

URL: <http://www.ece.lsu.edu/ee3755>

Offered by:

David M. Koppelman

Room 3191 P. Taylor Hall

578-5482, koppel@ece.lsu.edu, <http://www.ece.lsu.edu/koppel>

Tentative office hours: M-F 14:00-15:00.

Should already know...

... how to design with logic.

Will learn...

... how to design a rudimentary computer.

Prerequisite by Course

EE 2730 (Digital Logic II) or equivalent.

Prerequisite by Topic

Logic design.

Binary number representation and arithmetic.

Programming of some kind.

Computer Organization Text

“Computer organization & design,” David A. Patterson & John L. Hennessy (Required).

Verilog Texts

(Optional) Brown & Vranesic, “Fundamentals of Digital Logic with Verilog Design.” (The text used in EE 2720 and 2730.)

(Optional) Ciletti, “Advanced Digital Design with the Verilog HDL.” Only buy this if you are rich or if you want to learn Verilog very well.

Daniel C. Hyde, “Bucknell Handbook on Verilog HDL.” Free, linked to Web site.

Software manuals, Linked to Web site.

Graded Material

40% Midterm Exam

40% Final Exam

20% Homework

Lowest homework grade dropped.

Hardware Description using Verilog

A commonly used language for designing digital hardware.

Computer Arithmetic

How computers add, subtract, multiply, and divide.

Integer and floating-point.

Will use Verilog to design circuits.

RISC Processor Programming

The basics of programming an easy-to-program processor, MIPS.

Computer Organization

Design hardware to execute MIPS programs.

This material will be continued in EE 4720.

Digital Design Hardware Description Languages (HDLs)

Designs for digital parts captured in a *HDL description*.

Popular HDLs: *Verilog*, *VHDL*.

An HDL description is fed to:

A *simulator*, to see what the design does.

(Whether it does what it's supposed to do.)

A *synthesis program*, to prepare the design for fabrication or downloading.

Chips may be fabricated using the synthesized description.

The synthesized description might be downloaded to a special chip (FPGA).

HDL Material Covered in Course

Language used: Verilog

Subset of the language covered.

Enough to implement processor arithmetic units and other datapath elements.

HDL Topics Covered

Writing *structural descriptions* of hardware.

Writing simple *behavioral descriptions* of hardware.

Relationship between descriptions and synthesized hardware.

HDL Topics Not Covered

Less-common structural elements and *delay specifiers*.

Much event- and delay-related behavioral code.

Will not know enough to write a good *testbench*.

Thorough HDL treatment may be given in other courses.

HDL Software Used

Host System

ECE Linux systems.

Simulator

Cadence Incisive Enterprise Simulator

Synthesis

Cadence Encounter RTL Compiler

Computer Arithmetic Topics

Integer arithmetic algorithms. (Mostly review).

Integer adder implementations.

ALU implementation.

Basic integer multiplication and division implementations.

High-speed integer multiplier implementation.

Floating-point representations.

Floating-point arithmetic algorithms.

Floating-point adder implementation.

RISC Processor

A type of processor that became popular in 80's.

RISCs simple to program and to implement (design hardware for).

Simplicity allows high-speed implementation.

Starting in 80's all new major general-purpose processors were RISCs.

RISC Families:

MIPS: Ownership: Independent, then Silicon Graphics, now ??.

PA (Precision Architecture): Ownership: Hewlett-Packard.

SPARC: Ownership: Oracle / SPARC International.

POWER, PowerPC: Ownership: IBM, Apple/IBM/Motorola.

Alpha: Ownership: DEC, Compaq, ??

MIPS used in Patterson & Hennessy text (and so used in class).

Covered for the MIPS processor.

MIPS Processor organization:

What machine-language programs can access.

What machine-language instructions can do.

MIPS Programming

Subset of instructions covered.

Processors Covered in Other Courses

EE 3750 /3751: IA-32 (Intel 80x86, Pentium)

EE 4720: MIPS, SPARC, some IA-64 and other architectures.

Building a “Toy” MIPS Processors

Use Verilog.

Topics

Functional Simulator: Simple Design, Inefficient Hardware

Multicycle Hardwired Control

Material continued in EE 4720.