Course Outline

COMP2121
Microprocessors and Interfacing
Teaching Period 1 2010

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
Sri Parameswaran	Course Convenor	sridevan at cse.unsw.edu.au	57223
Haris Javaid	Course Admininstrator	harisj at cse.unsw.edu.au	

Consultations

Staff Name	Day/Time	Location

Course Details

Course Code:	COMP2121
Course Title:	Microprocessors and Interfacing
Units of Credit:	6
Course WebSite:	http://www.cse.unsw.edu.au/~cs2121/
Handbook Entry:	http://www.handbook.unsw.edu.au/undergraduate/courses/2010/COMP2121.html

Course Summary

This course explores ...

Course Aims

This course aims to introduce important topics in microprocessors and microcontrollers. If you successfully complete this course, you will have a working knowledge of microprocessors and interfacing, and will be able to solve problems related to the topics covered in this course, and will be proficient in AVR assembly programming.

By the end of this subject you will be able to:

- understand how numbers and characters are represented in computers
- understand basic digital components
- apply general concepts of Instruction Set Architectures (ISAs)
- interpret a number of ISAs including the AVR
- independently write assembler programs
- interface a microcontroller with typical low level I/O systems
- build a small working embedded system

Student Learning Outcomes

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

Graduate Attribute	Where Acquired
the skills involved in scholarly enquiry	Coursewide
an in-depth engagement with relevant disciplinary knowledge in its interdisciplinary context	Coursewide
the capacity for analytical and critical thinking and for creative problem solving	Labs and Project
the ability to engage in independent and reflective learning	Coursewide
the skills to locate, evaluate and use relevant information (Information Literacy)	Assignment
the capacity for enterprise, initiative and creativity	Labs and Project
an appreciation of, and a responsiveness to, change	Project
a respect for ethical practice and social responsibility	Coursewide

Assumed Knowledge

Before commencing this course, students should:

- be able to competently program in C;
- be able to represent integers in binary;
- be able to work independently; and
- be able to work in a group;

These are assumed to have been acquired in the courses COMP1021 or COMP1091 or COMP1721 or COMP1921 or COMP1927 .

Teaching Rationale

This course teaches the theory and skills needed to design and implement systems utilising microprocessor systems. Passing this course involves keeping up with the theory and putting in the time to complete the lab exercises. The lab work brings the theory alive and clarifies deeper issues. Without the theory, the lab work is difficult to master. Each therefore complements the other. Both require time. If you allocate sufficient time each week, you won't have any problems passing, and it can be a very rewarding experience. It is our belief that certain activities make a world of difference to your learning experience at University. You should:

- actively participate in lectures - if something is unclear, ask for clarification - if you don't understand something, try to figure it out when you reflect upon the material during private study, and if you still don't get it, raise it during the next class
- don't imagine you are the only one having trouble with something; chances are somebody else is having trouble as well. However, lecturers aren't necessarily aware of your problems, so please let them know when something is troubling you
- be prepared to concentrate during classes - if you only pay them half your attention, you'll probably only get half as much out of the experience as you would had you focused on the task completely
- actively participate in tutorials - prepare beforehand - know which problems you have difficulty solving - be prepared to participate in deriving a solution during the tutorial
- prepare for labs by reading up what is expected well before your lab session - sometimes you will need to complete some preparatory exercise - lab time is too valuable to waste on preparations that could have been completed beforehand
- make use of the staff; pester your lecturer, your tutor, and your lab demonstrator when you don't understand something

Teaching Strategies

- Concepts will be introduced during lectures. Students are expected to review these and read related material in order to form a theoretical understanding.
- Practical skills will be developed through laboratory exercises. Staff will provide these beforehand with sufficient time to prepare for them.
- Class test will be used to encourage students to study the subject and to determine the level of understanding achieved.
- A project will be given to you to complete. This project will use most of the concepts learned in class.

Assessment

There will be an assignment and a project for this course each counting towards 25% of your final grade.

- Assignment (10%)
- Project (15%)

The assessment for this course will consist of: a 3 hour final exam which will be worth 50%; a mid semester exam will count towards 10% of your final mark; Labs will count towards 15% of your final mark; and the aforementioned assignments will make up the last 25%. The final examination examine whether you:

- understand how numbers and characters are represented in computers
- understand basic digital components
- can apply general concepts of Instruction Set Architectures (ISAs)
- can interpret a number of ISAs including the AVR
- can independently write assembler programs

The first assignment will examine your ability to interpret other ISAs The second assignment will examine your ability to build a small working embedded system The labs will examine whether you can:

- independently write assembler programs
- interface a microcontroller with with typical low level I/O
- build a small working embedded system

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

Programmer model of computer organization using assembly and machine language. Process of translation from high-level language to machine instructions. Number representation, computer arithmetic, instruction sets, I/O interfacing, I/O interrupts, buses, memory systems, serial communication and analog input and output.

Resources for Students

Text Book

- Fredrick M. Cady: Microcontrollers and Microcomputers —Principles of Software and Hardware Engineering.

Further References:

1. Mano, M.M. & Kime, C.R.: Logic and Computer Design Fundamentals, 3rd Edition, Prentice-Hall (2003).
2. Brian Kernighan & Dennis Ritchie: The C Programming Language, 2nd Ed., Prentice Hall, 1988, ISBN: 0-13-110362-8
3. David Patterson and John Hennessy: Computer Organization & Design: The HW/SW Interface," 2nd Ed Morgan Kaufmann, 1998, ISBN: 1 - 55860 - 491X

Course Evaluation and Development

Student feedback on this course will be obtained via electronic survey at the end of session, and will be used to make continual improvements to the course. Students are also encouraged to provide informal feedback during the session, and to let the lecturer in charge know of any problems, as soon as they arise. Suggestions will be listened to very openly, positively, constructively and thankfully, and every reasonable effort will be made to address them.

The CATEI evaluation from the last time I taught this course showed that students were generally satsified with all aspects of the course. Thus we intend to maintain the same style and structure for the up-coming offering. .