COURSE TITLE : EMBEDDED SYSTEMS
COURSE CODE : 5041
COURSE CATEGORY : A
PERIODS/WEEK : 4
PERIODS/SEMESTER : 52/5
CREDITS : 4

TIME SCHEDULE

MODULE	TOPICS	PERIODS
1	AVR Microcontroller Architecture.	12
2	Assembly Language Programming.	14
3	AVR Programming Using Embedded C.	14
4	Embedded Systems.	12
TOTAL		**52**

Course General Outcome :

Module	GO	On completion of the study of this course the students will be able:
1	1	To understand AVR architecture.
2	2	To understand basics of AVR programming.
3	3	To understand AVR programming using C.
	4	To understand AVR timers and interrupts.
4	5	To understand the architecture of embedded systems and embedded OS.
	6	To know about advanced development boards.

GO - General Outcome

On the completion of the study the student will be able:

MODULE I AVR MICROCONTROLLER ARCHITECTURE

1.1.0 To understand AVR architecture

1.1.1 To describe about AVR family.
1.1.2 To list the features of AVR family.

1.1.3 To compare various members of the AVR family.
1.1.4 To compare the AVR with microcontroller offered by other manufactures.
1.1.5 To explain the block diagram of AVR ATmega32 microcontroller.
1.1.6 To explain the General purpose registers of ATmega32 microcontroller.
1.1.7 To explain the data memory of ATmega32 microcontroller.
1.1.8 To explain I/O memory (SFRs).
1.1.9 To describe Internal data SRAM of ATmega 32 microcontroller.
1.1.10 To compare SRAM and EEPROM in ATmega32 chips.
1.1.11 To illustrate ATmega32 status register.
1.1.12 To explain different addressing modes of ATmega 32 microcontroller.

MODULE II ASSEMBLY LANGUAGE PROGRAMMING

2.1.0 To understand the basics of AVR programming.

2.1.1 To explain the structure of Assembly Language Program.
2.1.2 To describe AVR Microcontroller Data Formats and Assembler Directives
2.1.3 To list the steps to create an AVR Assembly Language Program
2.1.4 To explain the Data Transfer, Arithmetic and Logic Instructions
2.1.5 To explain the rotate and shift instructions
2.1.6 To explain the Branch Instructions and Looping
2.1.7 To explain the Call Instructions and Stack
2.1.8 To describe AVR Time Delay and Instruction Pipeline
2.1.9 To explain Delay Calculation for the AVR
2.1.10 To describe I/O Port Programming in AVR
2.1.11 To write Simple Assembly Language Programs
2.1.12 To explain Macros
2.1.13 To compare Macros and Subroutines

MODULE III AVR PROGRAMMING USING EMBEDDED C

3.1.0 To understand AVR Programming in C.

3.1.1 To describe data types and time delays in C.
3.1.2 To describe I/O programming in C.
3.1.3 To explain logic operations in C.
3.1.4 To describe data conversion programs in C.
3.1.5 To explain data serialization in C.
3.1.6 To describe memory allocation in C.

3.2.0 To understand AVR timers and interrupts.

3.2.1 To explain programming timers 0, 1, and 2
3.2.2 To describe AVR interrupts
3.2.3 To explain the programming of timer interrupts
3.2.4 To explain programming external hardware interrupts
3.2.5 To state interrupt priority in the AVR microcontroller
3.2.6 To explain about serial Communication- I^2C and SPI
3.2.7 To describe ATmega32 connection to RS232

MODULE IV EMBEDDED SYSTEMS

4.1.0 To understand the architecture of embedded systems and embedded OS.

4.1.1 To define an embedded system
4.1.2 To explain the concept of embedded systems
4.1.3 To list the characteristic features of an embedded system
4.1.4 To explain the architecture of an embedded system
4.1.5 To list the application areas of embedded system
4.1.6 To explain the specialities of embedded system
4.1.7 To list the types of embedded operating system
4.1.8 To describe various activities of an embedded OS such as task, task scheduling, context switching, mutual exclusions and inter task communications
4.1.9 To describe about memory management and timer services
4.1.10 To explain the general architecture of an embedded operating system
4.1.11 To state the role of kernal in embedded OS
4.1.12 To list the different categories of embedded OS and give examples for each

4.2.0 To know about advanced development boards.

4.2.1 To describe the concept of arduino development board
4.2.2 To describe the concept of raspberry pie development board

CONTENTS

MODULE II Assembly language programming

Structure of assembly language program - data formats - assembler directives - AVR assembly language programming - data transfer - arithmetic and logic instructions - shift and rotate instructions - branch instructions and looping - call instructions and stack - time delay and instruction pipeline - delay calculation - I/O port programming in AVR - simple assembly language programs - macros - comparison with subroutines.

MODULE III AVR programming using embedded C

Data types and time delays in C - I/O programming - logic operations - data conversion programs - data serialization - memory allocation - programming of timers 0 - timer 1 - timer2- AVR interrupts - programming of timer interrupts - programming external hardware interrupts - interrupt priority in AVR microcontroller - serial communication- I^2C and SPI - connection to RS232.

MODULE IV Embedded systems

Embedded system - concept - characteristic features - architecture - application areas - specialities - embedded operating system - types - activities of an embedded OS like task, task scheduling, context switching, mutual exclusions and inter task communications - memory management and timer services - general architecture of OS - kernal - categories of embedded OS - examples - concept of arduino and raspberry pie development boards.

Text Book

1. The AVR Microcontroller and Embedded Systems using assembly and C - Muhammad Ali Mazidi, Sarmad Naimi and Sepehr Naimi - Pearson Education.
2. Embedded / Real Time Systems Concept, Design and Programming The ultimate reference - Dr. K V K Prasad (Dreamtech).