COURSE TITLE : MICROCONTROLLERS
COURSE CODE : 6132
COURSE CATEGORY : A
PERIODS/WEEK : 5
PERIODS/SEMESTER : 75
CREDITS : 5

TIME SCHEDULE

MODULE	TOPICS	PERIODS
1	AVR Microcontrollers & Assembly Language Programming	**18**
2	AVR Programming in C	**18**
3	Timer, Interrupt, Programming	**19**
4	Interfacing sub systems with AVR	**20**

Course General Outcomes:

Sl.	G.O	On completion of this course the student will be able :
1	1	To understand AVR Microcontrollers and its Architecture
	2	To know AVR Assembly Language Programming
	3	To apply Assembly Language Instructions in AVR
2	1	To understand I/O Port Programming
	2	To understand AVR Programming in C
3	1	To understand Timer and Interrupt Programming
4	1	To understand the interfacing of various systems with AVR microcontroller

Specific Outcomes:

MODULE –I AVR Microcontrollers & Assembly Language Programming

1.1 To understand AVR Microcontrollers and its Architecture

1.1.1 To explain the basics of Microcontrollers
1.1.2 To Compare and contrast Microcontrollers and Microprocessors
1.1.3 To describe AVR family of Microcontrollers
1.1.4 To list the AVR features
1.1.5 To explain AVR Architecture with block diagram

1.2 To comprehend AVR Assembly Language Programming

1.2.1 To state the usage of instructions in Data Memory
1.2.2 To write AVR data formats
1.2.3 To explain the Program counter and Program ROM space
1.2.4 To explain the RISC and Harvard architecture in AVR

1.3 To apply Assembly Language Instructions in AVR

1.3.1 To Explain Branch instructions and looping
1.3.2 To Explain Call instructions and stack
1.3.3 To explain Time Delay Loop

MODULE – II AVR Programming in C

2.1 To Describe I/O Port programming

2.1.1 To explain I/O port programming in AVR

2.1.2 To explain I/O bit Manipulation Programming .

2.2 To understand AVR Programming in C

2.2.1 To list Data types in C

2.2.2 To explain Time Delays in C

2.2.3 To discuss I/O programming in C

2.2.4 To explain Logic Operations in C

MODULE – III Interfacing sub systems with AVR

3.1 To understand Timer and Interrupt Programming in C

3.1.1 To describe Timers 0,1, and 2 programming

3.1.2 To explain Counter Programming

3.1.3 To explain AVR interrupts

3.1.4 To illustrate Timer interrupts

MODULE –IV Interfacing sub systems with AVR

4.1 To understand the interfacing of various systems with AVR microcontroller

4.1.1 To explain AVR serial port programming

4.1.2 To Illustrate LCD interfacing

4.1.3 To Illustrate Keyboard interfacing

4.1.4 To Illustrate ADC interfacing

4.1.5 To Illustrate DAC interfacing

4.1.6 To Illustrate Sensor interfacing

CONTENT DETAILS

MODULE –I AVR Microcontrollers & Assembly Language Programming

Basics of Microcontrollers - Compare and contrast Microcontrollers and Microprocessors –Overview of AVR family of Microcontrollers - AVR features - AVR Architecture with block diagram – The general Purpose Registers in the AVR

AVR Data Memory-Using instructions with the Data Memory – AVR status Register - AVR data formats - Program counter and Program ROM space- RISC and Harvard architecture in AVR

Branch instructions and Looping- Unconditional branch instruction - Call instructions and stack – AVR Time delay and Instruction pipeline

MODULE – II AVR Programming in C

I/O port programming in AVR – I/O bit Manipulation Programming.

Data types and Time Delays in C - I/O programming in C –

Logic Operations in C- Data Conversion Prpograms in C – Data Serialization in C

MODULE – III Timer, Interrupt, Programming

Programming Timers 0, 1, and 2 - Counter Programming - AVR interrupts – Programming Timer Interrupts –Programming External Hardware interrupts – Interrupt priority in the AVR -

MODULE –IV Interfacing sub systems with AVR

Basics of Serial Communication – ATMEGA32 connection to RS232 - AVR serial port programming in C- LCD interfacing - Keyboard interfacing - ADC interfacing -- DAC interfacing - Sensor interfacing

Note: Assembly Language Programming is not necessary from Module II onwards.

TEXT BOOK:

1. The AVR Microcontroller and Embedded Systems Using Assembly and C By Muhammad Ali Mazidi, Sarmad Naimi, & Sepehr Naimi - Pearson Education

REFERENCES:

1. Introduction to Embedded Systems - Shibu K.V, Mc Graw Hill.First Edition
2. Embedded C - Michael J. Pont, Pearson Education, Second Edition
3. Embedded Systems - Raj Kamal, , Mc Graw Hill, Second Edition

WEB SITE:

http://freevideolectures.com