

## Liverpool John Moores University

Title: Embedded Systems  
Status: Definitive  
Code: **6504SEPA** (129469)  
Version Start Date: 01-08-2021

Owning School/Faculty: Computer Science and Mathematics  
Teaching School/Faculty: Beaconhouse IC Islamabad

Team	Leader
David Lamb	Y
Glyn Hughes	

**Academic Level:** FHEQ6      **Credit Value:** 20      **Total Delivered Hours:** 46  
**Total Learning Hours:** 200      **Private Study:** 154

### Delivery Options

Course typically offered: Semester 2

Component	Contact Hours
Lecture	11
Workshop	33

**Grading Basis:** 40 %

### Assessment Details

Category	Short Description	Description	Weighting (%)	Exam Duration
Artefacts	AS1	Design, implementation, and evaluation of embedded system	50	
Exam	AS2	Examination	50	2

### Aims

*To provide an overview of designing and engineering embedded systems, including high-level hardware architectures and software systems with references to architectures, communication and synchronisation.*

*To investigate the development of a connected embedded system and appropriate support software services.*

## Learning Outcomes

After completing the module the student should be able to:

- 1 Critically survey technologies and methods used in embedded systems design and development.
- 2 Solve execution issues intrinsic to embedded architectures and develop software for embedded systems alongside other software systems.
- 3 Appraise communications standards and techniques used in embedded systems.
- 4 Critically evaluate operational issues in embedded and concurrent systems.

## Learning Outcomes of Assessments

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

Embedded system	1	2
Examination	3	4

## Outline Syllabus

*Embedded Systems: Basic Architectures / Issues*

- baremetal, superloops and real-time operating systems
- interrupt-driven execution

*GPIO – Getting data in / out and electronic-software interfacing*

- basic related interfacing/electronics concepts
- analogue-digital conversion and PWM
- noise reduction/filtering

*Memory and storage: resource constrained systems*

- Programmer-centred memory management : stack, heap and global/statics
- Smart pointers and automatic release / garbage collection

*Serial over GPIO – SPI, I2C, flash/SD card storage*

- Bus systems and line arbitration / access
- shared clock / asynchronous vs. synchronous systems
- Hardware-support and bit-banged (software-defined) implementations

*WiFi and Internet connectivity*

- common library and driver support
- socket programming and stream parsing
- RESTful server and smart client provisioning
- Automatic update mechanisms

*Pattern-based embedded software design*

- Superloop and/vs Strategy / State / State Table / Scheduling
- Façade / Proxy / Mediator / interfacing

*Concurrent vs. Serial execution*  
*-Liveness and Deadlock*  
*-Data Races and Atomicity*

## **Learning Activities**

Didactic, lecture-based theory and applied examples.  
Workshop activities exploring implementation and engineering challenges present in developing microcontroller solutions.  
Problem-based learning centred on coursework assignment tasks.

## **Notes**

This module broadens a Software Engineer's horizons to include system and software development for embedded systems, with consideration of the Internet of Things. Students are required to have considerable high-level programming knowledge by level 6; this will be expanded to consider working with lower-level architectural concerns and development software for "baremetal" systems.