PEIXXX: EMBEDDED CONTROL SYSTEMS

L	T	P	Cr
2	1	2	3.5

Introduction: Introduction to Embedded Systems, Its Architecture and system Model, Microprocessors &Microcontrollers, Introduction to the ARM Processor architecture, Embedded Hardware Building Block.

Microprocessor Architecture: Core Architecture, Reset, Power architecture, Low power modes, Clock Functions, Memory organization and system, addressing modes, instruction set, Input & Output port, Data Conversion, RAM & ROM Allocation, Timer programming, Exception Processing–Watch dog, Soft Resets and Interrupts, Communications – SPI, RS232, I2C, CAN and Ethernet, Analog-to-Digital Conversion System.

Embedded programming: C and Assembly language programming, Programming Style, Declarations and Expressions, Arrays, Qualifiers and Reading Numbers, Decision and Control Statements. Real-time Operating Systems (RTOS), Basic concepts of RTOS and its types, Task, process & threads, interrupt routines in RTOS, Multiprocessing and Multitasking, Preemptive and non-pre-emptive scheduling, Task communication shared memory, message passing, Concurrency, Re-entrancy, Intertask communication, Inter process Communication – synchronization between processes-semaphores, Mailbox, pipes, priority inversion, priority inheritance,
Implementation of RTOS with some case studies.

Development tools and Programming: Hardware and Software development tools, Code warrior tools- Project IDE, Compiler, Assembler and Debugger, JTAG and Hardware Debuggers, Interfacing Real Time Clock and Temperature Sensors with I2C and SPI bus.

Case Study: Embedded System Application using Microcontrollers: Product specification, Hardware design, Software design, System configuration, Integration of HW & SW, Product testing, Performance tools, Bench marking, Reports, User manual. – RTOS Micro Controller -issues in selection of processors.

Laboratory Work (if any):

Programming of microcontroller with Integrated development environment (IDE), Use of JTAG and Hardware Debuggers, Input Devices and Output Devices with their Programming, programming for Interrupts, Clock Functions, LCD interfacing, Interfacing Keypad and Switch Debouncing, ADC, DAC, Real Time Clock, Temperature Sensors with I2C and SPI bus. Interfacing to Motor, LCDs, Transducer, RS-232 Interface and their Examples.

Course learning outcome (CLO): After the completion of the course the students will be able to

1. Express the building block of microcontrollers and specifically S12X architecture.
2. Elucidate the C-programming using IDE like code warrior for S12X microcontroller and can develop the programs for timers, PWM etc.
3. Demonstrate the interfacing modules (ADC, LCD etc.) in control applications.
4. Express understanding of real time operating system.

Recommended Books:

1. *Barrett, S.F. and Pack, J.D., Embedded Systems, Pearson Education (2008).*
2. *Haung, H.W., The HCS12 / 9S12: An Introduction to Software and Hardware Interfacing, Delmar Learning (2007).*
3. *Fredrick, M.C., Assembly and C programming for HCS12 Microcontrollers, Oxford University Press (2005).*
4. *Ray, A.K., Advance Microprocessors and Peripherals – Architecture, Programming and Interfacing, Tata McGraw□Hill (2007)*

Evaluation Scheme:

Evaluation Elements	Weightage (%)
MST	20
EST	40
Sessionals (May include Assignments/ Projects/ Tutorials/ Quizes/ Lab Evaluations)	40