REPORT

12-Week Industry Based Internship

on

"VLSI Design & Implementation using ISM Framework with RISC-V"

19-01-2026 to 10-04-2026

Venue: GTU – School of Engineering and Technology, Academic Block 5, GTU Campus, Chandkheda,

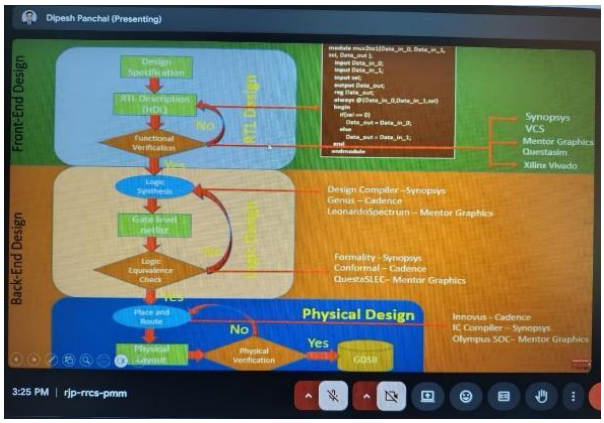
Ahmedabad-382424

Coordinator: Prof. Gautam D. Makwana

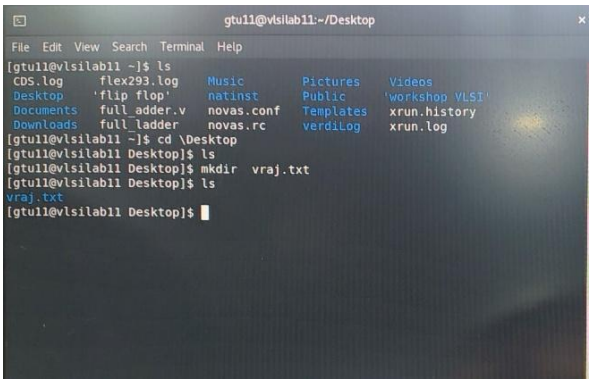
Week:1 Basic and Advanced Linux Commands with Shell Scripting & TCL Scripting

Date: 19-01-2026 to 23-01-2026

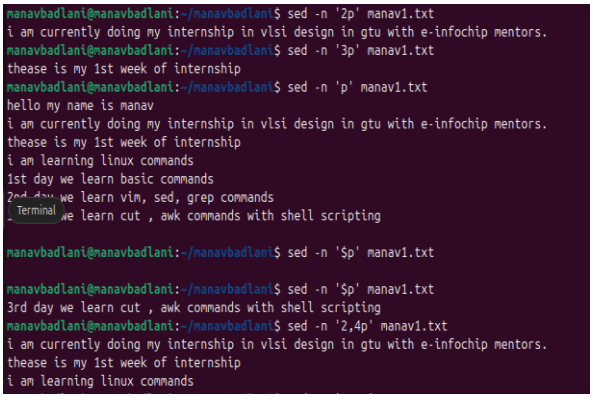
Session Details



ASIC Design Flow



Linux Basic Commands



Sed Commands

Topics covered

Objective:

- The primary objective of this week was to develop a strong foundation in Verilog HDL and understand its importance in digital system design and hardware modeling.
- The sessions aimed to introduce RTL (Register Transfer Level) concepts and enable students to model both combinational and sequential circuits effectively.
- Another key goal was to build clarity on fundamental concepts such as blocking and non-blocking assignments, which play a crucial role in synthesizable design.

Session Details:

- The week began with an introduction to Verilog HDL, where the structure of modules, input-output ports, and internal signal declarations were explained in detail.
- Students were introduced to important data types such as *wire* and *reg*, along with their usage in modeling different types of circuits.
- Various modeling styles, including behavioral, dataflow, and structural modeling, were discussed with practical examples to demonstrate their real-world applications.
- Further sessions introduced text-processing commands such as Grep and Sed, which are widely used for searching and modifying patterns in large design files and reports. Students practiced various commands and learned how these tools simplify repetitive text-processing operations.

The syntax of Tcl command is as follows:

```
commandName argument1 argument2 ... argumentN
```

Let's see a simple example of Tcl command:

```
#!/usr/bin/tclsh

puts "Hello, world!"
```

When the above code is executed, it produces the following result:

```
Hello, world!
```

In the above code, 'puts' is the Tcl command and "Hello World" is the argument1. As said before, we have used "" to group two words.

Let's see another example of Tcl command with two arguments:

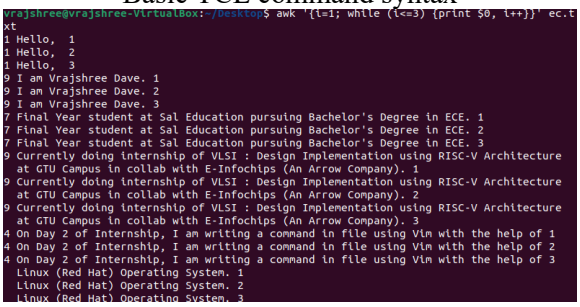
```
#!/usr/bin/tclsh

puts stdout "Hello, world!"
```

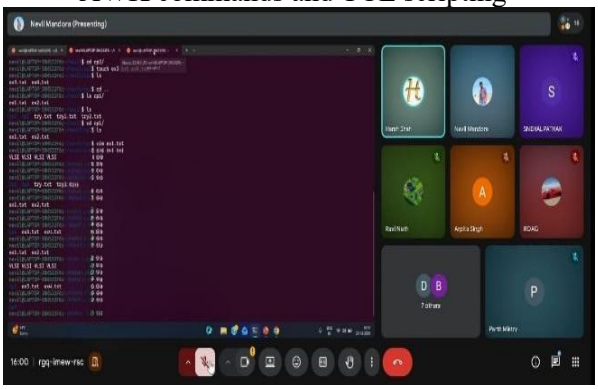
When above code is executed, it produces the following result:

```
Hello, world!
```

Basic TCL command syntax



AWK commands and TCL scripting



Reverse Presentations

- TCL scripting fundamentals were also covered during the week. The sessions included TCL syntax, variables, procedures, loops, conditional statements, and list handling concepts. The role of TCL scripting in automating EDA tools and design tasks was discussed in detail.

Topics Covered:

- ASIC Design Flow from RTL to GDSII implementation.
- Linux file system structure and command-line operations.
- VIM editor commands and text editing operations.
- Pattern searching using Grep and text substitution using Sed.
- Data extraction using Cut and text processing using Awk.
- Shell scripting concepts including loops and conditional statements.
- TCL scripting fundamentals used in VLSI automation.

Tools Used:

- Linux Operating System
- VIM Editor
- Grep, Sed, Cut, and Awk Commands
- Shell Scripting
- TCL Scripting

Learning Outcome:

- By the end of the week, students gained a clear understanding of the ASIC design flow, Linux commands, and scripting concepts. They also developed practical skills in automation, file handling, and technical communication through hands-on sessions and reverse presentations.

Week:2 Verilog Fundamentals and RTL Modeling

Date: 26-01-2026 to 30-01-2026

Session Details

Instructor explaining Verilog concepts to students during the training session

Students performing hands-on Verilog coding and simulation during the practical session

Instructors assisting students individually during hands-on Verilog coding session

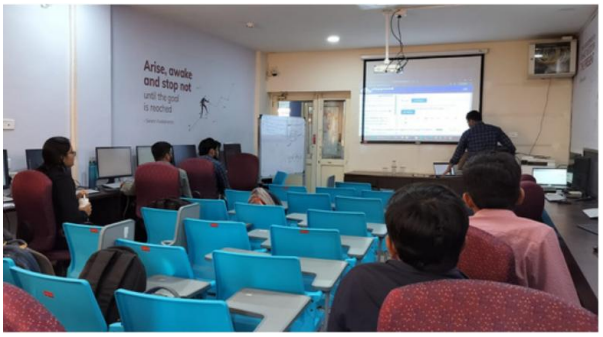
Topics Covered

Objectives:

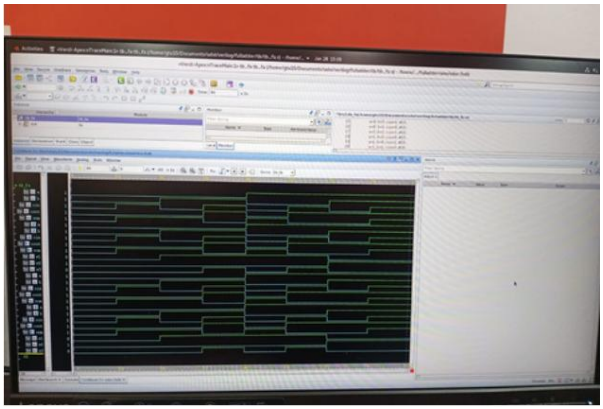
- The primary objective of this week was to develop a strong foundation in Verilog HDL and understand its role in RTL-based digital circuit design.
- The sessions focused on learning different modeling styles and implementing combinational and sequential circuits using Verilog HDL.
- Another important objective was to improve coding, simulation, and debugging skills through practical hands-on sessions and interactive discussions.

Session Details:

- The week began with an introduction to Verilog HDL, where students learned the structure of Verilog modules, input-output declarations, and commonly used data types such as *wire* and *reg*. Different modeling styles including behavioral, structural, and dataflow modeling were explained with practical examples to demonstrate their applications in digital circuit design.
- The sessions further focused on combinational circuit modeling, where students implemented circuits such as multiplexers, adders, encoders, and decoders using Verilog code. The concepts of continuous assignment statements and procedural blocks were discussed in detail to explain the behavior of combinational logic circuits.
- Sequential circuit modeling concepts were introduced later in the week, where students learned about flip-flops, registers, counters, and clock-based



Instructor presenting the output of combinational circuit by running verilog code.



Waveform generated by students

Reverse Presentations

logic design. Special emphasis was given to sensitivity lists and the role of clock signals in sequential circuits.

- The difference between blocking (=) and non-blocking (<=) assignments was explained through simulation examples and coding demonstrations. Students practiced writing and debugging Verilog code during hands-on sessions, which improved their understanding of synthesizable RTL design.

Topics Covered:

- Fundamentals of Verilog HDL and RTL design concepts.
- Verilog module structure, syntax, and data types.
- Behavioral, structural, and dataflow modeling styles.
- Combinational circuit design using Verilog HDL.
- Sequential circuit modeling and clock-driven logic design.
- Blocking and non-blocking assignments with practical examples.
- Simulation and waveform analysis techniques for digital circuits.

Tools Used:

- Verilog HDL for RTL coding and digital circuit modeling.
- Simulation tools for verifying functionality and analyzing output waveforms.
- Waveform viewers for debugging and timing analysis of digital circuits.

Learning Outcome:

- By the end of the week, students gained a clear understanding of Verilog HDL, RTL design concepts, and the implementation of combinational and sequential circuits.

Week:3 System Verilog Fundamentals and Verification Concepts

Date: 02-02-2026 to 06-02-2026

Session Details

Explanation of verilog concepts

Explanation of SystemVerilog layered testbench code

Hands on Verilog coding and simulation

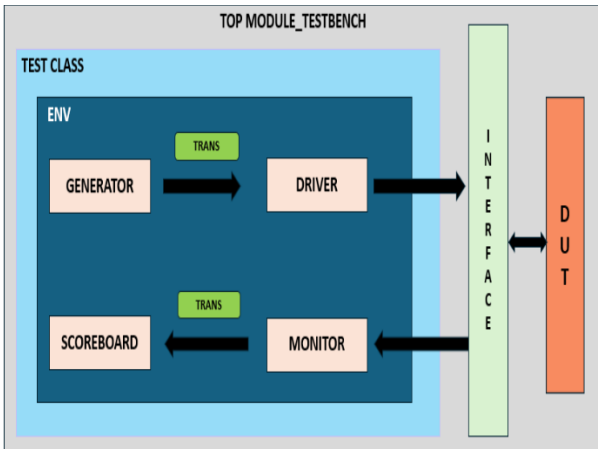
Topics Covered

Objectives:

- The primary objective of this week was to introduce students to System Verilog and understand its role in verification-oriented digital design.
- The sessions focused on learning SystemVerilog data types, Object-Oriented Programming (OOP) concepts, synchronization techniques, and layered testbench architecture.
- Another important objective was to improve practical verification skills through hands-on coding sessions and interactive discussion

Session Details:

- The week began with an introduction to SystemVerilog data types, where students learned about 2-state and 4-state data types along with their applications in RTL and verification environments. Different array types including fixed arrays, dynamic arrays, associative arrays, and queues were also discussed with practical examples.
- The sessions further focused on Object-Oriented Programming concepts in SystemVerilog, including classes, objects, inheritance, and copy mechanisms. Practical examples were demonstrated to explain the importance of OOP concepts in creating reusable and modular verification environments.
- Advanced verification concepts such as semaphores, mailboxes, interfaces, and clocking blocks were introduced during the sessions. Students learned how synchronization and communication mechanisms are used in verification environments to avoid race conditions



Top-module testbench demonstration


Reverse Presentations


Reverse Presentations

and improve testbench reliability.

- Reverse presentation sessions and interactive discussions were conducted at the end of the week, where students explained the concepts learned throughout the sessions. These activities helped improve conceptual clarity, presentation skills, and technical communication abilities.

Topics covered:

- Fundamentals of SystemVerilog and verification concepts.
- 2-state and 4-state data types.
- Fixed, dynamic, associative, and queue arrays.
- Object-Oriented Programming concepts including classes and inheritance.
- Synchronization techniques using semaphores and mailboxes.

Tools Used:

- SystemVerilog for verification-oriented coding and simulation.
- Simulation tools for executing and verifying SystemVerilog testbenches.
- Waveform analysis tools for observing verification results and timing behavior.

Learning outcome:

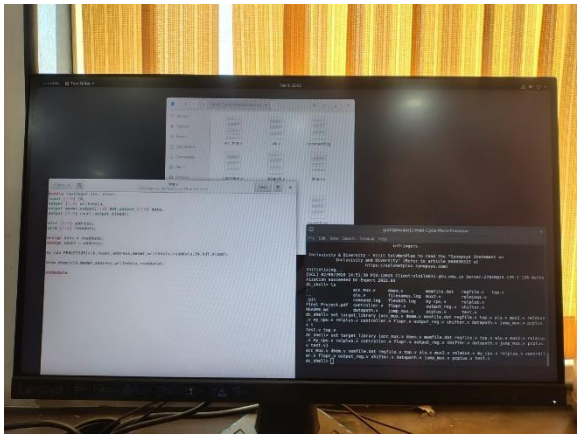
- By the end of the week, students gained a clear understanding of SystemVerilog fundamentals, OOP concepts, and layered verification environments. They also improved their practical coding, simulation, and verification skills through hands-on sessions and technical discussions.

Week:4 Introduction to DFT and Scan Architecture Concepts

Date: 09-02-2026 to 13-02-2026

Session Details

Introduction to DFT



Design Rule Check Rule (DRCs)

Topics Covered

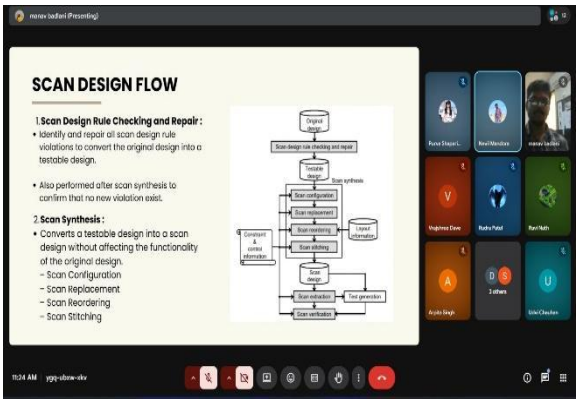
Objectives:

- The primary objective of this week was to introduce students to Design for Testability (DFT) concepts and understand the importance of testing in ASIC design.
- The sessions focused on fault models, scan architectures, scan synthesis flow, and Design Rule Checks (DRCs) used in digital circuits.
- Another important objective was to improve students' understanding of scan-based testing techniques and DFT verification methodologies through practical discussions and reverse presentations.

Session Details:

- The week began with an introduction to Design for Testability (DFT), where students learned about testing metrics, fault models, and the importance of scan architectures in improving ASIC testability.
- The scan design flow including scan synthesis, scan insertion, and scan extraction was explained in detail, along with the implementation of scan chains in digital circuits.
- The sessions also covered DFT-related Design Rule Checks (DRCs) such as gated clocks, combinational loops, asynchronous reset signals, and bidirectional I/O issues affecting testability.
- Reverse presentations, technical discussions, and Linux practice sessions were conducted to strengthen conceptual understanding and improve communication skills.

Design for Testability (DFT) Presentation Session



Reverse Presentations

Topics covered:

- Fundamentals of Design for Testability (DFT).
- Testing metrics and common fault models.
- Full scan and partial scan architectures.
- Scan synthesis, scan insertion, and scan extraction flow.
- DFT-related Design Rule Checks (DRCs).
- Gated clocks, combinational loops, and asynchronous reset issues.
- Linux command practice and revision of scan flow concepts.

Tools Used:

- Scan synthesis and scan insertion commands.
- DFT verification and DRC checking commands.
- Linux environment for command-line operations and automation practice.

Learning outcome:

- By the end of the week, students gained a clear understanding of DFT concepts, scan architectures, and fault models used in ASIC testing. They also improved their understanding of scan synthesis flow, DRC verification techniques, and testing methodologies through practical discussions and reverse presentation activities.

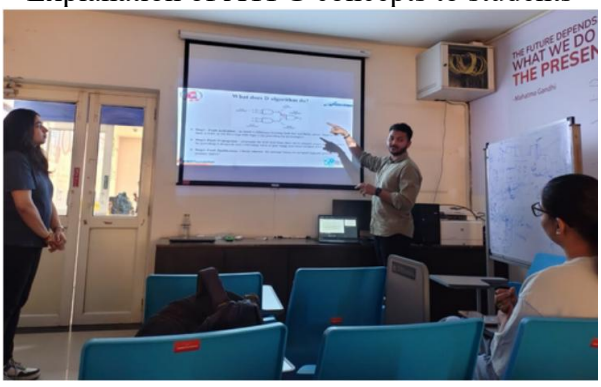
Week:5 Introduction to ATPG Concepts

Date: 16-02-2026 to 20-02-2026

Session Details

Explanation of ATPG concepts to students

Explanation of ATPG concepts to students



Reverse Presentation

Topics Covered

Objective:

- The primary objective of this week was to introduce students to Automatic Test Pattern Generation (ATPG) and understand its importance in Design for Testability (DFT).
- The sessions focused on fault models, fault coverage analysis, simulation concepts, and RTL synthesis flow using industry-standard methodologies.
- Another important objective was to improve practical understanding of testing and synthesis processes through hands-on sessions and reverse presentations.

Session Details:

- The week began with an introduction to ATPG concepts, where students learned about stuck-at faults, fault coverage, and the importance of test pattern generation in digital circuit testing. Reverse presentation sessions were also conducted to reinforce previously learned DFT concepts and improve conceptual clarity.
- The sessions further focused on simulation activities and verification concepts. Students learned the importance of simulation in validating circuit functionality and understanding digital design behavior before hardware implementation.
- RTL synthesis concepts were introduced using the open-source synthesis tool Yosys. Students learned the synthesis flow including reading Verilog files, elaboration, optimization,

Reverse Presentation

Hand on Practice sessions


Synthesis by student

technology mapping, and generation of synthesized netlists.

- Practical sessions and interactive discussions were conducted throughout the week to strengthen understanding of ATPG, simulation, and synthesis methodologies. Students actively participated in coding, debugging, and technical discussions to improve practical implementation skills.

Topics covered:

- Fundamentals of Automatic Test Pattern Generation (ATPG).
- Fault models and fault coverage analysis.
- Simulation and verification concepts.
- RTL synthesis flow and technology mapping.
- Optimization and netlist generation using Yosys.
- Reverse presentation and technical discussion activities.

Tools Used:

- Yosys for RTL synthesis and technology mapping.
- Simulation tools for functional verification and debugging.
- Verilog HDL for RTL coding and testing activities.

Learning Outcome:

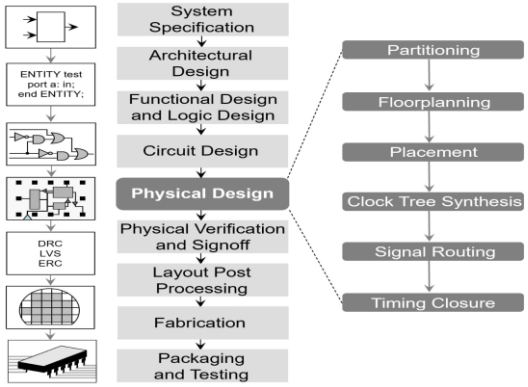
- By the end of the week, students gained a clear understanding of ATPG concepts, simulation methodologies, and RTL synthesis flow. They also improved their practical skills in synthesis, debugging, and functional verification through hands-on sessions and technical discussions.

Week:6 Physical Design Flow, Floor Planning, Placement and Routing (PnR)

Date: 23-02-2026 to 27-02-2026

Session Details

Introduction to Physical Design (PD)



Physical Design Flow

Synthesis of Top Level 1 Counter Design

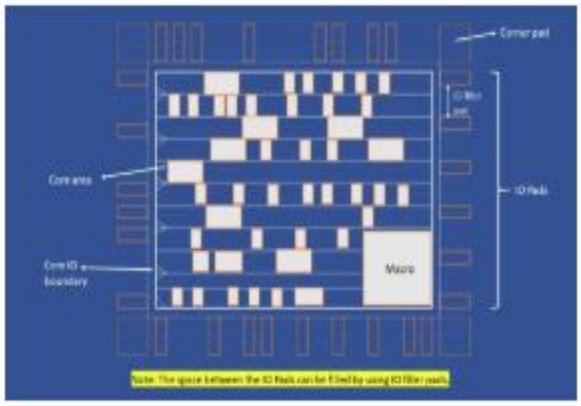
Topics Covered

Objective:

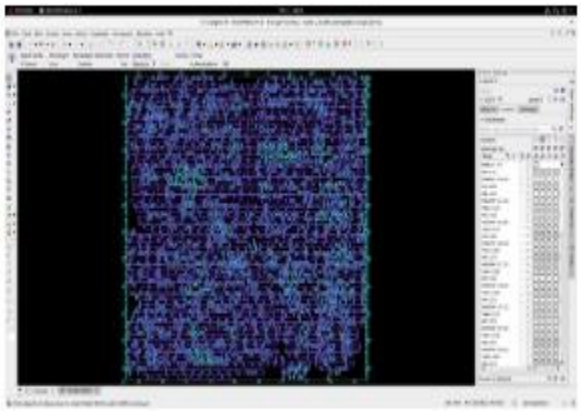
- The primary objective of this week was to introduce students to the Physical Design (PD) flow used in ASIC implementation and understand the stages involved in converting synthesized RTL into physical layout.
- The sessions focused on synthesis concepts, floor planning, macro placement, power planning, and placement methodologies used in backend VLSI design.
- Another important objective was to familiarize students with timing analysis concepts and Physical Design tools used in industry workflows.

Session Details:

- The week began with an introduction to the Physical Design flow, where students learned about the importance of backend design in ASIC implementation. Various input files required for Physical Design, including Verilog netlists, LEF files, library files, and timing constraints, were discussed in detail.
- The sessions further focused on synthesis stages such as translation, optimization, and technology mapping. Timing analysis concepts including critical paths, slack calculation, and timing violations were also explained with practical examples.
- Floor planning concepts including core size estimation, macro placement, standard cell row creation, and fly line analysis were introduced during the sessions. Students also learned about power planning techniques such as power rings, power stripes, and IR drop



Synthesis of Top Level 1 Counter Design



Power Planning of i2c_master_top design

Hands on session

reduction methods used in chip design.

- Placement concepts including global placement, detailed placement, placement blockages, and routing considerations were discussed in detail. Practical discussions and demonstrations helped students understand the overall backend Physical Design flow.

Topics covered:

- Introduction to Physical Design (PD) flow.
- Input files and constraints required for backend implementation.
- Synthesis stages and timing analysis concepts.
- Floor planning and macro placement methodologies.
- Power planning techniques and IR drop reduction.
- Placement flow and placement blockage concepts.
- Routing considerations in backend Physical Design.

Tools Used:

- Synopsys Design Compiler (DC).
- ICC2 tool environment for backend Physical Design concepts.
- Timing analysis and report generation commands.

Learning Outcome:

- By the end of the week, students gained a clear understanding of the Physical Design flow, floor planning, placement, and power planning concepts used in ASIC implementation. They also improved their understanding of timing analysis and backend design methodologies through practical discussions and technical sessions.

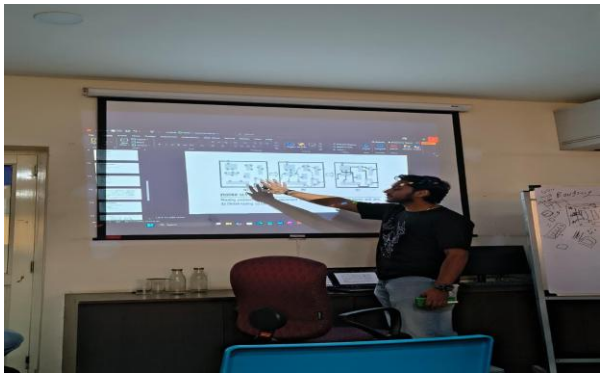
Week:7 Physical Design Flow, Floor Planning, Placement and Routing (PnR)

Date: -02-2026 to 20-02-2026

Session Details

Explanation of Place and Route (PnR) flow

Interactive lecture



Explanation of Routing process

Topics Covered

Objective:

- The primary objective of this week was to introduce students to the Place and Route (PnR) flow and understand its role in Physical Design implementation.
- The sessions focused on sign-off methodologies, Physical Verification techniques, and verification checks required before fabrication.
- Another important objective was to improve students' understanding of backend verification concepts through practical discussions, practice sessions, and reverse presentations.

Session Details:

- The week began with an introduction to the PnR flow, where students learned the stages involved in converting synthesized netlists into manufacturable physical layouts. Concepts such as floor planning, placement, routing, congestion handling, and timing closure were discussed in detail.
- The sessions further focused on sign-off and Physical Verification concepts, including Design Rule Checks (DRC) and Layout Versus Schematic (LVS) verification. Students learned the importance of these verification techniques in ensuring layout correctness and manufacturability.
- Practice sessions were conducted where students analyzed common layout violations and discussed methods for resolving DRC and LVS errors. Interactive technical discussions

Reverse Presentation

Reverse Presentation

Explanation of Formality Flow

helped strengthen understanding of verification methodologies used in backend VLSI design.

- Reverse presentation sessions were also conducted during the week, where students explained concepts related to PnR flow, routing methodologies, and Physical Verification. These activities improved conceptual clarity, technical communication, and presentation skills.

Topics Covered:

- Fundamentals of Place and Route (PnR) flow.
- Floor planning, placement, and routing concepts.
- Congestion handling and timing closure techniques.
- Sign-off methodologies and backend verification flow.
- Design Rule Checks (DRC) and Layout Versus Schematic (LVS).
- Physical Verification concepts and layout validation techniques.
- Reverse presentations and technical discussions.

Tools used:

- Physical Design and verification tools for layout analysis.
- DRC and LVS verification methodologies.
- Backend design flow and routing analysis environments.

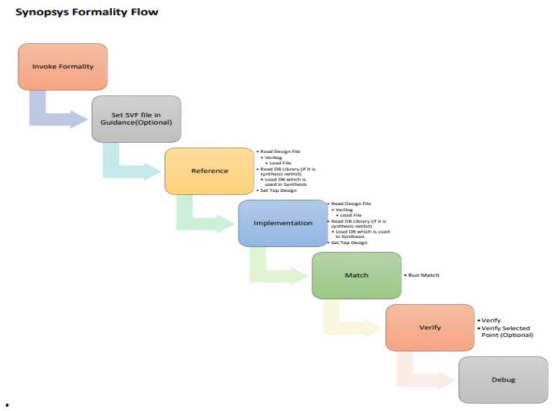
Learning Outcome:

- By the end of the week, students gained a clear understanding of the PnR flow, sign-off methodologies, and Physical Verification techniques used in backend ASIC design.

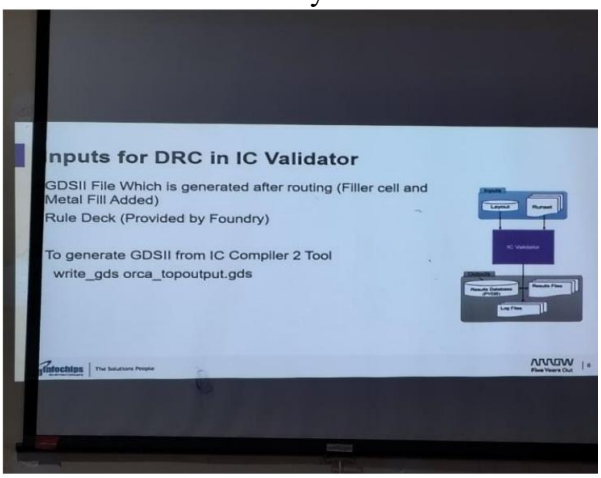
Week:8 Formality Flow, PrimeTime Analysis, StarRC and IC Validator (ICV)

Date: 09-03-2026 to 13-02-2026

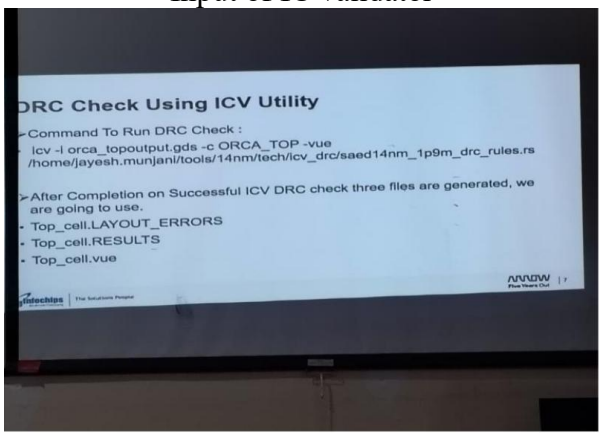
Session Details



Formality Flow



Input of IC validator



Output of IC validator

Topics Covered

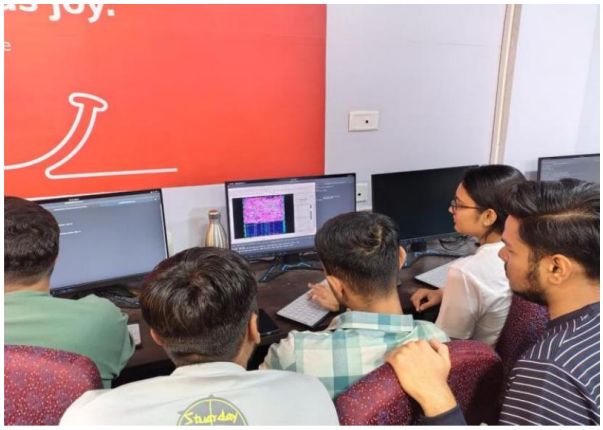
Objective:

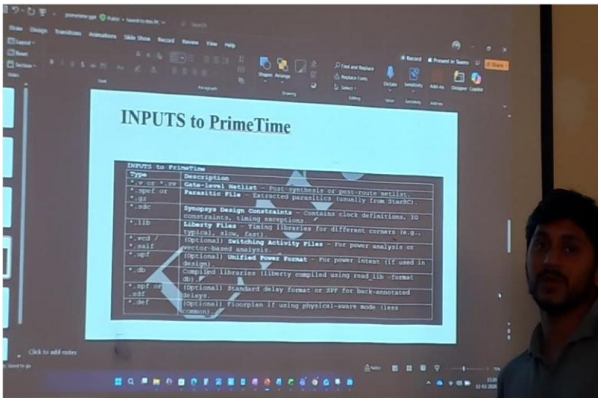
- The primary objective of this week was to introduce students to advanced backend verification and timing analysis tools used in Physical Design sign-off flow.
- The sessions focused on Formality flow, PrimeTime timing analysis, StarRC parasitic extraction, and IC Validator (ICV) verification methodologies.
- Another important objective was to improve students' understanding of timing closure, physical verification, and backend analysis through practical sessions and technical discussions.

Session Details:

- The week began with an introduction to the PnR flow, where students learned the stages involved in converting synthesized netlists into manufacturable physical layouts. Concepts such as floor planning, placement, routing, congestion handling, and timing closure were discussed in detail.
- The sessions further focused on sign-off and Physical Verification concepts, including Design Rule Checks (DRC) and Layout Versus Schematic (LVS) verification. Students learned the importance of these verification techniques in ensuring layout correctness and manufacturability.
- Practice sessions were conducted where students analyzed common layout violations and discussed methods for resolving DRC and LVS errors. Interactive technical discussions helped strengthen understanding of

Hands on session on starRC Flow


Hands on session on starRC Flow



Explanation of PrimeTime

verification methodologies used in backend VLSI design.

- Reverse presentation sessions were also conducted during the week, where students explained concepts related to PnR flow, routing methodologies, and Physical Verification. These activities improved conceptual clarity, technical communication, and presentation skills.

Topics Covered:

- Fundamentals of Place and Route (PnR) flow.
- Floor planning, placement, and routing concepts.
- Congestion handling and timing closure techniques.
- Sign-off methodologies and backend verification flow.
- Design Rule Checks (DRC) and Layout Versus Schematic (LVS).
- Physical Verification concepts and layout validation techniques.
- Reverse presentations and technical discussions.

Tools Used:

- Synopsys Formality for Logical Equivalence Checking.
- IC Validator (ICV) for physical verification.
- StarRC for parasitic extraction.
- PrimeTime for Static Timing Analysis (STA).

Learning OUtcome:

- By the end of the week, students gained a clear understanding of Formality flow, timing analysis, parasitic extraction, and physical verification techniques used in backend ASIC design.

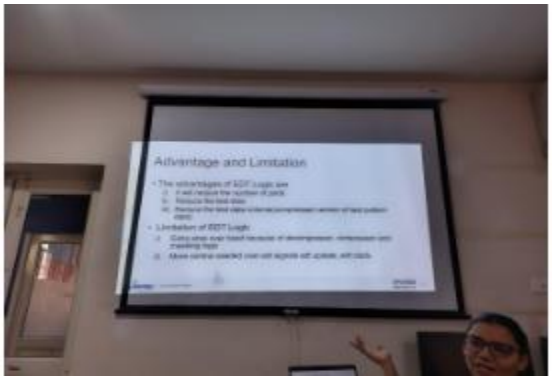
Week:9 Formality Flow, Prime Time Analysis, Star RC and IC Validator (ICV)

Date: 16-03-2026 to 20-03-2026

Session Details

Hands-on Linux Lab Session

Verilog Revision Lecture



EDT Advantages and Limitations

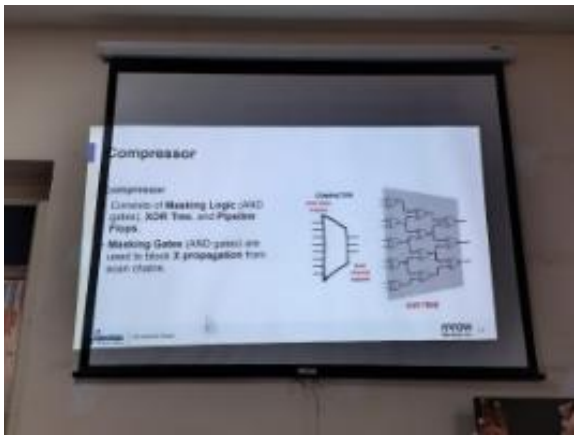
Topics Covered

Objective:

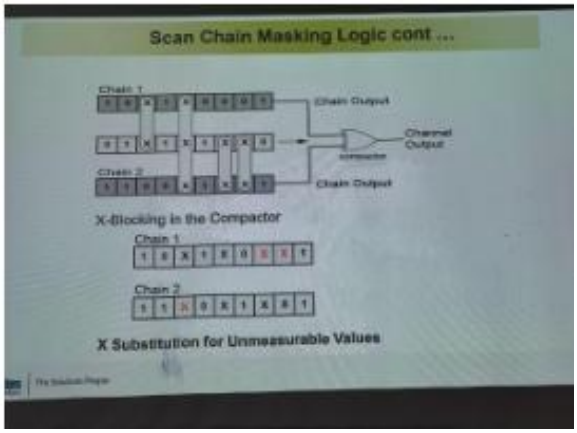
- The primary objective of this week was to revise the concepts covered throughout the internship, including Linux commands, Verilog HDL, Design for Testability (DFT), and scripting techniques.
- The sessions focused on strengthening students' understanding of RTL design, verification concepts, ATPG methodologies, and advanced testing techniques such as EDT compression logic.
- Another important objective was to improve problem-solving abilities and practical implementation skills through technical discussions and revision activities.

Session Details:

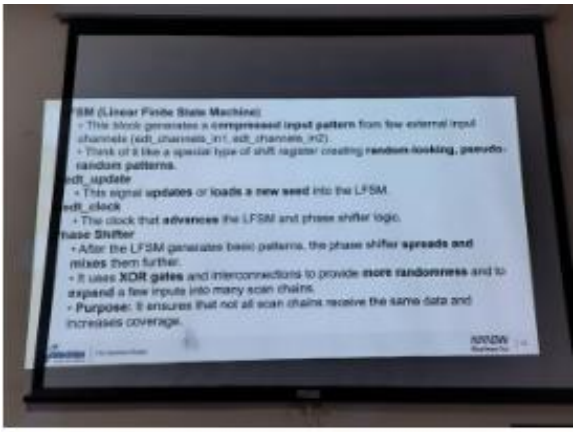
- The week began with revision sessions on Linux commands, Shell scripting, TCL scripting, and text-processing utilities. Students practiced advanced command-line operations, scripting syntax, loops, conditional statements, and automation techniques commonly used in VLSI workflows.
- The sessions further focused on Verilog HDL revision, including module structure, modeling styles, data types, FSM design, sequence detectors, and synthesizable coding practices. Practical discussions were conducted on RTL coding mistakes and debugging techniques.
- DFT concepts and ATPG methodologies were revised in detail,



EDT Compressor Architecture



LFSM and Phase filter



Scan masking & X-blocking

including fault models, scan architectures, fault coverage analysis, and EDT (Embedded Deterministic Test) compression logic. Students learned concepts such as LFSM decompressor, phase shifter, XOR compactor, masking logic, and scan chain handling techniques.

- Interactive discussions and doubt-solving sessions were conducted throughout the week to reinforce conceptual understanding and improve technical communication skills. Practical examples and architecture explanations helped students strengthen their understanding of advanced testing methodologies.

Topics Covered:

- Revision of Linux commands and scripting concepts.
- Verilog HDL modeling styles and RTL coding practices.
- FSM design and sequence detector implementation.
- Design for Testability (DFT) and ATPG concepts.
- EDT compression architecture and scan chain handling.
- Fault models, fault coverage, and testing methodologies.
- Technical discussions and practical revision activities.

Tools Used:

- Linux environment and Shell/TCL scripting tools.
- Verilog HDL for RTL coding and simulation activities.
- DFT and ATPG concepts for testing methodology discussions.

Learning Outcome:

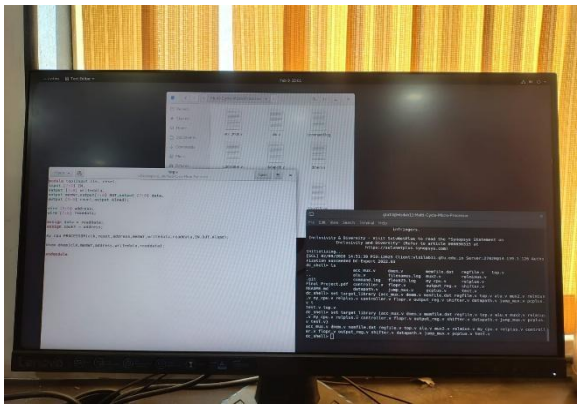
- By the end of the week, students strengthened their understanding of Linux, Verilog HDL, DFT concepts, and advanced testing methodologies.

Week:10 Reverse Presentations and Final Technical Discussions

Date: 23-03-2026 to 27-03-2026

Session Details

Introduction to DFT



Multi-Cycle Microprocessor Design: RTL

Development and DC Synthesis

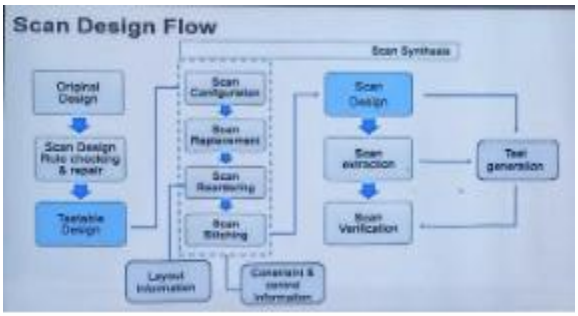
Topics Covered

Objective:

- The primary objective of this week was to revise and strengthen the concepts learned throughout the internship through reverse presentations and technical discussions.
- The sessions focused on Verilog HDL, Design for Testability (DFT), Physical Design flow, FSM concepts, and mini-project discussions.
- Another important objective was to improve students' technical presentation skills, conceptual clarity, and problem-solving abilities through interactive activities and discussions.

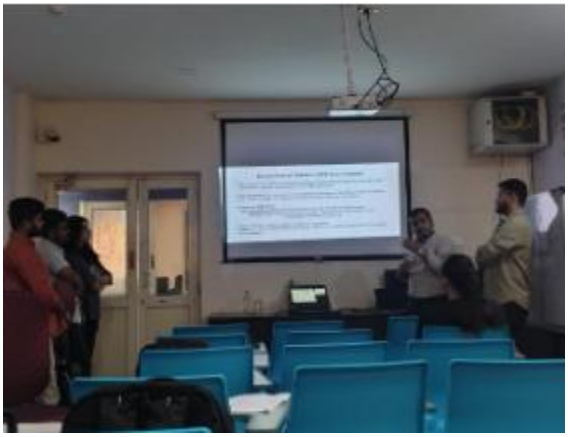
Session Details:

- The week began with reverse presentation sessions on Verilog HDL and digital design concepts, where students presented topics such as modeling styles, RTL design approaches, FSM implementation, and sequence detector concepts. These activities helped reinforce understanding of front-end design methodologies.
- The sessions further focused on Design for Testability (DFT) concepts, including scan architectures, fault models, ATPG flow, and scan chain methodologies. Technical discussions were conducted on improving testability and understanding practical testing challenges in ASIC design.
- Physical Design concepts including floor planning, power planning, placement, routing, and timing analysis were also revised during the sessions. Students discussed backend



Scan Design Flow


Design for Testability-Reverse Presentation


Overview of Adhoc DFT Techniques


Hands on Practice

implementation flow and sign-off concepts used in ASIC Physical Design.

- Interactive discussions and QnA sessions were conducted on FSM design, Verilog implementation logic, and mini-project ideas such as counters, sequence detectors, and controller designs. These sessions helped improve practical understanding and technical communication skills.

Topics Covered:

- Verilog HDL and RTL design concepts.
- FSM implementation and sequence detector design.
- Design for Testability (DFT) and ATPG methodologies.
- Scan architectures and fault models.
- Physical Design flow and timing analysis concepts.
- Placement, routing, and sign-off methodologies.
- Mini-project discussions and practical implementation ideas.

Tools Used:

- Verilog HDL for RTL design discussions and implementation concepts.
- DFT and Physical Design methodologies for backend flow discussions.
- Presentation tools and waveform analysis examples for technical explanation sessions.

Learning Outcome:

- By the end of the week, students strengthened their understanding of Verilog HDL, DFT concepts, and Physical Design methodologies. The week concluded with reverse presentations.

Week:11 RTL to GDSII Project Definition and Implementation

Date: 30-03-2026 to 03-04-2026

Session Details

Reverse Presentation

Reverse Presentation

Reverse Presentation

Topics Covered

Objective:

- The primary objective of this week was to introduce students to complete RTL to GDSII project implementation flow and understand the integration of front-end and back-end VLSI concepts in practical design projects.
- The sessions focused on project definition methodologies, design planning, implementation flow, and practical understanding of ASIC development stages.
- Another important objective was to improve project analysis, implementation planning, and practical problem-solving skills through guided discussions and practice sessions.

Session Details:

- Practical sessions and discussions were conducted throughout the week to help students understand design methodologies, implementation challenges, and optimization considerations involved in project development. Students actively participated in technical discussions and implementation planning activities.
- Reverse presentations and interactive QnA sessions were also conducted to strengthen conceptual clarity and improve technical communication skills related to project implementation methodologies.
- Practical sessions and discussions were conducted throughout the week to help students understand design methodologies, implementation challenges, and optimization considerations involved in project

Reverse Presentation


Reverse Presentation


Felicitation of Instructor from eInfochips

development. Students actively participated in technical discussions and implementation planning activities.

- Reverse presentations and interactive QnA sessions were also conducted to strengthen conceptual clarity and improve technical communication skills related to project implementation methodologies.

Topics Covered:

- RTL to GDSII implementation flow.
- ASIC project definition and design planning methodologies.
- Integration of RTL, DFT, synthesis, and Physical Design concepts.
- Verification and backend implementation considerations.
- Project implementation challenges and optimization techniques.
- Technical discussions and practical implementation planning activities.

Tools Used:

- RTL design and verification methodologies for project discussions.
- Backend Physical Design concepts and implementation workflows.
- Technical presentation and project planning methodologies.

Learning Outcome:

- By the end of the week, students gained a clear understanding of RTL to GDSII project implementation flow and improved their understanding of integrating front-end and back-end VLSI concepts into practical ASIC development projects.

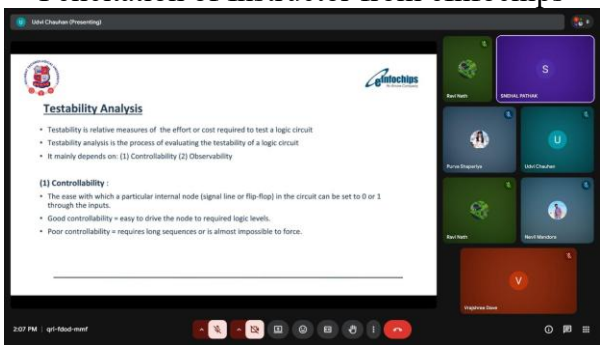
Week:12 RTL to GDSII Project Development and Reverse Presentation

Date: 06-04-2026 to 10-04-2026

Session Details

Felicitation of Instructor from eInfochips

Felicitation of Instructor from eInfochips



Student giving reverse presentation in Online Mode

Topics Covered

Objective:

- The primary objective of this week was to continue RTL to GDSII project activities and strengthen students' understanding of practical ASIC design implementation methodologies.
- The sessions focused on project execution, implementation discussions, reverse presentations, and technical evaluation of project concepts.
- Another important objective was to improve technical presentation skills, implementation planning, and overall conceptual understanding of the complete VLSI design flow.

Session Details:

- The week focused on RTL to GDSII project definition and implementation discussions, where students continued learning practical ASIC development methodologies and implementation flow. Students analyzed project requirements, implementation stages, and backend considerations involved in design execution.
- Technical discussions were conducted on project planning, verification flow, Physical Design considerations, and optimization techniques required for successful ASIC implementation.
- Practice sessions and implementation discussions helped students strengthen their understanding of integrating RTL design, DFT, synthesis, and backend Physical Design concepts into complete project workflows.
- Reverse presentation sessions were conducted at the end of the week, where

Felicitation of Instructor from eInfochips

Felicitation of Instructor from eInfochips

Students and instructors taking group pictures

students explained their understanding of project implementation methodologies and practical design flow concepts. These activities improved technical communication and presentation skills.

Topics Covered:

- RTL to GDSII project implementation flow.
- ASIC design planning and implementation methodologies.
- Verification and Physical Design considerations in projects.
- Integration of front-end and back-end VLSI concepts.
- Project optimization and implementation discussions.
- Reverse presentations and technical evaluation sessions.

Tools Used:

- RTL design and backend implementation methodologies.
- Verification and Physical Design concepts for project discussions.
- Technical presentation and implementation planning techniques.

Learning Outcome:

- By the end of the week, students strengthened their understanding of practical RTL to GDSII implementation flow and improved their technical presentation, project analysis, and implementation planning skills through discussions and reverse presentation activities.

GUJARAT TECHNOLOGICAL UNIVERSITY

School of Engineering and Technology

12 Weeks Internship on VLSI Design and implementation using ISM framework with RISC-V

19th January 2026 to 10th April 2026

Participant List

Sr. No.	Name of Student (As required in the Certificate)	Enrolment No.	Name of GTU Affiliated College
1	VRAJSHREE S. DAVE	230673111002	SAL Institute of Technology and Engineering Research
2	PARTH PRASHANTKUMAR MISTRY	230133111019	Government Engineering College, Gandhinagar
3	SNEHAL KISHORBHAI PATHAK	220170111094	Vishwakarma Government Engineering College
4	BADLANI MANAV NARESHBHAI	220210111003	Government Engineering College, Bhavnagar
5	MANDORA NEVIL GORDHANBHAI	220200111008	Government Engineering College, Rajkot
6	KANANI DIP PRAVINBHAI	230283111010	L. D. COLLEGE OF ENGINEERING
7	CHAUHAN UDVI DIVYESH	220130111021	Government Engineering College, Gandhinagar
8	PATEL YASH ARVINDBHAI	230133111032	Government Engineering College, Gandhinagar
9	PANCHAL BHARGAV SANJAYBHAI	230283111023	L. D. COLLEGE OF ENGINEERING
10	DHRUVI CHAVDA	220170111025	Vishwakarma Government Engineering College
11	MALEK MOHAMMAD TAWHID SAJEEDMIYA	230283111014	L. D. COLLEGE OF ENGINEERING
12	ARPITA SINGH	230673111001	SAL Institute of Technology & Engineering Research
13	PATEL RUDRA KRUNALKUMAR	220130111149	Government Engineering College, Gandhinagar
14	RAVI NATH JOGI	220670111001	SAL Institute of Technology and Engineering Research
15	DHUMALE JITENKUMAR ARUNBHAI	251370796003	GTU-SCHOOL OF ENGINEERING & TECHNOLOGY
16	SOLANKI PARJANYA KESHUBHAI	251370796001	GTU-SCHOOL OF ENGINEERING & TECHNOLOGY
17	PATEL APURVA SURESHBHAI	251370796004	GTU-SCHOOL OF ENGINEERING & TECHNOLOGY