Government of Karnataka
DEPARTMENT OF TECHNICAL EDUCATION

Program	**Electronics and communication Engg.**	**Semester**	**V**
Course Name	**Embedded Technology**	**Type of Course**	**Integrated**
Course Code	**25EC51IC**	**Contact Hours**	**104 Hours/sem**
Teaching Scheme	**L: T:P :: 4:0:4**	**Credits**	**06**
CIE Marks	**50**	**SEE Marks**	**50 (Theory)**

1. Rationale:

This framework offers a comprehensive approach to understanding embedded systems and ARM controllers by combining foundational concepts, practical applications, and emerging technologies. It begins with an introduction to embedded systems, their classifications, and applications across various domains, followed by a detailed exploration of ARM architecture, instruction sets, and programming in both C and assembly.

Hands-on exercises, such as interfacing sensors, actuators, and motors, reinforce theoretical knowledge through practical implementation, while real-world applications like traffic light systems and cloud-based sensor data transfer ensure relevance to industry needs.

The focus on emerging trends, including AI, IoT, and edge computing, along with activities like hardware familiarization, simulation-based tasks, and report preparation, provides a holistic understanding of embedded system design and functionality.

2. **Course Outcomes:** At the end of the course, the student will be able to:

CO-01	Analyze the typical components of an embedded system focusing on communication interfaces.
CO-02	Summarize the architectural features of ARM controllers and their applications.
CO-03	Examine the ARM controller instruction set and develop programs in assembly and C to solve simple real-world problems.
CO-04	Evaluate exceptions and interrupt handling mechanisms and discuss emerging trends in embedded systems.

3. Course content:

WEEK	CO	PO	Theory	Practice
1	1	1,7	**Introduction to Embedded Systems** 1. Definition and characteristics. 2. Classification of embedded systems. 3. Applications of Embedded Systems in various fields, Differences between embedded and general-purpose systems with examples. 4. Block diagram of Embedded System.	Video Demonstration of working of any two Embedded Systems and prepare a report on it. 1. Smart watch 2. ATM 3. Smart vehicles
2	1	1,7	1. Core of embedded system. Memory: Types of ROM & RAM, Role of Sensors and Actuators. Communication Interfaces: 2. Onboard Interface: I2C, SPI, 3. Onboard Interface: UART, One Wire and Parallel interface. 4. External Interface: RS232, RS485, USB.	1. Visit the nearest Home Appliances shop to know the features and working of different embedded systems available in that shop and prepare a report on it. 2. a) Salient features of LPC2148 b) Interpretation of datasheet of LPC2148. (block diagram, pin description etc.)
3	1	1,4, 7	1. External Interface: IEEE1394, Infrared, Bluetooth, WiFi, Zigbee and GPRS. 2. Little Endian and Big Endian concept, CISC vs. RISC, Harvard vs. Von-Neumann architectures. 3. Timing circuits: reset, watchdog timer, 4. Timing circuits: Brownout protection and RTC.	1. Familiarization of ARM-7 Development Board Kit, I/O pin details of kit, Integrated peripherals/ports/bus, Interfacing modules. 2. Familiarization of a) Kiel µVision4 simulation IDE. b) Philips Flash Utility.
4	2	1,2, 3,4, 7	**Introduction to ARM controllers.** 1. ARM Definition, Importance. ARM Design Philosophy: Power Consumption, Code Density, Price, Size, Debug Technology, Core Processor. 2. Features of ARM Instruction Set: Listing and elaborating the features.	1. Structure of Embedded C syntax, defining Ports, directions, etc. 2. Sample Embedded C program e.g., Simulation of storing bytes in ports and verification.

			3. Embedded system Hardware: Block Diagram, ARM Processor, Controllers, Peripherals, Bus. 4. ARM Bus Topology : Block Diagram, Master/slave, Physical/protocol.	
5	2	1,2, 3,4, 7	1. AMBA Bus Protocol : ASB, APB, AHB. 2. Embedded System Software: Block Diagram, Hardware, Initialization, Device Drivers, Operation System, Application. 3. Applications of ARM Processor: Listing of application fields and usage. 4. ARM Core Dataflow model: Architecture and sub blocks.	1. Embedded C program to blink 8 LEDS at a time in ARM kit. 2. Embedded C program to blink alternate LEDs in ARM kit.
6	2	1,2, 3,4, 7	1. ARM Core Dataflow model: Architecture and sub blocks. 2. Registers: General Purpose Registers, CPSR/SPSR bit configuration. 3. Processor Modes : Privileged and Non-Privileged modes. 4. Banked Registers: 17 registers in user/system mode and 20 hidden registers.	1) Embedded C Program to ON/OFF relay for every 5 secs by interfacing relay card to ARM kit. 2) Embedded C program to ON buzzer for 2 secs and off for 3 sec by interfacing Buzzer to ARM kit.
7	2	1,2, 3,4, 7	1. Pipeline: Concept, 3-stage operation. 2. Pipeline Characteristics: Illustration with an example and Listing characteristics. 3. Exceptions, Interrupts and Vector Table. 4. Structure of ARM Assembly Module : Syntax, illustration with example.	1. Embedded C program to rotate stepper motor in clockwise and anticlockwise direction by interfacing Stepper motor to ARM kit. 2. Embedded C program to rotate DC motor in clockwise and anticlockwise direction by interfacing DC motor to ARM kit.
8	2	1,2, 3,4, 7	1. Assembler directives : Definition, syntax with examples. 2. Introduction to Thumb: Differences between ARM and Thumb Instructions, Thumb Register usage (only listing). 3. ARM CPU core Instruction set types: Thumb -16 bit, ARM-32 bit, ARM assembly Instruction format Syntax illustration with example. 4. Condition codes : Suffix, flags, Description.	1. Embedded C program to generate Triangular Wave by interfacing DAC to ARM kit. 2. Embedded C program to generate Sine Wave by interfacing DAC to ARM kit.

9	3	1,2, 3,4, 7	1. Barrel Shifter: Block diagram, LSL, LSR, ASR, ROR, RRX. 2. Classifications of Instruction Sets, Data processing Instructions with examples. 3. Arithmetic and Logical instructions with examples. 4. Simple Assembly Programs with logic and illustration of $6x^2 - 9x + 2$ mathematical equations and square of a number (1 to 9) using a lookup table.	1. Simulate Assembly Level Program (ALP) to compute $6x^2 - 9x + 2$. 2. Simulate Assembly Level Program (ALP) to find the square of a number (1 to 9) using a lookup table.
10	3	1,2, 3,4, 7	1. Branching and Load and Store instructions with examples Simple Assembly Programs with logic and illustration. 2. a) Program to find the sum of an array of 16-bit numbers and store the 32-bit result. b) Program to find the length of null terminated string. 3. a) Program to convert Hexadecimal to ASCII. b) Program to convert ASCII to Hexadecimal. 4. a) Program to find the smallest number. b) Program to find the largest number.	1. Simulate Assembly Level Program (ALP) to find the sum of an array of 16-bit numbers and store the 32-bit result. 2. Simulate Assembly Level Program (ALP) to find the length of null terminated string.
11	3	1,2, 3,4, 7	1. Exceptions Handling (Procedure). 2. ARM Processor Exceptions and Modes (all seven modes with purpose) 3. Vector Table (mode, offset, Branching) 4. Exception Priorities and Link Register Offsets.	1. Simulate Assembly Level Program (ALP) to find the smallest / largest number. 2. Simulate Assembly Level Program (ALP) to Convert Hexadecimal to ASCII and vice versa.
12	4	1,2, 3,4, 7	1. Interrupts (IRQ, FIQ, SWI) 2. Assigning Interrupts (Procedure) 3. Interrupt Latency (Nested, Prioritization) 4. IRQ and FIQ Exceptions, Flow Block Diagram Enabling and Disabling IRQ and FIQ Exceptions.	1. Interface a temperature sensor (e.g., LM35) with an ARM controller and display the temperature. **OR** 2. Write a program to capture sensor data from Temperature/ PIR/ ultrasonic and gas sensors and transfer to cloud using Embedded C programming

13	4	1,2, 3,4, 7	Emerging Trends in Embedded Systems 1. Edge Computing. 2. Artificial Intelligence (AI) 3. Machine Learning (ML) 4. IoT (Internet of Things)	Implement a traffic light system where different LEDs represent red, yellow, and green traffic lights. Implement time delays for each state. **OR** Implement any real time application using emerging technologies.

4. References.

- ➢ ARM System Developer’s Guide by Andrew N. Sloss, Dominic Symes, Chris Wright
- ➢ Introduction to Embedded Systems by Shibhu K.V.
- ➢ Embedded Systems: A Contemporary Design Tool – James K. Peckol.
- ➢ AVR Microcontroller and Embedded Systems – Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi
- ➢ ARM Assembly Language –William Hohl, CRC Press
- ➢ ARM Programming Techniques –from ARM website
- ➢ Embedded Systems: A Contemporary Design Tool-James K.
- ➢ LPC 2148 USER MANUAL

5. CIE Assessment Methodologies:

Sl. No	CIE Assessment	Test Week	Duration (minutes)	Max marks	
1.	CIE-1 Theory Test	4	90	50	Average of all CIE=50 Marks
2.	CIE-2 Practice Test	7	180	50	
3	CIE-3 Theory Test	10	90	50	
4.	CIE-4 Practice Test	13	180	50	
5	CIE-5 Portfolio evaluation of all the activities through Rubrics.	1-13		50	
				Total	50 Marks

6. Practice Assessment Methodologies:

Sl. No	SEE – Theory Assessment	Duration	Exam Paper Max marks	Exam Paper Max Marks scale down to (Conversion)	Min marks to pass
1.	Semester End Examination-Theory	3 Hours	100	50	20

7. CIE Theory Test model question paper:

Program	**Electronics and Communication Engg.**			**Semester -V**	
Course Name	**Embedded Technology**			**Test**	**I/III**
Course Code	**25EC51IC**	**Duration**	**90 min**	**Marks**	**50**
Name of the Course Coordinator:					
Note: Answer any one full question from each section. Each full question carries equal marks.					
Q.No	**Questions**		**Cognitive Level**	**Course Outcome**	**Marks**

Section - 1

1	a) Explain the key characteristics of an embedded system.	L2	1	7
	b) Compare the embedded systems and general-purpose systems with suitable examples.	L2		8
	c) Explain the block diagram of an embedded system with detailed descriptions of each component.	L2		10
2	a) Discuss the role of sensors and actuators in an embedded system, with examples of their applications.	L2	1	7
	b) Explain the Little endian and Big endian concepts and their importance in embedded systems.	L2		8
	c) Classify embedded systems and provide examples for each category.	L2		10
Section - 2				
3	a) Describe the differences between onboard communication interfaces such as I2C, SPI, and UART.	L3	1	7
	b) Develop an embedded C program to blink 8 LEDS at a time in ARM.	L3	2	8
	c) Illustrate the AMBA Bus Protocols (ASB, APB, AHB) and their significance in communication within embedded systems.	L3	2	10
4	a) Illustrate the role of watchdog timers in enhancing the reliability of embedded systems.	L3	1	7
	b) Develop an embedded C program to blink alternate LEDS in ARM.	L3	2	8
	c) Discuss the benefits of ARM processors in embedded systems, focusing on power consumption and code density.	L3	2	10
Note for the course coordinator: Each question may have one, two or three subdivisions. Optional questions in each section carry the same weightage of marks, cognitive level and course outcomes.				

Signature of the Course Coordinator	Signature of the HOD	Signature of the IQAC Chairman

8. CIE Practice Test model question paper:

Program	**Electronics and Communication Engg.**			**Semester**	**V**
Course Name	**Embedded Technology**			**Test**	**II/IV**
Course Code	**25EC51IC**	**Duration**	**180 min**	**Marks**	**50**
Name of the Course Coordinator:					

Questions	CO	Marks
Writing two programs and execution of any one program.		50
Scheme of assessment		

a) Writing two programs . **b) Conduction of any one program.** **d) Result** **e) Viva**	**20** **10** **10** **10**
Total Marks	**50**

Signature of the Course Coordinator **Signature of the HOD** **Signature of the IQAC Chairman**

9. Suggestive Activities:

The List is an Example and not inclusive of all possible activities of the course. Students and Faculty are encouraged to choose activities that are relevant to the topic.

Sl.No.	Suggested Activities
01	Use a keypad to enter a password for securing a system. If the correct password is entered, the system will activate (e.g., open a door or arm/disarm an alarm).
02	Use a soil moisture sensor to monitor the soil's moisture level and activate a water pump automatically when the moisture level is low.
03	Interface an IR sensor with LPC2148 to detect objects and activate devices like motors or alarms.
04	Generate basic tones (beeps or simple melodies) using LPC2148's DAC or PWM outputs to create sound.

10. Rubrics for Assessment of Activity (Qualitative Assessment):

Sl. No.	Dimension	Beginner	Intermediate	Good	Advanced	Expert	Students Score
		10	20	30	40	50	
1		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	40
2		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	30
3		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	50
4		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	20
	Average Marks=(40+30+50+20)/4=35						**35**

Note: Dimension and Descriptor shall be defined by the respective course coordinator as per the activities

11. Equipment/software list with Specification for a batch of 30 students:

Sl.No.	Particulars	Specifications	Quantity
01	Computers	i5/i7/i10 processor, 8/16 GB RAM , 512 GB SSD, 2 GB graphics card.	30
02	LPC2148 Development Board/Kit with RS232 / USB cable and adaptor.		30
03	Interfacing Modules	Buzzer, LED, DAC, Traffic Light, Stepper Motor, DC Motor and Temperature Sensor	5 Each
04	Integrated Development Environments (IDEs)	Keil µVision	
05	Digital multimeter		20
06	Dual trace oscilloscope	30MHz	20