Course Title: Embedded System Programming **Credit: 3**
Course No: CSIT.425.3 **Number of period per week: 3+3**
Nature of the Course: Theory + Lab **Total hours: 45+45**
Year: Fourth, Semester: Eighth
Level: B. Sc. CSIT

1. Course Introduction

Embedded Systems are everywhere. Every time you look at your watch, answer the phone, take a picture, and drive cars you are interacting with an embedded system. They far outnumber traditional computers. Learning to design and program embedded systems is a critical skill that is necessary for many industry and scientific jobs. In this course you will learn the basics of designing, interfacing, configuring, and programming embedded systems.

2. Objectives

After completing the course students will know

- How building and loading programs differ from desktop or server computers
- Basic debugging techniques--a critical skill when working with minimally endowed embedded systems
- Handling different types of memory
- Interrupts, and the monitoring and control of on-chip and external peripherals
- Determining whether you have real-time requirements, and whether your operating system and application can meet those requirements
- Task synchronization with real-time operating systems and embedded Linux

3. Specific Objectives and Contents

Specific Objectives	Contents
• Define Embedded Systems and its components • Understand design of Embedded Systems • Discuss C language and other languages used for Embedded Programming	**Unit I: Introduction (5)** 1.1. What is Embedded System (ES)?, Real-time Systems, Components of ES 1.2. Requirements that Affect Design Choices, Embedded Design Examples (Digital Watch, Video Game Player) 1.3. Embedded Software Developer, C language for Embedded Programming, Other Embedded Languages

• Understand Hardware Basics of Embedded Systems • Discuss Embedded Processors and Communication Basics • Exemplify and Demonstrate Embedded Programs • Explain process of compiling, linking, and locating programs • Exemplify compiling, linking, and locating process • Describe and Exemplify Downloading and Debugging of Embedded Programs	**Unit II: Embedded Hardware & Software (12)** 2.1. Hardware Basics: Schematic Fundamentals, Memory Map, How to Communicate?, Processor, PXA255 XScale Processor, External Peripherals, Hardware Initialization 2.2. Embedded Programs: Hello World Program, LED Blinking Program, Role of Infinite Loop 2.3. Compiling, Linking and Locating: Build process, Compiling, Linking, Startup Code, Locating, Building the LED Blinking Program (compile, link and locate), Format the Output File, Makefiles 2.4. Downloading and Debugging: Downloading LED Blinking Program, Debug Monitors (Downloading and Running Programs with ReBoot) 2.5. Remote Debuggers, Emulators, Other Useful Tools
• Discuss memory system and types used in Embedded Systems • Demonstrate effect of Endianness in Embedded Software Development • Explain memory testing and problem related to this • Demonstrate techniques used for validating memory content	**Unit III: Memory (8)** 3.1. Types of Memory, Types of RAM and ROM, Hybrid Types, DMA, Endian Issues, Endianness in Devices and Networking 3.2. Memory Testing, Common Memory Problems, Electrical Wiring Problems, Missing Memory Chips, Improperly Inserted Chips 3.3. Developing Test Strategy: Data Bus Test, Address Bus Test, Device Test 3.4. Validating memory Content (Checksum & CRC), Using Flash Memory, Working with Flash Memory, Flash Drivers
• Discuss and exemplify Bit manipulation techniques • Demonstrate the use of serial device driver • Understand device driver design and APIs • Explain interrupt and use of interrupt service routines • Use peripherals and	**Unit IV: Peripherals and Interrupt (10)** 4.1. Control and Status Registers, Bit Manipulation (Testing, Setting, Clearing, Toggling, and Shifting Bits, Bitmasks, Bit fields), Struct Overlays 4.2. Device Driver Philosophy: Serial Device Driver, (Register Interface, State Vraiables, Initialization Routine, Device Driver API) 4.3. Testing Serial Device Driver, Extending Functionality, Device Driver Design 4.4. Interrupts: Overview, Priority, Levels and Edges, Enabling and Disabling, Interrupt Map, Interrupt

interrupts to improve LED Blinking program	Service Routine 4.5. Shared Data and Race Conditions, Improved LED Blinking Program, Working of Timers,
• Understand role of Real-time scheduling is Embedded Systems • Discuss Tasks and it implementation or execution in Embedded Systems • Describe Interrupts and Interrupt Handling Mechanism • Discuss operating system examples used in Embedded Systems	**Unit V: Operating Systems II (10)** 5.1. Purpose, Scheduler, Real-time Scheduling, Scheduling Points, Locking and Unlocking 5.2. Task States, Task Context, Task Priorities, Task Mechanics, Task Synchronization 5.3. Message Passing, Other Functionality, Interrupt Handling, RTOS Characteristics, When to use RTOS?, RTOS Selection Process 5.4. eCos Examples: Introduction, Task Mechanics, Task Synchronization, Message Passing, Interrupt handling 5.5. Embedded Linux Examples: Introduction, Accessing Hardware, Task Mechanics, Task Synchronization, Message Passing, Interrupt handling

Evaluation System

Undergraduate Programs							
External Evaluation	**Marks**	**Internal Evaluation**	**Weightage**	**Marks**	**Practical**	**Weightage**	**Mark**
End semester examination	60	Assignments	20%	20	Practical Report copy	25%	20
(Details are given in the separate table at the end)		Quizzes	10%		Viva	25%	
		Attendance	20%		Practical Exam	50%	
		Internal Exams	50%				
Total External	60	Total Internal	100%	20		100%	20
Full Marks 60+20+20 = 100							

External evaluation

1. **End semester examination:**
   It is a written examination at the end of the semester. The questions will be asked covering all the units of the course. The question model, full marks, time and others will be as per the following grid.

2. **External Practical Evaluation:**
   After completing the end semester theoretical examination, practical examination will be held. External examiner will conduct the practical examination according to the above mentioned evaluation. There will be an internal examiner to assist the external examiner. Three hours time will be given for the practical examination. In this examination Students must demonstrate the knowledge of the subject matter.

Full Marks: 100, Pass Marks: 45, Time: 3 Hrs

Nature of question	Total questions to be asked	Total questions to be answered	Total marks	Weightage
Group A: multiple choice*	20	20	20×1 = 20	60%
Group B: Short answer type questions	7	6	6×8 = 48	60%
Group C: Long answer type questions	3	2	2×16 =32	60%
			100	100%

Each student must secure at least 50% marks in internal evaluation in order to appear in the end semester examination. Failed student will not be eligible to appear in the end semester examinations.

Internal evaluation

Assignment: Each student must submit the assignment individually. The stipulated time for submission of the assignment will be seriously taken.

Quizzes: Unannounced and announced quizzes/tests will be taken by the respective subject teachers. Such quizzes/tests will be conducted twice per semester. The students will be evaluated accordingly.

Attendance in class: Students should regularly attend and participate in class discussion. Eighty percent class attendance is mandatory for the students to enable them to appear in the end semester examination. Below 80% attendance in the class will signify NOT QUALIFIED (NQ) to attend the end semester examination.

Presentation: Students will be divided into groups and each group will be provided with a topic for presentation. It will be evaluated individually as well as group-wise. Individual students have to make presentations on the given topics.

Mid-term examination: It is a written examination and the questions will be asked covering all the topics in the session of the course.

Discussion and participation: Students will be evaluated on the basis of their active participation in the classroom discussions.

Instructional Techniques: All topics are discussed with emphasis on real-world application. List of instructional techniques is as follows:

- Lecture and Discussion
- Group work and Individual work
- Assignments
- Presentation by Students
- Quizzes
- Guest Lecture

Students are advised to attend all the classes and complete all the assignments within the specified time period. If a student does not attend the class(es), it is his/her sole responsibility to cover the topic(s) taught during that period. If a student fails to attend a formal exam/quiz/test, there won't be any provision for re-exam. Unless and until the student clears one semester he/she will not be allowed to study in the following semesters.

Laboratory Work

Students should practice small scale Embedded programs that uses processor architecture, memory system, peripherals and interrupt. Besides this student should use Embedded OS features discussed in class.

Prescribed Text

- Programming Embedded Systems, 2nd Edition, Anthony Massa, Michael Barr, O'Reilly Media, Inc, *2006*
- Computers as Components: Principles of Embedded Computing System Design, W. Wolf, Morgan Kaufmann, Second Edition, 2008.
- Introduction to Embedded Systems, A Cyber-Physical Systems Approach, 2011
- Introduction to Embedded Systems, David Russell, 2010.