<table>
<thead>
<tr>
<th>Module Code</th>
<th>CS7004</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Module Name</strong></td>
<td>Embedded Systems</td>
</tr>
<tr>
<td><strong>Module Short Title</strong></td>
<td>N/a</td>
</tr>
<tr>
<td><strong>ECTS weighting</strong></td>
<td>5 Credits</td>
</tr>
<tr>
<td><strong>Semester/term taught</strong></td>
<td>Michaelmas Term</td>
</tr>
<tr>
<td><strong>Contact Hours</strong></td>
<td>Lecture hours: 21</td>
</tr>
<tr>
<td></td>
<td>Lab hours: 12</td>
</tr>
<tr>
<td></td>
<td>Tutorial hours: 0</td>
</tr>
<tr>
<td></td>
<td>Total hours: 33</td>
</tr>
<tr>
<td><strong>Module Personnel</strong></td>
<td>Lecturing staff: Jonathan Dukes</td>
</tr>
</tbody>
</table>

### Learning Outcomes

On successful completion of the module, students will be able to:

1. Describe the structure and operation of embedded systems, with particular emphasis on embedded system software, including embedded operating systems.
2. Evaluate and describe the requirements of an embedded system for a specific application.
3. Design, develop, test and evaluate small-scale embedded systems for a specific application using real-world hardware platforms (LPC2468 development board), programming tools (Keil ARM MDK) and systems software (FreeRTOS).
4. Describe the operation and make use of peripheral modules and devices, including, for example, interrupt controllers, serial communication devices, LCD and touchscreen controllers and analogue-to-digital converters.
5. Use appropriate literature, documentation and other resources to expand the knowledge that they can apply to the design, development and evaluation of embedded systems.

### Module Learning Aims

The module will give students the opportunity to gain the knowledge and skills necessary to begin developing embedded systems software. Students taking the module will study a real embedded hardware platform in depth (based on the widely used ARM7TDMI microcontroller) and will use this platform in conjunction with industry-standard software tools to develop embedded systems of moderate complexity. Topics covered will be in the broad areas of computer architecture, systems software and I/O. Throughout the module, students will be given opportunities to consider issues of particular relevance in embedded systems design (e.g. development cost, power, performance and reliability).

The final four weeks of the module are devoted to practical project work, which is a core component of the module. Each student will engage in the design, development, testing and evaluation of a small-scale embedded system, putting into practice the knowledge gained in the module.

The module assumes that students have substantial prior programming experience and a basic knowledge of microprocessor systems and computer architecture.
### Module Content

1. **Computer Architecture**
   - Microcontroller basics review
   - Memory mapped I/O
   - Exceptions and interrupts
   - Interrupt-driven application development

2. **Systems Software**
   - C Programming review; tools and techniques for embedded systems development
   - Operating systems for embedded systems
   - Task scheduling
   - Inter-process communication

3. **I/O**
   - Device interconnects (e.g. I²C)
   - A/D, D/A conversion

4. **Reliability**
   - Case-studies
   - Reliability models

### Recommended Reading List
Details of recommended texts, literature, documentation and other resources will be made available on the module website. ([See mymodule.tcd.ie](http://mymodule.tcd.ie))

### Module Pre Requisite

### Module Co Requisite

### Assessment Details
Assessment is by written examination (contributing 50% towards the overall mark) and practical project work (contributing 50% towards the overall mark). Supplemental assessment is by examination only (100%).

### Module approval date
N/a

### Approved By
N/a

### Academic Start Year
N/a

### Academic Year of Data
N/a