CPE300: Digital System Architecture and Design

Fall 2011

MW 17:30-18:45 CBC C316

Outline

- Intro to Computer Systems and Architecture
- Need for this class
- Course objectives
- Views of the general purpose machine
  - Assembly/Machine language programmer
  - Computer architect
  - Digital logic designer

The General Purpose Machine

- What is a computer?
- Merriam-Webster dictionary definition
  - **com·put·er**: one that computes; specifically : a programmable usually electronic device that can store, retrieve, and process data

- Not just the laptop or machine (multipurpose machine) on a desk
  - All around in everything
    - Cars, thermostats, washer, dryer, toys, supercomputers, Mechanical Turk?

5 Classical Computer Components

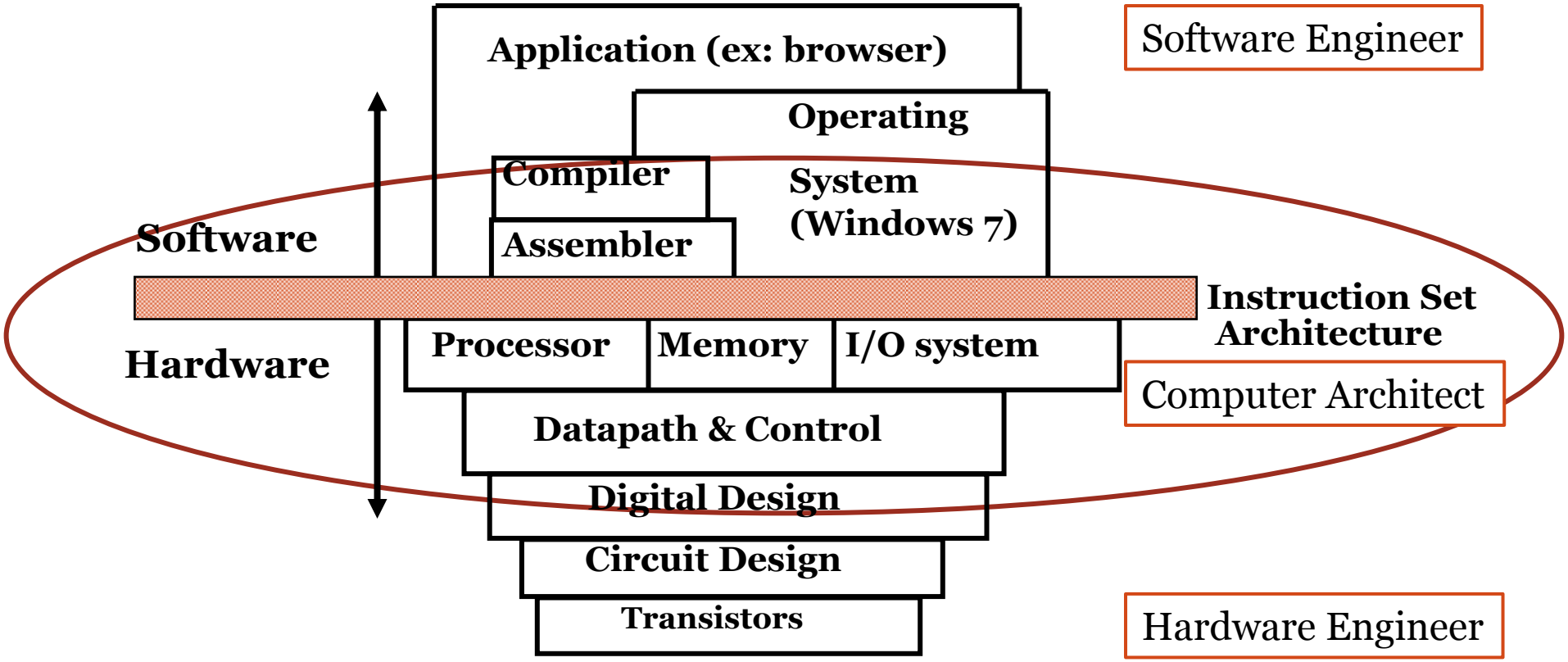
- Control – the "brain"
- Datapath – the "brawn"

} Processor

- Memory – where programs and data live when running
- Input
  - E.g. keyboard, mouse, disk
- Output
  - E.g. disk, display, printer

} Devices

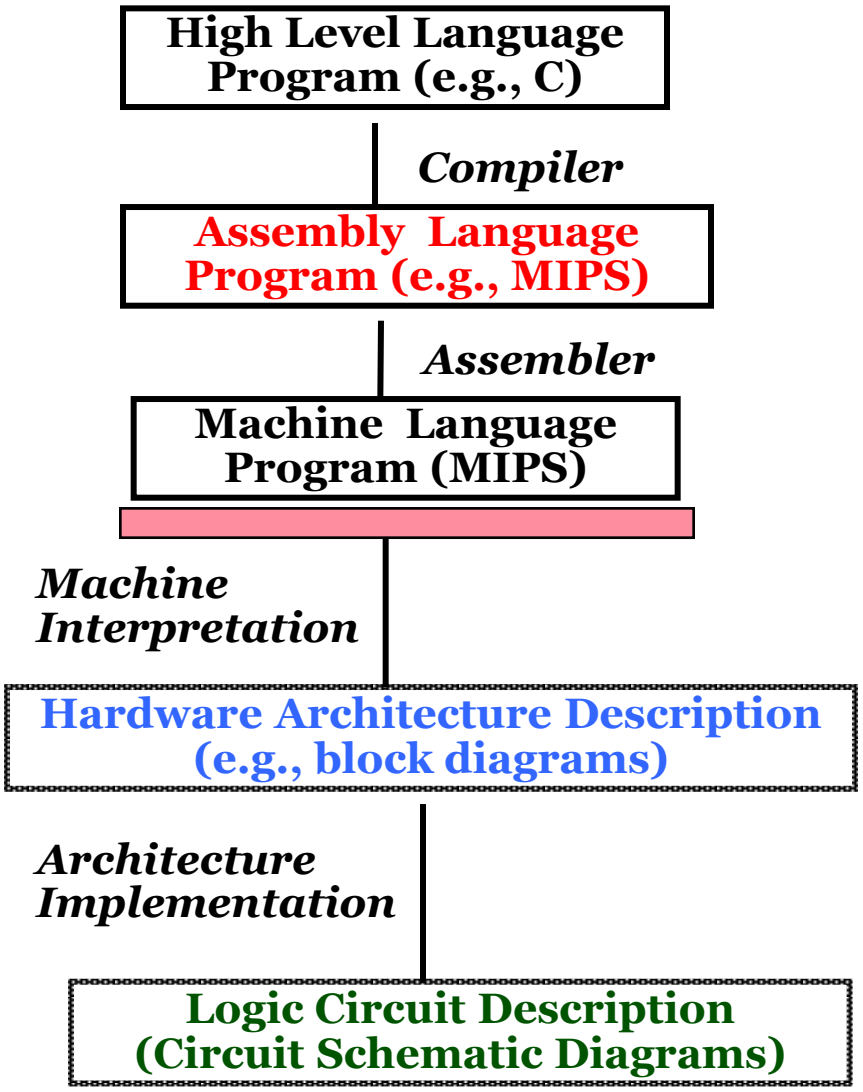
Machine Structures



- Coordination of many levels of abstraction

Slide from UC Berkeley CS61C

Levels of Representation/Interpretation

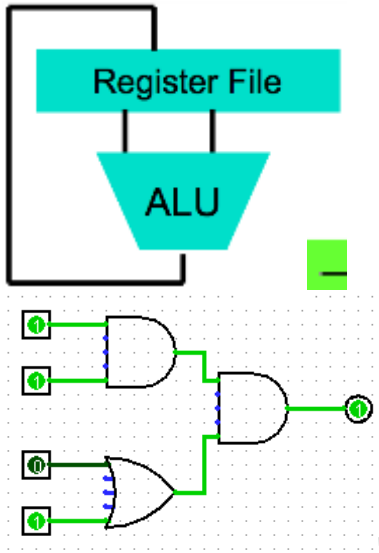



```
temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;
```

```
lw   $t0, 0($2)
lw   $t1, 4($2)
sw   $t1, 0($2)
sw   $t0, 4($2)
```

Anything can be represented as a *number*, i.e., data or instructions

```
0000 1001 1100 0110 1010 1111 0101 1000
1010 1111 0101 1000 0000 1001 1100 0110
1100 0110 1010 1111 0101 1000 0000 1001
0101 1000 0000 1001 1100 0110 1010 1111
```



Slide from UC Berkeley CS61C

Computer Systems and Architecture

- Concerned with the hardware software interface
  - It is what must be known in order to achieve the highest possible performance

- **Computer system design and architecture**: blueprint which describes requirements and basic design for parts of a computer; e.g. cpu, memory access.
  - Essentially how to go from transistors and gates to a useful application
  - What the machine looks like (computer organization) and how you talk to the machine (ISA).

Why is this class needed?

- Lower level knowledge of computer is necessary for optimized speed and specialized tasks
  - Direct device manipulation or lack of HLL

- Computers are rapidly changing
  - Moore's Law – 2x transistors/chip every 2 years
    - Greatly improved processor performance
  - Multi-core processors
  - Parallelism

Current Machine Structures

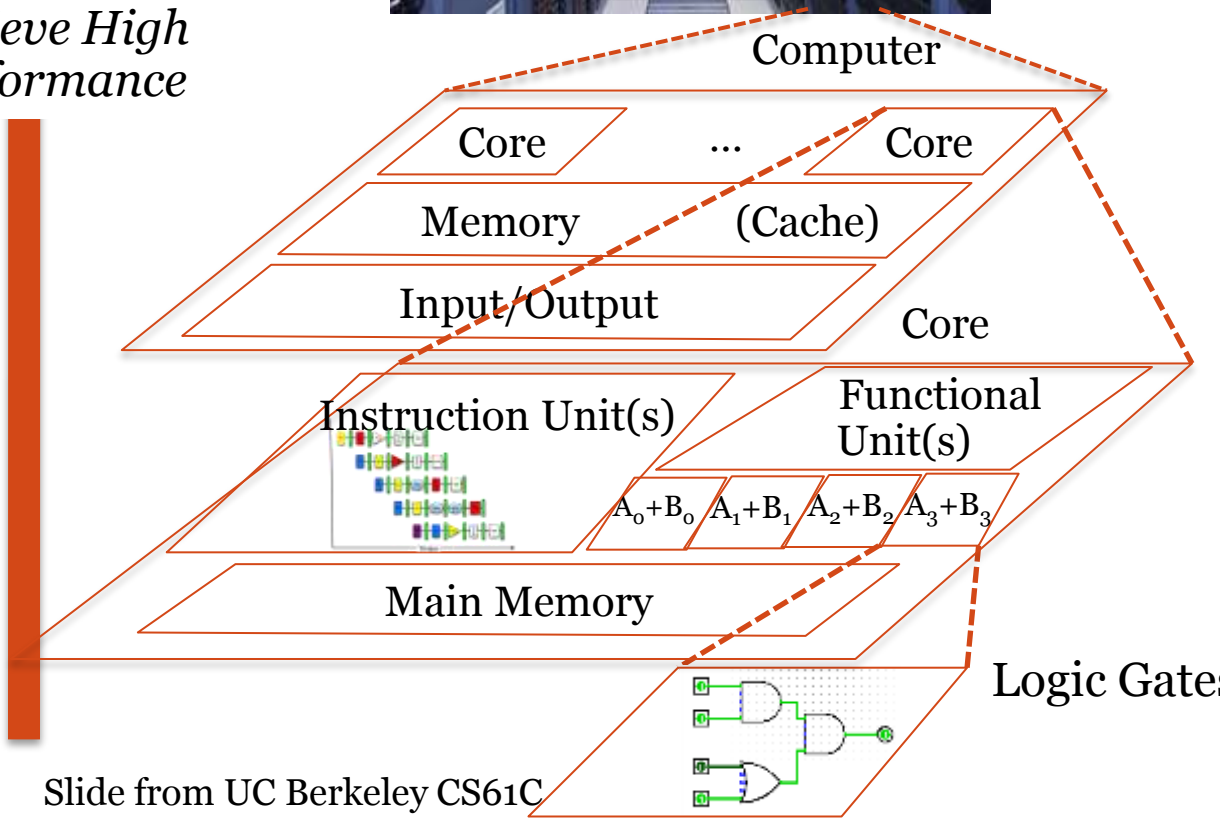
Software | *Hardware*

- Parallel Requests
  Assigned to computer
  e.g., Search "Katz"
- Parallel Threads
  Assigned to core
  e.g., Lookup, Ads
- Parallel Instructions
  >1 instruction @ one time
  e.g., 5 pipelined instructions
- Parallel Data
  >1 data item @ one time
  e.g., Add of 4 pairs of words
- Hardware descriptions
  All gates functioning in parallel at same time

Harness Parallelism & Achieve High Performance

Warehouse Scale Computer

Smart Phone



Course Goals

- Understanding Structure and Function of Digital Computer at 3 Levels

- Multiple levels of computer operation
  - Application level
  - High Level Language(s), HLL, level(s)
  - Assembly/machine language level: instruction set
  - System architecture level: subsystems & connections
  - Digital logic level: gates, memory elements, buses
  - Electronic design level
  - Semiconductor physics level

This course

- Interactions and relations between levels
  - View of machine at each level
  - Tasks and tools at each level
  - Examine design trade-offs and effect on performance

Four Views of Computer

- User View
  - Who the machine is designed for (us for a PC)
- Machine Language Programmer
  - Concerned with behavior and performance of machine when programmed at lowest level (machine language)
- Computer Architect
  - Concerned with design and performance at (sub) system levels
- Logic Designer
  - Concerned with design at the digital logic level

User View

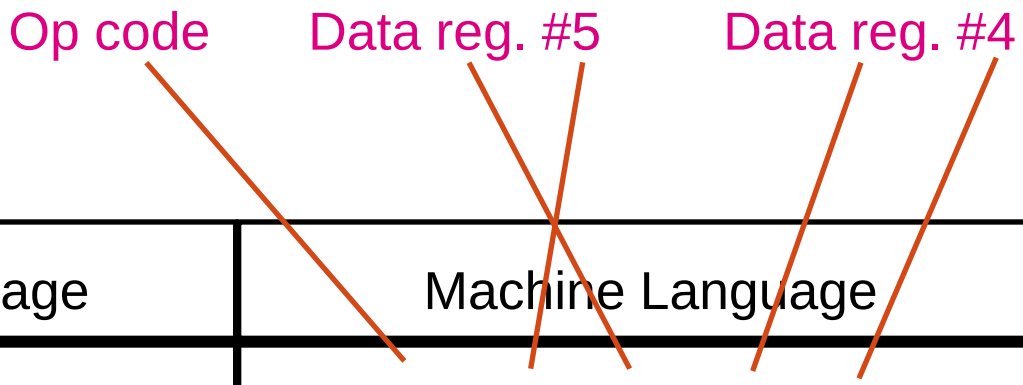
- Person employing computer to do useful work
- Internal structure of machine is almost entirely obscured
  - Operating system or application software
- Sees software, speed, storage capacity, and peripheral device functionality
- This is the common view, not the engineer's view

Machine/Assembly Programmer's View

- Machine language
  - Set of fundamental instructions the machine can execute
  - Expressed as patterns of 0's and 1's
- Assembly language
  - Alpha numeric equivalent of machine language
  - Human oriented mnemonics (human readable)
- Instruction Set Architecture (ISA)
  - Instruction set – set of all machine operations
  - Memory
  - Programmer-accessible registers

Machine and Assembly Language

- Assembler
  - Convert from assembly into machine code (native language)



MC68000 Assembly Language	Machine Language
MOVE.W D4, D5	0011 101 000 000 100
ADDI.W #9, D2	00000001 10 111 100 0000 0000 0000 1001

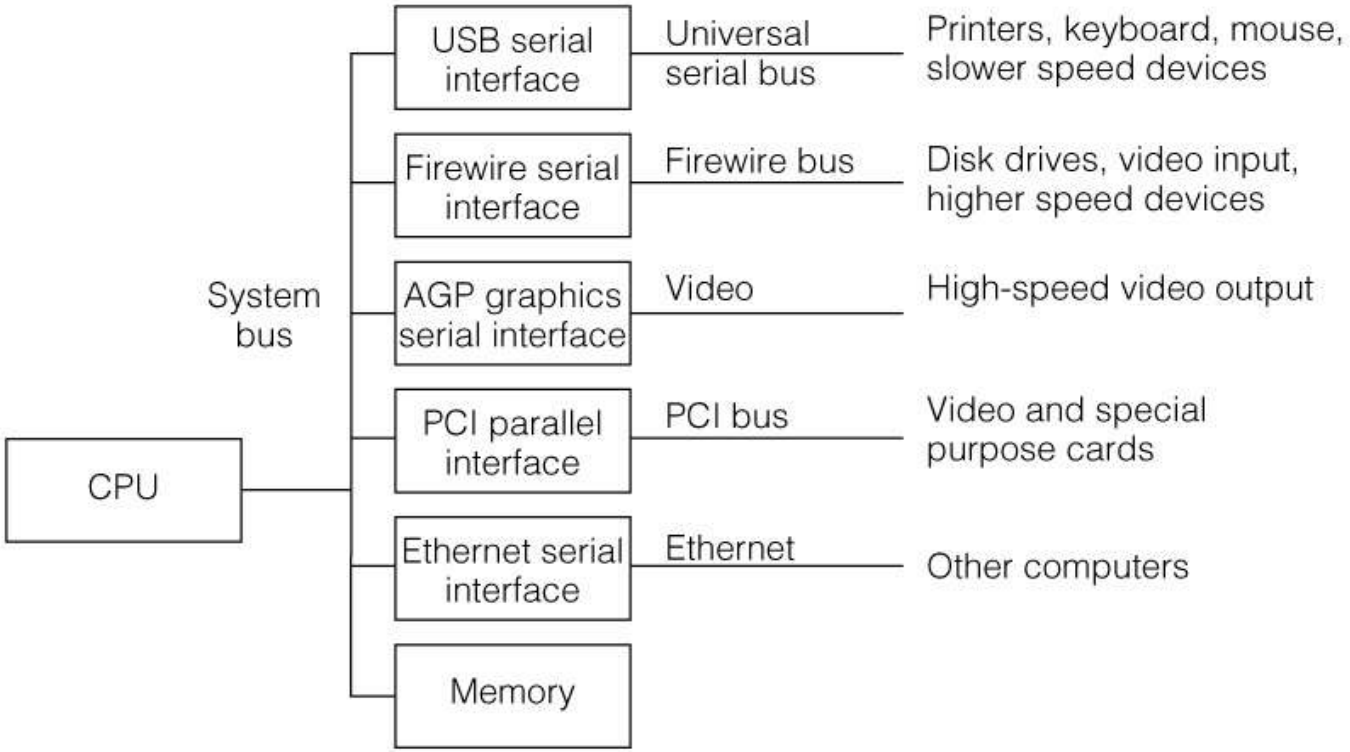
Table 1.2 Two Motorola MC68000 instructions

Computer Architect View

- Architect is concerned with design & performance
- Designs the ISA for optimum programming utility and optimum performance of implementation
- Designs the hardware for best implementation of the instructions
- Uses performance measurement tools, such as benchmark programs, to see that goals are met
- Balances performance of building blocks such as CPU, memory, I/O devices, and interconnections
- Meets performance goals at lowest cost

System Diagrams

- Machine as collection of functional units and their interconnections

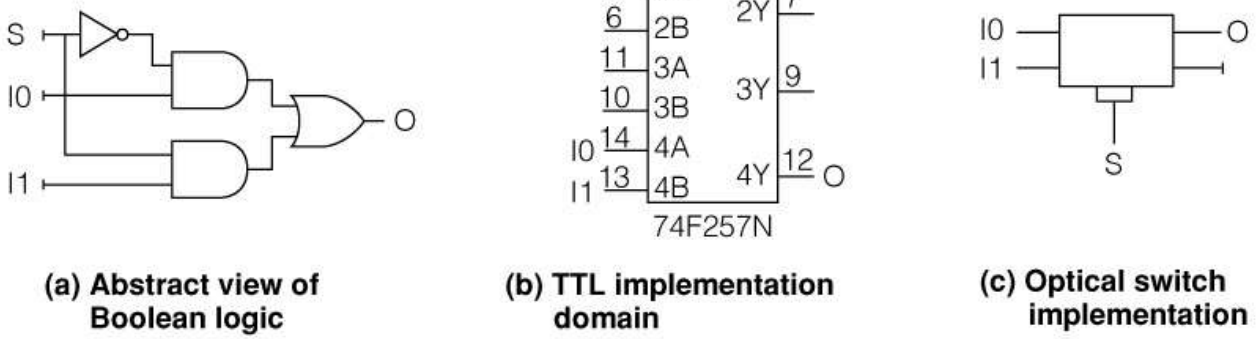


Logic Designer View

- Designs the machine at the logic gate level
- The design determines whether the architect meets cost and performance goals
- Architect and logic designer can often be the same person/team

Implementation Domains

- Domain is usually abstracted
- Computer design is complex
  - Well defined "building" blocks are used repeatedly



(a) Abstract view of Boolean logic

(b) TTL implementation domain

(c) Optical switch implementation

Concluding Remarks

- This course will study a general purpose machine at three different levels of abstraction
  - Machine/Assembly language level
  - Architecture level
  - Logic design level
- Abstraction used to build system as layers
- 5 classic components of all computers