Liverpool John Moores University

Warning: An incomplete or missing proforma may have resulted from system verification processing

Title:	Microprocessors and Software
Status:	Definitive
Code:	**4002ELE** (120034)
Version Start Date:	01-08-2016
Owning School/Faculty:	Built Environment
Teaching School/Faculty:	Built Environment

Team	Leader
Mahamoud Ahmed	Y
Princy Johnson	
Andrew Attwood	

Academic Level:	FHEQ4	**Credit Value:**	20	**Total Delivered Hours:**	74
Total Learning Hours:	200	**Private Study:**	126		

Delivery Options
Course typically offered: Standard Year Long

Component	Contact Hours
Lecture	24
Practical	48

Grading Basis: 40 %

Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Exam	Exam	Exam	70	2
Report	AS1	Low Level Programming	15	
Report	AS2	Operating systems and Development	15	

Aims

Provide an overview of the operation of modern microprocessors and the

mechanisms used to represent and process information. Design and implement applications written in both low level and high level languages.

Learning Outcomes

After completing the module the student should be able to:

1	Describe the techniques applied to represent information within a Microprocessor.
2	Identify the fundamental components of a Microprocessor.
3	Describe the instruction set of a computer contrasting RISC and CISC approaches.
4	Produce an application that demonstrates an understanding of the registers that constitute a Microprocessor.
5	Describe the role of modern Operating Systems in embedded, mobile, desktop and server environments.
6	Produce an application that utilises a high level language and interacts with an operating system.

Learning Outcomes of Assessments

The assessment item list is assessed via the learning outcomes listed:

Exam	1	2	3	5
Low Level Programming	5	6		
Operating Systems & Dev	3	4		

Outline Syllabus

Binary, HEX, 2s Compliment, big and little endian, IEEE 754, ASCII, UNICODE.

Processor core and cache hierarchies, Buses, Memory Organisation, Cache Coherency, Multicore.
X86, ARM instruction Sets, 80% 20% ratio.

Application Scheduling, Security, Interrupt Handling, Libraries, Communications.
Variables, Arrays, Iteration, Selection, I/O, Structures, Dynamic memory, flow charts.

Assembly Programming, Memory transfers, interaction with IO.

Learning Activities

Lecture, demonstration and practical activities applying topics discussed.

Notes

This module introduces the fundamentals of Computer architecture and the development of High and low level software.