ACADEMY OF TECHNOLOGY

ECE DEPARTMENT

PROPOSED COURSE STRUCTURE FOR

"EMBEDDED SYSTEM DESIGN"

Introduction:

An embedded system is some combination of computer hardware and software, either fixed in capability or programmable, that is designed for a specific function or for specific functions within a larger system. Industrial machines, agricultural and process industry devices, automobiles, medical equipment, cameras, household appliances, airplanes, vending machines and toys as well as mobile devices are all possible locations for an embedded system.

With growth and advancements in the field of electronics and wireless communications, devices around us are able to communicate in a better way than one can imagine. Today, IoT (Internet of Things) is one of the hottest topics of the industry and has taken its place in conventional business Jargon. However, it brought host of challenges for developers — as they need to develop devices that allow seamless connectivity.

The future of embedded systems and IoT lies in the advancement of technologies that enable faster communication with high interwoven connections between different devices. No doubt, the future of **IoT embedded devices** is going to be bright with the easy access of internet in every corner of the world.

According to a recent report from analyst firm IDC, intelligent systems (primarily embedded systems) will likely see a compound annual growth rate of 7.2% from 2015 to 2020(NASSCOM Report) with revenues exceeding $2.2 trillion in 2020, driven in large part by the internet of things. The study predicts that other fast-growing market segments will include wearable, advanced driver assistance systems, drones, smart homes, smart buildings, video surveillance, 3D printers and transportation.

Pre-requisite: Basic Knowledge of Microprocessor & Microcontroller (Microprocessor and Microcontroller in 5th semester), Basic knowledge of programming (CS201-2nd semester)

Details of Course structure:

Course Name: Embedded System Design Credit: 3(36 Classes)

Semester 6th Stream: ECE

Paper Code: EC-604 C : Embedded System Design (No Lab, Free Elective Paper)

Syllabus Structure:

Module No.	Unit No.	Topics	No. of Class
1		**Fundamentals of Embedded System**	4
	1.1	Core of the embedded system, Memory, Sensors (resistive, optical, position, thermal) and Actuators (solenoid valves, relay/switch, opto-couplers), Communication Interface, Embedded firmware (RTOS, Drivers, Application programs), Power-supply (Battery technology, Solar), PCB and Passive components, Safety and reliability, environmental issues. Ethical practice.	
	1.2	Characteristics and quality attributes (Design Metric) of embedded system. Real time system's requirements, real time issues, interrupt latency.	
	1.3	Embedded Product development life cycle, Program modeling concepts: DFG, FSM, Petri-net, UML	
2		**Embedded Hardware and Design**	8
	2.1	Introduction to ARM-v7-M (Cortex-M3), ARM-v7-R (CortexR4) and comparison in between them	
3		**Embedded Serial Communication**	10
	3.1	Study of basic communication protocols like SPI, SCI (RS232, RS485), I2C, CAN, Field-bus (Profibus), USB (v2.0), Bluetooth, Zig-Bee, Wireless sensor network	
4		**Embedded Software, Firmware Concepts and Design**	14
	4.1	Embedded C-programming concepts (from embedded system point of view): Optimizing forSpeed/Memory needs, Interrupt service routines, macros, functions, modifiers, data types, device drivers, Multithreading programming. (Laboratory work on J2ME Java mobile application).	
	4.2	Basic embedded C programs/applications for ARM-v7, using ARM-GCC-tool-chain, Emulation of ARM-v7 (e.g. using QEMU), and Linux porting on ARM-v7 (emulation) board CASE STUDY: 1) Medical monitoring systems, 2) Process control system (temp, pressure) 3) Soft real time: Automated vending machines, 4) Communication: Wireless (sensor) networks.	
	4.3	Real time operating system: POSIX Compliance , Need of RTOS in Embedded system software, Foreground/Background systems, multitasking, context switching, IPC, Scheduler policies, Architecture of kernel, task scheduler, ISR, Semaphores, mailbox, message queues, pipes, events, timers, memory management, RTOS services in contrast with traditional OS.	
	4.4	Introduction to µCOS-II RTOS, study of kernel structure of µCOS-II, Synchronization in µCOS-II, Inter-task communication in µCOS-II, Memory management in µCOS-II, porting of RTOS on ARM-v7 (emulation) board, Application developments using µCOS-II.	
	4.5	Introduction Linux OS, Linux IPC usage, basic device (drivers) usage.	

Course Outcome:

Upon successful completion, our students will be

- highly competitive on the national and international job market, both in the industry as high - skilled expert and in the academia as a researcher or prospective PhD student. After studies, the students will be able to make a career as e.g., engineers, project leaders, system architects, programmers or researchers in the fields of e.g., automotive industry, robotics, telecom, industrial process control, consumer electronics etc.

- able to acquire knowledge and understand fundamental embedded systems design paradigms, architectures, possibilities and challenges, both with respect to software and hardware,

- able to analyze a system both as whole and in the included parts, to understand how these parts interact in the functionality and properties of the system

- Able to practically apply gained theoretical knowledge in order to design, analyze and implement embedded systems, e.g. integrating embedded subsystems and applications in building a fully functional autonomous robot.

- Apply formal method, testing, verification, validation and simulation techniques and tools in order to engineer reliable and safe embedded systems,

- Demonstrate a deeper understanding of the electronics and physical principles used for embedded biomedical measuring systems

Book Ref:

1. Introduction to Embedded Systems : Shibu K. V. (TMH)
2. Embedded System Design – A unified hardware and software introduction: F. Vahid (John Wiley)
3. Embedded Systems : Rajkamal (TMH)
4. Embedded Systems : L. B. Das (Pearson)
5. Embedded System design : S. Heath (Elsevier)
6. Embedded microcontroller and processor design: G. Osborn (Pearson)
7. Embedded Systems: Frank Vahid , Wiley India, 2002
8. Embedded Microcomputer Systems – Real Time Interfacing – Jonathan W. Valvano; Cengage Learning; Third or later edition.