

GUJARAT TECHNOLOGICAL UNIVERSITY

ELECTRONICS & COMMUNICATION (EMBEDDED SYSTEM) (54)

REAL TIME OPERATING SYSTEM FUNDAMENTALS

SUBJECT CODE: 2715402

SEMESTER: I

Type of course: Embedded Software Design

Prerequisite: Fundamental Knowledge of programming language C, Micro-controller programming

Rationale: Embedded system engineer must have knowledge of real-time systems. He/she must address issues in real-time system design.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks						Total Marks
L	T	P		Theory Marks		Practical Marks				
			ESE (E)	PA (M)	PA (V)		PA (I)			
					ESE	OEP	PA	RP		
3	2#	2	5	70	30	20	10	10	10	150

Content:

Sr. No.	Contents	Total Hrs.	% Weightage
1	Embedded Firmware design approaches Low level language and high level language – Superloop architecture – Requirement of an operating system	2	5
2	Real time system concepts Real-time system – Hard real-time, soft real-time and firm real-time systems	4	10
3	Basic of Operating System Kernel – Kernel services – Process management, I/O management and memory management – Kernel-space and user-space – GPOS and RTOS – Real-time kernel services – RTOS requirement – Real-time kernel services	5	10
4	Process and threads Process – Process states – Process control block – Threads – Multiprocessing and multitasking – Co-operative multitasking, preemptive multitasking and non-preemptive multitasking	6	15
5	Scheduling FCFS – LCFS – Shortest job first – Priority based scheduling – Round robin – Rate monotonic scheduling – Earliest deadline first scheduling	8	20
6	Inter process communication Pipes – Message Queue – Mailbox Communication – Remote Procedure Call(RPC) – Sockets	7	15
7	Synchronization Shared data – Reentrant functions – Racing – Priority	10	25

	Inversion – Mutual Exclusion – Sleep & wakeup – Dining Philosophers’ Problem – Readers-Writers Problem – Producer-Consumer Problem – Semaphore – Deadlock – Deadlock detection and recovery – Deadlock avoidance – Starvation		
--	---	--	--

Reference Books:

1. Shibu K V, Introduction to Embedded Systems, Tata Mc Graw Hill Publication, ISBN: 978-0-07-014589-4
2. David E. Simon, An Embedded Software Primer, Pearson Education Publication, ISBN: 978-81-7758-154-6
3. Andrew S. Tanenbaum, Modern Operating Systems, Third Edition, Pearson Education Publication, ISBN: 978-81-317-2003-5
4. Jean J. Labrosse, Micro C/OS-II The Real-Time Kernel, Second Edition, CRC Press Publisher, ISBN: 978-1578201037

Course Outcome:

1. To understand Real-Time Operating System requirements and concepts.
2. To solve design issues involved with real-time embedded systems
3. To analyze the performance of a hard real time system.
4. To inculcate an ability to program an embedded system with multitasking concepts

List of Experiments:

1. Configure timer to generate time delays for embedded system.
2. Configure timer with Interrupt Service Routine and use in super loop architecture.
3. Introduction to Real Time Kernel RL-RTx.
4. Use Scheduling algorithms available in RL-RTx Kernel for multitasking application design.
5. Introduction to multithreading and study of Functions for POSIX Threads
6. Design the multitasking application for sharing memory among the tasks.
7. Use Semaphore for providing exclusive excess to input output device.
8. Study Inter-process communication mechanisms in RTOS environment.
9. Study Micro C OS-ii functions for RTOS based application development
10. Implement data transfer using UART protocol in RTOS environment.

Lab sessions may include implementation & study of real time operating systems other than stated above.

Open Ended Problems

1. Implement a real time intruder detector system.
2. Implement open source RTOS “Free RTOS” on ARM based development board. Implement 3 tasks to blink 3 LEDs with different delays.
3. Install Robotics Operating System (ROS) on your machine and develop code to grab video from camera.
4. Configure real time Linux kernel and write a code to create 10 threads and close all threads after specific time delay.
5. Write a simple device driver to read value from Hex keyboard

Major Equipments:

Hardware Platform

ARM based development board.
8051 based kit.

List of Software:

Keil
Eclipse
Linux
IAR Embedded Workbench

Learning website:

<http://micrium.com>
<http://www.freertos.org>
http://www.keil.com/support/man/docs/rlarm/rlarm_ar_artxarm.htm
<https://rt.wiki.kernel.org>