ESE-3005 Embedded Systems Architecture II

Computer Studies

Course Number:	Co-Requisites:	Pre-Requisites:
ESE-3005	N/A	ESE-2005
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

In this course students build on their knowledge gained from introductory courses to design and implement complete embedded applications. Applications combine multiple communication interfaces and hardware peripherals to provide complete embedded solutions that students may encounter as a professional. Specifically, applications make use of various peripherals such as touch screens, cameras, microphones and speakers, GPIOs, timers, GPS modules, Bluetooth, WiFi, and ADC/DACs.

Course Learning Outcomes/Course Objectives

1. **Apply advanced services in a multi-tasking OS environment on an embedded target**
   - 1.1 Create system and startup services with systemd software suite;
   - 1.2 List the benefits and drawbacks of systemd
   - 1.3 Create a systemd script that automatically starts an application at boot up;
   - 1.4 List the software components that comprise the systemd suite of software;
   - 1.5 Create udev rules that perform actions based on system events;
   - 1.6 List the benefits of using the udev device manager
   - 1.7 Monitor emitted device events using the udevadm tool;
   - 1.8 Create a udev rule to mount a USB mass storage drive

2. **Create a udev rule to mount a USB mass storage drive**
   - 2.1 Build a Linux image using the Yocto Project;
   - 2.2 Configure a cross-compiler and toolchain using the Yocto project;
   - 2.3 Configure Eclipse to work with a cross compiler and toolchain;

2.4 Setup a remote execution and remote debugging environment using Eclipse and gdbserver.

3. Design and implement an embedded GUI application that makes use of thetouchscreen
   - 3.1 Add a touchscreen as an alternate display using U-Boot arguments;
   - 3.2 Add a touchscreen as the primary display by editing the device tree;
   - 3.3 Build and install the QT toolchain using BitBake;
   - 3.4 Install and configure QT creator;
   - 3.5 Create a GUI application using QT creator.

4. Design and implement an embedded application that makes use of the Cameramodule, microphone and audio output
   - 4.1 Use GStreamer to display a test pattern using VLC.
   - 4.2 Connect camera hardware to the i.MX Linux development board;
   - 4.3 Display streaming images from an embedded camera device to an Ethernet connected host running VLC.
   - 4.4 Use the ‘arecord’ application to record and playback audio recorded from the ‘line-in’ microphone jack of a i.MX development board;
   - 4.5 Use the ‘aplay’ application to play sound files from a speaker connected to the ‘line-out’ jack of the i.MX development board;
   - 4.6 Write an application that displays video from the camera and records and plays back audio from the microphone.

5. Design and implement an application that collects data using a Bluetooth sensor.
   - 5.1 Setup a Bluetooth connection between two embedded i.MX boards;
   - 5.2 Transfer files between two Bluetooth-connected i.MX boards using obex transfer;
   - 5.3 Setup a WiFi connection on a i.MX board;
   - 5.4 Transfer files using scp between two WiFi connected i.MX boards;
   - 5.5 Write an application to transfer data between two Bluetooth-connected i.MX boards and then between two WiFi connected boards in a daisy-chained fashion;
   - 5.6 Setup a WiFi Access Point on a i.MX board that forwards packets via Ethernet;
   - 5.7 Write an application that collects data from Bluetooth receivers, forwards the data over WiFi to an AP which then stores the data on a PC connected to the i.MX over Ethernet.

6. Design and implement an application that collects GPS data over serial from a GPS module.
   - 6.1 Setup and configure a GPS module using a standard UART on the i.MX board;
   - 6.2 Read the serial data using PuTTY or minicom running on the i.MX board;
   - 6.3 Write an application that parses the NMEA sentences received from the GPS module and displays the current location information;
   - 6.4 Use the PPS signal from a GPS module connected through GPIO to synchronize

the system clock;

6.5 Modify the Linux kernel using the BitBake menuconfig command and add support for a PPS source through a GPIO;

6.6 Edit the i.MX board device tree and add the pps-gpio node;

6.7 Build a custom built device tree and corresponding Linux image using BitBake;

6.8 Add chronyd and gpsd to a BitBake recipe;

6.9 Synchronize the system time using Chronyd and GPSd on a custom built Linux image.

7. **Implement a userspace driver that uses the Linux Gadget framework that uses theUSB on-the-go port and presents itself as a device to a host**

7.1 Build a custom Linux image using the BitBake menuconfig command and enable Linux support for USB devices using the GadgetFS USB gadget filesystem;

7.2 Edit a Linux device tree to set an OTG port from ‘Host’ to ‘Peripheral’ and build the resulting Linux image;

7.3 Setup and enable a USB GadgetFS device using the Linux command line;

7.4 Write an application that sends and receives bytes over a USB connection from a USB host to the i.MX development board configured as a USB device.

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

http://www.variwiki.com/index.php?title=VAR-SOM-MX6_Yocto_Jethro_New_R4

b. Supplemental

None

Student Evaluation

Assignments 60%
8 @ 7.5% each
Laboratory Sessions 40%
8 @ 5% 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 and assignments. Each lab contains the required preliminary information for an end of unit assignment.

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.