

Chapter 1

Welcome Aboard

Based on slides © McGraw-Hill
Additional material © 2004/2005 Lewis/Martin

Recurring Themes

Abstraction

- Allows us to manage seemingly insurmountable complexity

Hardware v. Software

- Separation of hardware and software is artificial

CIT 593

1-2

Recurring Theme #1: Abstraction

Abstraction hides details

Examples

- Using a bread machine/microwave
- Going to 30th Street Station from Walnut St.

Computing examples

- Java methods
- C functions
- LC-3 instructions
- Logical gates
- Transistors

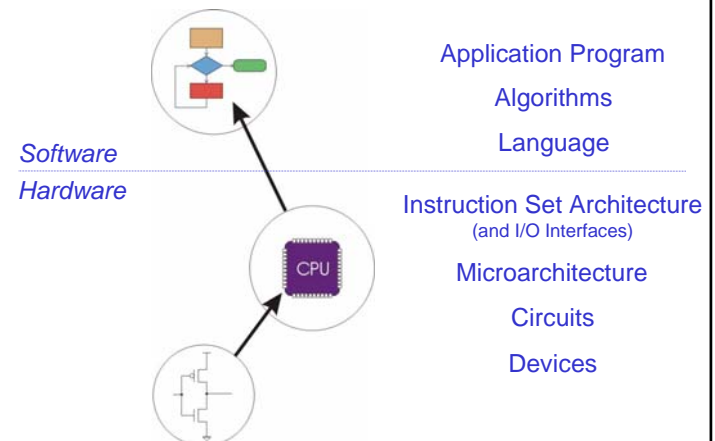
Bottom line

- Often best to operate at highest level of abstraction
- Dangerous to completely ignore lower levels of abstraction

CIT 593

1-3

Computer System: Layers of Abstraction



CIT 593

1-4

Recurring Theme #2: Hardware vs. Software

Artificial Divide

- Hardware: physical computer and specifications associated w/ it
- Software: Operating System, application software

Conventional Approach

- We really care about *computation*
- Hardware best understood by those who know software
- Software best understood by those who know hardware

Greatness arises from blurring the HW/SW line

- a computing system works best when both capabilities and limitations are accounted.
- Example: Special video graphics processor

CIT 593

1-5

Very Big Ideas

These are core to what computing is all about!

Universality

- All computers can compute the same thing*

Layered Abstraction

- We can build very complex systems from simple components

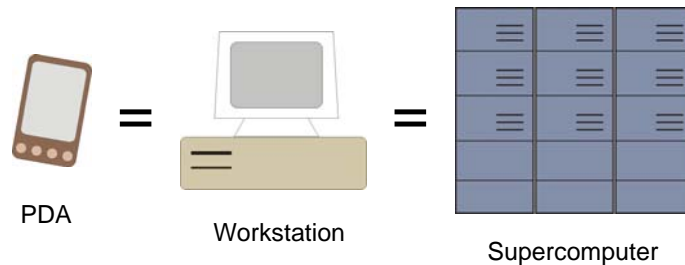
CIT 593

1-6

Big Idea #1: Universal Computing Device

All computers *can* compute exactly the same things*

*given enough time and memory



CIT 593

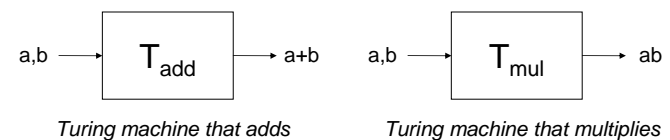
1-7

Turing Machine

Mathematical model of a device that can perform any computation – Alan Turing (1937)

- Ability to read/write symbols on an infinite “tape”
- State transitions, based on current state and symbol

Every computation can be performed by some Turing machine. (*Turing's thesis*)



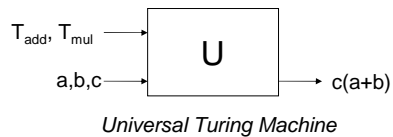
CIT 593

1-8

Universal Turing Machine

Turing described a Turing machine that could implement all other Turing machines

- Inputs: data, plus a description of computation (Turing machine)



U is programmable – so is a computer!

- Instructions are part of the input data
- A computer can emulate a Universal Turing Machine, and vice versa

Therefore, a computer is a universal computing device!

CIT 593

1-9

From Theory to Practice

In theory

- Computers can compute anything that's possible to compute
- Given enough *memory* and *time*

In practice

- Solving *real* problems requires computing under constraints
- Time
 - Weather forecast, next frame of animation, ...
- Cost
 - Cell phone, automotive engine controller, ...
- Power
 - Cell phone, handheld video game, ...

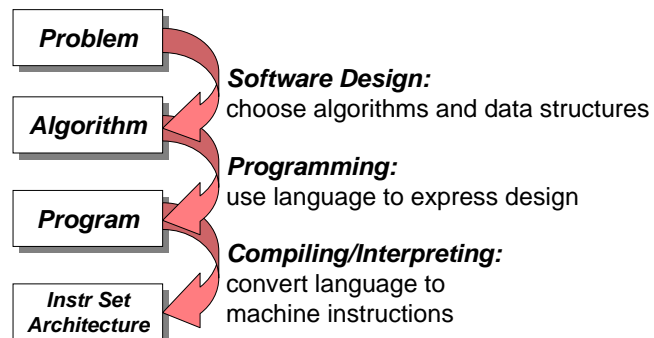
CIT 593

1-10

Big Idea #2: Layered Abstraction

How do we solve a problem using a computer?

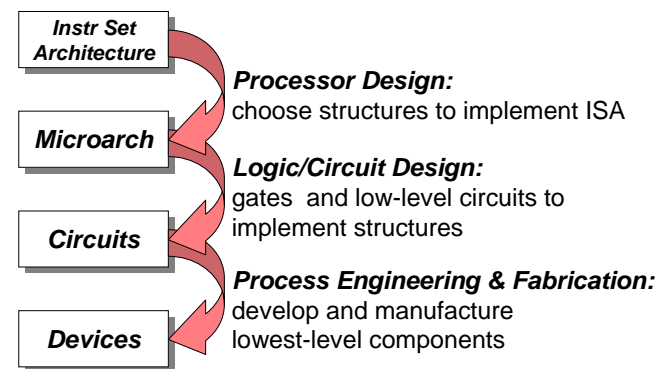
- Systematic sequence of transformations between layers of abstraction. . .



CIT 593

1-11

Deeper and Deeper...



CIT 593

1-12

Descriptions of Each Level

Problem Statement

- Stated using "natural language"

Algorithm

- Step-by-step procedure, guaranteed to finish
- Definiteness, effective computability, finiteness

Program

- Express the algorithm using a computer language
- High-level language (we will learn C language for the course)

Instruction Set Architecture (ISA)

- Specifies the set of instructions the computer can perform
 - what kinds of operations ? (e.g. add or multiply)
 - what data is needed for each operation? (integer or floating point)
- We will learn LC3 Assembly Language for the course

CIT 593

1-13

Descriptions of Each Level (cont.)

Microarchitecture

- Detailed organization of a processor implementation
- Different implementations of a single ISA

Logic Circuits

- Combine basic operations to realize microarchitecture
- Many different ways to implement a single function (e.g., addition)

Devices

- Properties of materials, manufacturability

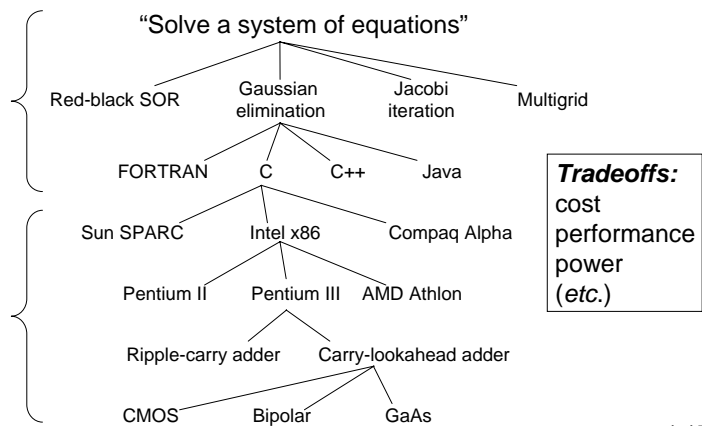
CIT 595 - Digital System Organization & Design

- Covers the Microarchitecture, Logic Circuits and Devices in detail

CIT 593

1-14

Many Choices at Each Level



CIT 593

1-15

Next Time

Lecture

- Computing - Unix/Linux
- Chapter 2: Bits and Bytes

Reading

- History of Computers
- Chapter 1 - Welcome Aboard (Yale & Patt text)

Lookout for an homework assignment next week

CIT 593

1-16