|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TSC Category** | Development and Implementation | | | | | |
| **TSC Title** | Embedded Systems Programming | | | | | |
| **TSC Description** | Program an embedded system using permitted programming interfaces provided by the system to support creation of devices that do not operate on traditional operating systems | | | | | |
| **TSC Proficiency Description** | **Level 1** | **Level 2** | **Level 3** | **Level 4** | **Level 5** | **Level 6** |
|  |  |  | **ICT-DIT-4007-1.1** | **ICT-DIT-5007-1.1** |  |
|  |  |  | Develop software applications and drivers to run in embedded systems, including rapid prototyping as well as the implementation of embedded software or firmware | Plan end to end process of incorporating embedded systems in hardware and devices, validating and optimising embedded software systems in different application areas |  |
| **Knowledge** |  |  |  | * Low-level programming languages and software syntax * Embedded systems software architectures and interfaces * Relevant operating systems, drivers and microcontrollers * Control requirements for embedded system * Tools for development and debugging of embedded software, including editor, assembler and cross assembler * Rapid prototyping techniques * Types and uses of sensors, electrical and electronic devices and components, and electrical wiring systems * Types and characteristics of microcontrollers, programming devices and programming software * Data collection, transmission and communication protocols * Security considerations for sensor networks | * Suitability and application of different programming languages for different purposes or contexts * Embedded firmware and software engineering principles * Types, characteristics and operating principles of binary and analogue input and output devices * Hardware design tools, techniques and hardware control programming * Range of software development and software configuration management tools * Operating System coding techniques, interfaces and hardware subsystems * Schematics, component data sheets and electronic test equipment * Troubleshooting on embedded targets * System optimisation techniques for both hardware and software performance * Sensor properties and their application to electronic system programming |  |
| **Abilities** |  |  |  | * Develop software applications and drivers to run in an embedded operating system * Interpret hardware and software communication and control requirements * Conduct rapid prototyping of embedded control systems * Implement embedded firmware or software drivers and applications on a microcontroller * Employ hardware and software tests to test and analyse embedded programs and digital electronics * Test logic connectivity and integrity of physical designs * Verify embedded software designs according to quality and regulatory guidelines * Manage all records and metrics related to embedded software development process * Identify root cause of issues related to embedded software | * Plan end to end process from inception to deployment of embedded systems or microcontrollers for use in hardware and devices * Define hardware and software communication and control requirements * Align embedded system development with best practices for coding, reuse and portability * Introduce new and emerging coding techniques or languages suitable for embedded systems programming * Review coding, testing and design criteria * Create technical manuscripts of embedded software or firmware operation * Optimise embedded software systems in different application areas * Solve problems using electronic circuits, control programs and software-hardware interface |  |
| **Range of Application** |  | | | | | |