EN164: Design of Computing Systems
Lecture 12: Processor / Single-Cycle Design 1

Professor Sherief Reda
http://scale.engin.brown.edu
Electrical Sciences and Computer Engineering
School of Engineering
Brown University
Spring 2011

[ material from Patterson & Hennessy, 4th ed and Harris 1st ed ]
Processor organization (microarchitecture)

- Multiple implementations for a single architecture:
  - Single-cycle
    - Each instruction executes in a single cycle
  - Multi-cycle
    - Each instruction is broken up into a series of shorter steps
  - Pipelined
    - Each instruction is broken up into a series of steps
    - Multiple instructions execute at once.
Introduction

- CPU performance factors
  - Instruction count
    - Determined by ISA and compiler
  - CPI and Cycle time
    - Determined by CPU hardware
- We will examine two MIPS implementations
  - A simplified version
  - A more realistic pipelined version
- Simple subset, shows most aspects
  - Memory reference: lw, sw
  - Arithmetic/logical: add, sub, and, or, slt
  - Control transfer: beq, j
Architectural state

- Determines everything about a processor:
  - PC
  - 32 registers
  - Memory
Single-Cycle MIPS Processor

- Datapath
- Control

Flowchart:

1. Fetch instruction @ PC
2. Decode instruction
3. Fetch Operands
4. Execute instruction
5. Store result
6. Update PC
Single-Cycle Datapath: \( \text{lw} \) fetch

- First consider executing \( \text{lw} \)
- **STEP 1**: Fetch instruction

![Instruction Memory Diagram]

![Register File Diagram]

![Data Memory Diagram]
Single-Cycle Datapath: \( lw \) register read

- **STEP 2:** Read source operands from register file
Single-Cycle Datapath: $lw$ immediate

- **STEP 3:** Sign-extend the immediate
Single-Cycle Datapath: \( \texttt{lw} \) address

- **STEP 4:** Compute the memory address
Single-Cycle Datapath: \( \downarrow \) memory read

- **STEP 5:** Read data from memory and write it back to register file
Single-Cycle Datapath: $lw$ PC increment

- **STEP 6:** Determine the address of the next instruction