



Title:	Embedded Systems Programming APPROVED
Long Title:	Embedded Systems Programming
Module Code:	SOFT8037
Duration:	1 Semester
Credits:	5
NFQ Level:	Advanced
Field of Study:	Computer Science
Valid From:	Semester 1 - 2017/18 (September 2017)
Module Delivered in	1 programme(s)
Module Coordinator:	Sean McSweeney
Module Author:	Sean McSweeney
Module Description:	Embedded software is essential to the operation of many of the machines and devices that power modern society and resides in everything from pacemakers to unmanned aerial vehicles. Embedded software is fundamentally different to application software as it is strongly dependent on the underlying hardware and requires knowledge of both hardware and software to be developed effectively. In this module the learner will develop the knowledge and skills necessary to develop effective embedded software for the Advanced RISC machine architecture.
Learning Outcomes	
<i>On successful completion of this module the learner will be able to:</i>	
LO1	Compare and contrast contemporary microprocessor architectures.
LO2	Evaluate and employ embedded software design best practices.
LO3	Design embedded software systems for a specific microprocessor using an appropriate machine language instruction set.
LO4	Deploy, test and debug software developed for an embedded system.
Pre-requisite learning	
Module Recommendations	
<i>This is prior learning (or a practical skill) that is strongly recommended before enrolment in this module. You may enrol in this module if you have not acquired the recommended learning but you will have considerable difficulty in passing (i.e. achieving the learning outcomes of) the module. While the prior learning is expressed as named CIT module(s) it also allows for learning (in another module or modules) which is equivalent to the learning specified in the named module(s).</i>	
Incompatible Modules	
<i>These are modules which have learning outcomes that are too similar to the learning outcomes of this module. You may not earn additional credit for the same learning and therefore you may not enrol in this module if you have successfully completed any modules in the incompatible list.</i>	
No incompatible modules listed	
Co-requisite Modules	
No Co-requisite modules listed	
Requirements	
<i>This is prior learning (or a practical skill) that is mandatory before enrolment in this module is allowed. You may not enrol on this module if you have not acquired the learning specified in this section.</i>	
No requirements listed	

Module Content & Assessment

Indicative Content

Microprocessor Architectural Overview

Microprocessor review, CISC/RISC/MISC architecture comparison.

Advanced RISC Machine Architecture

Registers, addressing modes, hardware debugging support, applications.

Advanced RISC Machine Assembly

Instruction set, instruction format, data types, pipelining, debugging.

Embedded Software Design

Program structure, design patterns, design validation, concurrent programming, formal embedded software design (FSMs/ASMs/SFGs), security and reliability considerations.

Programming Embedded Devices

Special data types and access, program flow, combining mid and low level languages, device libraries, optimization techniques.

Assessment Breakdown

	%
Course Work	100.00%

Course Work

Assessment Type	Assessment Description	Outcome addressed	% of total	Assessment Date
Written Report	A written report assessing the learners understanding of contemporary micro-controller architectures and the implication of particular architectures on embedded software design.	1,2	30.0	Week 4
Short Answer Questions	An in-class exam that will require the learner to demonstrate understanding of the theoretical content of the module.	1,2,3,4	20.0	Week 6
Project	A project in which the learner will be expected to design, develop, deploy, test and debug an embedded software system for a given design specification.	2,3,4	50.0	Week 13

No End of Module Formal Examination

Reassessment Requirement

Coursework Only

This module is reassessed solely on the basis of re-submitted coursework. There is no repeat written examination.

The institute reserves the right to alter the nature and timings of assessment

Module Workload

Workload: Full Time				
<i>Workload Type</i>	<i>Workload Description</i>	<i>Hours</i>	<i>Frequency</i>	<i>Average Weekly Learner Workload</i>
Lecture	Lecture delivering theory underpinning learning outcomes.	2.0	Every Week	2.00
Lab	Practical computer-based lab supporting learning outcomes.	2.0	Every Week	2.00
Independent & Directed Learning (Non-contact)	Independent & directed learning.	3.0	Every Week	3.00
Total Hours				7.00
Total Weekly Learner Workload				7.00
Total Weekly Contact Hours				4.00

Workload: Part Time				
<i>Workload Type</i>	<i>Workload Description</i>	<i>Hours</i>	<i>Frequency</i>	<i>Average Weekly Learner Workload</i>
Lecture	Lecture delivering theory underpinning learning outcomes.	2.0	Every Week	2.00
Lab	Practical computer-based lab supporting learning outcomes.	2.0	Every Week	2.00
Independent & Directed Learning (Non-contact)	Independent & directed learning.	3.0	Every Week	3.00
Total Hours				7.00
Total Weekly Learner Workload				7.00
Total Weekly Contact Hours				4.00

Module Resources

Recommended Book Resources

- Elecia White 2011, *Making Embedded Systems: Design Patterns for Great Software*, O'Reilly Media [ISBN: 9781449302146]

Supplementary Book Resources

- Chris Hobbs 2015, *Embedded Software Development for Safety-Critical Systems*, CRC Press [ISBN: 9781498726702]
- Joseph Yiu 2013, *The Definitive Guide to ARM® Cortex-M3 and Cortex-M4 Processors*, 3rd Ed., Newnes [ISBN: 9780124080829]
- James W. Grenning 2011, *Test Driven Development for Embedded C (Pragmatic Programmers)*, Pragmatic Bookshelf [ISBN: 9781934356623]
- Gary Stringham 2009, *Hardware/Firmware Interface Design: Best Practices for Improving Embedded Systems Development*, Newnes [ISBN: 9781856176057]

This module does not have any article/paper resources

Other Resources

- Website: *Embedded Development Articles*
<http://www.embedded.com/Development>
- Website: *ARM Embedded Resources*
<https://developer.arm.com/embedded>
- Website: *The Ganssle Group*
<http://www.ganssle.com/articles.htm>
- Website: *IEEE Embedded Software Letters - various*
<http://ieeexplore.ieee.org/xpl/RecentIss ue.jsp?punumber=4563995>

Module Delivered in

Programme Code	Programme	Semester	Delivery
CR_KDNET_8	Bachelor of Science (Honours) in Computer Systems	8	Mandatory