Master Course Syllabus

School of Engineering and Computer Science
Washington State University Vancouver

CS 214

Design of Logic Circuits

3 Semester Hours
(2 lecture hours, 3 laboratory hours)

Catalog Description

Design and application of combinational logic circuits with exposure to modern methods and design tools; introduction to sequential logic circuits.

Prerequisite Courses

CS 121 – Program Design and Development

Or

CS 251 – C Programming Language

Prerequisite Topics

- Novice ability to program using an imperative programming language

Measured Course Outcomes

Students taking this course will:

1. Design and implement combinational logic circuits using basic logic gates *(contributes to performance criterion* **B-1***).*
2. Perform minimization of logic expressions *(contributes to performance criterion* **B-1***)*.
3. Design logic circuits using or implementing arithmetic operations, multiplexers, encoders, simple latches and flip-flops *(contributes to performance criterion* **B-1***)*.
4. Design, optimize and implement a finite state machine *(contributes to performance criterion* **B-1***)*.

Required Textbooks

One of:

Engineering Digital Design, R. F. Tinder, Academic Press, 2000 (ISBN 0-12-691295-5)

Contemporary Logic Design, R. H. Katz, Prentice Hall, 2nd Edition, 2004 (ISBN 0-20-130857-6)

Introduction to Digital Logic Design, J. P. Hayes, Addison-Wesley Publishing Co., 1993 (ISBN 0-20-115461-7)

Reference Material

None specified.

Major Topics Covered in the Course

1. Overview and history of digital technology
2. Number systems, binary arithmetic and encoding
3. Binary state terminology, CMOS terminology and symbology, basic logic functions, Boolean algebra
4. CMOS transistor level gate implementation and an overview of integrated circuit design, manufacture and economics
5. Logic minimization and use of minimization tools or techniques
6. Design and application of multiplexers, decoders, encoders, code converters, comparators, parity circuits, and shifters
7. Introduction to VHDL or Verilog with application to the behavioral and structural representation of basic combinational logic devices
8. Application of programmable logic devices (ROMs, PLAs, PALs) to combinational logic design
9. Design and application of basic arithmetic circuits: adders, adder/subtractors, multipliers, dividers, and arithmetic logic units (ALUs)
10. Introduction to timing defects (hazards) in combinational logic circuits
11. Introduction to sequential logic machines-latches, flip-flops, registers, and counters
12. Finite state machines (Mealy & Moore types) and state minimization
13. Implementation of digital logic using FPGAs

Laboratory Projects

Project Area	Weeks
Basic combinational logic circuit implementation and testing	2
Implementation of latches and flip-flops using gates or transistors	1
Simple finite state machine implementation and testing	2
Implementation of a simple data path with control logic	2
Representation/simulation of digital logic using VHDL or Verilog	2
Implementation of digital logic in an programmable device	2

CSAB Category Content

	FUNDAMENTAL	ADVANCED		FUNDAMENTAL	ADVANCED
Data Structures	0	0	Computer Organization and	3	0

			Architecture		
Algorithm & Software Design	0	0	Concepts of Programming Languages	0	0

Oral and Written Communications

There are no significant oral or written communications required in this course.

Social and Ethical Issues

This course contains no significant coverage of social and ethical issues.

Theoretical Content

Topic	Hours
Machine representation of numbers	3
Boolean algebra	6
Finite state machines	3

Problem Analysis

Students apply various analysis techniques, including Karnaugh mapping, state table minimization and hazard analysis, to logic problems. These analyses enable the student to recognize and evaluate the tradeoffs between speed, complexity and correctness inherent in various logic implementations.

Solution Design

Students are assigned 6-8 logic design problems for which they are expected to arrive at designs and/or implementations which meet speed or complexity constraints while performing a specified function. Representative problems solved in class by the instructor, textbook material and homework problems provide students with the skills and knowledge to be applied to developing their own designs. Implementations of student designs are tested by the student in the laboratory.

CC2001

This course provides coverage of topics in the following areas (hours listed are minimums):

DS2. Basic logic	2
AR1. Digital logic and digital systems [core]	6
AR2. Machine level representation of data [core]	3

Course Coordinator:	Dick Lang
Last Updated:	October 5, 2006 (Approved)
Syllabus Version Number:	1.3