Informatics 3

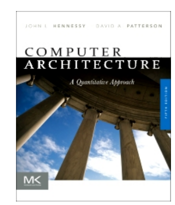
Computer Architecture

Dr. Vijay Nagarajan and Prof. Nigel Topham

Institute for Computing Systems Architecture, School of Informatics
University of Edinburgh

General Information

- Instructors : Vijay Nagarajan (vijay.nagarajan@ed.ac.uk)
  Nigel Topham (npt@staffmail.ed.ac.uk)
- TA: Matthew Bielby (s0679596@sms.ed.ac.uk)
- Assignments (25%)
  - Assignment1 (5%) : Out 7/2/13 **due 14/2/13**
  - Assignment2 (20%) : Out 28/2/13 **due 21/3/13**
- Tutorials
  - Starts 3rd week
- Book:
  - Hennessy & Patterson - Computer Architecture: A Quantitative Approach – Morgan Kaufmann – 5th edition (4th edition also ok)



- Lecture slides + book = lecture notes
- Please interrupt with questions at any time

What is Computer Architecture ?

What is Computer Architecture?

- Architecture
  - Science and art of interconnecting building **materials** to construct various **buildings**, subject to **constraints**
  - Materials: brick, concrete, glass etc.
  - Buildings: house, Office, Auditorium etc.
  - Constraints: cost, safety, time etc.

- Computer Architecture
  - Science and art of interconnecting hardware **components** to create **computers**, subject to **constraints**
  - Hardware components: gates, circuit technology etc.
  - Computers: desktop, server, mobile phone etc.
  - Constraints: performance, energy, cost etc.

Three Aspects of Computer Architecture

- Instruction Set Architecture (ISA)
  - Programmer/compiler view
  - Instructions visible to the (system) programmer
  - Opcode, addressing mode, architectural registers etc.

- Microarchitecture
  - Processor designer view
  - Logical organization that implements the ISA
  - Pipelining, functional units, caches, physical registers etc.

- Hardware
  - Chip designer view
  - Detailed logic design and the packaging technology
  - Gates, CMOS process etc.

Three Aspects of Computer Architecture

- Instruction Set Architecture (ISA)
  - Programmer/compiler view
  - Instructions visible to the (system) programmer
  - Opcode, addressing mode, architectural registers etc.
- Microarchitecture
  - Processor designer view
  - Logical organization that implements the ISA
  - Pipelining, functional units, caches, physical registers etc.
- Hardware
  - Chip designer view
  - Detailed logic design and the packaging technology
  - Gates, cells, CMOS process etc.

Course Focus

What do we cover in this course?

- Introduction
  - What is Computer Architecture?
  - Technological trends

- Measuring performance

- Instruction Set Architecture

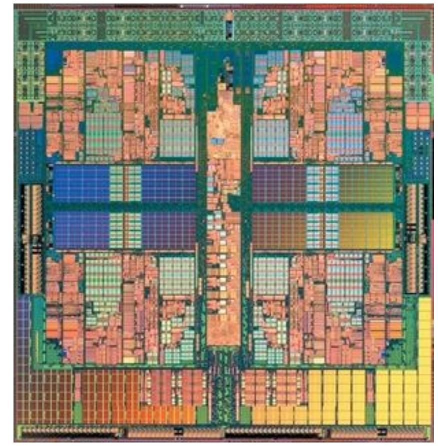
- Processor architecture
  - Pipelining, branch prediction, dynamic scheduling

- Memory systems
  - Cache hierarchies
  - High-performance memory organizations

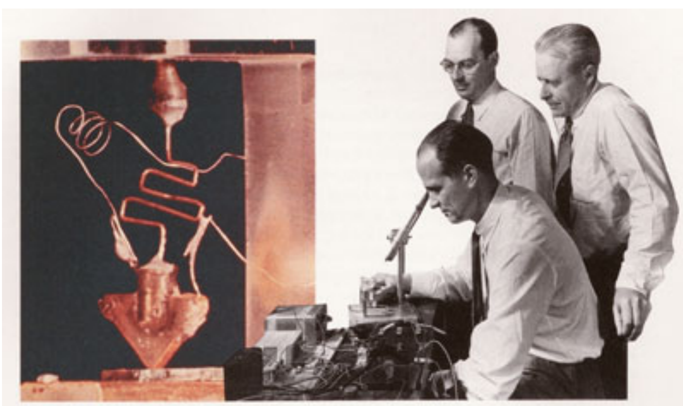
- Introduction to parallel architectures

The Rise of the Electronic Computer

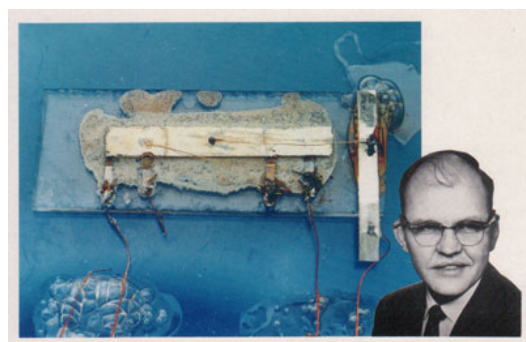
- Today's microprocessors evolved over a period of 40 years
- Key developments were:
  - First transistor in 1947
  - First integrated circuit in 1958
  - First microprocessor in 1972
- Since the first electronic computers were created in the late 1940's, performance has increased at a dramatic rate, due to:
  - Advances in integrated circuit technology
  - **Computer Architecture**
  - Compiling Techniques
  - Algorithm Development



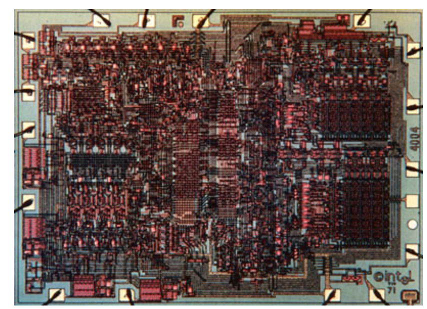
AMD 'Phenom' quad-core



William Shockley, and the first transistor (1947)



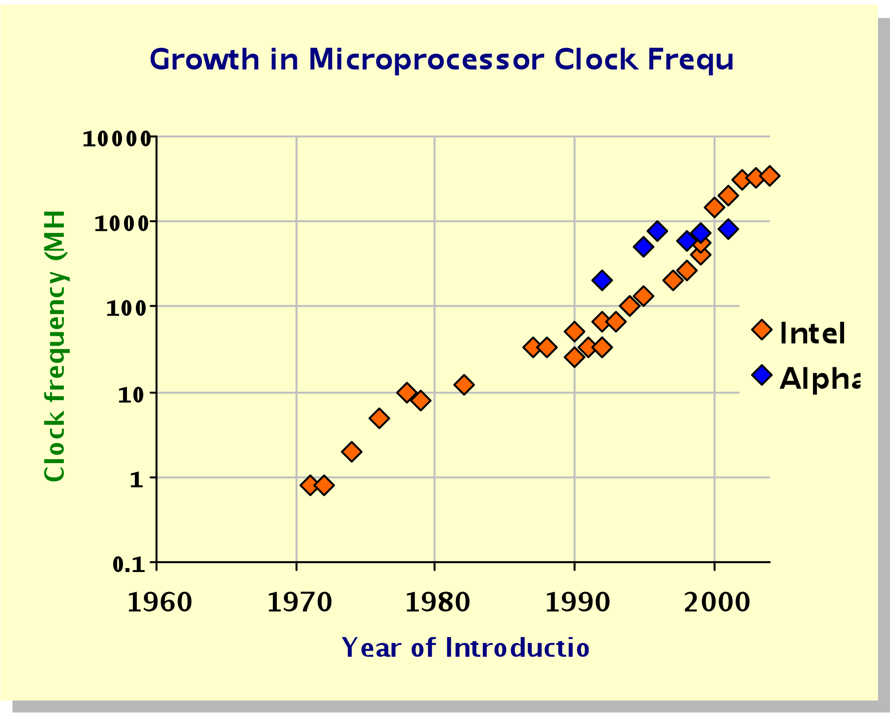
Jack Kilbey with the first IC (1958)

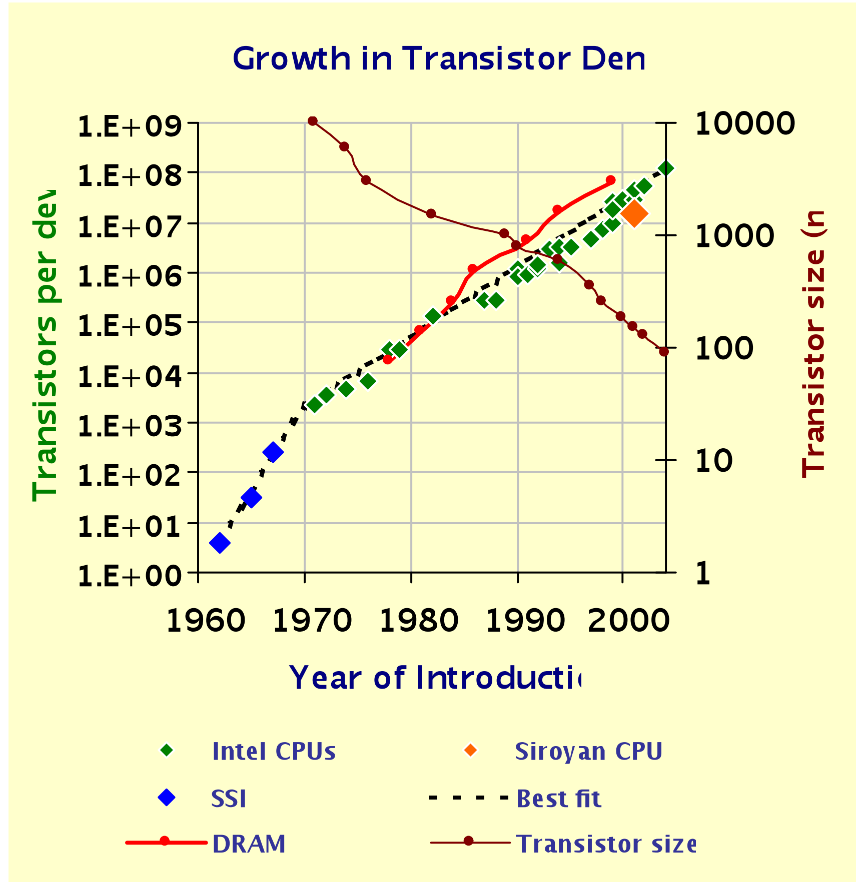


Intel 4004 (1972)

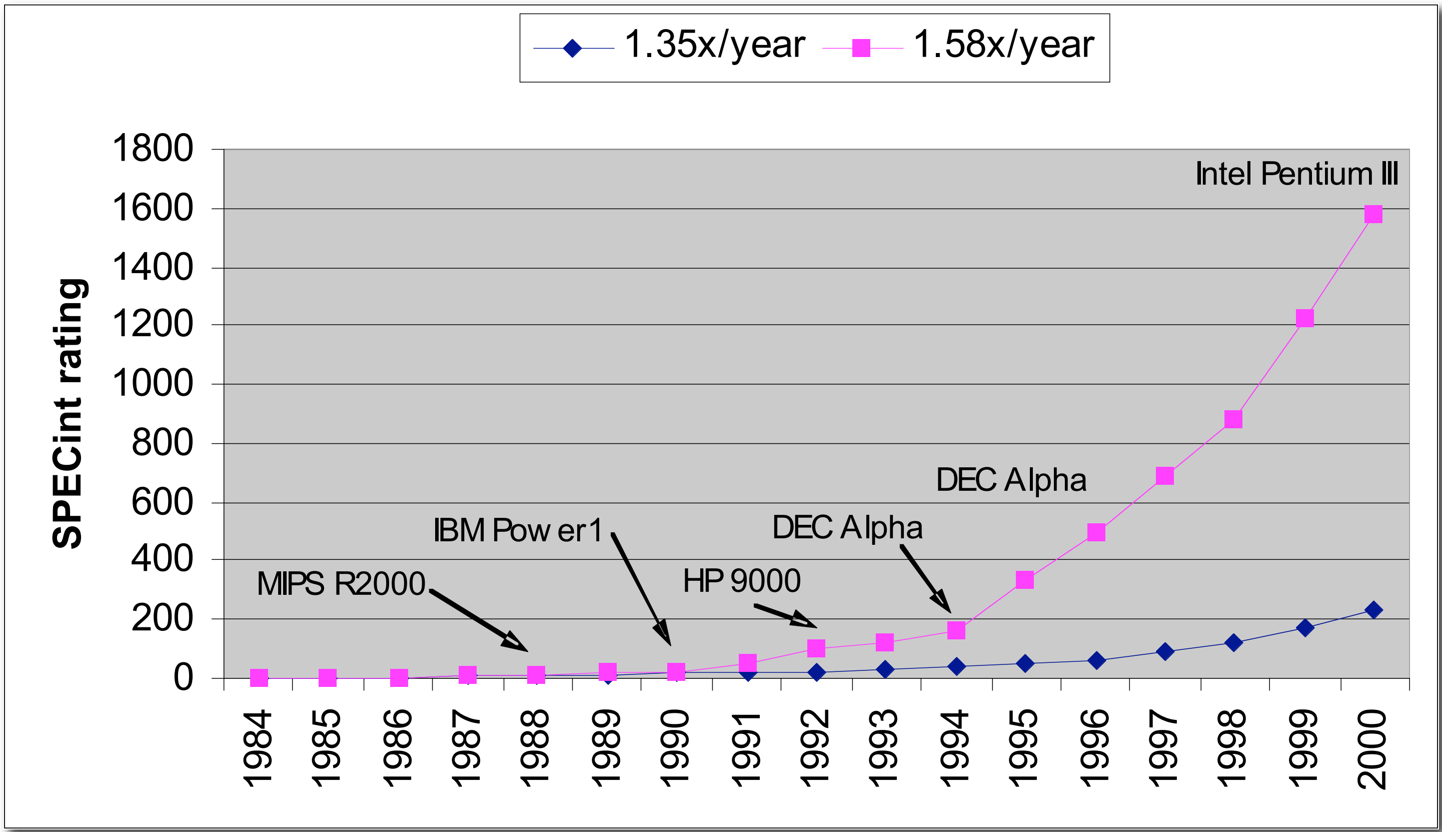
The Evolution of Computer Technology

- 1965 – Gordon Moore's "Law"
  - Densities double every year (2x)
- 1975 – Moore's Law revised
  - Densities double every 2 years (1.42x)
- Actually 5x every 5 years (1.35x)





Tracking Technology: The role of CA



H&P
Fig. 1.1

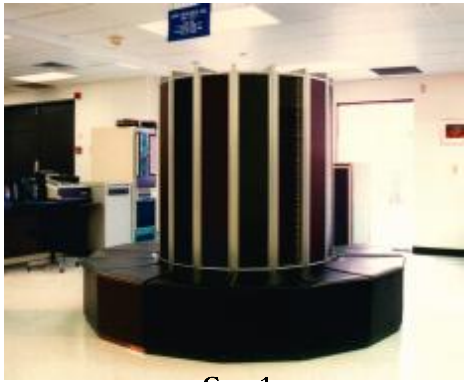
- **Bottom-line: architectural innovation complements technological improvements**

What drives Computer Architecture?

- Early computers – simplest possible implementation
- CDC 6600, 7600 and IBM S/360 model 91
  - Emergence of pipelined processors
  - Parallel execution units
  - Hardware detects parallelism between instructions
- `Scientific' computing
  - Driven mostly by US military requirements
  - Development of vector processing
- Microprocessor revolution
  - Driven by commercial and desktop use
  - Not initially a threat to mainframes
  - Eventually rendered mainframes obsolete
- Architecture of microprocessors?
  - Started simple, just like early computers
  - Re-used concepts from 6600, 7600 and 360/91
  - But also integrated fast memory close to CPU
- What drives CA in 2013?

CDC 6600

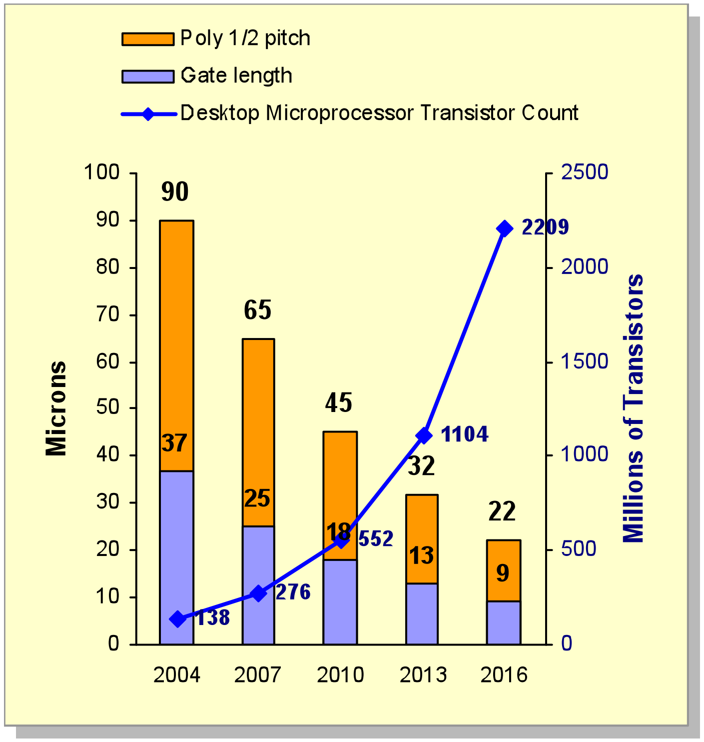
IBM S/360 model 91



Cray 1

Future Technology Predictions (2003)

- Moore's Law will continue to ~2016
- Procs. will have 2.2 billion transistors
- DRAM capacity to reach 128 Gbit
- Procs. clocks should reach 40 GHz?
  - Should have reached 10 GHz by now!



Source: International Technology Roadmap for Semiconductors, 2003

State-of-the-art - January 2013

- IBM Power7+
  - 2.1 billion transistors
  - 4, 6 or 8 core versions
  - 4 SMT threads per core
  - 32nm CMOS
  - 4.4 GHz clock

- AMD Bulldozer
  - 1.2 billion transistors
  - 4, 8 cores
  - 32nm
  - 4.2 GHz
  - 125 W

- Intel i7 ‘Ivy bridge’ 4-core + GPU
  - 1.4 billion transistors
  - 22nm CMOS silicon
  - 4 cores, 32 threads
  - 3.5 GHz
  - 77 Watts

End of Frequency Scaling

- Frequency of processors have stopped scaling since ~ 2005

- Why?
  - Power wall

- Parallel architectures have arrived
  - Performance through parallelism
  - Software must expose parallelism
  - CA should enable this

Current Trends in High Performance CA

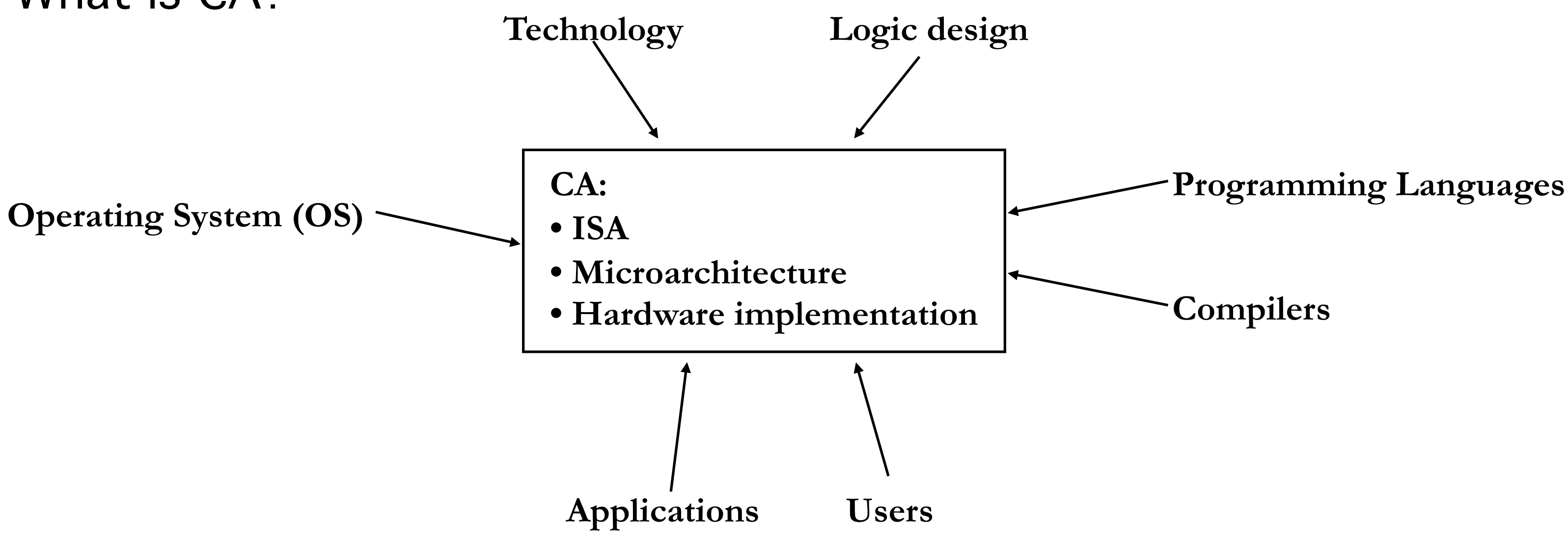
- Very complex processor design:
  - "Hybrid" branch prediction (MIPS R14000)
  - Out-of-order execution (MIPS R14000)
  - Multi-banked on-chip caches (Alpha 21364, now defunct)

- Parallelism and integration at chip level:
  - Chip-multiprocessors (Sun "Niagara", IBM Power5, Intel Core 2 Duo)
  - Multithreading (Intel Hyperthreading, IBM Power5, Sun "Niagara")
  - Embedded Multi-core Systems On a Chip (ARM MP, most mobile phones)
  - CPU+GPU, e.g. AMD Fusion (2010 – 2011)

- Power-conscious designs

Summary: Computer Architecture

- What is CA?



- Objectives of CA:
  - High-performance
  - Low cost
  - Reliability
  - Low power consumption

Why study CA?

- Technology is always changing
  - CA should adapt
- Requirements are always changing
  - Speed, power, cost, reliability etc.
- Understand computer performance
- Get a (design or research) hardware job
  - ARM, Intel, IBM etc.