| | |
|---|---|
| **Module Title**<br>**Real-Time Embedded Systems**<br><br>**Keywords**<br>Real-time Systems, Embedded Systems,<br>Microcontrollers, Interface Design | Reference    EN5503<br>SCQF          SCQF<br>Level                 11<br>SCQF Points    15<br>ECTS Points    7.5<br>Created    May 2002<br>Approved    January 2010<br>Amended    August 2011<br>Version No.          2 |

# This Version is No Longer Current

## Prerequisites for Module

Computer architecture
(EN3540) or equivalent

## Corequisite Modules

None.

## Precluded Modules

None.

## Aims of Module

To enable the student to
develop the skills and
knowledge involved in the
design and implementation of
real-time embedded systems.

## Learning Outcomes for Module

Interface Design: bus systems,
address decoding, registers and
buffering, development of interface
driver software.

## Indicative Student Workload

| | Full<br>Time | Part<br>Time |
|---|---|---|
| *Contact Hours* | | |
| Laboratories | 26 | 26 |
| Lectures/Tutorials | 12 | 12 |
| | | |
| *Directed Study* | | |
| Directed Self<br>Study/Coursework<br>preparation | 37 | 37 |
| | | |
| *Private Study* | | |
| Private Study | 75 | 75 |

## Mode of Delivery

The module is taught using a
structured programme of lectures,
tutorials, laboratories and

On completion of this module, students are expected to be able to:

1. Design interfaces between microprocessors and peripheral devices.
2. Design and implement software for real-time embedded systems which control and monitor external hardware.

## Indicative Module Content

Microcontroller based systems: architecture, integrated peripherals, timers, serial peripheral interfaces, exception handling. Real-time systems: Multi-tasking, real-time operating systems. Inter-task communication and synchronisation. Resource scheduling, allocation and protection, structures of queues and tables, device interfaces, task scheduling. Methods and tools for the development of real-time systems. Embedded Systems: Nature of embedded systems, applications, hardware requirements, case studies, impact on software development. Software development process. Debugging support.

Fuzzy Logic: for real-time microcontroller based systems.

student-centred learning. The development of a practical real-time embedded system will form a major element of the practical work.

## Assessment Plan

|  | Learning Outcomes Assessed |
|---|---|
| Component 1 | 1 |
| Component 2 | 2 |

Component 2 is a mini project which involves the development of high-level software for real-time applications implemented on micro-controller systems. (Weighting 70%)

Component 1 is a coursework which involves the design and development of a microprocessor interface. (Weighting 30%)

## Indicative Bibliography

1. CADY, F. M., 2007. Software and Hardware Engineering. 2nd ed. Oxford University Press.
2. VALVANO, J. W., 2006. Developing Embedded Software in C Using IC11/IC12/Metrowerks. Brooks/Cole Pub. Co..
3. BARRY, R., 2010. FreeRTOS eBook Standard Edition: Using FreeRTOS Real Time Kernel - A Practical Guide. Real Time Engineering Ltd..