2023
TP-153

Course Specification
(Bachelor)

Course Title:	Microprocessors and Embedded Systems
Course Code:	EE1531
Program:	Electrical Engineering
Department:	Electrical Engineering
College:	College of Engineering
Institution:	Imam Mohammad Ibn Saud Islamic University
Version:	V5
Last Revision Date:	01-01-2025

Table of Contents

A. General information about the course:

1. Course Identification

1. Credit hours: (2)					
2. Course type					
A.	☐University	☐College	☒ Department	☐Track	☐Others
B.	☒ Required		☐Elective		
3. Level/year at which this course is offered: (5th level, 3rd year)					
4. Course general Description:					
This course provides an introduction to microprocessors and embedded systems, focusing on their role in real-world engineering applications. It emphasizes the design and implementation of embedded solutions using high-level programming languages, such as C and Python, with hands-on experience on Atmel microcontrollers used in Arduino platforms. Students will gain the foundational knowledge required to program, interface, and debug embedded systems, focusing on modular programming techniques, interfacing sensors, and controlling hardware peripherals.					
5. Pre-requirements for this course (if any)**:**					
EE1333, GE1108					
6. Co-requisites for this course (if any)**:**					
--					
7. Course Main Objective(s):					
The course is structured to provide students with: • Basic understanding of microprocessors and microcontrollers. • Hands-on experience in building embedded projects using Arduino boards. • Skills to write, debug, and test embedded code in C and Python. • Understanding of microcontroller architecture, I/O handling, and peripheral interfacing. This course is ideal for students with limited experience in microprocessors and programming, as it introduces concepts from the ground up and builds towards complete embedded system design.					

2. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	-	-
2	E-learning	-	-
3	Hybrid • Traditional classroom	40 20	67% 33%

No	Mode of Instruction	Contact Hours	Percentage
	• E-learning		
4	Distance learning	-	-

3. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	**Lectures**	30
2.	**Laboratory/Studio**	15
3.	**Field**	-
4.	**Tutorial**	-
5.	**Others (self learning)**	15
Total		60

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	**Knowledge and understanding**			
K1	Analyze the architecture and operation of microcontrollers/ microprocessors	1.1	Focus on the theoretical understanding of microcontrollers.	Activity completion and exams
2.0				
S1	Develop and debug C/Python programs for microcontroller/micro processor-based systems	2.3	Enhances students' analytical and programming skills.	Activity completion and exams
S1	Interface hardware components with microcontrollers/ microprocessors	2.5	Designing hardware-software solutions teaches students how to meet performance, cost,	Activity completion and exams

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
			and time constraints.	
S2	Implement communication protocols and real-time applications.	6.2	Promotes engineering design and technical analysis skills.	Activity completion and exams
3.0	**Values, autonomy, and responsibility**			
V3	Work effectively in teams to design and build embedded projects.	5.1	Prepares students for teamwork and professional collaboration.	Peer review and project evaluation

C. Course Content

No	List of Topics	Contact Hours
1	Introduction to Microprocessors and Microcontrollers: Comparison between microprocessors and microcontrollers; real-world applications	4
2	Overview of Atmel Microcontrollers and Arduino IDE: Introduction to the Atmega328p used in Arduino boards; setting up the Arduino IDE **LAB1: Introduction to Arduino IDE: Setting up and testing simple LED programs**	4
3	Basics of C Programming for Microcontrollers: Data types, loops, and control structures in C for embedded systems	4
4	GPIO Programming in Arduino: Interfacing LEDs, switches, and relays **LAB2: Programming GPIO Pins: Controlling LEDs and switches**	4
5	Introduction to Python/C++ for Embedded Systems: MicroPython/C++ and its usage on ESP32 boards	4
6	Timers, Interrupts, and Delays: Using timers and interrupts to manage tasks asynchronously	4
7	Communication Protocols: UART, SPI, and I2C: Interfacing with sensors and displays via communication protocols **LAB3: Communication protocols: Implementing UART and I2C communication.**	4
8	Midterm Exam: Covers microcontroller basics, programming, and GPIO interfacing	4
9	Sensor Interfacing: Working with temperature, motion, and proximity sensors **LAB4: Sensor Interfacing: Interfacing temperature and motion sensors.**	4
10	Motor Control and PWM: Controlling **DC motors, servo motors**, and fans using PWM signals **LAB 5: Motor Control: Driving DC Motors and servos using PWM.**	4
11	Real-Time Applications: Implementing real-time tasks using timers and interrupts	4

	Final Project: Integrating multiple sensors and actuators in a real-time project.	
12	Memory Management and Data Logging: Working with EEPROM and SD cards for data storage	4
13	Project Development: Prototype Design: Planning, coding, and testing a small embedded project	4
14	Debugging and Troubleshooting: Identifying and fixing software/hardware bugs	4
15	Final Project Presentation and Course Review: Present group projects and reflect on learning outcomes	4
Total		**60**

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	Midterm	7th week	30%
2.	Lab Work	12th week	15%
3.	Quizzes & Assignments	All Along	--
4.	Project	13th to 14th week	15%
5.	Final Exam	Final Exam week	40%

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.).

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	**The 8051 Microcontroller and Embedded Systems** by Muhammad Ali Mazidi (for understanding core microcontroller concepts)
Supportive References	**"Exploring Arduino: Tools and Techniques for Engineering Wizardry"** by Jeremy Blum (focus on Arduino programming and projects).
Electronic Materials	• Computer animations and online resources supplied by the instructor. • **MicroPython for ESP32 Development** by Jacob Beningo (for Python-based microcontroller development).
Other Learning Materials	

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	One classroom: fits up to 25 students with white board.

Items	Resources
Technology equipment (projector, smart board, software)	A laptop computer connected to a projector to display PowerPoint presentations
Other equipment (depending on the nature of the specialty)	• **Arduino Uno boards (Atmega328p)** • **ESP32 boards** for MicroPython experiments • **Sensors and actuators** (temperature sensors, DC motors, etc.) • **Development software**: Arduino IDE, Python IDE, and Keil uVision (optional).

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	Students	Indirect
Effectiveness of Students assessment	Students	Indirect
Quality of learning resources	Relevant Focus Group	Indirect
The extent to which CLOs have been achieved	Dept. Quality Committee	Direct
Other		

Assessors (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)

Assessment Methods (Direct, Indirect)

G. Specification Approval

COUNCIL /COMMITTEE	
REFERENCE NO.	
DATE	