ELEC4633
Real Time Engineering

Course Outline – Semester 1, 2017

Australia's Global University | Faculty of Engineering | School of Electrical Engineering and Telecommunications

Course Staff

Course Convener: Dr. Branislav Hredzak, Room 642, b.hredzak@unsw.edu.au

Tutor: Dr. Branislav Hredzak, Room 642, b.hredzak@unsw.edu.au

Laboratory Contact: Dr. Zhenyu Liu, Room G15A, z.liu@unsw.edu.au
Laboratory Demonstrators: Raymond Yesheng Cui yesheng.cui@unsw.edu.au
Edward Dai e.dai@student.unsw.edu.au

Consultations: You are encouraged to ask questions on the course material, after the lecture class times in the first instance, rather than via email. Lecturer consultation times will be advised during lectures. You are welcome to email the tutor or laboratory demonstrator, who can answer your questions on this course and can also provide you with consultation times. ALL email enquiries should be made from your student email address with ELEC/TELExxxx in the subject line, otherwise they will not be answered.

Keeping Informed: Announcements may be made during classes, via email (to your student email address) and/or via online learning and teaching platforms – in this course, we will use Moodle https://moodle.telt.unsw.edu.au/login/index.php. Please note that you will be deemed to have received this information, so you should take careful note of all announcements.

Course Details

Contact Hours

The course consists of 2 hours of lectures each week, a 1-hour tutorial every other week, and a 3-hour laboratory session every other week. Tutorials start in week 3 and labs in week 2.

	Day	Time	Location
Lectures	Tuesday	1200-1400 Weeks: 1-12	OMB 230
Tutorials	Wednesday	1600-1700 Even weeks: 2-12	OMB 230
	Wednesday	1600-1700 Odd weeks: 3-13	OMB 230
Labs	Tuesday	1500-1800 Even weeks: 2-12	ElecEng 201
	Tuesday	1500-1800 Odd weeks: 3-13	ElecEng 201
	Wednesday	0900-1200 Even weeks: 2-12	ElecEng 201
	Wednesday	0900-1200 Odd weeks: 3-13	ElecEng 201

Elec4633 is a 6 UOC course; expected workload is 8-10 hours per week during the 12 week session.

Context

Real Time Engineering is concerned with the design and implementation of computer-based real time systems, and deals with the hardware and software issues associated with ensuring they work in a practical and real time sense. Broadly speaking, a system is said to be real time if it adheres to constraints in time. The real time system may be one centred around applications in control systems, signal processing, instrumentation, simulation, and many more. A real time system may be governed by a single stand-alone computer, or a computer embedded within the application itself, and hence known as an *embedded system.*

Assumed Knowledge

The subject follows on from material covered in introductory courses in real time instrumentation (ELEC3145 Real Time Instrumentation) and embedded systems (ELEC2142 Embedded Systems Design), however these are not pre-requisites. It is assumed that students will have a basic understanding of real time concepts, will be able to program in the C+ language, and will have had a small exposure in the use of embedded and control systems.

Course Objectives

The objective of this course is to equip students with the necessary skills to analyse, design and implement computer-based real time systems, as well as critically evaluate their performance. With this course and some further experience, students should be able to convert a worded problem specification into a functional and reliable real time solution which satisfies all requirements. Although real time systems encompass a very broad range of application areas, a central theme in this course is the application of real time computing for the purpose of signal processing and control.

The course aims to give students fundamental knowledge in real time operating systems, including scheduling, kernels, and inter-process communication, as well as skills in the effective use of embedded computers.

Learning Outcomes

At the successful completion of this course, you should be able to:

1. develop high-level design strategies for a given real time design problem statement, and to be able to objectively assess the strengths and weaknesses of each strategy,
2. demonstrate an ability to think about embedded system design issues, and to design and implement simple real time embedded systems from problem specification to code generation,
3. demonstrate an understanding of real time operating system concepts, such as scheduling, multitasking and task/context switching, inter-process communication, and have a knowledge of the different types of real time kernels,
4. demonstrate an understanding of, and a basic competence in, such real time and embedded system aspects as input/output device programming, interrupt programming, and programming of microcontrollers,
5. demonstrate an understanding of issues of safety, reliability, and performance analysis, in real time embedded systems,
6. demonstrate an understanding of a real-time Linux kernel, and be able to use it to construct simple to moderately complex real-time programs.

Syllabus and Schedule

Not all of the topics listed below will be covered in the lecture program, but a majority will be. A tentative schedule for the syllabus is:

1. An introduction to real time and embedded systems. Sound embedded system design. Characterising real-time systems.

2. Real time operating systems. Software architectures. Concurrent process. Tasks and threads. Real time kernels. State machines. Linux and RTAI. Scheduling. Dynamic and static scheduling. Rate-Monotonic scheduling. Context switching. Task management.
3. Inter-process communication. Data buffering. Semaphores. Critical regions. Mutual exclusion. Message passing. Memory management and global memory. Priority inversion.
4. Real time systems analysis. Safety and reliability. Fault tolerance techniques. Performance analysis. Execution time prediction.
5. Embedded system development. Microcontrollers. Real time embedded operating systems.

Graduate Attributes

Graduate attributes are those which the University and/or Faculty of Engineering agrees students should develop during their degree. Further information can be obtained in the document. This course aims to contribute to students attaining the following graduate attributes:

- The ability to engage in independent and reflective learning, addressed through the laboratory program exercises, and through independent assignment exercises.
- The skills of effective communication, which are addressed by tutorial and assignment problems and assessed in the final exam.
- The capacity for enterprise, initiative and creativity, which is addressed by the laboratory program, where students can use their initiative in solving the specified problems in a number of different ways.
- The capacity for analytical and critical thinking and for creative problem-solving, which is addressed by tutorial and lab exercises, again, where students need to provide critical evaluation of their solutions to problems in order to determine their merits and pitfalls.

Teaching Strategies

A problem-based learning approach is partly employed in delivering this course. For each section of the course, different examples of real systems are introduced in lectures as a way of explaining the concepts. The analysis, and sometimes design, for each example will lend itself in assisting students to satisfy the learning objectives above. The lab program aims to support the lecture program, and provides the student with hands-on software design in interacting with, and controlling real hardware. The laboratory program is challenging, and as such, students must come prepared for each laboratory in order to complete the lab program on time.

The tutorial sessions will also be problem based, with examples of real systems introduced as problem specifications for the students to design a solution. To help facilitate the solutions, tutorials will involve class discussion.

Students will be required to do home-based work, which will include the self-guided tutorial questions, related to course material covered in lectures.

Assessment

Final Examination		50%
Mid-session Test		20%
Laboratory Component:	Laboratory Assignment	10%
	Laboratory Checkpoints	20%

Final Examination The final examination will be a 2 hour, closed book exam dealing with material in texts, real-systems engineering concepts, and requiring simple program specification and coding examples.

Mid-session Test There will be a mid-session test in Week 7 (date, time, and location to be advised). The aim of the test will be to assess one, or more, of the learning outcomes stated above.

Lab Assignment More information will be provided in Week 10, however it will be due at the end of Week 12.

Laboratory Component

Laboratory work commences in Week 2 and all experiments are done in Room 201. Students are only required to attend labs once per fortnight. Each student will carry out 3 lab experiments, and each experiment is made up of a series of checkpoints, some of which are bonus checkpoints. This component of the assessment is worth 20 marks. A signature and mark is required for each checkpoint. Each signature and mark will be grudgingly given, based on completion of the checkpoint, the answers to specific checkpoint related questions, a sufficient understanding of the work done and concepts explored, and a satisfactory record (via a lab book) of the checkpoint completion; unsatisfactory records will not be rewarded, and the decision of the tutor will be final. Partial marks for each checkpoint may be given. Tutors may be asked for advice on what is necessary to obtain a signature. More details will be contained in the laboratory handouts. Laboratory work is an essential component of this subject.

Assessment submission guidelines

Hand-in times for the lab assignment will be strictly enforced. A 20% penalty will be imposed for each day or part of a day that the lab assignment is late.

Relationship of Assessment Methods to Learning Outcomes

	Learning outcomes					
Assessment	1	2	3	4	5	6
Laboratory Assignment	✓	✓	✓	✓	✓	✓
Laboratory Checkpoints	✓	✓	✓	✓	✓	✓
Mid-session quizzes	✓	✓	✓	✓	✓	✓
Final exam	✓	✓	✓	✓	✓	✓

Course Resources

Recommended Textbooks and Reading Material

The following is a list of books used as references during the course. Each book tends to place an emphasis on different areas of real time systems design, and as such, there is no one book that can be prescribed a standalone text.

1. **James K. Peckol**, Embedded Systems: A Contemporary Design Tool, Wiley, 2008.
2. **Steven F. Barrett, Daniel J. Pack**, Embedded Systems: Design and Applications with the 68HC12 and HCS12, Prentice Hall, 2005.
3. **Jane W. S. Liu**, *Real-Time Systems*, Prentice Hall, 2000.
4. **Alan C. Shaw**, Real-Time Systems and Software, Wiley, 2001.
5. **Daniel W. Lewis**, Fundamentals of Embedded Software: Where C and Assembly Meet, Prentice Hall, 2002.
6. **Phillip A. Laplante**, **Seppo J Ovaska**, Real-Time Systems Design and Analysis – Tools for the Practitioner, IEEE Press, 2012.

The Laboratory Program

The laboratory work will take place in EE201. This lab is equipped with computers each supporting the Linux Operating System, together with the RTAI kernel. Some arrangements may be made for work outside the formal laboratory times, but other classes do use the lab, and will take priority.

All practical work is to be recorded in a bound workbook. A bound student exercise book would be suitable for the purpose. The workbook will be perused by laboratory supervisors at checkpoints through the session see Assessment). The workbook should be a diary of your thoughts as you undertake the labs and assignments. Program specifications should be clearly noted, together with clever implementation ideas, results of tests, and suggested extensions. Documentation which does not appear in the programs should be in the notebook. Please see the attached appendix for more explanation of keeping lab notebooks.

All laboratory experiments will be described in simple documents which detail the work to be undertaken, but which leave considerable flexibility in the methods by which this might be achieved. There are three laboratory experiments, however each experiment will work towards the common goal of designing and implementing a real time embedded control system. The expectation is that you will have to work at times other than the assigned laboratory periods to complete the work. Lab1 will be nominally completed in two laboratory sessions, Lab 2 in three laboratory sessions and Lab 3 in one laboratory session.

Experiments

The goal of the laboratory program is to design and implement a reliable real time embedded control system. The bench process to be controlled will be a DC motor. At the successful completion of the lab program, the position of the DC motor will be controlled directly and automatically by an embedded microcontroller/computer, with supervision and operator interface being supplied by the RTAI kernel driven desktop PC. The overall task is simplified by breaking it down into smaller tasks. The philosophy is to design the software system firstly on the desktop PC, and then once tested, port it, bit-by-bit to the microcontroller.

Software

A real-time operating system will be used throughout the course. The OS is RTAI, written as a real-time extension" to the Linux operating system.

Linux is a full-featured POSIX-like OS (so that it "looks" like UNIX), and is becoming an increasingly popular choice of OS. Its source is available as freeware, or alternatively, a distribution can be purchased at a small cost through a variety of distributors.

RTAI is designed to give real-time capabilities to Linux. It does not replace Linux, but simply allows real-time processes to run while Linux operates fully in the background as a low priority process.

Online Resources

All material available in electronic format, will be available in Moodle:

https://moodle.telt.unsw.edu.au/login/index.php

Each student enrolled will be granted access to the ELEC4633 subject page in Moodle, where your login is your standard zPass login.

Moodle will be used for such things as:

- displaying/posting notices/messages;
- posting lecture notes / tutorial handouts / lab exercises / short quizzes;
- quizzes;
- hosting discussions (only related to the subject) between class/teacher class/class etc;
- posting grades as they become available.

It is encouraged that students seeking advice/help on matters related to the course material seek help from other students, either in person, or via the discussion board in Moodle.

Note: The discussion board is not to be used as a chat forum for subject matter not related to the subject.

Continual Course Improvement

Students are advised that the course is under constant revision in order to improve the learning outcomes of its students. Please forward any feedback (positive or negative, as long as it is constructive) on the course to the course convener or via the myExperience process.

Based on the myExperience process feedback additional access to the lab room EE201 will be arranged this year.

Appendix on Laboratory Notebooks

According to Barrett [Barrett et.al. 2005], "Lab notebooks are used to record the process of scientific discovery, project evolution, design rationale, steps in engineering analysis, procedures followed, and raw data collected. ... Furthermore, a carefully maintained notebook allows for adequate reconstruction of original work years from the original entry ...".

Barrett also offers some guidelines that should be followed in maintaining a "good" lab book. The following points are based on the guidelines.

1. Ensure the book is bound so that individual pages cannot be lost or removed. By the same token, do not record material on loose sheets of paper, unless they are properly bound.
2. Make sure all entries are sequential in chronological order. Do not mix up recorded lab material with lecture or tutorial (or other) notes.
3. Ensure that the material is recorded in a legible fashion, so that others can read it and use it to reconstruct your experiment.
4. Number pages sequentially.

5. Do not obliterate error, cross them out with a single line.

Importantly, the lab notebooks should NOT be written as formal lab reports where much of the elements in the quote above are not included.

Reference

Barrett S. F., and Pack D. J., "Embedded Systems: Design and Applications with the 68HC12 and HCS12", Prentice Hall, New Jersey, 2005.

Appendix on Plagiarism

The following is an official, now mandatory inclusion in all UNSW course handouts.

"Plagiarism is the presentation of the thoughts or work of another as one's own. [1] Examples include:

- direct duplication of the thoughts or work of another, including by copying work, or knowingly permitting it to be copied. This includes copying material, ideas or concepts from a book, article, report or other written document (whether published or unpublished), composition, artwork, design, drawing, circuitry, computer program or software, web site, Internet, other electronic resource, or another person's assignment without appropriate acknowledgement
- paraphrasing another person's work with very minor changes keeping the meaning, form and/or progression of ideas of the original;
- piecing together sections of the work of others into a new whole;
- presenting an assessment item as independent work when it has been produced in whole or part in collusion with other people, for example, another student or a tutor; and,
- claiming credit for a proportion a work contributed to a group assessment item that is greater than that actually contributed. [2]

Submitting an assessment item that has already been submitted for academic credit elsewhere may also be considered plagiarism.

"The inclusion of the thoughts or work of another with attribution appropriate to the academic discipline does not amount to plagiarism.

"Students are reminded of their Rights and Responsibilities in respect of plagiarism, as set out in the University Undergraduate and Postgraduate Handbooks, and are encouraged to seek advice from academic staff whenever necessary to ensure they avoid plagiarism in all its forms.

"The Learning Centre website is the central University online resource for staff and student information on plagiarism and academic honesty. It can be located at:

https://student.unsw.edu.au/plagiarism

The Learning Centre also provides substantial educational written materials, workshops, and tutorials to aid students, for example, in:

- correct referencing practices;
- paraphrasing, summarising, essay writing, and time management;

- appropriate use of, and attribution for, a range of materials including text, images, formulae and concepts.

Individual assistance is available on request from The Learning Centre.

"Students are also reminded that careful time management is an important part of study and one of the identified causes of plagiarism is poor time management. Students should allow sufficient time for research, drafting, and the proper referencing of sources in preparing all assessment items.

---

1 Based on that proposed to the University of Newcastle by the St James Ethics Centre. Used with kind permission from the University of Newcastle.
2 Adapted with kind permission from the University of Melbourne.

Appendix A: Targeted Graduate Capabilities

Electrical Engineering and Telecommunications programs are designed to address the following targeted capabilities which were developed by the school in conjunction with the requirements of professional and industry bodies:

- The ability to apply knowledge of basic science and fundamental technologies;
- The skills to communicate effectively, not only with engineers but also with the wider community;
- The capability to undertake challenging analysis and design problems and find optimal solutions;
- Expertise in decomposing a problem into its constituent parts, and in defining the scope of each part;
- A working knowledge of how to locate required information and use information resources to their maximum advantage;
- Proficiency in developing and implementing project plans, investigating alternative solutions, and critically evaluating differing strategies;
- An understanding of the social, cultural and global responsibilities of the professional engineer;
- The ability to work effectively as an individual or in a team;
- An understanding of professional and ethical responsibilities;
- The ability to engage in lifelong independent and reflective learning.

Appendix B: UNSW Graduate Capabilities

The course delivery methods and course content directly or indirectly addresses a number of core UNSW graduate capabilities, as follows:

- Developing scholars who have a deep understanding of their discipline, through lectures and solution of analytical problems in tutorials and assessed by assignments and written examinations.
- Developing rigorous analysis, critique, and reflection, and ability to apply knowledge and skills to solving problems. These will be achieved by the laboratory experiments and interactive checkpoint assessments and lab exams during the labs.
- Developing capable independent and collaborative enquiry, through a series of tutorials spanning the duration of the course.
- Developing independent, self-directed professionals who are enterprising, innovative, creative and responsive to change, through challenging design and project tasks.

Appendix C: Engineers Australia (EA) Professional Engineer Competency Standard

	Program Intended Learning Outcomes	
PE1: Knowledge and Skill Base	PE1.1 Comprehensive, theory-based understanding of underpinning fundamentals	✓
	PE1.2 Conceptual understanding of underpinning maths, analysis, statistics, computing	✓
	PE1.3 In-depth understanding of specialist bodies of knowledge	✓
	PE1.4 Discernment of knowledge development and research directions	
	PE1.5 Knowledge of engineering design practice	✓
	PE1.6 Understanding of scope, principles, norms, accountabilities of sustainable engineer practice	
PE2: Engineering Application Ability	PE2.1 Application of established engineering methods to complex problem solving	✓
	PE2.2 Fluent application of engineering techniques, tools and resources	✓
	PE2.3 Application of systematic engineering synthesis and design processes	
	PE2.4 Application of systematic approaches to the conduct and management of engineer projects	
PE3: Professional and Personal Attributes	PE3.1 Ethical conduct and professional accountability	
	PE3.2 Effective oral and written communication (professional and lay domains)	✓
	PE3.3 Creative, innovative and pro-active demeanour	✓
	PE3.4 Professional use and management of information	✓
	PE3.5 Orderly management of self, and professional conduct	
	PE3.6 Effective team membership and team leadership	✓