Building Computers from Digital Circuits

CS 217

Computer Architecture

• Simplified computer architecture

- CPU
- Memory
- Network
- Audio
- Disk
- Video
SPARC Architecture

Toy Architecture (from CS126)
Circuit components

- **AND gate**: $x \& y$
- **OR gate**: $x \mid y$
- **NOT gate**: $\sim x$
- **D Flipflop**
- **Wire**
- **Wires crossing**: $n$ wires in parallel
- **Wire connection**

How do you build a computer out of these components?

SPARC Architecture

![SPARC Architecture Diagram](image)
SPARC Architecture

Addition

Arithmetic Logic Unit (ALU)

- Perform arithmetic operations
  - Boolean operations
  - Adding
  - etc.
1-Bit Arithmetic Logic Unit

SPARC Architecture
**Shifter**

- Shift bits to right or left

*Figure 4-4*

---

**1-Bit Shifter**

- Shift bits one spot to right or left

*Figure 3-17*
SPARC Architecture

- Bank of registers connected to data buses
8-Bit Register

8-Bit Register

Tannenbaum Fig 4-2 15

Tannenbaum Fig 4-2 16
Storage Hierarchy

- Registers
  32-128, 1-5ns access time (CPU cycle time)

- Cache
  1KB – 4MB, 20-100ns (multiple levels)

- Memory
  64MB – 2GB, 200ns

- Disk
  1GB – 100GB, 10ms

- Long-term Storage
  1TB, 1-10s

SPARC Architecture

[Diagram of SPARC Architecture]

Paul Fig 1.5
Memory

• Just string flip flops together?

Addressable Memory

N-bit input
l_2 l_1 l_0

K-bit address
A_1 A_0

demux

memory cells

O_2 O_1 O_0

N-bit output

cs Chip select
rd Read/write
oe Output enable
**Demultiplexer**

K-bit address

Exactly one of these output is set

N-bit input

K-bit address

Chip select

Read/write

Output enable

N-bit output

Tannenbaum Fig 3-14 23

Tannenbaum Fig 3-28
Multiplexer

Any/all of these output are set

N-bit input

K-bit address

Chip select
Read/write
Output enable

Output

N-bit output

Tannenbaum Fig 3-12

Tannenbaum Fig 3-28
SPARC Architecture

Design

- Possible strategies
  - Bottom-up
  - Top-down
Summary

• Gates -> Computer
  ◦ Components
  ◦ Abstraction
  ◦ Design

• Next time
  ◦ Instruction set
  ◦ Instruction processing