

25UAMT416	EMBEDDED SYSTEMS	Category	L	T	P	R	Credit	Year of Introduction
		PEC	3	0	0	0	3	2025

Preamble: This course introduces the fundamentals of Embedded Systems, including system architecture, design methodologies, real-time concepts, and hardware–software co-design. It also covers firmware development, RTOS principles, and industry applications such as automotive and IoT systems.

Prerequisite: 25UCST302 Operating Systems, 25UCST304 Switching Theory and Logic Design

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO 1	Explain embedded system architecture, key components, characteristics, and the system design process.	K2 Understand
CO 2	Apply system modeling techniques and analyze hardware–software co-design methodologies for optimized embedded system development.	K3 Apply
CO 3	Implement task scheduling, communication, synchronization mechanisms, and RTOS selection in real-time embedded systems.	K3 Apply
CO 4	Apply appropriate development life cycle models to design and develop embedded firmware, and implement embedded system applications in automotive and IoT domains.	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	1	1	-	-	-	-	-	--	2	1
CO2	3	3	3	2	2	-	-	-	1	--	--	3	2
CO3	3	3	2	2	2	-	-	-	1	--	--	3	2
CO4	2	2	3	2	2	-	-	-	2	1	1	3	3
Avg	2.75	2.5	2.5	1.75	1.75	-	-	-	1	1	1	2.75	2

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*	*	*			
CO2	*	*	*	*	*	*			
CO3	*	*	*	*	*	*		*	
CO4	*	*	*	*	*	*			

WKs Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*			

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS

Module 1 (10 hrs)

Definitions of Embedded Systems-Embedded Systems and General Computing Systems-History and Evolution-Classification of Embedded Systems- Application Areas-Purpose and Characteristics. Building Blocks of a Typical Embedded System-System Core-Memory-Embedded System Design Process-Requirements Analysis-System Specification-Architecture Design- Hardware Design- Software Design.

Module 2 (10 hrs)

Computational Models in Embedded Design – Data Flow Graph, Control Data Flow Graph, State Machine Model, Sequential Program Model, Concurrent Process Model, Object-Oriented Model. Hardware Software Co-Design – Traditional Embedded Development Cycle, History, Advantages of the Co-Design Methodology, The Co-Design Process, Fundamental Issues in Hardware Software Co-Design. Hardware software trade-offs.

Module 3 (12 hrs)

Operating System Basics-Types of OS, Tasks, Processes, Threads Multiprocessing & Multitasking Task Scheduling-Non-Preemptive (FIFO, LIFO, SJF), Preemptive (SRT, RR, Priority-based, Rate-based), Task Communication – Shared Memory, Message Passing, Remote Procedure Call and Sockets. Task Synchronization – Synchronization Issues – Race Condition, Deadlock, Priority Inversion, Priority Inheritance, Priority Ceiling. Synchronization Techniques – Spin Lock, Sleep & Wakeup, Semaphores. Selection of an RTOS for an Embedded Design – Functional and Non-Functional Requirements.

Module 4 (10 hrs)

Embedded Firmware Design-Firmware Design Approaches, Firmware Development Languages. Hardware-Firmware Integration, Embedded Development Life Cycle (EDLC)- Objectives, Development Models- Waterfall Model, Incremental Model, Evolutionary Model, Spiral Model. Industry Case Studies, Automotive & Aerospace Systems-Networked Control Systems, Internet of Things (IoT)- IoT Architectures, IoT Networks.

Text Books:

1. Shibu, K. V. (2017). *Introduction to embedded systems* (2nd ed.). McGraw Hill Education.
2. Peckol, J. K. (2019). *Embedded systems: A contemporary design tool* (2nd ed.). John Wiley & Sons.
3. Wolf, M. (2016). *Computers as components: Principles of embedded computing system design* (4th ed.). Morgan Kaufmann.

Reference Books:

1. Staunstrup, J., & Wolf, W. (2013). *Hardware/software co-design: Principles and practice*. Springer.
2. Kamal, R. (2011). *Embedded systems: Architecture, programming and design*. Tata McGraw-Hill Education.
3. Gajski, D. D., et al. (2009). *Embedded system design: Modeling, synthesis and verification*. Springer.
4. Marwedel, P. (2017). *Embedded system design: Embedded systems foundations of cyber-physical systems, and the Internet of Things*. Springer.

END SEMESTER EXAMINATION – MODEL QUESTION PAPER

Course Code	25UAMT416
Course Name	EMBEDDED SYSTEMS
Max. Marks: 60	Duration: 2.5 Hours

Part A (Answer all questions. Each question carries 3 marks)				
Qn No .	Question	Marks	CO	BL
1.	Define an embedded system. Differentiate it from a general-purpose computing system.	3	CO1	L2
2.	Explain the building blocks of a typical embedded system.	3	CO1	L2
3.	Explain the Data Flow Graph and State Machine Model used in embedded design.	3	CO2	L2
4.	Describe the advantages of hardware-software co-design methodology.	3	CO2	L2

5.	Compare non-preemptive and preemptive task scheduling with examples.	3	CO3	L2
6.	Explain Race Condition and Deadlock in embedded systems with an example.	3	CO3	L2
7.	Explain the Waterfall Model and Spiral Model used in embedded firmware development.	3	CO4	L2
8.	Describe IoT architectures and the role of embedded systems in IoT.	3	CO4	L2

Part B (Answer any one full question from each Module. Each carry 9 marks)				
Qn No.	Question	Marks	CO	BL
Module I				
9.1	Explain the classification of embedded systems with examples. Describe the purpose and characteristics of embedded systems.	5	CO1	L2
9.2	Describe the embedded system design process covering requirements analysis, system specification, architecture design, hardware design, and software design.	4	CO1	L2
OR				
10	Explain the history and evolution of embedded systems. Describe the application areas and building blocks of a typical embedded system.	9	CO1	L2
Module II				
11.a	Explain the computational models: Data Flow Graph, Control Data Flow Graph, and Sequential Program Model used in embedded design.	5	CO2	L2
11.b	Describe hardware-software trade-offs in embedded system design with suitable examples.	4	CO2	L3
OR				
12	Explain the hardware-software co-design methodology: traditional development cycle, co-design process, and fundamental issues in co-design.	9	CO2	L2
Module III				
13	Explain task scheduling in embedded systems: Non-Preemptive (FIFO, SJF) and Preemptive (Round Robin, Priority-based) scheduling. Compare their efficiency with examples.	9	CO3	L2
OR				
14.a	Explain task synchronization techniques: Spin Lock, Semaphores, and Sleep & Wakeup. Describe Priority Inversion	5	CO3	L2

	and Priority Inheritance.			
14.b	Describe the functional and non-functional requirements for selecting an RTOS for embedded system design.	4	CO3	L3
Module IV				
15.a	Explain the embedded development life cycle (EDLC) models: Waterfall, Incremental, Evolutionary, and Spiral models.	5	CO4	L2
15.b	Describe IoT architectures and IoT networks. Explain how embedded systems are applied in automotive and IoT domains.	4	CO4	L3
OR				
16	Explain firmware design approaches and firmware development languages. Describe hardware-firmware integration with a case study in automotive/aerospace systems.	9	CO4	L3