Annexure-06

Approved Course structure & Syllabus for M. Tech (V21)

COURSE STRUCTURE
AND
DETAILED SYLLABUS

For

M. Tech

(Embedded Systems & VLSI)

Academic Year 2021-2022

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 2021-22

I Semester

Sl. No.	Course Code	Course Name	L	T	P	C
1.	V21ESVT01	System Design through VERILOG	3	-	-	3
2.	V21ESVT02	Embedded Systems Design	3	-	-	3
3.	V21ESVT03 V21ESVT04 V21ESVT05	**ELECTIVE-1** Programming Languages for Embedded Systems Parallel processing System On Chip & Applications	3	-	-	3
4.	V21ESVT06 V21ESVT07 V21ESVT08	**ELECTIVE-II** Digital System Design CPLD & FPGA Architectures And Applications VLSI Signal Processing	3	-	-	3
5.		Research methodology and IPR	2	0	0	2
6.	V21ESVL01	System Design through Verilog Lab	-	-	4	2
7.	V21ESVL02	Embedded Systems Design Lab		-	4	2
8.	Aud. 1	Audit Course 1	2	0	0	0
			16	**0**	**8**	**18**

Total Contact Hours: 24 **Total Credits: 18**

II Semester

Sl. No.	Course Code	Course Name	L	T	P	C
1.	V21ESVT09	Analog and Digital CMOS VLSI Design	3	-	-	3
2.	V21ESVT10	Real Time Operating Systems	3	-	-	3
3.	V21ESVT11 V21ESVT12 V21ESVT13	**ELECTIVE-III** MEMS Technology & Applications Design for Testability Semiconductor Memory Design And Testing	3	-	-	3
4.	V21ESVT14 V21ESVT15 V21ESVT16	**ELECTIVE-IV** Hardware Software Co-Design Embedded Computing Communication Buses and Interfaces	3	-	-	3
5.	V21ESVL03	Analog and Digital CMOS VLSI Design Lab	-	-	4	2
6.	V21ESVL04	Real time Operating Systems Lab		-	4	2
7.	V21ESVL05	Mini project	0	0	4	2
8.	Aud. 2	Audit course 2	2	0	0	MNC
			14	**0**	**12**	**18**

Total Contact Hours: 26 **Total Credits: 18**

III Semester*

Sl. No.	Course Code	Course Name	L	T	P	Credits
1.	V21ESVT17 V21ESVT18 V21ESVT19	1.IOT and its Applications 2.Low Power VLSI Design 3.MOOCs Course	3	0	0	3
2.	V21ESVOE01	1.Operations Research 2.Cost Management of Engineering projects 3. MOOCs Course	3	0	0	3
3.	V21ESVP01	Dissertation phase-I/Industrial Project **(to be continued and evaluated next semester)**	0	0	20	10#
					Total Credits	**16**

Evaluated and Displayed in IV semester Marks list**.**

*Students going for Industrial project/Thesis will complete these courses through MOOCs

IV Semester

Sl. No.	Course Code	Course Name	P.Os	Category	L	T	P	C
1.	V21ESVP02	Project/Dissertation phase-II (continued from III semester)			0	0	32	16
							Total Credits	**16**

Total Credits : 66

Audit course 1&2

1. English for Research paper Writing

2. Disaster Management

3. Value Education

4. Constitution of India

5. Pedagogy Studies

6. Stress Management by Yoga

7. Personality Development through Life Enlightenment Skills

I Semester

SYLLABUS

I Sem.	SYSTEM DESIGN THROUGH VERILOG	Course Code: V21ESVT01	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcomes:

- CO1: Outline basic concepts of RTL code for digital circuits **[K2]**
- CO2: Model RTL codes for digital circuit at gate and data flow level **[K3]**
- CO3: Model RTL codes for digital circuit at behavioural level **[K3]**
- CO4: Model RTL codes for digital circuit at switch level modelling and outline the concepts of task, function and complier directives **[K3]**
- CO5: Analyze Synthesize of Combinational and Sequential Circuits **[K4]**

UNIT-I
INTRODUCTIONTOVERILOG:
Verilog as HDL, Levels of design description, concurrency, module, simulation and synthesis, testbench, functional verification,programminglanguageinterface (PLI),simulationand synthesistools.

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

UNIT-II
GATELEVELMODELLING:
Introduction,AND gate primitive, module structure, other gate primitives, illustrative examples, tristate gates, arrayof instances of primitives, design of Flip flops with gate primitives, delays, strengths and contention resolution, nettypes,designofbasic circuits.

DATA FLOWLEVEL MODELLING
Introduction, continuous assignment structures, delays and continuous assignments, assignment to vectors.

UNIT-III
BEHAVIORALMODELLING:
Introduction, operationsandassignments,initial construct,alwaysconstruct,examples,assignmentswith 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
SWITCHLEVELMODELLING
Basictransistor switches, CMOS switch, Bidirectional gates and time delays with switch primitives, instantiations withstrengthsanddelays,strengthcontentionwithtriregnets, 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

SYNTHESISOFCOMBINATIONAL ANDSEQUENTIAL LOGIC USINGVERILOG: Synthesisof combinational logic: Net list of structured primitives, a set of continuous assignment statements and level sensitivecyclic behavior with examples, Synthesis of priority structures, Exploiting logic don't care conditions. Synthesis ofsequential logic with latches: Accidental synthesis of latches and Intentional synthesis of latches, Synthesis ofsequentiallogicwithflip-flops,Synthesisofexplicitstatemachines.

TEXTBOOKS:

1. DesignthroughVerilogHDL—T.R.PadmanabhanandB. BalaTripuraSundari,WSE, IEEEPress,2004.
2. AdvancedDigitalDesignwithVerilog HDL—MichaelD.Ciletti,PHI,2005.

REFERENCES:

1. FundamentalsofLogicDesignwithVerilog—Stephen.BrownandZvonkoVranesic,TMH,2005.
2. AVerilogPrimier—J.Bhasker,BSP,2003.

I Sem.	EMBEDDED SYSTEM DESIGN	Course Code: V21ESVT02	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:

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.

Cortex-M3 Implementation Overview-the Pipeline, A detailed block diagram, Bus Interfaces on the Cortex-M3,Other Interfaces on the Cortex-M3, the External PPB, Typical Connections,Reset Types and Reset Signals.

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.	Programming Languages for Embedded Systems (Elective-I)	Course Code: V21ESVT03	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcomes:

At the end of this course, students will be able to

- CO1: Write an embedded C application of moderate complexity.
- CO2: Develop and Analyze algorithms in C++.
- CO3: Differentiate interpreted languages from compiled languages.

UNIT-I: Embedded „C" Programming Bitwise operations, Dynamic memory allocation, OS services.Linked stack and queue, sparsematrices, Binary tree. Interrupt handling in C, Code optimization issues.Embedded Software Development Cycle and Methods (Waterfall, Agile)

UNIT-II: Object Oriented Programming Introduction to procedural, modular, object-oriented and genericprogramming techniques, Limitations of procedural programming, objects, classes, data members,methods, data encapsulation, data abstraction and information hiding, inheritance, polymorphism

UNIT-III: CPP Programming: „cin", „cout", formatting and I/O manipulators, new and deleteoperators, Defining a class, data members and methods, „this" pointer, constructors, destructors, friendfunction, dynamic memory allocation

UNIT-IV: Overloading and Inheritance: Need of operator overloading, overloading the assignment,Overloading using friends, type conversions, single inheritance, base and derived classes, friendClasses, types of inheritance, hybrid inheritance, multiple inheritance, virtual base class,Polymorphism, virtual functions.

UNIT-V: Templates: Function template and class template, member function templates and templatearguments, Exception Handling: syntax for exception handling code: try-catch- throw,Multiple Exceptions.Scripting Languages:

Overview of Scripting Languages – PERL, CGI, VB Script, Java Script.
PERL: Operators, Statements Pattern Matching etc. Data Structures, Modules, Objects, TiedVariables, Inter process Communication Threads, Compilation & Line Interfacing.

Text Books:

1. Michael J. Pont, "Embedded C", Pearson Education, 2nd Edition, 2008
2. Randal L. Schwartz, "Learning Perl", O"Reilly Publications, 6th Edition 2011

Reference Books:

1. A. Michael Berman, "Data structures via C++", Oxford University Press, 2002
2. Robert Sedge wick, "Algorithms in C++", Addison Wesley Publishing Company, 1999
3. Abraham Silberschatz, Peter B, Greg Gagne, "Operating System Concepts", John Willey& Sons, 2005Kaufmann.

I Sem.	Parallel Processing (Elective I)	Course Code: V21ESVT04	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcomes:

At the end of this course, students will be able to

- CO1: Identify limitations of different architectures of computer
- CO2: Analysis quantitatively the performance parameters for different Architectures
- CO3: Investigate issues related to compilers and instruction set based on type of Architectures.

UNIT-I: Overview of Parallel Processing and Pipelining, Performance analysis, Scalability

UNIT-II: Principles and implementation of Pipelining, Classification of pipelining processors, Advanced pipelining techniques, Software pipelining

UNIT-III: VLIW processors Case study: Superscalar Architecture- Pentium, Intel Itanium Processor and Ultra SPARC,MIPS on FPGA, Vector and Array Processor, FFT Multiprocessor Architecture

UNIT-IV: Multithreaded Architecture, Multithreaded processors, Latency hiding techniques, Principles of multithreading, Issues and solutions

UNIT-V: Parallel Programming Techniques: Message passing program development, Synchronous and asynchronous message passing, Shared Memory Programming, Data Parallel Programming, Parallel Software Issues. Operating systems for multiprocessors systems customizing applications on parallel processing platforms

Text Books:

1. Kai Hwang, Faye A. Briggs, “Computer Architecture and Parallel Processing”, MGH
International Edition
2. Kai Hwang, “Advanced Computer Architecture”, TMH
3. V. Rajaraman, L. Sivaram Murthy, “Parallel Computers”, PHI.

Reference Books:

1. William Stallings, “Computer Organization and Architecture, Designing for Performance“Prentice Hall, Sixth edition
2. Kai Hwang, ZhiweiXu, “Scalable Parallel Computing”, MGH
3. David Harris and Sarah Harris, “Digital Design and Computer Architecture”, Morgan

I Sem.	System on Chip & Applications (Elective I)	Course Code: V21ESVT05	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:
The student will be able to

- CO1: Describe SOC System Approach, design and its Architecture –**[K2]**
- CO2: Discuss the selection of processor and its micro architecture for SOC –**[K2]**
- CO3: Discuss Memory Design for SOC –**[K2]**
- CO4: Explain the concepts of bus models and Interconnect Architectures –**[K2]**
- CO5: Explain SOC based Applications –**[K2]**

UNIT-I

Introduction to the System Approach System Architecture, Components of the system,Hardware & Software, Processor Architectures, Memory and Addressing.System level interconnection, an approach for SOC Design, System Architecture and Complexity.

UNIT-II

Processors Introduction , Processor Selection for SOC, Basic concepts in Processor Architecture, Basic concepts in Processor Micro Architecture, Basic elements in Instruction handling. Buffers: minimizing Pipeline Delays, Branches, More Robust Processors, Vector Processors and Vector Instructions extensions, VLIW Processors, Superscalar Processors.

UNIT-III

Memory Design for SOC Overview of SOC external memory, Internal Memory, Size, Scratchpads and Cache memory, Cache Organization, Cache data, Write Policies, Strategies for line replacement at miss time, Types of Cache, Split – I, and D – Caches, Multilevel Caches, Virtual to real translation , SOC Memory System, Models of Simple Processor – memory interaction.

UNIT-IV

Interconnect Customization and Configuration Inter Connect Architectures, Bus: Basic Architectures, SOC Standard Buses, Analytic Bus Models, Using the Bus model, Effects of Bus transactions and contention time. SOC Customization: An overview, Customizing Instruction Processor, Reconfiguration Technologies, Mapping design onto Reconfigurable devices, Instance- Specific design, Customizable Soft Processor, Reconfiguration - overhead analysis and trade-off analysis on reconfigurable Parallelism.

UNIT-V

Application / Case Studies:
Zynq system on chip design – Secure Boot, Analog Data Acquisition, System Monitoring using the Zynq-7000 AP SOC Processing System with the XADC AXI Interface.
Cypress- PSoC4- Architecture, GPIO Pins and its applications - down counter, sine wave Generator using PSOC 4 device.

TEXT BOOKS:

1. Computer System Design System-on-Chip - Michael J. Flynn and Wayne Luk, Wiely India Pvt. Ltd.
2. ARM System on Chip Architecture – Steve Furber –2nd Ed., 2000, Addison Wesley Professional.
3. Embedded Processing with the ARM Cortex-A9 on the Xilinx Zynq-7000 All Programmable SoC-Louise H. Crockett Ross A. Elliot Martin A. Enderwitz Robert W. Stewart
4. Cypress PSoC User Manual

REFERENCE BOOKS:

1. Design of System on a Chip: Devices and Components – Ricardo Reis, 1st Ed., 2004, Springer
2. Co-Verification of Hardware and Software for ARM System on Chip Design (Embedded Technology) – Jason Andrews – Newnes, BK and CDROM.
3. System on Chip Verification – Methodologies and Techniques – PrakashRashinkar, Peter Paterson and Leena Singh L, 2001, Kluwer Academic Publishers.

I Sem.	Digital System Design (Elective II)	Course Code: V21ESVT06	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:

The student will be able to

- CO1: Describe the algorithms for minimization of functions
- CO2: Describe the algorithms for minimization of PLDs.
- CO3: Design large scale digital systems.
- CO4: Discuss the fault model and diagnosis in combinational and sequential Circuits.

UNIT-I: Minimization Procedures and CAMP Algorithm

Review on minimization of switching functions using tabular methods, k-map, QM algorithm, CAMP-I algorithm, Phase-I: Determination of Adjacencies, DA, CSC, SSMs and EPCs, CAMPI algorithm, Phase-II: Passport checking, Determination of SPC, CAMP-II algorithm: Determination of solution cube, Cube based operations, determination of selected cubes are wholly within the given switching function or not, Introduction to cube based algorithms.

UNIT-II: PLA Design, PLA Minimization and Folding Algorithms

Introduction to PLDs, basic configurations and advantages of PLDs, PLA-Introduction, Block diagram of PLA, size of PLA, PLA design aspects, PLA minimization algorithm (IISc algorithm), PLA folding algorithm(COMPACT algorithm)-Illustration of algorithms with suitable examples.

UNIT -III: Design of Large Scale Digital Systems

Algorithmic state machine charts-Introduction, Derivation of SM Charts, Realization of SM Chart, control implementation, control unit design, data processor design, ROM design and PAL design aspects, digital system design approaches using CPLDs, FPGAs and ASICs.

UNIT-IV: Fault Diagnosis in Combinational Circuits

Faults classes and models, fault diagnosis and testing, fault detection test, test generation, testing process, obtaining a minimal complete test set, circuit under test methods- Path sensitization method, Boolean difference method, properties of Boolean differences, Kohavi algorithm, faults in PLAs, DFT schemes, built in self-test.

UNIT-V: Fault Diagnosis in Sequential Circuits

Fault detection and location in sequential circuits, circuit test approach, initial state identification, Haming experiments, synchronizing experiments, machine identification, distinguishing experiment, adaptive distinguishing experiments.

TEXT BOOKS:

1. Logic Design Theory-N. N. Biswas, PHI
2. Switching and Finite Automata Theory-Z. Kohavi, 2nd Edition, 2001, TMH
3. Digital system Design using PLDd-Lala

REFERENCE BOOKS:

1. Fundamentals of Logic Design – Charles H. Roth, 5th Ed., Cengage Learning.
2. Digital Systems Testing and Testable Design – MironAbramovici, Melvin A. Breuer and Arthur D. Friedman- John Wiley & Sons Inc.

I Sem.	CPLD & FPGA Architectures and Applications (Elective II)	Course Code: V21ESVT07	L	T	P	C
			3	0	0	3

Syllabus Details

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, theActel 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/ SamihaMourad, 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.	VLSI Signal Processing (Elective II)	Course Code:V21ESVT08	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.

I Sem.	System Design through Verilog Lab	Course Code: V21ESVL01	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: V21ESVL02	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.

II Semester
SYLLABUS

II Sem.	Analog and Digital CMOS VLSI Design	Course Code: V21ESVT09	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcomes:

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

CO1: Describe the concept of MOS structure and physical design of CMOS **(K2)**

CO2: Design the CMOS Inverters and various CMOS combinational logic circuits **(K4)**

CO3: Design the CMOS different Sequential logic circuits **(K4)**

CO4: Describe the concept of modelling of MOS and Analog CMOS Sub-Circuits **(K2)**

CO5: Describe the CMOS Op-Amps & it's Applications. **(K2)**

UNIT-I: **Review of MOS structures and Physical design flow:**
Basic MOS structure and its static behaviour, Quality metrics of a digital design: Cost,Functionality, Robustness, Power, and Delay, Stick diagram and Layout, Wire delay models.Physical design flow: Floor planning, Placement, Routing, CTS, Power analysis and IR drop estimation-static and dynamic.

UNIT-II CMOS INVERTER AND COMBINATIONAL LOGIC:
Inverter: Static CMOS inverter, Switching threshold and noise margin concepts and theirEvaluation, Dynamic behaviour, Power consumption. Combinational logic: Static CMOSdesign, Logic effort, Rationed logic, Pass transistor logic, Dynamic logic, Speed and powerdissipation in dynamic logic, Cascading dynamic gates, CMOS transmission gate logic.

UNIT-III SEQUENTIAL LOGIC:
Static latches and registers, Bi-stability principle, MUX based latches, Static SR flip-flops,Master-slave edge-triggered register, Dynamic latches and registers, Concept of pipelining,Pulse registers, and Non-bistable sequential circuit.

UNIT -IV CMOS MODELING AND ANALOG SUB- CIRCUITS
CMOS Device Modelling - Simple MOS Large-Signal Model, Other Model Parameters,Small-Signal Model for the MOS Transistor, Sub-threshold MOS Model. MOS Switch, MOSDiode, MOS Active Resistor, Current Sinks and Sources, Current Mirrors-Cascode currentMirror and Wilson Current Mirror, Current and Voltage References, Band gap Reference.

UNIT-V CMOS AMPLIFIERS:
Inverters- Active load inverter, current source inverter, push-pull inverter, DifferentialAmplifiers- large signal analysis, small signal analysis, Cascode Amplifiers, CurrentAmplifiers, Output Amplifiers, Design of CMOS Op Amps, Compensation of Op Amps,Design of Two-Stage Op Amps, Power- Supply Rejection Ratio of Two-Stage Op Amps,Cascode Op Amps, Characterization of Comparator, Two-Stage comparator design.

II Sem.	Real Time Operating Systems	Course Code: V21ESVT10	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcomes
Upon the completion of the course student will be able to

- CO1: Illustrate real time programming concepts.
- CO2: Apply RTOS functions to implement embedded applications
- CO3: Understand fundamentals of design consideration for embedded

Applications.

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

UNIT-III

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, Scheduleability 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”, WileyInterscience.
2. Hermann Kopetz, (1997), “Real-Time Systems: Design Principles for Distributed EmbeddedApplications”, Kluwer.
3. Insup Lee, Joseph Leung, and Sang Son, (2008) “Handbook of Real-Time Systems”, Chapman andHall.Krishna and Kang G Shin, (2001), “Real-Time Systems”, McGraw Hill.

II Sem.	MEMS Technology and its Applications (Elective-III)	Course Code: V21ESVT11	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:
The student will be able to

- CO1: Describe the concepts of MEMS and Microsystems.
- CO2: Describe various possible materials for MEMS based devices.
- CO3: Describe various process steps involved in fabrication of MEMS devices.
- CO4: Describe various micro sensors and micro actuators.
- CO5: Describe various MEMS devices and their applications.

UNIT-I:

MEMS AND MICROSYSTEM
Introduction to MEMS, Microsystems and microelectronics, Multidisciplinary nature of MEMS, Miniaturization and its Benefits, Scaling laws in Miniaturization, MEMS Design Considerations, Advantages of MEMS Technology, Applications of MEMS

UNIT-II:

MATERIALS FOR MEMS
Introduction, Substrates & wafers, Active Substrate Materials, Silicon as a Substrate Material, Silicon Compounds, Piezoelectric Crystals, Polymers, Packaging Materials.

UNIT-III:

MICROFABRICATION
Introduction, Fabrication Process – Wafer processing, Photolithography, Ion implantation, Oxidation, Chemical vapor deposition (CVD), Physical vapor deposition, Deposition by Epitaxy, Etching, Manufacturing Process -Bulk Micromachining, Surface Micromachining and LIGA Process, Packaging technology, System level packaging, single and multichip packaging. Microsystem packaging, interfacingsin Microsystem packaging.

UNIT-IV:

MEMS BASED SENSORS AND ACTUATORS
Introduction, working principles of Microsystem - Micro Sensors, Micro Actuators and MEMS with Micro sensors: Pressure sensors, Temperature sensors, Humidity sensors, Accelerometers, Gyroscopes, Biomedical Sensors, Chemical sensors, MEMS with micro actuators: Microgrippers, Micromotors, Micro gears and Micropumps. Microfluidics.

UNIT-V:
RF MEMS
RF MEMS devices: Switch parameters- Basics of switching - Mechanical Switches- Electronic switches for RF and microwave applications – Approaches for low-actuation-voltage switches, MEMS based Reconfigurable Antennas, Reconfigurable Filters and Phase shifters.

Textbooks:

1. Tai-Ran Hsu, MEMS and Microsystems: Design, Manufacture, and Nanoscale Engineering, 2nd Edition, John Wiley & Sons, Inc., Hoboken, New Jersey, 2008.
2. Gabriel M Rebeiz, "RF MEMS - Theory Design and Technology", John Wiley, 2004
3. Microsystem Design by Stephen D. Senturia, Springer International, Edition,2010.

Reference Books:

1. Marc Madou, ―Fundamentals of Micro Fabrication‖ CRC Press
2. Mohamed Gad-el-Hak, ―The MEMS Handbook‖, CRC Press
3. Julian W.Gardner, Vijay K.Varadan, Osama O. AwadelKarim, "Micro sensors MEMS and Smart Devices", John Wiby& sons Ltd., 2001.
4. Iannacci, J. (2013). *Practical guide to RF-MEMS*. John Wiley & Sons.

II Sem.	Design for Testability (Elective-III)	Course Code: V21ESVT12	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:

The students will be able to

- CO1: Interpret the concepts of modelling digital circuits and simulation.
- CO2: Describe modelling of faults and its testing for SSF.
- CO3:Explain various techniques of testing.

UNIT-I: Modeling:

Modeling digital circuits at logic level, register level and structural level. Levels of modeling.

Logic Simulation: Types of simulation, delay models, element evaluation, hazard detection, gate level event driven simulation.

UNIT-II: Fault Modeling:

Logic fault models, fault detection and redundancy, fault equivalence and fault location. Single stuck and multiple stuck – fault models. Fault simulation applications, general techniques for combinational circuits.

UNIT-III: Testing for Single Stuck Faults (SSF):

Automated test pattern generation (ATPG/ATG) for SSFs in combinational and sequential circuits, functional testing with specific fault models. Vector simulation – ATPG vectors, formats, compaction and compression, selecting ATPG tool.

UNIT-IV: Design for Testability:

Testability trade-offs techniques. Scan architectures and testing – controllability and observeability, generic boundary scan, fully integrated scan, storage cells for scan design. Board level and system level DFT approaches. Boundary scans standards. Compression techniques – different techniques, syndrome test and signature analysis.

UNIT-V: Built-in-Self-Test (BIST):

BIST concepts and test pattern generation, specific BIST architectures – CSBL, BEST, RTS, LOCST, STUMPS, CBIST, RTD, SST, CATS, CSTP, BILBO. Brief ideas on some advanced BIST concepts and design for self-test at board level.

Reference Books

1. MironAbramovici, Melvin A.Breur, Arthur D.Friedman, Digital Systems Testing and Testable Design, Jaico Publishing House, 2001.
2. Michael L.Bushnell, VishwaniD.Agrawal, Essentials of Electronic Testing, Springer, 2000.
3. Michael D.Ciletti, Modeling, Synthesis, and Rapid Prototyping with the Verilog HDL. Prentice Hall, 1999.

II Sem.	Semiconductor Memory Design and Testing (Elective-III)	Course Code: V21ESVT13	L	T	P	C
			3	0	0	3

Syllabus Details

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.	Hardware Software Co-Design (Elective-IV)	Course Code:V21ESVT14	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:

The students will be able to

- CO1: Describe co-design architectures, methods and algorithms.
- CO2: Describe prototyping emulation and target architecture using embedded Systems.
- CO3: Explain the compilation techniques.
- CO4: Distinguish the various design specifications and verifications.
- CO5: Describe the system level specifications and design using languages.

UNIT-I: Co- Design Issues:

Co- Design Models, Architectures, Languages, A Generic Co-design Methodology.

Co- Synthesis Algorithms Hardware software synthesis algorithms: hardware – software partitioning distributed system co-synthesis.

UNIT-II:

Prototyping and Emulation Prototyping and emulation techniques, prototyping and emulation environments, future developments in emulation and prototyping architecture specialization techniques, system communication infrastructure

Target Architectures Architecture Specialization techniques, System Communication infrastructure, target Architecture and Application System classes, Architecture for control dominated systems (8051-Architectures for High performance control), Architecture for Data dominated systems (ADSP21060, TMS320C60) and Mixed Systems.

UNIT-III:

Compilation Techniques and Tools for Embedded Processor Architectures Modern embedded architectures, embedded software development needs, compilation technologies and practical consideration in a compiler development environment.

UNIT-IV:

Design Specification and Verification Design, co-design, the co-design computational model, concurrency coordinating con current computations, interfacing components, design verification, implementation verification, verification tools, interface verification.

UNIT-V:

Languages for System-Level Specification and Design-I System-level specification, design representation for system level synthesis, system level specification languages.

Languages for System-Level Specification and Design-II

Heterogeneous specifications and multi-language co-simulation, the cosyma system and lycos system.

TEXT BOOKS:

1. Hardware / Software Co- Design Principles and Practice – JorgenStaunstrup, Wayne
Wolf – 2009, Springer.
2. Hardware / Software Co- Design - Giovanni De Micheli, MariagiovannaSami, 2002,
Kluwer Academic Publishers.

REFERENCE BOOKS:

1. A Practical Introduction to Hardware/Software Co-design -Patrick R.Schaumont -
2010 – Springer Publications.

II Sem.	Embedded Computing (Elective-IV)	Course Code:V21ESVT15	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:

The students will be able to

- CO1: Understand the concepts of Linux OS programming –**[K2]**
- CO2: Describe the different software development tools **[K2]**
- CO3: Explain different interfacing modules – **[K2]**
- CO4: Discuss the networking basics –**[K2]**
- CO5: Explain the basic concepts of LPC17xx Microcontroller –**[K2]**

UNIT-I

Programming on Linux Platform:

System Calls, Scheduling, Memory Allocation, Timers, Basics of Embedded Linux, Root File System, Busy Box.

UNIT-II

Introduction to Software Development Tools

GNU GCC,make, gdb, static and dynamic linking, C libraries, compiler options, code optimization switches.

UNIT-III

Interfacing Modules

Sensor and actuator interface, data transfer and control, GSM module interfacing with data processing and display.

UNIT-IV

Networking Basics

Sockets, ports, UDP, TCP/IP, client server model, socket programming.

UNIT-V

LPC 17xx microcontroller- Internal memory, GPIOs, Timers, ADC, UART and other serial interfaces, PWM, RTC, WDT.

TEXT BOOKS:

1. Modern Embedded Computing - Peter Barry and Patrick Crowley, 1st Ed., Elsevier/Morgan Kaufmann, 2012.
2. Linux Application Development - Michael K. Johnson, Erik W. Troan, Adission Wesley, 1998.
3. Assembly Language for x86 Processors by Kip R. Irvine

REFERENCE BOOKS:

1. Operating System Concepts by Abraham Silberschatz, Peter B. Galvin and Greg Gagne.
2. Technical references and user manuals on www.arm.com.
3. The Design of the UNIX Operating System by Maurice J. Bach Prentice-Hall
4. UNIX Network Programming by W. Richard Stevens.

II Sem.	Communication Buses and Interfaces (Elective-IV)	Course Code: V21ESVT16	L	T	P	C
			3	0	0	3

Syllabus Details

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

- CO1: Select a particular serial bus suitable for a particular application.
- CO2: Develop APIs for configuration, reading and writing data onto serial bus.
- CO3: Design and develop peripherals that can be interfaced to desired serial bus.

UNIT-I
Serial Busses- Cables, Serial busses, serial versus parallel, Data and Control Signal- data frame, data rate, features Limitations and applications of RS232, RS485, I2C , SPI

UNIT-II: CAN
ARCHITECTURE- ISO 11898-2, ISO 11898-3, Data Transmission- ID allocation, Bit timing, Layers- Application layers, Object layer, Transfer layer, Physical layer, Frame formats- Data frame, Remote frame, Error frame, Over load frame, Ack slot, Inter frame spacing, Bit spacing, Applications.

UNIT-III: PCIe
Revision, Configuration space- configuration mechanism, Standardized registers, Bus enumeration, Hardware and Software implementation, Hardware protocols, Applications.

UNIT-IV: USB
Transfer Types- Control transfers, Bulk transfer, Interrupt transfer, isochronous transfer. Enumeration- Device detection, Default state, addressed state, Configured state, enumeration sequencing. Descriptor types and contents- Device descriptor, configuration descriptor, Interface descriptor, Endpoint descriptor, String descriptor.Device driver.

UNIT-V
Data streaming Serial Communication Protocol- Serial Front Panel Data Port(SFPDP) configurations, Flow control, serial FPDP transmission frames, fiber frames and copper cable.

TEXTBOOKS

1. A Comprehensive Guide to controller Area Network – Wilfried Voss, Copperhill Media
Corporation, 2nd Ed., 2005.
2. Serial Port Complete-COM Ports, USB Virtual Com Portsand Ports for Embedded Systems- JanAxelson, Lakeview Research, 2nd Ed.,

REFERENCES

1. USB Complete – Jan Axelson, Penram Publications.
2. PCI Express Technology – Mike Jackson, Ravi Budruk, Mindshare Press.

II Sem.	Analog and Digital CMOS VLSI Design Lab	Course Code: V21ESVL03	L	T	P	C
			0	0	4	2

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

- CO1 -Analyse the Characteristics of MOS Device **(K3)**
- CO2 -Analyse the basic MOS Amplifiers and current mirrors **(K3)**
- CO3 -Design the various MOS Amplifiers. **(K4)**
- CO4 -Demonstrate various CMOS combinational Digital circuits **(K2)**
- CO5- Demonstrate various CMOS Sequential Digital circuits **(K2)**

The students are required to design and implement the Circuit and Layout of any 10Experiments using CMOS 130nm Technology with Mentor Graphics Tool/Cadence/ Synopsys/Industry Equivalent Standard Software.

List of Experiments:

1. MOS Device Characterization and parametric analysis
2. Common Source Amplifier
3. Common Source Amplifier with source degeneration
5. Simple current mirror
6. Cascade current mirror.
7. Wilson current mirror.
8. Differential Amplifier.
8. Full Adder
9. RS-Latch
10. Clock Divider
11. JK-Flip Flop
12. Synchronous Counter
13. Asynchronous Counter
14. Static RAM Cell

II Sem.	Real Time Operating Systems Lab	Course Code:V21ESVL04	L	T	P	C
			0	0	4	2

Syllabus Details

- The Students are required to write the programs using C-Language according to the Experimentrequirements using RTOS Library Functions and macros ARM-926 developer kits and ARMCortex.
- The following experiments are required to develop the algorithms, flow diagrams, source codeand perform the compilation, execution and implement the same using necessary hardware kitsfor verification. The programs developed for the implementation should be at the level of anembedded system design.
- The students are required to perform at least SIX experiments from Part-I and TWO experimentsfrom 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'sWrite'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.

III Semester
SYLLABUS

III Sem.	IoT and its Applications	Course Code: V21ESVT17	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome: The student will be able to

- CO1: Describe M2M and IOT Technologies. **[K2]**
- CO2: Explain the layers, protocols and communication technologies in IOT. **[K2]**
- CO3: Illustrate various hardware components required for IOT applications**. [K2]**
- CO4: Discuss the cloud technologies and their services. **[K2]**
- CO5: Explain the IoT Applications. **[K2]**

UNIT-I INTRODUCTION [1, 2]
Introduction from M2M to IoT - An Architectural Overview, building architecture, Main design principles and needed capabilities, An IoT architecture outline, M2M and IoT Technology Fundamentals; Sensors, Actuators, RFID, Wireless Sensor Networks, Devices and gateways.

UNIT-II IOT PROTOCOLS & COMMUNICATION TECHNOLOGIES [2, 4]
Functionalityof Layers in IoT, IoT Connectivity – IEEE 802.15.4, Wi-Fi, Bluetooth, Zigbee, LPWAN, 5G.
Study of protocols - 6LoWPAN, CoAP, MQTT.

UNIT-III DESIGN AND DEVELOPMENT [3, 4]
Design Methodology, Embedded computing logic, IoT system building blocks, Raspberry Pi - Board details, sensor/actuator Interface using Python Programming.

UNIT-IV Cloud Computing [3, 4]
Structured Vs. Unstructured Data and Data in Motion Vs. Data in Rest, Role of Machine Learning; Data Collection, Storage and Computing Using a Cloud Platform for IoT Applications/Services, AWS for IoT – Introduction to Amazon EC2.

UNIT-V IOT APPLICATIONS [2, 3]
CASE STUDIES/INDUSTRIAL APPLICATIONS: Case Studies - Home appliances, Smart and Connected Cities, Public Safety, Agriculture, Introduction to Industry 4.0.

TEXTBOOKS:

1. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, StamatisKarnouskos,
David Boyle, “From Machine-to-Machine to the Internet of Things: Introduction to a New
Age of Intelligence”, 1st Edition, Academic Press, 2014.
2. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet
Of Things, David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry and Cisco Press 800 East 96th Street Indianapolis, Indiana 46240 USA
3. Daniel Minoli, “Building the Internet of Things with IPv6 and MIPv6: The Evolving World
Of M2M Communications”, ISBN: 978-1-118- 47347-4, Willy Publications
4. IOT (Internet of Things) Programming: A Simple and Fast Way of Learning IOT by David
Etter(Author)
5. Internet of Things - By Raj Kamal, McGraw-Hill Education. Copyright.

REFERENCE BOOKS:

1. From Internet of Things to Smart Cities: Enabling Technologies - edited by Hongjian Sun, Chao Wang, Bashar I. Ahmad, CRC Press -2018.
2. Peter Waher, "Learning Internet of Things", PACKT publishing, BIRMINGHAM
3. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer
4. Vijay Madisetti and ArshdeepBahga, "Internet of Things (A Hands-on- Approach)", 1st Edition, VPT, 2014.

III Sem.	Low Power VLSI Design	Course Code: V21ESVT18	L	T	P	C
			3	0	0	3

Syllabus Details

Course Outcome:

The students will be able to

- CO1: Identify various sources of power consumption
- CO2: Estimate the power consumption sing simulation and probabilistic

Approaches.

- CO3: Discuss low power design at various levels of abstraction.
- CO4: Discuss clock distribution for low power dissipation.

UNIT-I: Introduction

Need for low power VLSI chips, Sources of power dissipation. Emerging Low power approaches. Dynamic dissipation in CMOS, Transistor sizing & gate oxide thickness, Impact of technology Scaling, Technology & Device innovation.

UNIT-II: Power estimation Simulation Power analysis:

SPICE circuit simulators, gate level logic simulation, capacitive power estimation, static state power, gate level capacitance estimation, architecture level analysis, data correlation analysis in DSP systems. Monte Carlo simulation.

Probabilistic power analysis:

Random logic signals, probability & frequency, probabilistic power analysis techniques, signal entropy.

UNIT-III: Low Power Design Circuit level:

Power consumption in circuits. Flip Flops & Latches design, high capacitance nodes, low power digital cells library

Logic level:

Gate reorganization, signal gating, logic encoding, state machine encoding, pre-computation logic

UNIT-IV: Low power Architecture & Systems:

Power & performance management, switching activity reduction, parallel architecture with voltage reduction, flow graph transformation, low power arithmetic components, low power memory design.

UNIT-V: Low power Clock Distribution:

Power dissipation in clock distribution, single driver vs. distributed buffers, Zero skew vs. tolerable skew, chip & package co design of clock network

Algorithm &architectural level methodologies: Introduction, design flow, Algorithmic level analysis & optimization, Architectural level estimation & synthesis.

TEXTBOOKS:

1. Gary K. Yeap, “Practical Low Power Digital VLSI Design”, KAP, 2002

2. Rabaey, Pedram, “Low power design methodologies” Kluwer Academic, 1997

REFERENCES BOOKS:

1. Kaushik Roy, Sharat Prasad, “Low-Power CMOS VLSI Circuit Design” Wiley, 2000