NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL



Honors in

Embedded Systems

(for students of B.Tech., ECE)

(w.e.f. 2021-22)

Department of

Electronics and Communication Engineering

SCHEME OF INSTRUCTION

Honors Program – Embedded Systems

S.No	Course Code	Course Title	L	T	P	Credits	Sem
1	EC5102	Advanced Digital System Design	3	0	0	3	V
2	EC5103	Embedded System Design	3	0	0	3	V
3	EC5153	Hardware/Software Co-design	3	0	0	3	VI
4	EC 5171	Advanced Computer Architecture	3	0	0	3	VI
5	EC 5113	FPGA System Design	3	0	0	3	VII
6	EC 5170	IOT for Industrial Applications	3	0	0	3	VIII
		Total				18	

Course Code: EC5102	ADVANCED DIGITAL SYSTEM DESIGN	Credits 3-0-0:3

Course Outcomes:

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

CO	Analyze the digital circuits for different applications
CO	Develop Specifications for digital systems
CO	Design and develop the digital circuits using VHDL and Verilog
CO	Develop test strategies for digital systems
CO	Design robust digital systems

Detailed Syllabus:

Chapter-1INTRODUCTION: Digital System Design Process, EDA tools and design viewpoints, Behavioral, dataflow, and gate level descriptions.

Chapter-2 HARDWARE DESCRIPTION LANGUAGES: VHDL and Verilog modeling concepts, Behavioral and Structural architecture descriptions: Concurrent and Sequential statements, Event driven Simulation.

Chapter-3 BUILDING BLOCKS FOR DIGITAL SYSTEMS: Tristate buffers, multiplexers, latches, flip-flops, registers, counters, arithmetic and logic circuits (ALU Design), Finite State Machines.

Chapter-4 DESIGN METHODOLOGY: Synchronous Systems, Top-Down Design, Register Transfer Level Design, Algorithmic State Machines, and Synthesis from VHDL.

Chapter-5 IMPLEMENTATION ISSUES: Design Case Studies**,** Hardware Testing & Design for testability, test vectors, fault analysis for combinational and sequential circuits. Timing and signal conventions, synchronization, Signaling Circuits: Terminations, Transmitter and receiver circuits. Microprogramming, Micro-programme sequencer 2910. Introduction to Programmable Logic Devices (FPGAs, CPLDs).

Learning Resources:

Text Books:

1. William I Fletcher: An Engineering approach to Digital Design, Eastern Economy Edition, PHI Limited, 2000
2. Digital System Engineering: William J Dally and John W Poulton Published by Cambridge University Press
3. A VHDL Primer: Jayaram Bhasker Published by Prentice-Hall India
4. Digital Design and Verilog fundamentals: Joseph Cavanaugh Published by CRC Press.
5. The Art of Digital Design: An Introduction to Top down design: Franklin P Processer and David E Winkel Published by Prentice-Hall.
6. Digital Systems Design: Charles H.Roth,Jr., LIzy Kurian John.

Course Code: EC5103	Embedded System Design	Credits 3-0-0: 3

Course Outcomes:

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

CO1	Understand the evolution of microprocessors and microcontrollers and its architectures
CO2	Understand the evolution and architectures of ARM processors.
CO3	Analyse and understand the instruction set and development tools of ARM
CO4	Understand the architectural features of ARM cortex M4 microcontrollers.
CO5	Understand the exception, interrupts and interrupt handling schemes
CO6	Understand the hardware and interfacing peripheral devices to ARM cortex M4

Detailed Syllabus:

Chapter 1 Introduction to Embedded Systems:
Embedded systems Overview, Characteristics of embedded computing applications. Design Challenges, Common Design Metrics, Embedded systems Design flow. Processor Technology, IC Technology, Trade-offs.

Chapter 2 Specification Techniques:
State charts, Modeling Hierarchy, Specification Description Language (SDL), Petri Nets, embedded systems modeling with Petri Nets, Unified Modeling Language (UML), Activity diagram, class diagram, component diagram, use-case diagram, sequence diagram. UML specification examples.

Chapter 3 Introduction to ARM Processors:
Introduction to ARM processors, Evolution of ARM processors, pipeline organization, ARM Processor cores and CPU cores. Introduction to ARM Cortex-M Processors, ARM Cortex-M4 processor‘s architecture, Programmer‘s model, Special registers, Operation Modes, Memory map, Memory access attributes and overview of Interrupts and exceptions.

Chapter 4 ARM Cortex-M4 programming: Assembly basics, Instruction set, Data transfer, Data processing, conditional and branch instructions, barrier and saturation operations, Cortex-M4-specific instructions,Thumb2 instructions, Keil Microcontroller Development Kit for ARM, Typical program compilation flow, Sample arithmetic and logical assembly language programs

Chapter 5 Embedded Systems Interfacing: Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I2C),RS-232,Universal Serial Bus(USB),CAN, IrDA, Bluetooth, PCI and AMBA bus protocols.

Chapter 6 Cortex-M4 Implementation and applications: Detailed block diagram, Bus interfaces oncortex-M4, External PPB interface, typical connections, reset types and signals. Getting started with µ Vision. Applications: Flashing of LEDS using Shift Register, Interfacing stepper motor, Interfacing temperature sensor, Interfacing ADC, Interfacing Real Time Clock, Interfacing of Analog Keypad.

Learning Resources:

Text Books:

1. Joseph Yiu, The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors, Newnes Publications; 3rd edition, 2013.
2. Ata Elahi-Trever Arjeski, ―ARM Assembly language with hardware experiment‖, Springer Int. Publishing, 2015.
3. Santanu Chattopadhya, ―Embedded System Design‖, PHI, 2nd Edition.
4. Frank Vahid and Tony Givargis, ―Embedded System Design‖, John Wiley & sons Inc.3rd Edition.

Reference Books:

1. Wrox, ― Professional Embedded ARM Development‖
2. William hohl and Christoper Hinds, ―ARM assembly language fundamentals and Techniques CRC, 2nd edition,2015.

Course Code: EC5153	HARDWARE / SOFTWARE CO- DESIGN	Credits 3-0-0: 3

Course Outcomes:

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

CO 1	Understand Hardware/Software Co-design
CO	Model the data flow and implement the same through software and hardware.
CO	Design the Control Flow on Transistor Structures.
CO	Understand the design principles in SoC Architecture
CO	Design CORDIC and Crypto coprocessor.

Detailed Syllabus:

The Nature of Hardware and Software: Introducing Hardware/Software Co-design, The Quest for Energy Efficiency, The Driving Factors in Hardware/Software Co-design, The Dualism of Hardware Design and Software Design.

Data Flow Modeling and Transformation: Introducing Data Flow Graphs, Analyzing Synchronous Data Flow Graphs, Control Flow Modeling and the Limitations of Data Flow, Transformations.

Data Flow Implementation in Software and Hardware: Software Implementation of Data Flow, Hardware Implementation of Data Flow, Hardware/Software Implementation of Data Flow.

Analysis of Control Flow and Data Flow: Data and Control Edges of a C Program, Implementing Data and Control Edges, Construction of the Control Flow Graph 4.4 Modern Bipolar, Transistor Structures, Construction of the Data Flow Graph.

Finite State Machine with Data path: Cycle-Based Bit-Parallel Hardware, Hardware Modules, Finite State Machines with Data path, FSMD Design Example: A Median Processor.

System on Chip: The System-on-Chip Concept, Four Design Principles in SoC Architecture, SoC Modeling in GEZEL. Applications: Trivium Crypto-Coprocessor, CORDIC Coprocessor.

Learning Resources:

Text Books:

1. Patrick Schaumont, A Practical Introduction to Hardware/Software Co-design, Springer, 2010.
2. Ralf Niemann, Hardware/Software Co-Design for Data flow Dominated Embedded Systems, Springer, 1998.

Course Code: EC5171	ADVANCED COMPUTER ARCHITECTURE	Credits 3-0-0: 3

Course Outcomes:

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

CO1	Understand the micro-architectural design of processors.
CO2	Evaluate performance of different architectures with respect to various parameters
CO3	Analyze performance of different ILP techniques
CO4	Identify cache and memory related issues in multi-processors
CO5	Understand the concept of NOC

Detailed Syllabus:

Introduction to high performance computing, RISC philosophy and overview of pipelined architecture. Performance evaluation of pipelined architecture. Limitations of scalar pipelines, Instruction level parallelism.

Superscalar architecture: Instruction flow optimization, Handling branches, Advanced optimization in instruction flow, register flow techniques: Register renaming and out of order execution.

Advanced data flow techniques: Instruction reuse and value prediction, Memory data flow, advanced memory data flow architectures, performance evaluation of superscalar architectures.

Multi-threading, Simultaneous multithreaded (SMT) architectures, SMT architecture: Choices,

SMT performance on various designs, SMT architecture: OS impact and adaptive architectures

Multi scalar architecture, Multi-core Architectures, Multi-core Interconnect–NOC, Network-on-Chip, Cache Coherence, Cache Consistency model, Dynamic Core architectures, GP-GPU Architecture, CPU-GPU Integration.

Learning Resources:

Text Books:

1. J.L. Hennessy, and D.A. Patterson, Computer Architecture: A quantitative approach, Fifth Edition, Morgan Kaufman Publication, 2012.
2. J.P. Shen and M.H. Lipasti, *Modern Processor Design*, MC Graw Hill, Crowfordsville, 2005.

Course Code: EC5113	FPGA SYSTEM DESIGN	Credits 3-0-0: 3

Course Outcomes:

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

CO1	Understand FPGA design flow.
CO2	Understand the building blocks of commercially available FPGA/CPLDs
CO3	Develop VHDL/Verilog models and synthesize targeting for Vertex, Spartan FPGAs
CO4	Develop parameterized library cells and implement system designs using parameterized cells

Detailed Syllabus:

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.

FPGAs/CPLDs: Programming Technologies, Commercially available FPGAs: Xilinx‘s Vertex and Spartan, Actel‘s FPGA, Altera‘s FPGA/CPLD, 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.

CASE STUDY – Applications using Kintex-7, Virtex-7, Artix-7

Learning Resources:

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, ZvonkoG. Vranesic, Field Programmable Gate Arrays, 2nd Edition, Springer, 1992.

Course Code:5170	IOT FOR INDUSTRIAL APPLICATIONS	Credits 3-0-0: 3

Course Outcomes:

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

CO1	Understand IOT design requirements
CO2	Compare various technologies and protocols
CO3	Study storage and intelligent analytics
CO4	Analyze security requirements along with threat model
CO5	Design and experiment various use cases

Detailed Syllabus:

Architectures: IoT and the connected world, Architecture of IoT, Security issues, Opportunities for IoT.

The Web of Things: Linked data, Enterprise data, Importance of security, privacy, and authenticity, Industry standards, Web of Things layer as the driver for IoT systems.

Lessons from the Internet: Relevance of Internet to network of Things, network management, security, mobility and longevity, desirable features of a distributed architecture for a system of things.

Technologies: Wireless protocols, Connectivity options, Low-power design, range extension techniques, data-intensive IoT.

Data storage and analysis: Managing high rate sensor data, Processing data streams, Data consistency in an intermittently connected or disconnected environment, Identifying outliers and anomalies.

Security in IOT: Threat models, Defensive strategies and examples

Use cases: Smart Buildings, Smart health, Home automation, Location tracking

Smart Cities: Collection of information including opportunistic sensing, crowd sensing, and adhoc sensing Response of the system including analytics and optimization, distributed action, people as intelligent actuators, the risk for cyber-attacks in centralized and distributed systems, MAC Layer Protocols for IoT.

Learning Resources:

Text Books:

1. Designing the Internet of Things Adrian McEwen, Hakim Cassimally Wiley 2013
2. Enterprise IoT Naveen Balani Create Space Independent Publishing Platform 2016