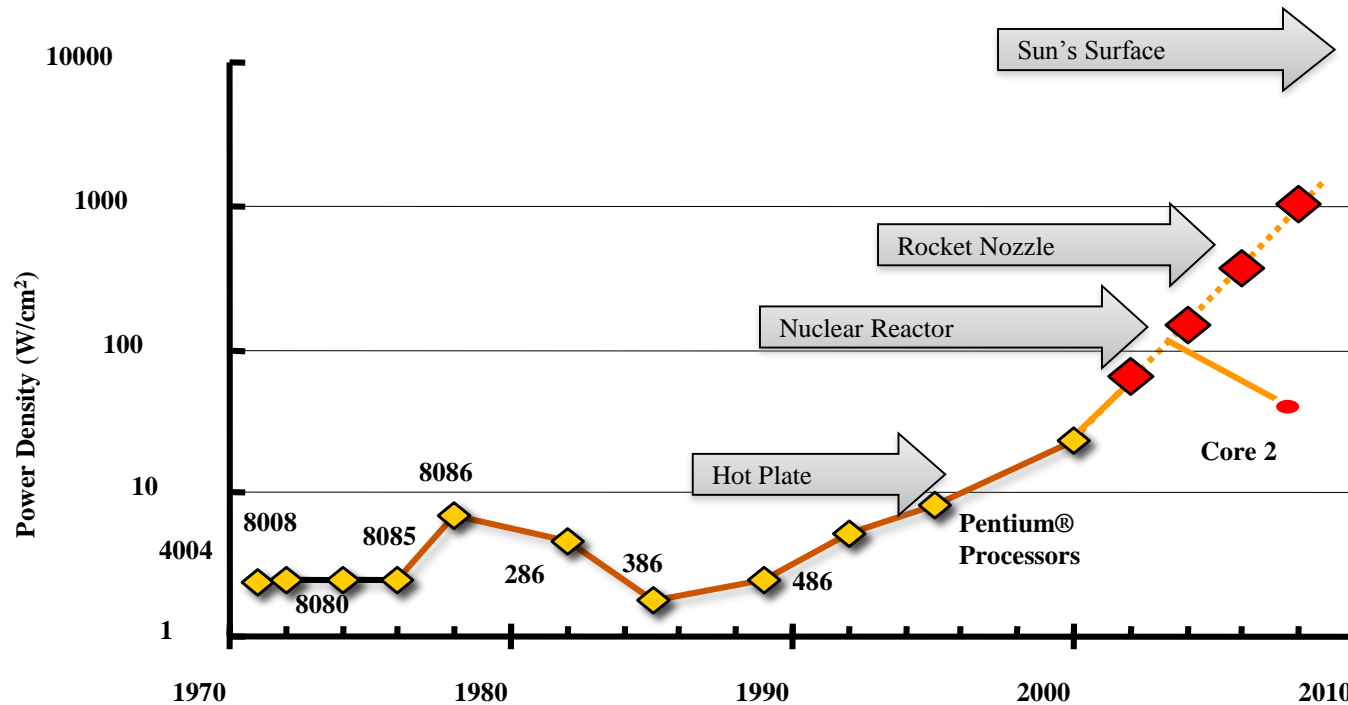
- 7-wide superscalar
- Aggressive HW prefetcher

Dynamic Code Optimization

- Optimize once, use many times
- OOO execution without the power



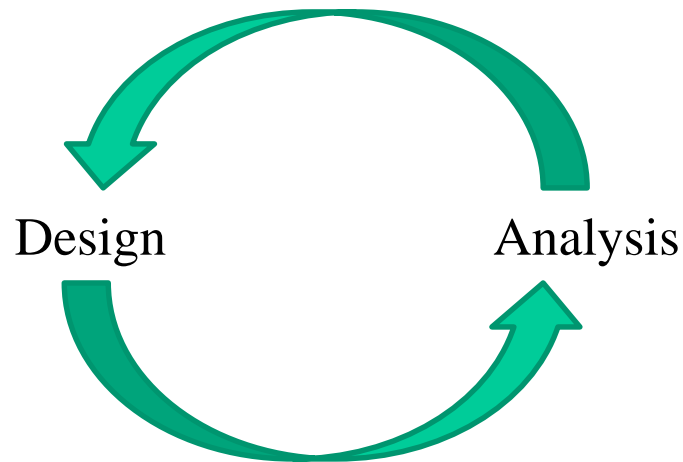
Power Density in Microprocessors



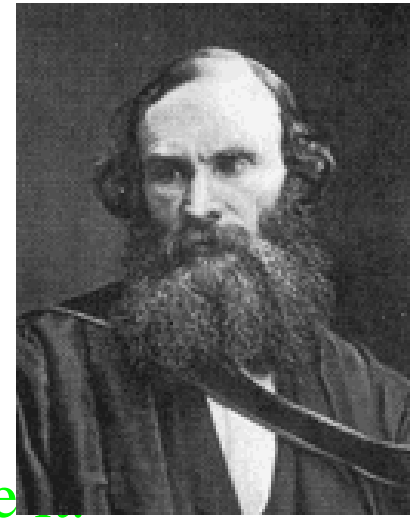
Source: Intel®

Why Performance Evaluation?

- For better Processor Designs
- For better Code on Existing Designs
- For better Compilers
- For better OS and Runtimes



Lord Kelvin



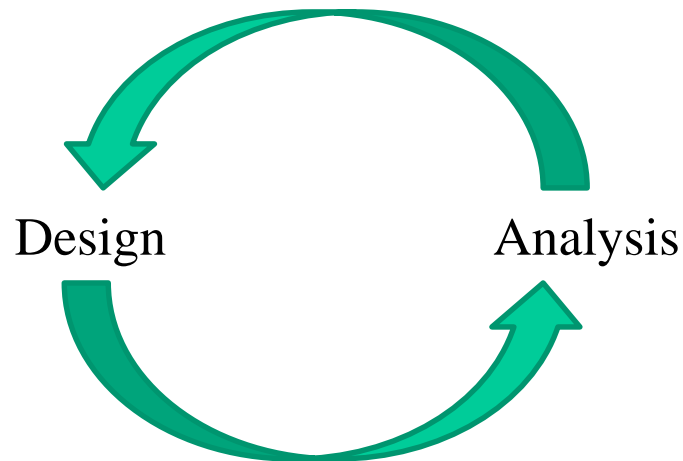
“To measure is to know.”

"If you can not measure it, you can not improve it."

"I often say that when you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely in your thoughts advanced to the state of *Science*, whatever the matter may be." [PLA, vol. 1, "Electrical Units of Measurement", 1883-05-03]

Designs evolve based on Analysis

- Good designs are impossible without good analysis
- Workload Analysis
- Processor Analysis

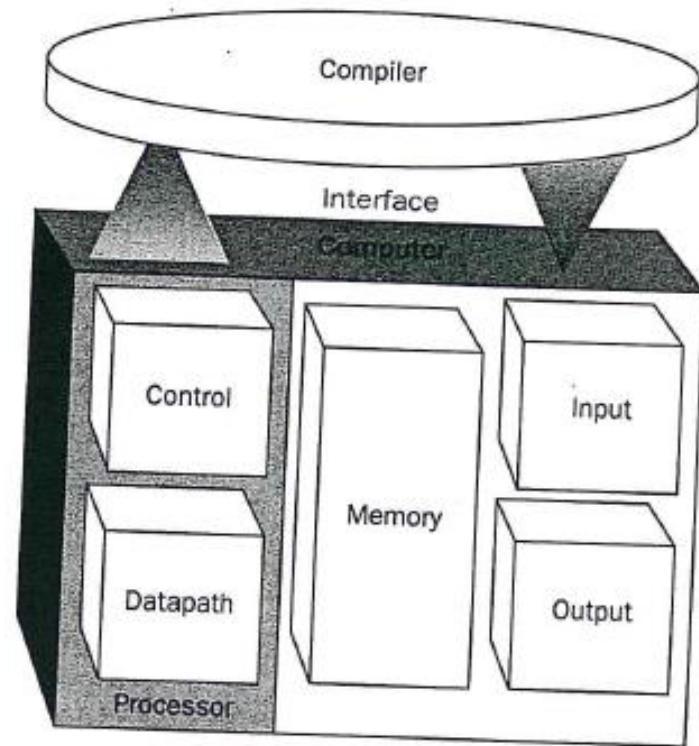


Performance Evaluation - an integral part of good computer architecture

Graphic in Patterson & Hennessy's first edition of the Computer Organization book – **Five Classic Components of a Computer**



Evaluating Performance



Metrics

- **Latency**: time to completely execute a certain task
- **Throughput**: amount of work that can be done over a period of time
- **Power**: instantaneous power during execution of a program
- **Energy**: Total energy consumption during the execution of the whole program
- **Reliability**: Failure rate
- CPI, IPC, MIPS, MFLOPS, MTTF, MTBF, AVF, Transactions/minute, Transactions/hour, MIPS/watt, Watts, Joules, Joules/instr, etc

“Iron Law” of Processor Performance

Processor Performance = Execution Time

$$= \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Cycles}}{\text{Instruction}} \times \frac{\text{Time}}{\text{Cycle}}$$

(code size) *(CPI)* *(cycle time)*

CPI is often used for single-core processors when code size is same and cycle time is same between cases being compared.

Challenges in Performance Evaluation

- Complexity of Processors
- Complexity of Modern Workloads

Performance Evaluation of Early Non-pipelined Processors

Add with Carry (ADC)

Simple non-pipelined processors/microcontrollers

Attached is a datasheet from Motorola 68HC11

Non-overlapped operations

Fixed number of cycles

Add up the cycles according to the addressing mode of the instruction

Operation: $ACCX \leftarrow (ACCX) + (M) + (C)$

Description: Adds the contents of the C bit to the sum of the contents of ACCX and M and places the result in ACCX. This instruction affects the H condition code bit so it is suitable for use in BCD arithmetic operations (see DAA instruction for additional information).

Condition Codes and Boolean Formulae:

S	X	H	I	N	Z	V	C
—	—	⚡	—	⚡	⚡	⚡	⚡

H $X3 \cdot M3 + M3 \cdot \overline{R3} + \overline{R3} \cdot X3$

Set if there was a carry from bit 3; cleared otherwise.

N R7

Set if MSB of result is set; cleared otherwise.

Z $\overline{R7} \cdot \overline{R6} \cdot \overline{R5} \cdot \overline{R4} \cdot \overline{R3} \cdot \overline{R2} \cdot \overline{R1} \cdot \overline{R0}$

Set if result is \$00; cleared otherwise.

V $X7 \cdot M7 \cdot \overline{R7} + \overline{X7} \cdot \overline{M7} \cdot R7$

Set if a twos complement overflow resulted from the operation; cleared otherwise.

C $X7 \cdot M7 + M7 \cdot \overline{R7} + \overline{R7} \cdot X7$

Set if there was a carry from the MSB of the result; cleared otherwise.

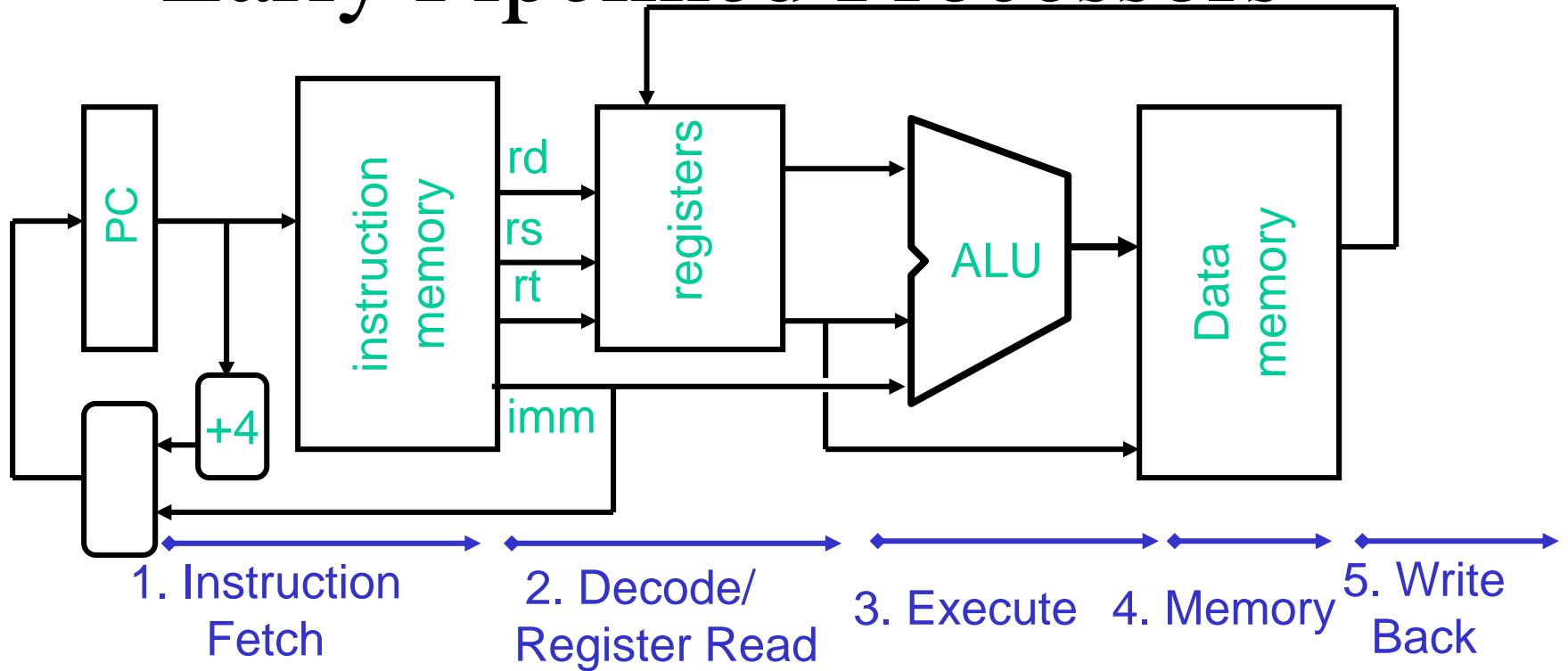
Source Forms: ADCA (opr); ADCB (opr)

Addressing Modes, Machine Code, and Cycle-by-Cycle Execution:

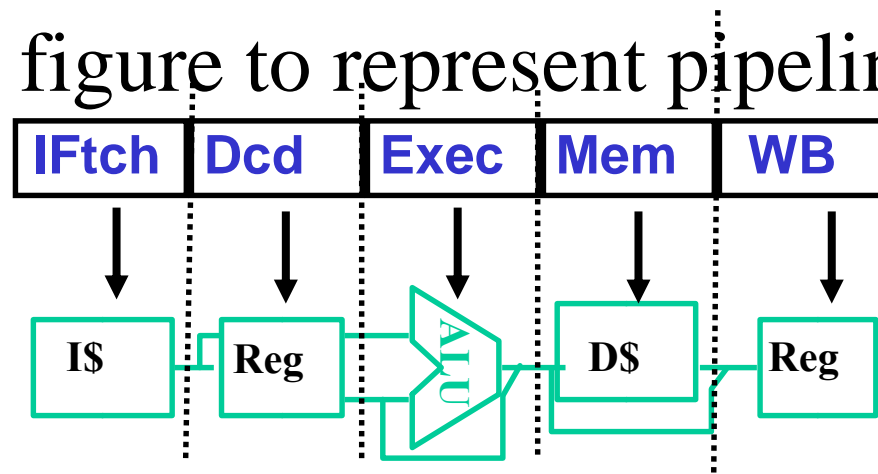
Cycle	ADCA (IMM)			ADCA (DIR)			ADCA (EXT)			ADCA (IND, X)			ADCA (IND, Y)		
	Addr	Data	R/W	Addr	Data	R/W	Addr	Data	R/W	Addr	Data	R/W	Addr	Data	R/W
1	OP	89	1	OP	99	1	OP	B9	1	OP	A9	1	OP	18	1
2	OP+1	ii	1	OP+1	dd	1	OP+1	hh	1	OP+1	ff	1	OP+1	A9	1
3				00dd	(00dd)	1	OP+2	ll	1	FFFF	—	1	OP+2	ff	1
4							hhll	(hhll)	1	X+ff	(X+ff)	1	FFFF	—	1
5										Y+ff	(Y+ff)	1	Y+ff	(Y+ff)	1

Cycle	ADCB (IMM)			ADCB (DIR)			ADCB (EXT)			ADCB (IND, X)			ADCB (IND, Y)		
	Addr	Data	R/W	Addr	Data	R/W	Addr	Data	R/W	Addr	Data	R/W	Addr	Data	R/W
1	OP	C9	1	OP	D9	1	OP	F9	1	OP	E9	1	OP	18	1
2	OP+1	ii	1	OP+1	dd	1	OP+1	hh	1	OP+1	ff	1	OP+1	E9	1
3				00dd	(00dd)	1	OP+2	ll	1	FFFF	—	1	OP+2	ff	1
4							hhll	(hhll)	1	X+ff	(X+ff)	1	FFFF	—	1
5										Y+ff	(Y+ff)	1	Y+ff	(Y+ff)	1

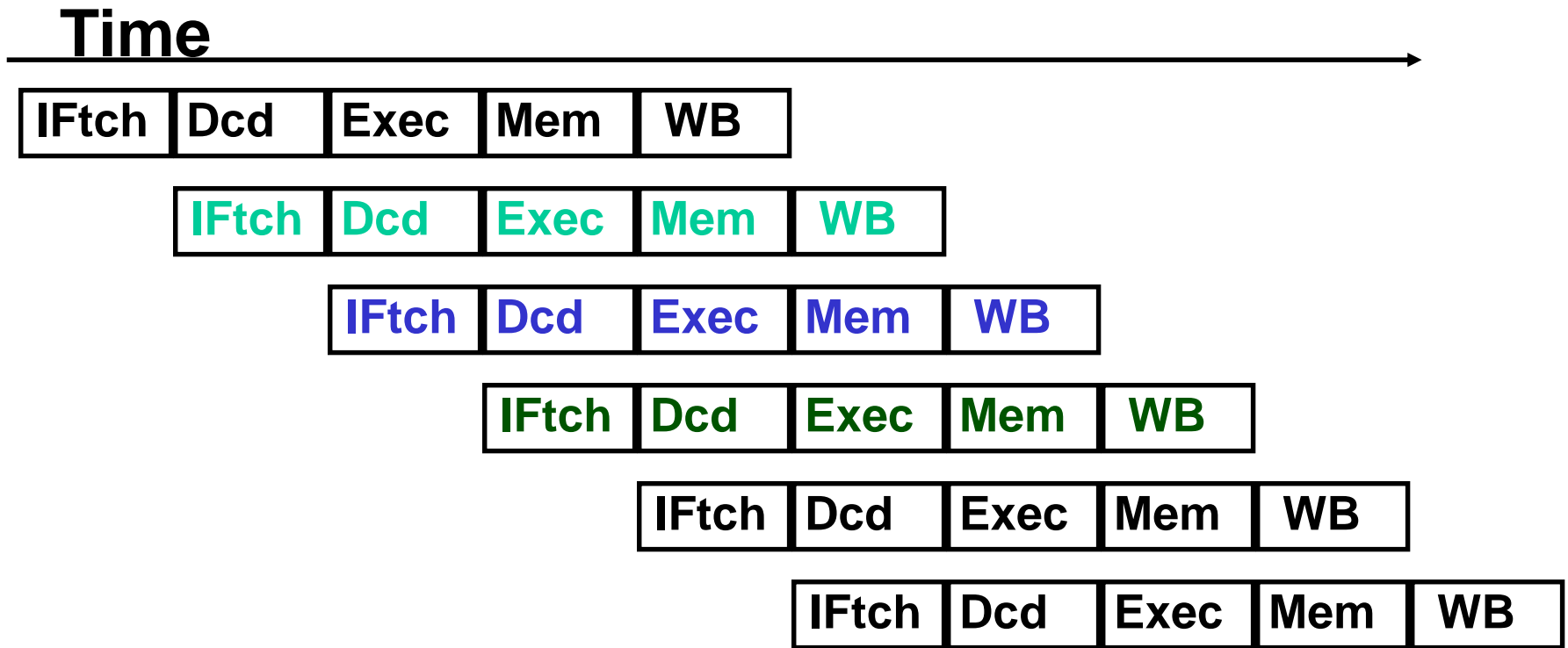
Early Pipelined Processors



- Use datapath figure to represent pipeline



Pipelined Execution Representation



- Evaluate by creating a simulator that mimics this process. Dealing of instruction dependencies and data forwarding etc. modeled in the simulator.

Processor Challenges

Superscalar Processors

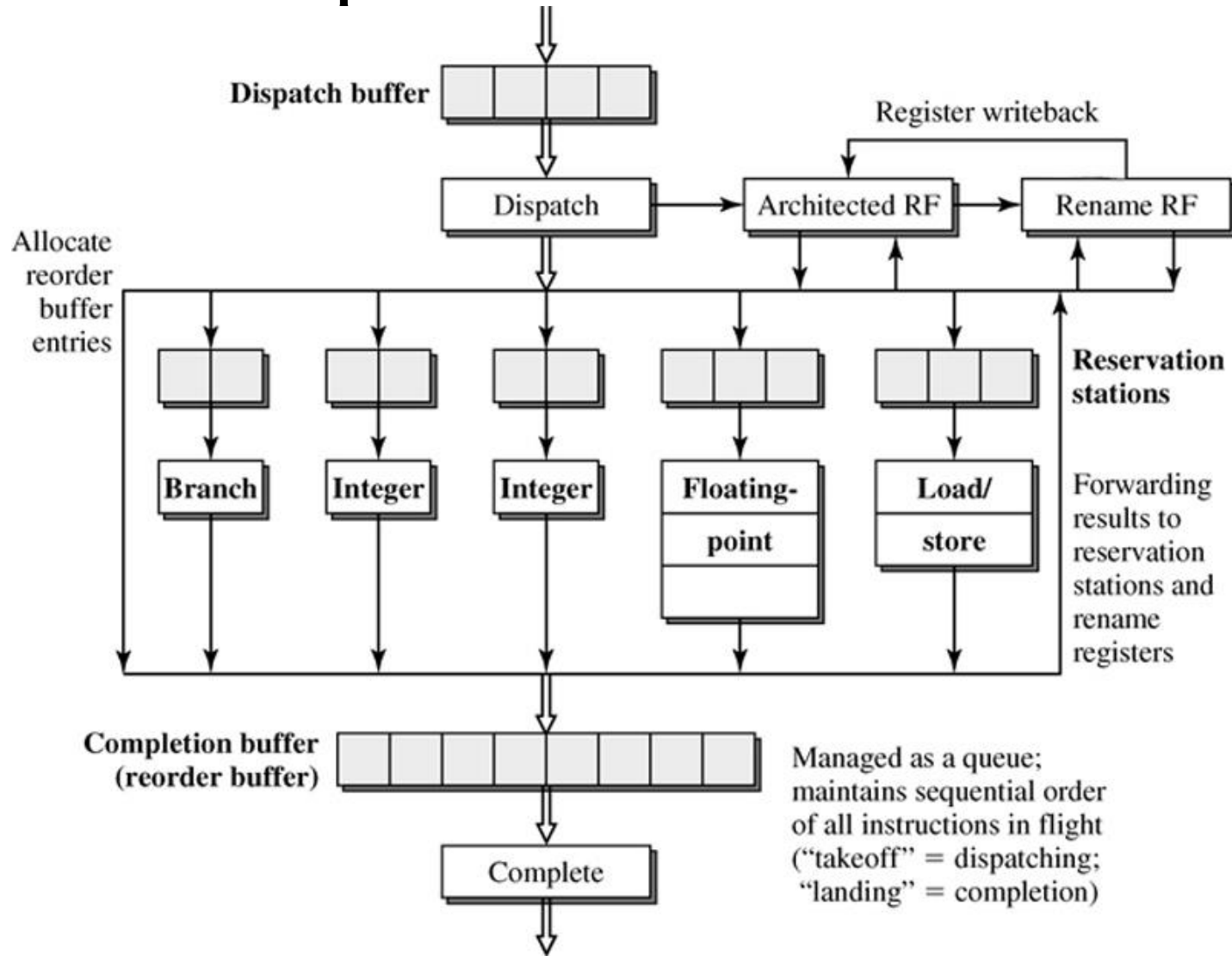
Simultaneously Multithreaded Processors (SMT)
(Also called Hyperthreading)

Multicore Processors

Each core can be Single-threaded

Each core can be Hyperthreaded

Superscalar Processors



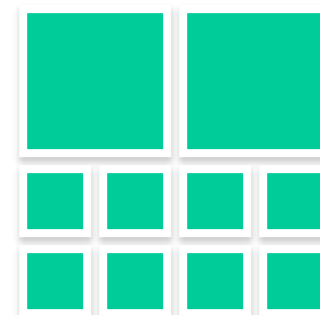
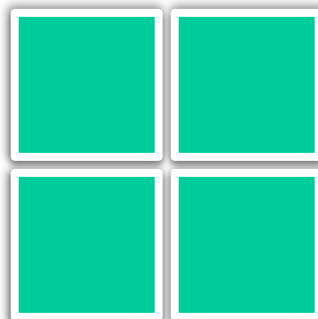
Multicore Processors

- Efficient utilization of big transistor budgets
- Wide superscalars are power hungry
- Have several cores albeit simple
- Operate at a lower energy point
- Run in parallel to recoup lost performance

Heterogeneous Architectures

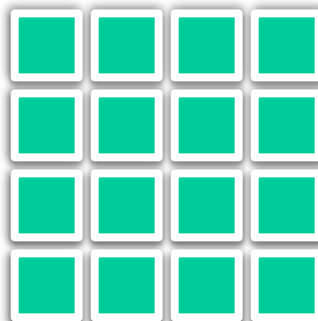
Single ISA Heterogeneous

Cores with same ISA, but with different microarchitectures

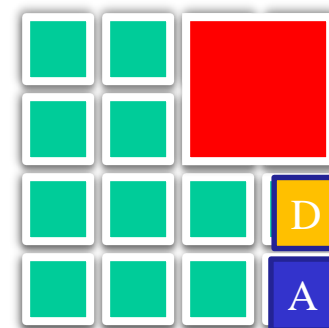
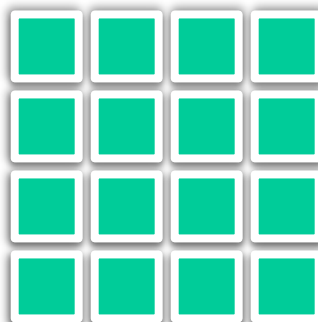


Multiple ISA Heterogeneous

One or more ISAs and Accelerators (main ISA, DSP processor ISA, hardware accelerators)



GPGPU_s



Workload Challenges

Virtualized Workloads

Multiple non-parallelizable applications may be running on multiple cores


Parallelizable Applications

Operating Systems and Runtimes –
Dynamic Mapping, Scheduling


Compiler optimizations

Complex Workloads - Heterogeneous Architectures

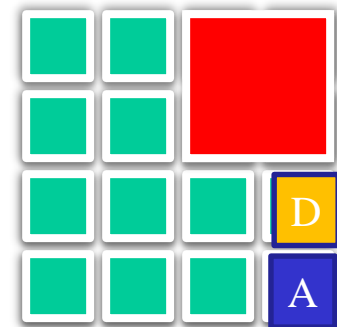
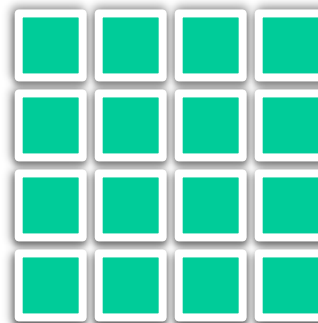
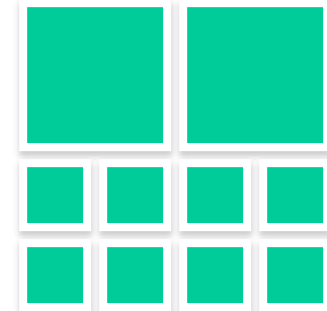
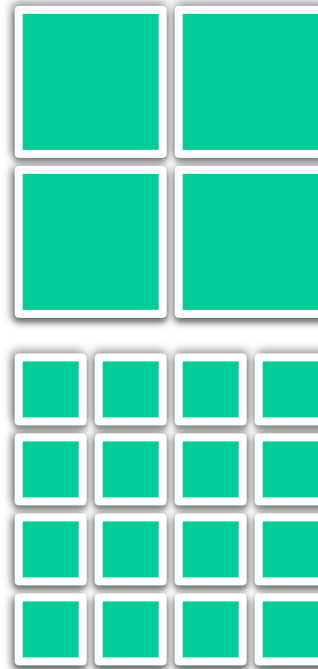
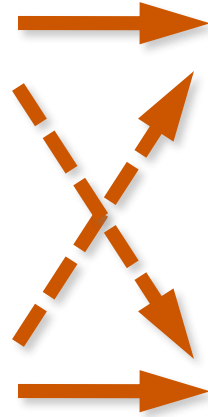

Multiprogrammed workloads: e.g. SPEC CPU



Multithreaded workloads: e.g. PARSEC



Diversity inside programs



“Iron Law” of Processor Performance

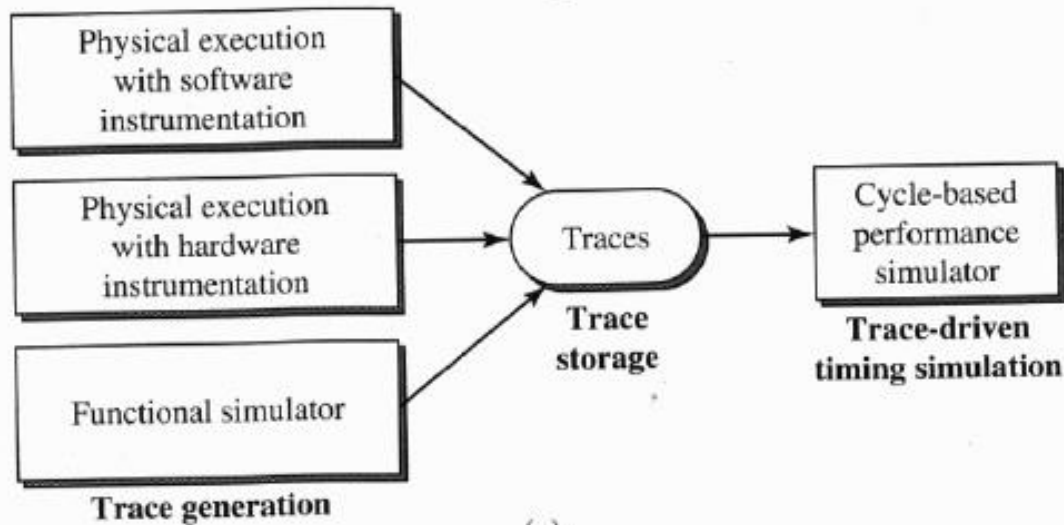
Processor Performance = Execution Time

$$= \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Cycles}}{\text{Instruction}} \times \frac{\text{Time}}{\text{Cycle}}$$

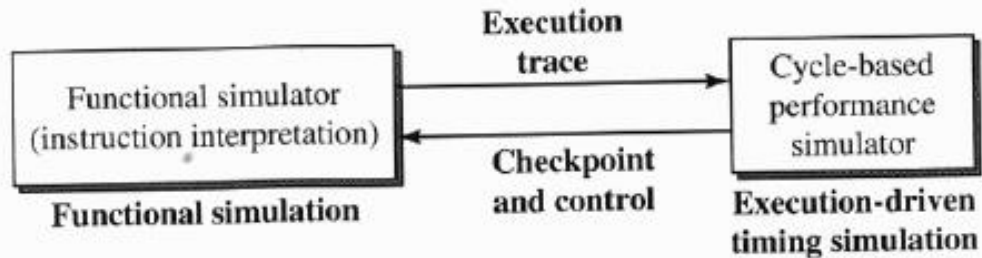
(code size) (CPI) (cycle time)

CPI is often used for single-core processors when code size is same and cycle time is same between cases being compared.

Simulation Methods



(a)



(b)

Classification of Techniques

- Performance Modeling
 - Simulation
 - Trace-Driven Simulation
 - Execution Driven Simulation
 - Complete System Simulation
 - Event-Driven Simulation
 - Statistical Simulation
 - Analytical Modeling
 - Probabilistic Models
 - Queuing Models
 - Markov Models
 - PetriNet Models
- Performance Measurement
 - On-Chip Hardware Monitoring
 - Off-Chip Hardware Monitoring
 - Software Monitoring
 - Microcoded Instrumentation

PRESILICON EVALUATION

- Required in early design stages
- Before prototypes can be built
- Pre-silicon
- Very important because many design decisions are made based on this
- Timeliness of products are important in today's competitive world

POST-SILICON EVALUATION

- To improve current generation compilers
- To improve current generation operating systems and runtimes
- To improve current generation hardware
- To improve next generation of products

Evaluation of Modern and Future Processors

Huge Challenge

Evaluating one processor is hard enough

Evaluating all the software and hardware layers involved

The design process, the tradeoff evaluation, depends largely on the performance evaluation. Your company's future depends on the performance (P, P, E) estimates you project for potential designs.