COURSE DESCRIPTION

Dept., Number  CSC 137  Course Title  Computer Organization
Semester hours  4  Course Coordinator  Nikrouz Faroughi

URL (if any):  http://gaia.ecs.csus.edu/~faroughi/

Catalog Description

An introduction to computer organization and architecture. Topics include combinational devices, sequential and synchronized circuits, memory, bus structures, input/output and interrupt structures, CPU organization, control unit design and organization, and an introduction to modern processor and memory features. Projects include construction of a complete simple system using a schematic simulator and HDL. Prerequisite: At least a C- grade in CSC 28, CSC 35 and CSC 130, and full CSC, CPE, or MATH/CSC major status.

Textbook


References


Course Goals

1. Provide students with a thorough understanding of the organization of computer systems including CPU, memory, and I/O architectures.
2. Students will design a working model of a simple processor and memory system starting from basic logic gates.
3. Students will acquire a basic understanding of the architectural characteristics of modern computer systems, including pipelined and superscalar designs, RISC and CISC concepts, and modern memory organization.

Prerequisites by Topic

Thorough understanding of:
- Unsigned binary integers: representation, conversion, arithmetic.
- Basic Assembly Language: Instruction types, addressing modes.
- Design and construction of a combinatorial circuit.

Basic understanding of:
- 2’s complement representation.
- Hexadecimal notation.
- Karnaugh map reduction.
- Design of state-transition diagram for simple regular language recognizers.
Major functions implemented in a CPU.
Instruction cycle.
Memory addressability.

Exposure to:
• Interrupt mechanisms.
• Parallel handshaking and polling loops.
• Hardware and software stack.
• Privileged and non-privileged modes.
• Floating-point standard.

Major Topics Covered in the Course

1. Review of basic concepts (1).
2. Gates, switches and transistors; gate delay; wafer fabrication (1).
3. Combinational devices, including ALU and PLA (4).
4. Sequential circuits: D-latch, D-FF, clocked circuits, registers and counters, state machine implementation (5).
5. Bus Architecture: tri-state logic, bus cycles, access and arbitration, modern bus architectures (2).
6. Simple CPU organization: Data path and control unit (hardwired vs. micro-programmed) (4).
7. Processor architecture: instruction cycles, registers, ISA, addressing modes, pre-fetching, pipelining, RISC/CISC, super-pipelined and superscalar architectures (8).
8. Memory: bit-cell, address decoding, memory chip internals, multi-chip module memory technologies and types, SRAM and DRAM, current technologies, error detection and correction, memory bottleneck, memory hierarchy, cache organization, cache coherence, virtual memory and memory management (11).
10. Survey of modern processor systems (3).
11. Exams (2).

Outcomes

Thorough understanding of:
• Application of standard combinational devices.
• Design and implementation of sequential circuits.
• Organization of a simple processor.
• Basic memory organization and hierarchy.

Basic understanding of:
• Role of gate delay in circuit performance.
• Modern memory technologies.
• Hardwired and micro programmed control.
• Fundamental concepts of RISC and pipelined architectures.
• Cache memory organization and options.
• Virtual memory and memory management.
• Interrupt structures.

Exposure to:
• Construction of gates from switches/transistors.
• IC fabrication process.
• Error detection and correction.
• Concepts of super-pipelined and superscalar architectures.
• Hardware description languages.
• Multiprocessor systems and cache coherence.

Laboratory Projects

The laboratory projects are divided into two portions: lab portion and homework portion. Lab portion is done in the lab during the scheduled lab period and the homework portion is completed prior to the next scheduled lab and verified by the instructor.

1. Introduction to Logicworks; Ripple-carry adder (1 week).
2. Karnaugh map review; breadboard activity, build ALU (1 week).
3. Flip-flops in LogicWorks and breadboard (1 week).
4. FSM implementation (1 week).
5. Tristates, buses, registers (1 week).
6. Bit-cell and BC array; Multichip module (1 week).
7. Introduction of project CPU architecture; test program, ROM (1 week).
8. RTL of Fetch cycle (1 week).
9. Implement fetch cycle (1 week).
10. RTL of execute cycle and optional cycles (1 week).
11. Implement execution cycle for load, store (1 week).
12. Pipelined fetch cycle (1 week).
13. Introduction to Verilog: FSM (1 week).
15. Completion of processor/memory system (1 week).

Estimated Curriculum Category Content (Semester hours)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td></td>
<td></td>
<td>Data Structures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Design</td>
<td></td>
<td></td>
<td>Prog. Languages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comp. Arch.</td>
<td>4.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Oral and Written Communications

No significant component.

Social and Ethical Issues

No significant component.

Theoretical Content

This is the first course in digital logic and computer architecture/organization for computer science students. The course covers the theory of combinational circuit design from standard gates, design partitioning, synchronous circuits, sequential circuit design, CPU data path, and memory hierarchy.

Problem Analysis

Students experiment with analyzing design solutions for correctness, timing issues, and alternative solutions for performance.

Solution Design

Students experiment with design and simulation of basic building blocks of a CPU and a computer.