# Course Outline

# COMP3222 / COMP9222
# Digital Circuits & Systems

### Teaching Period 2 2011

## Contents


- Course Staff
- Course Details
- Course Summary
- Course Aims
- Student Learning Outcomes
- Assumed Knowledge
- Teaching Rationale
- Teaching Strategies
- Assessment
- Academic Honesty and Plagiarism
- Course Schedule
- Resources for Students
- Course Evaluation and Development


## Course Staff

**Contact Details**

| Staff Name | Role | Email | Phone |
|---|---|---|---|
| Oliver Diessel | Course Convenor | odiessel at cse.unsw.edu.au | 57384 |

**Consultations**

Please contact Oliver by email in order to arrange a meeting.

## Course Details

| | |
|---|---|
| **Course Code:** | COMP3222 / COMP9222 |
| **Course Title:** | Digital Circuits & Systems |
| **Units of Credit:** | 6 |
| **Course WebSite:** | http://www.cse.unsw.edu.au/~cs3222/ |
| **Handbook Entry:** | http://www.handbook.unsw.edu.au/undergraduate/courses/2011/COMP3222.html |

## Course Summary

This course teaches students the fundamentals of digital design.

The course introduces the components of digital systems, explains how these are described in the VHDL hardware description language and, through a series of laboratory exercises, familiarizes the student with the implementation of digital circuits using FPGA prototyping boards.

Students are expected to attend lectures, tutorials and laboratories.

Students will be assessed via the hand-in of solutions to problems, the completion of lab exercises, a class test and final theoretical and practical exams.

## Course Aims

This course aims to provide students with a knowledge of problem solving with digital systems (computer systems and digital circuits). The basic building blocks of combinational and sequential circuits are introduced to enable students to develop circuit solutions to problems and to understand the design and operation of hardware models of digital and computer systems. Students will be introduced to the VHDL hardware description language as a means of describing circuits. Computer aided design tools will be used to specify, simulate and implement a variety of simple systems. Students will learn how to implement and test their designs using Field-Programmmable Gate Arrays.

# Student Learning Outcomes

After completing this course, students will be able to:

- design and implement combinational and synchronous logic circuits,
- distinguish various types of synchronous logic circuits,
- understand engineering concepts in the design of digital circuits,
- understand the role of hardware description languages in digital circuit implementation,
- describe simple hardware functions using a hardware description language,
- make use of CAD tools to enter, simulate and synthesize circuit designs, and
- understand the purpose of and steps involved in digital circuit implementation using Field-Programmable Gate Arrays.

This course contributes to the development of the following graduate attributes:

| **Graduate Attribute** | Where Acquired |
|---|---|
| an in-depth engagement with relevant disciplinary knowledge in its interdisciplinary context | theoretical and practical aspects of digital circuit design |
| the capacity for analytical and critical thinking and for creative problem solving | skills involved in the design and analysis of digital circuits |
| the capacity for enterprise, initiative and creativity | digital design using contemporary methods |
| the skills required for collaborative and multidisciplinary work | digital design as a fundamental engineering skill |

# Assumed Knowledge

Before commencing this course, students should:

- be aware of the operation and/or programming of digital and/or computer systems, and
- have a basic understanding of mathematical logic, discrete mathematical concepts, electrical circuit theory and electronics.

These are assumed to have been acquired in the courses ELEC1111, COMP2121, or equivalent courses and their prerequisites.

# Teaching Rationale

An in-depth understanding of digital circuits and systems is fundamental to your ability to become a successful digital designer and computer engineer.

This course founds studies in computer architecture, embedded systems and configurable logic design, and also forms the basis for a deeper understanding of the hardware underpinnings of operating systems, compilers, networks, graphics systems, etc.

In this course we attempt to provide you, the student, with a rigorous and thorough grounding in the essential skills you will need to carry out digital design activities in the follow-on courses of Computer Architecture, Embedded Systems Design (Design Project A) and Configurable Systems Design (Design Project B).

To that end, we aim to provide a systematic, bottom-up coverage of the essential skills and theory, and to provide a top-down view of the broader area to motivate and highlight the interrelationship of the topics.

In order to learn, students are expected to play an active role in participating by continuously thinking about *HOW* what is said in lectures or read in texts or on the web relates to what you already know from this course and related studies. You are encouraged and expected to ask questions to clarify contradictions or uncertainties in your understanding as we proceed.

In this course, we also place considerable value on putting theory into practice through guided laboratory and tutorial exercises. Staying up-to-date with these will help enormously in picking up the skills we expect you to have acquired by the end of the course.

# Teaching Strategies

This course will involve:

- Lectures, to introduce concepts, show examples
- Tutorials, to reinforce concepts and provide additional examples, and
- Lab Work, to introduce technology and put theory into practice.

The course will also involve:

- Hand-in Problems, to encourage practice of the key skills,
- Theory Tests, to encourage study and more formally gauge uptake of theoretical skills, and

- Practical Test, to encourage participation in laboratory work and to determine whether acquired skill levels are sufficient to pass the course.

## Assessment

| Assessment Item | Contribution |
|---|---|
| Weekly hand-in problems | 15% |
| Labs | 15% |
| Week 8 Class Test | 15% |
| Final Theory Test | 25% |
| Final Practical Test | 30% |
| **TOTAL** | **100%** |

## Academic Honesty and Plagiarism

**Plagiarism** is defined as "using the words or ideas of others and presenting them as your own". UNSW and CSE treat plagiarism as academic misconduct, which means that it carries penalties as severe as being excluded from further study at UNSW. There are several on-line sources to help you understand what plagiarism is and how it is dealt with at UNSW:

- Learning Centre: Plagiarism and Academic Integrity
- MyUNSW: Plagiarism and Academic Misconduct
- CSE: Addendum to UNSW Plagiarism Guidelines
- CSE: Yellow Form (whose terms you have agreed to)

Make sure that you read and understand these. Ignorance is not accepted as an excuse for plagiarism.

## Course Schedule

| Week | Lecture | Text* | Lab |
|---|---|---|---|
| 1 | Introduction | Ch 1 & 2 | — |
| 2 | Implementation technology | Ch 3 | 1. Installation, simulation |
| 3 | Optimizing logic functions | Ch 4.1-4.7 & 4.12 | 1. cont. implementation, minimization |
| 4 | Number representation & arithmetic circuits | Ch 5.1-5.5 | 2. Arithmetic circuits |
| 5 | Combinational building blocks | Ch 6 | 2. cont. |
| 6 | Flip-flops | Ch 7.1-7.7 | 3. Combinational building blocks |
| 7 | Registers | Ch 7.8-7.13 | 4. Flip-flops & counters |
| BREAK | — | — | — |
| 8<br>Â | Designs using FFs & registers<br>Wed: CLASS TEST | Ch 7.14-7.16<br>Â | 4. cont.<br>Â |
| 9 | Synchronous sequential circuits | Ch 8.1-8.4 | 5. Finite state machines |
| 10 | Synchronous sequential circuits | Ch 8.5-8.9 | 5. cont. |
| 11 | Digital system design | Ch 10 | 6. Blackjack player |
| 12 | Course wrap-up | — | 6. cont. |
| 13 | — | — | 6. cont. |

* Chapter references are to the course textbook — see below.

| Activity | When | Where |
|---|---|---|
| Lectures<br>Â | Mon 12-14<br>Wed 12-13 | Mathews 102<br>Mathews 102 |
| Tutes<br>Â | Tue 11-12<br>Thu 15-16 | Quadrangle G055<br>Quadrangle G055 |
| Labs<br>Â<br>Â | Wed 10-12<br>Wed 14-16<br>Thu 16-18 | K17 Lyre Lab<br>K17 Lyre Lab<br>K17 Lyre Lab |

## Resources for Students

**Course textbook:**

- Stephen Brown & Zvonko Vranesic, Fundamentals of Digital Logic with VHDL Design, 3ed, McGraw-Hill, 2009

**Recently used texts (still quite useful):**

- Katz and Borriello, Contemporary Logic Design (2nd edition)
- Mano and Kime, Logic and Computer Design Fundamentals (3rd Edition)

**Other resources:**

- Altera on-line User Guides and Data Sheets

# Course Evaluation and Development

This course is evaluated each session using the CATEI system.

In the previous offering of this courses, students enjoyed the course because it combines practical work with theory. They also felt more weight should be given to the lab work, less heavy assignments should be given, and that less slides should be used in lectures.

Based on their comments, I have replaced the assignments with more regular hand-ins and a class test. This will save students' time, allow us to reduce the coverage of the final theory exam and provide more time for the practical exam. I will also try to place the lecture notes on a diet!