ACADEMIC REGULATIONS FOR B.TECH PROGRAMME UNDER AUTONOMOUS STATUS

(W.E.F. THE ADMITTED BATCH OF 2020-21)

HONORS/ MINOR PROGRAMME FRAMEWORK:

i) A candidate shall be eligible to register for Honor or Minor degree along with regular B.Tech degree. A candidate shall earn 20 credits in addition to the 160 credits to get Honor / Minor degree along with regular B.Tech degree. A candidate shall be permitted to register either for Honors or for Minor and not for both simultaneously.

ii) A candidate shall be permitted to register for Honors / Minor program at the beginning of4th semester subject to a maximum of two additional courses per semester, provided that the student must have acquired a minimum of 8.00 CGPA up to the end of 2ndsemester without any backlogs. In case of the declaration of the 3rd semester results after the commencement of the 4th semester and if a candidate fails to score the required minimum of 8.00 CGPA, his/her registration for Honors / Minor Programme stands cancelled and he/she shall continue with the regular Programme.

iii) In case a student fails to meet the CGPA requirement for Degree with Honors / Minor at any point after registration, he/she will be dropped from the list of students eligible for Degree with Honors / Minors and they will receive regular B.Tech degree only. However, such students will receive a separate grade sheet mentioning the additional courses completed by them.

iv) Honors / Minor must be completed simultaneously with a major degree program. A student cannot earn Honors/ Minor after he/she has already earned bachelor's degree.

v) A Candidate iseligibletooptforHonorsProgrammeofferedbytheconcernedDepartment/Discipline and he/she will be awarded B.Tech. (Honors) in the concerned Discipline.

vi) Candidates who are desirous of pursuing their special interest areas in chosen discipline of Engineering may opt for additional courses in minor specialization groups(Specialized Tracks) offered by the concerned department and he/she will get Major degree of concerned Discipline with minor degree of Specialized Track.

vii) Candidates who are desirous of pursuing their special interest areas other than the chosen discipline of Engineering may opt for additional courses in minor specialization groups (General Tracks) offered by the department other than their parent department and he/she will get Major degree of concerned Discipline with minor degree in other department.

viii) CandidatescanalsooptforIndustryrelevanttracksofanybranchlikeDataMiningtrack,IOTtrack,Machinelearningtracketcorindustrytrackssuchas ArtificialIntelligence (AI), Machine Learning (ML), Data Science (DS), Robotics, Electric vehicles, VLSI etc. to obtain the Minor Degree and he/she will get Major degree of concerned discipline with minor degree in industry track.

ix)In the case of Honors, out of 20 additional Credits to be acquired, 16 credits shall be earned by undergoing specified courses listed as pools, with four courses, each carrying4 credits. The

remaining 4 credits must be acquired through two MOOCs courses, which shall be domain specific, each with 2 credits and with a minimum duration of 8 weeks as recommended by the Board of studies.If the MOOC course is a pass course without anygrades,the gradeto beassignedas decidedby the AcademicCouncil.

x)In the case of Minor, out of 20 additional Credits to be acquired, 16 credits shall beearnedbyundergoingspecifiedcourseslistedbytheconcernedBoSalongwithprerequisites. It is the responsibility of the student to acquire/completeprerequisitebefore taking the respective course.A student shall be permitted to choose only thosecoursesthathe/shehasnotstudiedinanyformduringtheProgramme.Theremaining4 credits must be acquired through two MOOCs courses. The courses must be ofminimum 8 weeks induration. Student has to acquire a certificate from the agenciesapproved by the BoS with grading or marks or pass. If the MOOC course is a passcourse without any grades, the grade to be assigned as decided by the AcademicCouncil.

xi)If a candidate drops (or terminated) from the Honors / Minor program, they cannotconvert the earned credits into free or core electives; they will remain extra. Theseadditional courses will find mention in the transcript (but not in the degreecertificate). Insuch cases, the student may choose between the actual grade or a "pass (P)" gradeand also choose to omit the mention of the course as for the following: All the coursessuccessfully completed under the dropped Minors will be shown in the transcript. Courses which were not completed under the droppedMinorwillnot beshowninthe transcript.

The credit contribution of these additional subjects to the computation of CGPA, however, would be considered as nil.

For offering Honors / Minor courses in any department as regular course work minimum 10 and 20 candidates are to be enrolled respectively. Else the courses are to be completed by MOOCS courses as suggested in the AICTE / APSCHE guidelines.

Track-1 Digital System Design

List of courses

S.No	Name of Course	Credits
1.	VHDL Programming	4
2.	Digital IC design using HDL	4
3.	FPGA based system design	4
4.	VLSI Testing Principles	4

Track-2 IoT based Embedded system Design

List of courses

S.No	Name of Course	Credits
1.	Digital Electronics	4
2.	Microprocessor and its applications	4
3.	Microcontroller and its interfacing	4
4.	Introduction to IoT and Embedded systems	4

Track-1 Digital System Design

List of courses

S.No	Name of Course	Credits
1.	VHDL Programming	4
2.	Digital IC design using HDL	4
3.	FPGA based system design	4
4.	VLSI Testing Principles	4

VHDL PROGRAMMING

ECE XXX **Credits:4**

Instruction : 3 periods & 1 Tutorial/Week Sessional Marks:40

End Exam : 3 Hours End Exam Marks:60

Prerequisites: Digital Electronics and Logic Design

Course Outcomes:

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

1.	Apply the knowledge of HDL for modeling and functional verification of Digital circuits.
2.	Analyze digital circuits using gate level and data flow Verilog HDL modeling
3.	Analyze digital circuits using behavioral Verilog HDL modeling
4.	Design and synthesize a digital circuit for complex systems using Verilog HDL
5.	Program and synthesize a given problem statement using state machines and Verilog HDL

SYLLABUS

UNIT I **[9 Periods]**

INTRODUCTION TO VHDL: VHDL view of a device, basic terminology, Entity declaration, Architecture Body, Configuration declaration, package declaration, package body, model analysis. Identifiers, data objects, data types, Operators.

BEHAVIOURAL MODELLING: Entity declaration, Architecture body, process statement, variable assignment statement, signal assignment statement, wait statement, if statement, case statement, Null statement, Loop statement, exit statement, next statement, assertion statement.

UNIT II **[9 Periods]**

DATA FLOW MODELING: Concurrent signal assignment statement**,** concurrent versus sequential signal assignment, Delta delay model, multiple drivers, Conditional signal assignment, selected signal assignment, block statement, concurrent assertion statement.

STRUCTURAL MODELLING: component declaration, component instantiation.

UNIT III **[9 Periods]**

GENERICS AND CONFIGURATIONS: Generics, configuration specification, configuration declaration, Default rules.

SUBPROGRAMS AND OVERLOADING: Subprograms: Functions, Procedures, Declarations. Subprogram overloading, Operator overloading.

UNIT IV **[9 Periods]**

PACKAGES AND LIBRARIES: Package declaration, package body, Design libraries.

SYNCHRONOUS SEQUENTIAL CIRCUITS: Moore and Mealy FSM, Design and implementation of sequence detector, serial adder, code converter.

UNIT V **[9 Periods]**

MODEL SIMULATION: simulation, writing a test bench: waveform generation, using vectors, monitoring behavior. Hardware modeling examples: Modeling synchronous logic, State machine modeling.

TEXT BOOKS:

1. Jayaram Bhaskar, A VHDL Primer, 3rd edition, Prentice Hall PTR, 1999

REFERENCE BOOKS:

1. Douglas L. Perry, VHDL: Programming by example, 4th edition, McGraw Hill Education, 2017

DIGITAL IC DESIGN USING HDL

ECE XXX **Credits:4**

Instruction : 3 periods & 1 Tutorial/Week Sessional Marks:40

End Exam : 3 Hours End Exam Marks:60

Prerequisites: Digital Electronics and Logic Design

Course Outcomes:

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

1.	Interpret the importance of EDA tools and its flow for VLSI designs
2.	Model logic gates ,half adder, full adder ,various digital blocks by using modern tools with HDL
3.	Construct verilog HDL models for combinational and sequential circuits using gate level, behavioral level and dataflow level
4.	Build CMOS circuits using Verilog switch level programming
5.	Apply design rule checks and timing parameters to digital circuits and model the state machines

SYLLABUS

UNIT I 10 Periods

Introduction to Electronic Design Automation: Introduction, FPGA Design flow, ASIC Design flow, architectural design, logic design, Physical design of IC. Simulation, verification and testing. EDA Tools: FPGA Design, ASIC Design.

FPGA Based Front End Design-Implementation, FPGA configuration, User constraints Xilinx 3000 Series FPGA architecture, ALTERA FLEX 10K Series CPLD architecture

UNIT II 10 Periods

Verilog Language Constructs: Verilog as HDL, Levels of Design Description, Concurrency, Simulation and Synthesis, Functional Verification, System Tasks, Programming Language Interface (PLI), Module, Simulation and Synthesis Tools, Test Benches. Keywords, Identifiers, White Space Characters, Comments, Numbers, Strings, Logic Values, Strengths, Data Types, Scalars and Vectors, Parameters, Memory, Operators, System Tasks, Exercises

UNIT III 10 Periods

Gate level Modeling and Dataflow Modeling: AND Gate Primitive, Module Structure, Other Gate Primitives, Tri-State Gates, Array of Instances of Primitives, Additional Examples, Design of Flip-flops with Gate Primitives, Delays, Strengths and Contention Resolution, Net Types, Design of Basic Circuits, Exercises. Continuous Assignment Structures, Delays and Continuous Assignments, Assignment to Vectors, Operators.

UNIT IV 10 Periods

Behavioral and Switch Level Modeling: Introduction, Operations and Assignments, Functional Bifurcation, 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, Simulation Flow. i$f$and if-else constructs, repeat construct, for loop, , while loop, forever loop, parallel blocks, force-release construct, Event. Basic Transistor Switches, CMOS Switch, Bi-directional Gates, Time Delays with Switch Primitives, Instantiations with Strengths and Delays, Strength Contention with Trireg Nets, Exercises

UNIT V 10 Periods

System Tasks, Functions, UDP and SM Charts: Introduction, Parameters, Path Delays, Module Parameters, System Tasks and Functions. File Based Tasks and Functions, Compiler Directives, Hierarchical Access, General observations, Exercises.

User-Defined Functions, Tasks and Primitives-Introduction, Function, Tasks, User- Defined Primitives (UDP), FSM Design (Moore and Mealy Machines), State Machine Charts, Derivation of SM Charts, Realization of SM Charts, Examples based on SM charts

TEXT BOOKS:

1. T.R. Padmanabhan and B. Bala Tripura Sundari," Design through Verilog HDL" WSE, IEEE Press, 2004**(UNIT-I,II,III,IV &V)**
2. J. Bhaskar" A Verilog Primier" ,First edition ,BSP, 2003**(UNIT-I,II,III,IV &V)**

REFERENCE BOOKS:

1. Brown and ZvonkoVranesic Stephen" Fundamentals of Logic Design with Verilog "TMH, 2005.

2. Michael D. Ciletti "Advanced Digital Design with Verilog HDL ",Second edition, PHI, 2005.

FPGA BASED SYSTEM DESIGN	
ECE XXX	**Credits:4**
Instruction: 4 Periods week	Sessional Marks:40
End Exam: 3 Hours	End Exam Marks:60

Prerequisites: Digital Logic Design, HDL

COURSE OUTCOMES

After undergoing the course, students will be able to	
1.	Explore Basics of computer FPGA architecture and the digital design fundamentals.
2.	Explain the FPGA internals
3.	Implement the combinational logic circuits on FPGA.
4.	Implement the sequential logic circuits on FPGA.
5.	Explain different design methodologies and implement different high end applications on FPGA

SYLLABUS

UNIT I **[10 Periods]**

FPGA-BASED SYSTEMS: Introduction, Basic Concepts: Boolean Algebra, Schematics and Logic Symbols, Digital Design and FPGAs: The Role of FPGAs, FPGA Types, FPGAs vs. Custom VLSI.
FPGA-BASED SYSTEM DESIGN: Goals and Techniques, Hierarchical Design, Design Abstraction , Methodologies

UNIT II **[10 Periods]**

FPGA FABRICS: Introduction, FPGA Architectures, SRAM-Based FPGAs
OVERVIEW: Logic Elements, Interconnection Networks, Configuration
PERMANENTLY PROGRAMMED FPGAS: Anti-fuses Flash Configuration, Logic Blocks, Interconnection Networks, Programming Chip I/O
CIRCUIT DESIGN OF FPGA FABRICS: Logic Elements, Interconnect
Architecture of FPGA Fabrics: Logic Element Parameters, Interconnect Architecture, Pinout

UNIT III **[10 Periods]**

COMBINATIONAL IMPLEMENTATION: Logic Implementation for FPGAs: Syntax-Directed Translation, Logic Implementation by Macro , Logic Synthesis, Technology-Independent Logic Optimization, Technology-Dependent Logic Optimizations Logic Synthesis for FPGAs
Physical Design for FPGAs: Placement, Routing

UNIT IV **[10 Periods]**

SEQUENTIAL IMPLEMENTATION: The Sequential Machine Design Process

SEQUENTIAL DESIGN STYLES: State Transition and Register-Transfer Models, Finite-State Machine Theory, State Assignment, Verilog Modeling Styles

RULES FOR CLOCKING: Flip-Flops and Latches, Clocking Disciplines

PERFORMANCE ANALYSIS: Performance of Flip-Flop-Based Systems, Performance of Latch-Based Systems, Clock Skew, Retiming ,Power Optimization

UNIT V **[10 periods]**

ARCHITECTURE : Introduction

BEHAVIORAL DESIGN : Data Path-Controller Architectures, Scheduling and Allocation, Power, Pipelining

DESIGN METHODOLOGIES : Design Processes, Design Standards, Design Verification

DESIGN EXAMPLE: Digital Signal Processor

TEXT BOOKS:

1. Wayne wolf, FPGA-based System Design, Pearson Publication, 1 January 2005, ISBN-10 8129710900
2. Cem Unsalan, Bora Tar, Digital System Design with FPGA: Implementation Using Verilog and VHDL, Publisher: McGraw-Hill Education, Year: 2017, ISBN: 1259837904,9781259837906

REFERENCE BOOKS:

1. Kilts, Advanced FPGA Design Architecture, Implementation, and optimization, Wiley (1 January 2016) ISBN-10 : 9788126561728, ISBN-13 : 978-8126561728

2. R. C. Cofer Benjamin F. Harding, Rapid System Prototyping with FPGAs: Accelerating the Design Process,: 0750678666,Publisher : Newnes (27 October 2005), ISBN-10 : 9780750678667, ISBN-13 : 978-0750678667

VLSI TESTING PRINCIPLES	
ECE XXX	**CREDITS: 4**
Instruction: 3 Periods & 1 Tutorial/Week	Sessional Marks: 40
End Exam : 3 Hours	End Exam Marks: 60

Prerequisites: Digital Logic design, VLSI

Course Outcomes:

At the end of the course, the student will be able to:	
1.	Design a fault tolerant digital system
2.	Analyze self-checking circuits.
3.	Implement and test combinational circuits.
4.	Develop and test a sequential circuits.
5.	Design an automatic testing system using BIST for basic sequential circuits

SYLLABUS

UNIT I **[9Periods]**

BASIC CONCEPTS: Reliability concepts, Failure & Faults, Reliability and failure rate, Relation between reliability and Meantime between failure, Maintainability and Availability, Reliability of series, Parallel and Parallel-Series combinational circuits.'

FAULT TOLERANT DESIGN: Basic concepts – Static, dynamic, hybrid, Triple Modular Redundant System, Self purging redundancy, Sift out redundancy (SMR), SMR Configuration, Use of error correcting code, Time redundancy and software redundancy.

UNIT II **[9Periods]**

SELF CHECKING CIRCUITS: Basic concepts of Self checking circuits, Design of Totally Self Checking checker, Checkers using m out of n codes, Berger code, Low cost residue code
.

FAIL SAFE DESIGN: Strongly fault secure circuits, fail-safe design of sequential circuits using partition theory and Berger code, totally self-checking PLA design.

UNIT III **[9Periods]**

DESIGN FOR TESTABILITY FOR COMBINATIONAL CIRCUITS: Basic concepts of testability, controllability and observability, the Reed Muller's expansion technique, OR-AND-OR design, use of control and syndrome testable design. Theory and operation of LFSR, LFSR as Signature analyzer, Multiple-input Signature Register.

UNIT IV **[9Periods]**

DESIGN FOR TESTABILITY FOR SEQUENTIAL CIRCUITS: Controllability and observability by means of scan register, Storage cells for scan design, classic scan design, Level Sensitive Scan Design (LSSD).

UNIT V **[9Periods]**

BUILT IN SELF TEST: BIST concepts, Test pattern generation for BIST exhaustive testing, Pseudorandom testing, pseudo exhaustive testing, constant weight patterns, Generic offline BIST architecture.

TEXT BOOKS:

1. Digital Systems Testing and Testable Design-M. Abramovili, M.A. Breues, A. D. Friedman – Jaico publications.2001
2. Fault Tolerant & Fault Testable Hardware Design-Parag K. Lala (PHI) 2020

REFERENCE BOOKS:

1. Fault tolerant systems- Israel Koren, C.Manikrishna, Morgan Kaufmann, 2007.

Track-2 IoT based Embedded system Design

List of courses

S.No	Name of Course	Credits
1.	Digital Electronics	4
2.	Microprocessor and its applications	4
3.	Microcontroller and its interfacing	4
4.	Introduction to IoT and Embedded systems	4

DIGITAL ELECTRONICS	
ECE XXX	**Credits:4**
Instruction: 3 periods & 1 e/week	Sessional marks:40
End exam: 3 hours	End exam marks:60

Pre -requisites: Nil

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Perform conversions between different number systems and codes and apply the Boolean algebra to minimize the given logic expressions.
2.	Minimize the given Boolean expressions using K-Map (up to four variables) and QM method (up to 5 variables).
3.	Design and Analyze combinational logic circuits.
4.	Design and Analyze sequential logic circuits.
5.	Analyze the characteristics of logic families and compare their performance in terms of performance metrics.

SYLLABUS

UNIT – I **[09 Periods]**

NUMBER SYSTEMS

Number representation, Conversion of bases, Binary Arithmetic, Representation of Negative numbers, Binary codes: weighted and non-weighted BOOLEAN ALGEBRA: Basic definitions, Axiomatic Definitions, Theorems and properties, Boolean Functions, Canonical and standard forms.

(TB1- chapters 1 & 2)

UNIT – II **[09 Periods]**

LOGIC MINIMIZATION

The K-Map Method: Two variable map, Three variable map, four variable map Prime Implicants, Don't care conditions, NAND and NOR implementation, Quine-Mccluskey (QM) (upto five variables) Technique. (**TB1- chapters 3)**

UNIT – III **[09 Periods]**

COMBINATIONAL LOGIC DESIGN

Combinational circuits, Analysis Procedure, Design Procedure, Code Converters (BCD to XS3(XS3 to BCD)), Gray to Binary (Binary to Gray), Binary Adder-Subtractor, Decimal adder, Binary Multiplier, Magnitude comparator, Decoders, Encoders, Multiplexers. De-Multiplexer, Hazards.

(TB1- chapters 4 & 9.7)

UNIT – IV **[09 Periods]**

SEQUENTIAL LOGIC DESIGN

Introduction to Latch and Flip flop, clocked S-R, JK, D, T flip flops. Excitation table of Flipflop, Flip flop conversion, Clocked flip flop design, Edge triggered flip flop, applications of flipflops. Registers, Applications of Shift registers, universal shift register, Ripple counters, Synchronous counters, counter with unused states, Ring counters, Johnson counter. **(TB2- chapters 7 & 8 (till 8.5))**

UNIT – V **[09 Periods]**

LOGIC FAMILIES

Introduction, Characteristics of Digital ICs, Resistor Transistor Logic (RTL), Diode Transistor Logic (DTL), Transistor Transistor Logic (TTL), Emitter Coupled Logic (ECL), CMOS Logic, Interfacing CMOS and TTL. (**TB2- chapter 4)**

TEXT BOOKS:

1. M. Morris Mano and Michael D. Ciletti, "Digital Design", 4th Edition, Pearson Publishers, 2001.
2. R.P Jain, "Modern Digital Electronics", 3rd Edition, TMH, 2003.

REFERENCE BOOKS:

1. William I. Fletcher, "An Engineering Approach to Digital Design", PHI, 1980.
2. John F. Wakerly, "Digital Design Principles and Practices", 3rd Edition, Prentice Hall, 1999.

MICROPROCESSORS AND ITS APPLICATIONS	
ECE XXX	**Credits:4**
Instruction: 3 Periods & 1 Tut/week	Sessional Marks:40
End Exam: 3 Hours	End Exam Marks:60

Prerequisites: Digital Electronics.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	**Gain comprehensive** knowledge of the architecture of 8-bit 8085 Microprocessor and its interrupt structure
2.	**Familiarize** the instruction set of 8085 & apply them to write assembly language programs.
3.	Able to organize the hardware involved in BIU & EU of 8086 microprocessor & analyze the minimum and maximum mode 8086 systems using timing diagrams
4.	**Familiarize** the instruction set of 8086 & apply them to write assembly language programs.
5.	**Develop** applications that will provide solution to real world problems by Interfacing 8086 Microprocessor with various peripherals.

SYLLABUS

UNIT I **[9 Periods]**

OVERVIEW OF 8085 ARCHITECTURE

Introduction to Microprocessors and Microcomputers, Internal Architecture and Functional Description of INTEL 8085 Microprocessor, Interrupt Structure of 8085

UNIT II **[9 Periods]**

INSTRUCTION SET AND ASSEMBLY LANGUAGE PROGRAMMING FOR 8085 Addressing modes, Data Transfer instructions, Arithmetic instructions, Logical instructions, Branching instructions, Machine Control and I/O instructions, Stack and Subroutines, Assembly language Programming.

UNIT III **[9 Periods]**

OVERVIEW OF 8086 ARCHITECTURE

Architecture of 8086, Register organization, Memory segmentation. Physical memory organization. signal description of 8086, Minimum mode 8086 system and timings, Maximum mode 8086 system and timings.

UNIT IV **[9 Periods]**

INSTRUCTION SET AND ASSEMBLY LANGUAGE PROGRAMMING OF 8086: Addressing modes, instruction set, assembler directives (Significant), macros and operators. Simple programs involving arithmetic, logical, branch and string manipulation instructions.

UNIT V **[9 Periods]**

INTERFACING

Memory interfacing to 8086 (Static RAM & EPROM). Methods of parallel data transfer, 8255A Internal block diagram and system connections, 8255A operational modes and initialization, constructing and sending 8255A control words, interfacing to 8086. Interfacing Stepper motor, D/A and A/D converters

TEXT BOOKS:

1. Ramesh S. Gaonkar, Architecture Programming and Applications, 3rd Edition, Penram International Pvt. Ltd.

2. D. V. Hall, Microprocessors and Interfacing, Revised 2nd edition 2006, TMH,.
3. A.K. Ray and K.M. Bhurchand, Advanced Microprocessors and Peripherals, 2nd edition, 2006, TMH.

REFERENCE BOOKS:

1. John Uffenbeck, The 8086/8088 Family: Design, Programming And Interfacing, PHI
2. N. Senthil Kumar, M. Saravanan, and S. Jeevananthan, Microprocessors and Microcontrollers, OUP India

MICROCONTROLLERS & ITS INTERFACING	
ECE XXX	**Credits:4**
Instruction: 3 Periods & 1 Tut/week	Sessional Marks:40
End Exam: 3 Hours	End Exam Marks:60

Prerequisites:

Digital Electronics, Computer Architecture & Organization, Microprocessors and Interfacing

Course Outcomes:

By the end of the course, the student will be able to:	
1.	**Acquire** knowledge of the architecture and operation of Intel 8051 microcontroller.
2.	**Develop** assembly language programs for data transfer, arithmetic, logical, and branching operations using instruction set of 8051 and **apply** them in control applications
3.	**Analyze** the hardware features like timers, memory, interrupts and serial communication available in 8051 Microcontroller Family of devices.
4.	**Develop** applications that will provide solution to real world problems by Interfacing 8051 Microcontroller with peripherals such as ADC, DAC, keyboard, and display.
5.	**Develop** applications that will provide solution to real world problems by Interfacing 8051 Microcontroller with peripherals such as Relay, PWM, DC, and Stepper motor.

SYLLABUS

UNIT I **[9 Periods]**

8051 MICROCONTROLLER:

Introduction to Microcontrollers, comparing Microprocessors and Microcontrollers, Architecture of 8051 Microcontroller, Register organization of 8051, SFRs, Pin configuration of 8051.

UNIT II **[9 Periods]**

ASSEMBLY LANGUAGE PROGRAMMING OF 8051

Addressing modes of 8051. Data Transfer and Logical Instructions. Arithmetic Operations, Decimal Arithmetic. Jump and Call Instructions. Assembly language programming of 8051 microcontroller.

UNIT III **[9 Periods]**

I/O PORT AND INTERRUPT'S PROGRAMMING

Input/Output Ports and Circuits, External Memory, Counters/Timers and modes of Timers, Serial data Input/Output, Interrupts.

UNIT IV **[9 Periods]**

INTERFACING-I

LCD and Keyboard interfacing; ADC, DAC, and Sensor interfacing; Interfacing to external memory.

UNIT V **[9 Periods]**

INTERFACING-II

8051 interfacing with the 8255; Motor control: Relays and optoisolators, stepper motor interfacing, DC motor interfacing and PWM.

TEXT BOOKS:

1. Muhammed Ali Mazidi, Janice Gillispie Mazidi, Rolin D Mc Kinlay , The 8051 Microcontroller and Embedded Systems Using Assembly and C, 2nd Edition, Pearson Education, 2008.
2. Kenneth. J. Ayala, Dhananjay V. Gadre, The 8051 Microcontroller & Embedded Systems Using Assembly and C, 1st edition, Cengage learning, 2010

REFERENCE BOOKS:

1. Satish Shah, 8051 Microcontrollers: MCS 51 Family and Its Variants, 1/e, Oxford University Press, 2010.

INTRODUCTION TO EMBEDDED SYSTEM AND IoT	
ECE XXX	**Credits:4**
Instruction: 3 Periods & 1 Tut/week	Sessional Marks:40
End Exam: 3 Hours	End Exam Marks:60

Pre -requisites: Microprocessors and Microcontrollers

Course outcomes:

By the end of the course, the student will be able to:	
1.	Acquire knowledge of the embedded systems.
2.	Acquire knowledge of IoT.
3.	Design a PoC of an IoT system using Raspberry Pi/Arduino.
4.	Apply data analytics and use cloud offerings related to IoT.
5.	Analyze applications of IoT in real time scenario

SYLLABUS

UNIT I **[9 Periods]**

INTRODUCTION TO EMBEDDED SYSTEMS

Embedded systems; processor embedded into system; Embedded hardware units and devices in a system; embedded software in a system; examples of embedded systems; embedded system on chip (SOC) and use of VLSI circuit design technology; complex system design and processor; Design process in embedded system; classification of embedded systems.

UNIT II **[9 Periods]**

FUNDAMENTALS OF IoT

Evolution of Internet of Things – Enabling Technologies – IoT Architectures: oneM2M, IoT World Forum (IoTWF) and Alternative IoT models – Simplified IoT Architecture and Core IoT Functional Stack -– Fog, Edge and Cloud in IoT – Functional blocks of an IoT ecosystem – Sensors, Actuators, Smart Objects and Connecting Smart Objects

UNIT III **[9 Periods]**

DESIGN AND DEVELOPMENT

Design Methodology – Embedded computing logic – Microcontroller, System on Chips – IoT system building blocks – Arduino – Board details, IDE programming – Raspberry Pi – Interfaces and Raspberry Pi with Python Programming.

UNIT IV **[9 Periods]**

DATA ANALYTICS AND SUPPORTING SERVICES

Structured Vs Unstructured Data and Data in Motion Vs Data in Rest – Role of Machine Learning – No SQL Databases – Hadoop Ecosystem – Apache Kafka, Apache Spark – Edge Streaming Analytics and Network Analytics – Xively Cloud for IoT, Python Web Application Framework – Django – AWS for IoT – System Management with NETCONF-YANG.

UNIT V [9 Periods]

CASE STUDIES/INDUSTRIAL APPLICATIONS

Cisco IoT system – IBM Watson IoT platform – Manufacturing – Converged Plantwide Ethernet Model (CPwE) – Power Utility Industry – GridBlocks Reference Model – Smart and Connected Cities: Layered architecture, Smart Lighting, Smart Parking Architecture and Smart Traffic Control.

TEXT BOOKS:

1. Raj Kamal, Embedded systems: architecture, programming and design, Tata McGraw-Hill Education.
2. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, ―IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, Cisco Press, 2017

REFERENCE BOOKS:

1. Arshdeep Bahga, Vijay Madisetti, ―Internet of Things – A hands-on approach, Universities Press, 2015
2. Olivier Hersent, David Boswarthick, Omar Elloumi , ―The Internet of Things – Key applications and Protocols, Wiley, 2012 (for Unit 2).
3. Jan Ho¨ ller, Vlasios Tsiatsis , Catherine Mulligan, Stamatis , Karnouskos, Stefan Avesand. David Boyle, "From Machine-to-Machine to the Internet of Things – Introduction to a New Age of Intelligence", Elsevier, 2014.
4. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), ―Architecting the Internet of Things, Springer, 2011.
5. Michael Margolis, Arduino Cookbook, Recipes to Begin, Expand, and Enhance Your Projects, 2nd Edition, O'Reilly Media, 2011.