MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

Manipal School of Information Sciences (MSIS)

Manipal Academy of Higher Education, Manipal

Outcome Based Education (OBE) Framework

Two Year full time Postgraduate Program

Master of Engineering - ME (Embedded Systems)

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

TABLE OF CONTENTS

Contents

NATURE AND EXTENT OF THE PROGRAM 3

PROGRAM EDUCATION OBJECTICE (PEO) 4

GRADUATE ATTRIBUTES 5

QUALIFICATIONS DESCRIPTORS 8

PROGRAM OUTCOMES.....10

COURSE STRUCTURE, COURSEWISE LEARNING OBJECTIVE, AND COURSE OUTCOMES (COS) 13

NATURE AND EXTENT OF THE PROGRAM

An engineering graduate skillset requirement is changing with invent of the new technologies. In particular, the impact of Embedded Systems provides a high employability in the industry. Embedded Systems is a large scale implementation technology which is embodied in a wide spectrum of microcontrollers, processors, networked products, systems, and sensors, which take advantage of advancements in computing power, electronics miniaturization, and network interconnections to offer new capabilities which was not previously possible.

Master of Engineering - ME (Embedded Systems) Program is a comprehensive two-year postgraduate program, which aims to provide hands-on experience to prepare industry-ready Embedded Systems professionals. The program Master of Engineering - ME (Embedded Systems) helps engineering graduates to specialize in the field of Embedded Systems and enables them to learn how Embedded Systems devices can be programmed and developed for Varity of application domain. The Embedded Systems program balances between the treatment of high-level systems design and the engineering of subsystem components. One set of courses relate to the central ideas of systems development: Linux device drivers, computer architecture, micro-controllers, and the general ideas of embedded systems design. The other set of courses include the study of principles of real-time operating systems, design of data structures and algorithms, and digital signal processing.

Master of Engineering - ME (Embedded Systems) postgraduate degree would welcome graduates from any discipline with 50% mark in qualifying exam. Students after successfully completing the program will get career opportunities as a consumer electronics, telecommunications, automotive, aerospace and defence, industrial electronics, robotics.

PROGRAM EDUCATION OBJECTICE (PEO)

The overall objectives of the Learning Outcomes-based Curriculum Framework (LOCF) for **Master of Engineering - ME (Embedded Systems) program are as follows.**

PEO No	Education Objective
PEO 1	Enable to draw upon fundamental and advanced knowledge in order to apply analytical and computational approach to solve technological problems in embedded systems.
PEO 2	Introduce state of art technologies in the area of embedded system and inculcate ethical practices to make industry ready professional.
PEO 3	Promote scientific and societal advancement through research and entrepreneurship.

GRADUATE ATTRIBUTES

S No.	Attribute	Description
1	**Scholarship of Knowledge**	Acquire in-depth knowledge of specific discipline or professional area, including wider and global perspective, with an ability to discriminate, evaluate, analyse and synthesise existing and new knowledge, and integration of the same for enhancement of knowledge.
2	**Critical Thinking**	Analyse complex engineering problems critically, apply independent judgement for synthesising information to make intellectual and/or creative advances for conducting research in a wider theoretical, practical and policy context.
3	**Problem Solving**	Think laterally and originally, conceptualise and solve engineering problems, evaluate a wide range of potential solutions for those problems and arrive at feasible, optimal solutions after considering public health and safety, cultural, societal and environmental factors in the core areas of expertise.
4	**Research Skill**	Extract information pertinent to unfamiliar problems through literature survey and experiments, apply appropriate research methodologies, techniques and tools, design, conduct experiments, analyse and interpret data, demonstrate higher order skill and view things in a broader perspective, contribute individually/in group(s) to the development of scientific/technological knowledge in one or more domains of engineering.
5	**Usage of modern tools**	Create, select, learn and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, to complex engineering activities with an understanding of the limitations.

6	**Collaborative and Multidisciplinary work**	Possess knowledge and understanding of group dynamics, recognise opportunities and contribute positively to collaborative-multidisciplinary scientific research, demonstrate a capacity for self-management and teamwork, decision-making based on open-mindedness, objectivity and rational analysis in order to achieve common goals and further the learning of themselves as well as others.
7	**Project Management and Finance**	Demonstrate knowledge and understanding of engineering and management principles and apply the same to one's own work, as a member and leader in a team, manage projects efficiently in respective disciplines and multidisciplinary environments after consideration of economic and financial factors.
8	**Communication**	Communicate with the engineering community, and with society at large, regarding complex engineering activities confidently and effectively, such as, being able to comprehend and write effective reports and design documentation by adhering to appropriate standards, make effective presentations, and give and receive clear instructions.
9	**Life-long Learning**	Recognise the need for, and have the preparation and ability to engage in life-long learning independently, with a high level of enthusiasm and commitment to improve knowledge and competence continuously.
10	**Ethical Practices and Social Responsibility**	Acquire professional and intellectual integrity, professional code of conduct, ethics of research and scholarship, consideration of the impact of research outcomes on professional practices and an understanding of responsibility to contribute to the community for sustainable development of society.

11	**Independent and Reflective Learning**	Observe and examine critically the outcomes of one's actions and make corrective measures subsequently, and learn from mistakes without depending on external feedback.

QUALIFICATIONS DESCRIPTORS

1. Demonstrate

   (i) A systematic, extensive, coherent knowledge and understanding of an academic field of study as a whole and its applications, links to related disciplinary areas/subjects of study; including a critical understanding of the established theories, principles and concepts, and of a number of advanced and emerging issues in the field of Embedded System;

   (ii) Procedural knowledge that creates different types of professionals related to the Embedded System domain, including research and development, teaching, government and public service;

   (iii) Professional skills in the domain of microcontrollers, real time operating systems, device drivers, embedded system design, embedded systems, data structures, digital signal processing, internet of things, multicore programming optimization including a critical understanding of the latest developments, and an ability to use established techniques in the domain of digital media.

2. Demonstrate comprehensive knowledge about microcontrollers, interfacing I/O devices, sensors, actuators, communication devices, protocols, relating to essential and advanced learning areas pertaining to the embedded systems, techniques and skills required for identifying problems and issues related.

3. Demonstrate skills in identifying information needs, collection of relevant quantitative and/or qualitative data drawing on a wide range of sources, analysis and interpretation of data.

4. Methodologies as appropriate to the subject(s) for formulating evidence based solutions and arguments.

5. Use knowledge, understanding and skills for critical assessment of a wide range of ideas and complex problems and issues relating to the chosen field of study.

6. Communicate the results of studies undertaken in an academic field accurately in a range of different contexts using the main concepts, constructs and techniques of the Embedded system.

7. Address one's own learning needs relating to current and emerging areas of study, making use of research, development and professional materials as appropriate, including those related to new frontiers of knowledge.

8. Apply one's disciplinary knowledge and transferable skills to new/unfamiliar contexts and to identify and analyse problems and issues and seek solutions to real-life problems.

PROGRAM OUTCOMES

After successful completion of Master of Engineering - ME (Embedded Systems), Students will be able to:

PO No	Attribute	Competency
PO 1	Scholarship of Knowledge	Acquire in-depth knowledge of Embedded Systems domain, with an ability to discriminate, evaluate, analyze, synthesize the existing and new knowledge, and integration of the same for enhancement of knowledge.
PO 2	Critical Thinking	Analyze complex Embedded Systems Eco System critically, apply independent judgement for synthesizing information to make intellectual and/or creative advances for conducting research in a wider theoretical, practical and policy context.
PO 3	Problem Solving	Think laterally and originally, conceptualize and solve Embedded Systems problems, evaluate a wide range of potential solutions for those problems and arrive at feasible, optimal solutions after considering public health and safety, cultural, societal and environmental factors in the core areas of expertise.
PO 4	Research Skill	Extract information pertinent to unfamiliar problems through literature survey and experiments, apply appropriate research methodologies, techniques and tools, design, conduct experiments, analyze and interpret data, demonstrate higher order skill and view things in a broader perspective, contribute individually/in group(s) to the development of scientific/technological knowledge in one or more domains of engineering.
PO 5	Usage of modern tools	Create, select, learn and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modelling, to complex engineering activities with an understanding of the limitations.

PO 6	**Collaborative and Multidisciplinary work**	Possess knowledge and understanding of group dynamics, recognize opportunities and contribute positively to collaborative-multidisciplinary scientific research, demonstrate a capacity for self-management and teamwork, decision-making based on open-mindedness, objectivity and rational analysis in order to achieve common goals and further the learning of themselves as well as others.
PO 7	**Project Management and Finance**	Demonstrate knowledge and understanding of engineering and management principles and apply the same to one's own work, as a member and leader in a team, manage projects efficiently in respective disciplines and multidisciplinary environments after consideration of economic and financial factors
PO 8	**Communication**	Communicate with the engineering community, and with society at large, regarding complex engineering activities confidently and effectively, such as, being able to comprehend and write effective reports and design documentation by adhering to appropriate standards, make effective presentations, and give and receive clear instructions.
PO 9	**Life-long Learning**	Recognize the need for and have the preparation and ability to engage in life-long learning independently, with a high level of enthusiasm and commitment to improve knowledge and competence continuously.
PO 10	**Ethical Practices and Social Responsibility**	Acquire professional and intellectual integrity, professional code of conduct, ethics of research and scholarship, consideration of the impact of research outcomes on professional practices and an understanding of responsibility to contribute to the community for sustainable development of society.

PO 11	**Independent and Reflective Learning**	Observe and examine critically the outcomes of one's actions and make corrective measures subsequently and learn from mistakes without depending on external feedback.

COURSE STRUCTURE, COURSEWISE LEARNING OBJECTIVE, AND COURSE OUTCOMES (COS)

FIRST YEAR: ME (Embedded Systems)

Semester: 1 **Semester: 2**

Subject Code	Subject Title	L	T	P	C	Subject Code	Subject Title	L	T	P	C
CSE 601	Data Structures and Algorithms	3	-	-	3	ESD 603	Digital Signal Processing	3	-	-	3
CSE 602	Real Time Operating Systems	3	-	-	3	ESD 604	Device Drivers	3	-	-	3
ESD 601	Advanced Computer Architecture	3	-	-	3	ESD 605	Embedded Systems	3	-	-	3
ESD 602	Microcontrollers and its Applications	3	-	-	3	ESD 606	Embedded Software Design	3	-	-	3
	Elective - 1	3	-	-	3		Elective - 2	3	-	-	3
CSE 601L	Data Structures and Algorithms Lab	-	-	3	1	ESD 603L	Digital Signal Processing Lab	-	-	3	1
CSE 602L	Real Time Operating Systems Lab	-	-	3	1	ESD 604L	Device Drivers Lab	-	-	3	1
ESD 601L	Advanced Computer Architecture Lab	-	-	3	1	ESD 605L	Embedded Systems Lab	-	-	3	1
ESD 602L	Microcontrollers and its Applications Lab	-	-	3	1	ESD 606L	Embedded Software Design Lab	-	-	3	1
	Elective - 1 Lab	-	-	3	1		Elective - 2 Lab	-	-	3	1
ESD 695	Mini Project - 1	-	-	4	-	IOT 696	Mini Project -2	-	-	-	4
ESD 697	Seminar - 1	-	-	1	-	IOT 698	Seminar - 2	-	-	-	1
Total		**15**	**-**	**15**	**25**	**Total**		**15**	**-**	**15**	**25**

SECOND YEAR (FINAL YEAR):

III and IV Semester		
IOT 799	Project Work	25
Total Number of Credits to Award Degree		75

List of Electives(Theory)

Elective - 1		Elective - 2	
Code	**Subject**	**Code**	**Subject**
CSE-610	Computer Networks	CSE-605	Mobile Application Development using Android
CSE-604	Database Programming in Java	CSE-611	Web Application Development
IOT-607	Internet of Things	CSE-612	Multicore Program Optimization
		CSE-631	IT Project Management
		BDA-614	Big Data and Data Visualization
		EDA-601	High Level Digital Design
		ENP-601	Entrepreneurship

List of Electives (Lab)

Elective - 1		Elective - 2	
Code	**Subject**	**Code**	**Subject**
CSE-610L	Computer Networks Lab	CSE-605L	Mobile Application Development using Android Lab
CSE-604L	Database Programming in Java Lab	CSE-611L	Web Application Development Lab
IOT-607L	Internet of Things Lab	CSE-612L	Multicore Program Optimization Lab
		CSE-631L	IT Project Management Lab
		BDA-614L	Big Data and Data Visualization Lab
		EDA-601L	High Level Digital Design Lab
		ENP-601L	Entrepreneurship Lab

Name of the Institution / Department: Manipal School of Information Sciences (MSIS)

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Data Structures and Algorithms
Course Code: CSE 601	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Basic Programming – preferably C
Synopsis:	This Course provides insight on 1. This course introduces students to elementary data structures and design of algorithms. 2. Students learn how to design optimal algorithms with respect to time and space 3. Students learn how to implement link list, stack, queues, searching and sorting techniques, sets, trees and graphs. 4. Students learn the design of divide and conquer technique, dynamic programming, greedy technique and back tracking.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Specify and analyse algorithms.
CO 2:	Learn and design programs for implementation of linear and nonlinear data structure.
CO 3:	Learn and design programs for sorting and searching.
CO 4:	Illustrate application of divide and conquer technique, dynamic programming, greedy technique and back tracking.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*			*							
CO 2	*	*				*					
CO 3	*					*					
CO 4	*	*				*					

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction	
Algorithm Specification, Performance Analysis	At the end of the topic student should be able to: 1. Define algorithms (C1) 2. Analyse algorithms. (C6)
Unit 2: Algorithm Analysis Techniques	
Analysis of Recursive Programs, Solving Recurrence Equations, General Solution for a large class of Recurrences.	1. Define recursive programs (C2) 2. Design simple recursive programs (C6) 3. Solve recurrence relations (C6)
Unit 3: Elementary data structures	
Implementation of Lists, Stacks, Queues	1. Design singly linked list (C6) 2. Design doubly linked list(C6) 3. Explain the concepts of array-based stacks (C2) 4. Explain the concepts of pointer-based stacks (C2) 5. Design and implement Queues. (C6)
Unit 4: Sorting & Searching Techniques	
Quick sort, Heap sort, Merge sort, Binary search, linear search, Fibonacci search	1. Develop algorithm for insertion sort, bubble sort and selection sort. (C6) 2. Develop and analyse algorithm for quick sort (C6) 3. Develop and analyse algorithm for heap sort (C6) 4. Develop and analyse algorithm for merge sort (C6) 5. Design and analyse algorithms for binary, linear and Fibonacci search (C6)
Unit 5: Operations on Sets	

Introduction to Sets, A Linked- List implementation of Set, The Dictionary, The Hash Table Data Structure	1. Develop data structures for sets (C6) 2. Design a linked list-based implementation of sets (C6) 3. Design a Dictionary (C6) 4. Design Data structure for hash table (C6)
Unit 6: Trees	
Basic Terminology, Implementation of Trees, Binary Trees, Binary Search Trees	1. Examine the concepts of trees. (C3) 2. Design and implement general trees (C6) 3. Design and implement binary trees (C6) 4. Design and implement binary search trees (C6)
Unit 7: Graphs	
Basic definitions, Representation of Graphs, Minimum Cost Spanning Tree, Single Source Shortest Paths, All-Pairs Shortest Path	1. Define graphs (c6) 2. Design data structure for graphs (c6) 3. Formulate an algorithm to solve minimum cost spanning tree(c6) 4. Formulate an algorithm to solve Single source shortest path (c6) 5. Formulate an algorithm to solve All- pair shortest path(c6)
Unit 8: Algorithm Design Techniques	
Divide-and-Conquer Algorithms, Dynamic Programming, Greedy Algorithms, Backtracking	1. Design of divide and conquer algorithms (C6) 2. Solve max min, Strassen's matrix multiplication, multiplication of long integers problem. (C6) 3. Design of dynamic programming techniques (C6) 4. Solve matrix chain order problem (C6) 5. Design of greedy algorithms(C6)

	6. Solve Knap-sack, job scheduling with deadlines and optimal storage on tapes problems. (C6) 7. Design of Back tracking algorithms (C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Seminar	-	-
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Clinic	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*		
Sessional Examination 2		*	*	*
Assignment/Presentation	*	*	*	*

End Semester Examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• "Introduction to Algorithms" Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest. • "Data Structures& Algorithms" Aho, Hopcroft and Ulmann • "Data structures and algorithm analysis in C" Mark Allen Weiss • "Computer Algorithms" : Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Real Time Operating Systems
Course Code: CSE 601	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Basic Programming – preferably C

Synopsis:	This Course provides insight on 1. This course introduces students to basics of operating systems and real operating systems. 2. This course helps the student to understand the concepts of process management, scheduling, synthetization and dead locks. 3. This course helps the students to learn thread-based programming. 4. Students learn the concept of memory management. 5. Students learn the salient features of real time operating systems
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Examine the evolution of operating systems and real time operating systems.
CO 2:	Design programs based on threads.
CO 3:	Explain the concepts involved in process management, scheduling, synthetization of processes.
CO 4:	Explain the concepts involved in memory management, detecting, avoiding and recover from dead locks.
CO 5:	Explain the concepts of real time systems and real time operating systems

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*								
CO 2	*	*	*								
CO 3	*		*								
CO 4	*		*								
CO 5		*	*	*							

Course content and outcomes:

Content	*Competencies*
Unit 1: Introduction to OS and RTOS	
Essential features of an OS, Single Processor Systems and Multiprocessor Systems, Essential Features of Batch Processing, Time sharing, Multiprogramming, Interactive systems, User mode and Kernel Mode operations, Distinction between function call and system call, Real time operating system and real time embedded systems.	At the end of the topic student should be able to: 1. Identify the features of OS and RTOS (C2) 2. Distinguish between single processor and multi-processor systems (C2) 3. Identify the features of batch processing, time sharing, multi programming and interactive systems (C2) 4. Distinguish between user and kernel modes (C2) 5. Distinguish between function and system calls (C2)
Unit 2: Process Management	
A process in memory, process state, PCB, Process scheduling, scheduling Queues, Types of schedulers, Process system calls - IPC using Shared Memory, IPC using Sockets.	1. Describe a process, process state, process control block (C2) 2. Illustrate scheduling algorithms, scheduling queues (C3) 3. Examine process related system calls (C1) 4. Illustrate methods for inter process communication through share memory and sockets (C3)
Unit 3: Multithreaded Programming	
Introduction, benefits, multithreading models, Pthreads, Win32 threads, Threading Issues, Thread pools Linux threads.	1. Summarize the benefits of multi-threading (C2) 2. Discover threading issues (C2) 3. Illustrate programs using p threads (C3) 4. Examine the benefits of thread pools (C3)
Unit 4: Process Scheduling	
Introduction, scheduling criteria, scheduling Algorithms – FCFS, SJF, PS, RR, Multilevel Queues, Multilevel	1. Distinguish between scheduling algorithms (C2) 2. Examine the criteria for scheduling (C3)

feedback Queue Scheduling, Scheduling evaluations.	3. Explain FCFS, SJF, PS, RR, Multi-level queues, multi-level feedback queues scheduling algorithms (C2) 4. Evaluate the scheduling algorithms (C5)
Unit 5: Synchronization	
Introduction, Critical Section Problem, Petersons Solutions, synchronization hardware, Semaphores, usage, implementations; Deadlocks and starvation, Classical problem of synchronization – Bounded Buffer problem, Reader's Writer's problem, Dining Philosophers problem, sleeping barber's problem; Monitors.	1. Define critical section problem (C1) 2. Demonstrate Software solutions to critical section problems (C3) 3. Demonstrate hardware solution for process synchronization (C3) 4. Describe the usage and implementation of semaphores (C1) 5. Define dead locks and starvation (C1) 6. Illustrate solutions to classical synchronization problems like bounded buffer, readers writers, dining philosophers and sleeping barbers (C3)
Unit 6: Deadlocks	
Introduction, deadlock, characterization, methods for handling deadlocks, deadlock prevention, deadlock avoidance, recovery from deadlock.	1. Define dead locks (C2) 2. Examine methods for handling dead locks (C4) 3. Illustrate various dead lock algorithms (C3)
Unit 7: Memory Management	
Memory Management Strategies, Virtual Memory Management.	1. Examine various memory management strategies(C4) 2. Examine the evolution of memory management (C4) 3. Illustrate the benefits of paging and segmentation(C3) 4. Examine the implementation of demand paging(C4) 5. Examine the various virtual memory concepts (C4)
Unit 8: Real Time Systems	
Overview of Real Time Systems, Real Time clocks and Real Time Scheduling Algorithms	1. Examine the concepts involved in the design of real time systems (C3) 2. Design of real time clocks in various real time languages(C5)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Sessional Examination 1	*	*			
Sessional Examination 2			*	*	
Assignment/Presentation				*	*
End Semester Examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• "Operating System principles", Seventh Edition, Abraham Silberschatz, Peter Galvvin, Grag Gagne. John Wiley Publications • "Real – Time Systems and Programming Languages", Allan Burns, Andy Wellings. • "Operating Stems Concepts and Design", Milan Milenkovic • "Design of Unix Operating System", Maurice Bach (IPC) • "The C Programming Language", Kerninghan & Ritchie

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Advanced Computer Architecture
Course Code: ESD 601	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Basic Programming – preferably C, digital systems

Synopsis:	This Course provides insight on 1. Basics difference between computer architecture and organisation, recoise buses, defining the architecture 2. Able to understand various architectural instruction sets, defining the registers, deign ALU, adders, multipliers, booths algorithms and division algorithms, arm processors 3. Designing of instruction sets, design of control units 6. Students able to develop the execution unit based on applications
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Distinguish between computer architecture and organisation, buses , memory, various types of instruction execution
CO 2:	Analyse register ALU design, booths & division algorithms, the problem definition, complete design and development of application on ARM
CO 3:	Justify the processing section of control unit and PLA

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*									
CO 2		*	*								
CO 3				*	*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction	
Computer Architecture Vs Organization – Classification -Von Neumann vs	At the end of the topic student should be able to:

Harvard Architecture – Computer Architecture Vs Embedded Architecture - System on Chip vs System on Board – CPU – Address Bus – Data Bus – Control Bus –RAM – ROM – Instruction Set Architecture - Classifications – RISC vs CISC vs VLIW.	1. List the difference between computer architecture and organisation(C1) 2. Describe data bus, control bus, address bus w.r.t. to technical aspects(C2) 3. Distinguish between RAM and ROM (C2) 4. Distinguish between RISC and CISC (C2) 5. Describe the programming model of ARM (C2)
Unit 2: Introduction to ARM Processor	
Architecture - Components - Instruction level registers and Purpose– Special Function Registers Vs General Purpose Registers – Register Design - ALU– ALU design – Adder – Types of Adders – Logical Block Design – Multiplier – Multiplier Design - Barrel Shifter – Design of Barrel Shifter – Sequential Multiplication Algorithm - Booths Algorithm – Division Algorithm (Restoring and Non-Restoring).	1. Write asm program to display "HELLO WORLD" using ARM instructions(C5) 2. Design a 4bit GPR for following actions: load external data, rotate left, rotate right, increment. (C5) 3. Compute and Analyse booths algorithm for two four bit numbers 4x-6 (C4) 4. Explain the restore algorithm by taking an example 11 by 3 (C5) 5. In a computer instruction format the instruction length and size of the address filed are 11 and 4 bits. Architecture already had 6 two address and 24 zero address instruction. What is the maximum number of 1 address instruction that can be added to instruction set. Justify your answer (C6)
Unit 3: Instruction Set Architecture	
ARM & Thumb Instructions – Addressing Modes – Types of Instructions – Endianness – Assembly Programming - Instruction Designing- Huffman Encoding technique for designing instruction sets – Control Unit – Hardwired and Microprogrammed Approach - Firmware –Coprocessor – Floating Point Number – FPU	1. Explain thumb instruction programming model in ARM7 (C2) 2. Define huffmans encoding technique with adequate example(C3) 3. Considering the relative frequency's for set of instructions, by encoding them using huffman's method, calculate the redundancy(C5) 4. Design a Booths multiplier to multiply 4 bit 2's compliment numbers using D Flip-flops and PLA for booths multiplier (C6)

Unit 4: Memory	
Types of Memory – Memory Hierarchy - Static – Dynamic RAM – ROM & ROM Types – Cache Memories –Performance Considerations - Virtual Memories - MMU & MPU - Secondary Storage	1. Define the characteristic of memory's (C2) 2. Write the block diagram of 1Kx8 RAM using two 1Kx4 chips (C5) 3. Explain synchronous and asynchronous bus with timing diagram. (C4)
Unit 5: Introduction to Pipelining	
Advantage – Data Hazards – Instruction Hazzards – Influence of Instructions sets – Datapath and Control Considerations	1. What is meant by hazards? Explain briefly(C4) 2. Describe how data hazard can be minimized by complier scheduling(C4) 3. List and explain 4 schemes which helps to reduce branch hazards(C5) 5. Explain how delayed branch can be scheduled (C5)
Unit 6: Introduction to Parallel Processing	
Parallelism in Uniprocessor & Multicore Systems – Parallel Computer Structures – Architectural Classification Schemes – Applications - Principles and Vector Processing – Structures and Algorithms for Array Processors	1. Describe the DLX instructions, their formats with help of example(C3) 2. Explain flynn's classification of computers(C4) 3. Differentiate between shared memory versus distributed memory(C3) 4. Differentiate between parallel computing versus serial computing (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04

Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Sessional Examination 1	*	*			
Sessional Examination 2			*		*
Assignment/Presentation		*	*		*
End Semester Examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. CV Hamacher, Vranseic and Zaky , "Computer Organization", Fifth Edition, Tata-MacgrawHill 2. Rafiquzzamann ,"Modern Computer Architecture",Chandra,Galgotia Publications 3. John L Hennessy and David A Patterson ,"Computer Architecture: A Quantitative approach", 2nd Edition 4. John L Hennessy and David A Patterson ,"Principles of Computer Architecture", Prentice Hall 5. Shivarama Danadamudi, "Guide to RISC Processors for Programmers & Engineers", Springer Publications. 6. "ARM Architecture Reference Manual", David Seal ,Addison-Wesley,2nd Edition 7. "AMBA Specification", ARM7TDMI Datasheet. 8. "Computer Organisation and Design", David A Patterson, John L Hennessy 9. David Seal, "ARM Architecture Reference Manual", 2nd Edition, Addison-Wesley Professional.

	10. Steve Furber,"ARM System-on-Chip Architecture",2nd Edition, Addison-Wesley Professional, ISBN-13: 078-5342675191,ISBN-10: 0201675196 11. William Hohl, Christopher Hinds,"ARM Assembly Language: Fundamentals and Techniques",2nd Edition, ISBN-13: 978-1482229851, ISBN-10: 1482229854 12. Andrew Sloss, Dominic Symes, Chris Wright,"ARM System Developer's Guide: Designing and Optimizing System Software",1st Edition,The Morgan Kaufmann Series in Computer Architecture and Design, ISBN-13: 978-1558608740, ISBN-10: 1558608745 1. 13. Websites & Transaction Papers

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Microcontrollers and its Applications
Course Code: ESD 602	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Microprocessor architecture , Assembly language and Number systems

Synopsis:	This Course provides insight on 1. This course provides the knowledge of Intel 8051 and ARM Microcontrollers. 2. This course provides the knowledge of Microcontroller architecture, Registers and Instruction sets to write Assembly and Embedded C Programming. 3. This course provides the concept of Interfacing and Programming Sensors and Peripherals to Microcontrollers. 4. This course provides the concept of Designing Embedded Systems using Microcontrollers.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Employ the knowledge of Microcontrollers to build Embedded systems.
CO 2:	Explain the concept of Programming Microcontrollers using Assembly and Embedded C.
CO 3:	Design Embedded Systems by interfacing Sensors and Actuators.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*										
CO 2	*	*			*						
CO 3	*		*		*						

Course content and outcomes:

Content	***Competencies***
Unit 1: Introduction to Microprocessor & Microcontrollers	

Comparison – Variants – Types – General – ASIC – PLD – Introduction to Motherboard(Desktop) - Introduction to Embedded Board – Compare and Contrast - Application Types – Single Tasking – Multitasking – Multi-Application	At the end of the topic student should be able to: 1. Explain about the differences of Microprocessor and Microcontrollers(C2) 2. Describe Microcontroller Architecture (C2) 3. Explain the Register sets, Programming model and Memory map of Microcontroller(C2) 4. Describe about Microcontroller Instruction set. (C2) 5. Write the Applications using Microcontrollers. (C3)
Unit 2: Introduction to ARM Microcontrollers	
Programming Model – Processor Modes – ARM vs Thumb Introduction to LPCxxxx Microcontrollers – Features – Detailing of Pins - Memory Map Concepts – RAM & ROM - Interrupts Concepts (Internal & External)	1. Describe ARM Microcontroller architecture. (C2) 2. Describe the architecture of ARM Microcontrollers. (C2) 3. Apply knowledge of ARM Microcontroller architecture to rig up Embedded system circuits(C3) 4. Develop a Prototype of Embedded systems using ARM Microcontroller(C5, P3)
Unit 3: Reset Circuitry	
Crystals - Introduction to GPIO – Registers – Input /Output Configuration – Pull Up and Pull Down Resistor Concept – Interfacing with LED – Interfacing Push Buttons – LCD – Stepper Motor – DC Motor	1. Describe Crystal oscillator. (C2) 2. Describe Pull Up and Pull Down Resistor Concept. (C2) 3. Illustrate Interfacing LED, Push Buttons, LCD, Stepper Motor – DC Motor with microcontroller. (C2)
Unit 4: Relays	
Types of Relays – Interfacing	Describe Relay and its with interfacing external peripherals to Microcontrollers. (C4)
Unit 5: Timer, Counter Introduction	

Configuration – Programming	1. Describe about timers, counters and its usage with Microcontrollers(C4)
Unit 6: Serial vs Parallel Bus	
Serial vs Parallel Bus - Compare and Contrast – Terminology: Baud Rate – Bit Rate – RS232 – DB9 handshaking concepts - Configuring Registers – Programming for UART modules.	1. Describe about Serial and Parallel communication protocols(C2)
Unit 7: Introduction to SPI and I2C Protocol	
Detailed Discussion – Bit Banging – Interfacing with SPI and I2C Devices – RTC / ADC /DAC.	1. Describe SPI, I2C standards and its Interfacing with SPI and I2C Devices – RTC / ADC /DAC. (C3) 2. Explain about how to establish multi controller communications using communication protocols (C3)
Unit 8: Introduction to ADC and DAC	
Types – Chips - Register Configuration – Interfacing	1. Summarize types of ADC, DAC and its usage with Microcontroller. (C2)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2		*	*
Assignment/Presentation	*		*
End Semester Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. William Hohl, Christopher Hinds,"ARM Assembly Language: Fundamentals and Techniques",2nd Edition, ISBN-13: 978-1482229851, ISBN-10: 1482229854 2. Andrew Sloss, Dominic Symes, Chris Wright,"ARM System Developer's Guide: Designing and Optimizing System Software",1st Edition,The Morgan Kaufmann Series in Computer Architecture and Design, ISBN-13: 978-1558608740, ISBN-10: 1558608745 3. David Seal, "ARM Architecture Reference Manual", 2nd Edition, Addison-Wesley Professional. 4. Steve Furber,"ARM System-on-Chip Architecture",2nd Edition,Addison-Wesley Professional, ISBN-13: 078-5342675191,ISBN-10: 0201675196 5. Douglas V. Hall,"Microprocessors and Interfacing",Mcgraw Hill Educatin ,ISBN-10 1259006158,ISBN-13 9781259006159,2012. 6. Websites & Transaction Papers

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Computer Networks
Course Code: CSE 610	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Basic of Computer Communication and Networks
Synopsis:	This Course provides insight on 1. The concepts, applications, reference models of computer networks. 2. The functional knowledge about communication devices, IP addresses and routing algorithms for networking. 3. The implementation of routing, congestion, transport, multitasking and application layer protocols for the wired using simulation and analysis.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Identify the goals and applications of computer networks, able to explain the classification of networks and reference models.
CO 2:	Describe the functions of communication devices, IP addressing techniques.
CO 3:	Demonstrate routing algorithms, congestion control mechanisms and transport layer protocols.
CO 4:	Examine application, multicasting and management protocols functions.

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*										
CO 2		*									
CO 3			*								
CO 4			*								

Course content and outcomes:	
Content	***Competencies***
Unit 1: Computer Networks	
Definition, Network goals, Classification of networks, ISO – OSI and TCP/IP reference model	At the end of the topic student should be able to: 1. Define Computer Networks (C1). 2. Describe any four applications of computer networks (C2).

	3. Write the classifications of computer networks (C3). 4. Explain the basic network topologies with suitable failure scenarios (C2). 5. Describe the responsibilities of each layer in a reference model (C2).
Unit 2: Communication Devices	
Network Interface card, modem, hub, switch, repeater, bridge, router and gateway.	1. Describe various network devices operating across various layers of TCP.IP Stack. (C2)
Unit 3: Internet addresses	
Classes of IP addresses, Subnetting and Supernetting TCP/IP networks, Internet routing, host name resolution, Mapping IP addresses to physical addresses, ARP and RARP	1. Describe on various routing and communication techniques with suitable diagram (C2). 2. Design different classes of networks with address ranges (C5). 3. Write the purpose of various protocols at Internet layer (C3) 4. Design of subnetworks and super networks (C5).
Unit 4: Routing Algorithms	
Shortest path, Flow based, Distance vector, Link state, Hierarchical	1. Explain the roles of various routing algorithm in finding shortest routes (C5) 2. Modify the routing table of a router in the given subnet (C3) 3. Show how hierarchical routing reduce the routing table contents (C3)
Unit 5: Congestion control algorithms	
Congestion – causes, Congestion Prevention Techniques, Congestion Reactive Techniques	1. Describe various congestion prevention and reactive techniques used in the routers and hosts (C2) 2. Write the various causes of congestion in a network (C3)
Unit 6: Internet Transport protocols (TCP & UDP)	
Connection establishment and termination, Flow Control, Time out and retransmission process	1. Write the roles of different port numbers used for communication purposes (C2) 2. Explain the operations of various protocols at transport layer (C2).

	3. Describe flow control and congestion control mechanisms acting at transport layer. (C4).
Unit 7: Application Protocols	
SMTP, DHCP, DNS, FTP	1. Describe the roles of various protocols at application layer (C2) 2. Compare the connection establishment, information transfer and connection termination phases of different application layer protocols (C4). 3. Show the configuration, implementation and usage of various the application layer protocols. (C3).
Unit 8: Multicasting and other Protocols	
Multicasting, IPV6, IGMP, ICMP, VOIP	1. Explain multicasting protocol. (C2) 2. Write the functions of IP in voice communication (C3) 3. Demonstrate various network management protocol. (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO5
Sessional Examination 1	*	*	*		*
Sessional Examination 2		*	*	*	*
Assignment/Presentation			*	*	*
End Semester Examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. "Internetworking with TCP/IP Vol I : Principles, Protocols and Architecture", Douglas E Comer, III Ed. PHI, 1997. 2. "Microsoft TCP/IP on Windows NT 4.0", MCSE. 3. "Internetworking with TCP/IP Client–Server Programming and applications, BSD Socket version", Douglas E Comer and David L Stevens, Vol. III. 4. "TCP/IP Illustrated, Volume I, The Protocols", W Richard Stevens, International Student Edition, 1999. 5. "Advanced Internet Technologies", Uyless Black, PHI 6. "High Performance Communication Networks", Jean Warland & Praveen Varaiya – Morgan Kaufmann

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Database Programming in Java
Course Code: CSE 604	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Basic programming knowledge
Synopsis:	1. To provide fundamental knowledge of various object oriented programming concepts and database concepts. 2. To design and develop database applications using java programming language.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Explain major principles of object oriented programming concepts
CO 2:	Discuss the different elements of java programming language
CO 3:	Design databases using the conceptual model
CO 4:	Develop a java application for various database requirements

Mapping of COs to POs

Cos	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*									
CO 2	*	*	*		*						
CO 3		*		*							
CO 4			*		*						

Course content and outcomes:	
Content	*Competencies*
Unit 1: Introduction	
Object Orientation (OO) Concepts	At the end of the topic student should be able to: 1. Illustrate with an example the major principles such as classes, objects, encapsulation, inheritance, polymorphism (C2) 2. Distinguish between procedure oriented programming and object oriented programming (C2)
Unit 2: Introduction to Java	
Data types, Operators, Control Statements.	1. Discuss features of java programming language (C2)

	2. Discuss the term platform independence specific to java programming language (C2) 3. Explain various data types, operators and control statements (C2)
Unit 3: Classes in java	
Class fundamentals, Constructs, Garbage collection, Inner Classes	1. Define class structure in java programming language (C1) 2. Discuss various components of class structure which includes concepts constructors, variables, methods using java (C2) 3. Explain the mechanism garbage collection (C2) 4. Illustrate the use of inner classes (C2)
Unit 4: Inheritance	
Introduction to Java Inheritance, Multilevel inheritance, Abstract, final classes	1. Define different types of inheritance (C1) 2. Explain abstract classes (C2) 3. Discuss final classes (C2) 4. Apply abstract classes and final classes in applications (C3)
Unit 5: Packages, Interfaces	
Package, access control, Interfaces.	1. Illustrate the use of packages in an application (C2) 2. List various access control mechanism (C1) 3. Define java interfaces (C1) 4. Apply interfaces in applications. (C3)
Unit 6: I/O API's	
Reader, Writer APIs, File Management	1. List the types of steam classes available (C1) 2. Write java program to read data from different types of files (C3) 3. Discuss file management in java (C2)
Unit 7: Exception Handling	
Using exception handling, Creating user defined exceptions.	1. Discuss the types of exception handle (C2) 2. Explain user define exception class (C2)
Unit 8: Java Applets, Applications	
Java Applets, life cycle, methods, java Application	1. Define java applets (C1) 2. Discuss life cycle of java applets (C2)

	3. Distinguish between java applets and java applications (C2)
Unit 9: Introduction to Swing	
Swing components, Event handling, layout managers	1. Distinguish between AWT components and swing components (C2) 2. Define features of swing components (C1) 3. Apply different swing components, layout managers in java applications (C3) 4. Discuss event delegation model (C2)
Unit 10: Introduction to Database concepts	
Primary goal of RDBMS, Purpose of Database System, Characteristics of the Database Approach, Actors on the Scene, Workers behind the scene, Advantages of Using a DBMS, Views of Data	1. Define Relational database management (C1) 2. Discuss the purpose of database system (C2) 3. Explain characteristics of the database approach (C2) 4. List actors on the scene and workers behind the scene (C1) 5. Discuss advantages of using DBMS (C2)
Unit 10: SQL	
Basic Structure, Set Operations, Aggregate Functions, Null Values, Nested Subqueries, Derived Relations, Views, Modification of the Database, Joined Relations, Data-Definition Language	1. Explain basic structure of SQL statement (C2) 2. Discuss set operations (C2) 3. Explain different types of aggregate functions (C2) 4. Explain Views, nested queries, joined relations (C2) 5. Discuss data definition language (C2)
Unit 10: Introduction to JDBC	
JDBC Architecture, Connecting to an ODBC Data Source, JDBC Connection, JDBC Implementation, Resultset Processing, Prepared statement, Other JDBC Classes, Moving the cursor in scrollable Result Sets, Making updates to Updatable Result Sets.	1. Explain JDBC architecture (C2) 2. Explain JDBC connection and its implementation (C2) 3. Explain different types of jdbc classes which are required for database applications (C2)

Learning strategies, contact hours and student learning time

Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*		
Sessional Examination 2			*	*
Assignment/Presentation	*	*	*	*
End Semester Examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• *Patrick Naughton and Herbert Schildt - "JAVA 2 - The Complete Reference" VII Edition, Tata McGraw Hill.* • *George Reese - "Database Programming with JDBC and Java", O'Reilly* • *"Database system Concepts, Third Edition",Author: Abraham Silberschatz (Bell Laboratories), Henry F. Korth(Bell Laboratories) and S. Sudarshan (Indian Institute of Technology, Bombay, Publishers: The McGraw-Hill Companies, Inc.* • *"Fundamentals of Database systems, Third Edition". Author: Elmasri and Navath*

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Internet of Things
Course Code: IOT 607	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Computer Networks, Programming aspects.
Synopsis:	This Course provides insight on 1. Various elements involved in the development of application for IoT. 2. Understanding of protocols across IoT stack 3. Scripting languages like shell and python. 4. Client Server architecture and Python APIs of Socket programming. 5. Database and Python Database connectivity, Python Web Programming, IoT Framework
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Describe the developmental aspects of the application in IoT.
CO 2:	Demonstrate the usage of networking protocols across IoT stack.
CO 3:	Demonstrate the fundamental concepts in Client Server architecture and database implementation and usage with Python API's.

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*										
CO 2	*	*			*						
CO 3	*		*		*						

Course content and outcomes:	
Content	*Competencies*
Unit 1: Internet of Things	
IoT Protocols – Logical Design - Enabling Technologies - Levels – IoT vs M2M – Design Methodology – Domain Specific Applications	At the end of the topic student should be able to: 1. Outline the integration of various elements of IoT ecosystem. (C2)
Unit 2: Introduction to Python	

Datatypes - Constructs – Packages	1. Employ Datatypes, Constructs, Packages in python programming. (C2)
Unit 3: Wireless Sensor Networks	
Protocol Standards – Issues – Routing – Applications	1. Describe Protocol Standards, Routing, Issues in Wireless Sensor Networks. (C2)
Unit 4: Bluetooth	
Introduction – Protocol Stack - RF Classes – Radio Technologies – Service Discovery – Device Discovery – Profiles – Security (Discovering Bluetooth) - Hardware	1. Explain the aspects of Bluetooth technology. (C2)
Unit 5: Zigbee	
- Frequency - Channels – Topology - Zigbee Protocol Stack - PHY - MAC Layer - Working – Frame Structure – Beacon – Non-Beacon Communication - Zigbee PDU – Zigbee Hardware – API Mode and AT mode communication.	1. Describe Protocol Standards, Routing, Issues in Zigbee. (C2)
Unit 6: Internet Protocol	
Introduction to IPv4 and IPv6 – IPv4 Headers – Ipv6 Headers	1. Demonstrate the implementation of IPv4 and IPv6 protocol in TCP/IP protocol stack. (C3)
Unit 7: 6LoWPAN - 6LoWPAN architecture	
simple, extended and ad-hoc networks. Issues in determining IPv6 links in LLNs and illustration of the undetermined link addressing model. IPv6 addressing in 6LoWPAN.	1. Indicate the 6LoWPAN architecture for resource constrained devices. (C2)
Unit 8: Sockets	
Introduction to Sockets – Client Server Architecture –Unix Sockets – PORTS – Python APIs of Sockets – TCP socket programming using Python – UDP – RAW packets python programming.	1. Outline Client Server Architecture. (C1)
Unit 9: Databases & Web Programming	
Introduction to Databases – File System vs RDBMS – ER Diagram – Python Database connectivity (CRUD) - Web Server Concepts - Python Web Programming – IoT Framework.	1. Illustrate the socket communication using python API's for RWA, stream and datagram-oriented use cases. (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2		*	*
Assignment/Presentation		*	*
End Semester Examination	*	*	*

Feedback Process	• End-Semester Feedback
1. Reference Material	• Arshdeep Bhaga, Vijay Madishetti, “Internet of things:A hands on Approach”, Universities Press, ISBN:978172719547 • Robert Faludi,”Building Wireless Sensor Networks”,Orielly, 2012 • Jean-Philippe Vasseur,Adam Dunkels,"Interconnecting Smart Objects with IP: The Next Internet",Morgan Kaufmann Publishers,2010,ISBN:0123751659 9780123751652 • Marco Schwartz,"Internet of Things with the Arduino Yun",Packt Publishing,2014

	• Charalampos Doukas,"Building Internet of Things With the Arduino: Volume 1",CreateSpace Independent Publishing Platform,2012 • Todor Cooklev ,“Wireless communication standards”, IEEE Press • Houda Labiod, Hossam Afifi, Costantino De Santis, “Wi-Fi, Bluetooth, Zigbee and WiMAX”, Springer Publications • Madhushree Ganguli , “Getting started with Bluetooth”, Premier Press, 2002, ISBN 1931841837, 9781931841832.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Data Structures and Algorithms Lab
Course Code: CSE 601L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Basic Programming – preferably C
Synopsis:	This Course provides insight on 1. This course introduces students to elementary data structures and design of algorithms. 2. Students learn how to design optimal algorithms with respect to time and space 3. Students learn how to implement link list, stack, queues, searching and sorting techniques, sets, trees and graphs. 4. Students learn the design of divide and conquer technique, dynamic programming, greedy technique and back tracking
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Specify and analyse algorithms
CO 2:	Learn and design programs for implementation of linear and non-linear data structure.
CO 3:	Learn and design programs for sorting and searching.
CO 4:	Illustrate application of divide and conquer technique, dynamic programming, greedy technique and back tracking.
CO 5:	Learn to organise the code for scalability and maintainability.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1		*									
CO 2		*	*		*			*			
CO 3		*	*		*			*			
CO 4		*	*		*			*			
CO 5		*	*		*			*			

Course content and outcomes:	
Content	***Competencies***
Unit 1: Elementary data structures	

Implementation of Lists, Stacks, Queues	At the end of the topic student should be able to: 1. Design and Implement singly linked list 2. Design and Implement doubly linked list 3. Design and Implement array-based stack 4. Design and Implement pointer-based stack 5. Design and Implement array-based queues. 6. Design and Implement pointer-based queues.
Unit 2: Sorting & Searching Techniques	
Quick sort, Heap sort, Merge sort, Binary search, linear search, Fibonacci search	1. Design and implement programs for insertion sort, bubble sort and selection sort. 2. Design and implement programs for quick sort 3. Design and implement programs for heap sort 4. Design and implement programs for merge sort 5. Design and implement programs for binary, linear and Fibonacci search
Unit 3: Trees	
Basic Terminology, Implementation of Trees, Binary Trees, Binary Search Trees	1. Write a program to implement binary trees 2. Write a program to implement binary search trees 3. Tree traversal technique
Unit 4: Graphs	
Basic definitions, Representation of Graphs, Minimum Cost Spanning Tree,	1. Write programs to represent a graph using adjacency matrix and adjacency list techniques

Single Source Shortest Paths, All-Pairs Shortest Path	2. Write a program to implement minimum cost spanning tree 3. Write a program to solve Single source shortest path problem 4. Write a program to solve All- pair shortest path problem
Unit 5: Algorithm Design Techniques	
Divide-and-Conquer Algorithms, Dynamic Programming, Greedy Algorithms, Backtracking	1. Write a program to solve max min problem 2. Write a program to solve Strassen's matrix multiplication problem 3. Write a program to solve matrix chain order problem 4. Write programs to solve knap-sack, job scheduling with dead line and optima storage on taps problems. 5. Write programs to solve n queens and graph colouring problems

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination

	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Sessional Examination 1	*	*			
Sessional Examination 2		*	*	*	
Assignment/Presentation	*	*	*	*	*
Laboratory Examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. "Introduction to Algorithms" Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest. 2. "Data Structures& Algorithms" Aho, Hopcroft and Ulmann 3. "Data structures and algorithm analysis in C" Mark Allen Weiss 4. "Computer Algorithms" : Ellis Horowitz, SartajSahni, SanguthevarRajasekaran

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Real Time Operating Systems Lab
Course Code: CSE 602L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Knowledge on C programming, Operating System concepts
Synopsis:	This Course provides insight on 1. Basics of operating systems and real operating systems. 2. Understand the concepts of process management, scheduling, synthetization and dead lock. 3. Learn thread-based programming. 4. Learn the concept of memory management. 5. Learn the salient features of real time operating systems
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Experiment process creation, process hierarchies and multi-thread concepts.
CO 2:	Apply process-scheduling algorithms and process synchronization concepts on various scenarios.
CO 3:	Apply memory management techniques on various scenarios

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*		*						
CO 2	*	*	*		*						
CO 3	*	*	*		*						

Course content and outcomes:

Content	***Competencies***
Unit 1:	
Basics of C programming: String manipulation, file handling.	At the end of the topic student should be able to: Practice basic C programming concepts (C3)

Unit 2:	
Process creation, fork, exec, wait, multi thread concepts.	Experiment process creation, process hierarchies and multi-thread concepts. (C4)
Unit 3:	
Process scheduling algorithms	Apply process-scheduling algorithms on various scenarios. (C3)
Unit 4:	
Process synchronization concepts.	Experiment process synchronization concepts (C4)
Unit 5:	
Memory management techniques	Apply memory management techniques on various scenarios (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Assignment/Presentation			*

Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• Text mining handbook: advanced approaches in analyzing unstructured data Feldman, Ronen and James Sanger, 9780521836579, CUP, 2008 • Linked Lexical Knowledge Bases Iryna Gurevych, Judith Eckle-Kohler, Michael Matuschek,9781627059749, Morgan & Claypool, 2016 • Introduction to information retrieval Manning, Christopher D. and Prabhakar Raghavan and Hinrich Schutze, 9780521865715, Cambridge University Press,2008 • Text mining: classification, clustering and applications Srivastava, Ashok and Mehran Sahami (eds.)., 9781420059403, Chapman & Hall,2009 • Weiss, S. M., Indurkhya, N., Zhang, T. (2010). Fundamentals of Predictive Text • Mining. Springer: New York. ISBN: 978-1849962254 • Pustejovsky, J. and Stubbs, A. (2012). Natural Language Annotation for Machine Learning. O'Reilly. • Foundations and Trends in Information Retrieval, 2(1-2): 1–135. Available online at: http://www.cs.cornell.edu/home/llee/opinion-mining-sentiment-analysis-survey.html. • Manning, C. D., Raghavan, P., and Schutze, H. (2008). Introduction to Information Retrieval, Chapters 6 and 13-18, Cambridge University Press. Available online at: http://nlp.stanford.edu/IR-book/ • Articles: https://www.healthcatalyst.com

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Advanced Computer Architecture Lab
Course Code: ESD 601L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Basic Programming – preferably C, digital systems

Synopsis:	This Course provides insight on 1. Students able to understand various software tools, usages. 2. Able to understand difference between Behavioural and structural coding 3. Students will understand for gate level implementation 4. Develop the circuit model based on application
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	students able to understand knowledge of usage of software's
CO 2:	students able to construct the basic gates using various flatforms
CO 3:	Construction of circuit models based on application

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1		*									
CO 2			*								
CO 3			*	*	*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Installation of Xilinx Design suit	
Installation of Xilinx IDE, Project creation, building a project, running a sample project	At the end of the topic student should be able to: 1. Design and Demonstration of simple gate level project(C2)
Unit 2: Introduction Digital Basics	
Development of various Boolean equations, , Boolean functions using K	1. Practicing construction of circuits using combinational gates (C4)

map using combinations circuits, design of MSI models.	
Unit 3: Introduction of sequential design Circuits	
Designing basic elements of FlipFlops, latches, development of various types of flipflops, development finite state machines of class A And B.	1. Development of various types of flip flops(C4) 2. Design of various counter circuits, up counter, down counters, universal counter circuits , melay machine, moore machines, s (C6)
Unit 4: Multipliers and ADDERS	
Design and development of various types of adders, multipliers, Rom based multipliers, shift registers	1. Design and simulation of various bits of adders circuits (signed and unsigned numbered)(C6) 2. Designing and simulation of Booths multipliers of various inputs bits(C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Sessional Examination 1		*	*		
Sessional Examination 2				*	*
Assignment/Presentation				*	*
Laboratory examination		*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Samir Palnitkar " Verilog HDL A Guide to Digiatl Design and synthesis" 2. William H Gothamn Fundamentals of digital electronics 2nd edition 3. Donald E Thomas and Phillp R Mooray " The Verilog Hardware Deisgn Langauges. 4. CV Hamacher, Vranseic and Zaky , "Computer Organization", Fifth Edition, Tata-MacgrawHill 5. Rafiquzzamann ,"Modern Computer Architecture",Chandra,Galgotia Publications 6. John L Hennessy and David A Patterson ,"Computer Architecture: A Quantitative approach", 2nd Edition 7. John L Hennessy and David A Patterson ,"Principles of Computer Architecture", Prentice Hall 8. Internet sources.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Microcontrollers and its Applications Lab
Course Code: ESD 602L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Microprocessor architecture , Assembly language and Number systems
Synopsis:	This Course provides insight on 1. This course provides the knowledge of Intel 8051 and ARM Microcontrollers. 2. This course provides the knowledge of Microcontroller architecture, Registers and Instruction sets to write Assembly and Embedded C Programming. 3. This course provides the concept of Interfacing and Programming Sensors and Peripherals to Microcontrollers. 4. This course provides the concept of Designing Embedded Systems using Microcontrollers.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Employ the knowledge of Microcontrollers to build Embedded systems.
CO 2:	Explain the concept of Programming Microcontrollers using Assembly and Embedded C.
CO 3:	Design Embedded Systems by interfacing Sensors and Actuators.

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*		*		*						
CO 2	*	*			*						
CO 3	*	*	*		*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction to Microprocessor & Microcontrollers	
Comparison – Variants – Types – General – ASIC – PLD – Introduction to Motherboard(Desktop) - Introduction to Embedded Board – Compare and Contrast - Application Types –	At the end of the topic student should be able to: 1. List different IDE's to program Microcontrollers (C1)

Single Tasking – Multitasking – Multi-Application	2. Design a Environment with tools required to build Embedded systems using Microcontrollers (C3)
Unit 2: Introduction to ARM Microcontrollers	
Programming Model – Processor Modes – ARM vs Thumb Introduction to LPCxxxx Microcontrollers – Features – Detailing of Pins - Memory Map Concepts – RAM & ROM - Interrupts Concepts (Internal & External)	1. Demonstrate ARM Processor architecture specification using LPC 2148 Microcontroller Board (C3) 2. Demonstrate a peripherals of ARM Microcontroller using LPC 2148 Microcontroller Board (C3)
Unit 3: Reset Circuitry	
Crystals - Introduction to GPIO – Registers – Input /Output Configuration – Pull Up and Pull Down Resistor Concept – Interfacing with LED – Interfacing Push Buttons – LCD – Stepper Motor – DC Motor	1. Design an Digital notice board using LPC 2148 Microcontroller board to understand Peripherals on board (C3) 2. Design an Automated Fan / AC / Temperature control system using on chip sensors and peripherals of LPC 2148 Microcontroller board (C3)
Unit 4: Relays	
Types of Relays – Interfacing	1. Demonstrate working of Relay by controlling High voltage devices like DC Motor interfacing to ARM Microcontroller (C4)
Unit 5: Timer, Counter Introduction	
Configuration – Programming	1. Design a Digital clock using ARM Microcontroller using on chip Timer and Counter (C3)
Unit 6: Serial vs Parallel Bus	
Serial vs Parallel Bus - Compare and Contrast – Terminology: Baud Rate – Bit Rate – RS232 – DB9 handshaking concepts - Configuring Registers – Programming for UART modules.	1. Design a Master and Slave architecture using Microcontrollers and establish communication using on chip serial UART(c4)
Unit 7: Introduction to SPI and I2C Protocol	

Detailed Discussion – Bit Banging – Interfacing with SPI and I2C Devices – RTC / ADC /DAC.	1. Design a Serial wired communication among multiple Microcontrollers and sensors using I2C (c4) 2. Design a Serial wired communication among Microcontroller and multiple sensors in Master and Slave Architecture using SPI (c4)
Unit 8: Introduction to ADC and DAC	
Types – Chips - Register Configuration – Interfacing	1. Design a Data Acquisition system ARM Microcontroller (C4)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	

Sessional Examination 2		*	*
Assignment/Presentation	*	*	
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• William Hohl, Christopher Hinds,"ARM Assembly Language: Fundamentals and Techniques",2nd Edition, ISBN-13: 978-1482229851, ISBN-10: 1482229854 • Andrew Sloss, Dominic Symes, Chris Wright,"ARM System Developer's Guide: Designing and Optimizing System Software",1st Edition,The Morgan Kaufmann Series in Computer Architecture and Design, ISBN-13: 978-1558608740, ISBN-10: 1558608745 • David Seal, "ARM Architecture Reference Manual", 2nd Edition, Addison-Wesley Professional. • Steve Furber,"ARM System-on-Chip Architecture",2nd Edition,Addison-Wesley Professional, ISBN-13: 078-5342675191,ISBN-10: 0201675196 • Douglas V. Hall,"Microprocessors and Interfacing",Mcgraw Hill Educatin ,ISBN-10 1259006158,ISBN-13 9781259006159,2012. • Websites & Transaction Papers

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Computer Networks Lab
Course Code: CSE 610	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Basic of Computer Communication and Networks
Synopsis:	This Course provides insight on 1. Practical learning of concepts, applications, reference models of computer networks. 2. Functional knowledge about communication devices, IP addresses and routing algorithms for networking. 3. Implementation of routing, congestion, transport, multitasking and application layer protocols for analysis
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Identify the tools for applications of computer networks.
CO 2:	Describe the functions of communication devices, IP addressing techniques.
CO 3:	Demonstrate routing algorithms, congestion control mechanisms and transport layer protocols.
CO 4:	Examine application, multicasting and management protocols functions.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*										
CO 2		*									
CO 3			*								
CO 4			*								

Course content and outcomes:	
Content	***Competencies***
Unit 1: Computer Networks	
Definition, Network goals, Classification of networks, ISO – OSI and TCP/IP reference model	At the end of the topic student should be able to: 1. List different tools available for network implementation (C1). 2. Identify suitable network tools (C2) 3. Design different topologies of wired network (C5).

Unit 2: Communication Devices	
Network Interface card, modem, hub, switch, repeater, bridge, router and gateway.	1. Construct networks using various communication devices (C5) 2. Analyse packet forwarding methods of communication devices (C4).
Unit 3: Internet addresses	
Classes of IP addresses, Subnetting and Supernetting TCP/IP networks, Internet routing, host name resolution, Mapping IP addresses to physical addresses, ARP and RARP	1. Construct networks using Subnetting and Supernetting (C5) 2. Analyse ARP and RARP packet formats (C4)
Unit 4: Routing Algorithms	
Shortest path, Flow based, Distance vector, Link state, Hierarchical	1. Design of a network using different routing protocols (C5) 2. Modify the routing table content of a router in a network (C3)
Unit 5: Congestion control algorithms	
Congestion – causes, Congestion Prevention Techniques, Congestion Reactive Techniques	1. Design of a network using various congestion prevention and reactive techniques (C5)
Unit 6: Internet Transport protocols (TCP & UDP)	
Connection establishment and termination, Flow Control, Time out and retransmission process	1. Construct networks using different transport layer protocols (C5) 2. Analyse flow control and congestion control at transport layer (C4).
Unit 7: Application Protocols	
SMTP, DHCP, DNS, FTP	1. Design of a network using different application layer protocols (C5)
Unit 8: Multicasting and other Protocols	
Multicasting, IPV6, IGMP, ICMP, VOIP	1. Demonstrate multicasting protocol. (C3) 1. Construct networks for voice communication (C5) 2. Demonstrate various network management protocol. (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*

Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO5
Sessional Examination 1	*	*		*	*
Sessional Examination 2			*	*	*
Assignment/Presentation			*	*	*
Laboratory examination			*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• "Internetworking with TCP/IP Vol I: Principles, Protocols and Architecture", Douglas E Comer, III Ed. PHI, 1997. • "Microsoft TCP/IP on Windows NT 4.0", MCSE. • "Internetworking with TCP/IP Client–Server Programming and applications, BSD Socket version", Douglas E Comer and David L Stevens, Vol. III. • "TCP/IP Illustrated, Volume I, The Protocols", W Richard Stevens, International Student Edition, 1999. • "Advanced Internet Technologies", Uyless Black, PHI • "High Performance Communication Networks", Jean Warland & Praveen Varaiya – Morgan Kaufmann

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Database Programming in Java Lab
Course Code: CSE 604L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Basic Programming knowledge
Synopsis:	This Course provides insight on 1. To provide fundamental knowledge of various object oriented programming concepts and database concepts. 2. To design and develop database applications using java programming language
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Apply object oriented programming concepts in a java application
CO 2:	Practice various types of UI based applications
CO 3:	Manipulate database using various SQL Commands
CO 4:	Write java applications for various database requirements

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1		*	*		*						
CO 2		*	*		*						
CO 3		*	*		*						
CO 4		*	*		*						

Course content and outcomes:	
Content	*Competencies*
Unit 1: Installation of JDK tools	
Installation of JDK tools, setting environment variables for java application, writing simple java program, practice to compile and run java application	At the end of the topic student should be able to: 1. Use of JDK tools for java application (C3) 2. Solve the issues related to java setting environment variables (C3) 3. Analyse simple java application (C4)
Unit 2: Introduction to OOP's concepts	
Implementation of OOP's concepts in java application such as encapsulation, various types of Inheritance,	1. Apply OOP's concepts in java application (C3) 2. Solve the issues such as multiple inheritance, exception handling(C3)

polymorphism. Apart from this other techniques such as exception handling, packages, interfaces, IO streams.	3. Write java programs to understand more about file read and write (C3)
Unit 3: Introduction to Window based applications	
Implementation of window based applications using swing components such as forms, menu based applications. Applying event handling mechanism to the applications	1. Write UI applications for different look and feel (C3) 2. Use of swing components and layout managers for UI design (C3) 3. Test UI applications (C3)
Unit 4: Database applications using JDBC driver	
Installation of JDBC driver, use of it in database applications, creating database, manipulating data through window based applications	1. Test various Structured Query Language (SQL) commands (C4) 2. Write database applications using JDBC driver and mysql database (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	36	72
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-
Revision		
Assessment		
TOTAL	**72**	**126**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO4
Sessional Examination 1	*	*		

Sessional Examination 2			*	*
Assignment/Presentation	*	*	*	*
Laboratory examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• Patrick Naughton and Herbert Schildt – "JAVA 2 – The Complete Reference", Tata McGraw Hill. • George Reese - “Database Programming with JDBC and Java”, O'Reilly • “Database system Concepts”,Author: Abraham Silberschatz (Bell Laboratories), Henry F. Korth(Bell Laboratories) and S. Sudarshan (Indian Institute of Technology, Bombay, Publishers: The McGraw-Hill Companies, Inc. • “Fundamentals of Database systems”. Author: Elmasri and Navath

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Internet of Things Lab
Course Code: IOT 607L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Computer Networks, Programming aspects.

Synopsis:	This Course provides insight on 1. Various elements involved in the development of application for IoT. 2. Understanding of protocols across IoT stack. 3. Scripting languages like shell and python. 4. Client Server architecture and Python APIs of Socket programming. 1. Database and Python Database connectivity, Python Web Programming, IoT Framework.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Explain basic principles of Python programming language. (C2)
CO 2:	Demonstrate the usage of networking protocols across IoT stack using Raspberry Pi and Cloud. (C3)
CO 3:	Demonstrate the fundamental concepts in Client Server architecture, database implementation and web programming with Python API's. (C3)

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*										
CO 2	*	*		*	*						
CO 3	*		*		*						

Course content and outcomes:

Content	***Competencies***
Unit 1: Python	
Introduction to Python datatypes, constructors, functions, Python Class, Modules, exception Handling, Python Packages	At the end of the topic student should be able to: Employ Datatypes, Constructs, Packages in python programming. (C2)
Unit 2: Raspberry PI IoT Board	

Introduction to RPI, Raspberry Pi - Installation, first boot configuration, Raspberry Pi - Sensor Interfacing, Sending data to Cloud.	Demonstrate the usage of RPI in IoT Application Scenario. (C3)
Unit 3: Things Board Cloud	
Installation of things board Platform, Device, assets & dashboard Creation, population of data.	Illustrate the usage of things board Platform. (C4)
Unit 4: Socket Programming	
Unix Socket Programming - Client Server Architecture, Python Socket Programming - Client Server Architecture, RAW packets python programming	Illustrate the socket communication using python API's for RWA, stream and datagram-oriented use cases. (C3)
Unit 5: Databases	
Python Database connectivity (CRUD) - Web Server Concepts - Python Web Programming – IoT Framework.	Demonstrate the usage of databases, web programming using Python API . (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination

Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	*
Assignment/Presentation		*	*
Lab Semester Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• Arshdeep Bhaga, Vijay Madishetti, “Internet of things:A hands on Approach”, Universities Press, ISBN:978172719547 • Robert Faludi,”Building Wireless Sensor Networks”,Orielly, 2012 • Jean-Philippe Vasseur,Adam Dunkels,"Interconnecting Smart Objects with IP: The Next Internet",Morgan Kaufmann Publishers,2010,ISBN:0123751659 9780123751652 • Marco Schwartz,"Internet of Things with the Arduino Yun",Packt Publishing,2014 • Charalampos Doukas,"Building Internet of Things With the Arduino: Volume 1",CreateSpace Independent Publishing Platform,2012 • Todor Cooklev ,“Wireless communication standards”, IEEE Press • Houda Labiod, Hossam Afifi, Costantino De Santis, “Wi-Fi, Bluetooth, Zigbee and WiMAX”, Springer Publications • Madhushree Ganguli , “Getting started with Bluetooth”, Premier Press, 2002, ISBN 1931841837, 9781931841832.

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Seminar - 1
Course Code: ESD 697	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Communication Skill
Synopsis:	This Course provides insight on 1. To select, search and learn technical literature. 2. To Identify a current and relevant research topic. 3. To prepare a topic and deliver a presentation. 4. To develop the skill to write a technical report. 5. Develop ability to work in groups to review and modify technical content
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Show competence in identifying relevant information, defining and explaining topics under discussion.
CO 2:	Show competence in working with a methodology, structuring their oral work, and synthesizing information.
CO 3:	Use appropriate registers and vocabulary, and will demonstrate command of voice modulation, voice projection, and pacing.
CO 4:	Demonstrate that they have paid close attention to what others say and can respond constructively.
CO 5:	Develop persuasive speech, present information in a compelling, well-structured, and logical sequence, respond respectfully to opposing ideas, show depth of knowledge of complex subjects, and develop their ability to synthesize, evaluate and reflect on information.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*							*	*		*
CO 2	*							*	*		*
CO 3	*							*	*		*
CO 4	*							*	*		*
CO5:	*							*	*		*

Learning strategies, contact hours and student learning time

Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	-	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	14	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-
Clinic	-	-
Practical	-	-
Revision	-	-
Assessment	-	-
TOTAL	**14**	**-**

Assessment Methods:	
Formative:	**Summative:**
Seminar Topic Selection	
Synopsys review	
PPT Review	

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Presentation	*	*	*	*	*

Feedback Process	End-Semester Feedback
Reference Material	Particular to the chosen Seminar

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Mini Project - 1
Course Code: ESD 695	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 1
No of Credits: 4	**Prerequisites:** Any programming language and circuit basics
Synopsis:	Students are expected to select a problem in the area of their interest and the area of their specialization that would require an implementation in hardware / software or both in a semester
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Apply the objectives of the project work and provide an adequate background with a detailed literature survey
CO 2:	Breakdown the project into sub blocks with sufficient details to allow the work to be reproduced by an independent researcher
CO 3:	Compose hardware/software design, algorithms, flowchart, methodology, and block diagram
CO 4:	Evaluate the results
CO 5:	Summarize the work carried out

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1				*							
CO 2					*			*			
CO 3							*			*	
CO 4						*					*
CO5:							*				

Course content and outcomes:	
Content	***Competencies***
Phase 1	
Problem identification, synopsis submission, status submission, mid evaluation.	At the end of the topic student should be able to: 1. Identify the problem/specification (C1) 2. Discuss the project (C2) 3. Prepare the outline (C3) 4. Describe the status of the project (C2) 5. Prepare a mid-term project presentation report (C3) 6. Prepare and present mid-term project presentation slides (C3, C5) 7. Develop project implementation in hardware/software or both in chosen platform (C5)
Phase 2	
Status submission, final evaluation.	1. Prepare the progress report (C3) 2. Prepare the final project presentation report (C3) 3. Prepare and present final project presentation slides (C3, C5) 4. Modify and Develop implementation in hardware/software or both in chosen platform (C3, C5) 5. Justify the methods used and obtained results (C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	-	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	48	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-

Clinic	-	-
Practical	-	-
Revision	-	-
Assessment	03	-
TOTAL	**51**	**09**

Assessment Methods:	
Formative:	**Summative:**
Project Problem Selection	Mid-Term Presentation
Synopsys review	Second status review
First status review	Demo & Final Presentation

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Mid Presentation	*	*			
Presentation	*	*	*	*	*

Feedback Process	End-Semester Feedback
Reference Material	Particular to the chosen project

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Digital Signal Processing
Course Code: ESD 603	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** Computer Networks, Programming aspects.
Synopsis:	This Course provides insight on 1. Understanding of basics of Signal and Systems as pre-requisite. 2. Understanding the concepts of Fast Fourier Transforms. 3. Learning hardware implementation of systems. 4. Learning FIR and IIR Filter Designs. 5. Learning concepts of multi-rate signal processing in the form of sampling rate conversion, structures of sampling rate converters and some applications of sampling rate converters 6. Understanding three optimum Weiner filters, adaptive algorithm and transforming Weiner filters in to adaptive filters 7. Understanding architecture, memory management and pipelining concepts of TMS320C67XX processor through self-stud.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Analyse Fast Fourier Transform (FFT) algorithms on computational complexity. (C4)
CO 2:	Describe the structures for IIR and FIR filters. (C2)
CO 3:	Interpret Multirate Signal Processing and Adaptive Filters. (C3)
CO 4:	Explain architecture, memory management and pipelining concepts of General and TMS320C67XX Digital Signal Processor. (C2)

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*		*						
CO 2	*	*	*	*	*						
CO 3		*	*	*							
CO 4	*	*									

Course content and outcomes:	
Content	***Competencies***
Unit 1: Review: (Self Study)	

Introduction Classification of signals and systems, brief discussions on z-transform, inverse z-transform & Fourier transform, DFT, linear convolution using circular convolution & DFT	At the end of the topic student should be able to: 1. Outline types of signals and system. (C1) 2. Summarize z-transform, Fourier transform, convolution. (C2)
Unit 2: FFT Algorithms	
Radix-2 DIT-FFT Algorithm, DIF-FFT Algorithm. Assignments (Problems).	1. Identify Computation complexity of DFT, Introduction to Fast Fourier Transform (FFT) algorithm (C1) 2. Describe and Sketch Radix-2 Decimation in Time FFT (DIT-FFT) Algorithm and analyse its computation complexity (C2, C3, C4) 3. Describe and Sketch Radix-2 Decimation in Frequency FFT (DIF-FFT) Algorithm and analyse its computation complexity (C2, C3, C4)
Unit 3: Filter Structures	
IIR Filter Structure – Direct Form I & II, CSOS, PSOS & Transpose structures - FIR Filter Structures – Direct Form, Cascade form, Linear Phase Filter structures. Assignments (Problems).	1. List Components used in filter structures, System Representations, relation between the representations, classify of IIR and FIR Systems (C1, C2) 2. Explain and construct IIR Filter Structure – Direct Form-I, Direct Form–II, Cascade Form (CSOS), Parallel Form (PSOS) & Transpose of structures (C2, C5) 3. Explain and construct FIR Filter Structures – Direct Form, Cascade form (C2, C5) 4. Explain Linear Phase FIR Filter structure: Derivation, Frequency Response, Compute Computation Complexity and construct with number of filter coefficients being even and odd. (C3, C5)
Unit 4: Design of FIR filters	
Using Frequency Sampling & Windows - Assignments (Problems).	1. Introduction to Frequency sampling technique design 2. Describe Derivation of a Transfer Function for the system designed using frequency sampling technique when number of samples of impulse response / number of point DFT is even or odd.

	Construct hardware for the transfer functions. Concept of Comb filter and resonator (C6, C5) 3. Sample example to Design and implement FIR filter using Frequency Sampling technique to meet required impulse response (C5, P4) 4. Illustrate Frequency responses of frequency selective (LP, HP, BP and BR) filters, concept of frequency sampling in the frequency responses (C3) 5. Sample examples to Design and implement FIR filters with ideal frequency response using frequency sampling technique (C5, P4) 6. Discuss Concept of windowing in the design of FIR filter, Concept of Gibb's Phenomenon and its effect on frequency response, Use of window functions to eliminate Gibb's effect (C2) 7. Comparison of performances of filters designed with different window functions (C4) 8. Explain Steps involved in the design of FIR filters with ideal frequency response and non-ideal frequency response (C2) 9. Express Impulse responses of frequency selective filters (C2) 10. Sample examples to design ideal and non-ideal frequency selective filters using windows. (C5)
Unit 5: Design of IIR Filters	
Butterworth & Chebychev filters design using impulse invariance & bilinear transformation techniques, Design of IIR filter using pole placement technique. Assignments (Problems).	1. Discuss Concepts of Analog Butterworth LP filter, concept of Cut-off frequency, order of the filter, compute poles, pole locations in S-Plane, transfer function C3) 2. Explain Design steps of Analog Butterworth LP filter (C2) 3. Explain Chebychev polynomials, their properties, Analog Chebychev LP filter function, concepts of frequency response, order of filter, pole placements of Chebychev LP filters on S-Plane, compute poles, Transfer function of LP Chebychev filter (C3)

	4. Discuss Concepts of Impulse Invariance Transformation, S-Plane to Z-Plane mapping, steps in transformation (C2) 5. Discuss Concepts of Bilinear Transformation, frequency warping, pre-warping for the purpose of analog filter (Butterworth / Chebychev) design (C2) 6. Sample examples to design Butterworth and Chebychev LP filter using impulse invariance and bilinear transformations (C5)
Unit 6: Multirate Signal Processing	
Decimation, Interpolation, Sampling rate conversion by a rational factor, structures, Polyphase filter structures, Time variant Filter structure, Application of Multirate signal processing to Phase Shifter, Subband coding of Speech signal, Digital Filter Bank Implementation, QMF Filter bank	1. Introduction, need for multi-rate signal processing, explain concept of sampling rate conversion (C2) 2. Explain Decimation by an integer factor, block diagram, analyse of decimator in time domain and frequency domain C4) 3. Explain Interpolation by an integer factor, block diagram, analyse of interpolator in time domain and frequency domain (C4) 4. Explain Sampling rate conversion by a rational factor, block diagram, analyse in time domain and frequency domain (C2) 5. Construct Implementation of Sampling rate converters (C5) 6. Discuss Concepts and construction of Poly-phase filter (C2) 7. Construct Time variant Filter (C5) 8. Apply Multi-rate signal processing concept to Phase Shifter, Sub-band coding of Speech signal, Digital Filter bank Implementation, QMF Filter bank. (C3)
Unit 7: Adaptive Filters	

Class of Optimal Filters – Predictive Configuration, Filter Configuration, Concept of adaptive noise cancellation, Noise Canceller Configuration. LMS adaptive Algorithm, Application of LMS algorithm to the optimal filter configurations. Adaptive noise canceller as a high-pass filter	1. Outline adaptive filters, some matrix operation.(C1) 2. Explain Optimal Weiner Filters – Predictive Configuration, Filter Configuration, Noise Canceller Configuration (C2) 3. Explain Concept of LMS adaptive Algorithm (C2) 4. Apply LMS algorithm to the optimal filter configurations (C3)
Unit 7: DSP Processor	
Introduction to PDSPs – Multiplier and Multiplier Accumulator (MAC), Modified Bus structures and memory access schemes, Multiple access memory, Multiported Memory, VLIW architecture, Pipelining, Special addressing modes, On-chip Peripherals. TMS320C6711 DSP processor: Architecture, Instruction set and assembly language programming	1. Discuss Introduction to PDSPs – Multiplier and Multiplier Accumulator (MAC), Modified Bus structures and memory access schemes (C2) 2. Explain Concept of Multiple access memory, Multiported Memory, VLIW architecture (C2) 3. Explain Concept of Pipelining, Special addressing modes, On-chip Peripherals. (C2) 4. Explain Concepts on Architecture, memory organization and pipelining of TMS320c67XX (C2)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*		
Sessional Examination 2			*	
Assignment/Presentation		*	*	
End Semester Examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Sanjith K Mitra, "Digital Signal Processing", McGraw Hill Education, 4 Edition, July 2013. 2. Oppenheim and Schafer, "Digital Signal Processing", Pearson, First Edition, 1975. 3. Roman Kuc, "Digital Signal Processing", McGraw-Hill Education, 1988. 4. Proakis and Manolakis, "Digital Signal Processing", Prentice – Hall, Inc., Third Edition, 1996. 5. Rabinder and Gold, "Theory and Application of Digital Signal Processing", Prentice Hall India Learning Private Limited, 1988. 6. Hwei P Hsu, Schaum's Outline of "Signals and Systems", 3rd Edition, 2013. 7. Symon Haykins, "Signals and Systems", Wiley, Second Edition, 2002.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Device Drivers
Course Code: ESD 604	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** Basic C Programming
Synopsis:	This Course provides insight on 1. Insight into Linux kernel programming. 2. Knowledge about the framework used in building the Linux device driver. 3. Concept of designing proc and ioctl needed to build a device driver 4. Techniques to debug kernel programs 5. Insight into designing USB drivers.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Explain the broad concept of device drivers and build character drivers
CO 2:	Describe design of kernel modules and debugging these modules
CO 3:	Handle concurrency, race condition and understand the importance of time while designing a device driver
CO 4:	Allocate dynamic memory and communicating with devices though I/O ports
CO 5:	Demonstrate and design USB drivers on a kit

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*										
CO 2	*	*	*								
CO 3		*	*								
CO 4	*		*								
CO 5	*				*						

Course content and outcomes:	
Content	***Competencies***
Unit 1:	At the end of the topic student should be able to:
Introduction to Device Drivers	1. Describe the broad design of device driver (C3)

Unit 2:	
Building & Running Modules.	1. Compile and load modules using a make file (C4)
Unit 3:	
Character Driver.	1. Explain the structure of a character driver (C3)
Unit 4:	
Debugging Techniques.	1. Debug modules using prink, proc and kdb (C4) 2. Design of Ioctl used in building device drivers (C5)
Unit 5:	
Concurrency and Race Condition	1. Illustrate the problems associated with concurrent device drivers (C3) 2. Describe the problems associated with race condition while designing a device driver (C3)
Unit 6:	
Advanced Character Driver Operations	1. Execute bottom half through deferred work (C4)
Unit 7:	
Time, Delay and Deferred Work	1. Use the concept of delays (C2) 2. Explain the concept of timers in Linux kernel (C2)
Unit 8:	
Allocating Memory	1. Allocate dynamic memory (C3) 2. Explain the concept of memory barriers (C3)
Unit 9:	
Communicating with Hardware	1. Communicate with the devices through I/O ports (C4)
Unit 10:	
Interrupt Handling	1. Illustrate the concept of writing interrupt handlers (C4)
Unit 11:	
PCI Drivers, USB Drivers	1. Structure of a USB driver (C4) 2. Design a USB driver. (C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04

Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Sessional Examination 1	*	*			
Sessional Examination 2			*	*	
Assignment/Presentation		*		*	*
End Semester Examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Alessandro Rubini, “Linux Device Drivers”, (Nutshell Handbook), O'Reilly Publishers, 2009. 2. John Madieu, “Linux Device Drivers Development: Develop customized drivers for embedded Linux”, Packt Publishing, 2017. 3. Robert Love, “Linux Kernel Development”, Addison Wesley, Third Edition, 2010. 4. Daniel P. Bovet, Marco Cesati, “Understanding the Linux Kernel”, O'Reilly Media, Third Edition, 2008. 5. Wolfgang Mauerer, “Professional Linux Kernel Architecture”, Wrox, 2008. 6. Sreekrishnan Venkateswaran, “Essential Linux Device Drivers”, Prentice Hall, 2008. 7. W. Richard Stevens, Stephen A. Rago, “Advanced Programming in the UNIX Environment”, Addison Wesley, Third Edition, 2013.

	8. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, "Unix Network Programming, Vol1: Sockets", Pearson Education India, Third Edition, 2015.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Embedded Systems
Course Code: ESD 604	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** Microprocessor architecture, Microcontroller Architecture, Assembly language and Number systems

Synopsis:	This Course provides insight on 1. This course provides the knowledge of ARM Cortex M3 Processor architecture 2. This course provides the knowledge of Microcontroller based on ARM Processor architecture and its Registers and Instruction sets to write Assembly and Embedded C Programming. 3. This course provides the concept of Interfacing and Programming Sensors and Peripherals to Microcontrollers. 4. This course provides the concept of Communication Protocols required for multi-processor communication. 5. This course provides the concept of Real time operating systems on Microcontrollers. 6. This course provides the concept of Designing Real Time Embedded Systems using ARM Microcontroller.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Employ the knowledge of Microcontrollers to build Embedded systems. (C3)
CO 2:	Explain the concept of Programming ARM Microcontrollers using Assembly and Embedded C. (C2)
CO 3:	Design a Real time Embedded Systems by interfacing Sensors, Actuators and porting Real time operating systems. (C5)

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*									
CO 2	*	*	*		*						
CO 3	*	*	*		*						

Course content and outcomes:

Content	*Competencies*
Unit 1: Introduction to Embedded Systems	
Design Challenges, Processors Technology, Design Technology	At the end of the topic student should be able to: 1. Describe the Design issues in designing the Embedded Systems.(C1) 2. Discuss the design technology associated with Embedded Systems.(C2)
Unit 2: Introduction to ARM Cortex processor	
Variants of Cortex and ARM versions, Comparison of M-series processor, Architecture, Programmers Model, APSR register, Memory Model, Exception, Interrupts, Reset	1. Explain about ARM Processor architecture (C2) 2. Describe ARM Cortex m3 processor data path, Register set, Programming models and memory map (C2) 3. Describe about ARM Cortex M3 Processor Instruction set. (C2) 4. Describe about ARM Processor system bus and Interrupt controller (C2) 5. Describe about interrupt and Exception handling (C2) 6. Describe ARM Microcontroller architecture. (C2)
Unit 3: Instruction Set Architecture	
More on Memory System, Exceptions and Interrupts, NVIC, Memory Protection Unit, Assembly Programming, Embedded C programming, CMSIS, Startup Code	1. Describe ARM Cortex memory system. 2. Describe interrupt and Exception handling (C2) 3. Describe NVIC, Memory Protection Unit. (C2) 4. Discuss CMSIS implementation in ARM Cortex.(C2)
Unit 4: Introduction to LPC13/17xx Microcontroller	
Memory Mapping, Registers involved and programming with GPIO, PWM	1. Discuss Memory Mapping, Registers involved and programming with GPIO, PWM. (C3) 2. Apply knowledge of ARM Microcontroller architecture to rig up Embedded system circuits(C3)
Unit 5: Data Acquisition System	
ADC, Types of ADC, Choosing the ADC, DAC	1. Identifying various types of ADC. (C1) 2. Review ADC and DAC selection criteria. (C2)
Unit 6: Serial Communication	
UART, I2C, SPI, Interfacing	1. Discussing various types of Serial Communication mechanism. (C2)

Unit 7: USB BUS	
Speed Identification on the bus, States, Packets, Data flow types, Enumeration, Descriptors, USB Interface –C Programs	1. Identify USB types, Firewire devices, ports, cables. 2. Describing Enumeration, Descriptors mechanism in USB.(C2)
Unit 8: CAN BUS	
Introduction, Frames, Bit stuffing, Types of errors, Nominal Bit Timing, A simple application with CAN	1. Describe the nature of CAN and the basic CAN protocol, and the basic structure of a CAN network. (C2) **2.** Prepare a simple application with CAN. (C3)
Unit 9: Introduction to Multitasking in Microcontrollers	
Variants of RTOS, FreeRTOS, UCOS, uCLinux, FreeRTOS on Cortex based Microcontrollers, TASK CREATION, QUEQUES, SEMAPHORE, MUTEX, Application development	1. Describe about Real time operating systems role in building real time systems (C3) 2. Describe about Designing Real Time Embedded systems by interfacing peripherals and actuators (C2) 3. Design a Real time Embedded system by writing applications on top of Real time operating systems (C5)
Unit 10: Designing a Digital Camera	
Introduction, Requirement, Specifications, Implementation, Testing	**1.** Summarize the stages involved in designing a digital camera. (C2)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:

Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2		*	*
Assignment/Presentation	*	*	
End Semester Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Joseph Yiu, "The definitive guide to the ARM Cortex-M3", Elsevier, 2nd Edition, 2010. 2. Frank Vahid, Tony Givargis, "Embedded System Design: A Unified Hardware/Software Introduction", Wiley India, ISBN:81-265-0837-X, 2007. 3. Richard Barry, "NXP Semiconductors, LPC13xx/17xx User Manual", 2012. 4. NXP Semiconductors, "LPCzone Examples", 2012. 5. "FreeRTOS Reference Manual", Real Time Engineers Ltd., 2016.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Embedded Software Design
Course Code: ESD 606	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** Embedded Systems Object oriented padagrime
Synopsis:	This Course provides insight on 1. Students learn the concept of big data characteristics, batch and lambda architecture. 2. This course introduces students to basics file systems in Big Data 3. This course helps the student to understand the concepts of Hadoop framework, Spark framework and their internals. 4. This course helps the students to learn Map-reduce programming, Spark programming. 5. Students learn the different layers with use cases demonstrations.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	To build and analyse models for embedded application using the concept of UML. (C4).
CO 2:	To work with UML tools and represent the model using suitable diagrams. (C3)
CO 3:	To write applications using the OOP concepts.(C3)
CO 4:	To write applications using JAVA constructs for general purpose and embedded systems. (C3)

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*									
CO 2	*	*			*						
CO 3	*		*								
CO 4	*		*								

Course content and outcomes:	
Content	***Competencies***
Unit 1: Object-oriented principles of composition	
Inheritance - Aggregation and containment – Delegation -	At the end of the topic student should be able to: 1. Explain the concept of Inheritance. (C2)

Structural design patterns for composing objects.	2. Explain the concept of Aggregation and containment. (C2). 3. Explain the concept of Delegation. (C2). 4. Explain the concept of Structural design patterns for composing objects. (C2).
Unit 2: Specification of object-oriented systems	
Specification of object-oriented systems: UML for specifying functional requirements - Use cases and Scenarios - Subsystems, packages and deployment - Assigning responsibilities to objects in UML- Specifying quality attributes: Performance- Security – Privacy- Safety.	1. Basics of UML concepts. (C1). 2. Design UML modelling object-oriented systems. (C2). 3. Explain and illustrate Class diagrams Collaboration diagrams Sequence diagrams, State diagrams. (C3).
Unit 3: Modelling object-oriented systems:	
UML for modelling object-oriented systems- Class diagrams- Collaboration diagrams-Sequence diagrams- State diagrams.	1. Building models using UML tools. (C3)
Unit 4: Modelling real-time embedded systems behaviours	
UML real-time profile.	1. Review and understand the UML real-time profile. (C3).
Unit 5: Developing object-oriented systems in Java	
Classes, interfaces, methods- Generics- Scope rules and access control. Inner classes-Functional programming constructs – lambdas-Threads, concurrency control and timers-I/O, Streams and network I/O-Security and Cryptography.	1. Design Classes, interfaces, methods.(C3). 2. Explain the Generics-Scope rules and access control.(C3). 3. Explain and illustrate Functional programming constructs.(C3). 4. Explain and lambdas.(C3). 5. Explain and illustrate the Threads, concurrency control and timers-I/O, Streams and network concepts.(C3). 6. Explain and illustrate the I/O-Security and Cryptography.(C4).

Unit 6: Testing Java programs	
Challenges in testing object-oriented Program-Functional testing-Testing quality properties of the system- Java SE Embedded.	1. Discover the Challenges in testing object-oriented Program. (C3). 2. Explain the concept of Functional testing. (C2). 3. Testing quality properties of the system. (C2). 4. Practice Java SE Embedded. (C2).
Unit 7: Compact Profile	
Overview and technical details-Compact1, Compact2, and Compact3 profiles and their capabilities-Designing systems using embedded profile.	1. Overview and technical details-Compact1, Compact2, and Compact3 profiles and their capabilities. (C2). 2. Designing systems using embedded profile. (C2).
Unit 7: Realtime and Embedded Specification for Java:	
Real-time threads – Asynchrony-Time. Clocks and Timers-System and Options – POSIX realtime signals-Examples of programs using realtime specifications for Java.	1. Design Real-time threads. (C3). 2. Explain the concept of Asynchrony-Time. Clocks and Timers-Systems. (C4). 3. Design and explain the POSIX real-time signals. (C4). 4. Design Examples of programs using real-time specifications for Java. (C4).

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	36	72
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-
Clinic	-	-
Practical	36	72
Revision	-	-
Assessment	6	-

TOTAL	**78**	**144**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*		
Sessional Examination 2		*	*	
Assignment/Presentation	*	*	*	*
Laboratory examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition), Addison-Wesley Professional. 2003. 2. The Java Programming Language. Ken Arnold, James Gosling and David Holmes. Addison-Wesley Professional; 4 edition (August 27, 2005) 3. http://www.oracle.com/technetwork/java/embedded/resources/tech/compact-profiles-overview-2157132.html 4. Realtime Specification for Java 2.0 https://java.net/projects/rtsj-2/pages/Home

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Mobile Application Development using Android
Course Code: CSE-605	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** Basic knowledge of OOP's concepts, Java programming language

Synopsis:	This Course provides insight on 1. This course would provide fundamental knowledge about android platform. 2. The course will also provide skill sets to design and develop android applications for mobile devices. 3. This course will provide basic knowledge about android application communication of data which are hosted in remote systems.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Explain android architecture and framework
CO 2:	Discuss major building blocks of an android application
CO 3:	Write android applications using various UI components and data handling using SQLilte
CO 4:	Discuss advanced topics such as LBS, Mapping, Network connectivity, background threads, adapters

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*									
CO 2	*	*		*							
CO 3		*	*		*						
CO 4		*	*	*	*						

Course content and outcomes:

Content	***Competencies***
Unit 1: Introduction	
Introduction to Android and Eclipse environment, Android application framework, Unique aspects of	At the end of the topic student should be able to: 1. Explain android architecture (C2) 2. Discuss major building blocks such as activity, services, broadcast receiver and content provider (C2)

mobile application, software engineering issues for mobile application development	3. Identify different features in android studio (C1) 4. Discuss software engineering issues for mobile application development (C2)
Unit 2: Android building blocks	
Android manifest file, Dalvik virtual machine, DDMS, ADT, Adb, Android emulator, Activities and intents, creating a project, Android activity lifecycle, starting a new 'Hello World' Android application, Running and Debugging applications.	1. Explain Android manifest file (C2) 2. Discuss DVM, DDMS, android emulator (C2) 3. Describe android activity (C2) 4. Illustrate android activity lifecycle (C2) 5. Discuss the issues related running and debugging applications (C2)
Unit 3: Android Screen UI Components	
Layouts: LinearLayout, AbsoluteLayout, TableLayout, RelativeLayout, FrameLayout, ScrollView, Views: TextView, EditText, and Button views, TimePicker and DatePicker views, ListView and the Spinner views, Gallery and ImageSwitcher views, context sensitive menu .	1. Describe different types of layouts (C2) 2. Distinguish between various types of layout (C2) 3. Identify different types of android UI elements required for developing forms (C1)
Unit 4: Data management with SQLite	
SQLite architecture, creating and using databases, DBAdapter class, Common SQLite commands, creating triggers, logging insert, delete, update using SQLite, managing persistent data, Development of a simple healthcare application	1. Describe SQLite architecture (C2) 2. Discuss the use of SQLite database (C2) 3. Discuss the CRUD operations (C2) 4. Apply CRUD operations to develop a simple healthcare application (C3)
Unit 5: Advanced topics	
Adapters, background threads, Notifications, Location based services, Mapping, network connectivity services, telephony services	1. Explain adapter class (C2) 2. Discuss the various components of notification object in an android application (C2) 3. Discuss the use location based service classes (C2) 4. Identify the classes required for network applications (C1) 5. Define android service (C2) 6. Explain life cycle of service (C2)

	7. Discuss on background threads in android applications (C2)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*		
Sessional Examination 2			*	*
Assignment/Presentation		*	*	*
End Semester Examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Lauren Darcey and Shane Conder, "Sams Teach Yourself Android Application Development in 24 Hours", Sams Publishing, First Edition, ISBN-10: 0321673352, ISBN-13: 978-0321673350, 2010. 2. Ed Burnette, "Hello, Android: Introducing Google's Mobile Development Platform", Pragamatic, Third Edition, ISBN-10: 1934356565, ISBN-13: 978-1934356562, 2011.

	3. Rick Rogers and John Lombardo, "Android Application Development: Programming", O'Reilly Media, First Edition, ISBN-10: 0596521472 , ISBN-13: 978-0596521479 , 2009. 4. Reto Meier , "Professional Android 2 Application Development (Wrox Programmer to Programmer)", Wrox, Second Edition, ISBN-10: 0470565527, ISBN-13: 978-0470565520, 2010.

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

Name of the Program:	Master of Engineering - ME (Embedded Systems \)
Course Title:	Web Application Development
Course Code: CSE-611	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** Basic C , Fundamentals of Object Oriented Programming

Synopsis:	This Course provides insight on 1. This course provides an insight into web applications. 2. This course provides knowledge about databases and their use in developing web applications. 3. This course introduces ASP.NET technology and its use to build web applications. 4. This course provides knowledge about Client-Server Architecture. 5. This course provides knowledge about HTML, CSS, Javascript 6. This course provides knowledge about Web Service.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Design a database and normalize the database.
CO 2:	Identify the appropriate types of data to be stored.
CO 3:	Develop a Web application using ASP.NET.
CO 4:	Recognize the need for Client-Server Architecture.
CO 5:	Construct a web page using HTML and CSS.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*								
CO 2		*	*								
CO 3		*	*		*						
CO 4		*		*							
CO 5		*	*		*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Database Concepts	
Introduction to Transact SQL, Database optimization Techniques (Normalization), Creating database, Queries, sub queries, Joins, Stored Procedures, Triggers Tools: SQL server 2008 R2	At the end of the topic student should be able to: 1. Design and develop a database (C5) 2. Apply normalization to optimize a database (C3) 3. Manipulate data in a database through Queries (C3) 4. Create stored procedures, Triggers to manipulate data in database (C3, C5).
Unit 2: Programming	
ASP.NET: Introduction to ASP.NET, Clint server and web application design of presentation, business logic and storage functionality. WIN form and WEB form Designs using ASP.NET, Silverlight, Windows Client, HTML basics, CSS, AJAX, Java Scripts, Styling with Themes, Componentization (Code behind, Data Layers, User Controls), Roles and Profiles Tool: Visual Studio 2010	1. Use HTML to create webpages (C3, C5) 2. Use CSS and JavaScript on web pages (C3, C5) 3. Design and Develop web applications using ASP.Net (C3, C5) 4. Define, Client-Server Model (C1) 5. Discuss the need for Client-Server Architecture (C3, C5). 6. Develop forms using WIN forms (C3, C5). 7. Summarize the need for Componentization (C3, C6)
Unit 3: Web Services	
The life Cycle of a Web Service, Structure of Web Service, Creating a Web Service, SOAP, Data Contracts, Binding, Security, Discovery, Publishing, WSDL	1. Explain lifecycle of Web services (C2). 2. Define SOAP, Data Contracts (C2).

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02

Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Sessional Examination 1	*	*	*		
Sessional Examination 2			*	*	*
Assignment/Presentation	*	*	*	*	*
End Semester Examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Beginning ASP.NET 4.0 with C# byChris Hart, John Kauffman, David Sussman, Chris Ullman 2. Professional C# - Simon Robinson, Christian Nagel- Wiley Publishing, Inc.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Multicore Program Optimization
Course Code: CSE-612	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** computer architecture knowledge, programming – preferably c

Synopsis:	This Course provides insight on 1. They will understand learn various architectures and technological trends 2. Able to understand difference between single core system execution and multicore systems environments 3. Memory consistence models and applications 4. Able to understand optimisation and performances
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Distinguish between single core , multicore architectures, various architectures, trends, various levels of parallelisms
CO 2:	Illustrate Various cache coherence, issues, memory consistency models, various protocols , working principles , performances
CO 3:	Analyse Justification of primitives, optimisations, applications

Mapping of COs to POs											
COs	***PO 1***	***PO 2***	***PO 3***	***PO 4***	***PO 5***	***PO 6***	***PO 7***	***PO 8***	***PO 9***	***PO 10***	***PO 11***
CO 1	*	*									
CO 2				*	*						
CO 3			*	*							

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction to parallel computers	
Introduction, why parallel architecture, application trends, technology trends, architectural trends (chapter 1 of text 3(section 1.1))	At the end of the topic student should be able to: 1. Describe the instruction formats for I-type, R-type and J-type instructions of the DLX pipelined machine. Also provide one example each (c1) 2. Consider the 5 stages of the DLX pipelined instruction execution THREE instruction

	format. Indicate what actions take place in each stage for the above mentioned instructions (C2) 3. Discuss why multicore systems better than single core systems? Justify(c3) 4. Define the addressing modes for DLX instructions with example(c3)
Unit 2: Instruction Level Parallelism (ILP)	
(Chapter 3(section3.1) of text 2)	1. Write and define the DLX architecture (C4) 2. Define the performance issues in pipelines(C2) 3. What is meant by a data hazard? Explain. Also explain RAW, WAW and WAR hazards(C3) 4. What is meant by data hazard? Explain. Also describe how data hazard can be minimized by compiler scheduling(C3)
Unit 3: Cache memory	
(Chapter 5 of text2)	1. List and explain the 3 address mapping techniques. Provide one common example to each illustrate all techniques(C2) 2. Compare and contrast the 3 address mapping techniques. Which according to you is the best and why?(C4) 3. Explain direct mapping, set associative and the fully associative address mapping techniques. Provide one common example to illustrate all the 3 techniques(C4) 4. Define the significance of following terms i). write through ii) write allocate iii) no write allocate (C2)
Unit 4: Shared Memory Multiprocessors	
General architecture, Introduction to Interconnect, communication latency Problem of cache coherence; memory	1. Given the following data, determine the average memory access time and miss rate in each case? Which one has

consistency models: SC, PC, TSO, PSO, WO/WC, RC; snoopy protocol: invalidate vs. update, MSI,MESI, MOESI, MOSI; performance trade-offs; synchronization primitives: atomic primitives; locks: TTS, tickets, array; barriers: central and tree; performance implications in shared memory programs; chip multiprocessors: why CMP (Moore's law, wire delay); shared L2 vs. tiled CMP; core complexity; power/performance; (Chapter 5 of Text 3 and Intel manuals and other research papers)	the lower miss rate? A split 8 KB instruction cache with a 8 KB data cache? Or a 16 KB unified cache? Assume a hit takes 1 clock cycle and a miss penalty is 50 clk cycles. Also a load or store takes 1 extra clk cycle on a unified cache. For the 8 kB instruction, cache miss rate is 0.64% and for the 8 KB data cache it is 6.47% and for the unified cache it is 1.99%. (C5) 2. Define the write atomocity w.r.t. distributed shared multiprocessor system (C4) 3. Describe the performance trade off with MESI and MOSI protocols(C6) 4. Describe the performance trade off with MOESI and MESI protocols(C6)
Unit 5: Introduction to Basic optimization	
HotSpot, Faster Algorithms, ILP, Data Dependency, Branching, Memory, Loops, Slow Operations (Selected Topics in Chapter 1-14 of Text 1)	1. Define the software optimisation pitfalls(C3) 2. Define the attributes of benchmark with suitable example(C5) 3. Define the various issues which effects the memory performances(C3) 4. Describe with example the factors which effects the performances under loops (C4)
Unit 6: Introduction to Performance Tools (Intel Software Tools)	
Benchmark, Optimizing Compilers, Profilers, Performance Tools, Code Coverage Tools, Sampling vs Instrumentation	1. List miss rate reduction techniques and explain any 2 compiler optimization techniques in detail (C3) 2. What do you understand by hardware prefetching of data? How it is different from compiler controlled prefetching (C5) 3. Explain the 3 compiler optimization techniques(C3)
Unit 7: Introduction to Multicore Optimization	

ILP vs TLP, Data vs Task Parallelism, Parallel Application Case Studies, Parallelization Process (Chapter 2 of Text 3, Chapter 15 of Text 1)	1. Describe the difference between instruction level and task level parallelism with appropriate example(C2) 2. Explain the following (C2) (i) Shared memory versus distributed memory (ii) parallel computing versus serial computing 3. Explain, compare and contrast the following 4 terms: SISD, SIMD, MISD, MIMD(C3)
Unit 8:Programming for Performance	
Partitioning for performance, Data Access and Communication, Orchestration, Performance factors, Case-Studies (Chapter 3 of Text 3)	***1.*** Write the complete **state transition table** for **MSI** protocol. Let the table have the column like (i)Current state of the cache block (ii) Transaction generated or observed by the controller (iiI)Action by the controller (iv) New state of the cache block (iv)Data supplied by (C5) ***2.*** Explain how data can be partined for performance, which method best among all(C4) ***3.*** Assume a multiprocessor system consists of 3 processors, p1, p2, and p3, each having its own local cache, c1, c2, and c3, respectively. The coherence among the caches is maintained using MSI protocol. Describe the state transitions of all the three caches for the following memory operations on a memory block u. i. P2 reads U ii. P2 writes to u iii. P1 Reads to u iv. P1 writes to u v. P3 Writes to u vi. P3 reads u ***vii.*** P2 wites to u (C4)
Unit 9: MultiThreading with Open MP	

Threading, High Level vs Low Level Threading, Threading Goals and Issues, Introduction to OpenMP pragmas, Execution Model and Memory Model, Advanced OpenMP Topics (Chapter 16-17 of Text 1 and other material)	1.Explain the following openmp constructs with examples a) #pragma omp parallel b) #pragma omp for c) #pragma omp parallel shared(n) private (i) d) #pragma omp single e) #pragma omp section (C3) 2. . Explain the following openmp clauses with examples i. firstprivate ii. lastprivate iii. nowait iv. barrier v. critical (C4) 3. Explain the following openmp functions with examples. i. omp_get_thread_num() ii. omp_set_num_threads() iii. omp_get_num_threads() (C3)
Unit 10: Multithreaded Applications	
Some applications in Integer Programming, Digital Signal Processing (Video Codec) (Chapter 18 of Text 1and other material)	1. Describe comprehensively key features of various performance tools (C2) 2. Consider the water storage at dams, calculate the performance of the algorithms by considering the following factors -1. How much water will be stored during summer, rainy season, evaporated during summer and winter. Calculate the benchmarks, perfomace, identify the time consuming hotspots, find the mispredicted branches , loops, algorithmic issues. (C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04

Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2		*	*
Assignment/Presentation		*	*
End Semester Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. The Software Optimization Cookbook High Performance Recipes for IA-32 Platforms, Richard Gerber,Aart J. C Bik, Kevin B. Smith, and Xinmin Tian, 2nd Edition, Intel Press 2. JComputer Architecture: A Quantitative Approach, Morgan Kaufmann Publishers,. L. Hennesey andD. A. Patterson. 3rd Edition 3. Parallel Computer Architecture: A Hardware/Software Approach. Morgan Kaufmann Publishers, D.E. Culler, J. P. Singh, with A. Gupta, 2nd Edition

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	IT Project Management
Course Code: CSE 631	**Course Instructor:**
Academic Year: 2020 – 2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:** Familiarity in developing application using any high level language

Synopsis:	This Course provides insight on 1. The concept of software development process and project management 2. Illustrates the difference between a lab assignment and group project 3. Help the students to understand the finer points of Project management 4. Bring awareness about the processes, tools and techniques involved in the field of IT project management
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Illustrate the importance of project planning.
CO 2:	Discuss and demonstrate various tools applicable for different phases of the software project.
CO 3:	Illustrate the importance of Change management.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*									
CO 2		*	*								
CO 3	*		*								

Course content and outcomes:	
Content	***Competencies***
Unit 1: Software Project Planning	
Understand the Project Needs, Create the Project Plan, Diagnosing Project Planning Problems	At the end of the topic student should be able to: 1. Understand the project needs, necessity of plan, Define the Project Plan, Diagnosing Project Planning Problems (C1)
Unit 2: Estimation	

Elements of a Successful Estimate, Wideband Delphi Estimation, Other Estimation Techniques, Diagnosing Estimation Problems.	1. List the importance of estimation and describe different estimation techniques (C2) 2. Discuss the significance of Reviews and different review techniques (C2)
Unit 3: Project Schedules	
Building the Project Schedule, Managing Multiple Projects, Use the Schedule to Manage Commitments, Diagnosing Scheduling Problems.	1. Outline the steps in building project schedule.(C1) 2. Indicate mechanism of managing multiple projects. (C2)
Unit 4: Reviews	
Inspections, Deskchecks, Walkthroughs, Code Reviews, Pair Programming, Use Inspections to Manage Commitments, Diagnosing Review Problems.	1. Discuss the significance of Reviews and different review techniques (C2)
Unit 5: Software Requirements	
Requirements Elicitation, Use Cases, Software Requirements Specification, Change Control, Introduce Software Requirements Carefully, Diagnosing Software Requirements Problems	1. Introduce to requirement elicitation techniques, design and demonstrate the requirement documentation by field visits(C2)
Unit 6 : Design and Programming	
Review the Design, Version Control with Subversion, Refactoring, Unit Testing, Use Automation, Be Careful with Existing Projects, Diagnosing Design and Programming Problems	1. Illustrate the key steps in design and programming phase. Version control and unit testing significance (C3)
Unit 7: Software Testing	
Test Plans and Test Cases, Test Execution, Defect Tracking and Triage, Test Environment and Performance Testing, Smoke Tests, Test Automation, Postmortem Reports, Using Software Testing Effectively, Diagnosing Software Testing Problems	1. Define the test plans, significance of test phase and the test case characteristics. Introduce different types testing and significance of type of testing.(C2)
Unit 8: Understanding Change	

Why Change Fails, How to Make Change Succeed	1. Illustrate the necessity of Change management system – developing impact analysis document and its importance (C3).
Unit 9: Management and Leadership	
Take Responsibility, Do Everything Out in the Open, Manage the Organization, Manage Your Team	1. Understand the role of management in motivating the team, finer points of managing the team (C2)
Unit 10: Managing an Outsourced Project	
Prevent Major Sources of Project Failure, Management Issues in Outsourced Projects, Collaborate with the Vendor	1. Describe the differences of managing the outsourced project, typical point of conflicts(C2) 2. Review of the project management process (C2)
Unit 10: Process Improvement	
Life Without a Software Process, Software Process Improvement, Moving Forward	**1.** Analyse the projects without process and continuous process improvements initiatives needed for success of the project (C4)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3

Sessional Examination 1	*	*	
Sessional Examination 2	*		*
Assignment/Presentation	*	*	
End Semester Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. "Applied Software Project Management" By Jennifer Greene, Andrew Stellman (O'Reilly Publications) 2005. 2. "The Art of Project Management" By Scott Berkun (O'Reilly Publications) 2005.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Big Data and Data Visualization
Course Code: BDA 614	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First Year, Semester 1
No of Credits: 3	**Prerequisites:** Programing in Python or Java

Synopsis:	This Course provides insight on 1. This course aims to help students get started with Architectures of distributed file systems and distributed computing. 2. Students learn probability and statistical Inference techniques. 3. Students learn machine learning algorithms required for big data applications. 4. Students learn to map data attributes to graphical attributes, and strategic visual encoding based on known properties of visual perception.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Understand the architecture of distributed systems and distributed computing.
CO 2:	Identify the characteristics of datasets and compare the trivial data and big data for various applications.
CO 3:	Explain concept learning task and hypothesis space, distinguish between general and specific hypotheses, identify the maximally specific hypotheses, Describe version spaces and candidate elimination algorithm.
CO 4:	To solve problems associated with batch learning and online learning, and the big data characteristics such as high dimensionality, dynamically growing data and in particular scalability issues.
CO 5:	Practical experience building and evaluating visualization systems.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*								
CO 2	*	*	*								
CO 3	*	*	*	*							
CO 4	*	*	*								
CO 5	*	*	*				*				

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction to Big Data	
Terminology – Challenges - Architectures – Distributed File Systems – Google File System – Hadoop File Systems - Hadoop Ecosystems.	At the end of the topic student should be able to: 1. Describe architecture of Google file system. (C2) 2. Describe architecture of Hadoop systems. (C2)
Unit 2: Statistics	
Sampling Techniques - Data classification, Tabulation, Frequency and Graphic reporesentation - Measures of central value - Arithmetic mean, Geometric mean, Harmonic mean, Mode, Median, Quartiles, Deciles, Percentile - Measures of variation – Range, IQR, Quartile deviation, Mean deviation, standard deviation, coefficient variance, skewness, Moments & Kurtosis.	1. Define True Error of a hypothesis, ε-exhausted Version Space, PAC Learning and Agnostic Learning (C1). 2. Describe data sampling techniques. (C2)
Unit 3: Databases for Big Data	
Data science process – roles, stages in data science project – working with data from files – working with relational databases – exploring data – managing data – cleaning and sampling for modeling and validation – Big Table vs HBase introduction to NoSQL - HiveQL - Querying Data - Sorting And Aggregating, Map Reduce Scripts, Joins & Subqueries, HBase concepts-Advanced Usage, Schema Design, Advance Indexing.	1. Describe is Data Science. (C2) 2. Describe the characteristics of NoSQL. (C2) 3. Describe the principle of Map Reduce technique. (C2)
Unit 4: Machine Learning for Big Data	
Choosing and evaluating models – mapping problems to machine learning, evaluating clustering models,	1. Apply candidate-elimination algorithm to obtain most general and most specific

validating models – cluster analysis – K-means algorithm, Naïve Bayes – Memorization Methods – Linear and logistic regression – supervised and unsupervised learning - Issues regarding classification and prediction, Bayesian Classification, Classification by backpropagation, Classification based on concepts from association rule mining, Other Classification Methods, Classification accuracy.	hypotheses for the training examples. (C3) 2. Apply the concept of entropy and information gain to find the root node of the decision tree (C3). 3. Design a model using K-means classifier to predict how well products are accepted by the clients (C3).
Unit 5: Stream Computing in Big Data	
Introduction - Streaming Data – Sources – Difference between Streaming Data and Static Data. Overview of Large Scale Stream Processing Engines – Issues in Stream Processing - Phases in Streaming Analytics Architecture - Vital Attributes - High Availability – Low Latency – Horizontal Scalability-Fault Tolerance - Service Configuration and Management - Apache ZooKeeper - Distributed Stream Data Processing: Co-ordination, Partition and Merges, Transactions. Duplication Detection using Bloom Filters - Apache Spark Streaming Examples Choosing a storage system – NoSQL Storage Systems.	1. Understanding issues with stream processing in big data (C3). 2. Describe how big data systems achieve high availability and low latency. (C2) 3. Describe how Spark does in memory processing. (C3)
Unit 6: Security in Big Data	
Privacy – Identification of Anonymous People – Why Big Data Privacy is self-regulating? – Ethics – Ownership – Ethical Guidelines – Big Data Security – Organizational Security - Steps to secure big data – Classifying Data – Protecting – Big Data Compliance - HADOOP SECURITY DESIGN	1. Describe why Big Data Privacy is self-regulating. (C2) 2. Describe the steps to secure big data systems. (C2)
Unit 7: Data Visualization, Characterization – Data Wrangling	

Combining and Merging DataSets – Reshaping and Pivoting – Data Transformation – String Manipulation, Regular Expressions - DATA AGGREGATION, GROUP OPERATIONS ,TIMESERIES - GoupBy Mechanics – Data Aggregation – Groupwise Operations and Transformations – Pivot Tables and Cross Tabulations – Date and Time Date Type tools – Time Series Basics – Data Ranges, Frequencies and Shifting - WEB SCRAPING - Data Acquisition by Scraping web applications –Submitting a form - Fetching web pages – Downloading web pages through form submission – CSS Selectors - Data Visualization Tools	1. Understanding various formats of data. (C1) 2. Design programs to dynamically extract data from web. (C4) 3. Design programs to read data from various data sources. (C4) 4. Create visualization for time series data. (C4) 5. Create visualization for statistical distributions. (C4) 6. Create visualization for maps, Hierarchical data and network data. (C4)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5

Sessional Examination 1	*	*	*		
Sessional Examination 2			*	*	*
Assignment/Presentation	*	*	*	*	*
End Semester Examination	*	*	*	*	*
Laboratory examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. HADOOP: The definitive Guide, Tom White 4th edition, O Reilly Publication 2. Python for Data Analysis, Wes Mc Kinney, O Reilly Publication. 3. Practical Data Science with R, Nina Zumel, John Mount, Manning Publications. 4. Machine Learning, E. Alpaydin, MIT Press, 2010

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	High Level Digital Design
Course Code: EDA-601	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:**
Synopsis:	This Course provides insight on 1. To understand number representation and conversion between different representation in digital electronic circuits. 2. To analyze logic processes and implement logical operations using combinational logic circuits. 3. To understand characteristics of memory and their classification. 4. To understand concepts of sequential circuits and to analyze sequential systems in terms of state machines. 5. To understand concept of Programmable Devices, PLA, PAL, CPLD and FPGA and implement digital system using SystemVerilog. 6. To understand the AMBA bus protocol and types of buses
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Develop a digital logic and apply it to solve real life problems.
CO 2:	Analyse, design and implement combinational, sequential logic circuits.
CO 3:	Discuss different semiconductor memories.
CO 4:	Analyse digital system design using PLD.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1		*									
CO 2			*								
CO 3	*										
CO 4	*										

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction	
Review of Digital Design	At the end of the topic student should be able to: 1. Discuss number system in digital design. (C2) 2. Discuss Boolean algebra in digital design. (C2)

	3. Optimize the Boolean expression using k-maps. (C3)
Unit 2: Combinational circuits - Design steps	
Arithmetic Circuits - Full adder, Serial Adder, Adder/Subtractor, Ripple Carry Chain, Carry Look-Ahead adder, Carry Select Adder, ALU, Parity Generator, Comparator, Multiplier. PLA, PAL, PLD, CPLD, ROM, FPGA – Introduction	1. Design a combinational circuit for a given boolean expression (C5). 2. Discuss different types of combinational circuits like adders, multipliers and CPLD's. (C2)
Unit 3: Sequential circuits - Design steps	
Flip-flops, registers, counters.	1. Design sequential circuit using Flip-flops (C5)
Unit 4: Finite State Machines	
Introduction to FSMs, capabilities, minimization and transformation of sequential machines, Synchronous and asynchronous FSMs, Mealy and Moore machines, State assignment of synchronous sequential machines, Structure of sequential machines, Verification and testing of sequential circuits	1. Discuss Mealy and Moore machines (C2) 2. Design sequential circuit using Mealy and Moore machines (C5)
Unit 5: Verilog / System Verilog for design	
Verilog / System Verilog for design	1. Differentiate Verilog and System Verilog. (C4)
Unit 6: Introduction FPGA	
Introduction FPGA	2. Explain FPGA architecture. (C2)
Unit 7: Spartan III Architecture	
Spartan III Architecture	3. Discuss Spartan III Architecture. (C2)
Unit 8: Application on Digital Design	
FIFO Design [SNUG Paper], Cordic Algorithm [IEEE Paper]	1. Explain the working of FIFO (C2) 2. Explain cordic algorithm (C2)

Floating Point Arithmetic Blocks [IEEE Paper]: Floating point Addition, Floating point, subtraction, Floating point Multiplication, Floating point Division	3. Discuss different floating-point arithmetic operations (C2)
Unit 8: AMBA Bus Specification [ARM Specification]	
AMBA Bus Specification [ARM Specification]	1. Discuss different components of AMBA bus (C2) 2. Explain AHB and APB (C2)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*		
Sessional Examination 2			*	
Assignment/Presentation		*		*
End Semester Examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• "An Engineering Approach to Digital Design" , Flectcher • "SystemVerilog for design by Stuart Sutherland" , Simon Davidmann, Peter Flake • SNUG Paper [freely available] • IEEE Paper [MU campus available] • ARM Specification.

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Entrepreneurship
Course Code: ENP-601	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 3	**Prerequisites:**
Synopsis:	This course introduces students to the theory of entrepreneurship and its practical implementation. It focuses on different stages related to the entrepreneurial process, including business model innovation, monetization, small business management as well as strategies that improve performance of new business ventures. Cantered on a mixture of theoretical exploration as well as case studies of real-world examples and guest lectures, students will develop an understanding of successes, opportunities and risks of entrepreneurship. This course has an interdisciplinary approach and is therefore open to students from other Majors.
Course Outcomes (COs):	On successful completion of this course, students will be able to:
CO 1:	To impart knowledge on the basics of entrepreneurial skills and competencies to provide the participants with necessary inputs for creation of new ventures.
CO 2:	To familiarize the participants with the concept and overview of entrepreneurship with a view to enhance entrepreneurial talent
CO 3:	To appraise the entrepreneurial process starting with pre-venture stage
CO 4:	To Create and exploit innovative business ideas and market opportunities
CO 5:	To Build a mind-set focusing on developing novel and unique approaches to market opportunities
CO 6:	To explore new vistas of entrepreneurship in 21st century environment to generate innovative business ideas through case studies.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*										
CO 2				*							
CO 3			*								
CO 4						*					
CO 5								*			
CO 6										*	

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction to Entrepreneurship	
Meaning and Definition of Entrepreneurship-Employment vs Entrepreneurship, Theories of Entrepreneurship, approach to entrepreneurship, Entrepreneurs VS Manager	1. Explain the meaning of Entrepreneurship (C1) 2. Discuss the theories of Entrepreneurship (C1) 3. Discuss the approaches to Entrepreneurship (C1)
Unit 2: Entrepreneurial Traits	
Personality of an entrepreneur, Types of Entrepreneurs	1. Discuss the Personality traits of entrepreneurs. (C2)
Unit 3: Process of Entrepreneurship	
Factors affecting Entrepreneurship process	1. Identify the fundamentals and responsibilities of entrepreneurship (C2) 2. Exemplify one's capabilities in relation to the rigors of successful ventures (C3) 3. Identify and differentiates the different characteristics and competencies of an entrepreneurs (C2)
Unit 4: Business Start-up Process	
Idea Generation, Scanning the Environment, Macro and Micro analysis	1. Explain the Process of Business start up (C1) 2. Develop creativity and critical thinking in identifying opportunities (C5) 3. Apply innovative approaches in envisioning ones entrepreneurial career (C3)
Unit 5: Business Plan writing	
Points to be considered, Model Business plan	1. Identify different business models (C3) 2. Describe different parts of a business plan(C2)
Unit 6: Case studies	
Indian and International Entrepreneurship	1. Perform self-assessment and analyse entrepreneurial personal traits and competencies (C4) 2. Evaluate oneself and plan courses of action to help develop one's entrepreneurial characteristics and competencies. (C5)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	30	60
Quiz	02	04
Small Group Discussion (SGD)	02	02
Self-directed learning (SDL)	-	04
Problem Based Learning (PBL)	02	04
Case Based Learning (CBL)	-	-
Revision	02	-
Assessment	06	-
TOTAL	**44**	**74**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos						
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
Sessional Examination 1	*	*				
Sessional Examination 2			*	*		
Assignment/Presentation					*	*
End Semester Examination	*	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. NVR Naidu and T. Krishna Rao, "Management and Entrepreneurship", IK International Publishing House Pvt. Ltd 2008. 2. Mohanthy Sangram Keshari, "Fundamentals of Entrepreneurship", PHI Publications, 2005 3. Butler, D. (2006). Enterprise planning and development. USA: Elsevier Ltd. Gerber, M.E. (2008) Awakening the entrepreneur within. NY: Harper Collins.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Digital Signal Processing Lab
Course Code: ESD 603L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites:** Knowledge of Signals and Systems and Basic Knowledge of Matlab
Synopsis:	This Course provides insight on 1. Understanding of basics of Signal and Systems as pre-requisite. 2. Understanding the concepts of Fast Fourier Transforms. 3. Learning hardware implementation of systems. 4. Learning FIR and IIR Filter Designs. 5. Learning concepts of multi-rate signal processing in the form of sampling rate conversion, structures of sampling rate converters and some applications of sampling rate converters 6. Understanding three optimum Weiner filters, adaptive algorithm and transforming Weiner filters in to adaptive filters 7. Understanding architecture, memory management and pipelining concepts of TMS320C67XX processor through self-stud.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Use matlab to implement various DSP techniques. (C3)
CO 2:	Experiment DFT, LTI techniques and analyse the results. (C4)
CO 3:	Design FIR, Butterworth and Chebychev filters in matlab. (C5)

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*		*	*						
CO 2	*	*			*						
CO 3	*	*		*	*						

Course content and outcomes:	
Content	***Competencies***
Unit 1:	
Write matlab programs to Generate waves	At the end of the topic student should be able to:

Write matlab programs to Addition of two sequences Write matlab programs to Find convolution of two sequences and verify the result using built-in function User defined Matlab function to find convolution of two sequences and verify the result	1. Use Matlab to generate waves.(C3) 2. Use Matlab for addition of two sequences.(C3) 3. Compute convolution of two sequences using Matlab. (C3) 4. Analyse the convolution usinf built in functions. (C4) 5. Practice convolution user defined function in Matlab (C3)
Unit 2:	
Write matlab programs to Find DTFT of a sequence. Write matlab programs to Find DFT of a sequence and verify using built-in function User defined Matlab function to find DFT and verify the result Write matlab programs to Find convolution of two sequences using DFT Write matlab programs to Find the time response of an LTI system defined by either difference equation or transfer function	1. Experiment DTFT of a sequence using Matlab (C4) 2. Analyse the DFT of a sequence with built in function (C4) 3. Experiment DFT using Matlab (C4) 4. Compute convolution of two sequence using DFT in Matlab. (C3) 5. Experiment time response of an LTI system in Matlab (C4)
Unit 3:	
Write Matlab programs to find DFT using DIT-FFT and DIF-FFT algorithms, compare the result using built in function. Design FIR filters with frequency domain specification (LP, HP, BP and BR) using Frequency Sampling Technique and verify frequency response.	1. Analyse DIT-FFT and DIF-FFT algorithms. (C4) 2. Design FIR filters with frequency domain specifications. (C5)

Design FIR filter to meet required impulse response using Frequency Sampling Technique.	
Unit 4:	
Write Matlab programs to Design FIR filters with frequency domain specification (LP, HP, BP and BR) using different window functions and verify frequency response. Design analog Butterworth and Chebychev filters using built-in functions, transform them to digital filter and verify their frequency response (C2). Design digital Butterworth and Chebychev filters using built-in functions verify the frequency response (C2)	1. Design FIR filters with frequency domain specifications. (C5) 2. Design analog Butterworth and Chebychev filters using built-in functions. (C5) 3. Design digital Butterworth and Chebychev filters using built-in functions. (C5)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO1	CO2	CO3
Sessional Examination 1	*	*	
Assignment/Presentation			*
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• "Digital Signal Processing", Sanjith K Mitra • "Digital Signal Processing", Oppenheim and Schafer • "Digital Signal Processing", Roman Kuc • "Digital Signal Processing", Proakis and Manolakis • "Digital Signal Processing ", Rabinder and Gold • Shaum Out-Line Series • "Signals and Systems", Symon Haykins • DSP Processors and Fundamentals • "Multirate signal processing", Vaidyanathan • "Handbook of DSP", Elliot

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Device Drivers Lab
Course Code: ESD 604L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites:** Basic C Programming
Synopsis:	This Course provides insight on 1. Insight into Linux kernel programming. 2. Knowledge about the framework used in building the Linux device driver. 3. Concept of designing proc and ioctl needed to build a device driver 4. Techniques to debug kernel programs 5. Insight into designing USB drivers
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Understand basic Linux kernel programming with an introduction to kernel modules
CO 2:	Understand the concept of file operation with implementation of open, close, read, write system calls
CO 3:	Implement proc entries
CO 4:	Implementation of ioctls
CO 5:	Use tools to debug the kernel modules

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*			*						
CO 2		*			*						
CO 3	*		*								
CO 4		*	*								
CO 5	*		*		*						

Course content and outcomes:	
Content	***Competencies***
Unit 1:	
Introduction to Device Drivers	At the end of the topic student should be able to:

	1. Describe the broad design of device driver (C3)
Unit 2:	
Building & Running Modules.	1. Compile and load modules using a make file (C4)
Unit 3:	
Character Driver.	1. Explain the structure of a character driver (C3)
Unit 4:	
Debugging Techniques.	3. Debug modules using prink, proc and kdb (C4) 4. Design of Ioctl used in building device drivers (C5)
Unit 5:	
Concurrency and Race Condition	3. Illustrate the problems associated with concurrent device drivers (C3) 4. Describe the problems associated with race condition while designing a device driver (C3)
Unit 6:	
Advanced Character Driver Operations	1. Execute bottom half through deferred work (C4)
Unit 7:	
Communicating with Hardware	1. Communicate with the devices through I/O ports (C4)
Unit 8:	
Interrupt Handling	**1.** Illustrate the concept of writing interrupt handlers (C4)
Unit 9:	
PCI Drivers, USB Drivers	3. Structure of a USB driver (C4) 4. Design a USB driver. (C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*

Lecture	36	72
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-
Clinic	-	-
Practical	36	72
Revision	-	-
Assessment	6	-
TOTAL	**78**	**144**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Sessional Examination 1	*	*			
Sessional Examination 2			*	*	
Assignment/Presentation		*		*	*
Laboratory Examination	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Alessandro Rubini**, "Linux Device Drivers"**, (Nutshell Handbook), O'Reilly Publishers, 2009. 2. John Madieu, "**Linux Device Drivers Development: Develop customized drivers for embedded Linux"**, Packt Publishing, 2017. 3. Robert Love, **"Linux Kernel Development**", Addison Wesley, Third Edition, 2010. 4. Daniel P. Bovet, Marco Cesati, **"Understanding the Linux Kernel"**, O'Reilly Media, Third Edition, 2008.

	5. Wolfgang Mauerer, **"Professional Linux Kernel Architecture"**, Wrox, 2008. 6. Sreekrishnan Venkateswaran, **"Essential Linux Device Drivers"**, Prentice Hall, 2008. 7. W. Richard Stevens, Stephen A. Rago, **"Advanced Programming in the UNIX Environment"**, Addison Wesley, Third Edition, 2013. 8. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, **"Unix Network Programming, Vol1: Sockets",** Pearson Education India, Third Edition, 2015.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Embedded Systems Lab
Course Code: ESD 605L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites:** Microprocessor architecture , Microcontroller Architecture , Assembly language and Number systems

Synopsis:	This Course provides insight on 1. This course provides the knowledge of ARM Cortex M3 Processor architecture. 2. This course provides the knowledge of Microcontroller based on ARM Processor architecture and its Registers and Instruction sets to write Assembly and Embedded C Programming. 3. This course provides the concept of Interfacing and Programming Sensors and Peripherals to Microcontrollers. 4. This course provides the concept of Real time operating systems on Microcontrollers.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Illustrate the features of embedded systems, architecture of ARM7, Instruction set and development tools of ARM.
CO 2:	Experiment the architectural features of LPC13/17XX microcontrollers, interfacing peripheral devices to LPC2148.
CO 3:	Design a Real time Embedded Systems by interfacing Sensors and Actuators and porting Real time operating systems.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*		*						
CO 2	*	*			*						
CO 3	*	*	*		*						

Course content and outcomes:

Content	***Competencies***
Unit 1: Introduction to LPC13/17xx Microcontroller	
Introduction to LPC13/17xx Microcontroller - Hardware, SW.	At the end of the topic student should be able to:

	1. Summarise LPC13/17xx Microcontroller architecture and developmen t tools of ARM. (C2)
Unit 2: Interfacing LPC13/17xx Microcontroller	
Interfacing With LED, LCD Seven Segment Display, UART, HEX Keypad.	Experiment interfacing LPC13/17xx Microcontroller with I/O devices. (C2)
Unit 3:	
Introduction to Free RTOS, FreeRTOS API Calls, Task Creation, Queques, semaphore, mutex, RTOS application development.	1. Summarise FreeRTOS architecture. (C2) 2. Practise different API call in FreeRTOS. (C2) 3. Design a Real time Embedded system by writing applications on top of Real time operating systems (C5)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	*
Assignment		*	*
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. Joseph Yiu, "The definitive guide to the ARM Cortex-M3", Elsevier, 2nd Edition, 2010. 2. Frank Vahid, Tony Givargis, "Embedded System Design: A Unified Hardware/Software Introduction", Wiley India, ISBN:81-265-0837-X, 2007. 3. Richard Barry, "NXP Semiconductors, LPC13xx/17xx User Manual", 2012. 4. NXP Semiconductors, "LPCzone Examples", 2012. 5. "FreeRTOS Reference Manual", Real Time Engineers Ltd., 2016.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Embedded Software Design Lab
Course Code: ESD 606L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites:** Microprocessor architecture, Microcontroller Architecture , Assembly language, Object oriented concepts

Synopsis:	This Course provides insight on 1. Students learn the concept of big data characteristics, batch and lambda architecture. 2. This course introduces students to basics file systems in Big Data 3. This course helps the student to understand the concepts of Hadoop framework, Spark framework and their internals. 4. This course helps the students to learn Map-reduce programming, Spark programming. 5. Students learn the different layers with use cases demonstrations.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Experiment with UML tools and represent the model using suitable diagrams.
CO 2:	Develop applications using the OOP concepts
CO 3:	Develop applications using JAVA constructs for general purpose and embedded systems.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*	*	*						
CO 2	*	*	*	*	*						
CO 3	*	*	*	*	*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Modelling object-oriented systems:	

UML for modelling object-oriented systems- Class diagrams- Collaboration diagrams-Sequence diagrams- State diagrams.	1. Building models using UML tools. (C3)
Unit 2: Modelling real-time embedded systems behaviours	
UML real-time profile.	1. Review and understand the UML real-time profile. (C3).
Unit 3: Developing object-oriented systems in Java	
Classes, interfaces, methods- Generics-Scope rules and access control. Inner classes-Functional programming constructs – lambdas-Threads, concurrency control and timers-I/O, Streams and network I/O-Security and Cryptography.	1. Design Classes, interfaces, methods.(C3). 2. Explain the Generics-Scope rules and access control.(C3). 3. Explain and illustrate Functional programming constructs.(C3). 4. Explain and lambdas.(C3). 5. Explain and illustrate the Threads, concurrency control and timers-I/O, Streams and network concepts.(C3). 6. Explain and illustrate the I/O-Security and Cryptography.(C4).

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	*
Assignment		*	*
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition), Addison-Wesley Professional. 2003. 2. The Java Programming Language. Ken Arnold, James Gosling and David Holmes. Addison-Wesley Professional; 4 edition (August 27, 2005) 3. http://www.oracle.com/technetwork/java/embedded/resources/tech/compact-profiles-overview-2157132.html 4. Realtime Specification for Java 2.0 https://java.net/projects/rtsj-2/pages/Home

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Mobile Application Development using Android Lab
Course Code: CSE 605L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites:** Basic Android Programming
Synopsis:	This Course provides insight on 1. This course would provide fundamental knowledge about android platform. 2. The course will also provide skill sets to design and develop android applications for mobile devices. 3. This course will provide basic knowledge about android application communication of data which are hosted in remote systems.
Course Outcomes (COs):	On successful completion of this course, students will be able to

CO 1:	Use of major building blocks in an android application
CO 2:	Solve different issues associate with design of android applications
CO 3:	Write android applications using various UI components and data handling using SQLilte
CO 4:	Experiment advanced topics on android applications

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*		*								
CO 2	*	*	*	*	*						
CO 3			*		*						
CO 4		*		*							

Course content and outcomes:	
Content	***Competencies***
Unit 1: Part - 1: Installation of Android Studio	
Installation of Android Studio, environment setting, Project creation, building a project, running a sample project	At the end of the topic student should be able to: 1. Identify different features in android studio (C1) 2. Explain Android manifest file (C2) 3. Discuss DVM, DDMS, android emulator (C2) 4. Discuss the issues related running and debugging applications (C2)
Unit 2: Introduction to Android Screen UI Components	
Implementation of android applications using various android UI components and layouts	1. Practice by creating android applications using different types of layouts (C3) 2. Develop android applications using different types of views such as Listview, spinner, time picker and date picker (C3) 3. Illustrate the use of Gallery and ImageSwitcher views (C2)
Unit 3: Introduction to Data Management with SQLite	
Develop android applications for data handling	1. Implement android applications for content provider (C3)

	2. Apply shared preferences concept to android UI screen (C3) Apply CRUD operations to develop a simple healthcare application (C3)
Unit 4: Advanced topics	
Adapters, background threads, Notifications, Location based services, Mapping, network connectivity services, telephony services	1. Practice to generate notification object in an android application (C3) 2. Apply Location based services in android applications (C3) 3. Demonstrate android service life cycle in an android application (C3) 4. Understand the use of background threads in android applications (C3)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos

Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*	*	
Assignment		*	*	
Laboratory Examination	*	*	*	

Feedback Process	• End-Semester Feedback
Reference Material	• *Sams Teach Yourself Android Application Development in 24 Hours , Lauren Darcey and Shane Conder , ISBN-10: 0321673352 | ISBN-13: 978-0321673350 | Edition: 1* • *Hello, Android: Introducing Google's Mobile Development Platform, Ed Burnette, ISBN-10: 1934356565 | ISBN-13: 978-1934356562 | Edition: Third Edition* • *Android Application Development: Programming, Rick Rogers and John Lombardo, ISBN-10: 0596521472 | ISBN-13: 978-0596521479 | Edition: 1* • *Professional Android 2 Application Development (Wrox Programmer to Programmer) , Reto Meier , ISBN-10: 0470565527 | ISBN-13: 978-0470565520 | Edition:*

Name of the Program:		Master of Engineering - ME (Embedded Systems)
Course Title:		Web Application Development Lab
Course Code: CSE-611L		**Course Instructor:**
Academic Year: 2020-2021		**Semester:** First Year, Semester 2
No of Credits: 1		**Prerequisites:** Basic C , Fundamentals of Object Oriented Programming
Synopsis:	This Course provides insight on 1. This course provides an insight into web applications. 2. This course provides knowledge about databases and their use in developing web applications. 3. This course introduces ASP.NET technology and its use to build web applications. 4. This course provides knowledge about Client-Server Architecture. 5. This course provides knowledge about HTML, CSS, JavaScript 6. This course provides knowledge about Web Service.	

Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Construct a database and normalize the database.
CO 2:	Manipulate the appropriate types of data to be stored.
CO 3:	Develop a Web application using ASP.NET.
CO 4:	Construct a web page using HTML and CSS.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1		*	*		*						
CO 2		*	*		*						
CO 3		*	*		*						
CO 4		*	*		*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Database Concepts	
Introduction to Transact SQL, Database optimization Techniques (Normalization), Creating database, Queries, sub queries, Joins, Stored Procedures, Triggers Tools: SQL server 2008 R2	At the end of the topic student should be able to: 1. Design and develop a database (C5) 2. Create stored procedures, Triggers to manipulate data in database (C5).
Unit 2: Programming	
ASP.NET: Introduction to ASP.NET, Clint server and web application design of presentation, business logic	1. Use HTML to create webpages (C3) 2. Use CSS and JavaScript on web pages (C3) 3. Develop web applications using ASP.Net (C3) 4. Develop forms using WIN forms (C3).

and storage functionality. WIN form and WEB form Designs using ASP.NET, Silverlight, Windows Client, HTML basics, CSS, AJAX, Java Scripts, Styling with Themes, Componentization (Code behind, Data Layers, User Controls), Roles and Profiles Tool: Visual Studio 2010	
Unit 3: Web Services	
The life Cycle of a Web Service, Structure of Web Service, Creating a Web Service, SOAP, Data Contracts, Binding, Security, Discovery, Publishing, WSDL	1. Analyse Web services (C4). 2. Write SOAP, Data Contracts (C3).

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos				
Nature of assessment	CO 1	CO 2	CO 3	CO 4
Sessional Examination 1	*	*		
Sessional Examination 2			*	*
Assignment/Presentation	*	*	*	*
Laboratory examination	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• Beginning ASP.NET 4.0 with C# byChris Hart, John Kauffman, David Sussman, Chris Ullman • Professional C# - Simon Robinson, Christian Nagel- Wiley Publishing, Inc.

Name of the Program:		Master of Engineering - ME (Embedded Systems)
Course Title:		Multicore Program Optimization Lab
Course Code: CSE 612 L		**Course Instructor:**
Academic Year: 2020-2021		**Semester:** First Year, Semester 2
No of Credits: 1		**Prerequisites:** Basic Programming – preferably C, concepts of thread creation
Synopsis:	This Course provides insight on 1. Students able to understand various software tools, usages. 2. Able to understand difference between single core and multicore systems execution environment 3. Students will understand apple appropriate pragmas and its implementation	

	4. Develop the optimised software model based on application 5. Design and illustrate the performances of parallel threads and under the mutlicor environments.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Understand knowledge of usage of software's
CO 2:	Create multicore thread environmental based scenario with openmp pragmas
CO 3:	Construction of circuit models based on application, tools

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*									
CO 2			*		*						
CO 3			*		*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Basic C programming	
Understanding of basic c programming for large set of data and understanding time consumtion	At the end of the topic student should be able to: C programs: 1. WAP for matrix multiplication of order 1000x1000 (C3) 2. WAP for function for random number generation and calculate the time need for execution on single core and dual core systems (C3). 3. WAP for multiplying multiple arrays by a constant by even indices and odd indices (C3)
Unit 2: Introduction OPENMP pragmas	
Understanding various openmp pragma's, its syntax's	1. Understand the syntax's of openmp constructs, clauses, functions(c2,c3)
Unit 3: Introduction of THREADS AND PRAGRMA'S	
Writing code using multiple threads using openmp pragma's and comparisons of time consumptions	1. Design and implement multithreaded programs using openmp constructs, clauses, functions (C6)

with single and multicore environments.	2. Design and implement multithreaded program that finds th repetition of a number within a nxn matrix. Initialisation of matrix should be done within the parallel region, but by one thread(c5) 3. Design and implement multithreaded program that finds th repetition of a number within a nxn matrix. Initialisation of matrix should be done within the parallel region, but by one thread. The number to be searched shuld be input by user before entering the parallel region (c6) 4. Design and implement multithreaded program to compute the sum of two nxn matrices. The initialisation of matrix should be done within the parallel region, but by only thread.
Unit 4: understanding the optimisation tools	
Understanding the various tools which are available from various industry and internet sources	1. Running the sample code for application of Benchmark, Optimizing Compilers, Profilers (c4) Performance Tools, Code Coverage Tools, Sampling vs Instrumentation (c6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-

TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2			*
Assignment/Presentation			*
Laboratory examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. The Software Optimization Cookbook High Performance Recipes for IA-32 Platforms, Richard Gerber, Aart J. C Bik, Kevin B. Smith, and Xinmin Tian, 2nd Edition, Intel Press 2. JComputer Architecture: A Quantitative Approach, Morgan Kaufmann Publishers,. L. Hennesey and D. A. Patterson. 3rd Edition 3. Parallel Computer Architecture: A Hardware/Software Approach. Morgan Kaufmann Publishers, D. E. Culler, J. P. Singh, with A. Gupta, 2nd Edition 4. INTEL COOK BOOK and Manuals, IEEE Papers 5. openmp reference guide version 4.0/4.5 6. Internet sources

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	IT Project Management Lab
Course Code: CSE-631L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites:** Familiarity in developing application using any high level language
Synopsis:	This Course provides insight on 1. The concept of software development process and project management 2. Illustrates the difference between a lab assignment and group project 3. Help the students to understand the finer points of Project management 4. Bring awareness about the processes, tools and techniques involved in the field of IT project management.

Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Practice the project development through project planning.
CO 2:	Understand the finer points of Project management.
CO 3:	Bring awareness about the processes, tools and techniques involved in the field of IT project management.

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1			*	*							
CO 2					*				*		
CO 3			*		*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Software Project Planning	
Understand the Project Needs, Create the Project Plan, Diagnosing Project Planning Problems.	At the end of the topic student should be able to: 1. Discussion on tools needed for project management (C3)
Unit 2: Estimation	
Elements of a Successful Estimate, Wideband Delphi Estimation, Other Estimation Techniques, Diagnosing Estimation Problems.	1. Download and demonstrate the tools typically used for UML design. (C3)
Unit 3: Project Schedules	
Building the Project Schedule, Managing Multiple Projects, Use the Schedule to Manage Commitments, Diagnosing Scheduling Problems.	1. Design the application through the UML tool practiced (C4) 2. Develop the team with different roles assigned to each member – namely project manager, developer, tester and assign appropriate tasks (C4)
Unit 4: Reviews	
Inspections, Deskchecks, Walkthroughs, Code Reviews, Pair Programming, Use Inspections to	1. Develop basic set of programs and to illustrate the unit tests (C2)

Manage Commitments, Diagnosing Review Problems.	
Unit 5: Software Requirements	
Requirements Elicitation, Use Cases, Software Requirements Specification, Change Control, Introduce Software Requirements Carefully, Diagnosing Software Requirements Problems	1. Field visit to develop and practice the requirement elicitation (C3)
Unit 6: Design and Programming	
Review the Design, Version Control with Subversion, Refactoring, Unit Testing, Use Automation, Be Careful with Existing Projects, Diagnosing Design and Programming Problems	1. Illustrate the key steps in design and programming phase. Version control and unit testing significance (C3) 2. Review of various artefacts generated by project and revise the project management methodology to the team (C5)
Unit 7: Software Testing	
Test Plans and Test Cases, Test Execution, Defect Tracking and Triage, Test Environment and Performance Testing, Smoke Tests, Test Automation, Postmortem Reports, Using Software Testing Effectively, Diagnosing Software Testing Problems	1. Inter team testing set up based on requirement document(C5)
Unit 8: Understanding Change	
Why Change Fails, How to Make Change Succeed	1. Illustrate the necessity of Change management system – SVN hands on (C3).
Unit 9: Management and Leadership	
Take Responsibility, Do Everything Out in the Open, Manage the Organization, Manage Your Team	1. Discussion on the topic with the help of case study (C3)
Unit 10: Managing an Outsourced Project	
Prevent Major Sources of Project Failure, Management Issues in Outsourced Projects, Collaborate with the Vendor	2. Discussion on the topic with the help of case study (C3)
Unit 11: Process Improvement	

Life Without a Software Process, Software Process Improvement, Moving Forward	1. Post-mortem report generation of respective project by each team – review of the report and suggest areas of improvement (C4)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2			*
Assignment/Presentation	*		
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. "Applied Software Project Management" By Jennifer Greene, Andrew Stellman (O'Reilly Publications) 2005. 2. "The Art of Project Management" By Scott Berkun (O'Reilly Publications) 2005.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Big Data and Data Visualization Lab
Course Code: BDA 614L	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First year, Second semester
No of Credits: 1	**Prerequisites: Programming in Python, Java**
Synopsis:	This Course provides insight on 1. Students learn to handle big data in distributed computing architecture. 2. Installation and working on Hadoop and ecosystem 3. Build machine learning Models 4. Processing of data stream 5. Choose proper data visualization techniques
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Handle big data using Hadoop and its ecosystems.

CO 2:	Building machine learning algorithm using Spark.
CO 3:	Data Cleaning and Data Visualization.

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*	*	*		*	*					
CO 2	*	*	*	*	*	*			*	*	
CO 3	*	*	*	*	*	*		*			

Course content and outcomes:	
Content	***Competencies***
Unit 1: Big Data	
Introduction to Hadoop. Data Analysis using Hadoop ecosystems	At the end of the topic student should be able to: 1. Installation of Hadoop and Spark distributed systems. (C4) 2. Reading and writing data into HDFS (C2). 3. Develop scripts to transfer structured data from SQL database to HDFS. (C3) 4. Develop script to query the data from HDFS using Hive. (C4)
Unit 2: Machine Learning	
Machine Learning in Big Data. Stream processing in Big Data.	6. Design a model using K-means classifier to predict how well products are accepted by the clients (C4). 7. Develop applications using Stream processing in big data (C4).
Unit 3: Data Visualization	
Video encoding and processing techniques.	1. Design programs to dynamically extract data from web. (C4) 2. Develop visualization application for time series data. (C4) 3. Develop visualization application for statistical distributions. (C4) 4. Develop visualization application for maps, Hierarchical data and network data. (C4)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2		*	*
Assignment/Presentation	*	*	*
End Semester Examination	*	*	*
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. T. Hastie, R. Tibshirani and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer, 2nd Edition, 2009 2. Machine Learning for Big Data, Jason Bell, Wiley Big Data Series

	3. Big Data: Principles and best practices of scalable real-time data systems - Nathan Marz and James Warren. Manning Publisher. 4. Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale – Tom White, O'Reilly Publication 4th Edition. 5. Spark: The Definitive Guide: Big Data Processing Made Simple – Bill Chambers, Matei Zaharia, O'Reilly Publication 1st Edition

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	High Level Digital Design Lab
Course Code: EDA-601L	**Course Instructor:**
Academic Year: 2020-2021	**Semester:** First Year, Semester 1
No of Credits: 1	**Prerequisites:** Boolean Logic
Synopsis:	This Course provides insight on 1. To analyze logic processes and implement logical operations using combinational logic circuits and implement digital system using SystemVerilog. 2. To understand characteristics of memory and their classification and implement digital system using SystemVerilog. 3. To understand concepts of sequential circuits and to analyze sequential systems in terms of state machines and implement digital system using SystemVerilog.

	4. To understand concept of Programmable Devices, PLA, PAL, CPLD and FPGA and implement digital system using SystemVerilog. 5. To understand the AMBA bus protocol and types of buses and implement digital system using SystemVerilog
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Design and implement combinational circuits.
CO 2:	Design and implement sequential logic circuits.
CO 3:	Design and implement AMBA Bus protocol.

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1			*		*						
CO 2			*		*						
CO 3			*		*						

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction	
	At the end of the topic student should be able to: 1. Implement boolean expression using dataflow modelling. (C1)
Unit 2: Combinational circuits - Design steps	
	1. Experiment combinational circuits like adders, multipliers and CPLD's using SystemVerilog. (C4)
Unit 3: Sequential circuits - Design steps	
	1. Experiment sequential circuit using SystemVerilog (C4)
Unit 4: Finite State Machines	
	1. Experiment Mealy and Moore machines using SystemVerilog (C4)
Unit 5: Verilog / System Verilog for design	
	1. Differentiate Verilog and System Verilog. (C4)
Unit 6: Introduction FPGA	

	1. Experiment combinational and sequential circuits on Vertex-5 FPGA.
Unit 7: Spartan III Architecture	
	1. Experiment combinational and sequential circuits on Spartan III. (C4)
Unit 8: Application on Digital Design	
	1. Experiment FIFO using SystemVerilog (C4)
Unit 8: AMBA Bus Specification [ARM Specification]	
	1. Experiment AHB and APB using SystemVerilog (C4)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3

Sessional Examination 1	*	*	
Sessional Examination 2		*	
Assignment/Presentation			*
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	• "An Engineering Approach to Digital Design" , Flectcher • "SystemVerilog for design by Stuart Sutherland" , Simon Davidmann, Peter Flake • SNUG Paper [freely available] • IEEE Paper [MU campus available] • ARM Specification.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	**Entrepreneurship Lab**
Course Code: ENP-601L	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites: -**
Synopsis:	This Course provides insight on This course introduces students to the theory of entrepreneurship and its practical implementation. It focuses on different stages related to the entrepreneurial process, including business model innovation, monetization, small business management as well as strategies that improve performance of new business ventures. Cantered on a mixture of theoretical exploration as well as case studies of real-world examples and guest lectures, students will develop an understanding of successes, opportunities and risks of

	entrepreneurship. This course has an interdisciplinary approach and is therefore open to students from other Majors.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Understand the concept of entrepreneurship
CO 2:	To appraise the entrepreneurial process starting with pre-venture stage through group discussion
CO 3:	To Build a mind-set focusing on developing novel and unique approaches to market opportunities by considering case studies and understand the complete flow of entrepreneurship

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*					*		*			
CO 2						*					
CO 3								*		*	

Course content and outcomes:	
Content	***Competencies***
Unit 1: Introduction to Entrepreneurship	
Meaning and Definition of Entrepreneurship-Employment vs Entrepreneurship, Theories of Entrepreneurship, approach to entrepreneurship, Entrepreneurs VS Manager	1. Discuss the theories of Entrepreneurship (C1) 2. Discuss the approaches to Entrepreneurship (C1)
Unit 2: Process of Entrepreneurship	
Factors affecting Entrepreneurship process	1. Exemplify one's capabilities in relation to the rigors of successful ventures (C3) 2. Identify and differentiates the different characteristics and competencies of an entrepreneurs (C2)
Unit 3: Business Plan writing	
Points to be considered, Model Business plan	1. Identify different business models (C3) Describe different parts of a business plan(C2)

Unit 4: Case studies	
Indian and International Entrepreneurship	1. Perform self-assessment and analyse entrepreneurial personal traits and competencies (C4) 2. Evaluate oneself and plan courses of action to help develop one's entrepreneurial characteristics and competencies. (C5)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	12	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	-	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	03	-
Clinic	-	-
Practical	24	-
Revision	03	-
Assessment	06	-
TOTAL	**48**	**-**

Assessment Methods:	
Formative:	**Summative:**
Internal practical Test	Sessional examination
Theory Assignments	End semester examination
Lab Assignment & Viva	Viva

Mapping of assessment with Cos			
Nature of assessment	CO 1	CO 2	CO 3
Sessional Examination 1	*	*	
Sessional Examination 2			*
Assignment/Presentation		*	*
Laboratory Examination	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	1. NVR Naidu and T. Krishna Rao, "Management and Entrepreneurship", IK International Publishing House Pvt. Ltd 2008. 2. Mohanthy Sangram Keshari, "Fundamentals of Entrepreneurship", PHI Publications, 2005 3. Butler, D. (2006). Enterprise planning and development. USA: Elsevier Ltd. Gerber, M.E. (2008) Awakening the entrepreneur within. NY: Harper Collins.

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Mini Project - 2
Course Code: ESD 696	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 4	**Prerequisites:** Any programming language and circuit basics
Synopsis:	Students are expected to select a problem in the area of their interest and the area of their specialization that would require an implementation in hardware / software or both in a semester
Course Outcomes (COs):	On successful completion of this course, students will be able to

CO 1:	Apply the objectives of the project work and provide an adequate background with a detailed literature survey
CO 2:	Breakdown the project into sub blocks with sufficient details to allow the work to be reproduced by an independent researcher
CO 3:	Compose hardware/software design, algorithms, flowchart, methodology, and block diagram
CO 4:	Evaluate the results
CO 5:	Summarize the work carried out

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1				*							
CO 2					*			*			
CO 3							*			*	
CO 4						*					*
CO 5							*				

Course content and outcomes:	
Content	***Competencies***
Phase 1	
Problem identification, synopsis submission, status submission, mid evaluation.	At the end of the topic student should be able to: 1. Identify the problem/specification (C1) 2. Discuss the project (C2) 3. Prepare the outline (C3) 4. Describe the status of the project (C2) 5. Prepare a mid-term project presentation report (C3) 6. Prepare and present mid-term project presentation slides (C3, C5) 7. Develop project implementation in hardware/software or both in chosen platform (C5)
Phase 2	
Status submission, final evaluation.	1. Prepare the progress report (C3) 2. Prepare the final project presentation report (C3) 3. Prepare and present final project presentation slides (C3, C5)

	4. Modify and Develop implementation in hardware/software or both in chosen platform (C3, C5) 5. Justify the methods used and obtained results (C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	-	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	48	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-
Clinic	-	-
Practical	-	-
Revision	-	-
Assessment	03	-
TOTAL	**51**	**09**

Assessment Methods:	
Formative:	**Summative:**
Project Problem Selection	Mid-Term Presentation
Synopsys review	Second status review
First status review	Demo & Final Presentation

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Mid Presentation	*	*			
Presentation	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	Particular to the chosen project

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Seminar - 2
Course Code: ESD 698	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** First Year, Semester 2
No of Credits: 1	**Prerequisites:** Communication Skill
Synopsis:	1. To select, search and learn technical literature. 2. To Identify a current and relevant research topic. 3. To prepare a topic and deliver a presentation. 4. To develop the skill to write a technical report. 5. Develop ability to work in groups to review and modify technical content.

Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	Show competence in identifying relevant information, defining and explaining topics under discussion.
CO 2:	Show competence in working with a methodology, structuring their oral work, and synthesizing information.
CO 3:	Use appropriate registers and vocabulary, and will demonstrate command of voice modulation, voice projection, and pacing.
CO 4:	Demonstrate that they have paid close attention to what others say and can respond constructively.
CO 5:	Develop persuasive speech, present information in a compelling, well-structured, and logical sequence, respond respectfully to opposing ideas, show depth of knowledge of complex subjects, and develop their ability to synthesize, evaluate and reflect on information.

Mapping of COs to POs											
COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1	*							*	*		*
CO 2	*							*	*		*
CO 3	*							*	*		*
CO 4	*							*	*		*
CO5:	*							*	*		*

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	-	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	14	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-
Clinic	-	-
Practical	-	-
Revision	-	-
Assessment	-	-
TOTAL	**14**	**-**

Assessment Methods:	
Formative:	**Summative:**
Seminar Topic Selection	
Synopsys review	
PPT Review	

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Presentation	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	Particular to the chosen Seminar

Name of the Program:	Master of Engineering - ME (Embedded Systems)
Course Title:	Project Work
Course Code: ESD 696	**Course Instructor:**
Academic Year: 2020 - 2021	**Semester:** Second Year, Semester 3, 4
No of Credits: 25	**Prerequisites:** SDLC, Communication Skills, technical skills.
Synopsis:	• The project work aims to challenge analytical, creative ability and to allow students to synthesize, apply the expertise and insight learned in the core discipline.

	• Students build self-confidence, demonstrate independence, and develop professionalism on successfully completion of the project.
Course Outcomes (COs):	On successful completion of this course, students will be able to
CO 1:	To be acquainted with working environment and processes that in place at the relevant Industries.
CO 2:	To familiarize the challenges as relevant professionals.
CO 3:	Review the literature and develop solutions for real time onboard projects.
CO 4:	Write technical report and deliver presentation.
CO 5:	Apply engineering and management principles to achieve project goal.

Mapping of COs to POs

COs	*PO 1*	*PO 2*	*PO 3*	*PO 4*	*PO 5*	*PO 6*	*PO 7*	*PO 8*	*PO 9*	*PO 10*	*PO 11*
CO 1						*	*	*	*	*	*
CO 2					*						
CO 3	*	*	*	*	*						
CO 4	*	*	*	*							
CO5:						*	*	*	*	*	*

Course content and outcomes:	
Content	***Competencies***
Phase 1:	
Problem identification, synopsis submission, status submission, mid evaluation.	At the end of the topic student should be able to: 1. Identify the problem/specification (C1) 2. Discuss the project (C2) 3. Prepare the outline (C3) 4. Prepare a mid-term project presentation report (C3) 5. Prepare and present mid-term project presentation slides (C3, C5) 6. Develop project implementation in hardware/software or both in chosen platform (C5)
Phase 2	
Status submission, final evaluation.	1. Prepare the progress report (C3) 2. Prepare the final project presentation report (C3)

	3. Prepare and present final project presentation slides (C3, C5) 4. Modify and Develop implementation in hardware/software or both in chosen platform (C3, C5) 5. Justify the methods used and obtained results (C6)

Learning strategies, contact hours and student learning time		
Learning strategy	*Contact hours*	*Student learning time (Hrs)*
Lecture	-	-
Seminar	-	-
Quiz	-	-
Small Group Discussion (SGD)	14	-
Self-directed learning (SDL)	-	-
Problem Based Learning (PBL)	-	-
Case Based Learning (CBL)	-	-
Clinic	-	-
Practical	-	-
Revision	-	-
Assessment	-	-
TOTAL	**14**	**-**

Assessment Methods:	
Formative:	**Summative:**
Project Problem Selection	Mid-Term Presentation
Synopsys review	Second status review
First status review	Demo & Final Presentation

Mapping of assessment with Cos					
Nature of assessment	CO 1	CO 2	CO 3	CO 4	CO 5
Mid Presentation	*	*			
Presentation	*	*	*	*	*

Feedback Process	• End-Semester Feedback
Reference Material	Particular to the chosen project

PROGRAM OUTCOMES (POS) AND COURSE OUTCMES (COS) MAPPING

Sl.No.	Course Code	Course Name	Credits	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
1	CSE 601	Data Structures and Algorithms	3	*	*		*		*					
2	CSE 602	Real Time Operating Systems	3	*	*	*	*							
3	ESD 601	Advanced Computer Architecture	3	*	*	*	*	*						
4	ESD 602	Microcontrollers and its Applications	3	*	*	*		*						
5	CSE 610	Computer Networks	3	*	*	*								
6	CSE 604	Database Programming in Java	3	*	*	*	*	*						
7	IOT 607	Internet of Things	3	*	*	*		*						
8	CSE 601L	Data Structures and Algorithms Lab	1		*	*		*			*			
9	CSE 602L	Real Time Operating Systems Lab	1	*	*	*		*						
10	ESD 601L	Advanced Computer Architecture Lab	1		*	*	*	*						
11	ESD 602L	Microcontrollers and its Applications Lab	1	*	*	*		*						
12	CSE 610L	Computer Networks Lab	1	*	*	*								
13	CSE 604L	Database Programming in Java Lab	1		*	*		*						
14	IOT 607L	Internet of Things Lab	1	*	*	*	*	*						
15	ESD 695	Mini Project - 1	4				*	*	*	*	*		*	*

16	ESD 697	Seminar - 1	1	*							*	*		*
17	ESD 603	Digital Signal Processing	3	*	*	*	*	*						
18	ESD 604	Device Drivers	3	*	*	*		*						
19	ESD 605	Embedded Systems	3	*	*	*		*						
20	ESD 606	Embedded Software Design	3											
21	CSE 605	Mobile Application Development using Android	3	*	*	*	*	*						
22	CSE 611	Web Application Development	3	*	*	*	*	*						
23	CSE 612	Multicore Program Optimization	3	*	*	*	*	*						
24	CSE 631	IT Project Management	3	*	*	*								
25	BDA 614	Big Data and Data Visualization	3	*	*	*	*			*				
26	EDA 601	High Level Digital Design	3	*	*	*								
27	ENP 601	Entrepreneurship	3	*		*	*		*		*		*	
28	ESD 603L	Digital Signal Processing Lab	1	*	*		*	*						
29	ESD 604L	Device Drivers Lab	1	*	*	*		*						
30	ESD 605L	Embedded Systems Lab	1	*	*	*		*						

31	ESD 606L	Embedded Software Design Lab	1										
32	CSE 605L	Mobile Application Development using Android Lab	*	*	*	*	*						
33	CSE 611L	Web Application Development Lab		*	*		*						
34	CSE 612L	Multicore Program Optimization Lab	*	*	*		*						
35	CSE 631L	IT Project Management Lab			*	*	*				*		
36	BDA 614L	Big Data and Data Visualization Lab	*	*	*	*	*	*		*	*	*		
37	EDA 601L	High Level Digital Design Lab			*		*						
38	ENP 601L	Entrepreneurship Lab	*					*		*		*		
39	ESD 696	Mini Project - 2	4				*	*	*	*	*		*	*
40	ESD 698	Seminar - 2	1	*							*	*		*
41	ESD 799	Project Work	25	*.	*	*	*	*	*	*	*	*	*	*