Course Outline Public

Edit Resource (/COMP2121/19T2/resources/27794/edit)

Contents

- Course Details
- Course Summary
- Assumed Knowledge
- Student Learning Outcomes
- Teaching Strategies
- Teaching Rationale
- Student Conduct
- Assessment
- Course Schedule
- Resources for Students
- Course Evaluation and Development

Course Details

Course Code	COMP2121
Course Title	Microprocessors and Interfacing
Convenor	Hui Wu (http://www.cse.unsw.edu.au/~huiw)
Admin	Hasindu Gamaarachchi (/users/z5136909)
Classes	**Lectures** : Timetable for all classes (https://webcms3.cse.unsw.edu.au/COMP2121/19T2/timetable)
Consultations	Friday 2-5 pm
Units of Credit	6
Course Website	https://webcms3.cse.unsw.edu.au/COMP2121/19T2/ (https://webcms3.cse.unsw.edu.au/COMP2121/19T2/)
Handbook Entry	http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP2121.html (http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP2121.html)

Course Summary

The aims of this course are to introduce students to the basic concepts and major components of microprocessors and micro controllers and to provide students with knowledge and skills for solving problems with microprocessors and micro-controllers.

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.

Student Learning Outcomes

After completing this course, students will:

- know the fundamentals of microprocessors and microcontrollers;
- understand how C programs are converted into AVR machine instructions;
- be proficient in AVR assembly language programming;
- understand typical interface designs between hardware and software;
- understand how serial communication works;
- understand how analog signals are converted into digital signals and vice versa; and,
- obtain basic development skills for microprocessor / microcontroller applications.

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

Graduate Capability	Acquired in
Scholars capable of independent and collaborative enquiry, rigorous in their analysis, critique and reflection, and able to innovate by applying their knowledge and skills to the solution of novel as well as routine problems	All Components
Entrepreneurial leaders capable of initiating and embracing innovation and change, as well as engaging and enabling others to contribute to change	All Components
Professionals capable of ethical, self- directed practice and independent lifelong learning	All Components
Global citizens who are culturally adept and capable of respecting diversity and acting in a socially just and responsible way	All Components

Teaching Strategies

- Lectures - introduce concepts, show examples
- Lab Work - explore concepts in depth with practical tasks
- Project (with lab work)- allow students to solve significant problems and apply what they have learnt

In this course a variety of teaching strategies will be used.

1. Teaching in class, where lectures are delivered and interaction encouraged. The lack of interaction provided by the WEB lecture stream may be compensated for by consultation sessions.
2. Laboratory exercises are performed under supervision and assessed.
3. A project is given which is part of the lab work which will expand upon the concepts already learned, or will apply concepts learned to create an artifact.

You are expected to spend at least one hour on the course for every hour you spend in class (i.e., at least 6 hours per week). You should practice programming as much as possible. Creativity in subjects such as this will only be possible when you are an expert. It is no different from being a pianist, artist, or a mathematician. You should understand all introduced concepts clearly. If not, you are encouraged to stop and ask.

Teaching Rationale

This course is taught the way it is because knowledge and skills have to be gained.

Student Conduct

The **Student Code of Conduct** (Information (https://student.unsw.edu.au/conduct) , Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)) sets out what the University expects from students as members of the UNSW community. As well as the learning, teaching and research environment, the University aims to provide an environment that enables students to achieve their full potential and to provide an experience consistent with the University's values and guiding principles. A condition of enrolment is that students *inform themselves* of the University's rules and policies affecting them, and conduct themselves accordingly.

In particular, students have the responsibility to observe standards of equity and respect in dealing with every member of the University community. This applies to all activities on UNSW premises and all external activities related to study and research. This includes behaviour in person as well as behaviour on social media, for example Facebook groups set up for the purpose of discussing UNSW courses or course work. Behaviour that is considered in breach of the Student Code Policy as discriminatory, sexually inappropriate, bullying, harassing, invading another's privacy or causing any person to fear for their personal safety is serious misconduct and can lead to severe penalties, including suspension or exclusion from UNSW.

If you have any concerns, you may raise them with your lecturer, or approach the School Ethics Officer (mailto:ethics-officer@cse.unsw.edu.au) , Grievance Officer (mailto:grievance-officer@cse.unsw.edu.au) , or one of the student representatives.

Plagiarism is defined as (https://student.unsw.edu.au/plagiarism) 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:

- Plagiarism and Academic Integrity (https://student.unsw.edu.au/plagiarism)
- UNSW Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

Make sure that you read and understand these. Ignorance is not accepted as an excuse for plagiarism. In particular, you are also responsible that your assignment files are not accessible by anyone but you by setting the correct permissions in your CSE directory and code repository, if using. Note also that plagiarism includes paying or asking another person to do a piece of work for you and then submitting it as your own work.

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW staff and students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.

If you haven't done so yet, please take the time to read the full text of

- UNSW's policy regarding academic honesty and plagiarism (https://student.unsw.edu.au/plagiarism)

The pages below describe the policies and procedures in more detail:

- Student Code Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)
- Student Misconduct Procedure (https://www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf)
- Plagiarism Policy Statement (https://www.gs.unsw.edu.au/policy/documents/plagiarismpolicy.pdf)
- Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

You should also read the following page which describes your rights and responsibilities in the CSE context:

- Essential Advice for CSE Students (https://www.engineering.unsw.edu.au/computer-science-engineering/about-us/organisational-structure/student-services/policies/essential-advice-for-cse-students)

Assessment

Item	Topics	Due	Marks
Project	A monorail emulator	Week 11	15%
Labs	All topics	Weeks 2-10	25%
Final Exam	All topics	Exam period	60%

Course Schedule

Each week there are two lectures. The following topics will be covered:

- Instruction set architecture.
- Number systems
- AVR assembly programming and assembly process
- Interrupts
- Buses and I/O
- Serial communication
- Analog/digital and digital/analog conversions.

Resources for Students

Student Forum

Text Book:

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

Further References:

1. AVR Data Sheet (on line).
2. AVR Instruction Set (on-line).
3. AVR Assembler Tutorial (on-line).
4. LCD Manual (on-line).
5. Mano, M.M. & Kime, C.R.: Logic and Computer Design Fundamentals, 3rd Edition, Prentice-Hall (2003).
6. Brian Kernighan & Dennis Ritchie: The C Programming Language, 2nd Ed., Prentice Hall, 1988, ISBN: 0-13-110362-8.
7. 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

This course is evaluated each session using the myExperience system.

Resource created 11 months ago (Friday 31 May 2019, 01:41:04 PM), last modified 11 months ago (Wednesday 05 June 2019, 11:57:09 AM).

Comments

(/COMP2121/19T2/forums/search?forum_choice=resource/27794)

(/COMP2121/19T2/forums/resource/27794)

There are no comments yet.