NIRMA UNIVERSITY

Institute:	Institute of Technology
Name of Programme:	BTech Electronics & Communication Engineering
Course Code:	3EC705CC24
Course Title:	Embedded System
Course Type:	Core
Year of Introduction:	2024-25

L	T	Practical component				C
		LPW	PW	W	S	
3	-	2	-	-	-	4

Course Learning Outcomes (CLOs)

At the end of the course, the students will be able to

1. comprehend modern controller architecture along with bus protocols (BL2)
2. compare processes, threads and tasks, multitasking and multithreading, scheduling, in context of real time operating systems (BL3)
3. develop an application from embedded system using Embedded-C programming language (BL3)
4. design the model of the embedded system for given specifications. (BL5)

	Contents	Teaching hours (Total 45)
Unit I	**Introduction:** Embedded systems overview, comparison with general purpose computing system, application areas, purpose of embedded system, typical embedded system, characteristics of embedded systems, quality attributes of embedded systems, common design metrics and challenges, types and examples of embedded systems	**05**
Unit II	**Processors and Controller**: The ARM and RISC philosophy, ARM processor families ARM architecture revision & advanced family processors (Cortex-A, -R, -M, -secure core family), ARM architecture revision, general ARM processor architecture and fundamentals, ARM register architecture, ARM exceptions, interrupts, and the vector table, pipelines in ARM processor, ARM core extension, ARM instruction addressing modes, ARM core instruction set (data processing, branch, load-store etc.), ARM thumb instruction set overview	**14**
Unit III	**Real Time Operating Systems (RTOS):** Operating system basics and OS services, types of operating system, RTOS in embedded systems, task-process-threads, multiprocessing and multitasking, task scheduling, scheduling algorithms, co-operative scheduling, non-preemptive scheduling, pre-emptive scheduling, context switching, task-processes-thread, task communication, task synchronization, how to choose RTOS, introduction to open source RTOS	**12**
Unit IV	**Bus Protocol for Embedded Systems and Device Drivers**: Universal Serial Bus (USB), serial Peripheral Interface (SPI), inter-integrated Circuit (I2C), controller Area Network Bus (CAN Bus), case study of SPI & I2C for ARM Processor, basics of device driver (their functions, architecture, types, and implementations), case study of device driver development	**10**
Unit V	**Embedded System Programming:** Embedded programming tools and languages, good practices for Embedded-C, efficient C programming for ARM	**04**

Self-Study:
The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:
Laboratory work will be based on the above syllabus with a minimum of 10 experiments to be incorporated.

Suggested Readings/Reference:

1. Andrew N. Sloss, Dominic Symes and Chris Wright, ARM System Developer's Guide, Designing and Optimising System software, Elsevier
2. Shibu K. V, Introduction to Embedded Systems, Tata McGraw-Hill
3. Rajib Mall, Real-Time Systems: Theory and Practice, Pearson Education
4. Janice Gillispie Mazidi, Muhammad Ali Mazidi, and Rolin D. McKinlay, The 8051 Microcontroller and Embedded Systems: Using Assembly and C, Pearson Education
5. Steve Furber, ARM System-on-Chip Architecture, Addison-Wesley

Details of Laboratory
Suggested List of Experiments

Sr. No.	Practical	No. Of Hours
1.	To get familiarise with Keil µVision Integrated Development Environment (IDE), its features and Data types in Embedded C language for Embedded System development	02
2.	To develop Embedded C code for Input Output operations of 8051 microcontroller.	02
3.	To demonstrate Logic Operations in embedded C for programming of 8051 microcontroller	02
4.	To Appraise the concepts of utilisation of timers and counters in 8051 by preparing the Embedded C code	02
5.	To explore the concept of serial communication in 8051 by formulating the Embedded C code	02
6.	To analyse the concept of interrupt programming in 8051 by developing the Embedded C Code	04
7.	Getting started with ARM LPC2148 using Keil uVision IDE & To inspect the concept of GPIO Ports and Registers in LPC2148	02
8.	To understand the concept LCD interfacing with ARM7 Board by developing Embedded C Code	02
9.	To develop Embedded C code to generate Pulse Width Modulation (PWM) signal using ARM7	04
10.	To formulate Embedded C code to investigate the concept of data transmission and reception using Universal Asynchronous Receive-Transmit (UART)-serial communication protocol	04
11.	To develop Embedded C code for interfacing SPI peripheral with ARM7	04
12.	To develop Embedded C code for interfacing I2C peripheral with ARM7	04