Computer Organization Introduction

COE608: Computer Organization and Architecture

Dr. Gul N. Khan
http://www.ee.ryerson.ca/~gnkhan
Electrical and Computer Engineering
Ryerson University

Overview

- Course Management
- Introduction to Computer Architecture
  - Instruction Set Architecture
  - Machine Organization
- VHDL Review

Lectures and Labs

Half Notes

- You will need to take additional notes from lectures and the text and reference books.

Labs and Assignments

- Aimed at concept reinforcement and practical experience.

Lecture and Laboratory Material available at the course website

http://www.ee.ryerson.ca/~courses/coe608/

Assessment and Evaluation

Labs: 30%
Mid-term Exam: 30%
Final Examination: 40%

Course Text and Reference Materials

Recommended Text Book and support material

Computer Organization and Design, The Hardware/ Software Interface, 4th or 5th edition 2013.
David Patterson and John Hennessy
ISBN 978-0-12-374493-7, Morgan Kaufmann Pub. (Elsevier)

Useful Text Book for Labs: Embedded Core Design with FPGAs
Z. Navabi, McGraw Hill 2007
ISBN 978-0-07-147481-8 or ISBN 0-07-147481-1

VHDL Tutorial
Print or download from
http://www.ee.ryerson.ca/~courses/coe608/public_html/vhdl_tutorial.ps

Reference Book and other material:

Logic and Computer Design Fundamentals,
by *Morris Mano and Charles R. Kime*
Prentice-Hall 3rd or 4th Edition 2004, 2008.

VHDL Reference
Refer the files in the subdirectory at /usr/common/docs/VHDL/

Look for Tutorial and VHDL-Cookbook

- **In addition to the text and reference books, lectures and labs may contain material from other sources.**

Introduction

Computer design is a rapidly changing field:

- Relay ⇒ vacuum-tube ⇒ transistors ⇒ IC ⇒ VLSI ⇒ ULSI
- Doubling every 1.5-2.5 years
  - Memory capacity
  - Processor speed

  (Advances in technology and organization)

Things you will be learning:

- How CPU/computers work
- How to analyze their performance
- Issues affecting modern CPUs (processors)
- Designing a processor

Why learn this stuff?

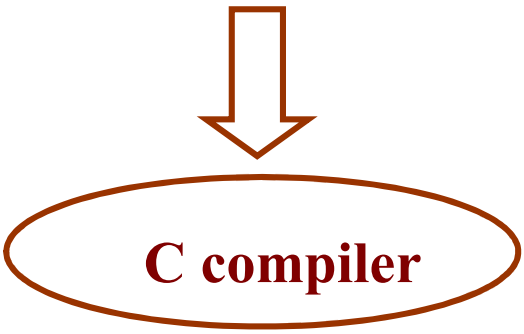
- To call yourself a "computer engineer"
- You want to build software for people to use (need performance)
- You make a good purchasing decision for computers
- Offer "expert" advice

Abstraction in Software

High-level language program (in C)

v[k] ⇔ v[k+1]; SWAP

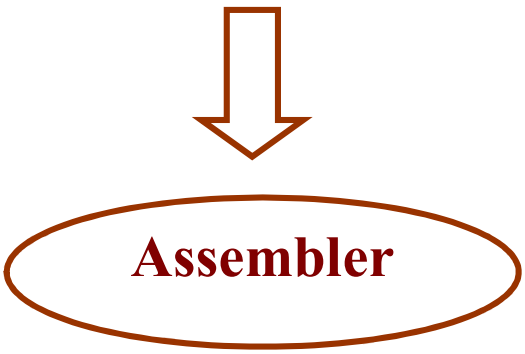
```
temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;
```



Assembly language

Program (for MIPS Processor)

```
lw $15, 0($2)
lw $16, 4($2)
sw $16, 0($2)
sw $15, 4($2)
```



Machine language

Program -for MIPS Logic Processor (M2000)

```
1000 1100 0110 0010 0000 0000 0000 0000
1000 1100 1111 0010 0000 0000 0000 0100
1010 1100 1111 0010 0000 0000 0000 0000
1010 1100 0110 0010 0000 0000 0000 0100
```

Instruction Set

A very important abstraction for computer architecture:

- Interface between hardware & low-level software
- Standardizes instructions and machine language bit patterns.
- Advantage:
  One can have different implementations of the same architecture.
- Disadvantage:
  It sometimes prevents using new innovations.

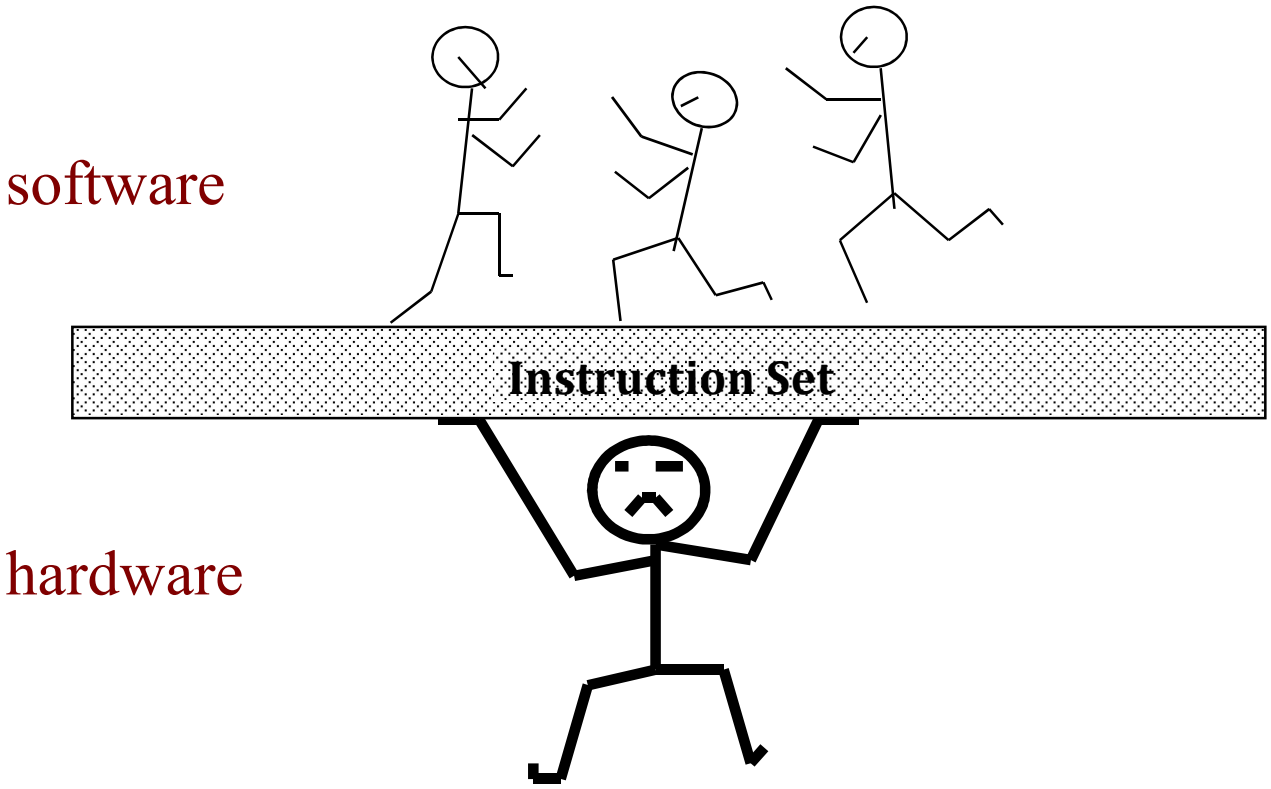
Is binary compatibility is extraordinarily important? (True or False)

An abstraction omits unnecessary details. But it helps to cope with complexity.

Instruction Set Architecture (ISA)

Main Goals for ISA Design

Maximize performance, minimize cost and reduce design time



Between Software and Hardware

Which is easier to modify?

ISA: Instruction Set Architecture

- Organization of Programmable Storage
- Data Types and Data Structures:
  Encoding and Representations
- Instruction Set
- Instruction Formats
- Modes of Addressing
  Accessing Data Items and Instructions
- Exceptional Conditions

Some examples of ISA

DEC Alpha	(v1, v3)	1992-97
HP PA-RISC	(v1.1, v2.0)	1986-96
SGI MIPS	(MIPS I-V)	1986-96
Sun SPARC	v8, v9, UltraSparc versions	1987-2011
Intel	(8086,80x86 Pentium II - IV,6, i3-7 series	1978-2011
ARM (v4-v7)	(7, 9, 11, Cortex R*, M*, A*	1998-201x

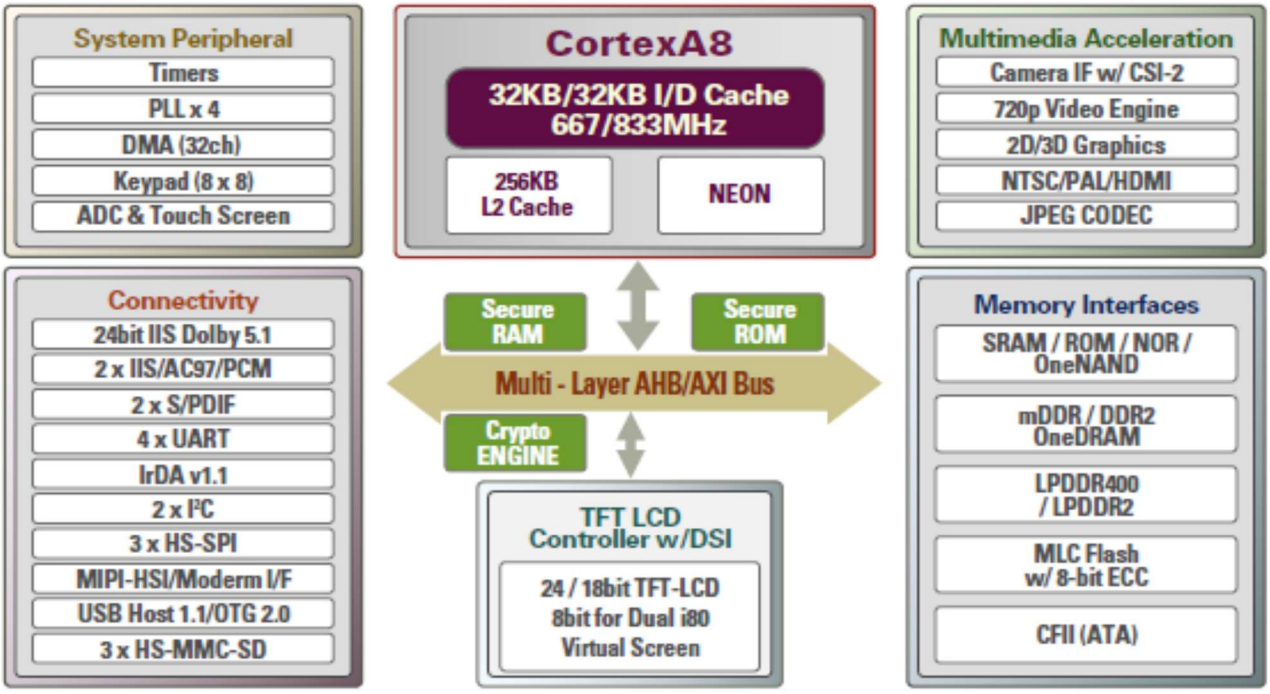
ARM Versions Cores and Architectures

- There is difference between ARM7™ & ARMv7.
- Look into ARM architecture on Wikipedia to get the full list.
- ARM doesn't make ICs but soft cores….maybe a few test chips.

Family	Architecture	Cores
ARM7TDMI	ARMv4T	ARM7TDMI(S)
ARM9 **ARM9E**	ARMv5TE(J)	ARM926EJ-S, ARM966E-S
ARM11	ARMv6 (T2)	ARM1136(F), 1156T2(F)-S, 1176JZ(F), ARM11 MPCore™
Cortex-A **Cortex-R** **Cortex-M**	ARMv7-A ARMv7-R ARMv7-M ARMv6-M	Cortex-A5,7,8,9,15 Cortex-R4(F) Cortex-M3, M4 Cortex-M1, M0

Samsung S5PC100 SoC

S5PC100 has a 32-bit ARM Cortex A8 microprocessor
Operates up to 833MHz with 64/32-bit bus architecture

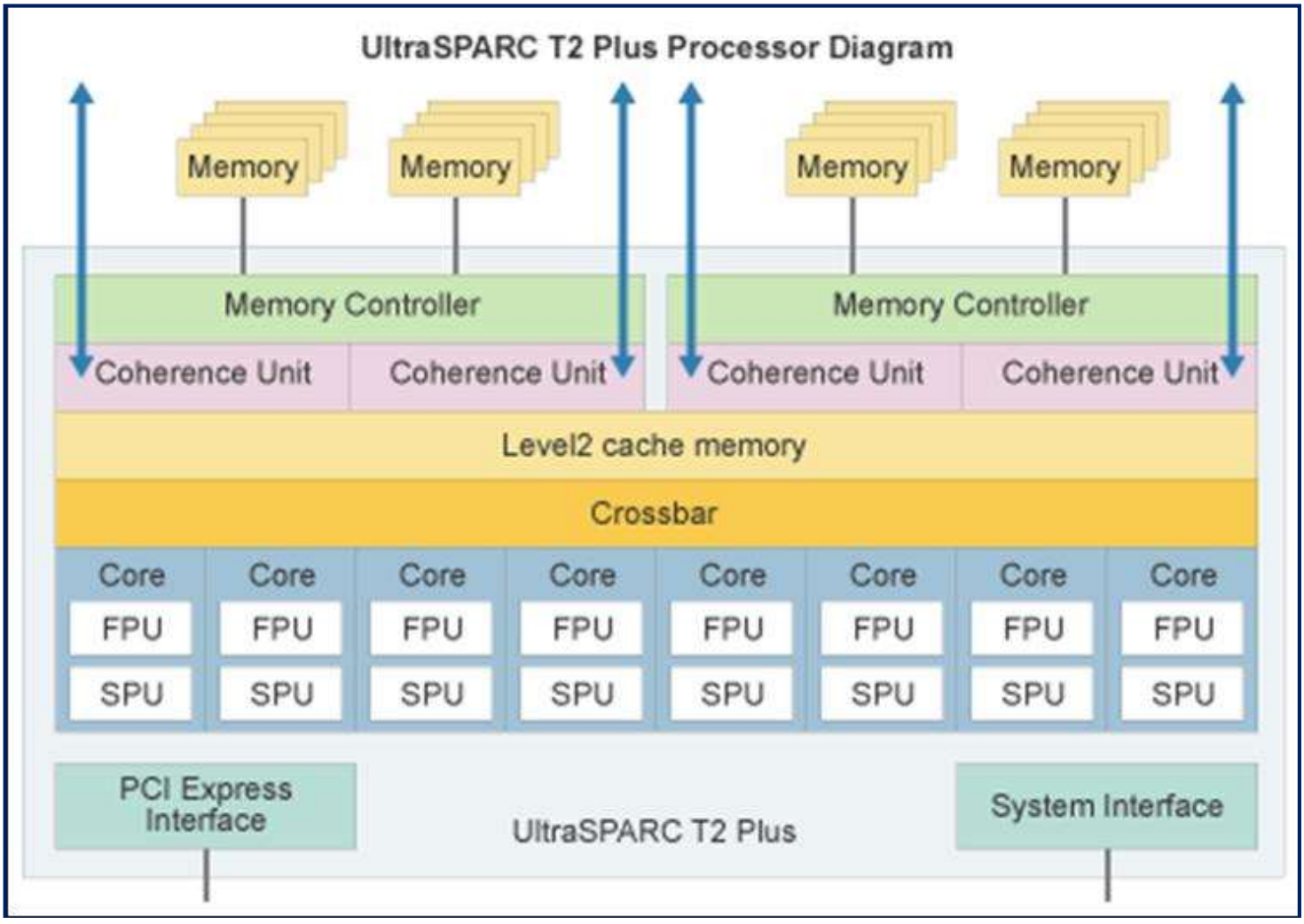


S5PC100 has various functionalities such as Wireless communication, Personal navigation, Camera, Portable gaming, Video player and Mobile TV into one device.

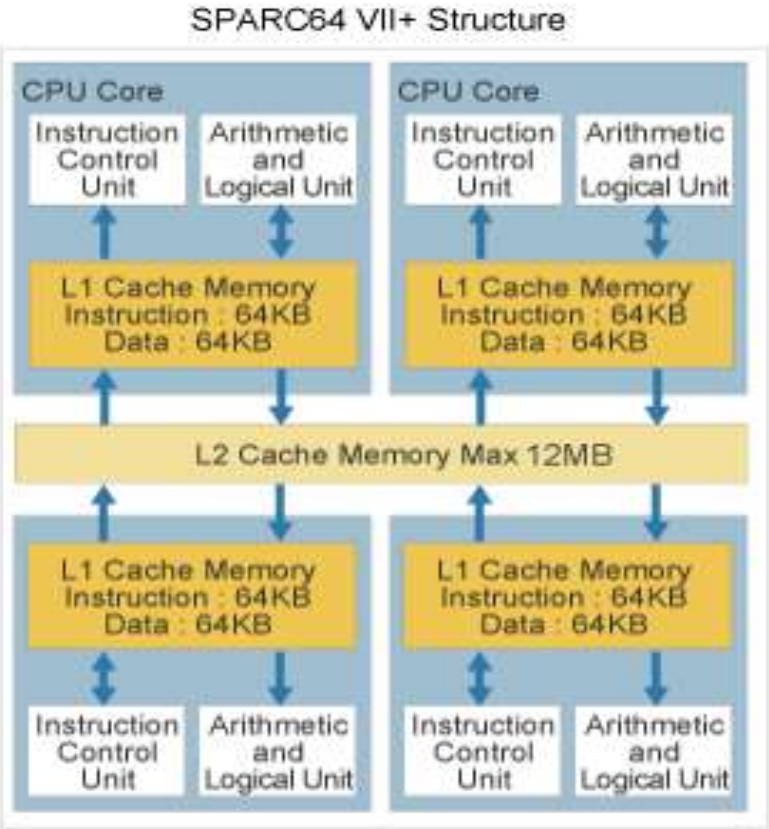
Used in iPhone 3GS and iPod touch 3rd generation.

UltraSPARC T2/T2-Plus

Ultra SPARCT2 has a 10Gbit Ethernet interface.



SPARC64

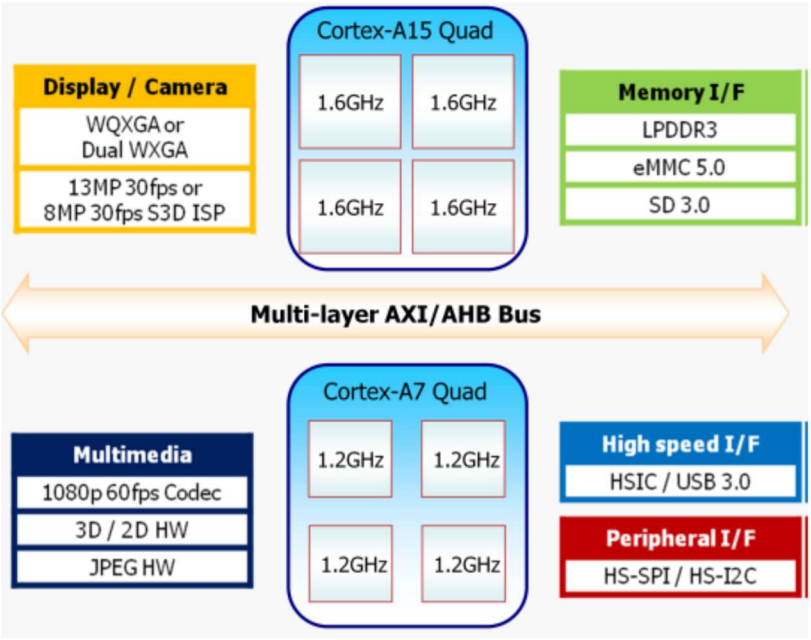


Multicore Architecture

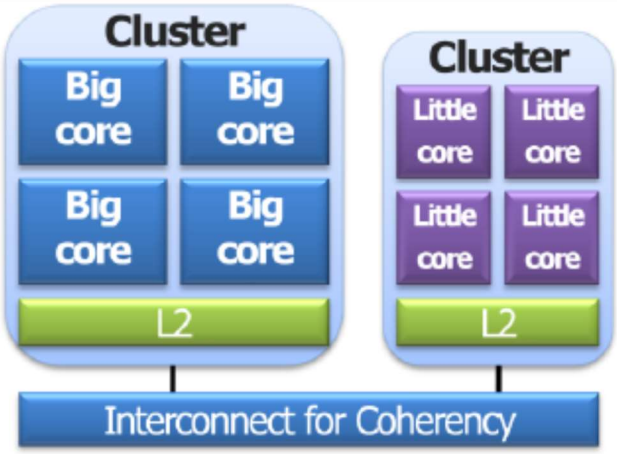
	SPARC64 VI	SPARC64 VII	SPARC64 VII+
CPU cores per chip	2	4	4
Level-1 cache	256KB per core	128KB per core	128KB per core
Maximum Level-2 cache	6MB with 12 ways at maximum (per CPU chip)	6MB with 12 ways at maximum (per CPU chip)	12MB with 12 ways (per CPU chip)

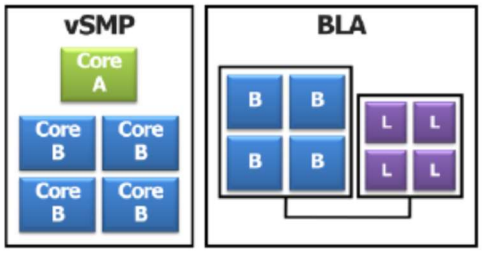
Also used in Fujitsu Supercomputer

Exynos 5410 Octa Processor



big.LITTLE Processing provides 50 percent energy saving with same performance by using A15 CPUs (big @ 1.6GHz) as well as efficiency tuned low Power A7 CPUs (LITTLE @ 1.2GHz).





- Octa core Mobile CPU with big.LITTLE processing
- 3D graphics – fast/efficient operation for smartphone/tablets. 12.8 GB/s memory bandwidth, 1080p 60 fps video.

- vSMP: used in Krait, a customized CPU of Qualcomm with 4 high performance cores Cortex-A15 and one low power Cortex A7.
- For a comparison, big.LITTLE processing is depicted in the right part of where B represents big core and L represents LITTLE core.
- 28nm enables 10 percent performance improvement from 32nm.

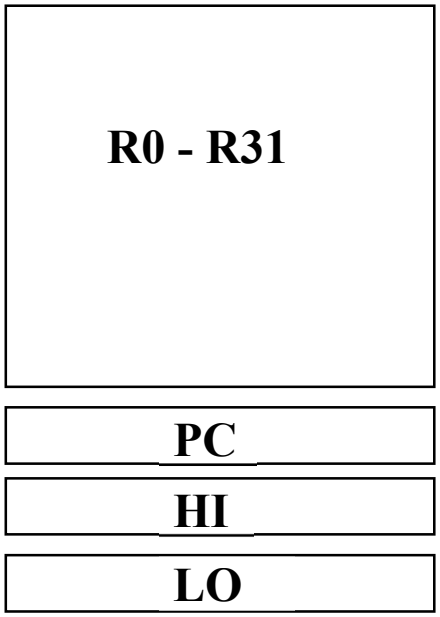
MIPS R2000/R3000 Processor

(First RISC based CPU series 1985/88)

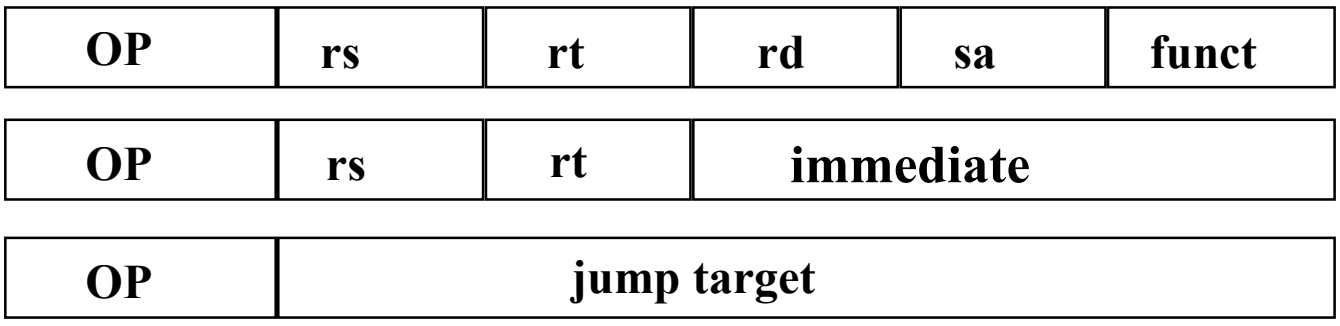
ISA: Instruction Set Architecture

Instruction Categories

- **Load/Store**
- **Computational**
- **Jump and Branch**
- **Floating Point**
  -Coprocessor
- **Memory Management**
- **Special**

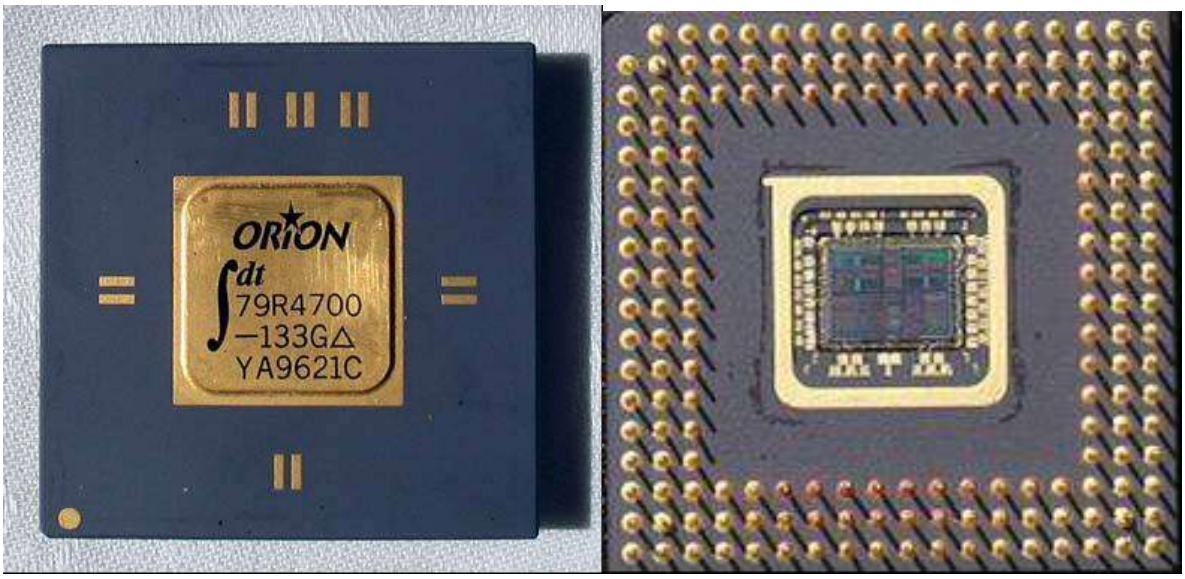


3 Instruction Formats: all 32 bits wide



MIPS R4700/R4000 Processor

64-bit Architectures (2009/2010)



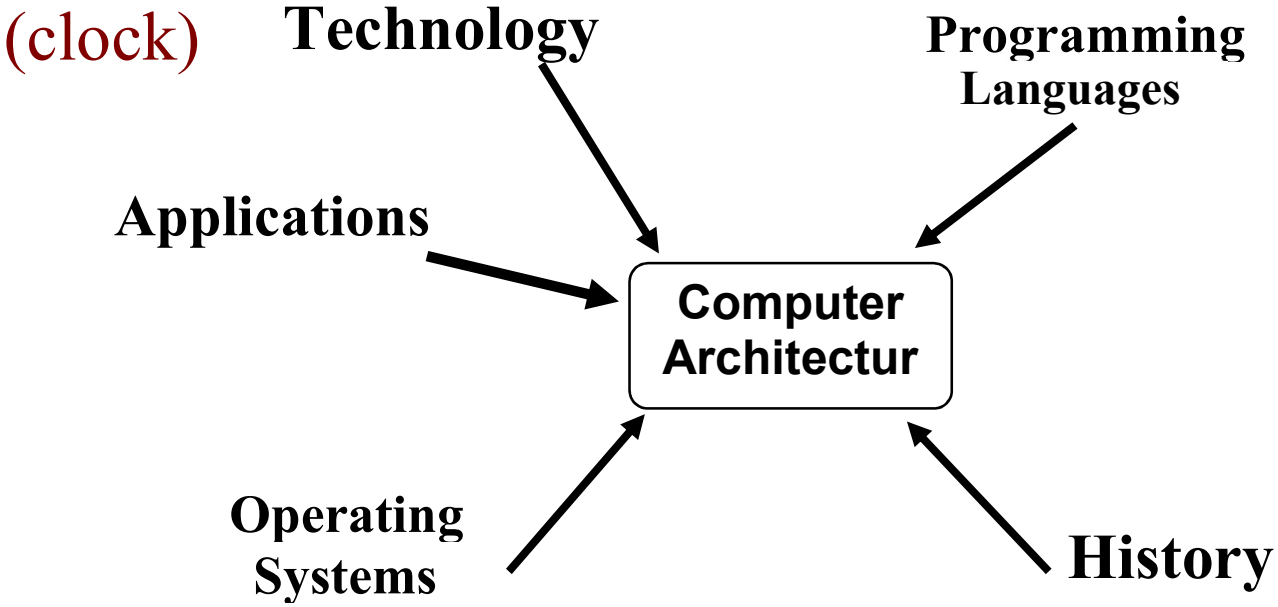
MIPS 4000

What is "Computer Architecture"

Computer Architecture =
Instruction Set Architecture +
Machine Organization

- Coordination of many *levels of abstraction*Under a rapidly changing set of forcesDesign, measurement and evaluation

A number of forces affect the computer architecture



Computer Technology

Dramatic Change

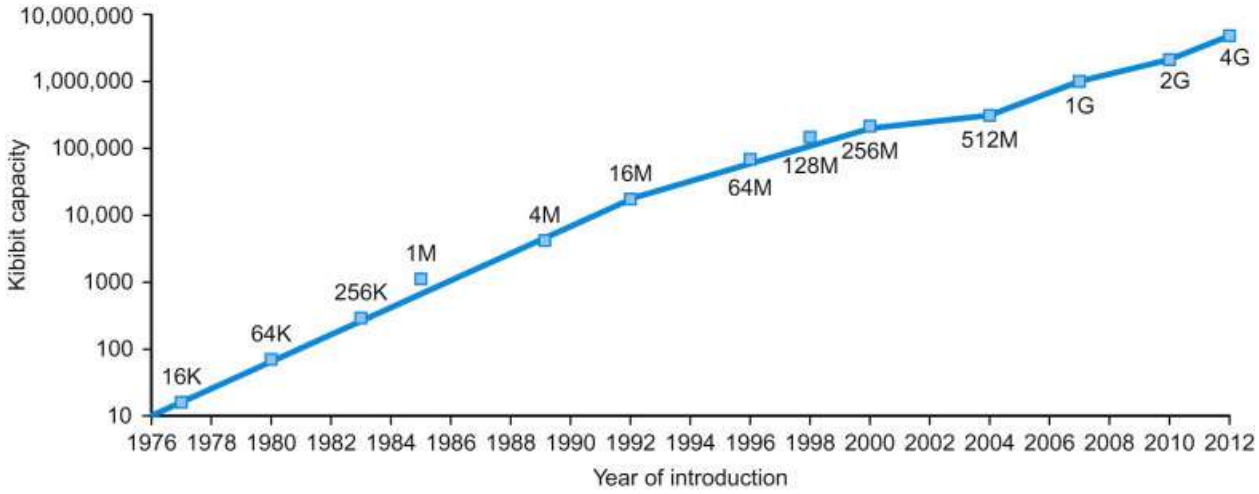
- **Processor**
  Performance: 100 fold in the last decade
  Speed: Doubled every 1.5 year

- **Memory**
  DRAM capacity: 64 fold in the last decade

  Memory speed: about 10% per year
  Cost per bit: Improves about 25% every year

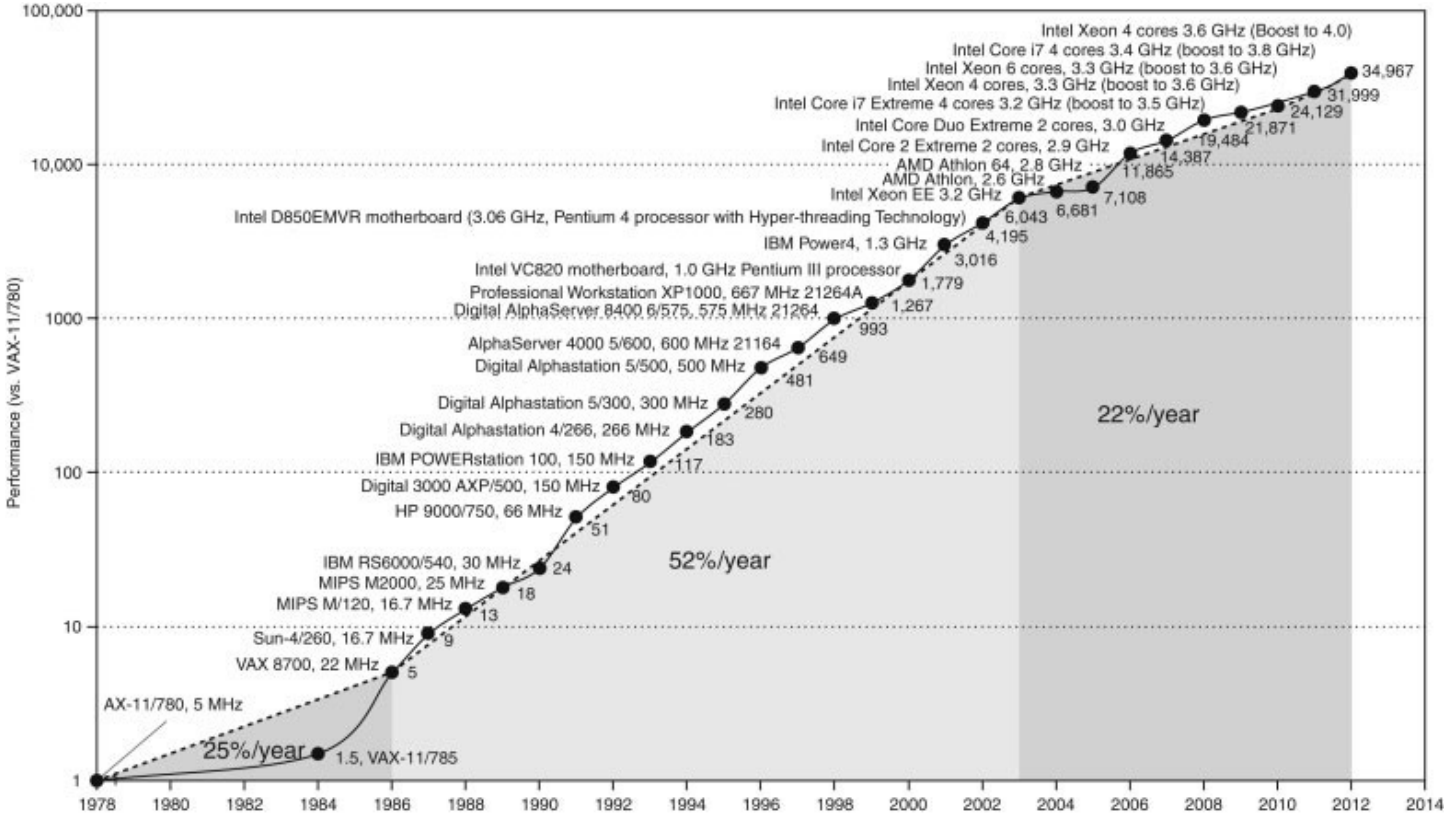
- **Disk**
  Capacity: Doubled every year
  250 fold in the last decade

Technology: Memory Capacity



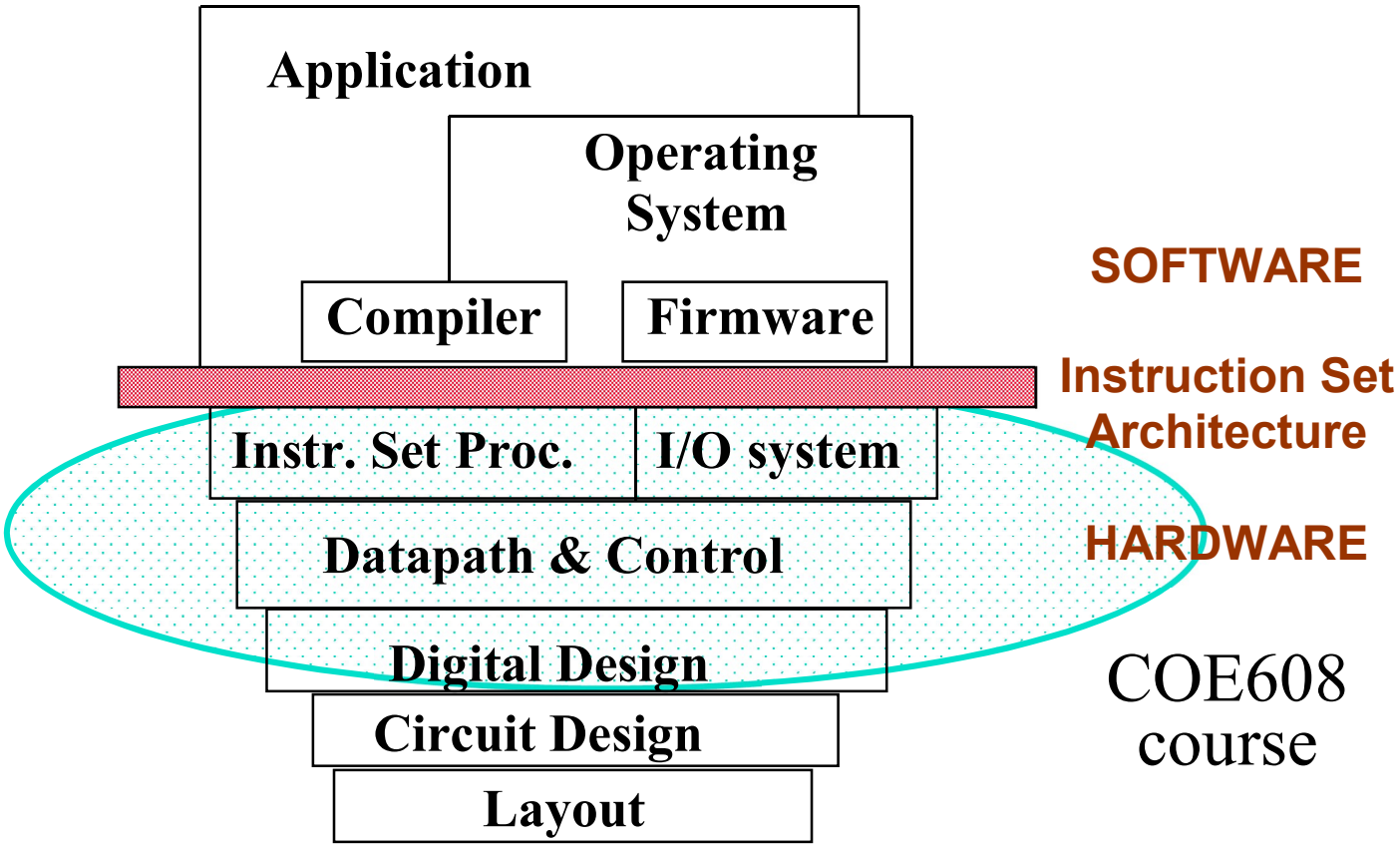
year	size (Mbit)	DRAM-Chip
1980	0.0625M	
1983	0.25M	
1986	1M	
1989	4M	
1992	16M	
1996	64M	
1998	128M	
2000	256M	
2005	512M	
2007	1G	
2010	2G	
2013	4G	

CPU Performance Growth



Computer Architecture

ISA is a subset of Computer Architecture



Computer architecture is described as

... the attributes of a [computing] system as seen by the programmer, *i.e.* the conceptual structure and functional behavior, as distinct from the organization of the data flows and controls the logic design, and physical implementation.

(Amdahl et al. 1964)

What is "Computer Architecture"

Computer Architecture =
Instruction Set Architecture +
Machine Organization

What is a computer?

Main Components:

-- Input (mouse, keyboard)
-- Output (display, printer)
-- Memory (disk drives, DRAM/SRAM, CD)
-- Network

Our primary focus:

The processor (datapath and control)
-- Implemented using millions of transistors.
-- Impossible to understand by looking at each transistor

-- We need ... CPU organization and architecture

Computer Organization

Levels of Organization

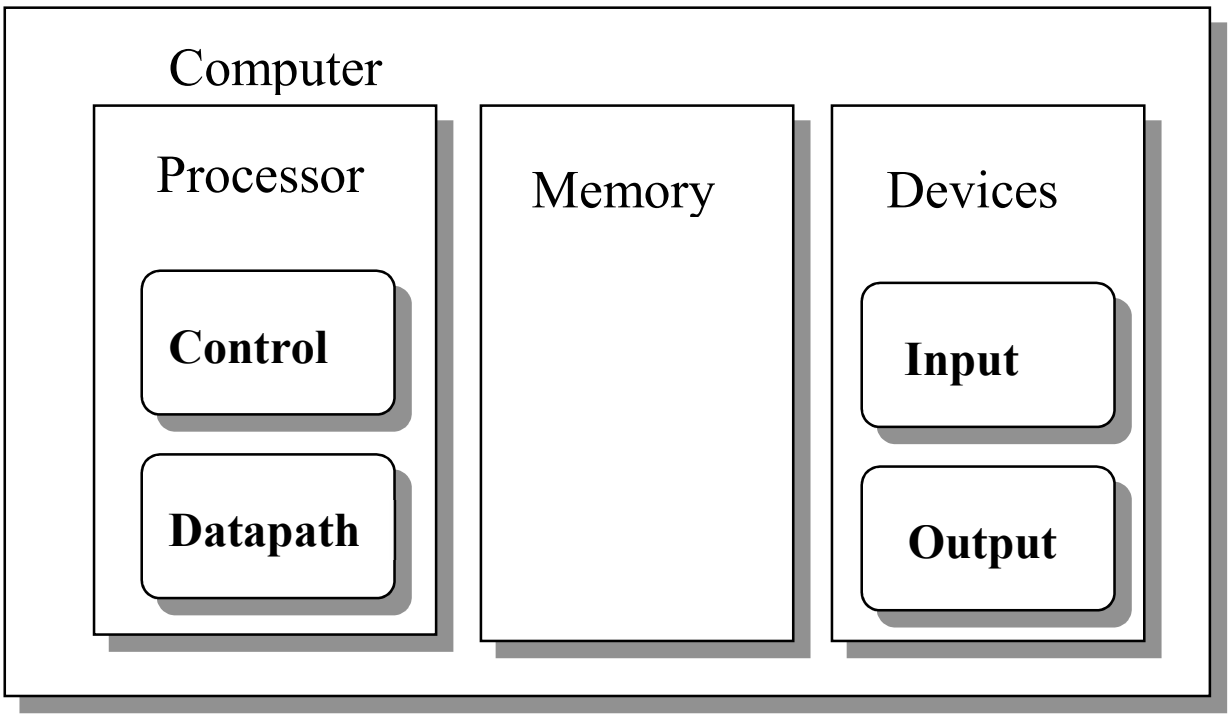
A Computer System Design Target

- Cost on processor 30%
- Cost on memory 25%

with min. memory (cache & DRAM) size

- Rest on I/O devices, power supplies, box or casing, etc.

A typical Computer System



Computer Organization

All computers consist of five components

- Processor
  - (1) datapath
  - (2) control
- (3) Memory
- (4) Input devices and (5) Output devices

Not all "memory" are created equally

- Cache: fast (expensive) memory are placed closer to the processor
- Main memory: Less expensive memory--we can have more

Input and output (I/O) devices have the most complex organization

- Wide range of speed: graphics vs. keyboard
- Wide range of requirements:
  speed, standard, cost ...
- Least amount of research (so far)

Where are we Heading?

- VHDL
  Download VHDL tutorial and documentation from the course web site
- An Instruction Set Architecture (ISA) (Chapter 2)
- Computer Arithmetic and How to build an ALU (Chapter 3)
- Performance Issues (Chapter 1)
  - Evaluating Performance
- Constructing a processor to execute CPU instructions (Chapter 4: Sections 4.1- 4.4)
  - Datapath Design
  - ASM Charts and Finite State Machines
  - Control Unit Design
- Pipelining to improve performance (Chapter 4: Sections 4.5 -- 4.8)