SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE
(AUTONOMOUS)

(Approved by AICTE, New Delhi, Affiliated to JNTUK, Kakinada)

Accredited by NAAC with 'A+' Grade

Recognised as Scientific and Industrial Research Organisation

SRKECR MARG, CHINA AMIRAM, BHIMAVARAM – 534204 W.G.Dt., A.P., INDIA

Regulation: R23	

ELECTRONICS AND COMMUNICATION ENGINEERING (Honors)

COURSE STRUCTURE
(With effect from 2023-24 admitted Batch onwards)

Course Code	Course Name	Year/ Sem	Cr	L	T	P	C.I.E	S.E.E	Total Marks
B23ECH101	Advanced Digital Design	III-I	3	3	0	0	30	70	100
B23ECH201	Design Verification using Verilog HDL	III-II	3	3	0	0	30	70	100
B23ECH301	Design Verification using Verilog HDL Laboratory	III-II	1.5	0	0	3	30	70	100
B23ECH401	Design Verification using System Verilog	IV-I	3	3	0	0	30	70	100
B23ECH501	Design Verification using System Verilog Laboratory	IV-I	1.5	0	0	3	30	70	100
B23ECH601	*MOOCS-I	III-I to IV-I	3	--	--	--	--	--	100
B23ECH701	*MOOCS-II	III-I to IV-I	3	--	--	--	--	--	100
		TOTAL	**18**	**9**	**0**	**6**	**150**	**350**	**700**

*Three MOOCS courses of any **ELECTRONICS AND COMMUNICATION ENGINEERING** related Program Core Courses from NPTEL/SWAYAM with a minimum duration of 12 weeks (3 Credits) courses other than the courses offered need to be taken by prior information to the concern. These courses should be completed between III Year I Semester to IV Year I Semester

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
B23ECH101	Honors	3	-	--	3	30	70	3Hrs

ADVANCED DIGITAL DESIGN

(Honors Degree Course in ECE)

Course Objectives:

1.	To design sequential circuits, FSMs, memory devices, and programmable logic devices including FIFO and glitch analysis
2.	To model programs in Verilog HDL for digital system design using behavioral, dataflow, and structural modelling

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

S.No	Outcome	Knowledge Level
1.	**Design** sequential circuits using latches, flip-flops, registers and counters.	K3
2.	**Analyze** finite state machines (FSMs) for various digital applications.	K4
3.	**Evaluate** memory architectures and FIFO mechanisms	K3
4.	**Apply** the concepts of Programmable Logic Devices (PLDs) and FPGA for digital design.	K3
5.	**Develop** digital circuits using Verilog HDL with different modeling techniques.	K3

SYLLABUS

UNIT-I (10Hrs)	Sequential Circuits Latches & Flip Flops, Excitation tables, FF Conversions, Delays in sequential circuits, Setup & hold times, Registers- bi-directional shift register, universal shift register, Counters- Johnson counter, ring counter, Sequence generators.
UNIT-II (10 Hrs)	Finite State Machine Mealy & Moore Machine, Sequence detectors, State Machine Examples(3-bit)
UNIT-III (8 Hrs)	Memories Classification of Memories, FIFO, Glitches
UNIT-IV (10 Hrs)	Programmable Logic Device Introduction to PROM, PAL & PLA, FPGA-Architecture, Organization, Programming technologies
UNIT-V (10 Hrs)	Introduction to Verilog HDL: Verilog as HDL Language Constructs and Conventions: Introduction, White Space, Comments, Operators, Strings, Keywords and Identifiers, Data Types-Value set, Nets, Registers, Vectors

	Introduction- Behavioral Modeling, Dataflow Modeling, Structural Modeling
Textbooks:	
1.	Fundamentals of Logic Design by Charles H. Roth Jr, JaicoPublishers,2006
2.	Verilog HDL - Samir Palnitkar, 2nd Edition, Pearson Education, 2009.
Reference Books:	
1.	Switching and finite automata theory Zvi.Kohavi, Niraj.K.Jha 3rd Edition,Cambridge UniversityPress,2009.
2.	Stephen M. Trimberger, "Field Programmable Gate Array Technology", Springer International Edition.
3.	T.R. Padmanabhan, B Bala Tripura Sundari, Design Through Verilog HDL, Wiley 2009.
e-Resources	
1.	https://www.chipverify.com/tutorials/verilog

				Course Code: B23ECH101	
		SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE (A)			**R23**
		III B.Tech. I Semester MODEL QUESTION PAPER			
		ADVANCED DIGITAL DESIGN			
		(Honors Degree Course in ECE)			
Time: 3 Hrs.				**Max. Marks: 70 M**	
		Answer **ONE Question** from **EACH UNIT**			
		All questions carry equal marks			
		Assume suitable data if necessary			
				10 x 2 = 20 Marks	
			CO	**KL**	**M**
1.	**a).**	What is the difference between a latch and a flip-flop?	**1**	**1**	**2**
	b).	Explain the significance of setup and hold times in sequential circuits.	**1**	**2**	**2**
	c).	Differentiate between Mealy and Moore state machines	**2**	**2**	**2**
	d).	What are the advantages of using state machines in digital design?	**2**	**3**	**2**
	e).	What is FIFO memory, and where is it used?	**3**	**3**	**2**
	f).	Define glitches in memory	**3**	**1**	**2**
	g).	Differentiate between PAL and PLA	**4**	**1**	**2**
	h).	What are the key architectural components of an FPGA?	**4**	**2**	**2**
	i).	What are the different modeling styles in Verilog HDL?	**5**	**1**	**2**
	j).	Explain the significance of registers and nets in Verilog HDL.	**5**	**2**	**2**
				5 x 10 = 50 Marks	
		UNIT-1			
2.	**a).**	Describe the process of converting one flip-flop type to another using excitation tables.	**1**	**2**	**5**
	b).	Discuss the design and working of a Johnson counter with timing diagrams	**1**	**4**	**5**
		OR			
3.		Design a 3-bit synchronous counter using JK flip-flops and explain the working process.	**1**	**3**	**10**
		UNIT-2			
4.	**a).**	Explain the working of a 3-bit sequence detector using a state machine.	**2**	**3**	**5**
	b).	What is a state diagram, and how is it used in FSM design?	**2**	**1**	**5**
		OR			
5.		Construct a sequence detector that identifies the pattern ‘101’ using a Mealy machine.	**2**	**3**	**10**
		UNIT-3			

6.	**a).**	Explain the differences between static and dynamic memory technologies.	**3**	**2**	**5**
	b).	List different types of memory devices with examples.	**3**	**1**	**5**
		OR			
7.		Describe the working principle of FIFO memory and its applications.	**3**	**2**	**10**
		UNIT-4			
8.		Explain how antifuse and flash programming technologies work in FPGAs.	**4**	**2**	**10**
		OR			
9.	**a).**	Explain in detail about FPGA architecture	**4**	**2**	**5**
	b).	Design a simple combinational circuit using a PLA	**4**	**3**	**5**
		UNIT-5			
10.	**a).**	Design a 2-to-4 decoder using behavioral modeling in Verilog.	**5**	**3**	**10**
		OR			
11.	**a).**	Explain the differences between behavioral and structural modeling.	**5**	**2**	**5**
	b).	What is a testbench in Verilog, and what is its purpose?	**5**	**1**	**5**

CO-COURSE OUTCOME **KL-KNOWLEDGE LEVEL** **M-MARKS**

NOTE : Questions can be given as A,B splits or as a single Question for 10 marks

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
B23ECH201	Honors	3	-	--	3	30	70	3Hrs

DESIGN VERIFICATION USING VERILOG HDL

(Honors Degree Course in ECE)

Course Objectives:

1.	To understand the ASIC design flow and the fundamental Verilog constructs used for hardware modeling and verification.
2.	To implement data types, operators, procedural blocks, and assignments in Verilog for digital circuit description and simulation.
3.	To analyze timing controls, race conditions, and execution order dependencies in sequential and parallel Verilog blocks.
4.	To Develop test benches, tasks, and functions for efficient and reusable verification of complex hardware systems.

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

S.No	Outcome	Knowledge Level
1.	**Apply** Verilog module declaration and instantiation to describe hardware components.	K3
2.	**Implement** data types, operators, and primitives for functional representation in Verilog.	K3
3.	**Develop** Testbenches using arrays, memories, system tasks, and procedural assignments.	K3
4.	**Identify** and resolve race conditions, timing issues, and execution order dependencies in Verilog simulations.	K4
5.	**Design** tasks and functions for modular and efficient verification of complex hardware circuits.	K4

SYLLABUS

UNIT-I (10Hrs)	ASIC Flow, Module declaration and Instantiation, Components of simulation, Procedural blocks.
UNIT-II (10 Hrs)	Lexical convections, Data types, Module Parameters, Operators, Primitives, Functional representation in Verilog
UNIT-III (10 Hrs)	Arrays, Memories, System tasks, compiler Directives, Continuous and Procedural Assignments.

UNIT-IV (10 Hrs)	Race Condition , Examples of Blocking & non blocking statements, Conditional Statements, loops Statements, Timing Controls Sequential and Parallel Blocks
UNIT-V (10 Hrs)	Tasks and Functions, Difference between task and Function, declaration, invocation. Introduction to Logic Synthesis-A Basic Introduction to Translation and Mapping Theoretical Concepts. (Reference Textbooks-Chapter 9,pp 275,Verilog HDL: A Guide to Digital Design and Synthesis by Samir Palnitkar-2nd Edition,2003, J.Bhasker-Latest edition, year of Publication)
Textbooks:	
1.	Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis," 2nd Edition, Pearson Education.
2.	Stephen Brown &ZvonkoVranesic, "Fundamentals of Digital Logic with Verilog Design," McGraw- Hill.
Reference Books:	
1.	Bhasker J., "A Verilog HDL Primer," 3rd Edition, Springer.
2.	Perry Douglas, "Verilog HDL," 5th Edition, McGraw-Hill.
3.	Michael Ciletti, "Advanced Digital Design with the Verilog HDL," Prentice Hall.
e-Resources	
1.	https://archive.nptel.ac.in/noc/courses/noc16/SEM2/noc16-ec08/
2.	https://onlinecourses.nptel.ac.in/noc20_ee76/preview

					Course Code: B23ECH201
		SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE (A)			**R23**
		III B.Tech. II Semester MODEL QUESTION PAPER			
		DESIGN VERIFICATION USING VERILOG HDL			
		(Honors Degree Course in ECE)			
Time: 3 Hrs.					**Max. Marks: 70 M**
		Answer **ONE Question** from **EACH UNIT**			
		All questions carry equal marks			
		Assume suitable data if necessary			
					10 x 2 = 20 Marks
			CO	**KL**	**M**
1.	**a).**	What are the key steps in the ASIC design flow?	**1**	**2**	**2**
	b).	Differentiate between module declaration and module instantiation in Verilog.	**1**	**3**	**2**
	c).	What are module parameters in Verilog, and why are they used?	**2**	**2**	**2**
	d).	Differentiate between continuous assignment and procedural assignment in Verilog.	**2**	**3**	**2**
	e).	How are arrays used for memory modeling in Verilog?	**3**	**3**	**2**
	f).	What is the purpose of $monitor and $display system tasks in Verilog?	**3**	**1**	**2**
	g).	What is a race condition in Verilog, and how can it be avoided?	**4**	**2**	**2**
	h).	Give one example each of blocking and non-blocking assignments in Verilog.	**4**	**3**	**2**
	i).	What is the difference between a task and a function in Verilog?	**5**	**2**	**2**
	j).	How is a function invoked in Verilog? Provide a basic syntax.	**5**	**2**	**2**
					5 x 10 = 50 Marks
		UNIT-1			
2.	**a).**	Explain the steps involved in the ASIC design flow with a brief description of each step.	**1**	**2**	**5**
	b).	Write a Verilog module for a 2-input AND gate and show how to instantiate it in another module.	**1**	**3**	**5**
		OR			
3.	**a).**	What are procedural blocks in Verilog? Explain the differences between initial and always blocks with examples.	**1**	**4**	**5**
	b).	Describe the role of lexical conventions in Verilog. What are the different types of tokens used in Verilog coding?	**1**	**2**	**5**
		UNIT-2			
4.	**a).**	Explain the different data types in Verilog with examples.	**2**	**2**	**5**
	b).	Write a Verilog module using parameters to define a 4-bit adder. Explain	**2**	**3**	**5**

		how parameters improve design flexibility.			
		OR			
5.	**a).**	Describe the different types of operators in Verilog with examples.	**2**	**2**	**5**
	b).	What are primitives in Verilog? Explain their role in functional representation with an example of a gate-level model.	**2**	**4**	**5**
		UNIT-3			
6.	**a).**	Describe the role of compiler directives in Verilog with examples of any three commonly used directives.	**3**	**3**	**5**
	b).	Illustrate the use of continuous and procedural assignments in Verilog with suitable code examples.	**3**	**4**	**5**
		OR			
7.	**a).**	Write a Verilog code to demonstrate the use of $time, $strobe, and $finish system tasks, and explain their outputs.	**3**	**4**	**5**
	b).	Explain the use of arrays in Verilog for memory modeling with a suitable example.	**3**	**4**	**5**
		UNIT-4			
8.	**a).**	Differentiate between blocking and non-blocking statements with suitable Verilog code examples.	**4**	**3**	**5**
	b).	Explain the different types of timing controls in Verilog with examples.	**4**	**3**	**5**
		OR			
9.	**a).**	Describe different types of conditional statements in Verilog and explain their usage with examples.	**4**	**3**	**5**
	b).	Compare sequential and parallel blocks in Verilog. Give examples to illustrate their differences.	**4**	**4**	**5**
		UNIT-5			
10.	**a).**	Write a Verilog program that uses both a task and a function to perform arithmetic operations (such as addition, subtraction, multiplication, and division). Explain the program step by step, highlighting the declaration, invocation, and execution flow.	**5**	**4**	**10**
		OR			
11.	**a).**	Explain the differences between tasks and functions in Verilog with examples.	**5**	**3**	**5**
	b).	Describe how arguments are passed to tasks and functions in Verilog. Give examples for both.	**5**	**4**	**5**

CO-COURSE OUTCOME **KL-KNOWLEDGE LEVEL** **M-MARKS**

NOTE : Questions can be given as A,B splits or as a single Question for 10 marks

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
B23ECH301	Honors	--	--	3	1.5	30	70	3 Hrs.

DESIGN VERIFICATION USING VERILOG HDL LABORATORY

(Honors Degree Course in ECE)

Course Objectives:

1	To familiarize students in HDL programming for the concepts of Digital Signal Processing (DSP), Computer Architecture and Organization.
2	To enhance students' comprehension of simulation and synthesis methodologies for Error Detection and Correction, Image Processing.

Course Outcomes: Students will be able to

S.No	Outcome	Knowledge Level
1	**Develop** HDL (Hardware Description Language) programs for various digital signal processing applications and computer architectures.	K3
2	**Develop** HDL (Hardware Description Language) programs for communications.	K3
3	**Design** HDL (Hardware Description Language) programs for image processing applications	K4

SYLLABUS

Any Four Experiments

1	Design and implement a Finite Impulse Response (FIR) filter.
2	Extend the basic ALU to perform floating-point addition, subtraction, multiplication, and division.
3	Implement a Direct Memory Access (DMA) controller that manages data transfer between memory and peripherals.
4	Implement Hamming Code and Cyclic Redundancy Check (CRC) for error detection and correction.
5	Design a module that converts an RGB image to a grayscale image using the luminance formula. Implement the design using behavioral modeling in Verilog.
6	Assignment-1
7	Assignment-2

EDA Tools/Hardware Required:

1.	EDA Tool that supports FPGA programming including Xilinx Vivado.
2.	Cadence/Synopsys/Tanner or Equivalent Industry Standard CAD Tool.
3.	Desktop computer with appropriate Operating System that supports the EDA tools.

Reference Books:

1	Digital Systems Design Using Verilog-Charles H. Roth.
2	Digital Design with Verilog - Course (nptel.ac.in)

3	https://www.sciencedirect.com/topics/computer-science/finite-impulse-response-filter
4	https://embeddedwala.com/Blogs/embeddedsystem/what-is-dma

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
B23ECH401	Honors	3	--	--	3	30	70	3 Hrs.

DESIGN VERIFICATION USING SYSTEM VERILOG

(Honors Degree Course in ECE)

Course Objectives:

1.	To explain SystemVerilog verification methodologies, including advanced data types and object-oriented programming.
2.	To develop SystemVerilog interfaces for bus protocols and transaction-level modeling.

Course Outcomes: Students will be able to

S.No	Outcome	Knowledge Level
1.	**Explain** hardware verification methodologies for specific design challenges.	K2
2.	**Classify** hardware components using SystemVerilog data types and complex testbench structures.	K2
3.	**Explain** strings, unions, structures, enums, and events in SystemVerilog for verification.	K2
4.	**Develop** SystemVerilog interfaces with modports and clocking blocks for verification.	K3
5.	**Utilize** reusable verification components using object-oriented programming in SystemVerilog.	K3

SYLLABUS

UNIT-I (10Hrs)	**Introduction to system Verilog:** System Verilog testbench architecture, Verilog vs System Verilog
UNIT-II (10 Hrs)	**SV data types:** 2 state vs 4 state variables, dynamic arrays, associate arrays and its usage
UNIT-III (10 Hrs)	**Advanced SV data types:** Strings, Unions Structures Enumerated Data Types, Events
UNIT-IV (10 Hrs)	**SV interfaces:** SV Interfaces, Mod Ports, Clocking Blocks Virtual Interface, Program Blocks
UNIT-V (10 Hrs)	**SV OOPs:** SV Class, Inheritance, This Operator, Super Operator, Shallow Copy, Deep Copy, Parameterized Classes, Typedef Classes, Polymorphism, Abstract Class, Encapsulation, Dynamic Casting, Scope Resolution Operators

Textbooks:	
1.	"SystemVerilog for Verification: A Guide to Learning the Testbench Language Features" by Chris Spear , third edition, springer
2.	“Writing Testbenches: Functional Verification of HDL Models" by Janick Bergeron, second edition, Springer
Reference Books:	
1	"SystemVerilog for Design" by Stuart Sutherland, Springer
2	IEEE Standard 1800-2017: IEEE Standard for SystemVerilog—Unified Hardware Design, Specification, and Verification Language
3	"UVM Primer: A Step-by-Step Introduction to the Universal Verification Methodology" by Ray Salemi, CreateSpace Independent Publishing Platform
1	**IEEE Xplore:** This digital library provides access to a vast collection of technical papers, articles, and standards, including the IEEE SystemVerilog standard. It's a valuable resource for in-depth research and understanding of advanced topics.
2	**EDA Vendor Websites:** Major Electronic Design Automation (EDA) vendors like Synopsys, Cadence, and Mentor Graphics (Siemens EDA) provide online documentation, application notes, and tutorials on their SystemVerilog-related tools and technologies. These resources can be very helpful for practical implementation and tool-specific knowledge.
3	**Verification Academy:** This website (verificationacademy.com) offers a wealth of resources on hardware verification, including articles, tutorials, and online courses. It covers various verification methodologies and SystemVerilog-related topics.

				Course Code: B23ECH401	
SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE (A)					**R23**
IV B.Tech. I Semester MODEL QUESTION PAPER					
DESIGN VERIFICATION USING SYSTEM VERILOG					
(Honors Degree Course in ECE)					
Time: 3 Hrs.				**Max. Marks: 70 M**	
Answer **ONE Question** from **EACH UNIT**					
All questions carry equal marks					
Assume suitable data if necessary					
10 x 2 = 20 Marks					
			CO	**KL**	**M**
1.	**a).**	What is the primary advantage of using SystemVerilog over Verilog for verification?	**1**	**1**	**2**
	b).	Name one key component of a typical SystemVerilog testbench architecture.	**1**	**1**	**2**
	c).	What is the primary use of dynamic arrays in SystemVerilog?	**2**	**1**	**2**
	d).	What is the key difference between a dynamic array and an associative array?	**2**	**1**	**2**
	e).	How are enumerated data types defined in SystemVerilog?	**3**	**1**	**2**
	f).	What is the primary use of events in SystemVerilog?	**3**	**1**	**2**
	g).	What is the primary purpose of an interface in SystemVerilog?	**4**	**1**	**2**
	h).	What is the role of a modport in a SystemVerilog interface?	**4**	**1**	**2**
	i).	What is the fundamental building block of object-oriented programming in SystemVerilog?	**5**	**1**	**2**
	j).	What is inheritance in OOP?	**5**	**1**	**2**
					5 x 10 = 50 Marks
		UNIT-1			
2.	**a).**	Explain the key differences between Verilog and SystemVerilog, focusing on features relevant to verification.	**1**	**2**	**5**
	b).	Describe the typical structure of a SystemVerilog testbench. What are the essential components?	**1**	**2**	**5**
		OR			
3.	**a).**	What are the advantages of using SystemVerilog for hardware verification compared to traditional Verilog?	**1**	**3**	**5**
	b).	Define the term "testbench" in the context of hardware verification. What is its purpose?	**1**	**1**	**5**
		UNIT-2			

4.	**a).**	Explain the difference between 2-state and 4-state data types in SystemVerilog. When would you choose one over the other?	**2**	**3**	**5**
	b).	Describe the use of dynamic arrays in SystemVerilog. How do they differ from static arrays?	**2**	**2**	**5**
		OR			
5.	**a).**	What are associative arrays, and what are some practical applications of using them in a verification environment?	**2**	**1**	**5**
	b).	Give an example scenario where using an associative array would be beneficial in a verification testbench.	**2**	**3**	**5**
		UNIT-3			
6.	**a).**	How are strings handled in SystemVerilog? Give an example of a common string operation used in verification.	**3**	**3**	**5**
	b).	Explain the concept of a union in SystemVerilog. When might you use a union in a verification environment?	**3**	**2**	**5**
		OR			
7.	**a).**	What are enumerated data types, and why are they useful in hardware verification?	**3**	**1**	**5**
	b).	Describe the role of events in SystemVerilog. How can they be used to synchronize different parts of a testbench?	**3**	**2**	**5**
		UNIT-4			
8.	**a).**	What is the purpose of an interface in SystemVerilog? How does it differ from a module in Verilog?	**4**	**3**	**5**
	b).	Explain the concept of modports. How do they contribute to interface design?	**4**	**2**	**5**
		OR			
9.	**a).**	What are clocking blocks, and why are they important in a verification environment?	**4**	**1**	**5**
	b).	Describe the use of virtual interfaces. When are they particularly useful?	**4**	**2**	**5**
		UNIT-5			
10.	**a).**	Explain the concepts of inheritance and polymorphism in SystemVerilog OOP. Give a simple example of each.	**5**	**2**	**5**
	b).	What is the difference between a shallow copy and a deep copy of an object? Why is this distinction important in verification?	**5**	**3**	**5**
		OR			
11.	**a).**	How do parameterized classes enhance code reusability in SystemVerilog?	**5**	**2**	**5**
	b).	Describe the use of "this" and "super" keywords in SystemVerilog classes.	**5**	**2**	**5**

CO-COURSE OUTCOME **KL-KNOWLEDGE LEVEL** **M-MARKS**

NOTE : Questions can be given as A,B splits or as a single Question for 10 marks

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
B23ECH501	Honors	--	--	3	1.5	30	70	3 Hrs.

DESIGN VERIFICATION USING SYSTEM VERILOG LABORATORY

(Honors Degree Course in ECE)

Course Objectives:

1	To familiarize students in HDL programming for the concepts of Digital Signal Processing (DSP), Computer Architecture and Organization.
2	To enhance students' comprehension of simulation and synthesis methodologies for Error Detection and Correction, Image Processing.

Course Outcomes:

S.No	Outcome	Knowledge Level
1	**Develop** HDL (Hardware Description Language) programs for various digital signal processing applications and computer architectures.	K3
2	**Develop** HDL (Hardware Description Language) programs for communications.	K3
3	**Design** HDL (Hardware Description Language) programs for image processing applications	K4

SYLLABUS

1	Design and implement a Finite Impulse Response (FIR) filter. Generation of bitfile, porting on FPGA and Programming it.
2	Extend the basic ALU to perform floating-point addition, subtraction, multiplication, and division. Generation of bitfile, porting on FPGA and Programming it.
3	Implement a Direct Memory Access (DMA) controller that manages data transfer between memory and peripherals. Generation of bitfile, porting on FPGA and Programming it.
4	Implement Hamming Code and Cyclic Redundancy Check (CRC) for error detection and correction. Generation of bitfile, porting on FPGA and Programming it.
5	Design a module that converts an RGB image to a grayscale image using the luminance formula. Implement the design using behavioral modeling in Verilog. Generation of bitfile, porting on FPGA and Programming it.
6	Assignment-1
7	Assignment-2

EDA Tools/Hardware Required:

1.	EDA Tool that supports FPGA programming including Xilinx Vivado.
2.	Cadence/Synopsys/Tanner or Equivalent Industry Standard CAD Tool.
3.	Desktop computer with appropriate Operating System that supports the EDA tools.

Reference Books:	
1	Digital Systems Design Using Verilog-Charles H. Roth.
2	Digital Design with Verilog - Course (nptel.ac.in)
3	https://www.sciencedirect.com/topics/computer-science/finite-impulse-response-filter
4	https://embeddedwala.com/Blogs/embeddedsystem/what-is-dma