

## When / Where

Monday, 9 December 2013, 10:00-12:00 CST

1110<sub>10</sub> Patrick F. Taylor Hall (Here)

## Conditions

Closed Book, Closed Notes

Bring one sheet of notes (both sides), 216 mm × 280 mm.

No use of communication devices.

## Format

Several problems, short-answer questions.

## Resources

Lecture “slides” used in class: <http://www.ece.lsu.edu/ee3755/ln.html>

Study Guides: Carry-Lookahead, Synthesis

Solved tests and homework: <http://www.ece.lsu.edu/ee3755/prev.html>

## Study Recommendations

Study this semester's homework assignments. Similar problems will probably appear on the exam.

Solve Old Problems—memorizing solutions **is not the same** as solving.

Following and understanding solutions **is not the same as** solving.

Use the solutions for brief hints and to check your own solutions.

## Previous Exams

Be sure to look at previous midterm and final exams.

## Verilog—Key Skills

Given a design in one form, write design in another:

Explicit Structural

Implicit Structural

Synthesizable Behavioral

Logic Diagram

## Verilog, The Simulation Language

### Value Set

Know what **x** and **z** mean and how to generate and use them.

### Modules and Instantiation

### Continuous Assignment and Expressions

### Implicit v. Explicit Structural Descriptions

### Behavioral Code

## Verilog, Design Flow and Synthesis

HDL Terminology

*Brown & Vranesic, Section 2.9*

Design Flow

Inference and Libraries

Combinational Synthesis

Sequential Synthesis

## Computer Arithmetic

Integer Adders

Carry Lookahead

*Brown & Vranesic, Section 5.4*

Hierarchical Carry Lookahead Adders

## Integer Multipliers

*Patterson & Hennessy, 4th Ed., Section 3.3*

Simple and streamlined multipliers.

Ordinary radix- $n$  multiplier and Booth recoding multiplier.

## Integer Dividers

*Patterson & Hennessy, 4th Ed., Section 3.4*



## FP Arithmetic

*Patterson & Hennessy, 4th Ed., Section 3.5*

Working with numbers in binary scientific notation.

IEEE 754 Format

FP Adder

## MIPS

### Definitions

*Patterson & Hennessy, 4th Ed., Sections 2.1, 2.2, 2.12*

### Integer Arithmetic and Logical Instructions

*Patterson & Hennessy, 4th Ed., In chapter 2*

### Control Transfer Instructions

*Patterson & Hennessy, 4th Ed., In chapter 2*

### Instruction Coding

*Patterson & Hennessy, 4th Ed., Section 2.5*

### Pseudo Instructions

*Patterson & Hennessy, 4th Ed., Section 2.12*

## MIPS Programming Skills

Write programs.

Read programs.

Understand instruction formats.

## MIPS Implementations

MIPS Functional (Single-Cycle) Simulator

MIPS Very Simple Implementation

## MIPS Functional Simulator

Executes each instruction in a single cycle.

Because of single-cycle execution needs extra hardware:

Two memory ports (one for instruction fetch, one for loads and stores).

A separate adder for arithmetic instructions and branch targets.

## MIPS Very Simple Implementation

Executes instructions in multiple cycles.

Same hardware can be re-used, such as memory port and ALU.

## MIPS Implementation Skills

Implement a new instruction.

Determine what an instruction does by looking at Verilog.

Sketch hardware synthesized from Verilog description.