

EMBEDDED SYSTEMS DESIGN AND PROGRAMMING

VIII Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEC024	Core	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45		Tutorial Classes: 0		Practical Classes: Nil			Total Classes: 45	
<p>OBJECTIVES: The course should enable the students to:</p> <ol style="list-style-type: none"> I. Imbibe knowledge about the basic functions, structure, concepts and applications of Embedded Systems. II. Understand Real time operating system concepts. III. Design interfacing of switches, displays and stepper motor. IV. Analyze different tools for development of embedded software. V. Be acquainted the architecture of advanced processors. <p>COURSE OUTCOMES (COs):</p> <p>CO 1: Understand the basic concepts of embedded system and various applications and characteristics, formalisms for system design of embedded system design</p> <p>CO 2: Discuss the concepts of C and develop the C programming examples with Keil IDE, and understand the concepts of interfacing modules using embedded C.</p> <p>CO 3: Understand the basic embedded programming concepts in C and assembly language.</p> <p>CO 4: Understand the fundamentals of RTOS and its programming and Task communication, Task synchronization with its issues and techniques. Develop examples using embedded software and understand the debugging techniques.</p> <p>CO 5: Discuss the concepts of advanced processors like ARM and SHARC and protocols of I2C and CAN bus.</p> <p>COURSE LEARNING OUTCOMES (CLOs):</p> <ol style="list-style-type: none"> 1. Understand basic concept of embedded systems. 2. Analyze the applications in various domains of embedded system. 3. Develop the embedded system and Design process and tools with examples. 4. Understand characteristics and quality attributes of embedded systems, formalisms for system design. 5. Understand the basic programming of c and its looping structure. 6. Analyze the embedded C programming in Keil IDE, and compiling and building the hardware. 7. Understand different concepts of display and keyboard interfacing using embedded C. 8. Understand different concepts of serial communication using embedded C and user interfacing. 9. Analyse the programming on switches. 10. Understanding the programming language tools. 11. Understand different concepts of display and keyboard interfacing using embedded C. 12. Understand different concepts of stepper motor interfacing. 13. Understand and analyze the RTOS concepts for firmware development. 14. Remember how to choose an RTOS, task scheduling, semaphores and queues, hard real-time scheduling considerations. 15. Understand the task communication, its programming and Task synchronization with its issues and techniques. 16. Develop host and target machines for linking to embedded software. 17. Develop debugging techniques for testing on host machine with examples. 18. Remember the advanced processors such as ARM and SHARC. 19. Understand the bus protocols such as I2C and CAN bus. 20. Design an application based on advanced technological changes. 								

Unit-I	EMBEDDED COMPUTING	Classes: 09
Definition of embedded system, embedded systems vs. general computing systems, history of embedded systems, complex systems and microprocessor, classification, major application areas, the embedded system design process, characteristics and quality attributes of embedded systems, formalisms for system design, design examples		
Unit-II	PROGRAMMING EMBEDDED SYSTEMS IN C	Classes: 09
Embedded systems programming in C, binding and running embedded C program in Keil IDE, building the hardware; The Project Header (MAIN.H), The Port Header (PORT.H), Example: Restructuring the Hello Embedded World example.		
Unit-III	EMBEDDED C APPLICATIONS	Classes: 09
Basic techniques for reading from port pins, Example: Reading and writing bytes, Example: Reading and writing bits (simple version), Example: Reading and writing bits (generic version). Basic techniques for reading and writing from I/O port pins, LED interfacing, interfacing with keyboards, displays, Stepper motor interfacing.		
Unit-IV	INTRODUCTION TO REAL – TIME OPERATING SYSTEMS	Classes: 09
Tasks and Task States, Semaphores, and Shared Data; Message Queues, Mailboxes and Pipes, Timer Functions, Events, Semaphores and Queues, Hard Real-Time Scheduling Considerations, Interrupt Routines in an RTOS Environment. Embedded Software Development Tools: Host and Target machines, Linker/Locators for Embedded Software, Getting Embedded Software into the Target System; Debugging Techniques: Testing on Host Machine.		
Unit-V	INTRODUCTION TO ADVANCED ARCHITECTURES	Classes: 09
ARM and SHARC, Processor and memory organization and Instruction level parallelism; Networked embedded systems: Bus protocols, I2C bus and CAN bus.		
Text Books:		
<ol style="list-style-type: none"> Wayne Wolf, Computers and Components, Elsevier. Kenneth J. Ayala, The 8051 Microcontroller, 3rd Edition, Thomson. K. V. K. K. Prasad, Embedded / Real-Time Systems: Concepts, Design & Programming. Michael J. Pont, Embedded C, 2nd Edition, Pearson Education, 2008 		
Reference Books:		
<ol style="list-style-type: none"> Labrosse, Embedding system building blocks, CMP publishers. Raj Kamal, Embedded Systems, TMH. Micro Controllers, Ajay V Deshmukhi, TMH. Frank Vahid, Tony Givargis, Embedded System Design, John Wiley Raj kamal, Microcontrollers, Pearson Education. David E. Simon, An Embedded Software Primer, Pearson Education. Muhammad Ali Mazidi, Janice Mazidi, Janice Gillispie Mazidi, 8051 Microcontroller and Embedded Systems. 		

Web References:

1. <https://www.smartzworld.com/notes/embedded-systems-es/>
2. <http://notes.specworld.in/embedded-systems-es/>
3. <http://education.uandistar.net/jntu-study-materials>
4. <http://www.nptelvideos.in/2012/11/embedded-systems.html>

E-Text Books:

1. <https://www.scribd.com/doc/233633895/Intro-to-Embedded-Systems-by-Shibu-Kv>
2. http://www.ee.eng.cmu.ac.th/~demo/think/_DXJSq9r3TvL.pdf
3. <https://www.scribd.com/doc/55232437/Embedded-Systems-Raj-Kamal>
4. https://docs.google.com/file/d/0B6Cyt4eS_ahUS1LTkVXB1hxa00/edit
5. <http://www.ecpe.nu.ac.th/ponpisut/22323006-Embedded-c-Tutorial-8051.pdf>

