

OFFICIAL UNDERGRADUATE COURSE OUTLINE FORM

Note: The University reserves the right to amend course outlines as needed without notice.

Course Code and Number: ELTR 202		Number of Credits: 3 Course credit policy (105)													
Course Full Title: Microprocessors/Microcontrollers and Data Acquisition															
Course Short Title: Microprocessors/Microcontroll.															
Faculty: Faculty of Applied and Technical Studies		Department (or program if no department): Electronics													
Calendar Description: Introduction to small microprocessor-based systems. Develop "C" programming and Assembly programming. Interfacing digital and analog signals with a computer-based system using common protocols and wireless/remote applications. Interface with common sensors and transducers.															
Prerequisites (or NONE):		Admission to the Automation and Robotics Technician diploma program or department permission.													
Corequisites (if applicable, or NONE):		NONE													
Pre/corequisites (if applicable, or NONE):		NONE													
Antirequisite Courses (<i>Cannot be taken for additional credit.</i>) Former course code/number: Cross-listed with: Equivalent course(s): <i>(If offered in the previous five years, antirequisite course(s) will be included in the calendar description as a note that students with credit for the antirequisite course(s) cannot take this course for further credit.)</i>		Course Details Special Topics course: No <i>(If yes, the course will be offered under different letter designations representing different topics.)</i> Directed Study course: No <i>(See policy 207 for more information.)</i> Grading System: Letter grades Delivery Mode: May be offered in multiple delivery modes Expected frequency: Winter only Maximum enrolment (for information only): 24													
Typical Structure of Instructional Hours <table border="1"> <tr> <td>Lecture/seminar</td> <td>30</td> </tr> <tr> <td>Supervised laboratory hours (design lab)</td> <td>20</td> </tr> <tr> <td></td> <td></td> </tr> <tr> <td></td> <td></td> </tr> <tr> <td></td> <td></td> </tr> <tr> <td>Total hours</td> <td>50</td> </tr> </table>		Lecture/seminar	30	Supervised laboratory hours (design lab)	20							Total hours	50	Prior Learning Assessment and Recognition (PLAR) PLAR is available for this course.	
Lecture/seminar	30														
Supervised laboratory hours (design lab)	20														
Total hours	50														
Scheduled Laboratory Hours Labs to be scheduled independent of lecture hours: <input type="checkbox"/> No <input checked="" type="checkbox"/> Yes		Transfer Credit (See bctransferguide.ca .) Transfer credit already exists: No Submit outline for (re)articulation: No <i>(If yes, fill in transfer credit form.)</i>													
Department approval		Date of meeting: November 9, 2021													
Faculty Council approval		Date of meeting: November 18, 2021													
Undergraduate Education Committee (UEC) approval		Date of meeting: January 28, 2022													

Learning Outcomes *(These should contribute to students' ability to meet program outcomes and thus Institutional Learning Outcomes.)*

Upon successful completion of this course, students will be able to:

1. Describe architecture of common microprocessors/controllers.
2. Program in C language, exposing students to various fundamental programming and interfacing techniques.
3. Investigate small microprocessor-based systems, with an emphasis on embedded system hardware and software design.
4. Demonstrate debugging techniques for microcontroller programs, including breakpoints, status, readouts, single-stepping, and crash dumps. Learn to debug hardware/software interaction problems.
5. Interface to I/O and interrupt handling methods.
6. Interface with digital and analog signals.

Recommended Evaluation Methods and Weighting *(Evaluation should align to learning outcomes.)*

Final exam:	50%	Quizzes/tests:	20%	%
	%	Lab work:	30%	%

Details:

NOTE: The following sections may vary by instructor. Please see course syllabus available from the instructor.

Texts and Resource Materials *(Include online resources and Indigenous knowledge sources. [Open Educational Resources](#) (OER) should be included whenever possible. If more space is required, use the [Supplemental Texts and Resource Materials form](#).)*

Type	Author or description	Title and publication/access details	Year
1. Textbook	Han-Way Huang	PIC Microcontroller: An Introduction to Software and Hardware Interfacing / Delmar	2004
2.			
3.			
4.			
5.			

Required Additional Supplies and Materials *(Software, hardware, tools, specialized clothing, etc.)***Course Content and Topics**

Unit 1: Basic programming concepts in Assembly and C
 Unit 2: Hardware interface: sensors
 Unit 3: Hardware interface: Controllers and Actuators
 Unit 4: Data collection systems and methods
 Unit 5: Analog and digital signals