

Program : Diploma in Electronics Engineering / Electronics and Communication Engineering / Biomedical Engineering	
Course Code : 5041	Course Title: Embedded Systems
Semester : 5 / 6 / 5	Credits: 4
Course Category: Program Core / Elective / Elective	
Periods per week: 4 (L:3, T:1, P:0)	Periods per semester: 60

Course Objectives:

- Introduce the importance of embedded systems
- Introduce ATMEGA 32 microcontroller and its suitability for embedded systems
- Familiarize embedded C programming
- Familiarize serial communication and data transfer between embedded systems

Course Prerequisites:

Topic	Course code	Course name	Semester
Basics of programming concepts	3045	Programming in C Python programming	3
	4049	Python programming	4
Basics of Microcontroller architecture	4041	Microcontroller and Application	4

Course Outcomes:

On completion of the course, the student will be able to:

CO _n	Description	Duration (Hours)	Cognitive Level
CO1	Explain the basics of embedded systems and its architecture	13	Understanding
CO2	Make use of AVR Microcontrollers to develop embedded programs using embedded C	16	Applying
CO3	Make use of AVR microcontroller to interface with various peripheral devices.	19	Applying

CO4	Familiarize RTOS	10	Understanding
	Series Test	2	

CO - PO Mapping

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2						
CO2	3	3					
CO3	3	3					
CO4	3						

3-Strongly mapped, 2-Moderately mapped, 1-Weakly mapped

Course Outline:

Module Outcomes	Description	Duration (Hours)	Cognitive Level
CO1	Explain the basics of embedded systems and its architecture		
M1.01	Describe embedded system, illustrate difference from general purpose computer	2	Understanding.
M1.02	Classify embedded systems, explain application areas and summarize purpose of embedded systems	2	Understanding
M1.03	Distinguish Hardware and software components of embedded systems	2	Understanding
M1.04	Describe the basic blocks in a typical embedded system	2	Understanding
M1.05	Describe Memory, Sensors, Actuators and I/O sub-systems	2	Understanding
M1.06	Distinguish Communication Interfaces – On board and external interfaces	3	Understanding

Contents:

Embedded Systems - Definition, difference from general purpose computers - Classification of embedded systems, Application areas,

Components of embedded system hardware, and Software embedded into the system

Architecture of embedded system – Building blocks of an embedded system , Core of embedded system – categories, Memory –ROM and RAM, Sensors, actuators and I/O sub

systems (LED, Opto coupler, Relay, Stepper motor). On board (I2C, SPI, UART) and external communication interfaces (RS 232, USB, Bluetooth, Wifi)

CO2	Make use of AVR Microcontrollers to develop embedded programs using embedded C		
M2.01	Familiarize AVR controllers family members and criteria to select a microcontroller	2	Understanding.
M2.02	Explain block diagram of Atmega32 and its blocks	2	Understanding
M2.03	Illustrate Registers, Memory organization, Status register, Program counter, I/O ports and its registers, Interrupts, priority of interrupts	3	Understanding.
M2.04	Illustrate Timers in AVR	3	Understanding.
M2.05	Develop embedded C programs for logic operations, data conversions and I/O operations.	2	Applying.
M2.06	Develop embedded C programs for time delay, time delay calculation. Timer programming and timer interrupts handling	2	Applying.
M2.07	Develop Programs to handle external hardware interrupts and programming based on priority of interrupts	2	Applying.
	Series Test - I	1	

Contents:

AVR Microcontroller Architecture - Comparison of AVR family members and Selection of a microcontroller, ATmega32- Simplified Block diagram of ATmega32 microcontroller - Registers, - data memory - I/O memory -SFRs - internal data SRAM, Status register, Program Counter and Program ROM space, I/O ports, Registers associated with I/O ports, Timers-0,1,2 (Block level) associated registers, Interrupts.

AVR programming using embedded C: Data types, I/O programming, logic operations - data conversion programs - time delays- programming of timers 0 - timer 1 - timer2, AVR interrupts - programming of timer interrupts - programming external hardware interrupts - interrupt priority in AVR microcontroller.

CO3	Make use of AVR microcontroller to interface various peripheral devices using embedded C		
M3.01	Illustrate the need for interfacing, and types of interfacing devices	2	Understanding
M3.02	Illustrate the interfacing of LED, Push button, Relay and Optocoupler with AVR	2	Applying
M3.03	Illustrate Sensors and Seven segment Display interfacing with AVR	2	Applying

M3.04	Make use of AVR to realize LCD and Keyboard interfacing	3	Applying
M3.05	Make use of AVR microcontroller to interface DC motor, Servo motor and stepper motor.	3	Applying
M3.06	Make use of AVR microcontroller to interface RTC and ADC.	2	Applying
M3.07	Realize I2C interfacing with AVR	3	Applying
M3.08	Understand SPI interface with AVR	2	Understanding
Contents: Interfacing of LED, Push button, Relay, Opto Coupler, Sensors (Temperature sensor, IR sensor) Seven segment Display, LCD, Keyboard Interfacing, Motor (DC, Servo, Stepper), RTC Interfacing, ADC interfacing On-board communication interfaces with AVR – I2C interfacing(like real time clock interfacing)and basics of SPI interfacing with AVR.			
CO4	Familiarize RTOS		
M4.01	Describe Kernel,Operating System Architecture and types	2	Understanding
M4.02	ExplainRTOS Kernel functions	3	Understanding
M4.03	ListSelection criteria for RTOS	2	Remembering
M4.04	Outline Micro C/OS-II and its services	2	Understanding
M4.05	List popular Real Time Operating Systems.	1	Remembering
	Series Test - II	1	
Contents: Operating system basics:- Kernel, types of operating systems- GPOS, RTOS. Real time operating systems:- Tasks, process, threads, multiprocessing and multi-tasking, task scheduling, types, threads and process scheduling, task communication, task synchronization, device drivers. List Selection criteria for RTOS. Overview of Micro C/OS-II and its services. List popular Real Time Operating Systems (any 10).			

Text / Reference:

T/R	Book Title/Author
T1	K.V. Shibu, Introduction to Embedded Systems, 2e, McGraw Hill Education India, 2016.
T2	Rajkamal, Embedded Systems Architecture, Programming and Design, TMH, 2003
R1	Muhammad Ali Mazidi, SarmadNaimi, &SepehrNaimi, The AVR Microcontroller and Embedded Systems Using Assembly and C, Pearson Education
R2	Michael J. Pont, Embedded C, Pearson Education, Second Edition

Online Resources:

Sl.No	Website Link
1	https://www.studyelectronics.in/embedded-programming-tutorial-chapter-1-beginners/
2	https://www.tutorialspoint.com/embedded_systems/index.htm
3	https://embeddedschool.in/avr-microcontroller-programming/
4	https://learn.sparkfun.com/tutorials/i2c/all
5	https://www.best-microcontroller-projects.com/i2c-tutorial.html
6	https://www.electronicwings.com/avr-atmega/atmega1632-i2c
7	https://www.electronicwings.com/avr-atmega/atmega1632-spi
8	https://www.intel.com/content/www/us/en/docs/programmable/683525/21-3/overview-of-the-microc-os-ii-rtos.html