18MC214EMBEDDED SYSTEMS

Course Description and Objectives:

The course emphasis on Comprehensive treatment of Embedded Hardware and Real Time Operating systems along with case studies in tune with the requirements of Industry. The example-driven approach will put students on a fast track to understanding embedded-system programming and applying what they learn to their projects.

Course Outcomes:

The student will be able to:

- Describe the differences between the general computing system and the embedded system, and also recognize the classification of embedded systems.
- Understand the architecture of the processor and its programming aspects (assembly level).
- Aware of interrupts, hyper threading and software optimization.
- Design real time embedded systems using the concepts of RTOS.

Skills:

- Explore to memory systems: cache mapping, virtual memory, and address translation.
- Analyze various system buses, protocols, and peripheral interaction.
- Implement and use various common software components of embedded systems.
- Analyze performance at the CPU, platform, and program levels, and to optimize various aspects of embedded programs.

Activities:

- Implementation of software on hardware platforms taking limitations such as memory size, processor capacity, and bandwidth into account.
- Development of reliable software taking fault tolerance and recovery into consideration.
- Development of correct and efficient software using fault detection and other test systems.

Syllabus

UNIT – 1 **12 Hours**

INTRODUCTION TO EMBEDDED SYSTEMS: Definition, Applications of ES, Embedded hardware units and devices, Embedded software, Design metrics in ES, Challenges in ES design, Host and target machines and getting embedded software into the target system.

UNIT – 2 **12 Hours**

ARCHITECTURE OF 8051:8051 Micro controller hardware, Input/output ports and circuits, External memory, Counter and timers, Serial data input/output, Interrupts.

UNIT – 3 **12 Hours**

ARM- EMBEDDED PROCESSOR: History, Architecture, Interrupt vector, Programming the ARM, ARM assembly language, Instruction set, Conditional execution, Arithmetic and logical compare.

UNIT – 4 **12 Hours**

ARM PROGRAMMING: Assembly programming, General structure of assembly language, Writing programs, Branch and load and store instructions, Read-only and read/write memory, Multiple register load and store.

UNIT – 5 **12 Hours**

REAL TIME OPERATING SYSTEMS: Introduction**,** Tasks and task states, Tasks and data, Reentrancy, Semaphores and shared data, Basic design principles, Inter process communication message queues, Mailboxes and pipes; Timer functions, Events, Memory management and interrupt routines in an RTOS environment.

Text Books:

1. Raj Kamal, “Embedded Systems”, 2nd Edition, Tata McGraw Hill, 2009.
2. Lyla B Das, “Embedded Systems an Integrated Approach”, 1st Edition, Pearson, 2012.

Reference Books:

1. David E. Simon, “An Embedded Software Primer”, 1st Edition, Pearson Education, 2008.
2. Wayne Wolf, “Computers as Components-principles of Embedded Computer system Design”, 1st Edition, Elseveir, 2009.
3. Labrosse, “Embedding system building blocks”, 2nd Edition, CMP Publishers, 2007.
4. Kenneth J. Ayala and Thomson, “The 8051 Microcontroller”, 3rd Edition, Thompson Delmar Learning, 2008.
5. Frank Vahid, Tony Givargis and John Wiley, “Embedded System Design, Microcontrollers”, 3rd Edition, Pearson Education, 2008.