UNSW SYDNEY | Australia's Global University

School of Electrical Engineering and Telecommunications

Semester 2, 2017
Course Outline

ELEC 2142
Embedded System Design

COURSE STAFF

Course Convener: Dr. Chamith Wijenayake, Hilmer building - room 643A, c.wijenayake@unsw.edu.au
Tutor: Dr. Chamith Wijenayake, Hilmer building - room 643A, c.wijenayake@unsw.edu.au
Dr. Phu Le, Room EE206, ngoc.le@unsw.edu.au
Laboratory Contact: Ms. Yen Nee Ho, yennee.ho@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 time for this course will be on every Friday 4-6 pm at Hilmer building room 643A. 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 ELEC2142 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 SUMMARY

Contact Hours

The course consists of 3 hours of lectures, a 1-hour tutorial, and a 2-hour laboratory session each week. **Tutorials will start in week 2 and laboratory sessions will start in week 4.**

	Day	Time	Location
Lectures	Tuesday	3pm - 5pm	ColomboThA
	Wednesday	1pm - 2pm	CLB 8
Tutorials	Monday	5pm - 6pm	Col LG02
	Tuesday	1pm - 2pm	Law303
	Tuesday	5pm - 6pm	Col LG02
	Wednesday	12pm - 1pm	CivEng G8
Laboratories	Monday	1pm – 3pm	ElecEng233
	Monday	3pm – 5pm	ElecEng233
	Tuesday	9am - 11am	ElecEng233
	Tuesday	11am - 1pm	ElecEng233
	Tuesday	1pm – 3pm	ElecEng233
	Wednesday	9am - 11am	ElecEng233
	Wednesday	11am - 1pm	ElecEng233
	Wednesday	4pm – 6pm	ElecEng233
	Thursday	9am - 11am	ElecEng233
	Thursday	1pm – 3pm	ElecEng233
	Friday	9am - 11am	ElecEng233
	Friday	11am - 1pm	ElecEng233
Consultation	Friday	4pm - 6pm	Hilmer - 643A

Context and Aims

Embedded systems are pervasive in all areas of society, and as such, knowledge of how to design them is a vital skill for all electrical engineers. The objective of this course is to equip students with the knowledge and skills that enable them to design basic embedded systems, where a microprocessor/microcontroller is the central element. The first half of the course will focus on ARM processor architecture, instruction sets, assembly language fundamentals and techniques. Input and output, floating point representation, interrupts, and exceptions will be covered in the second half of the course. At the completion of the course, students should be in a position to be able to design reliable embedded system using ARM processors in particular and other processors in general.

Indicative Lecture/Tutorial/Laboratory Schedule

Week	Lecture	Tutorial	Lab
1	Introduction to ARM architecture.		
2	Assembly language basics and data processing instructions.	1	
3	Memory access.	2	
4	Control flow.	3	0
5	Functions and subroutine calls.	4	1
6	Functions and subroutine calls (cont.). Input/output interfacing.	5	2
7	Input/output interfacing (cont.). Pseudo instructions and literal pools.	6	3
8	Exceptions and interrupts.	7	4
9	**Mid-session exam** (material up to end of week 6).	8	5
Break			
10	Exceptions and interrupts (cont.). Instruction encoding/decoding.	Mid-term exam discussion	6
11	Fixed point representations.	10	7
12	Floating point representations. Industry guest lecture (TBC).	11	Catch-up
13	Review.	12	Lab exam

Laboratory Activities

Lab	Description
0	Introduction to the QVGA Base Board, μVision, and Debugging
1	Data Types, Control Flow, Assembly Programming
2	Functions
3	Inputs and Outputs
4	Digital to Analog Conversion
5	LCD and Touchscreen
6	Interrupts part I
7	Interrupts Part II

Assessment

Laboratory Practical Experiments	25%
Lab Exam	10%
Mid-Semester Exam	10%
Fortnightly online quizzes	5%
Final Exam (2 hours)	50%

COURSE DETAILS

Credits

This is a 6 UoC course and the expected workload is 10–12 hours per week throughout the 13 week semester.

Relationship to Other Courses

This is a 2nd year course in the School of Electrical Engineering and Telecommunications. It is a core subject for students following a BE (Electrical) or (Telecommunications) program.

Pre-requisites and Assumed Knowledge

The pre-requisite for this course is ELEC2141: Digital Circuit Design and COMP1911/COMP1921: Computing 1A/1B. The fundamental knowledge for this course is based on the aforementioned pre-requisites. The students should have a good understanding, in particular, on number systems, C programming and basic computer architecture.

Following Courses

This course is a pre-requisite for TELE3118: Network technologies and ELEC4601: Advanced Digital and Embedded Systems Design.

Learning outcomes

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

1. Demonstrate an understanding of what an embedded system is, and what its main components are.
2. Demonstrate competency in working with and manipulating number systems including fixed-point and floating point binary numbers.
3. Demonstrate the fundamentals of assembly language programming.
4. Demonstrate a competency in using the C programming language for embedded applications.
5. Demonstrate the principles of “good” embedded software design.
6. Demonstrate an understanding of ARM processor architecture.
7. Demonstrate an understanding of mapping high level instructions to low level elementary instructions.
8. Use a mix of C and assembly to design embedded systems containing interrupts, multitasking, and I/O.

This course is designed to provide the above learning outcomes which arise from targeted graduate capabilities listed in ***Appendix A***. The targeted graduate capabilities broadly support the UNSW and Faculty of Engineering graduate capabilities (listed in ***Appendix B***). This course also addresses the Engineers Australia (National Accreditation Body) Stage I competency standard as outlined in ***Appendix C***.

Syllabus

Binary numbers, hexadecimal numbers, signed and unsigned numbers, 2s complement, status flags, ASCII. Programmer's model of ARM7TDMI processor core, registers, fetch-decode-execute cycle, ARM v4T instruction set architecture (ISA). Assembly language programming, data processing instructions, arithmetic operations, logical operations. Memory access instructions, load-store architecture, word and byte addressing, memory alignment, block data transfer. Control flow, conditional branches, loops, jump tables. Function and sub-routine calls, link register, stack, stack frames, register conventions and AAPCS standard. Fixed-point number, floating-point numbers, range and precision, IEEE 754 standard, number distribution, zero- infinity- and NaN-representations. Instructions format, instructions encoding/decoding. Compiler, assembler, linker, loader, assembler directives, pseudo-instructions, object files, relocation tables. Input/output, memory mapped I/O, polling, interrupts. Exceptions, software interrupts, traps, modes of operation, user mode and privileged mode, vector table.

TEACHING STRATEGIES

Delivery Mode

Teaching in this course aims at establishing a good fundamental understanding of the areas covered using:

- Formal face-to-face lectures, which provide you with a focus on the core analytical material in the course, together with qualitative, alternative explanations to aid your understanding;
- Tutorials, which allow for exercises in problem solving and allow time for you to resolve problems in understanding of lecture material;
- Laboratory sessions, which support the formal lecture material and also provide you with practical construction, measurement and debugging skills;
- Fortnight online quiz, which is based on the lecture materials covered in the previous fortnight and keeps you up to date with the lecture material.

Learning in this course

You are expected to attend all lectures, tutorials, labs, and mid-semester exams in order to maximise learning. You must prepare well for your laboratory classes and your lab work will be assessed. In addition to the lecture notes/video, you should read relevant sections of the recommended text. Reading additional texts will further enhance your learning experience. Group learning is also encouraged. UNSW *assumes* that self-directed study of this kind is undertaken in addition to attending face-to-face classes throughout the course.

Tutorial classes

You should attempt all of your problem sheet questions in advance of attending the tutorial classes. The importance of adequate preparation prior to each tutorial cannot be overemphasized, as the effectiveness and usefulness of the tutorial depends to a large extent on this preparation. Group learning is encouraged. Answers for these questions will be discussed during the tutorial class and the tutor will cover the more complex questions in the tutorial class. In addition, during the tutorial class, 1-2 new questions that are not in your notes may be provided by the tutor, for you to try in class. These questions and solutions may not be made available on the web, so it is worthwhile for you to attend your tutorial classes to gain maximum benefit from this course. Tutorials will begin in week 2 and attendance will be taken on a regular basis.

Laboratory program

The laboratory schedule is deliberately designed to provide practical, hands-on exposure to the concepts conveyed in lectures soon after they are covered in class. There will be 8 laboratory tasks. Each week, a new design problem related to the lectured material is presented. You will be required to step through the problem to a complete solution using the guidelines given per lab exercise. **You are strongly encouraged to read over all the material and attempt any code writing before coming to your lab session** as it will allow you to complete the required tasks within the allocated time slot.

Throughout the labs, NXP LPC2478 microcontroller (based on ARM7TDMI-S core) and Keil µVision4 Integrated Development Environment (IDE) will be used. In the first three exercises, tasks will be focused on various fundamental assembly programming techniques: data processing, control flow, and functions. The interaction of the processor with inputs and outputs peripherals and handling interrupts will be the subject of the other five labs. You are required to attend laboratory from Week 4 to Week 13. **Laboratory attendance WILL be kept, and you MUST attend at least 80% of the labs.** A lab exam will be held in Week 13. One bonus lab task (worth 5% bonus marks) may be released after week 9 for students looking for a challenge. Although you are strongly encouraged to attempt the bonus task, there will only be minimal help from the lab demonstrators, allowing you to further demonstrate your ability to analyse and resolve issues independently.

Laboratory Exemption

There is no laboratory exemption for this course. Regardless of whether equivalent labs have been completed in previous courses, all students enrolled in this course must take the labs. If, for medical reasons, (note that a valid medical certificate must be provided) you are unable to attend a lab, you will need to apply for a catch-up lab during another lab time, as agreed by the laboratory coordinator.

ASSESSMENT

The assessment scheme in this course reflects the intention to assess your learning progress through the semester. Ongoing assessment occurs through the lab checkpoints (see lab manual), fortnight quizzes, and the mid-semester exam.

Laboratory Assessment

Laboratories are primarily about learning, and the laboratory assessment is designed mainly to check your knowledge as you progress through each stage of the laboratory tasks. **You are required to purchase a printed lab manual from the school office and bring this manual to every lab session**. Each lab exercise will consist of one checkpoint that will be marked by the lab assessors. Although there is only one checkpoint, you will be marked for a number of tasks. It is important that you save or make records of your works as you will need them for the checkpoint. Demonstrators will be available to help students. Upon completion of a checkpoint, you will be required to write down your student and bench numbers on the Laboratory Queue Sheet and wait for the laboratory assessor to mark your work. The marking sheet will be completely closed off 10 minutes before the end of the lab session. Students may record their name on the sheet for the previous week's checkpoints in the first 30 minutes of the lab if they were not marked in the previous session.

The lab manual provides detailed additional background information that is necessary for the laboratory tasks and it is essential that you read the lab manual, complete any laboratory preparation and attempt any coding you may need before coming to the lab. Otherwise, you will most likely not finish in time to be marked for the checkpoint. Assessment marks will be awarded according to how much of the lab you were able to complete, your understanding of the experiment conducted during the lab and the topic covered by the lab, and the quality of the code you write during your lab work. Detailed information on the guideline for lab assessment is given in your lab manual. Students are required to read and sign the plagiarism declaration at the front of the lab manual. Any instance of clear plagiarism will result in zero mark for the entire laboratory component.

Laboratory Exam

A lab exam will be held in week 13 to test your knowledge on assembly language programming aspects. More details will be provided later during the course.

Mid-Semester Exam

The mid-session examination tests your general understanding of the course material, and is designed to give you feedback on your progress through the analytical components of the course. Questions may be drawn from any course material up to the end of week 6. It may contain questions requiring some (not extensive) knowledge of laboratory material, and will definitely contain numerical and analytical questions. Marks will be assigned according to the correctness of the responses. The mid-session exam will be held in week 9 during the regular 2 hour lecture slot.

Fortnight Quizzes

The fortnight quizzes are designed to gauge your capacity to follow and understand the lecture material and to keep you engaged with course concepts. Each quiz will consist of a number of multiple choice questions and will be marked according to the number of correct answers. The quizzes are a mandatory component of the overall assessment and failure to attempt a quiz will result in no marks being given for the quiz. Each quiz will be available for a period of two weeks and the results per quiz will be published at the end of the period. No late attempts will be permitted. Students must attempt all quizzes to pass this subject.

Final Exam

The exam in this course is a standard closed-book 2 hours written examination, comprising 4 or 5 compulsory questions. University approved calculators are allowed. The examination tests analytical and critical thinking and general understanding of the course material in a controlled fashion. Questions may be drawn from any aspect of the course (including laboratory), unless specifically indicated otherwise by the lecturer. Marks will be assigned according to the correctness of the responses. *Please note that you must pass the final exam in order to pass the course.*

Relationship of Assessment Methods to Learning Outcomes

	Learning outcomes							
Assessment	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Laboratory practical assessments	✓	-	✓	✓	✓	✓	-	✓
Fortnight quizzes	✓	✓	✓	-	-	✓	✓	-
Mid-semester exam	✓	✓	✓	-	✓	✓	✓	-
Lab exam	✓	-	✓	-	✓	✓	-	✓
Final exam	✓	✓	✓	-	✓	✓	✓	✓

COURSE RESOURCES

Textbooks

Prescribed textbook: There is no single textbook for this course

Suggested Reading

- William Hohl, ARM Assembly Language: Fundamentals and Techniques, CRC Press, 2015 (2nd Edition).
- Steve Furber, ARM System On-Chip, 2nd Edition, Addison-Wesley, 2000.

On-line resources

Moodle

As a part of the teaching component, Moodle will be used to disseminate teaching materials, host forums and occasionally quizzes. Assessment marks will also be made available via Moodle: https://moodle.telt.unsw.edu.au/login/index.php.

Mailing list

Announcements concerning course information will be given in the lectures and/or on Moodle and/or via email (which will be sent to your student email address).

OTHER MATTERS

Dates to note

Important Dates available at: https://student.unsw.edu.au/dates

Academic Honesty and Plagiarism

Plagiarism is the unacknowledged use of other people's work, including the copying of assignment works and laboratory results from other students. Plagiarism is considered a form of academic misconduct, and the University has very strict rules that include some severe penalties. For UNSW policies, penalties and information to help you avoid plagiarism, see https://student.unsw.edu.au/plagiarism. To find out if you understand plagiarism correctly, try this short quiz: https://student.unsw.edu.au/plagiarism-quiz.

Student Responsibilities and Conduct

Students are expected to be familiar with and adhere to all UNSW policies (see https://student.unsw.edu.au/guide), and particular attention is drawn to the following:

Workload

It is expected that you will spend at least **ten to twelve hours per week** studying a 6 UoC course, from Week 1 until the final assessment, including both face-to-face classes and *independent, self-directed study*. In periods where you need to need to complete assignments or prepare for examinations, the workload may be greater. Over-commitment has been a common source of failure for many students. You should take the required workload into account when planning how to balance study with employment and other activities.

Attendance

Regular and punctual attendance at all classes is expected. UNSW regulations state that if students attend less than 80% of scheduled classes they may be refused final assessment.

General Conduct and Behaviour

Consideration and respect for the needs of your fellow students and teaching staff is an expectation. Conduct which unduly disrupts or interferes with a class is not acceptable and students may be asked to leave the class.

Work Health and Safety

UNSW policy requires each person to work safely and responsibly, in order to avoid personal injury and to protect the safety of others.

Special Consideration and Supplementary Examinations

You must submit all assignments and attend all examinations scheduled for your course. You should seek assistance early if you suffer illness or misadventure which affects your course progress. All applications for special consideration must be **lodged online through myUNSW within 3 working days of the assessment**, not to course or school staff. For more detail, consult https://student.unsw.edu.au/special-consideration.

Continual Course Improvement

This course is under constant revision in order to improve the learning outcomes for all students. Please forward any feedback (positive or negative) on the course to the course convener or via the online student survey myExperience. You can also provide feedback to ELSOC who will raise your concerns at student focus group meetings. As a result of previous feedback obtained for this course and in our efforts to provide a rich and meaningful learning experience, we have continued to evaluate and modify our delivery and assessment methods including revised lecture notes and tutorials.

Administrative Matters

On issues and procedures regarding such matters as special needs, equity and diversity, occupational health and safety, enrolment, rights, and general expectations of students, please refer to the School and UNSW policies:
https://student.unsw.edu.au/guide
https://www.engineering.unsw.edu.au/electrical-engineering/resources

APPENDICES

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 digital and information literacy and lifelong learning skills through assignment work.

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 engineering 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 engineering 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	✓