B.E. ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
Choice Based Credit System (CBCS)
SEMESTER – VI
Real Time Operating System (3:0:0) 3
(Effective from the academic year 2021-22)

Course Code	21AM652	CIE Marks	50
Teaching Hours/Week (L: T:P)	3:0:0	SEE Marks	50
Total Number of Contact Hours	40	Exam Hours	3

Course Objectives:
This course will enable students to:
1 Setup and demonstrate the development environment for RTOS
2 Illustrate strategies to interface memory and I/O with RTOS kernels
3 Interpret tasks used in handling multiple activities
4 impart skills necessary to develop software for embedded computer systems using a real-time operating system

Module – 1
Introduction: Implication and Scope of Course and its Importance in Economic growth of Nation, Impact of the course on Societal Problems/ Sustainable Solutions/ National Economy, Career Perspective, Overview of the course in current Innovations and Research trends. Operating system objectives and functions, Virtual Computers, Interaction of O. S. & hardware architecture, Evolution of operating systems, Architecture of OS (Monolithic, Microkernel, Layered, kernel and Hybrid kernel structures), Batch, Multi programming, Multitasking, Multiuser, parallel, distributed & real –time O.S.
(8 Hours)

Module – 2
Scheduling algorithms: FCFS, SJF, Priority, Round Robin, UNIX Multi-level feedback queue scheduling, Thread Scheduling, Multiprocessor Scheduling concept, Concurrency: Principles of Concurrency, Mutual Exclusion H/W Support, software approaches, Semaphores and Mutex, Message Passing techniques.
(8 Hours)

Module – 3
STM32f4_RTOS_BSP, Testing STM32f4_RTOS_BSP, Time Management
Timer Management, Resource Management Disable/Enable Interrupts, Lock/Unlock
Semaphores, Mutex, Deadlocks, Synchronization.
(8 Hours)

Module – 4
Introduction to Internal Task, Idle Task, Tick Task, Statistics Task, Timer Task, ISR Handler Task, Scheduling, Preemptive Scheduling, Scheduling points, Round Robin Scheduling Context Switching , coding cooperation.
(8 Hours)

Module – 5
Interrupt Management, Interrupt Service Routine, Non-Kernel Aware Interrupt Service Routine CPUs with Multiple Interrupt Priorities, All Interrupts Vector to a Common Location, Every Interrupt Vectors to a Unique Location, Direct vs. Deferred Post Methods.
     Recap/Summary of the Course.
(8 Hours)

Course Outcomes:

The students will be able to:

CO1	Create, test and debug on RTOS environment
CO2	Implement Inter task communication mechanism.
CO3	Compare general purpose OS with RTOS
CO4	Demonstrate methods in storing, retrieving data in RTOS
CO5	Analyze performance of task during multitasking

Textbooks

1 Jean J Labrosse, "Micro C/OS-II, The Real Time Kernel", CMP, 3rd, 2016.
2 Mazidi, "STM32 Arm Programming for Embedded Systems", MircordigitalED, 1st 2018.

Reference Books

1 Colin walls, "Building a Real Time Operating System: RTOS from the Ground Up", Newness, 2020.
2 Sam, "Real-Time Embedded Components and Systems: With Linux and RTOS", Mercury Learning, 2015.