Approved Course structure & Syllabus for M.Tech (V25)

COURSE STRUCTURE
AND
DETAILED SYLLABUS

For

M. Tech

Embedded Systems & VLSI

ELECTRONICS & COMMUNICATION ENGINEERING

BRANCH

SRI VASAVI ENGINEERING COLLEGE

(AUTONOMOUS)

(Sponsored by Sri Vasavi Educational Society)

Approved by AICTE, New Delhi and Permanently Affiliated to JNTUK, Kakinada
Pedatadepalli, **TADEPALLIGUDEM – 534 101,** W.G. Dist., (A.P.)

COURSE STRUCTURE

Course Structure for
M.Tech (Embedded Systems &VLSI) w.e.f A.Y 2025-26

I Semester

Sl. No.	Course Code	Course Name	L	T	P	C
1.	V25ESVT01	System Design through VERILOG	4	-	-	4
2.	V25ESVT02	Embedded Systems Design	4	-	-	4
3	V25ESVT03	FPGA Design	4	-	-	4
4.	V25ESVT04 V25ESVT05 V25ESVT06	**Professional Elective-I** VLSI Architectures Scripting languages for VLSI VLSI system design	3	-	-	3
5.	V25ESVT07 V25ESVT08 V25ESVT09	**Professional Elective-II** System on Programming chip design CPLD & FPGA Architectures And Applications ARM Microcontroller based Design	3	-	-	3
6.	V25ESVL01	System Design through Verilog Lab	-	-	4	2
7.	V25ESVL02	Embedded Systems Design Lab	-	-	4	2
8.	V25ESVL03	Seminar-1	-	-	2	1
			18	**0**	**10**	**23**

Total Contact Hours: 28 **Total Credits: 23**

II Semester:

Sl.No	Course Code	Course Name	L	T	P	C
1.	V25ESVT10	Digital CMOS Circuit Design	4	-	-	4
2.	V25ESVT11	Embedded Real Time Operating Systems (ERTOS)	4	-	-	4
3.	V25ESVT12	VLSI Testing & Testability	4	-	-	4
4.	V25ESVT13 V25ESVT14 V25ESVT15	**Professional Elective-III** VLSI Signal processing Advanced VLSI Interconnects Semiconductor Memory Design and Testing VLSI Signal processing	3	-	-	3
5.	V25ESVT16 V25ESVT17 V25ESVT18	**Professional Elective-IV** System design using embedded Processors Internet of Things Embedded Network and Protocols	3	-	-	3
6.	V25ESVL04	Digital CMOS Circuit Design Lab	-	-	4	2
7.	V25ESVL05	Real time Operating Systems Lab	-	-	4	2
8.	V25ESVL06	Seminar-II	-	-	2	1
			18	**0**	**10**	**23**

Total Contact Hours: 28 **Total Credits: 23**

III Semester*

Sl. No.	Course Code	CourseName	L	T	P	Credits
1.	V25ESVT19	Research Methodology and IPR/Swayam12Week MOOC course- RM & IPR	3	0	0	3
2.	V25ESVL07	Summer Internship/Industrial training(8- 10)Weeks*	0	0	0	3
3.	V25ESVL08	Comprehensive Viva#	0	0	0	2
4.	V25ESVP01	Dissertation part-A	0	0	20	10
					Total Credits	**18**

- Student attended during summer/year break and assessment will be done in 3rdSem

\# Comprehensive viva can be conducted courses completed upto second sem

$ Dissertation-Part A internal Assessment

IV Semester:

Sl. No.	Course Code	Course Name	L	T	P	C
1.	V25ESVP02	Dissertation Part–B %	-	-	32	16
	Total		-	-	32	16

% Extrenal Assessment

Total Credits: 80

SEMESTER-I

SYLLABUS

I Sem.	SYSTEM DESIGN THROUGH VERILOG	Course Code:V25ESVT01	L	T	P	C
			4	0	0	4

Syllabus Details

Course Outcomes:

• CO1:Outline basic concepts of RTL code for digital circuits	**[K2]**
• CO2:ModelRTLcodesfordigitalcircuitatgateanddataflowlevel	**[K3]**
• CO3:ModelRTLcodesfordigitalcircuitat behavioral level	**[K3]**
• CO4: Model RTL codes for digital circuit at switch level modeling and outline the concept softask, function and complier directives	**[K3]**
• CO5:AnalyzeSynthesizeofCombinationalandSequentialCircuits	**[K4]**

UNIT-I INTRODUCTION TO VERILOG:

Verilog as HDL, Levels of design description, concurrency, module, simulation and synthesis, test bench, functional verification, programming language interface (PLI), simulation and synthesis tools.

LANGUAGE CONSTRUCTS AND CONVENTIONS:

Introduction, keywords, identifiers, whitespace characters, comments, numbers, strings, logic values, data types, scalars and vectors, parameters, memory, operators, system tasks.

UNIT-II GATE LEVEL MODELLING:

Introduction, AND gate primitive, module structure, other gate primitives, illustrative examples, tristate gates, array of instance of primitives, design of Flipflops with gate primitives, delays, strengths and content on resolution, net types, design of basic circuits.

DATA FLOW LEVEL MODELLING

Introduction, continuous assignment structures, delays and continuous assignments, assignment to vectors.

UNIT-III BEHAVIORAL MODELLING:

Introduction, operations and assignments, initial construct, always construct, examples, assignments with delays, wait construct, multiple always blocks, designs at behavioral level, blocking and non-blocking assignments, the case statement, if and if else constructs, assign-De assign construct, repeat construct, FOR loop, the disable construct, While loop, Forever loop, parallel blocks, force-release construct, event.

UNIT-IV SWITCH LEVEL MODELLING

Basic transistor switches, CMOS switch, Bidirectional gates and time delays with switch primitives, instantiations with strengths and delays, strength contention with triregnets, switch level modeling for NAND, NOR and XOR.

SYSTEM TASKS, FUNCTIONS, AND COMPILER DIRECTIVES: Introduction, System Tasks and Functions, File based Tasks and Functions, Compiler Directives, Hierarchical Directives, User-defined Primitives (UDP), FSM Design (Moore and Melay Machines).

UNIT-V: SYNTHESIS OF COMBINATIONAL AND SEQUENTIAL LOGIC USING VERILOG: Synthesis of combinational logic: Net list of structured primitives, a set of continuous assignment statements and level sensitive cyclic behavior with examples, Synthesis of priority structures, Exploiting logic don‘t care conditions. Synthesis of sequential logic with latches: Accidental synthesis of latches and Intentional synthesis of latches, Synthesis of sequential logic with flip-flops, Synthesis of explicit state machines.

TEXTBOOKS:

1. Design through Verilog HDL— T.R.Padmanabhan and B.BalaTripura Sundari, WSE, IEEE Press,2004.
2. Advanced Digital Design with Verilog HDL—Michael D.Ciletti, PHI,2005.

REFERENCES:

1. Fundamentals of Logic Design withVerilog— tephen.BrownandZvonkoVranesic,TMH,2005.
2. AVerilog Primier—J.Bhasker, BSP,2003.

I Sem.	EMBEDDED SYSTEM DESIGN	Course Code:V25ESVT02	L	T	P	C
			4	0	0	4

Course Outcomes:
The student will be able to

- CO1: Illustrate the ARM architecture and its memory management. (K2)
- CO2: Describe the ARM instruction set for ARM programming. (K2)
- CO3: Describe Thumb instruction set for ARM programming. (K2)
- CO4: Explain the basics of ARM Cortex-M3 (K2)
- CO5: Explain ARM Cortex-M3 interfacing. (K2)

UNIT-I:
ARM Architecture ARM Design Philosophy, Registers, PSR, Pipeline, Interrupts and Vector Table, Architecture Revision, ARM Processor Families. Introduction to ARM Cortex.

UNIT-II:
ARM Programming Model-I Instruction Set: Data Processing Instructions, Branch, Load, Store Instructions, PSR Instructions, Conditional Instructions.

UNIT-III:
ARM Programming Model-II Thumb Instruction Set: Register Usage, Other Branch Instructions, Data Processing Instructions, Single-Register and Multi Register Load-Store Instructions, Stack, Software Interrupt Instructions.

UNIT-IV
Introduction to ARM Cortex-M3 Processor-What Is the ARM Cortex-M3 Processor, Background of ARM and ARM Architecture, Instruction Set Development, The Thumb-2 Technology and Instruction Set Architecture, Cortex-M3 Processor Applications.
Cortex-M3 Basics-Registers, Operation Mode, Exceptions and Interrupts, Vector Tables, Stack Memory Operations, Reset Sequence

UNIT-V
Exceptions, Types, Priority, Vector Tables, Interrupt Inputs and Pending behavior, Fault Exceptions, Supervisor call and Pendable Service Call, Nested Vectored Interrupt Controller, Basic Interrupt Configuration.

TEXT BOOKS:

1. ARM Systems Developer's Guides- Designing & Optimizing System Software – Andrew N. Sloss, Dominic Symes, Chris Wright, 2008, Elsevier.

2.The Definitive Guide to the ARM® Cortex-M3 Second Edition-Joseph Yiu

3. ARM System-on-chip Architecture- Stephen Bo Furber - Addison-Wesley, 2000

REFERENCE BOOKS:

1. Embedded Microcomputer Systems, Real Time Interfacing – Jonathan W. Valvano – Brookes / Cole, 1999, Thomas Learning.

I Sem.	FPGA DESIGN	Course Code:V25ESVT03	L	T	P	C
			4	0	0	4

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

Understand FPGA design flow
CO-1: Identify the building blocks of commercially available FPGA/CPLDs K2
CO-2: Develop VHDL/Verilog models and synthesize targeting for Vertex, Spartan FPGAs K3
CO-3: Develop parameterized library cells and implement system designs using Parameterized K3

Syllabus:

UNIT-1: INTRODUCTION TO FPGAs: Evolution of programmable devices, FPGA Design flow, Applications of FPGA. DESIGN EXAMPLES USING PLDs: Design of Universal block, Memory, Floating point multiplier, Barrel shifter

UNIT-2: FPGAs/CPLDs: Programming Technologies, Commercially available FPGAs, Xilinx's Vertex and Spartan, Actel's FPGA, Altera's FPGA/CPLD.

UNIT-3: Building blocks of FPGAs/CPLDs: Configurable Logic block functionality, Routing structures, Input/output Block, Impact of logic block functionality on FPGA performance, Model for measuring delay.

UNIT-4: Routing Architectures: Routing terminology, general strategy for routing in FPGAs, routing for row – based FPGAs, introduction to segmented channel routing, routing for symmetrical FPGAs, example of routing in a symmetrical FPGA, general approach to routing in symmetrical FPGAs, independence from FPGA routing architectures, FPGA routing structures.

UNIT-5: FPGA architectural assumptions, the logic block, the connection block, connection block topology, the switch block, switch block topology, architectural assumptions for the FPGA CASE STUDY – Applications using Kintex-7, Virtex-7, Artix-7.

TEXT BOOKS:

1. John V. Old Field, Richrad C. Dorf, Field Programmable Gate Arrays, Wiley, 2008.
2. Data sheets of Artix-7, Kintex-7, Virtex-7
3. Stephen D. Brown, Robert J. Francis, Jonathan Rose, Zvonko G. Vranesic, Field Programmable Gate Arrays, 2nd Edition, Springer, 1992.

I Sem.	VLSI Architectures	Course Code:V25ESVT04	L	T	P	C
			3	0	0	3

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

CO1 Design RISC architecture and control units for a given instruction set.
CO2 Improve the performance of RISC processors by applying pipelining techniques
CO3 Translate DSP algorithms into efficient hardware architectures and design associated building blocks
CO4 Analyze the impact of retiming, unfolding, and folding on the performance of DSP architectures

Syllabus:

Unit I Instruction Set Architectures and CPU Performance: Overview of Instruction Set Architectures – CISC, RISC, and DSP Processors, CPU Performance and Its Factors, Evaluating Performance Metrics.

Unit II Design of RISC Processor: Designing the Datapath and Control Unit for a RISC Processor, Multicycle Implementation of RISC Architecture.

Unit III Enhancing Performance with Pipelining: Overview of Pipelining, Pipelined Datapath, Pipelined Control Unit, Pipeline Hazards – Data, Control, and Structural Hazards, Techniques for Hazard-Free Pipelined RISC Implementation.

Unit IV Multiprocessors and DSP Algorithm Representation: Introduction to Multiprocessors, Multiprocessors Connected by a Single Bus and Network, Network Topologies, Evolution vs. Revolution in Computer Architecture, DSP Algorithm Representation – Data Flow Graphs, Loop Bound and Iteration Bound, Algorithms for Computing Iteration Bound.

Unit V
Pipelining, Parallel Processing, and VLSI Performance Techniques: Introduction to Pipelining and Parallel Processing, FIR Filter Pipelining, Parallel Processing Techniques, Pipelining and Parallel Processing for Low Power, VLSI Architecture Optimization Techniques – Retiming, Unfolding, and Folding.

TEXT BOOKS:
1. D.A, Patterson And J.L. Hennessy, Computer Organization and Design: Hardware/ Software Interface, Elsevier, 2011, 4th Edition
2. Keshab Parhi, VLSI digital signal processing systems design and implementations, Wiley 1999

I Sem.	SCRIPTING LANGUAGES FOR VLSI	Course Code:V25ESVT05	L	T	P	C
			3	0	0	3

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

CO1 Gain fluency in programming with scripting languages K3
CO2 Create and run scripts using PERL/TCL/PYTHON in CAD Tools K4
CO3 Demonstrate the use of PERL/PYTHON/ TCL in developing system and web applications K4
CO4 Develop a real time project using PERL/PYTHON K5

Syllabus:

Unit I: Introduction to Scripts and Scripting: Basics of Linux, Origin of Scripting languages, scripting today, Characteristics and uses of scripting languages.

Unit II PERL: Introduction to PERL, Names and values, Variables and assignment, Scalar expressions, Control structures, Built-in functions, Collections of Data, Working with arrays, Lists and hashes, Simple input and output, Strings, Patterns and regular expressions, Subroutines, Scripts with arguments.

Unit III Advanced PERL: Finer points of Looping, Subroutines, Using Pack and Unpack, Working with files, Type globs, Eval, References, Data structures, Packages, Libraries and modules, Objects and modules in action, Tied variables, interfacing to the operating systems, Security issues.

Unit IV TCL: The TCL phenomena, Philosophy, Structure, Syntax, Parser, Variables and data in TCL, Control flow, Data structures, Simple input/output, Procedures, Working with Strings, Patterns, Files and Pipes, Example code.

Unit V Advanced TCL: The eval, source, exec and up-level commands, Libraries and packages, Namespaces, Trapping errors, Event-driven programs, Making applications 'Internet-aware', 'Nuts-and-bolts' internet programming, Security issues, TCL and TK integration. PYTHON: Introduction to PYTHON language, PYTHON-syntax, statements, functions, Built-in functions and Methods, Modules in PYTHON, Exception Handling.

TEXT BOOKS:

1. The World of Scripting Languages- David Barron, Wiley Student Edition, 2010.
2. PYTHON Web Programming, Steve Holden and David Beazley, New Riders Publications 12

REFERENCES:

1. TCL/TK: A Developer's Guide- ClifFlynt, 2003, Morgan Kaufmann Series.
2. Core PYTHON Programming, Chun, Pearson Education, 2006.
3. Learning Perl, Randal L. Schwartz, O' Reilly publications 6th edition 2011.
4. Linux: The Complete Reference", Richard Peterson McGraw Hill Publications, 6th Edition,2008.

I Sem.	VLSI SYSTEM DESIGN	Course Code:V25ESVT06	L	T	P	C
			3	0	0	3

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

CO1 Model the behaviour of a MOS Transistor
CO2 Understanding CMOS Inverter
CO3 Design combinational and sequential circuits using CMOS gates
CO4 Identify the sources of power dissipation in a CMOS circuit.
CO5 Analyze SRAM cell and memory arrays

Syllabus:

Unit I: MOS Transistors, CMOS Logic, CMOS Fabrication and Layout, Design Partitioning, Fabrication, Packaging,and Testing, MOS transistor Theory, Long Channel I-V Characteristics, C-V Characteristics, Non-Ideal I-V Effects, DC Transfer Characteristics. The CMOS Inverter: The Static CMOS Inverter -An Intuitive Perspective, Evaluating the Robustness of the CMOS Inverter: The Static Behavior, Performance of CMOS Inverter: The Dynamic Behavior.

Unit II: CMOS Processing Technology, CMOS Technologies, Layout Design Rules, CMOS Process Enhancements, Technology-Related CAD Issues, Manufacturing Issues, Circuit Simulation- A SPICE Tutorial, Device Models, Device Characterization, Circuit Characterization, Interconnect Simulation. Combinational Circuit Design, Circuit Families, Silicon-On-Insulator Circuit Design, Sub Threshold Circuit Design, Sequential Circuit Design, Circuit Design of Latches and Flip-Flops, Static Sequencing Element Methodology, Sequencing Dynamic Circuits, Synchronizers, Wave Pipelining.

Unit III: Power, Sources of Power Dissipation, Dynamic Power, Static Power, Energy- Delay Optimization, Low Power Architectures, Robustness, Variability, Reliability, Scaling, Statistical Analysis of Variability, VariationTolerant Design. Delay, Transient Response, RC Delay Model, Linear Delay Model, Logical Effort of Paths, Timing Analysis Delay Models, Datapath Subsystems, Addition/Subtraction, One/Zero Detectors, Comparators, Counters, Boolean Logical Operations, Coding, Shifters, Multiplication.

Unit IV: Array Subsystems, SRAM, DRAM, Read-Only Memory, Serial Access Memories, Content-Addressable Memory, Programmable Logic Arrays, Robust Memory Design, Special-Purpose Subsystems.

Unit V: CMOS Testing-The need for testing, Manufacturing test principles, Design strategies for test, Chip level test techniques, System level test techniques, Layout design for improved testability.

TEXT BOOKS:

1. Neil H.E. Weste, David Harris, Ayan Banerjee, CMOS VLSI Design – A Circuits and Systems Perspective, Pearson Education, 2006, 3rd Edition.

2. Neil H. E. Weste Kamran Eshraghian, Principles of CMOS VLSI DESIGN:A Systems Perspective, Pearson Education, 2006, 2nd Edition.

REFERENCE BOOKS:

1. Jan M RABAEY, Digital Integrated Circuits, Pearson Education, 2003, 2nd Edition.
2. Douglas A. Pucknell, Kamran Eshraghian, Basic VLSI Design, PHI,1994, 3 rd Edition.

I Sem.	System on Programming chip Design	Course Code:V25ESVT07	L	T	P	C
			3	0	0	3

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

CO1 Understand the fundamental concepts and components of System-on-Chip (SoC) design, K2
CO2 Analyze and compare processor architectures such as RISC, CISC, VLIW, Superscalar, and soft/firm/custom processors, K4
CO3 Evaluate various interconnection mechanisms like on-chip buses K4
CO4 Apply IP-based design methodologies in SoC development, K3
CO5 Design and assess SoC implementations and testing techniques, K5

Unit I: Introduction: Driving Forces for SoC - Components of SoC - Design flow of SoC Hardware/Software nature of SoC - Design Trade-offs – SoC Applications. System-level Design: Processor selection-Concepts in Processor Architecture: Instruction set architecture (ISA), elements in Instruction Handing-Robust processors: Vector processor, VLIW, Superscalar, CISC, RISC—Processor evolution: Soft and Firm processors, Custom-Designed processors- on-chip memory.

Unit II: Interconnection: On-chip Buses: basic architecture, topologies, arbitration and protocols, Bus standards: AMBA, CoreConnect, Wishbone, Avalon - Network-on-chip: Architecture- topologies-switching strategies – routing algorithms flow control, Quality-of-Service- Re- configurability in communication architectures.

Unit III: IP based system design: Introduction to IP Based design, Types of IP, IP across design hierarchy, IP life cycle, Creating and using IP – Technical concerns on IP reuse – IP integration - IP evaluation on FPGA prototypes.

Unit IV: SOC implementation: Study of processor IP, Memory IP, wrapper Design - Real-time operating system (RTOS), Peripheral interface and components, High-density FPGAs - EDA tools used for SOC design.

Unit V: SOC testing: Manufacturing test of SoC: Core layer, system layer, application layer - P1500 Wrapper Standardization - SoC Test Automation (STAT).

TEXT BOOKS:

1. Michael J.Flynn, Wayne Luk, “Computer system Design: System on- Chip”, Wiley-India, 2012.
2. Sudeep Pasricha, NikilDutt,“On Chip Communication Architectures: System on Chip Interconnect”, Morgan Kaufmann Publishers, 2008.
3. W.H.Wolf, “Computers as Components: Principles of Embedded Computing System Design”, Elsevier, 2008.

REFERENCE BOOKS:

1. Patrick Schaumont “APractical Introduction to Hardware/Software Codesign”, 2nd Edition, Springer, 2012.
2. Lin,Y-L.S. (ed.),“Essential issues in SOC design: designing complex systems-on chip. Springer, 2006.
3. Wayne Wolf, “Modern VLSI Design: IP Based Design”, Prentice-HallIndia, Fourth edition, 2009.

I Sem.	CPLD & FPGA Architectures and Applications	Course Code:V25ESVT08	L	T	P	C
			3	0	0	3

Course Outcome: The student will be able to

- CO1: Describe the Programmable Logic Devices
- CO2: Distinguish the various types of Field Programmable Gate Arrays
- CO3: Apply the typical applications on FPGAs

UNIT-I: Introduction to Programmable Logic Devices
Introduction, Simple Programmable Logic Devices – Read Only Memories, Programmable Logic Arrays, Programmable Array Logic, Programmable Logic Devices/Generic Array Logic; Complex Programmable Logic Devices –Architecture of Xilinx Cool Runner XCR3064XL CPLD, CPLD Implementation of a Parallel Adder with Accumulation.

UNIT-II: Field Programmable Gate Arrays Organization of FPGAs, FPGA Programming Technologies, Programmable Logic Block Architectures, Programmable Interconnects and Programmable I/O blocks in FPGAs, Dedicated Specialized Components of FPGAs and Applications of FPGAs.

UNIT –III: SRAM Programmable FPGAs: Introduction, Programming Technology, Device Architecture, the Xilinx XC2000, XC3000 and XC4000 Architectures.

UNIT –IV: Anti-Fuse Programmed FPGAs Introduction, Programming Technology, Device Architecture, the Actel ACT1, ACT2 and ACT3 Architectures.

UNIT –V: Design Applications General Design Issues, Counter Examples, a Fast Video Controller, A Fast DMA Controller and Designing Counters with ACT devices, Designing Adders and Accumulators with the ACT Architecture.

TEXT BOOKS:

1. Field Programmable Gate Array Technology - Stephen M. Trimberger, Springer International Edition.
2. Digital Systems Design - Charles H. Roth Jr, LizyKurian John, Cengage Learning.

REFERENCE BOOKS:

1. Field Programmable Gate Arrays - John V. Oldfield, Richard C. Dorf, Wiley India.
2. Digital Design Using Field Programmable Gate Arrays - Pak K. Chan/ Samiha Mourad, Pearson Low Price Edition.
3. Digital Systems Design with FPGAs and CPLDs - Ian Grout, Elsevier, Newnes.
4. FPGA based System Design - Wayne Wolf, Prentice Hall Modern Semiconductor Design Series.

I Sem.	ARM MICROCONTROLLER BASED DESIGN	Course Code:V25ESVT09	L	T	P	C
			3	0	0	3

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

CO1 Explore the selection criteria of ARM processors by understanding the functional level tradeoff issues. K2
CO2 Implementations on ARM developments towards the functional capabilities K4
CO3 Work with ASM level program using the instruction set. K2
CO4 Programming the ARM Cortex M. K5
CO5 Discuss about Floating Point Operations: K3

Syllabus:

Unit I ARM Embedded Systems: RISC design philosophy, ARM design philosophy, Embedded system hardware, Embedded system software. ARM Processor Fundamentals: Registers, CPSR, Pipeline, Exceptions, Interrupts and Vector Table, Core Extensions, Architecture Revisions, ARM Processor Families. Architecture of ARM Processors: Programmer's model, modes and states, special and floating-point registers, APSR, Memory system, MPU, Exceptions, NVIC, vector table, Fault handling, SCB, Debug, Reset sequence.

Unit II ARM Instruction Set: Data processing, branch, load-store, software interrupt, program status register instructions, loading constants, ARMv5E extensions, Conditional execution. Thumb Instruction Set: Thumb Register Usage, ARM-Thumb Interworking, Branch, Data Processing, Load-Store, Stack, and Software Interrupt Instructions.

Unit III Technical Details of Cortex M Processors: Overview of Cortex-M3 and M4: architecture, instruction set, block diagram, memory system, exception and interrupt support. Features: Performance, code density, low power, MPU, OS support, Cortex-M4-specific DSP features, Debug support, Scalability, Compatibility.

Unit IV Instruction Set of Cortex M: Instruction set background, comparison across Cortex-M processors, UAL syntax, instruction suffixes, Cortex-M4-specific instructions, Barrel shifter, Special instructions and register access.

Unit V Floating Point Operations: Floating point data and FPU overview (CPACR, FP registers, FPSCR, FPCCR, FPCAR, FPDSCR, MVFR0, MVFR1). DSP Applications: Dot product, Biquad filter, FIR, FFT and optimized DSP code writing for Cortex-M4.

TEXT BOOKS:

1. Andrew N.SLOSS, Dominic SYMES, Chris WRIGHT-ARM System Developer's Guide Designing and Optimizing System Software, Elsevier Publications, 2004.
2. Joseph Yiu, The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors by Elsevier Publications, 3rdEd.,

REFERENCE BOOKS:

1. Steve Furber- Arm System on Chip Architectures–Edison Wesley, 2000.
2. David Seal-ARM Architecture Reference Manual, EdisonWesley, 2000.

I Sem.	System Design through Verilog Lab	Course Code:V25ESVL01	L	T	P	C
			0	0	4	2

Syllabus Details

COURSE OUTCOMES:

CO1: Develop the simulation of combinational and sequential circuits using HDL Language.**[K3]**
CO2: Develop the synthesis of combinational and sequential circuits using HDL Language.**[K3]**
CO3: Analyze the implemented of digital logics with hardware module kit FPGA **[K4]**

The students are required to design the Verilog codes to perform the following experiments using necessary simulator (Xilinx ISE Simulator) to verify the logic functional operation and to perform the analysis with appropriate synthesizer (Xilinx ISE Synthesizer) and then verify the implemented logic function with hardware kits (FPGA kits).

The students are required to acquire the knowledge in the platform Xilinx by perform at least 10 experiments.

List of Experiments:

1) Logic gates

2) Adder-Subtractor

3) Multiplexer and DE multiplexer

4) Encoder and Decoder

5) ALU

6) Fire detection and control system using Combinational Logic Circuits

7) Flip Flops

8) LFSR

9) Up counter/Down counter

10) Synchronous RAM

11) Pattern detector using Moore/Melay machine

12) Traffic light controller using sequential logic circuit.

13) UART

I Sem.	Embedded Systems Design Lab	Course Code:V25ESVL02	L	T	P	C
			0	0	4	2

Syllabus Details

Course Outcomes:

At the end of the laboratory work, students will be able to:

CO1: Develop applications based on ARM Cortex-M3 processor using Cortex-M3 Development boards on the platform of co-coox and Arduino IDE.-**K3**

CO2: Develop the applications based on DSP C6713 evaluation kits and using Code Composer Studio (CCS).-**K3**

List of Assignments:

Part A:

Experiments to be carried out on Cortex-M3 development boards and using GNU Tool chain

1. Blink an LED with software delay, delay generated using the SysTick timer.
2. Control intensity of an LED using PWM implemented in software and hardware.
3. Control an LED using switch by polling method, by interrupt method and flash the LED once every five switch presses.
4. UART Echo Test.
5. Take Analog readings on rotation of rotary potentiometer connected to an ADC channel.
6. Temperature indication on an RGB LED.
7. Mimic light intensity sensed by the light sensor by varying the blinking rate of an LED.
8. Evaluate the various sleep modes by putting core in sleep and deep sleep modes.
9. System reset using watchdog timer in case something goes wrong.
10. Sample sound using a microphone and display sound levels on LEDs.

Part B:

Experiments to be carried out on DSP C6713 evaluation kits and using Code Composer Studio (CCS)

1. To develop a C code to compute Euclidian distance between any two Points.
2. To develop a C code for implementation of convolution operation.
3. To develop a C code to compute FFT.
4. To design and implement filters in C to enhance the features of given input sequence/signal.

Lab Requirements:

1. Coo-coX Software PlatForm.
2. Arduino IDE
3. Code Composer Studio(CCS)

Hardware:

1. The Development kits of ARM-Cortex Boards
2. DSP C6713 evaluation kits
3. Sensors for Interfacing
4. Serial cables, Network Cables and Recommended power Supply for the board.

SEMESTER-II
SYLLABUS

II Sem.	Digital CMOS Circuit Design	Course Code:V25ESVT10	L	T	P	C
			4	0	0	4

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

CO1 Analyze MOSFET behavior and CMOS inverter characteristics under static and dynamic conditions. K4
CO2 Design various combinational and sequential logic blocks using CMOS technology. K5
CO3 Optimize data path elements such as adders, multipliers, and barrel Shifters K4
CO4Design and evaluate memory architectures including SRAM and ROM Cells K5
CO5 Interpret and implement circuit layouts using stick diagrams and layout Rules K3

UNIT I MOS Transistor Principles and CMOS Inverter : MOSFET characteristics under Static and Dynamic Conditions, MOS Transistor Secondary Effects, CMOS Inverter – Static Characteristic, Dynamic Characteristic, Power, Energy, and Energy Delay Parameters, Stick Diagram and Layout Diagrams.

UNIT II Combinational Logic Circuits : Static CMOS Design, Different Styles of Logic Circuits, Logical Effort of Complex Gates, Static and Dynamic Properties of Complex Gates, Interconnect Delay, Dynamic Logic Gates.

UNIT III S Sequential Logic Circuits : Static Latches and Registers, Dynamic Latches and Registers, Timing Issues, Pipelines, Non-Bistable Sequential Circuits.

UNIT IV Arithmetic Building Blocks : Data Path Circuits, Architectures for Adders, Accumulators, Multipliers, Barrel Shifters, Speed and Area Tradeoffs.

UNIT V Memory Architectures : Memory Architectures and Memory Control Circuits: Read-Only Memories, ROM Cells, Read-Write Memories (RAM), Dynamic Memory Design, 6-Transistor SRAM Cell, Sense Amplifiers.

TEXT BOOKS:

1. JanRabaey, Anantha Chandrakasan, BNikolic,"Digital Integrated Circuits: A Design Perspective", Prentice Hall of India, 2nd Edition, Feb 2003
2. N.Weste, K.Eshraghian,"PrinciplesofCMOSVLSIDesign",AddisionWesley,2nd Edition, 1993

REFERENCE BOOKS:

1. MJ Smith,"ApplicationSpecificIntegratedCircuits",AddissonWesley,1997
2. Sung-MoKang &Yusuf Leblebici, "CMOS Digital Integrated Circuits Analysis and Design ",McGraw-Hill,1998

II Sem.	EMBEDDED REAL TIME OPERATING SYSTEMS (ERTOS)	Course Code:V25ESVT11	L	T	P	C
			4	0	0	4

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

CO1 Illustrate real time programming concepts. K3

CO2 Apply RTOS functions to implement embedded applications K3

CO3 Understand fundamentals of design consideration for embedded applications K2

CO4 Describe about the memory units and real time memory applications K4

CO5 Discuss communication Common Design Problems K3

Syllabus:

UNIT I Introduction to Real-Time Operating Systems: Defining an RTOS, The scheduler, Kernel Objects and services, Key characteristics of an RTOS. Task: Defining a Task, Task States and Scheduling, Typical Task Operations, Typical Task Structure, Synchronization, Communication and Concurrency.

UNIT II Semaphores: Defining Semaphores, Typical Semaphore Operations, Typical Semaphore Use. Message Queues: Defining Message Queues, Message Queue States, Message Queue Content, Message Queue Storage, Typical Message Queue Operations, Typical Message Queue Use. Pipes, Event Registers, Signals and Condition Variables.

UNITIII Exceptions and Interrupts: Exceptions and Interrupts, Applications of Exceptions and Interrupts, Closer look at exceptions and interrupts, Processing General Exceptions, Nature of Spurious Interrupts. Timer and Timer Services: Real-Time Clocks and System Clocks, Programmable Interval Timers, Timer Interrupt Service Routines. I/O Subsystems: I/O Concepts, I/O Subsystems.

UNIT IV Memory Management: Dynamic Memory Allocation in Embedded Systems, Fixed-Size Memory Management in Embedded Systems, Blocking vs. Non- Blocking Memory Functions, Hardware Memory Management Units. Modularizing an Application for Concurrency: An Outside-In Approach to Decompose Applications, Guidelines and Recommendations for Identifying Concurrency, Schedulability Analysis.

UNIT V Synchronization and Communication: Synchronization, Communication, Resource Synchronization Methods, Critical Section, Common Practical Design Patterns, Specific Solution Design Patterns. Common Design Problems: Resource Classification, Deadlocks, Priority Inversion.

Text Books

1. Qing Li, Caroline Yao (2003), “Real-Time Concepts for Embedded Systems”, CMP Books.

Reference Books

1. Albert Cheng, (2002), “Real-Time Systems: Scheduling, Analysis and Verification”, Wiley Inter science.

2. Hermann Kopetz, (1997), “Real-Time Systems: Design Principles for Distributed Embedded Applications”, Kluwer.

3. Insup Lee, Joseph Leung, and Sang Son, (2008) “Handbook of Real-Time Systems”, Chapman and Hall.Krishna and Kang G Shin, (2001), “Real-Time Systems”, McGraw Hill.

II Sem.	VLSI TESTING & TESTABILITY	Course Code:V25ESVT12	L	T	P	C
			4	0	0	4

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

CO1 Identify the significance of testable design K3

CO2 Understand the concept of yield and identify the parameters influencing the Same K2

CO3 Specify fabrication defects, errors, and faults K3

CO4 Implement combinational and sequential circuit test generation algorithms K4

CO5 Identify techniques to improve fault coverage K5

UNIT I Role of Testing in VLSI Design Flow, Testing at Different Levels of Abstraction, Fault, Error, Defect, Diagnosis, Yield. Types of Testing, Rule of Ten, Defects in VLSI Chip. Modelling Basic Concepts, Functional Modelling at Logic Level and Register Level, Structure Models, Logic Simulation, Delay Models. Various Types of Faults, Fault Equivalence and Fault Dominance in Combinational and Sequential Circuits.

UNIT II Fault Simulation Applications, General Fault Simulation Algorithms: Serial and Parallel, Deductive Fault Simulation Algorithms.

UNIT III Combinational Circuit Test Generation, Structural Vs Functional Test, ATPG, Path Sensitization Methods. Difference Between Combinational and Sequential Circuit Testing, Five and Eight Valued Algebra, Scan Chain- Based Testing Method.

UNIT IV D-Algorithm Procedure, Problems. PODEM Algorithm, Problems on PODEM Algorithm. FAN Algorithm, Problems on FAN Algorithm. Comparison of D, FAN and PODEM Algorithms. Design for Testability, Ad-Hoc Design, Generic Scan-Based Design.

UNIT V Classical Scan-Based Design, System Level DFT Approaches. Test Pattern Generation for BIST, Circular BIST, BIST Architectures. Testable Memory Design: Test Algorithms, Test Generation for Embedded RAMs.

TEXT BOOKS:

1. M. Abramovici, M. Breuer, and A. Friedman, “Digital Systems Testing and Testable Design, IEEE Press, 1990.
2. M. Bushnell and V. Agrawal, “Essentials of Electronic Testing for Digital, Memory & Mixed-Signal VLSI Circuits”, Kluwer Academic Publishers, 2000.

REFERENCE BOOKS:

1. Stroud,“A Designer’s Guide to Built-in Self-Test”, Kluwer AcademicPublishers,2002
2. V.Agrawal and S.C.Seth, Test Generation for VLSI Chips, Computer Society Press. 1989

II Sem.	ADVANCED VLSI INTERCONNECTS	Course Code:V25ESVT13	L	T	P	C
			3	0	0	3

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

CO1 Gain insight into transmission line parameters of VLSI interconnects. K3

CO2 Understand novel and emerging solutions for future VLSI interconnect technologies. K2

CO3 Analyze the impact of inductive effects in high-speed interconnects. K4

CO4 Examine the influence of quantum effects in nano scale interconnects. K4

Syllabus:

UNIT I Introduction: Introduction to VLSI Interconnects, The Distributed RC Interconnect Model, Elmore Delay in Interconnects, Scaling Effects in Interconnects, Simulation and Delay Mitigation in RC Interconnects.

UNIT II Inductive Effects: Inductive Effects in Interconnects, Distributed RLC Interconnect Model, Transmission Line Equations, When to Consider the Inductive Effects?, Equivalent Elmore Model for RLC Interconnects, Two- Pole Model of RLC Interconnects from ABCD Parameters, RLC Interconnect Simulation.

UNIT III Skin Effect and Electromigration: Origin of the Skin Effect, Effective Resistance at High Frequencies, Power Dissipation due to Interconnects, Electromigration in Interconnects, Mitigation of Electromigration.

UNIT IV Crosstalk: Capacitive Coupling in Interconnects, Crosstalk Effects in Two Identical Interconnects, Mitigation Techniques, Analysis and Simulation of Coupled Interconnects. Extraction of Capacitance, Extraction of Inductance, Estimation of Interconnect Parameters from S-parameters.

UNIT V Quantum Effects: Quantum Conductance, Quantum Capacitance, Kinetic Inductance, Graphene Nano ribbon Inter connects, Analysis and Simulation of Interconnect Considering Quantum Effects.

TEXT BOOKS:

1. Ashok K.Goel, High-SpeedVLSIInterconnects,2007.
2. Y.S.Diamand, Advanced Nanoscale ULSI Interconnects: Fundamentals and Applications, 2009.

REFERENCE BOOKS:

1.H.SPhilip Wong and DejiAkinwande, Carbon nanotube and Graphene Device Physics,2011.

II Sem.	Semiconductor Memory Design and Testing	Course Code:V25ESVT14	L	T	P	C
			3	0	0	3

Course Outcome: The students will be able to

- CO1: Describe concepts of volatile and non-volatile memory technologies.
- CO2: Discuss the fault modeling and testing memory devices.
- CO3: Explain the reliability and radiation effects of memory devices.
- CO4: Describe the advanced memory technologies.

UNIT-I: Random Access Memory Technologies

SRAM – SRAM Cell structures, MOS SRAM Architecture, MOS SRAM cell and peripheral circuit operation, Bipolar SRAM technologies, SOI technology, Advanced SRAM architectures and technologies, Application specific SRAMs, DRAM – DRAM technology development, CMOS DRAM, DRAM cell theory and advanced cell structures, BICMOS DRAM, soft error failure in DRAM, Advanced DRAM design and architecture, Application specific DRAM.

UNIT-II: Non-volatile Memories

Masked ROMs, High density ROM, PROM, Bipolar ROM, CMOS PROMS, EPROM, Floating gate EPROM cell, One time programmable EPROM, EEPROM, EEPROM technology and architecture, Non-volatile SRAM, Flash Memories (EPROM or EEPROM), advanced Flash memory architecture.

UNIT-III: Memory Fault Modeling Testing and Memory Design for Testability and Fault Tolerance

RAM fault modeling, Electrical testing, Pseudo Random testing, Megabit DRAM Testing, nonvolatile memory modeling and testing, IDDQ fault modeling and testing, Application specific memory testing, RAM fault modeling, BIST techniques for memory.

UNIT-IV: Semiconductor Memory Reliability and Radiation Effects

General reliability issues RAM failure modes and mechanism, Non-volatile memory reliability, reliability modeling and failure rate prediction, Design for Reliability, Reliability Test Structures, Reliability Screening and qualification, Radiation effects, Single Event Phenomenon (SEP), Radiation Hardening techniques, Radiation Hardening Process and Design Issues, Radiation Hardened Memory characteristics, Radiation Hardness Assurance and Testing, Radiation. Dosimetry, Water Level Radiation Testing and Test structures.

UNIT-V: Advanced Memory Technologies and High-density Memory Packing Technologies

Ferroelectric RAMs (FRAMs), GaAs FRAMs, Analog memories, magneto resistive RAMs (MRAMs), Experimental memory devices, Memory Hybrids and MCMs (2D), Memory Stacks and MCMs (3D), Memory MCM testing and reliability issues, Memory cards, High Density Memory Packaging Future Directions.

TEXT BOOKS:

1. Semiconductor Memories Technology – Ashok K. Sharma, 2002, Wiley.
2. Advanced Semiconductor Memories – Architecture, Design and Applications - Ashok K. Sharma- 2002, Wiley.
3. Modern Semiconductor Devices for Integrated Circuits – Chenming C Hu, 1st Ed, Prentice Hall.

II Sem.	VLSI Signal Processing	Course Code:V25ESVT15	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcomes On successful completion of the module, students will be able to:

CO1:Ability to modify the existing or new DSP architectures suitable for VLSI.

CO2:Understand the concepts of folding and unfolding algorithms and applications.

CO3:Ability to implement fast convolution algorithms.

CO4:Low power design aspects of processors for signal processing and wireless Applications.

UNIT -I

Introduction to DSP: Typical DSP algorithms, DSP algorithms benefits, Representation of DSP algorithms Pipelining and Parallel Processing Introduction, Pipelining of FIR Digital filters, Parallel Processing, Pipelining and Parallel Processing for Low Power Retiming Introduction, Definitions and Properties, Solving System of Inequalities, Retiming Techniques

UNIT –II

Folding and Unfolding: Folding- Introduction, Folding Transform, Register minimization Techniques, Register minimization in folded architectures, folding of Multi rate systems
Unfolding- Introduction, An Algorithm for Unfolding, Properties of Unfolding, critical Path, Unfoldingand Retiming, Applications of Unfolding

UNIT -III

Systolic Architecture Design: Introduction, Systolic Array Design Methodology, FIR Systolic Arrays,Selection of Scheduling Vector, Matrix Multiplication and 2D Systolic Array Design, Systolic Design forSpace Representations contain Delays.

UNIT -IV

Fast Convolution: Introduction – Cook-Toom Algorithm – Winogard algorithm – Iterated Convolution –Cyclic Convolution – Design of Fast Convolution algorithm by Inspection

Unit V: Digital lattice filter structures, bit level arithmetic, architecture, redundant arithmetic. Numerical strength reduction, synchronous, wave and asynchronous pipe lines, lowpower design. Low Power Design: Scaling Vs. Power Consumption, Power Analysis, Power Reduction techniques, Power Estimation Approaches

Text Books:

1. Keshab K. Parthi[A1], VLSI Digital signal processing systems, design and Implementation [A2],Wiley, Inter Science, 1999.
2. Mohammad Isamail and Terri Fiez, Analog VLSI signal and information processing, McGrawHill, 1994
3. S.Y. Kung, H.J. White House, T. Kailath, VLSI and Modern Signal Processing, Prentice Hall,1985.

II Sem.	SYSTEM DESIGN USING EMBEDDED PROCESSORS	Course Code:V25ESVT16	L	T	P	C
			3	0	0	3

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

CO1 Understand the fundamental concepts, architecture, and application areas K2
CO2 Explain the ARM Cortex-M3 architecture, its instruction sets, K2
CO3 Analyze exception handling mechanisms, Nested Vectored Interrupt Controller (NVIC), K4
CO4 Develop embedded programs using C and assembly language K4
CO5 Apply knowledge of STM32L15xxx microcontroller architecture K4

UNIT I Embedded Concepts: Introduction to embedded systems, Application Areas, Categories of embedded systems, Overview of embedded system architecture, Specialties of embedded systems, Recent trends in embedded systems, Architecture of embedded systems, Hardware architecture, Software architecture, Application Software, Communication Software, Development and debugging Tools. ARM Architecture: Background of ARM Architecture, Architecture Versions, Processor Naming, Instruction Set Development, Thumb-2 and Instruction Set Architecture.

UNIT II Overview of Cortex-M3:Cortex-M3 Basics: Registers, General Purpose Registers, Stack Pointer, Link Register, Program Counter, Special Registers, Operation Mode, Exceptions and Interrupts, Vector Tables, Stack Memory Operations, Reset Sequence. Instruction Sets: Assembly Basics, Instruction List, Instruction Descriptions. Cortex-M3 Implementation Overview: Pipeline, Block Diagram, Bus Interfaces on Cortex-M3, I-Code Bus, D-Code Bus, System Bus, External PPB and DAP Bus.

UNIT III Exceptions: Exception Types, Priority, Vector Tables, Interrupt Inputs and Pending Behavior, Fault Exceptions, Supervisor Call and Pendable Service Call. NVIC: Nested Vectored Interrupt Controller Overview, Basic Interrupt Configuration, Software Interrupts and SYSTICK Timer. Interrupt Behavior: Interrupt/Exception Sequences, Exception Exits, Nested Interrupts, Tail-Chaining Interrupts, Late Arrivals and Interrupt Latency.

UNIT IV Cortex-M3/M4 Programming: Overview, Typical Development Flow, Using C, CMSIS (Cortex Microcontroller Software Interface Standard), Using Assembly. Exception Programming: Using Interrupts, Exception/Interrupt Handlers, Software Interrupts, Vector Table Relocation. Memory Protection Unit and Other Cortex-M3 Features: MPU Registers, Setting Up the MPU, Power Management, Multiprocessor Communication.

UNIT V Cortex-M3/M4 Microcontroller: STM32L15xxx ARM Cortex M3/M4 Microcontroller: Memory and Bus Architecture, Power Control, Reset and Clock Control, STM32L15xxx Peripherals: GPIOs, System Configuration Controller, NVIC, ADC, Comparators, GP Timers, USART. Development and Debugging Tools: Software and Hardware tools like Cross Assembler, Compiler, Debugger, Simulator, In-Circuit Emulator (ICE), Logic Analyzer etc.

TEXT BOOKS:

1. The Definitive Guide to the ARM Cortex-M3, Joseph Yiu, Second Edition, Elsevier Inc. 2010.
2. Embedded/Real Time Systems Concepts, Design and Programming Black Book, Prasad, KVK.
3. David Seal "ARM Architecture Reference Manual", 2001 Addison Wesley, England; Morgan Kaufmann Publishers

REFERENCES:

1. Steve Furber, "ARM System-on-Chip Architecture", 2ndEdition, Pearson Education
2. Cortex-M series-ARM Reference Manual
3. Cortex-M3 Technical Reference Manual (TRM)

II Sem.	INTERNET OF THINGS	Course Code:V25ESVT17	L	T	P	C
			3	0	0	3

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

CO1 Analyze and compare various IoT hardware platforms K4
CO2 Understand the fundamentals of networking, OSI model, K2
CO3 Explain IoT architecture, communication patterns, K2
CO4 Develop IoT applications using web technologies, databases, K5
CO5 Evaluate advanced IoT use cases, sensor node integration, K4

UNIT I The IoT Networking Core: Technologies involved in IoT Development: Internet/Web and Networking Basics, OSI Model, Data transfer referred with OSI Model, IP Addressing, Point to Point Data transfer, Point to Multi Point Data transfer & Network Topologies, Sub-netting, Network Topologies referred with Web, Introduction to Web Servers, Introduction to Cloud Computing.

UNIT II IoT Platform Overview: Overview of IoT supported Hardware platforms such as Raspberry Pi, ARM Cortex Processors, Arduino and Intel Galileo boards. Network Fundamentals: Overview and working principle of Wired Networking equipment – Routers, Switches; Overview and working principle of Wireless Networking equipment – Access Points, Hubs etc. Linux Network Configuration Concepts: Networking configurations in Linux, Accessing Hardware & Device Files interactions.

UNIT III IoT Architecture: History of IoT, M2M – Machine to Machine, Web of Things, IoT protocols. Applications: Remote Monitoring & Sensing, Remote Controlling, Performance Analysis. The Architecture: The Layering concepts, IoT Communication Pattern, IoT Protocol Architecture, The 6LoWPAN. Security aspects in IoT.

UNIT IV IoT Application Development: Application Protocols. Back-end Application Designing: Apache for handling HTTP Requests, PHP & MySQL for data processing, MongoDB Object type Database, HTML, CSS & jQuery for UI Designing, JSON library for data processing, Security & Privacy during development. Application Development for Mobile Platforms: Overview of Android / iOS App Development tools.

UNIT V Case Study & Advanced IoT Applications: IoT applications in home, infrastructures, buildings, security, industries, home appliances, and other IoT electronic equipment. Use of Big Data and Visualization in IoT, Industry 4.0 concepts. Sensors and Sensor Nodes and interfacing using any embedded target boards (Raspberry Pi / Intel Galileo / ARM Cortex / Arduino).

TEXT BOOKS:

1. 6LoWPAN: The Wireless Embedded Internet, Zach Shelby, Carsten Bormann, Wiley
2. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, Dr. Ovidiu Vermesan, Dr. Peter Friess, River Publishers
3. Interconnecting Smart Objects with IP: The Next Internet, Jean-Philippe Vasseur, Adam Dunkels, Morgan Kuffmann

REFERENCES:

1. The Internet of Things: From RFID to the Next-Generation Pervasive Network ed Lu Yan, Yan Zhang, Laurence T. Yang, Huansheng Ning
2. Internet of Things (A Hands-on-Approach), Vijay Madisetti, Arshdeep Bahga
3. Designing the Internet of Things, Adrian Mc Ewen (Author), Hakim Cassimally
4. Asoke K Talukder and RoopaR Yavagal,“Mobile Computing,” Tata Mc Graw Hill, 2010.

II Sem.	EMBEDDED NETWORKS AND PROTOCOLS	Course Code:V25ESVT18	L	T	P	C
			3	0	0	3

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

CO1 Acquire knowledge on communication protocols of connecting Embedded Systems K3

CO2 Master the design level parameters of USB and CAN bus protocols. K2

CO3 Design Ethernet in Embedded networks considering different issues. K5

CO4 Acquire the knowledge of wireless protocols in Embedded domain. K4

Unit I Embedded Networking: Introduction – Serial/Parallel Communication – Serial communication protocols - RS232 standard – RS485 – Synchronous Serial Protocols - Serial Peripheral Interface (SPI) – Inter Integrated Circuits (I2C) – PC Parallel port programming - ISA/PCI Bus protocols – Firewire.

Unit II USB bus – Introduction – Speed Identification on the bus – USB States – USB bus communication Packets – Data flow types – Enumeration – Descriptors – PIC18 icrocontroller USB Interface – C Programs – CAN Bus – Introduction - Frames – Bit stuffing – Types of rrors – Nominal Bit Timing – PIC microcontroller CAN Interface – A simple application with CAN.

Unit III Elements of a network – Inside Ethernet – Building a Network: Hardware options – Cables, Connections and network speed – Design choices: Selecting components – Ethernet Controllers – Using the internet in local and internet communications – Inside the Internet protocol.

Unit IV Exchanging messages using UDP and TCP – Serving web pages with Dynamic Data – Serving web pages that respond to user Input – Email for Embedded Systems – Using FTP – Keeping Devices and Network secure.

Unit V Wireless sensor networks – Introduction – Applications – Network Topology – Localization – Time Synchronization – Energy efficient MAC protocols – SMAC – Energy efficient and robust routing – Data Centric routing

TEXT BOOKS

1. Embedded Systems Design: A Unified Hardware/Software Introduction-Frank Vahid, Tony Givargis, John & Wiley Publications, 2002
2. Parallel Port Complete: Programming, interfacing and using the PCs parallel printer port-Jan Axelson, Penram Publications, 1996.

REFERENCE BOOKS

1. Advanced PIC microcontroller projects in C: from USB to RTOS with the PIC18F series –Dogan Ibrahim, Elsevier 2008.
2. Embedded Ethernet and Internet Complete-Jan Axelson, Penram publications,2003.
3. Networking Wireless Sensors-Bhaskar Krishnama chari ,Cambridge press2005.

II Sem.	DIGITAL CMOS CIRCUIT DESIGN LAB	Course Code:V25ESVL04	L	T	P	C
			0	0	4	2

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

CO1 Have the ability to explain the VLSI Design Methodologies using Mentor Graphics Tools K3
CO2 Grasp the significance of various design logic Circuits in full-custom IC Design. K4
CO3 Have the ability to explain the Physical Verification in Layout Extraction K3
CO4 Fully Appreciate the design and analyze of CMOS Digital Circuits K4
CO5 Grasp the Significance of Pre-Layout Simulation and Post-Layout Simulation K5

List of Experiments:

1. Inverter Characteristics.
2. NAND and NOR Gate
3. XOR and XNOR Gate
4. 2:1 Multiplexer
5. Full Adder
6. RS-Latch
7. Clock Divider
8. JK-Flip Flop
9. Synchronous Counter
10. Asynchronous Counter
11. Static RAM Cell
12. Dynamic Logic Circuits
13. Linear Feedback Shift Register

Lab Requirements:

Software:
Mentor Graphics Tool/Cadence/ Synopsys/Industry Equivalent Standard Software

Hardware:
Personal Computer with necessary peripherals, configuration and operating System.

II Sem.	Real Time Operating Systems Lab	Course Code:V25ESVL05	L	T	P	C
			0	0	4	2

Syllabus Details

The Students are required to write the programs using C-Language according to the Experiment requirements using RTOS Library Functions and macros ARM-926 developer kits and ARM Cortex.

The following experiments are required to develop the algorithms, flow diagrams, source code and perform the compilation, execution and implement the same using necessary hardware kits for verification. The programs developed for the implementation should be at the level of an embedded system design.

The students are required to perform at least SIX experiments from Part-I and TWO experiments from Part-II.

List of Experiments:

Part-I:

Experiments using ARM-926 with PERFECT RTOS

1. Register a new command in CLI.
2. Create a new Task.
3. Interrupt handling.
4. Allocate resource using semaphores.
5. Share resource using MUTEX.
6. Avoid deadlock using BANKER‟S algorithm.
7. Synchronize two identical threads using MONITOR.
8. Reader's Write's Problem for concurrent Tasks.

Part-II

Experiments on ARM-CORTEX processor using any open source RTOS.
(Coo-Cox-Software-Platform)

1. Implement the interfacing of display with the ARM- CORTEX processor.
2. Interface ADC and DAC ports with the Input and Output sensitive devices.
3. Simulate the temperature DATA Logger with the SERIAL communication With PC.
4. Implement the developer board as a modem for data communication using Serial port communication between two PC‟s.

Lab Requirements:

Software:

Eclipse IDE for C and C++ (YAGARTO Eclipse IDE), Perfect RTOS Library, COO-COX Software Platform, YAGARTO TOOLS, and TFTP SERVER. LINUX Environment for the compilation using Eclipse IDE & Java with latest Version.

Hardware:

The development kits of ARM-926 Developer Kits and ARM-Cortex Boards.

Serial Cables, Network Cables and recommended power supply for the board.