GUJARAT TECHNOLOGICAL UNIVERSITY

BRANCH NAME: INSTRUMENTATION & CONTROL ENGINEERING (17)

SUBJECT NAME: EMBEDDED SYSTEM DESIGN

SUBJECT CODE: 2171711

B.E. 7th SEMESTER

Type of course: Core Engineering

Prerequisite:

1. Fundamental of Digital Logic Design, Register array, flip-flops, counter..
2. Decoder, Logic Gates, Number systems, etc.
3. 8051 Micro controller and its application
4. Assembly language and C language Programming

Rationale: Students should be familiarized with concepts of Embedded Systems and also have knowledge of programming, interfacing, debugging and implementing standalone systems for varied range of applications. This course is designed with two complementary goals: 1. to understand the scientific principles and concepts behind embedded systems, and 2. to obtain hands-on experience in programming embedded systems.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks						Total Marks
				Theory Marks			Practical Marks			
L	T	P	C	ESE (E)	PA (M)		PA (V)		PA (I)	
					PA	ALA	ESE	OEP		
4	0	2	6	70	20	10	20	10	20	150

L- Lectures; T- Tutorial/Teacher Guided Student Activity; P- Practical; C- Credit; ESE- End Semester Examination; PA- Progressive Assessment; OEP-Open Ended problem; AL-Active learning

Content:

Sr No	Content	Total Hr	Weightage %
1	**Introduction to embedded system** Definition of Embedded System, Embedded Systems Vs General Computing Systems, History of Embedded Systems, Introduction to Real Time Embedded system and general purpose computers, Embedded system components, Embedded System Design Process, Classification of an embedded system, Examples of an embedded system, Major Application areas	2	4
2	**Introduction to different embedded System Architectures** 2.1 Instruction Set Architecture 2.1.1 CISC and RISC instruction set architecture 2.2 Basic Embedded Processor/Microcontroller Architecture 2.2.1 CISC Examples 2.2.1.1 Motorola (68HC11) Example 2.2.1.2 8051 2.2.2 RISC Example	2	4

	2.2.2.1 ARM 2.2.2.2 AVR 2.2.3 DSP Processors 2.2.4 Harvard Architecture 2.2.4.1 PIC (Note:- Compare structures advantages and applications of different architectures.)		
3	**ARM Processor:** 3.1 ARM processor family, Application of ARM Processor, The Acorn RISC machine**,** Architectural inheritance, The ARM programmer's model, ARM development tools 3.2 Memory System Architecture 3.2.1 memory types:- RAM,ROM, UVROM, FLASH Memory, DRAM 3.2.2 Memory maps, Registers and addresses.	2	4
4	**ARM instruction set and Assembly programming:** Data processing instructions, Data transfer instructions, Control flow instructions, Conditional execution, ARM Condition codes, Software interrupt (SWI), Multiply instructions, Writing simple assembly language programs for ARM, 3-stage pipeline ARM organization, 5-stage pipeline ARM organization, Understanding of ARM instruction execution, Exceptions in ARM, Thumb programmer's model and instruction set	6	12
5	**I/O Devices of ARM processor** 5.1 General purpose I/O 5.2 Timers and counters 5.2.1.Watchdog timer 5.3 PWM device 5.4 Interrupt controllers 5.5 A/D and D/A converters 5.6 Serial communication devices	12	23
6	**Communication Bus protocols for embedded system** 6.1 **UART and SPI communication protocol** 6.2 **I2C communication protocol** I2C Protocol, Programming for I2C Protocol, Real time application using RTC, Advantages & Disadvantages of I2C Protocols. 6.3 **USB communication protocol** 6.4 **PCI-bus communication protocol** 6.5 **CAN protocol**	10	18
7	**Embedded C programming** 7.1 Overview of C compiler and optimization, Basic C data types, C Looping structures, Register allocations, function calls, pointer aliasing, structure arrangement, bit-fields, Division, floating point, Inline functions and inline assembly, Portability issues 7.2 "C" programs for General purpose I/O, general purpose timer, PWM Modulator, UART, I2C Interface, SPI Interface, ADC, DAC.	6	12

8	**RTOs base embedded system Design:-** 8.1 Introduction to Real-Time system, Definitions and examples, The Characteristics of RTOS 8.2 **Real-Time Operating Systems Architecture** RTOS Functionality, RTOS Characteristics, Operating system service, Process management, Timer and Event function, Memory management, Device , File and I/O subsystem management, 8.3 **Introducing to Tasks handling** □ Definition of Tasks □ Task Vs Thread □ Scheduling algorithm □ Context switching and latency □ Creating, controlling, deleting tasks □ Setting task priorities	12	23
	8.4 **Introducing ISR** □ ISR Overview □ RTOS Interrupt architecture □ ISR implementation , Interrupt routine in RTOS environment and handling of interrupt service calls □ ISR under RTOS , , Basic design using RTOS □ ISR to task communication, RTOS task scheduling models, Interrupt latency and response of tasks as performance metrics, OS security issue.		

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks					
R Level	U Level	A Level	N Level	E Level	C Level
7	**14**	**21**	**14**	**14**	**0**

Legends: R: Remembrance; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create and above Levels (Revised Bloom's Taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table

Reference Books:

1. ARM System on Chip Architecture by Steve Furber, Pearson Education
2. Embedded Systems -Raj Kamal, TMH.
3. Introduction to Embedded Systems -Shibu K.V, Mc Graw Hill.
4. Embedded System Design -Frank Vahid, Tony Givargis, John Wiley.
5. Embedded Systems –Lyla, Pearson, 2013
6. An Embedded Software Primer –Dav id E. Simon, Pearson Education.

Course Outcome: After learning the course the students should be able to:

1. evaluate embedded processor architecture and programming
2. interface with I/O devices to embedded processors
3. write program using either assembly level programming or C level language required for embedded system design
4. select advanced microprocessor, microcontroller and protocols for particular application

List of Experiments:

1. Write and simulate ARM assembly language programs for transfer data from one register to

another register

2. Write and simulate ARM assembly language programs for arithmetic and logical operations such as addition, subtraction multiplication, AND, XOR, OR…..

3. A) Take 3 numbers and perform multiply and accumulate.
   B) Take 2 numbers and perform addition of square of numbers

4. Write and simulate program in KEIL to perform following task.

   1. Addition of data in a single array
   2. Addition of 5 (32 bit) data stored in two array and store the result in third array
   3. Compare the data from the given array and store the smallest number in the register.

5. Write an assembly program to arrange given numbers in Ascending and Descending. Demonstrate original and arranged numbers in memory.

6. Write an assembly program to perform following string related operations.

   1. Convert the given string from lower to upper case.
   2. Convert the given string from upper to lower case.
   3. Convert given string into toggle case.

7. To interface LED with ARM microprocessor and Write a Program to blink LED's connected on P0.1 to P0.12 and reset them in reverse manner at the interval of 1 sec. Simulate program in simulator and observe output on board.

8. To interface switch with ARM microprocessor and write a program to read status of the switch.

   Example:- 4 switches (SW1 to SW4) are connected with ARM processor(LPC2148) board on pin P0.15, P0.13, P0.12 and P0.30 and 4 LEDs are connected on ARM processor board on pin P1.16. to P1.19. Write a program which will turn on LED1 if SW1 is pressed, turn on LED2 if SW2 is pressed, turn on LED3 if SW3 is pressed, turn on LED4 if SW4 is pressed. Turn off LEDs after some delay. Test program in Keil or any other software and observe output on kit.

9. To interface LCD with ARM microprocessor. Write and execute programs in C or assembly language for displaying text messages and numbers on LCD.

10. To write programs for serial communication between PC and ARM microprocessor.

11. To interfacing "LED touch screen display" with ARM processor .Also display key board on screen and take input from it.

12. Study Interfacing RTC with ARM processor.

13. To interface DC motor with ARM microprocessor. Write program to rotate DC motor in clockwise and anticlockwise direction with different speed.

14. To interface Stepper motor with ARM microprocessor. Write program to rotate motor in half step and full step mode.

15. Write a program to generate sinusoidal, triangular, saw tooth and square wave using DAC.

Design based Problems (DP)/Open Ended Problem:

- ➔ Interfacing of temperature sensor with ARM microprocessor board and display it on LCD screen
- ➔ Interface LED touch screen with ARM microprocessor and display keyboard on it also take input from onscreen keyboard
- ➔ Student mini project based on ARM microprocessor.

Major Equipment:

Computers, LPC2148 ARM processor board, KEIL software, PROTEUS software, etc.

List of Open Source Software/learning website:

http://nptel.ac.in/video.php

ACTIVE LEARNING ASSIGNMENTS: Preparation of power-point slides, which include videos, animations, pictures, graphics for better understanding theory and practical work – The faculty will allocate chapters/ parts of chapters to groups of students so that the entire syllabus to be covered. The power-point slides should be put up on the web-site of the College/ Institute, along with the names of the students of the group, the name of the faculty, Department and College on the first slide. The best three works should submit to GTU.