WHEN DOES COMPUTER ARCHITECTURE
EDUCATION BEGIN?
ASSEMBLY LANGUAGE AS COMPUTER ARCHITECTURE

JAMES R. GOODMAN
E-MAIL: goodman@cs.wisc.edu
HTTP://WWW-CS.WISC.EDU/~GOODMAN

DEPARTMENT OF COMPUTER SCIENCES
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
UNIVERSITY OF WISCONSIN-MADISON

FEBRUARY 3, 1996
WHAT role does ASSEMBLY LANGUAGE play in a COMPUTER SCIENCE CURRICULUM?

WHAT role does ASSEMBLY LANGUAGE play in a COMPUTER ARCHITECTURE CURRICULUM?
Department of Computer Sciences

- Mark Hill
- Guri Sohi
- David Wood
- Jim Goodman

Department of Electrical and Computer Engineering

Includes Computer Engineering Option

- Jim Smith
- Yuhen Hu
- Chuck Kime
- Kewal Saluja
COURSES

352 Digital Systems Fundamentals

CS/ECE 354 Machine Organization and Programming

CS 367 Introduction to Data Structures

Other CS Systems courses

Other ECE courses

CS/ECE 755 VLSI Design

CS/ECE 552 Intro. to Computer Architecture

CS 537 Intro. to Operating Systems

CS/ECE 752 Advanced Computer Architecture I

CS/ECE 757 Advanced Computer Architecture II
CS/ECE 354

- Prerequisite: familiarity with one HLL.
- An introduction to computer organization using assembly and machine language.
  - Representation of information
  - Computer arithmetic
  - Instruction sets
  - I/O
  - Interrupts

Projects involve detailed study and use of a specific computer hardware and software system.
PROBLEMS

• Students come from a variety of places
  ✦ Future Computer Science majors
  ✦ Future Computer Engineering majors
  ✦ Business majors, Physics majors, etc.

• Students have a variety of backgrounds
  ✦ Some have had 352 Digital Systems Fundamentals
  ✦ Some have had 367 Data Structures
  ✦ Some have had neither

• Lots of new ideas are needed all at once
  ✦ the instruction cycle,
  ✦ number representation
  ✦ memory organization
  ✦ addressing modes
  ✦ ...

© 1996 by J. Goodman
GOALS OF 354

- Understand Hardware/Software Interface
- *Not* to become proficient programmers
- Prepare non-architecture students for advanced courses

FIRST COMPUTER ARCHITECTURE COURSE
IMPLICATIONS OF NEW DIRECTION

• Focus is on Ideas, not Details

• Teach Assembly Language Top-down

• Use abstract computer

• Selectively ignore grungy details of real architectures

ASSEMBLY LANGUAGE IS A TOOL, NOT A SKILL TO BE MASTERED
Our Implementation

• SAL, MAL, TAL, based on MIPS RISC Architecture
• Simulator SPIM/SAL implements SAL, MAL, TAL.
• Discussion of alternative architectures
• Strong emphasis on representation and arithmetic
ADVANTAGES

• Start with familiar, abstract model: HLL
• Focus is on simple machine having only needed detail
• Introduce features when convenient
Summary

- Assembly language is the tool, not the goal
- This is a Computer Architecture course
- Course (and books) are mis-named