CPE/EEE 064 INTRODUCTION TO LOGIC DESIGN

Required Course

2002 – 2004 Catalog Data: EEE 064: Introduction to Logic Design. Covers the following topics: logic gates, binary number system, conversion between number systems, Boolean algebra, Karnaugh maps, combinational logic, digital logic design, flip-flops, programmable logic devices (PLDs), counters, registers, memories, state machines, designing combinational logic and state machines into PLDs, and basic computer architecture. Lab emphasizes the use of software equation entry design tools, the use of a schematic entry, and the use of a logic simulation design tool. Lab assignments are design-oriented. Prerequisites: CSc 015 or CSc 025. Cross-listed as CpE 064; only one may be counted for credit. 4 units.


Additional Support Material: Introduction to Logic Design, Laboratory Manual by Dr. Becker, Version 8 CSUS Foundation, HDL design package, Xilinx Corporation, ISE 3

Course Objectives:

1. Provide a comprehensive course in combinational logic design and sequential design for lower division students.
2. Stimulate and encourage lower division students to remain engineering majors (electrical or computer).
3. Expose students to a modern HDL design tool for use on programmable logic devices.
4. Introduce students to computer architecture.
5. Prepare Computer Engineering majors for advanced courses in logic, architecture, and design.

Prerequisites by Topic: Knowledge of a structured programming language (CSC 015 or CSC 025)

Topics Covered/Class Schedule/Evaluation:

1. Logic: binary number system, logic gates, Boolean algebra, minimization, K-maps, design techniques
2. Sequential: latches, flip-flops, registers, counters, state machine theory, state machine design, Moore and Mealy
3. Modern devices and design: simple PLDs, HDL (Verilog), simulation, download, tri-state
4. Functionality: multiplexors, coders, decoders, adders, displays, ALU
5. Architecture: memory, datapath, basic CPU architecture

Course Outline:

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
<th>Text Chapter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Logic gates, binary numbers, Boolean algebra</td>
<td>30-33, 90-100</td>
</tr>
<tr>
<td>2</td>
<td>Minimization, K-maps, design</td>
<td>reread 90-100, 112-129, 480-487</td>
</tr>
<tr>
<td>3</td>
<td>Design techniques and examples</td>
<td>131-141</td>
</tr>
<tr>
<td>4</td>
<td>Programmable logic device (logic only), Verilog</td>
<td>144-158, Xilinx datasheet</td>
</tr>
<tr>
<td>5</td>
<td>Latches, flip-flops, registers</td>
<td>198-209</td>
</tr>
<tr>
<td>6</td>
<td>Introduction to state machine</td>
<td>161-172, 182-186, 195-198</td>
</tr>
<tr>
<td>7</td>
<td>Review and midterm exam</td>
<td>reread 198-209</td>
</tr>
<tr>
<td>8</td>
<td>State machine design, Verilog</td>
<td>209-214, 216-218, 221-223</td>
</tr>
<tr>
<td>9</td>
<td>State machine continued, Moore and Mealy</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Decoders, drivers, displays, multiplexors, tri-state</td>
<td>73, 241-264</td>
</tr>
<tr>
<td>11</td>
<td>Arithmetic operations, signed numbers, adders, ALU</td>
<td>277-305</td>
</tr>
<tr>
<td>12</td>
<td>Review and midterm exam</td>
<td></td>
</tr>
</tbody>
</table>
13 Memory, datapath, read/write 339-347, 362-370
14 Basic computer architecture, introduce pipeline 398-409
15 Computer system, I/O 436-451, 458-464
Exam Week Final exam

Laboratory Schedule:

1. Introduction to logic gates and two simple designs (3 weeks).
2. Combinational logic design (4-bit adder and a 3-bit comparator) (2 weeks).
3. Sequential logic, latches, flip-flops, shift registers, and counters (3 weeks).
4. Introductory state machines (2 weeks).
5. State machine PLD, decoder PLD, driver and display (2 weeks).
6. Architecture, small CPU, controller (2 weeks).

Contribution of Course to Meeting the Professional Component:

- Course develops Boolean algebra, minimization techniques, and simulation skills.
- Laboratory experiments are one-third basic logic science and two-thirds design (state machines and small CPU).
- Course uses skills from structured programming to design solutions using Verilog.
- Science and Design Content Distribution: Science – 1.5 units or 37.50%; Design – 2.5 units or 62.50%.

Relationship of Course to Program Outcomes:

- #3. Problem solving: Boolean algebra, minimization, emphasis on logic design are applied in laboratory and included on exams.
- #8. Experimental work: Laboratory requires students to interpret logic functionality, signal levels, and high-impedance.
- #9. Integration of knowledge to solve design problems: Two thirds of lab experiences are designs focused on logic, sequential, state machine, and CPU.
- #11. Written communication: Basic writing skills in writing a lab report are required.
- #7. Use of contemporary tools for analysis and design: Two thirds of the laboratory is dependent upon the use of Verilog and Xilinx’s Foundation tool (to design, choose pins for PLD, simulate, and download design).

Course Coordinator: Ronald W. Becker  Date: May 15, 2003