ESE-3025 Embedded Real Time Operating Systems

Computer Studies

Course Number:	**Co-Requisites:**	**Pre-Requisites:**
ESE-3025	N/A	ESE-1025 and ESE-2025
Prepared by:	Takis Zourntos, Outline Creator	
Approved by:	Chris Slade, Dean School of Business	
Approval Date:	Tuesday, May 26, 2020	
Approved for Academic Year:	2020-2021	
Normative Hours:	75.00	

Course Description

This course provides an overview of multi-tasking operating systems and an in-depth study of real-time operating system (RTOS) programming techniques, building on the software foundations of ESE 1025 and ESE 2025. The course emphasizes hands-on RTOS development using Amazon FreeRTOS in embedded C on modern 32-bit cortex-M embedded platforms (e.g., NXP LPC1769 and LPC54114 MCUs). Topics covered include: preemptive versus cooperative scheduler operation; tick rate and time slicing; critical code; fixed, dynamic and hybrid task priority allocation; application-specific considerations; power management tactics; semaphores, mutexes and queues; debugging strategies; performance estimation (resources/bandwidth/queue-depth and latency).

Course Learning Outcomes/Course Objectives

1. **Describe the high-level operation of real-time embedded systems.**
   - 1.1 Contrast cooperative and preemptive modes of multi-tasking operating systems;
   - 1.2 Explain the differences between hard and soft real-time operation;
   - 1.3 Explain the concepts of deadline and predictability;
   - 1.4 List the typical parameters of a real-time task.
   - 1.5 Discuss latency with respect to real-time operating systems and the factors affecting it;
   - 1.6 Explain jitter with respect to real-time operating systems;
   - 1.7 Explain the concept of a real-time task, the meaning of task priority and time slicing;
   - 1.8 Discuss the RTOS “tick” and the tradeoff between RTOS responsiveness and power consumption;
   - 1.9 Discuss the idle task and its practical implications;
   - 1.10 Discuss the concept of scheduler overhead;

1.11 Discuss various types of scheduling algorithms, including Fixed Priority, Rate Monotonic, Earliest Deadline First, Round Robin and Best Effort modes of operation.

2. Develop practical RTOS applications on an actual embedded target

2.1 Discuss the operation of a Linux preemptive kernel;

2.2 List the preemption modes available under Linux;

2.3 Configure an RTOS development environment using the MCUXpresso IDE and cross toolchain from NXP;

2.4 Compose a FreeRTOS application template from memory, including generic tasks, main () function, xTaskCreate () calls and vStartScheduler ();

2.5 List all the library components of FreeRTOS and their purpose;

2.6 Provide a timing diagram of a hypothetical FreeRTOS application, given fixed task priorities, including several tasks with equal priority;

2.7 Discuss the meaning of Running and Not Running states, and meaning of Suspended , Ready and Blocked sub-states.

2.8 Explain the difference between a task delay implemented using a for (;;) loop, and a task delay based on vTaskDelay () or vTaskDelayUntil ();

2.9 Design FreeRTOS applications that use GPIO interfacing on an embedded target;

2.10 Describe the effects of various RTOS system parameters (found in FreeRTOSConfig.h) on the operation of a FreeRTOS application;

2.11 Design a FreeRTOS application that makes use of the FreeRTOS Queue API;

2.12 Design a FreeRTOS application that makes use of mutexes in FreeRTOS;

2.13 Discuss the meaning and implications of priority inversion;

2.14 Design a FreeRTOS application that uses semaphores and timers in FreeRTOS;

2.15 Debug a FreeRTOS application using the MCUXpresso/Eclipse cross toolchain;

2.16 Explain the operation of the Programmable Real-Time Unit Subsystem and Industrial Communication SubSystem (PRU-ICSS) found on TI-based embedded MCUs;

2.17 Write a real-time application under Linux using the PRU-ICSS and the enhanced (high-speed) GPIOs on a TI-based embedded platform.

3. Estimate the performance of an embedded real-time system

3.1 Discuss execution time, latency and preemption delay;

3.2 Discuss how processing (number and types of cores available), memory (quantity, access times, TCM), and I/O (interconnect, latency), affect the performance of a real-time system;

3.3 Describe measures available to the embedded systems designer to improve RTOS responsiveness and power consumption;

3.4 Estimate best-case, average-case, and worst-case task execution times;

3.5 Determine bounds on the RTOS tick.

Relationship to Vocational Learning Outcomes

This course provides the opportunity for you to achieve the following Program Vocational Learning Outcomes (VLO) which will be taught and evaluated at an taught (T), assessed (A) or culminating peformance (CP) level:

EMBT - Embedded Systems Engineering Design

VLO 1	Select appropriate design tools to meet quality standards and customer requirements when developing embedded systems products. (T, A)
VLO 2	Solve systems design problems through integration of hardware, software, sensors and actuators. (T, A)
VLO 3	Design, develop, test, configure and maintain embedded systems. (T, A)
VLO 5	Communicate effectively with diverse teams to disseminate ideas, requirements, implementations, findings and outcomes to complete embedded systems projects. (T, A)

Learning Resources

a. Required

Real-Time Embedded Components and Systems with Linux and RTOS by Sam Siewert and John Pratt; Mercury Learning and Information, 2nd ed.

b. Supplemental

None

Student Evaluation

Tests - 45% (3 equally weighted)

3 tests @ 15% each

Assignments - 15%

1 assignment @7%

1 assignment @8%

Laboratory Session - 40%

4 Lab Assignments equally weighted at 10% each

Grade Scheme

The round off mathematical principle will be used. Percentages are converted to letter grades and grade points as follows:

Mark (%)	Grade	Grade Point	Mark (%)	Grade	Grade Point
94-100	A+	4.0	67-69	C+	2.3
87-93	A	3.7	63-66	C	2.0
80-86	A-	3.5	60-62	C-	1.7
77-79	B+	3.2	50-59	D	1.0
73-76	B	3.0	0-49	F	0.0
70-72	B-	2.7			

Prior Learning Assessment and Recognition

Students who wish to apply for prior learning assessment and recognition (PLAR) need to demonstrate competency at a post-secondary level in all of the course learning requirements outlined above. Evidence of learning achievement for PLAR candidates includes:

- Other: If yes has been selected, you may choose to contact the Counselling Department for advice on Prior Learning Assessment.

Course Related Information

The course is designed primarily to deliver more emphasis on hands on experience via laboratory sessions.

College Related Information

Academic Integrity

Lambton College is committed to high ethical standards in all academic activities within the College, including research, reporting and learning assessment (e.g. tests, lab reports, essays).
The cornerstone of academic integrity and professional reputation is principled conduct. All scholastic and academic activity must be free of all forms of academic dishonesty, including copying, plagiarism and cheating.

Lambton College will not tolerate any academic dishonesty, a position reflected in Lambton College policies. Students should be familiar with the Students Rights and Responsibilities Policy, located at lambtoncollege.ca. The policy states details concerning academic dishonesty and the penalties for dishonesty and unethical conduct.

Questions regarding this policy, or requests for additional clarification, should be directed to the Lambton College Student Success Department.

Students with Disabilities

If you are a student with a disability please identify your needs to the professor and/or the Accessibility Centre so that support services can be arranged for you. You can do this by making an appointment at the Accessibility Centre or by arranging a personal interview with the professor to discuss your needs.

Student Rights and Responsibility Policy

Acceptable behaviour in class is established by the instructor and is expected of all students. Any form of misbehaviour, harassment or violence will not be tolerated. Action will be taken as outlined in Lambton College policy.

Date of Withdrawal without Academic Penalty

Please consult the Academic Regulations and Registrar's published dates.

Waiver of Responsibility

Every attempt has been made to ensure the accuracy of this information as of the date of publication. The content may be modified, without notice, as deemed appropriate by the College.

Students should note policies may differ depending on the location of course offering. Please refer to campus location specific policies:
LAMBTON COLLEGE POLICIES – applicable to all Lambton College students.

- Student Rights & Responsibilities & Discipline policy (2000-5-1)
- Test & Exam Writing Protocol (2000-1-6)

- Evaluation of Students (2000-1-3)
- (https://www.lambtoncollege.ca/custom/Pages/Policies/Policies.aspx)

CESTAR COLLEGE:

- https://www.lambtoncollege.ca/Programs/International/Lambton_in_Toronto/Student_Policies/

QUEENS COLLEGE:

- https://www.lambtoncollege.ca/Programs/International/Lambton_in_Mississauga/Student_Policies/

Note: It is the student's responsibility to retain course outlines for possible future use to support applications for transfer of credit to other educational institutions.