**REVISED AMENDMENT TO EXISTING SYLLABUS FOR ECE 315**
**Date Revision Shared with Students:** 3/27/2020

In response to the 2020 COVID-19 global pandemic, all courses at the University of Alberta moved to remote delivery, effective March 17, 2020. Subsequently, a decision was made to adjust the grading scheme for all University of Alberta courses for the Winter 2020 term to award one of the following grades: CR (credit) or NC (no credit) as indicated in the *University Calendar.*

The changes were put in place to ensure equity among students, preserve academic integrity in extraordinary circumstances, and to manage the issues presented by an uncertain future, including the impact on our students and human resources.

To ensure clarity for students, all Winter 2020 course outlines for both undergraduate and graduate courses are to be updated using this template to outline changes to exams, assessments and grading and shared with students.

**Course Code:** ECE 315

**Course Title:** Computer Interfacing

**Instructor Name:** Dr. Bruce F. Cockburn

**Instructor Contact Information:**

Electronic mail: cockburn@ualberta.ca

**Revised course assessment plan** (Please list all assessed assignments and weighting below)

The assessed coursework in the revised grading scheme will comprise the following four components:
1) Four graded assignments, weighted equally, for a total of 16% (formerly worth 10%)
2) Five laboratory exercises, weighted equally, for a total of 32% (formerly worth 20%)
3) One proctored midterm examination worth 32% (formerly worth 20%)
4) One take-home final assessment, worth 20% (replacing a proctored final exam worth 50%)

The final assessment will be a problem set distributed to the class through eClass in the form of an editable PDF document and a Google Docs document on Wednesday, April 15 at 9 am. The completed assessment document must be submitted using the on-line submission system of eClass with a deadline of Friday, April 17 at 11 am (the same completion time as for the originally planned final examination). Students who have registered with Accessibility Resources will be provided accommodations (e.g. time multipliers) equivalent to those that they were scheduled to receive for the final exam.

The marks in the four components will be combined into one percentage and mapped to a letter grade, using the same procedure as used in ECE 315 in previous terms. Students shall be awarded a grade of CR in the event that their grade under the letter grading system would have been "D" or above considering the new weights in the revised grading scheme. Students who would receive a grade of NC under this scheme will be offered an oral exam as an additional means of assessment. A grade of incomplete (IN) will be awarded when there is insufficient assessment information to make a decision between CR and NC.

IMPORTANT NOTES:

- If students require alterations in their approved accommodations, please contact arrec@ualberta.ca.
- The changes with respect to the mode of delivery of instruction and assessment and the changes to grading regulations do not constitute grounds for an appeal under academic appeal policies (i.e. grade or academic standing appeals may not be advanced on the grounds of these changes).
- When Incomplete (IN) status is assigned instructors are required to communicate with students about the mechanisms and requirements that will be implemented to determine the student final grade of CR or NC. For more information, including the time to complete the remaining course work see the University Calendar.

# ECE 315 COMPUTER INTERFACING (LEC B1 Wi20)



## Course Syllabus

### ECE 315 - Computer Interfacing

**UNIVERSITY OF ALBERTA**
**DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING**

**Course Outline Winter 2020**

**Lecture Section:**

LEC B1 (class number 82494), held on Mondays, Wednesdays and Fridays, from 12:00 to 12:50, in the South Academic Building room SAB 325. The South Academic Building lies to the east of the Student Union Building, across the quad.

**Course Text:**

There is no required textbook. The main references will be the lecture slides (available at the Lecture Materials section of this eClass site) and a wealth of on-line documentation (see the separate eClass site for the ECE 315 laboratory). Supplementary reference books are given below.

**List of Topics:**

| | | |
|---|---|---|
| **Ch. 1:** Introduction to Computer Interfacing | 0.4 | How ECE 315 fits in with respect to other courses. The role of embedded systems. Examples of embedded systems. The ECE 315 laboratory embedded system. Real-time embedded systems. Hard and soft real time. Performance metrics for embedded systems. |
| **Ch. 2:** Review of Technical Background | 0.6 | The nature of the computer interfacing problem. Types of interfaces. Embedded software development and software engineering. Assembly language vs. high-level programming languages. Review of the C language, with an emphasis on pointers. Review of C++ objects including class definitions and member functions. Dynamic memory allocation. |
| **Ch. 3:** Review of the Freescale ColdFire Microprocessor | 1.0 | Review of microprocessor terminology. Microcomputer architecture. Major kinds of microcomputer memory. Fetch-decode-execute cycle. Memory-mapped input/output. Microcontroller architecture. The ColdFire family of CPUs. The MCF54415-based MOD54415 microcomputer. User and supervisor modes. CPU registers. Memory organization and native data types. Big Endian and Little Endian address conventions. ColdFire machine language instruction format. Addressing modes. Major classes of machine instructions. Subroutines and parameter passing methods. Exceptions, traps and interrupts. Interrupt priority levels and masking. Interrupt-drive I/O. User and autovectored interrupts. The exception vector table. Exception handling. The reset exception. |
| **Ch. 4:** Software Concepts for Embedded Systems | 0.7 | Managing the complexity of software development. Foreground-background systems. Sleep modes. Tasks, processes and threads. Context switching. Software architectures for sharing CPU time in a multitasking system. Timing slicing. State-driven code. Preemptive and cooperative multitasking. Task state transitions in a preemptive multitasking system. Real-time operating systems and kernels. Mechanisms for protecting critical sections. Semaphores and message passing. Deadlock and deadling avoidance. Figures of merit for real-time kernels. |
| **Ch. 5:** The MicroC/OS Real-Time Kernel | 1.0 | A brief history of MicroC/OS. System architecture using MicroC/OS. Task priority conventions. A "Hello World!" example MicroC/OS system. Creating MicroC/OS application tasks. Basic MicroC/OS functions that cause task blocking and unblocking. Predifined MicroC/OS data type definitions. Functions for task creation, delayed task execution, semaphore management, and queue management. Macros for protecting critical sections. Interrupt handling in MicroC/OS. |
| **Ch. 6:** Serial Interfacing Using RS-232C, Ethernet and SPI | 1.0 | Asynchronous serial interfacing using RS-232C. Hardware flow control signals. Generic architecture of a serial interface. Standard data control methods: unconditional CPU-controlled, busy waiting, interrupt-driven, and direct memory access (DMA). The UART interface in the MCF54415. The Ethernet Local Area Network (LAN) standard. The evolution of Ethernet. Review of the Serial Peripheral Interface (SPI) standard. |

| | | |
|---|---|---|
| **Ch. 7:** The Ethernet Interface of the NetBurner MOD5234 Microcomputer | 1.0 | The media access control (MAC) and physical layer transceiver (PHY) layers. The MAC-PHY interface in the MOD54415. The Micrel KS8721BT PHY. The Fast Ethernet Controller (FEC) interface in the MCF54415. Survey of the major FEC registers. The transmit and receive data buffer management system. Software processing steps in the transmit and receive directions. |
| **Ch. 8:** Data Communications Concepts | 1.3 | Data communications protocols. Layered communications interfaces. Service types at a protocol layer: requests, indications, responses and confirmations. The seven-layer Open Systems Interconnection Reference Model (OSI-RM) protocol stack. Routers, bridges, repeaters and hubs. A history of large-scale data networks and the emergence of the global Internet. The User Datagram Protocol (UDP), Transmission Control Protocol (TCP) and the Internetworking Protocol (IP). Brief survey of standard TCP/IP application protocols: FTP, TELNET, HTTP, DNS, SMTP. Using sockets to access TCP/IP. Client-server communication. Simple client and server tasks in Unix. |
| **Ch. 9:** Interfacing MicroC/OS to TCP/IP | 0.7 | Interfacing small embedded systems to TCP/IP. NetBurner's client-server architecture for interfacing MicroC/OS to TCP/IP. An example echo server in MicroC/OS. Portable TCP/IP interfacing using the Lightweight IP (lwIP) interface. |
| **Ch. 10:** Synchronization and Flow Control | 1.0 | Coping with a wide range of data I/O rates. Polling and interrupt-driven I/O. Direct Memory Access (DMA). Common data structures using in I/O software: buffers, double buffers, first-in last-out stacks, first-in last-out queues, circular buffers, semaphores, and flags. The pros and cons of the standard flow control methods: X-ON/X-OFF, hardware flow control, ENQ/ACK, and window-based methods. |
| **Ch. 11:** Stepper Motor Control Using the Enhanced Time Processing Unit (eTPU) | 1.0 | Actuators, relays, servos and stepper motors. Simplified introduction into stepper motor operation and control. The full-step and half-step drive sequences. Introduction to the Enhanced Time Processing Unit (eTPU) in the MCF5234 microcontroller. Provided standard eTPU functions. Channel priorities in the eTPU. Survey of the major eTPU registers. The application programming interface to the eTPU stepper motor control function. The stepper motor C++ object used in the laboratory microcomputer. |
| **Ch. 12:** Microcomputer Busses and Direct Memory Access (DMA) | 1.3 | Local processor-memory busses and graphics busses. Input/output, peripheral and instrument busses. Backplane busses. Bus hierarchies in the Personal Computer (PC). Chipset-based PC bus architecture. Bus terminology and common acronyms. Common bus functions: data transfer, interrupt handling, arbitration among competing bus masters, utility functions. Sharing a bus with arbitration, exclusive control, cycle stealing, and split transactions. Synchronous, asynchronous and semi-synchronous busses. Unilateral, bilateral, and multilateral data transfer protocols. Bus trade-offs. Timing diagrams. Set-up time, delay time, hold time, and timing skew. Detailed survey of the the MCF5441X external bus. Bus resizing mechanism. The JTAG port. Single-address and two-address direct memory access (DMA). Review of the MCF54415 DMA controller. |
| **Ch. 13:** Survey of Representative Busses and Interfaces | 1.3 | The General-Purpose Interface Bus (GPIB). Wire-AND and tri-stateable bus connections. Standard C expansion busses with an emphasis on the Peripheral Component Interconnection (PCI) bus. Survey of the Universal Serial Bus (USB). Recent high-speed serial busses. |
| **Ch. 14:** Optional Material (time permitting) | 1.0 | Topics could include basic concepts in high-speed signal propagation, application of transmission line theory, and other topics relevant to Computer Interfacing. |

**Evaluation Scheme:**

| | |
|---|---|
| Assignments | 10.0% |
| Laboratory Exercises | 20.0% |
| Midterm Examination | 20.0% |
| Final Examination | 50.0% |
| Total | 100.0% |

**Problem Assignments:**

Five assignments will be released during the term. The handouts will be available on-line in the assignments section of this eClass site. The last assignment might not be marked for credit, although the model solutions would be provided.

**Laboratory Experiments:**

The course includes a laboratory component that consists of one tutorial session and five laboratory experiments that require word processed formal reports. Most of the laboratory experiments require prelab work, which will be checked at the start of the first session for each lab. Also, most laboratory experiments require a demonstration of a working system, which will be checked at the start of the first session of the next lab.

Upon registration, students are assigned to one of the laboratory sections, which run in ETLC E5-002 from 2:00 pm to 5:00 pm on alternate weeks: H21 (Tuesdays, class 96748), H31 (Wednesdays, class 82540), H41 (Thursdays, class 82678) and H51 (Fridays, class 91672). The first lab exercise, which is a tutorial that requires no prelab work, will be held on Tuesday, January 14 for H21, Wednesday, January 15 for H31; Thursday, January 16 for H41; and Friday, January 17 for H51. The detailed schedule of laboratory dates and deadlines is posted on the separate eClass for the ECE 315 laboratory. Students will perform the lab work in teams of two. Be sure to choose a compatible and reliable partner in the first week of the term. Teams will succeed or struggle as a unit, so choose your partner carefully.

Detailed information about the laboratory experiments and related documentation can be found on the separate eClass page for the ECE 315 Laboratory.

**Examinations:**

• The 50-minute Midterm Examination will be written in the classroom on Wednesday, February 26, 2020, instead of the lecture. A new location for the midterm examination might be announce later on during the term.
• The two-hour Final Examination has been *tentatively* scheduled by the Registrar's office to take place on Friday, April 17, 2020, starting at 2:00 pm (please refer to the Examination Timetable announced by the Registrar's Office for confirmation). The location of the exam will be announced later on in the term.
• The use of any kind of pocket calculator is permitted in the Midterm and Final Examinations. No communication is permitted between students by any means during the examination period.

**Plagiarism and Other Academic Misconduct:**

The University of Alberta is committed to the highest standards of academic integrity and honesty. Students are expected to be familiar with these standards regarding academic honesty and to uphold the policies of the University in this respect. Students are particularly urged to familiarize themselves with the provisions of the **Code of Student Behaviour** (online at: http://www.governance.ualberta.ca/en/CodesofConductandResidenceCommunityStandards/CodeofStudentBehaviour.aspx), and avoid any behaviour which could potentially result in suspicions of cheating, plagiarism, misrepresentation of facts and/or participation in an offence. Academic dishonesty is a serious offence and can result in suspension or expulsion from the University.

**Supplementary Reference Materials (not required):**

• Jean J. Labrosse, *MicroC/OS-II - The Real-Time Kernel*, 2nd ed., (CMP Books, 2002), ISBN 1-57820-103-9. Call number: TJ 217.7 L34 2002. A copy of this book has been placed on reserve in Cameron library. Note that this book describes a slightly different version of uCOS than the one we'll be using in the lab. However, much of the content is similar.

• Jack G. Ganssle and Stuart R. Ball (eds.), *Embedded Systems,* (Elsevier/Newnes, 2008), ISBN 9780750686259. Available on-line as an eBook through the University of Alberta library at: https://www.library.ualberta.ca/catalog/4258069

• The many documents posted to the ECE 315 laboratory eClass site.

• Documentation available from Micrium's website (http://micrium.com) upon free registration.

• Don Anderson and Dave Dzatko, *Universal Serial Bus System Architecture*, 2nd ed., (Addison Wesley, 2001), ISBN 0-201-30975-0.

• Behrouz A. Forouzan, *TCP/IP Protocol Suite*, (McGraw-Hill, 2003), ISBN 0-07-119962-4.

Last modified: Monday, 6 January 2020, 11:48 AM

◄ Contact Information

Jump to... Chapter 1 - Introduction to Computer Interfacing and Embedded Systems ►

Go