MNE3058: EMBEDDED CONTROL SYSTEMS

Effective Term
Semester B 2024/25

Part I Course Overview

Course Title
Embedded Control Systems

Subject Code
MNE - Mechanical Engineering
Course Number
3058

Academic Unit
Mechanical Engineering (MNE)

College/School
College of Engineering (EG)

Course Duration
One Semester

Credit Units
3

Level
B1, B2, B3, B4 - Bachelor's Degree

Medium of Instruction
English

Medium of Assessment
English

Prerequisites
CS1302 Introduction to Computer Programming AND
MNE2029/BME2029 Electrical and Electronic Principles I or equivalent

Precursors
Nil

Equivalent Courses
Nil

Exclusive Courses
Nil

Additional Information
#Prerequisites which are not part of the Major Requirement are waived for students admitted with Advanced Standing.

Part II Course Details

Abstract

The aim of this course is for the students to learn the fundamental principles of embedded mechatronic control and to gain practical skills for interfacing and integrating actuators and sensors with embedded microcontrollers within relatively complex mechatronic systems.

Course Intended Learning Outcomes (CILOs)

	CILOs	Weighting (if app.)	DEC-A1	DEC-A2	DEC-A3
1	Describe the basic elements and major issues involved in developing embedded software systems for mechatronic control.			x	
2	Design embedded software systems based on user specifications.			x	
3	Develop real-time mechatronic control software including interfaces with sensors and actuators for typical mechatronic applications.			x	
4	Apply machine intelligence and sensory feedback to extend the functionality of a mechatronic system.		x	x	x

A1: Attitude
Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.

A2: Ability
Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to real-life problems.

A3: Accomplishments
Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

Learning and Teaching Activities (LTAs)

	LTAs	Brief Description	CILO No.	Hours/week (if applicable)
1	Lecture	Large class activities could include lectures, group discussion, and writing minute papers, muddiest points and reflective journals.	1, 2, 3, 4	2 hrs/week
2	Laboratory Work	Laboratory work will mainly teach the students the basic skills to interface actuators and sensors with embedded microcontrollers and to develop embedded real-time control software.	3, 4	3 hrs/week for 4 weeks

3	Contextualised PBL	Contextualised PBL (Problem Based Learning): Typical embedded mechratonic control problems will be given to students to solve. The students are expected to work in teams for about 8 weeks to tackle the given problems. This learning activity will be mainly student-led but with some structural guidance from the teacher. At the end of the learning activity, a demonstration or competition will be organized for all the students to test and compare their solutions for the given problems.	3, 4	

Assessment Tasks / Activities (ATs)

	ATs	CILO No.	Weighting (%)	Remarks (e.g. Parameter for GenAI use)
1	Skill Test	3	5	Programming tasks will be given to students to test their basic programming skill; the duration of each test will not be more than 10 minutes.
2	Laboratory Report	1, 2, 3	15	
3	Contextualised PBL	2, 3, 4	30	Report submission and participate in competition
4	Test	1, 2	10	

Continuous Assessment (%)

60

Examination (%)

40

Examination Duration (Hours)

2

Additional Information for ATs

For a student to pass the course, at least 30% of the maximum mark for both coursework and examination should be obtained.

Assessment Rubrics (AR)

Assessment Task

1. Skill Test

Criterion
Ability to Develop a real-time software to handle input-output functions for a given problem.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Assessment Task
2. Laboratory Report

Criterion
Ability to write basic ARM C code to solve simple programs involving input-output functions.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Assessment Task
3. Contextualised PBL

Criterion
3.1 Ability to Design a real-time software for solving a given problem.
3.2 Ability to Develop a real-time software to control a mechatronic device for solving a given problem based on sensory feedback.
3.3 Ability to Apply machine intelligence and sensory feedback to handle some uncertainty in a given problem.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Assessment Task
4. Test

Criterion
Ability to Describe issues related to basic elements and major issues involved in developing embedded software systems.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)
Not even reaching marginal levels

Assessment Task
5. Examination

Criterion
Ability to Describe issues and Solve problems related to real-time embedded control systems.

Excellent (A+, A, A-)
High

Good (B+, B, B-)
Significant

Fair (C+, C, C-)
Moderate

Marginal (D)
Basic

Failure (F)

Not even reaching marginal levels

Additional Information for AR

Note: For a student to pass the course, at least 30% of the maximum mark for both coursework and examination should be obtained.

Part III Other Information

Keyword Syllabus

- Computer Architecture – Von Neumann, Harvard, CISC, RISC, Cache, Pipelining, Memory;
- Real-time Embedded Software Development – Interrupt, Polling, Task Scheduling, Mutual Exclusion, Deadlock, Starvation, Semaphores, Monitor;
- Mechatronic Control – Sensor Interfacing, Actuator Control, Signal Processing, Intelligent Control Algorithms;
- Computer Interfaces – PWM, ADC, DAC, Digital Input/Output, SPI, UART, In-System Programming;
- Embedded C Language – additional data structures for accessing registers in embedded controllers;
- Embedded Software Design Techniques and Tools – Function-oriented Design, Object-oriented Design, Dataflow diagram, Structure Chart, Flow-Chart, Pseudo-code, Data Dictionary;
- System Verification and Validation – A process for ensuring that the software being developed conforms to its specifications (verification) and meets the expectations of user (validation).

Reading List

Compulsory Readings

	Title
1	Nil

Additional Readings

	Title
1	UM10375 - LPC1311/13/42/43 User manual, 21 June 2012, NXP Semiconductors.
2	LPC1311/13/42/43 Data Sheet, 6 June 2012, NXP Semiconductors.
3	Getting started with NXP LPCXpresso – User Guide, 11 July 2012, NXP Semiconductors.
4	Daniel Page, Practical introduction to computer architecture, London Springer, c2009.
5	Joseph Yiu, The definitive guide to the ARM Cortex-M3, Elsevier, c2010.
6	Online Resources: 1.Website for downloading the LPCXpresso Software: https://www.lpcware.com/lpcxpresso/download 2.LPCXpresso Introduction part 1 training video : http://www.youtube.com/watch?feature=player_embedded&v=dV7rG2VdG9E 3.LPCXpresso Introduction part 2 training video : http://www.youtube.com/watch?feature=player_embedded&v=cLvGwmJAA7k 4.Website for downloading technical documents for LPC1343FBD48: http://www.nxp.com/products/microcontrollers/cortex_m3/lpc1300/LPC1343FBD48.html#documentation