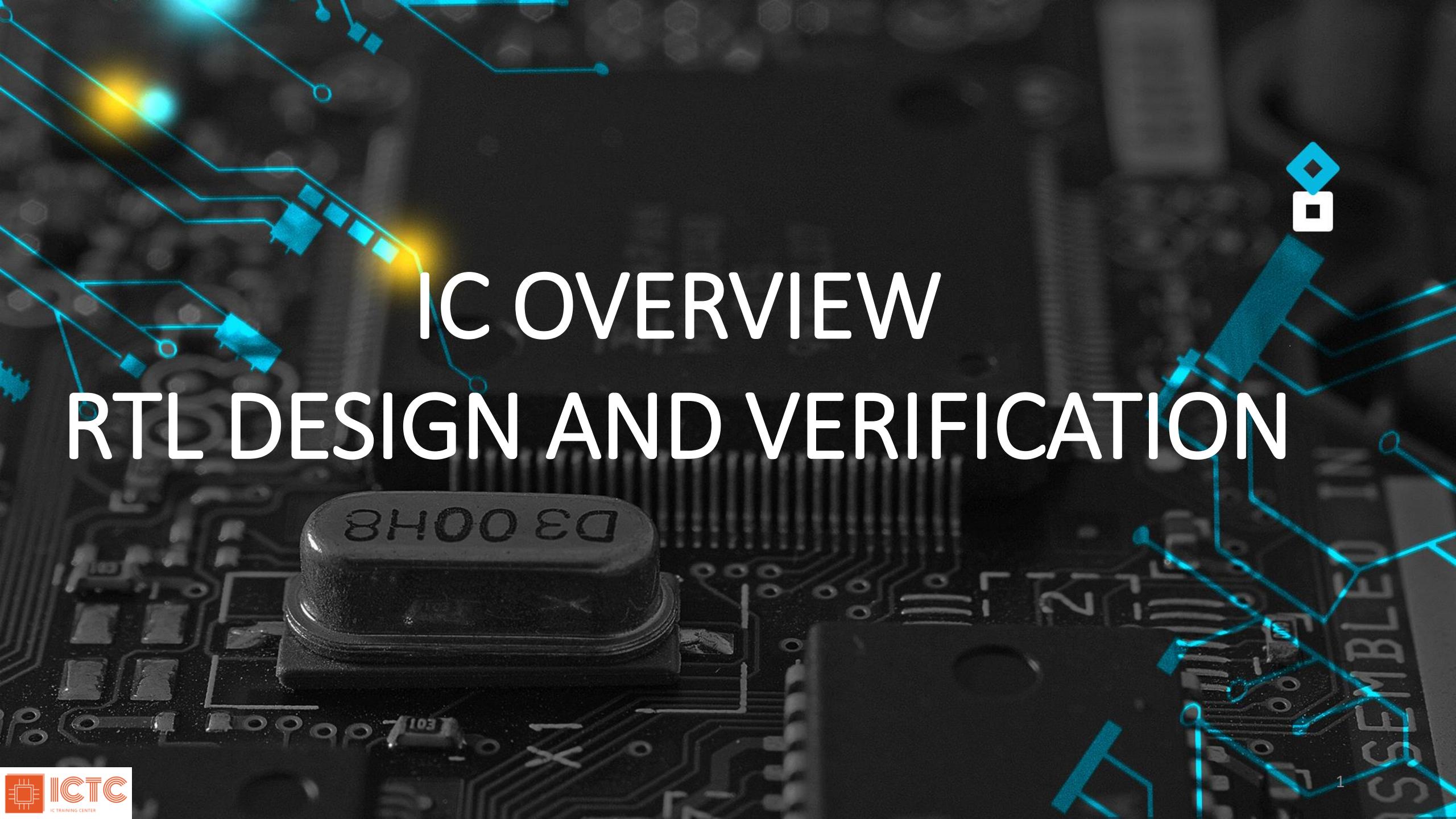
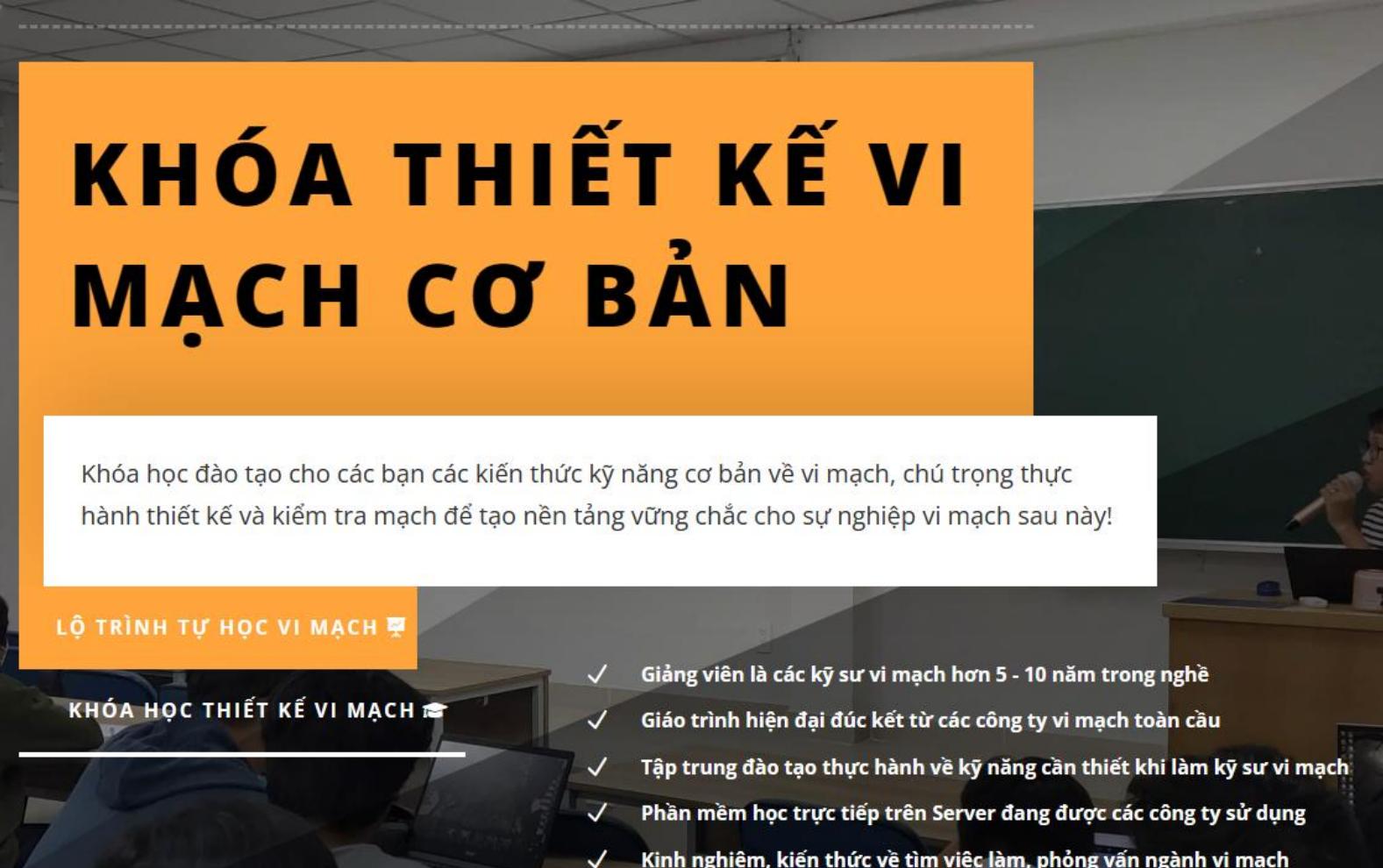


IC OVERVIEW RTL DESIGN AND VERIFICATION



COURSE INTRODUCTION

[Khóa Học Thiết Kế Vi Mạch Cơ Bản - Trung Tâm Đào Tạo Thiết Kế Vi Mạch ICTC](#)



The image shows a composite view. On the left is a large orange rectangular overlay containing the course title "KHÓA THIẾT KẾ VI MẠCH CƠ BẢN". Below it is a white text box with descriptive text. At the bottom left is a smaller inset showing a person working at a computer. On the right side, there's a photograph of a classroom where a teacher is speaking into a microphone at a podium.

**KHÓA THIẾT KẾ VI
MẠCH CƠ BẢN**

Khóa học đào tạo cho các bạn các kiến thức kỹ năng cơ bản về vi mạch, chú trọng thực hành thiết kế và kiểm tra mạch để tạo nền tảng vững chắc cho sự nghiệp vi mạch sau này!

LỘ TRÌNH TỰ HỌC VI MẠCH

KHÓA HỌC THIẾT KẾ VI MẠCH

- ✓ Giảng viên là các kỹ sư vi mạch hơn 5 - 10 năm trong nghề
- ✓ Giáo trình hiện đại đúc kết từ các công ty vi mạch toàn cầu
- ✓ Tập trung đào tạo thực hành về kỹ năng cần thiết khi làm kỹ sư vi mạch
- ✓ Phần mềm học trực tiếp trên Server đang được các công ty sử dụng
- ✓ Kinh nghiệm, kiến thức về tìm việc làm, phỏng vấn ngành vi mạch

COURSE INTRODUCTION



SUMMARY



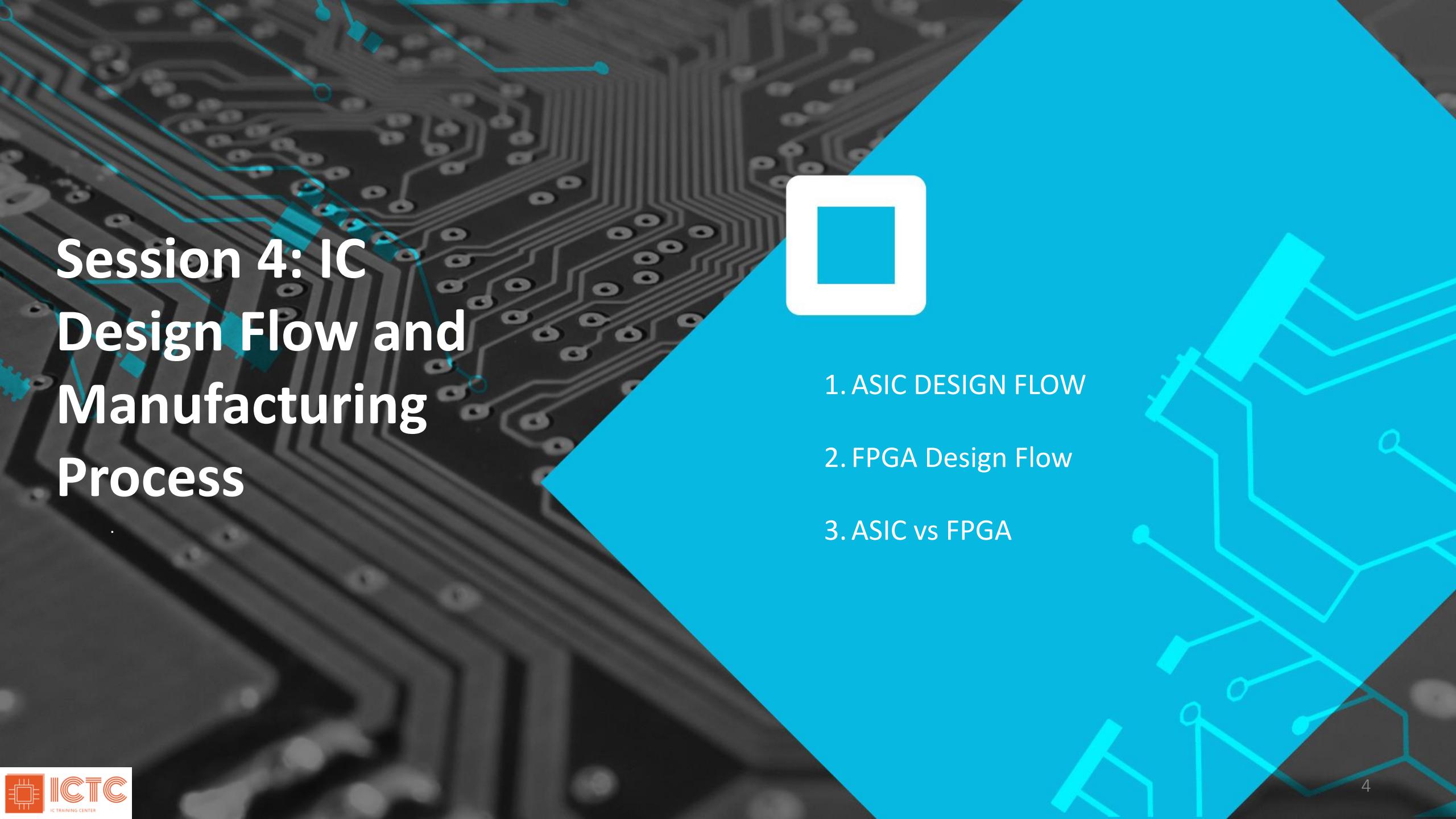
HOMEWORK



QUESTION



SELF-LEARNING



Session 4: IC Design Flow and Manufacturing Process



1. ASIC DESIGN FLOW
2. FPGA Design Flow
3. ASIC vs FPGA



ASIC Design Process

What is ASIC ?

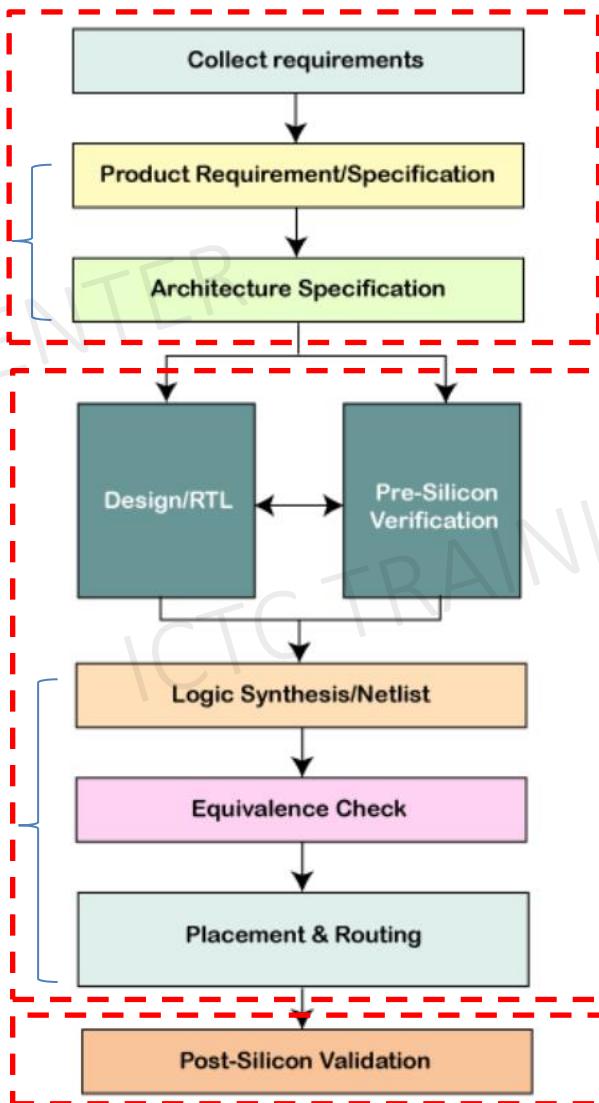


ASIC stands for Application-Specific Integrated Circuit.

- ❑ It's a type of integrated circuit customized for a particular use rather than general-purpose use.
- ❑ ASICs are designed to perform a specific task or set of tasks, which could range from simple functions like managing power in electronic devices to complex computations like those required for cryptocurrency mining.
- ❑ ASICs are often used in situations where software solutions are not efficient or suitable for the task.

ASIC Design Process

Digital



Stage 1: Customer requirements and architecture design

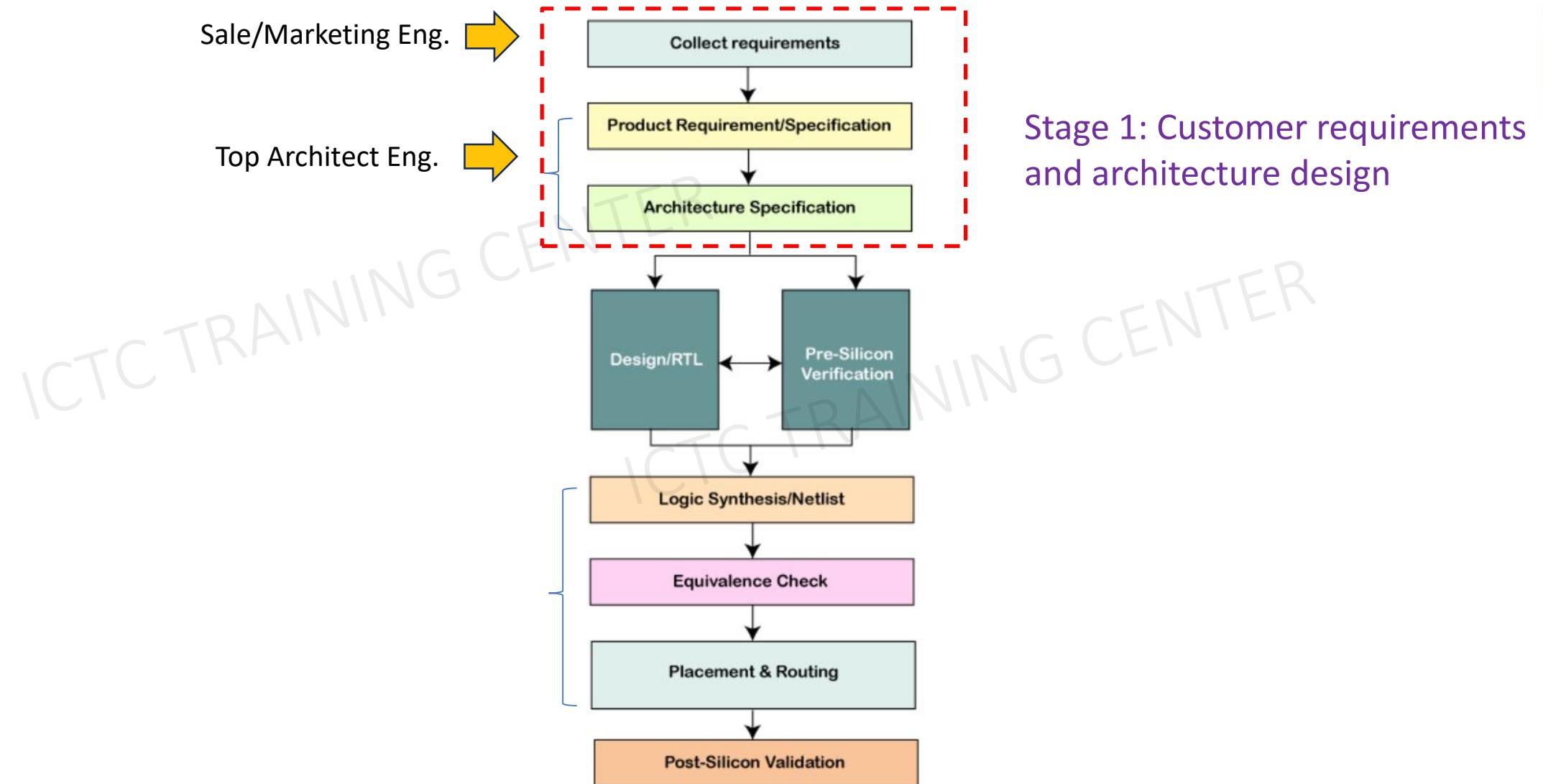
Stage 2: ASIC Design

Stage 3: Chip validation

ICTC TRAINING CENTER

ASIC Design Process

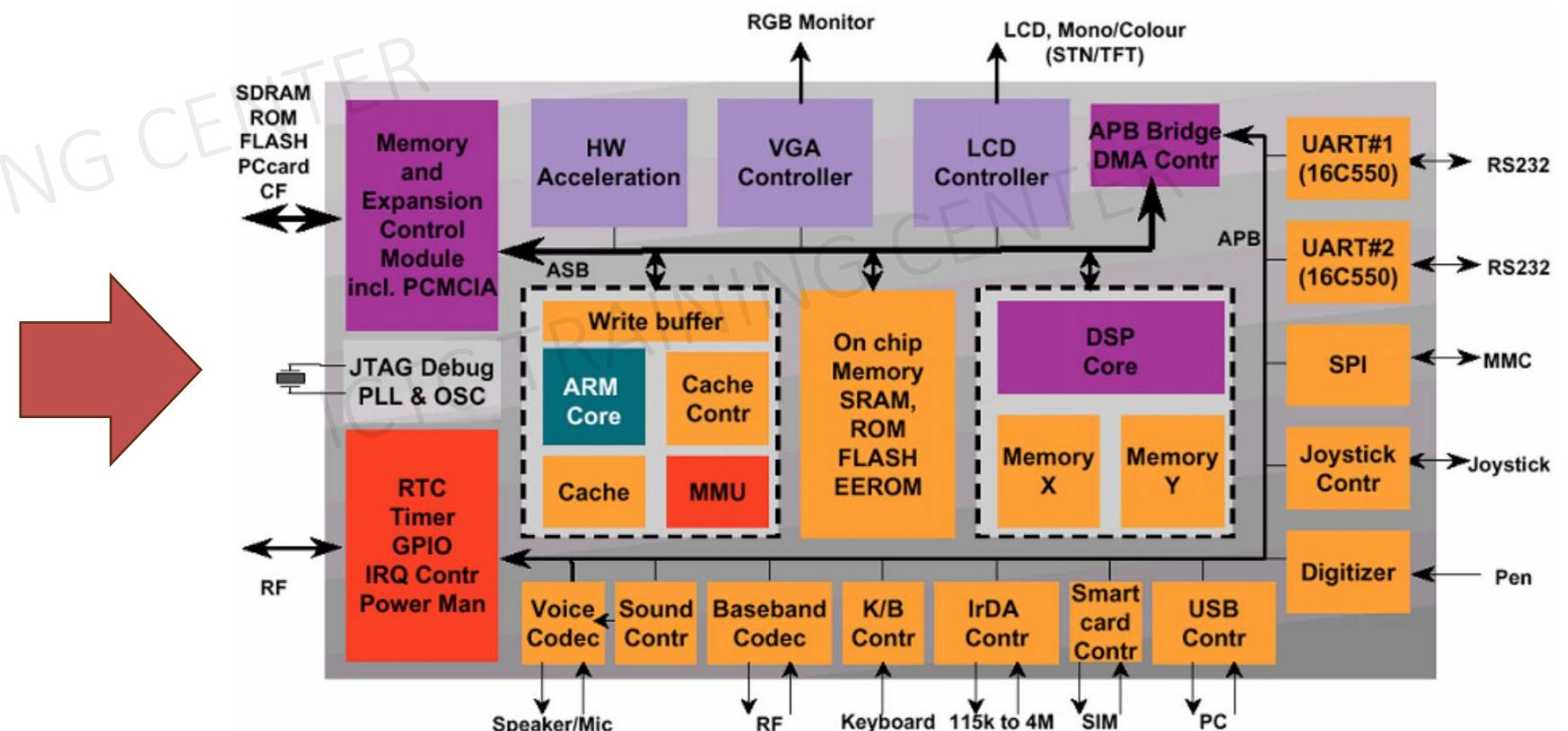
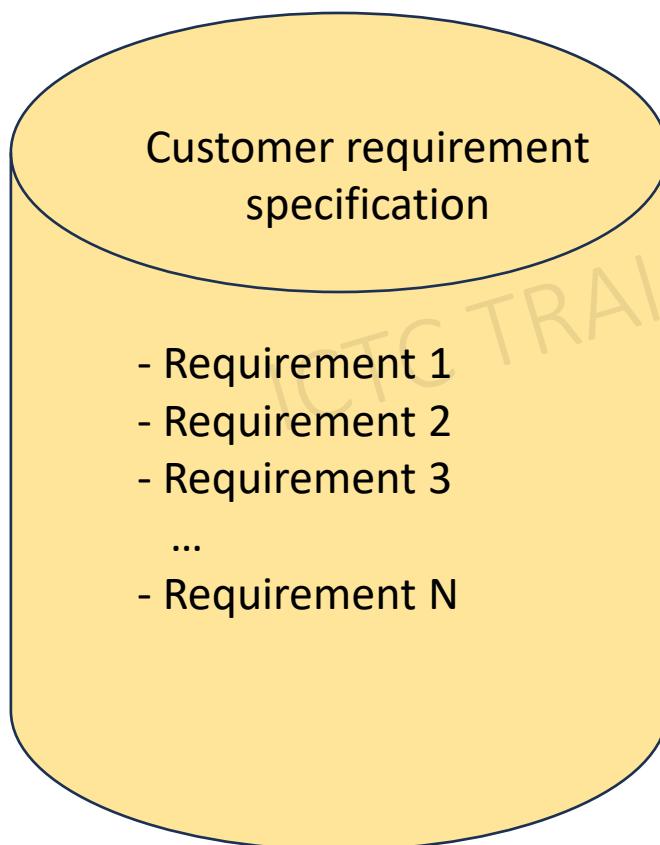
Stage 1: Customer Requirements and Architecture Design



ASIC Design Process

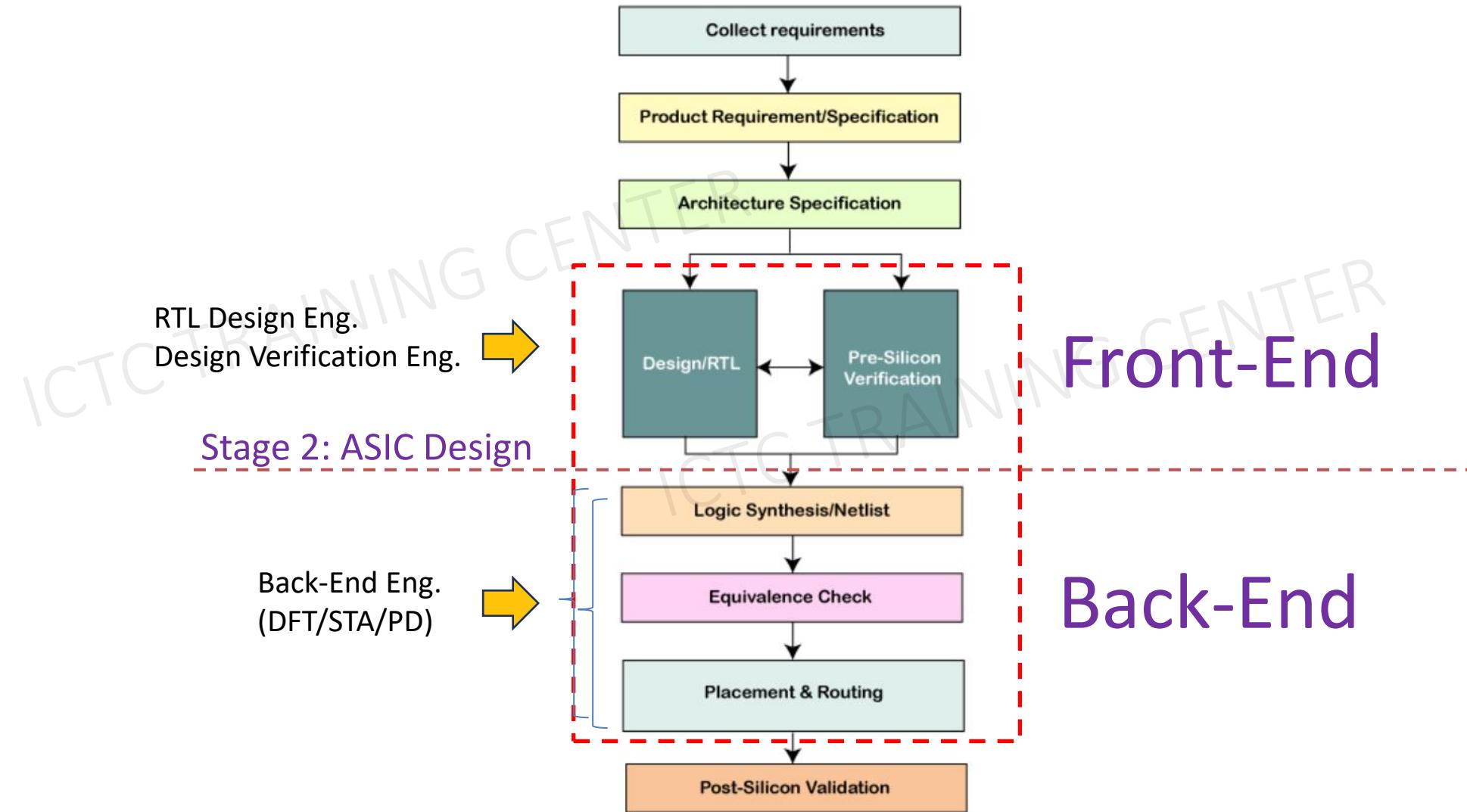
Stage 1

The requirements and applications are analyzed to form the chip system architecture



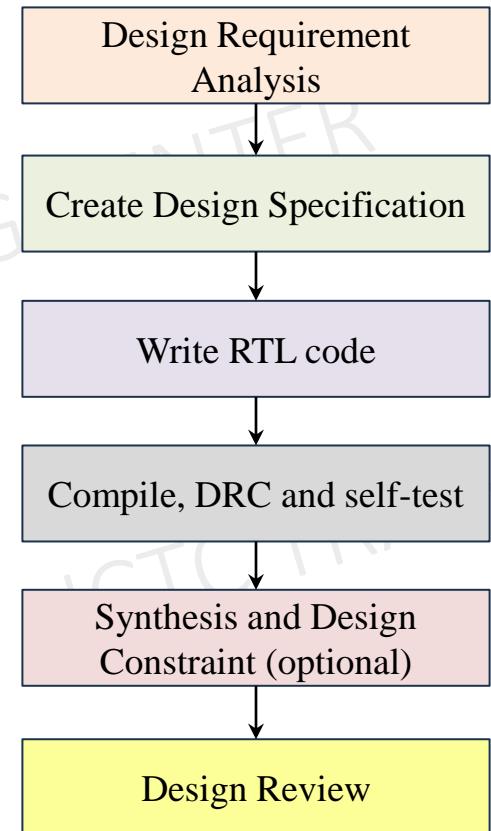
ASIC Design Process

Stage 2



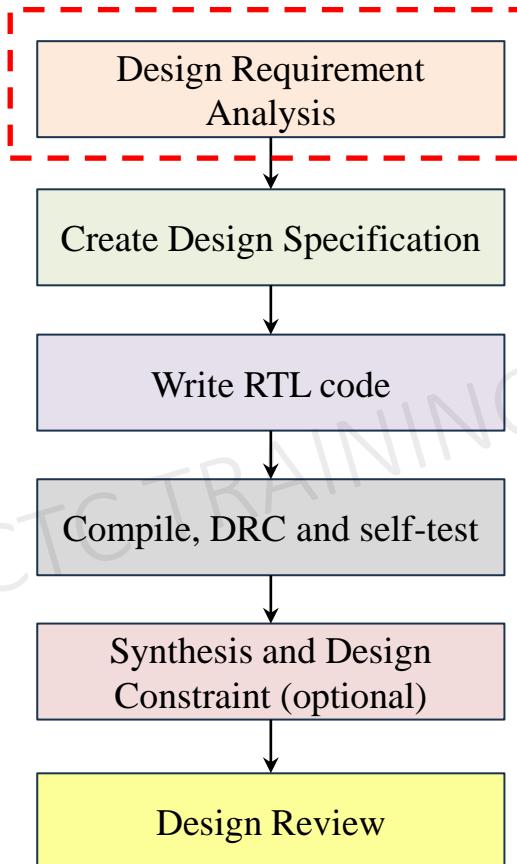
ASIC Design Process

RTL DESIGN PROCESS



ASIC Design Process

RTL DESIGN PROCESS – Design Requirement Analysis

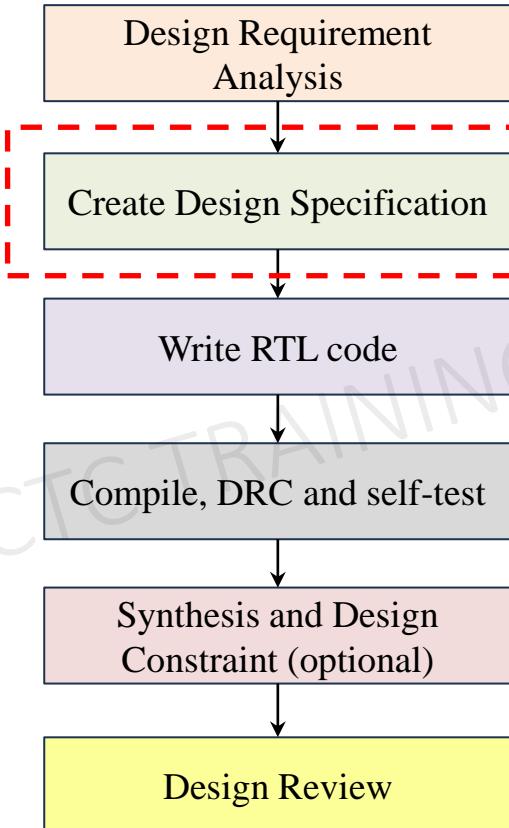


RTL design engineers' responsibilities are to analyze, estimate, evaluate the feasibility of the design requirements, and feedback to product team to adjust or rework the requirements with customers.



ASIC Design Process

RTL DESIGN PROCESS – Design Specification

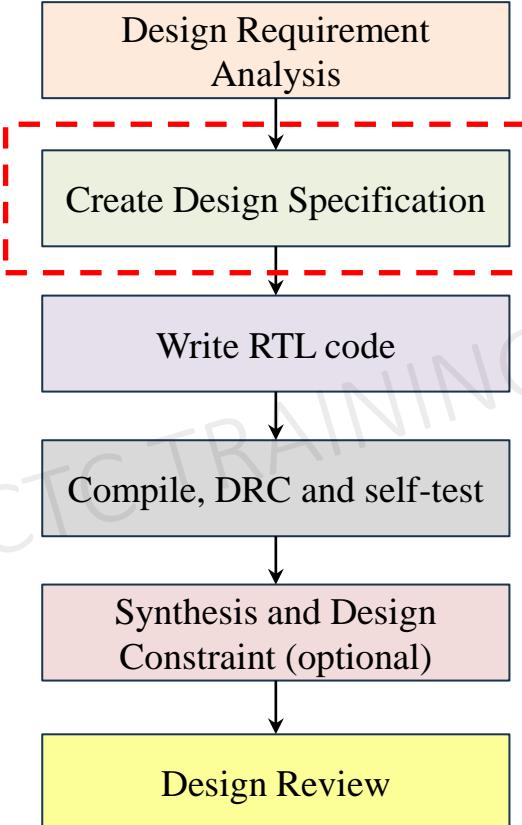


Design specification describes the desired operation, expressing the design ideas of RTL design engineers. This is a **mandatory** step in the RTL design process and must be performed before starting the RTL code writing process. The purpose of this document is:

- To clarify design ideas
- To avoid errors when writing RTL code
- To enhance error detection capabilities during design verification
- To enhance error detection capabilities during design review process

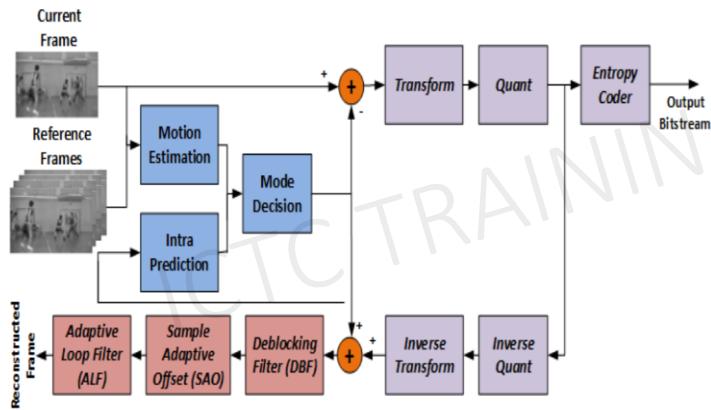
ASIC Design Process

RTL DESIGN PROCESS – Design Specification



Some basic part of a design specification

- Part 1: describe design's feature.
- Part 2: block diagram

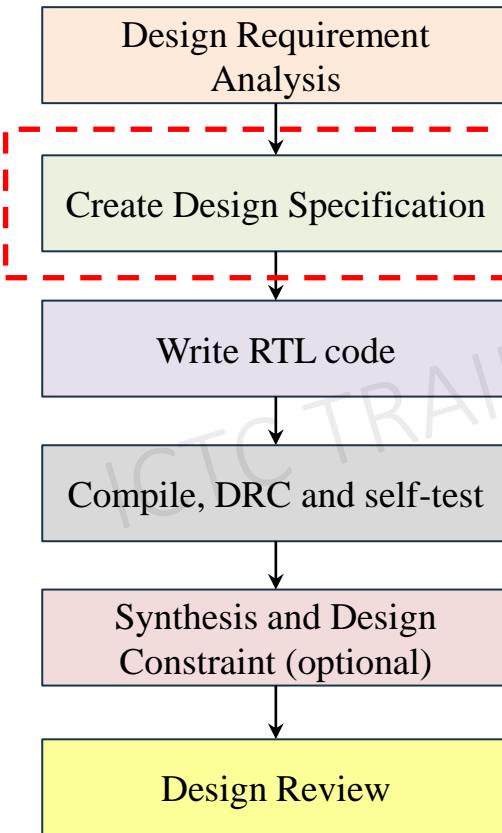


- Part 3: IO interface

From/to	Signal	I/O	Driving clock	Description
AHB master interface				
BUS	hbusreq	O	hclk	Bus request from master Not used
BUS	hgrant	I	hclk	Hgrant for the master Fixed 1'b1
BUS	haddr[31:0]	O	hclk	AHB address from to BUS slave

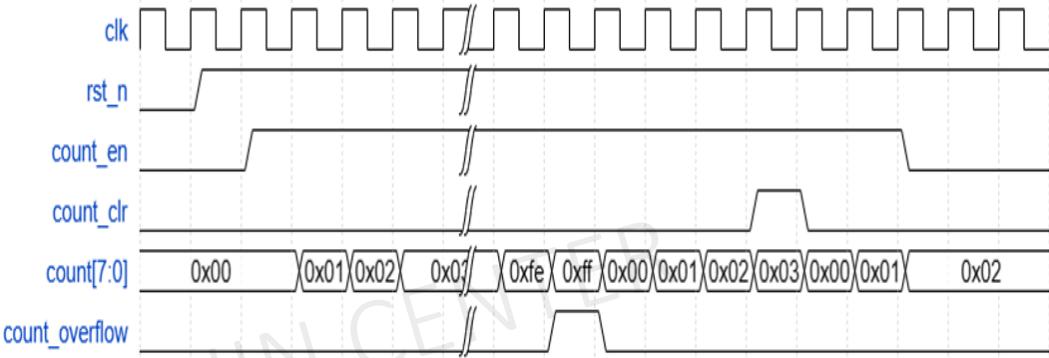
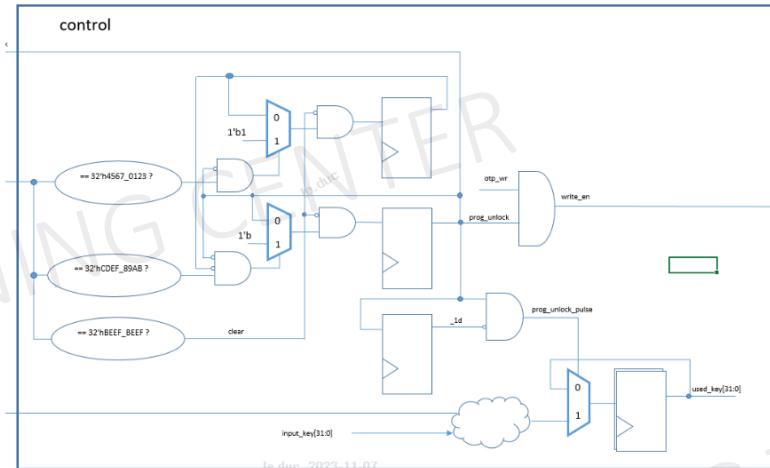
ASIC Design Process

RTL DESIGN PROCESS – Design Specification



Some basic part of a design specification (cont.)

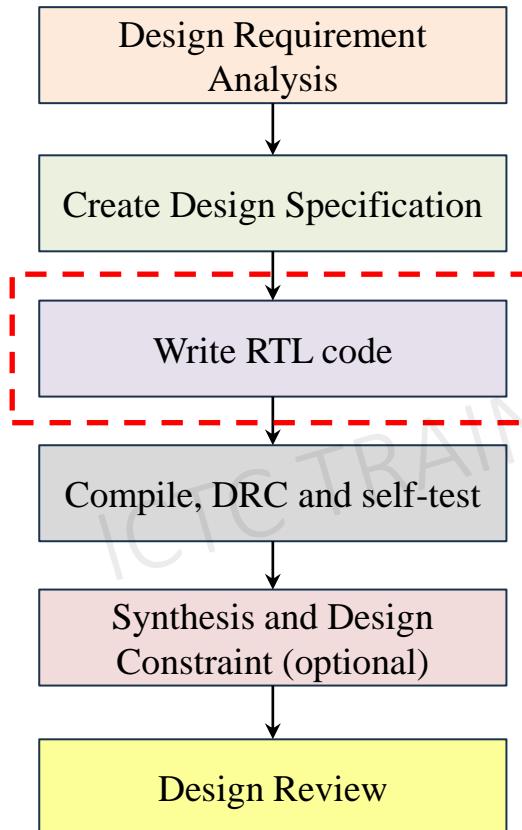
- **Part 4:** detail design description



- **Part 5:** Other requirement. This part lists requirements for other departments such as DV (Design Verification), DFT (Design For Test), or PD (Physical Design) if there are specific requirements regarding constraints, timing, delay chain, etc.

ASIC Design Process

RTL DESIGN PROCESS – Write RTL Code

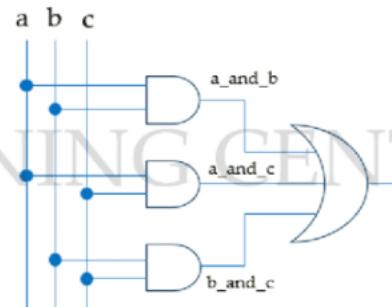


RTL (Register Transfer Level) is an abstraction level definition aims for describing hardware behavior by a HDL (hardware description language). There are 3 most popular HDLs:

- Verilog
- System Verilog
- VHDL

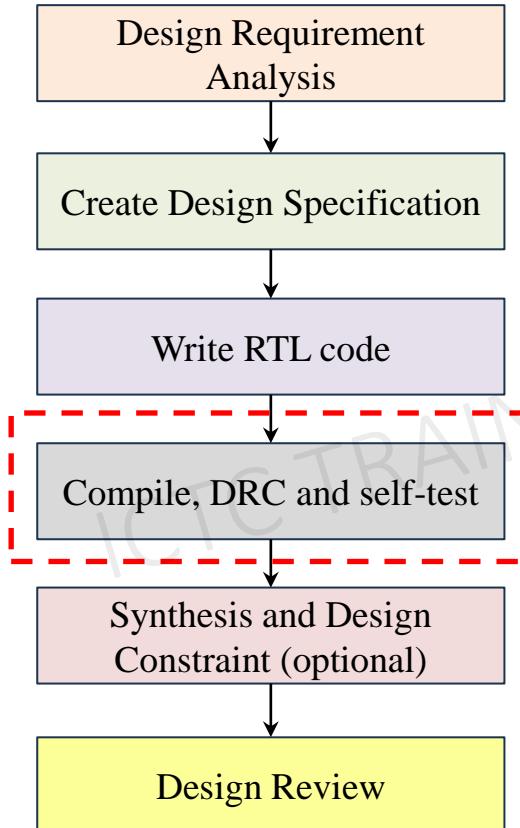
RTL code need to follow strict rules and constraints in order to be synthesized into a logic gate. Otherwise, it will be non-synthesizable and cannot be translated into gate netlist.

```
assign a_and_b = a & b;
assign a_and_c = a & c;
assign b_and_c = b & c;
assign z = a_and_b | a_and_c | b_and_c;
```



ASIC Design Process

RTL DESIGN PROCESS – Compile, DRC and Self-Test



Compile: To check for syntax errors, connection errors (when connecting wires between modules), and missing variable declarations...

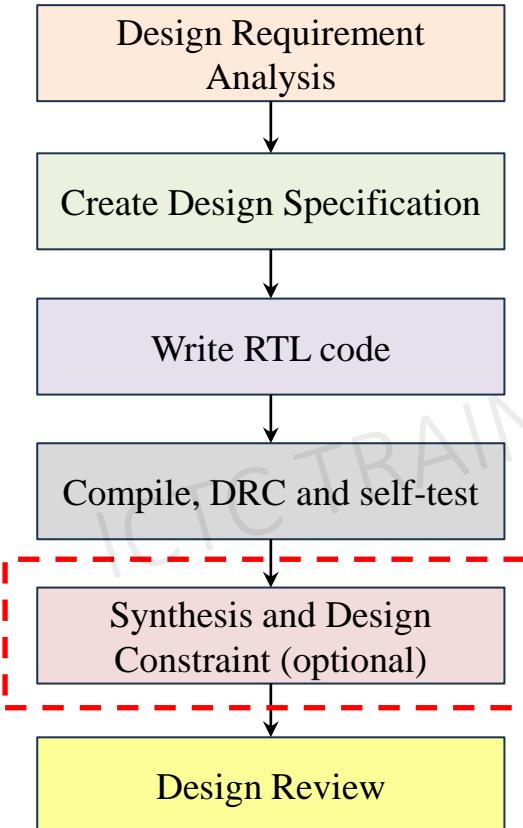
Design Rule Check (DRC): to detect additional errors related to non-compliance with design rules that may lead to unsynthesizable logic, or errors related to asynchronous clock domain crossings (CDC), or issues that may not be compatible with subsequent steps such as Design For Test (DFT) or Static Timing Analysis (STA).

Self-test: Design engineers also need to verify their designs to ensure that basic functionalities operate correctly before handing them over to the verification engineers for more in-depth testing.



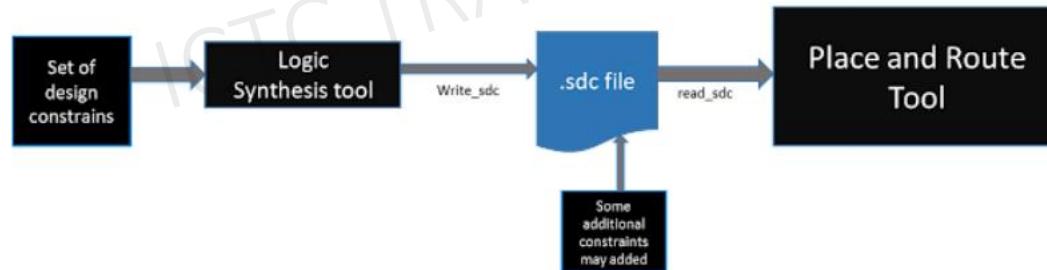
ASIC Design Process

RTL DESIGN PROCESS – Compile, DRC and Self-Test



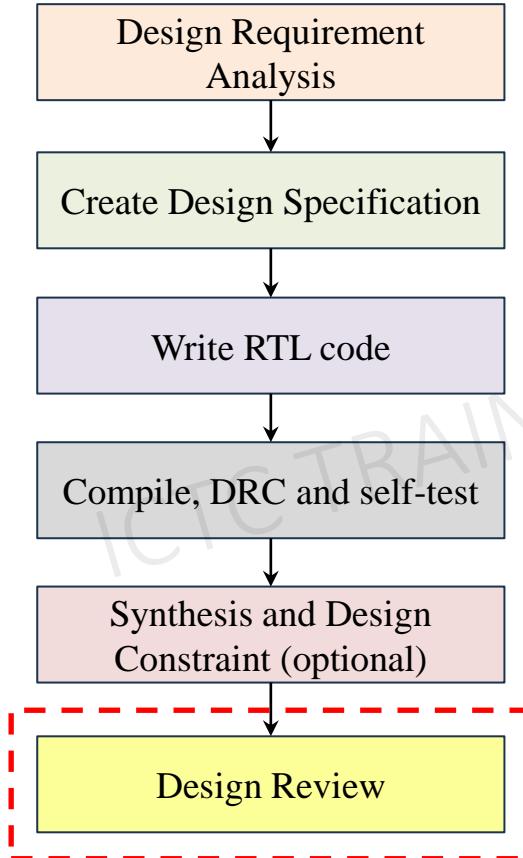
Synthesis: to check if there are any potential issues when run synthesis at later phase.

Design Constraint: specific requirement to BE about timing requirements, clock pulses, input and output delays, and other crucial parameters to achieve desired performance and functionality.



ASIC Design Process

RTL DESIGN PROCESS – Compile, DRC and Self-Test



Design Review is a process to review all the design steps are correct.

- Review the design specification match with project requirement.
- Review RTL code to match with design specification.
- Review DRC confirmation.
- Review design constraint is correct or not.



ASIC Design Process

RTL DESIGN PROCESS – QUESTIONS



Question 1: What is RTL, HDL stands for ?

Question 2: What's the difference between RTL and Verilog ?

Question 3: Draw waveform and write Verilog code, what should be done first in RTL design step ?

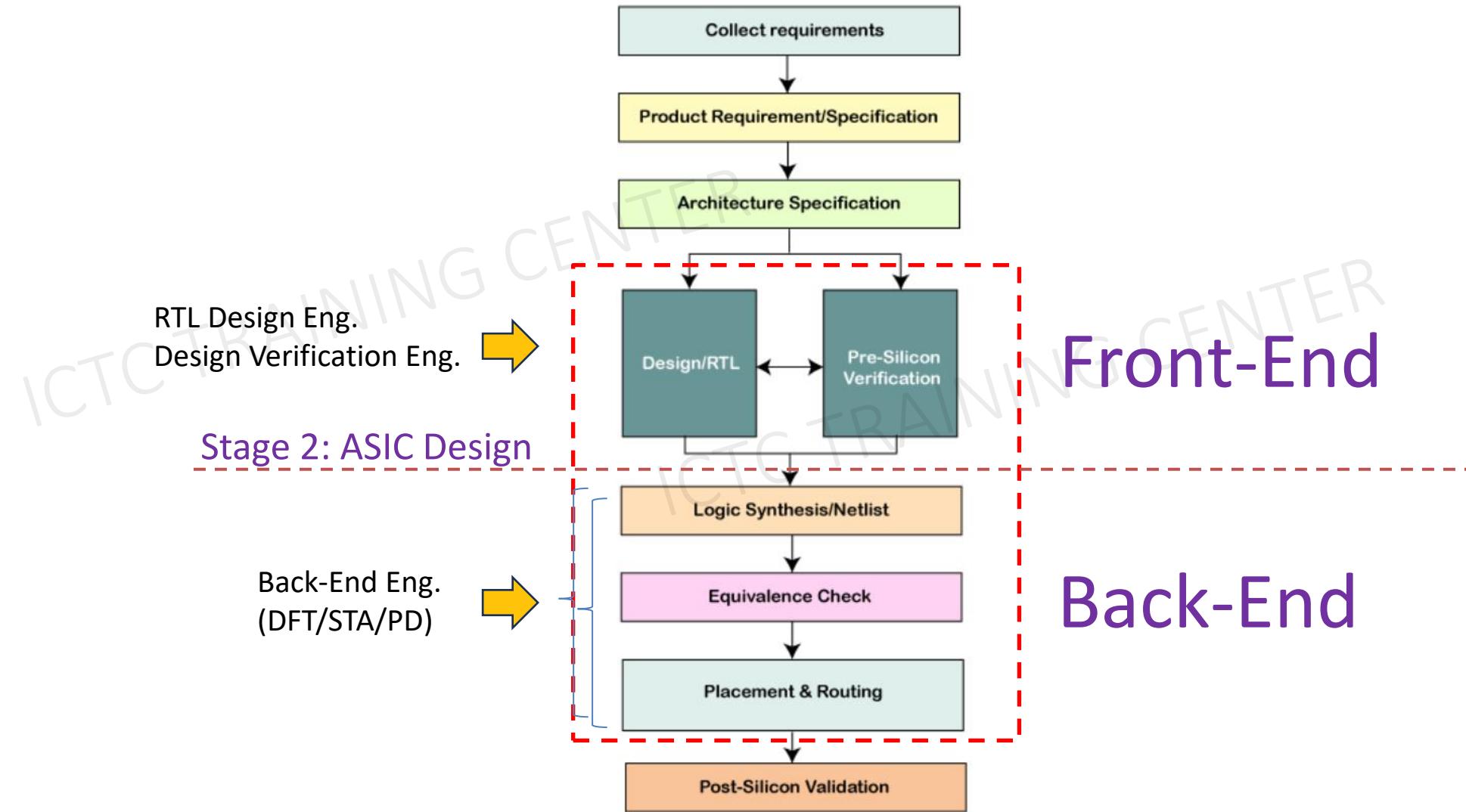
Question 4: How many percent of time you think that we need to spend for RTL coding in the whole RTL Design step ?

Question 5: How RTL designer can do to improve quality of RTL code ?



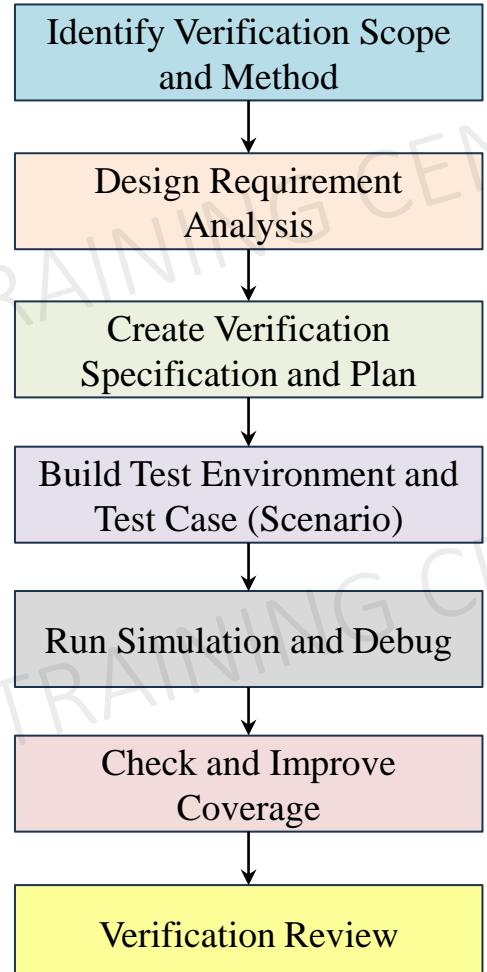
ASIC Design Process

Stage 2



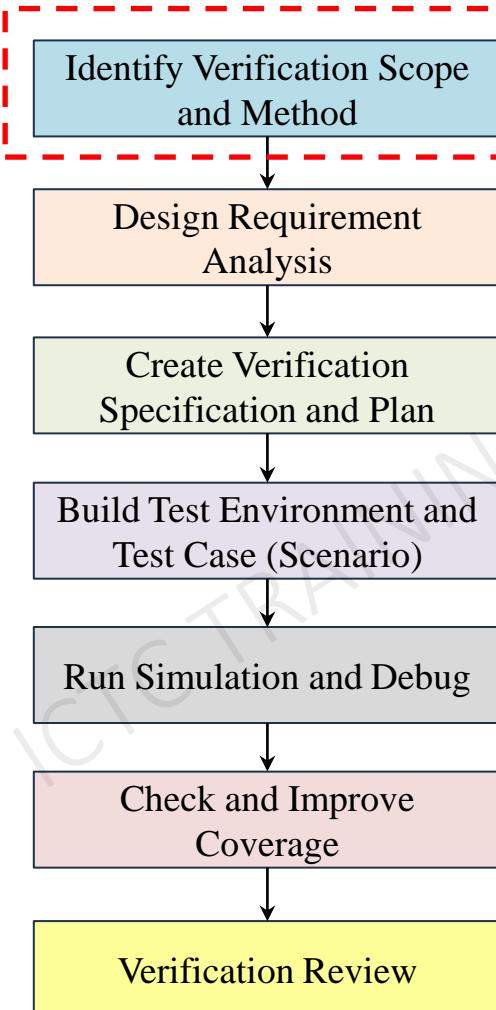
ASIC Design Process

DESIGN VERIFICATION PROCESS (DV)



ASIC Design Process

DV – Identify Verification Scope and Method



Verification scope: Need to determine testing at IP level or SoC level

- At the IP level, also known as UT (Unit Test) or stand-alone test, the IP will operate independently or within a small system on chip
- At the chip level, also known as ST (System Test) or CT (Combine Test), IPs are integrated into a complete system with full connectivity to other IPs.

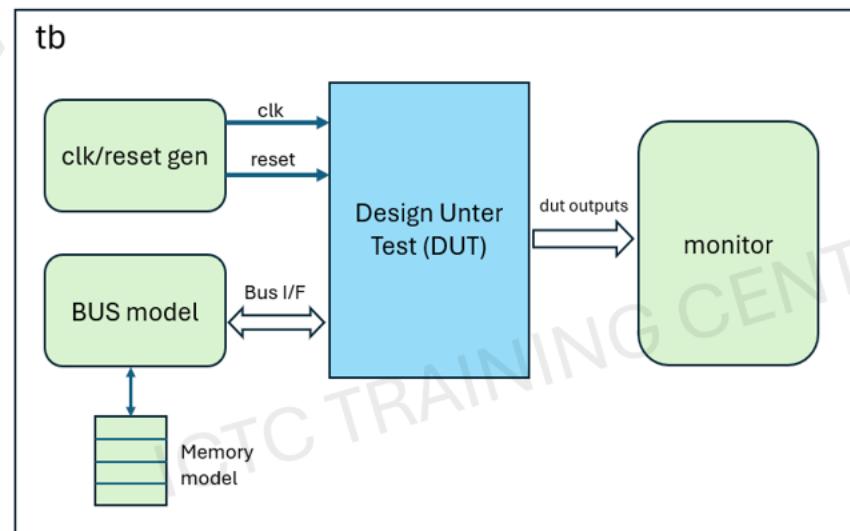


Figure: Unit Test Environment

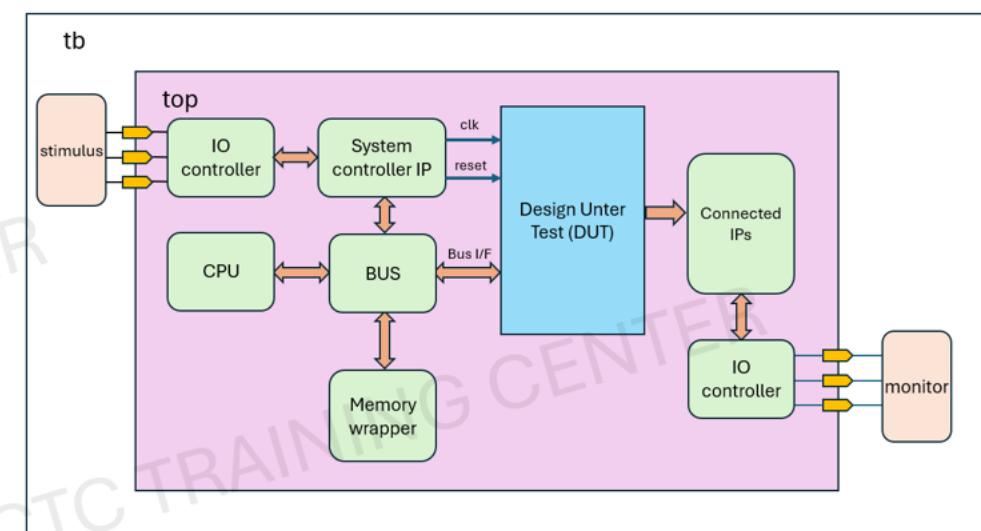
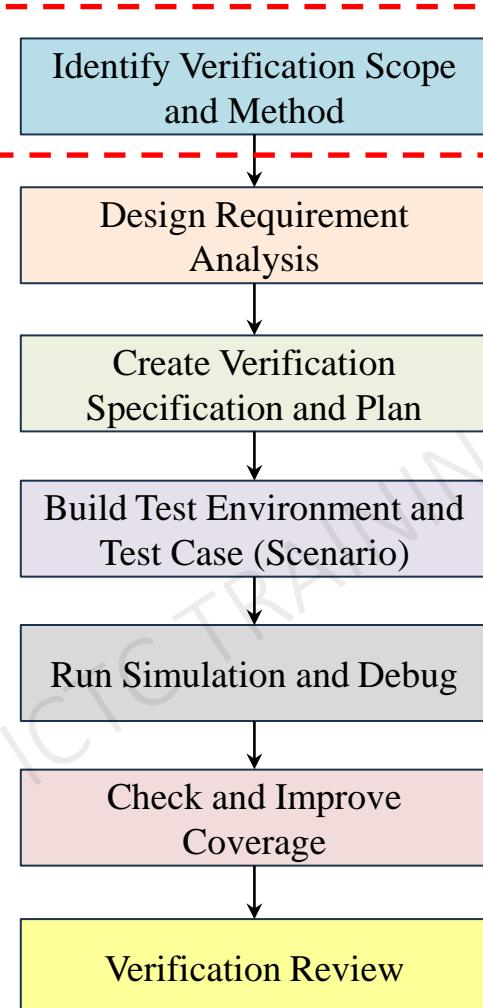


Figure: System Test Environment

ASIC Design Process

DV – Identify Verification Scope and Method



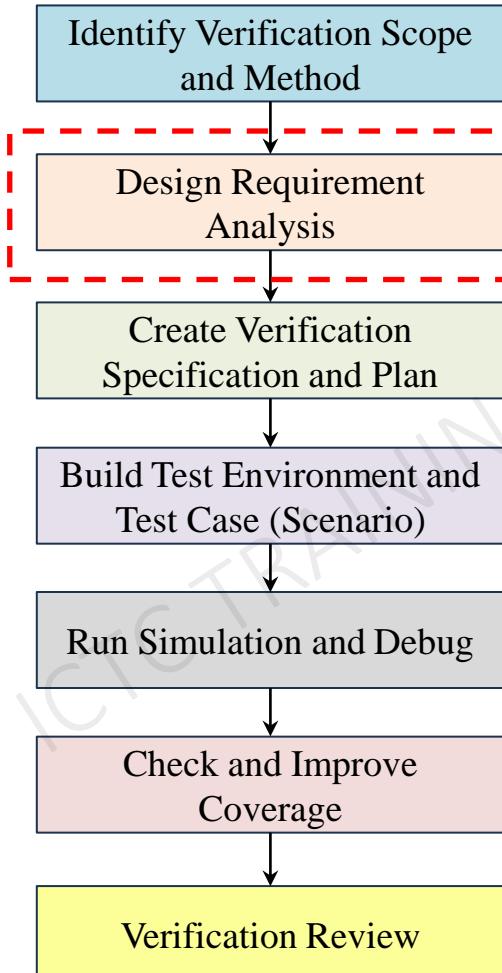
Verification method: depending on the scope, object of testing, or project requirements, the verification engineer will choose the appropriate testing method. Common testing methods:

- **Direct test:** a method where individual components or functionalities within a digital system, such as a System-on-Chip (SoC), are tested directly against specific criteria or requirements.
- **Random test:** a method that generate test cases or stimuli using random or pseudo-random methods, to uncover potential bugs or corner cases that might not be revealed by traditional directed testing alone.
- **Formal test:** a method using mathematical techniques to prove or verify the correctness of a digital design.



ASIC Design Process

DV – Design Requirement Analysis



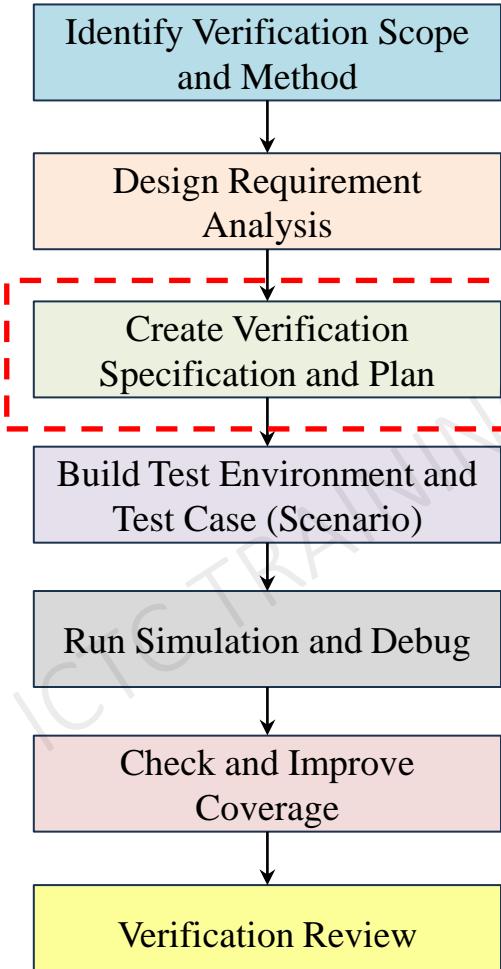
Design Requirement Analysis: DV engineers not only need to have a detailed understanding of the design but also comprehend the roles and functions of the design within the system. DV engineers often need to review the following documents.

- **Design Specification:** This documentation will help DV engineers understand how RTL engineers construct designs, thereby enabling them to envision the points that need to be verified.
- **SOC System Specification:** This document is used to describe the functionality of all functional blocks within the SoC. This documentation will help DV engineers understand the position of IPs within the SoC, as well as how to configure related functional blocks.
- **Customer Requirement Specification:** The requirements from the customer may be misunderstood or incomplete by RTL Design engineers. Therefore, DV engineers must also refer to and verify whether the current design adequately fulfills the customer's requirements.



ASIC Design Process

DV – Create Verification Specification and Plan



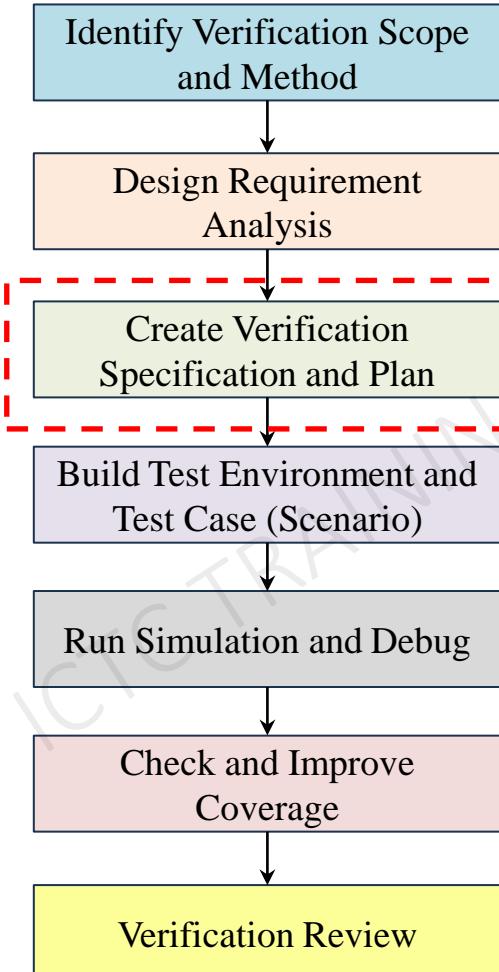
Verification Specification: Written by DV engineers to express the testing ideas, a test document typically includes the following contents:

- Basic goals and functions to be tested.
- Testing scope: whether it's at the IP level or system level.
- Testing methods: description of the methods used for testing (direct test, random test, formal test).
- Test environment specification: block diagram, description of functional blocks in the environment, interconnections between blocks, etc.



ASIC Design Process

DV – Create Verification Specification and Plan



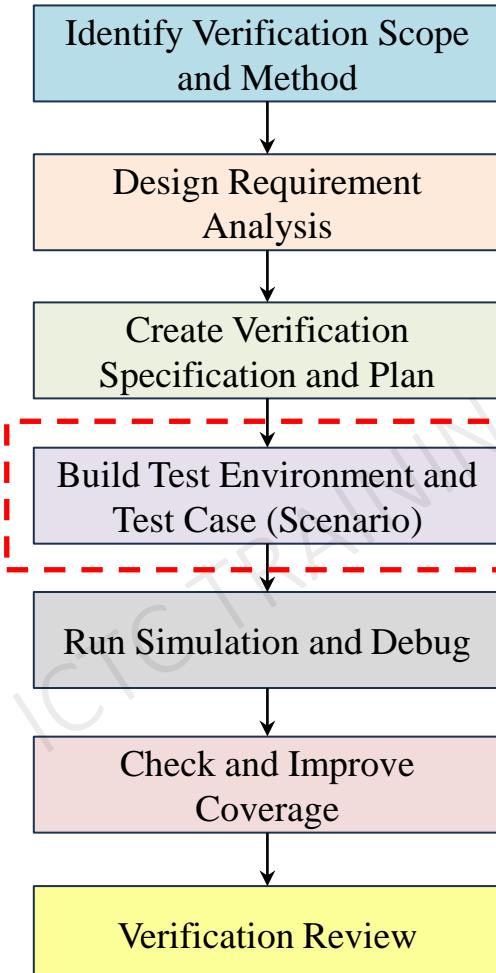
Verification Plan: A detailed document describing the plan and methods for verifying that the SoC meets functional, performance, and reliability requirements.

- **Verification item list:** list of items that needed to test
- **Verification method for each item:** Depending on each item, DV engineers will choose the appropriate testing method.
- **Test description:** This section will clearly demonstrate how to test the item, including what inputs need to be provided, how related IPs should be configured, how to observe the outputs, and what criteria need to be met to fulfill the requirements.
- **Classification:** Depending on the importance of each item, engineers will plan which items to test first and which ones can be tested later.
- **Test plan:** Each item will be specifically planned for testing. It will help to keep track the progress to ensure the project schedule.

ID	Item	Sub item 1	Sub item 2	Method	Class	Test sequence	Formal assertion	Pass condition	Plan Start	Plan End	Actual Start	Actual End	PIC
1	Address map	Address is 4KB	-	Direct	A	Toggle each bit of 12 bit address - R/W at address offset 0 - R/W at address offset 1 - R/W at address offset 2 - R/W at address offset 0x800	-	RW is OK Psel always toggle	2-Feb	3-Feb			
2		Boundary address	Address inside address map 0x0000_1000 - 0x0000_1FFF	Direct	A	Write/Read to 0x0000_1000 Then Write/Read to 0x0000_1FFF	-	RW is OK Psel is asserted	4-Feb	5-Feb			
3			Address outside address map 0x0000_1000 - 0x0000_1FFF	Direct	A	Write/Read to 0x0000_0FFF Write/Read to 0x0000_2000	-	RW is not hang-up Psel is not asserted	4-Feb	5-Feb			
		Reserved address	-	Formal	B	-	Use jasper gold CSR to check all the reserved addresses	Jasper gold reserved assertion is PASSED	2-Feb	10-Feb			

ASIC Design Process

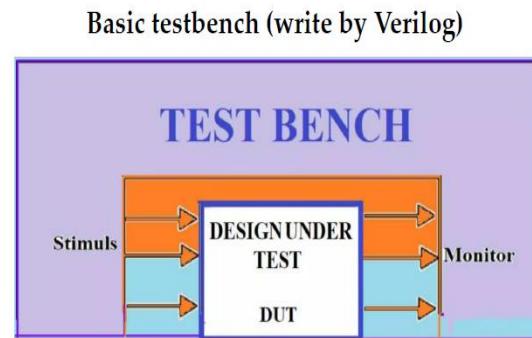
DV – Build Test Environment and Test Case



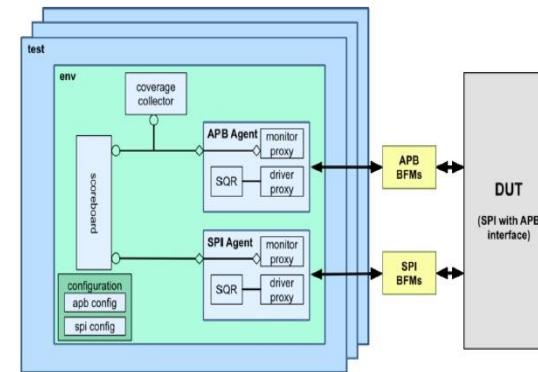
Test environment: Based on the verification specification, DV engineers will start building the verification environment. The verification environment plays a crucial role in ensuring that the IP/SoC meets the specifications and design requirements.

Below are the main components of a verification environment

- **Testbench:** The most critical component of the environment, the testbench is used to provide inputs, monitor behavior, and verify the correctness of outputs. It can be written in hardware description languages like Verilog, SystemVerilog, or VHDL.



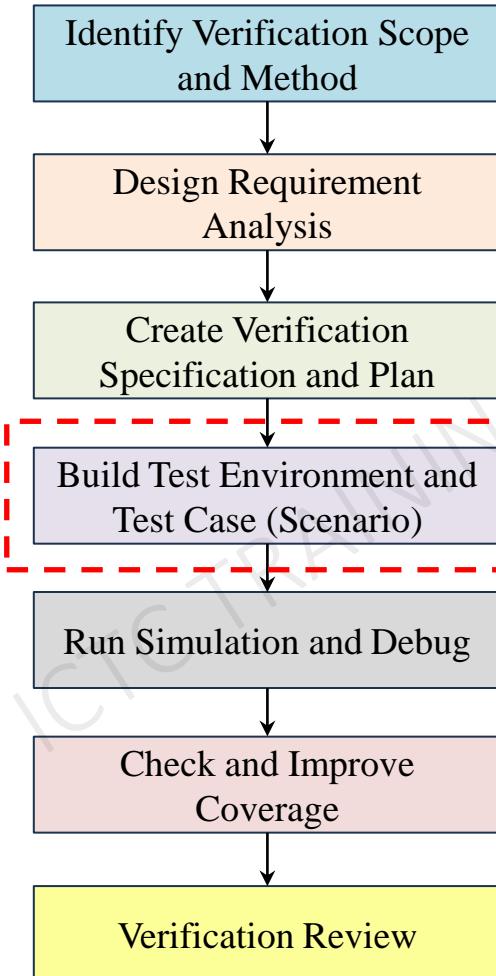
Basic testbench (write by Verilog)



- **Verification IP (VIP):** Verification IP (VIP) are pre-designed, reusable components that serve as models for standard protocols or interfaces commonly used in SOC designs such as PCIe, USB, Ethernet, DDR, AMBA, and more.

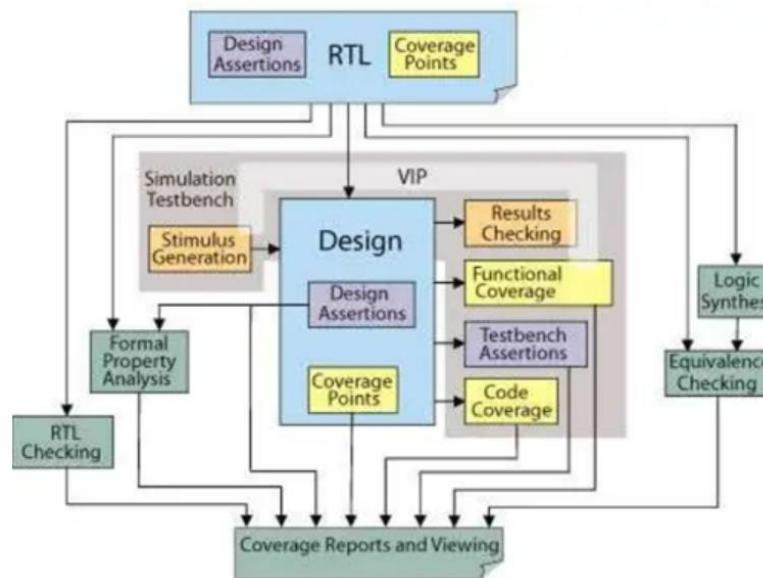
ASIC Design Process

DV – Build Test Environment and Test Case



Test environment (cont.):

- **Coverage:** are used to track the completeness of the verification process. These coverage models are integrated into the verification environment during its construction, allowing engineers to monitor and analyze the effectiveness of their test cases and ensure that critical parts of the design have been adequately verified.

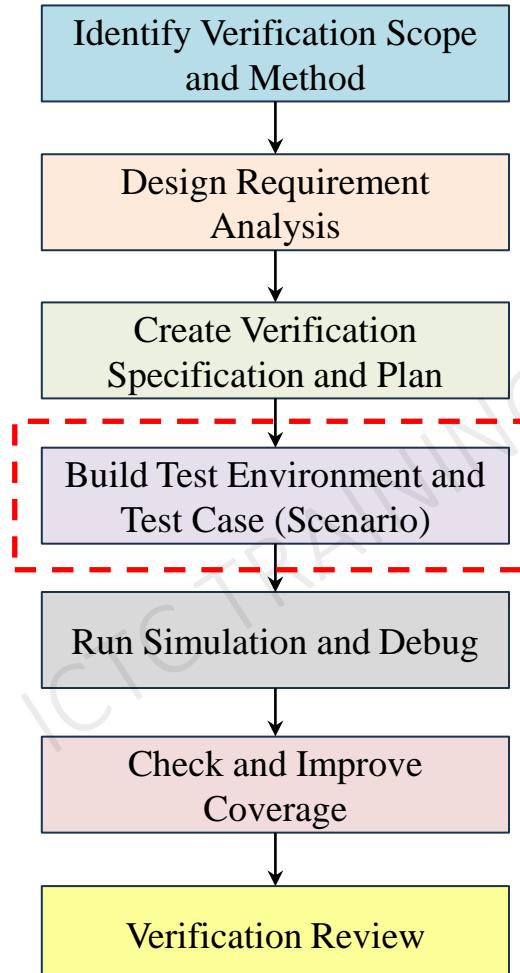


- **Simulation script:** to carry out various steps including compiling the design, testbench, testcase, providing preload files into memory, constructing directory trees, support dumping waveform, coverage database generation and simulation report analyzing.



ASIC Design Process

DV – Build Test Environment and Test Case



Test case (test scenario): Based on the description of the verification steps in the verification plan, DV engineers will begin writing test cases. This is the most intellectually challenging task for DV engineers. Test cases consist of the following main parts:

- Configuration of IPs and related IPs.
- Generating stimuli for the DUT.
- Checkers to verify the outputs.

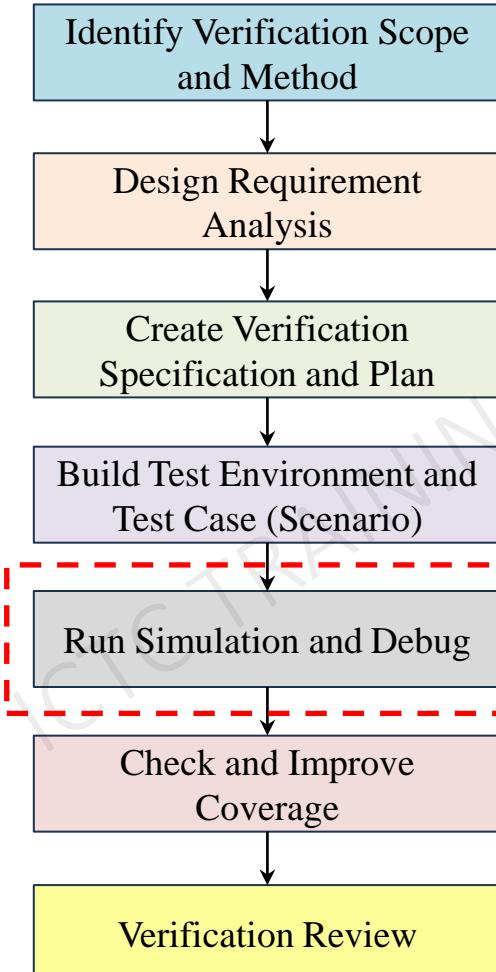
A good test case should have the following factors

- Clear and detailed
- Simple and specific
- Easy to reuse
- Cover all possible cases of the design
- Enough appropriate checker



ASIC Design Process

DV – Run Simulation and Debug



Compilation: before running simulation, DV engineers need to compile the whole environment to check syntax and connections, include:

- Compile the design under test (DUT)
- Compile the entire environment: testbench, model, VIP, interface ...
- Compile test cases

During the simulation process, if errors occur, DV engineers must debug them:

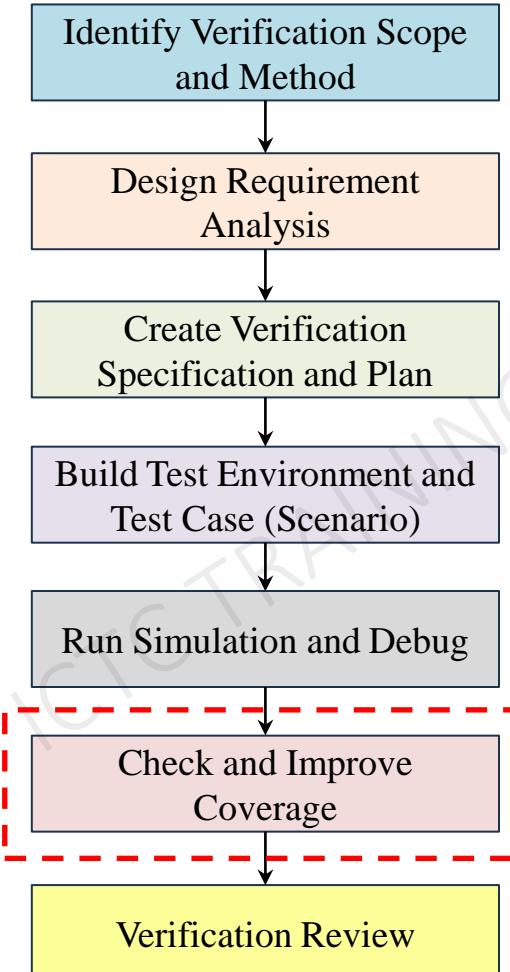
- Issues related to environment, testcase: DV must fix
- Issues related to DUT: feedback to designer to have deeper analysis or fixing.

The debugging process on chip level can sometimes be complex and requires experience and a certain level of understanding of the system. Therefore, DV engineers need to have wide knowledge of SoC Specification to effectively troubleshoot issues.



ASIC Design Process

DV – Check and Improve Coverage



Coverage: helps DV engineers identify areas they haven't yet tested. Here are some examples:

- Some lines of code that haven't been tested
- An else-if condition that hasn't been tested
- A logical combination of values that hasn't been tested
- Some states of a state machine that haven't been tested
-

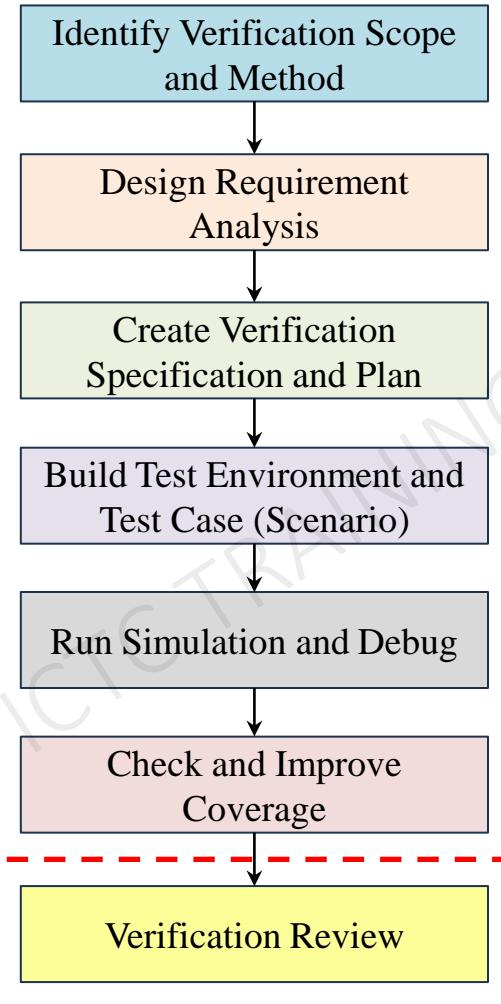
Coverage also helps RTL design engineers identify unreasonable lines of code and conditions that never occur. As a result, RTL engineers can optimize their designs to make them better.

Completing coverage is a MUST to demonstrate that the verification work is completed.



ASIC Design Process

DV – Verification Review



Review: when all stages are completed, a review session will be organized to review all the previous processes.

- Review the verification specification and verification plan to identify any deficiencies or lacking check for corner cases.
- Review the environment and testbench.
- Review the test cases to ensure correct sequencing and completeness of checkers.
- Review log files and waveforms.



ASIC Design Process

DESIGN VERIFICATION PROCESS – QUESTIONS

Question 1: What are the inputs for DV engineers?

Question 2: DV engineers need to be better at programming language than RTL Design Engineer?

Question 3: What are the differences between UT & ST?

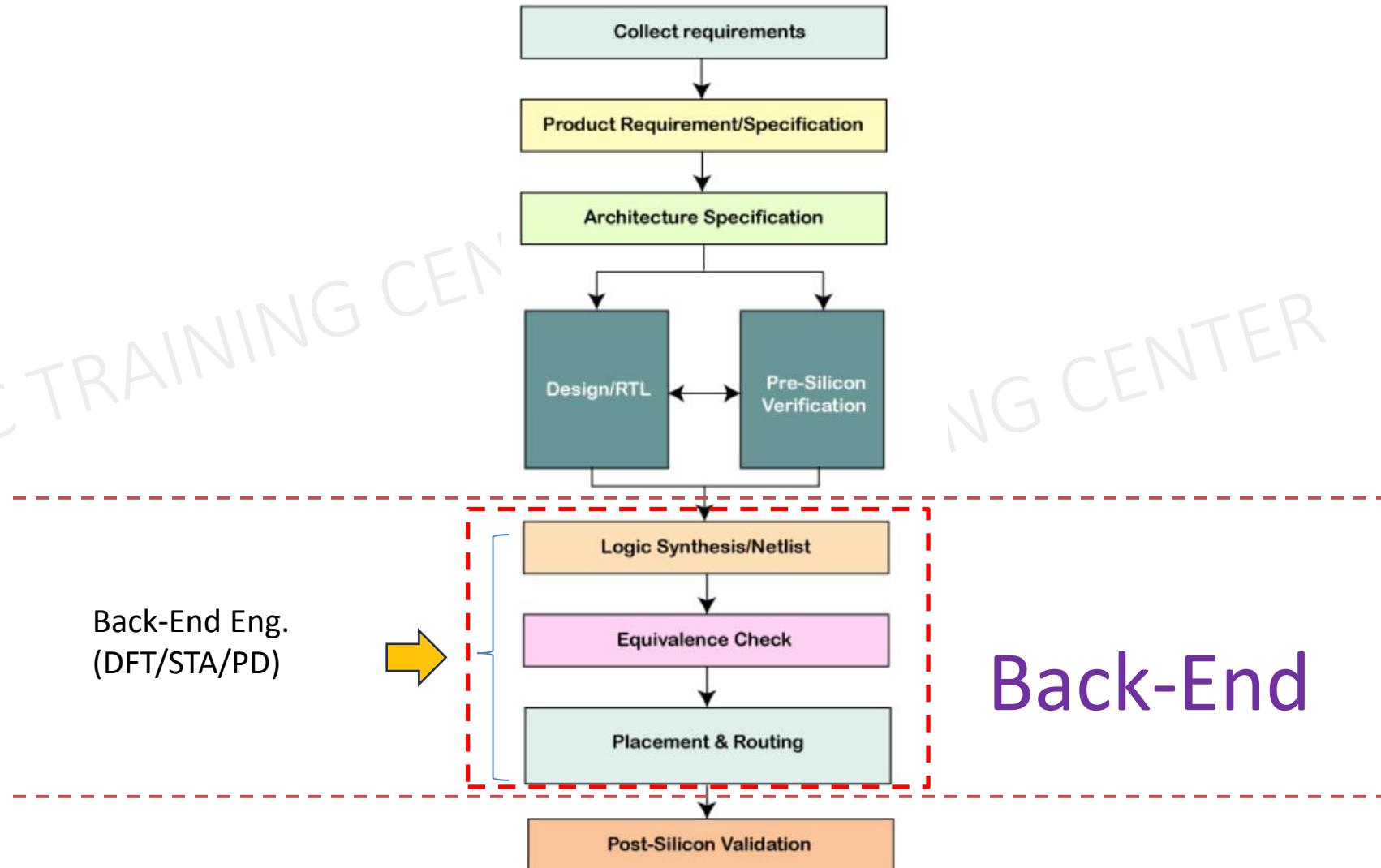
Question 4: How many verification methodologies are mentioned in the lecture?

Question 5: How to avoid missing bugs of the design?



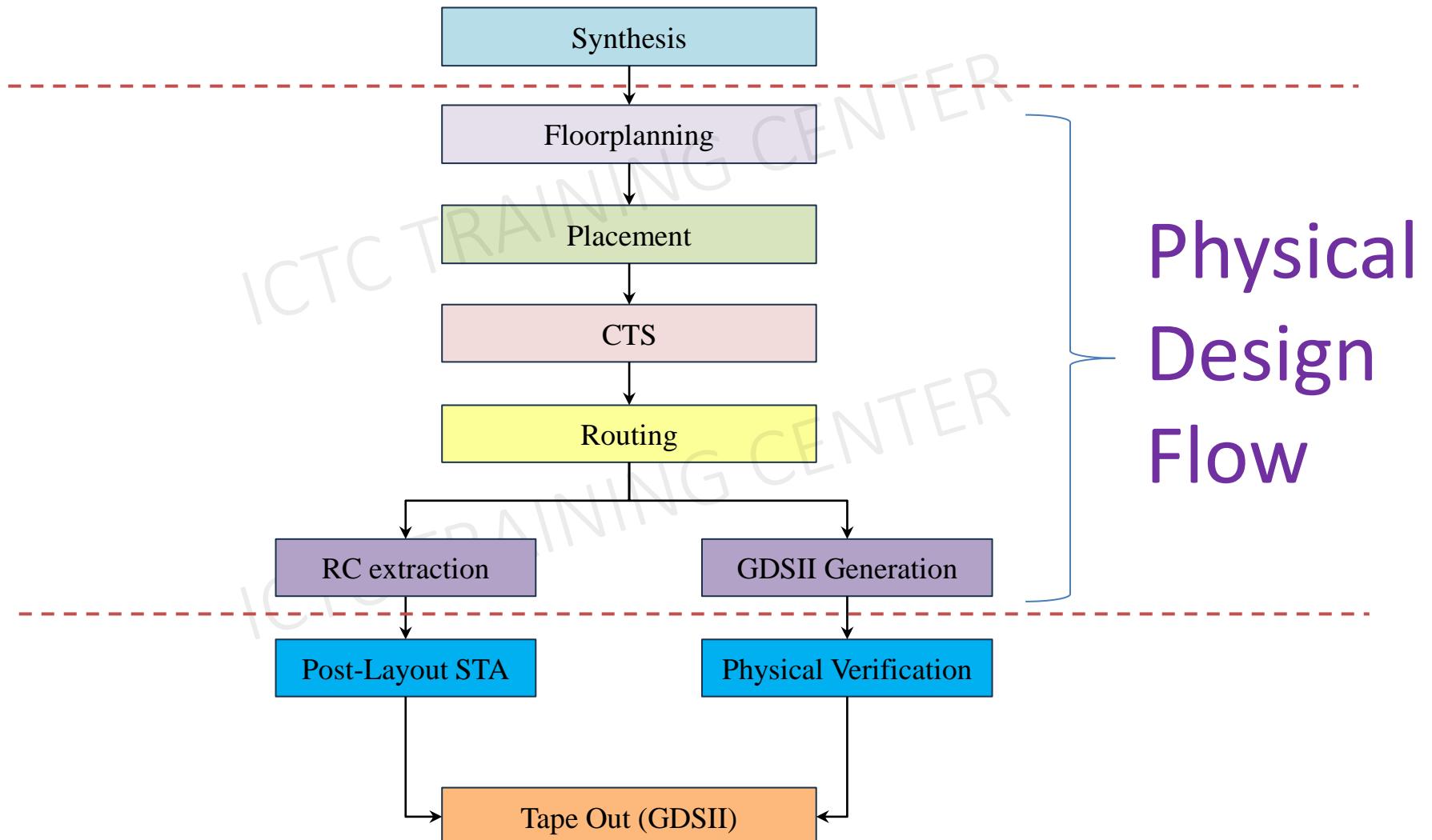
ASIC Design Process

BACK END



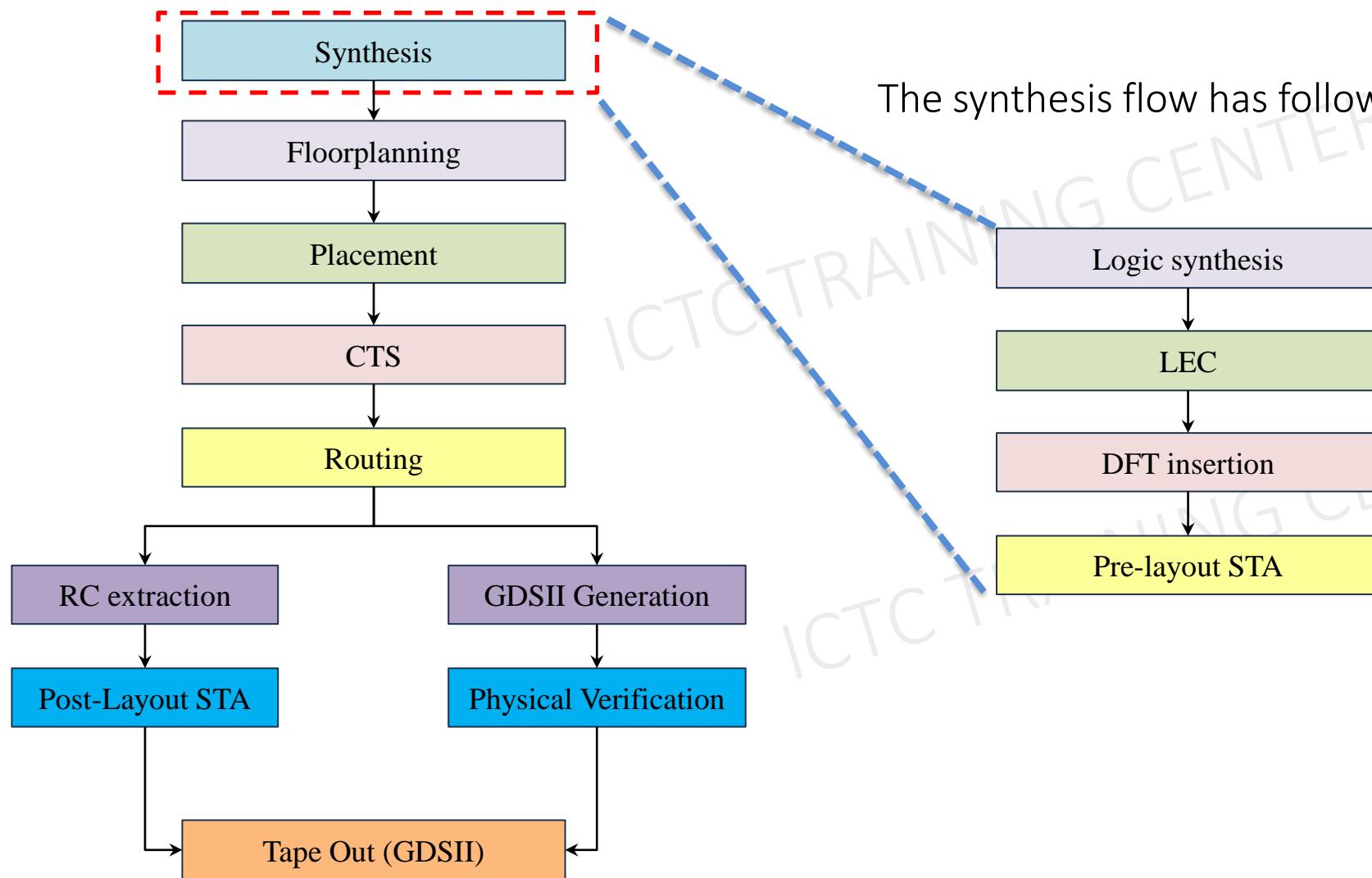
ASIC Design Process

BACK-END DESIGN PROCESS



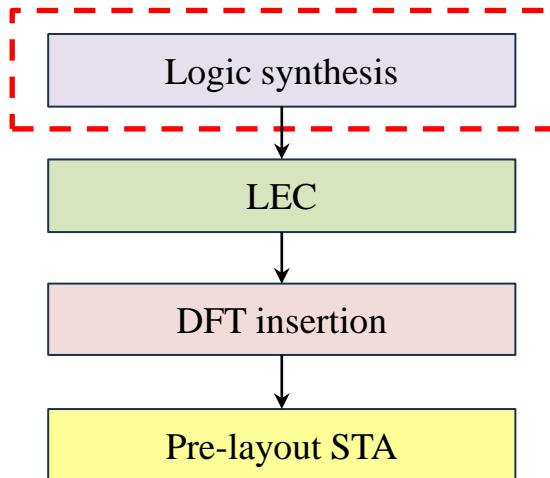
ASIC Design Process

BACK-END DESIGN PROCESS - SYNTHESIS

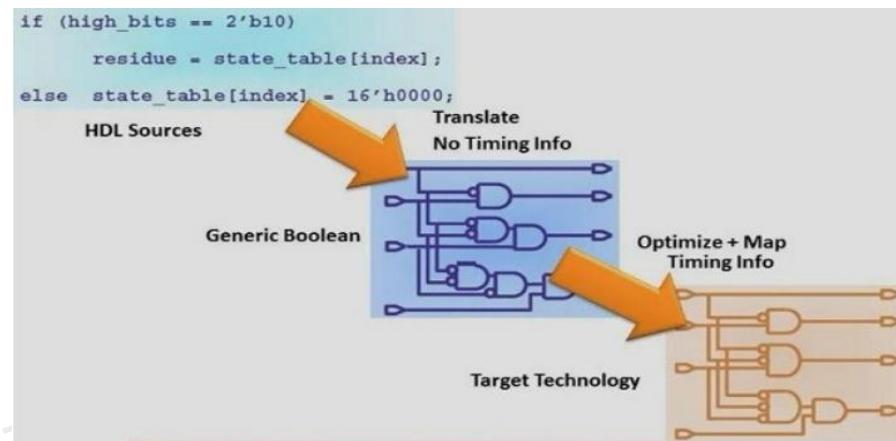


ASIC Design Process

BACK-END DESIGN PROCESS - SYNTHESIS



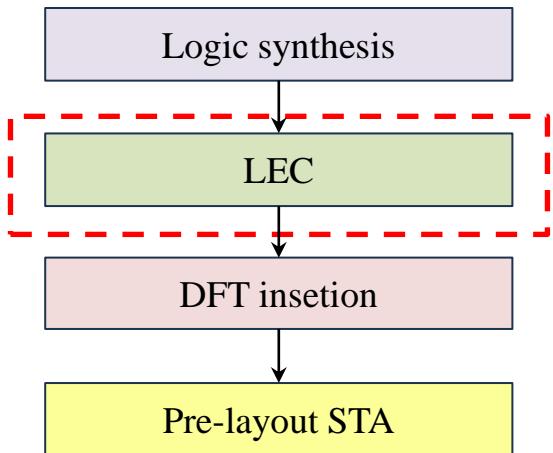
Logic synthesis: is the process of using Electronic Design Automation (EDA) tools to translate RTL code into a gate-level netlist, which serves as input for the Back-End design stages. If the RTL code does not conform to predefined rules, it will be non-synthesizable and cannot perform synthesis.



This timing info (sdf file – standard delay format) is sent back to DV to do the gate simulation for timing check.
This step is done either by BE engineers or FE product engineers based on each company design process.

ASIC Design Process

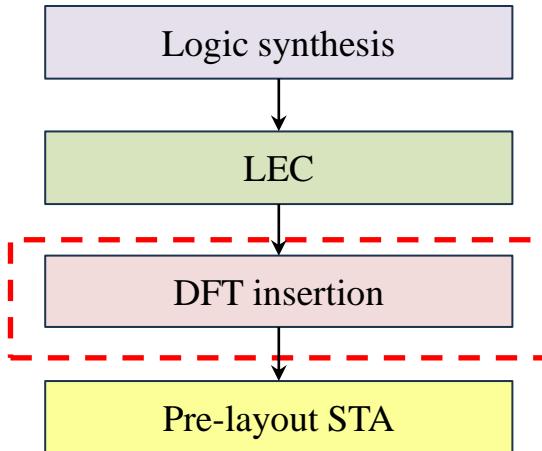
BACK-END DESIGN PROCESS – LEC



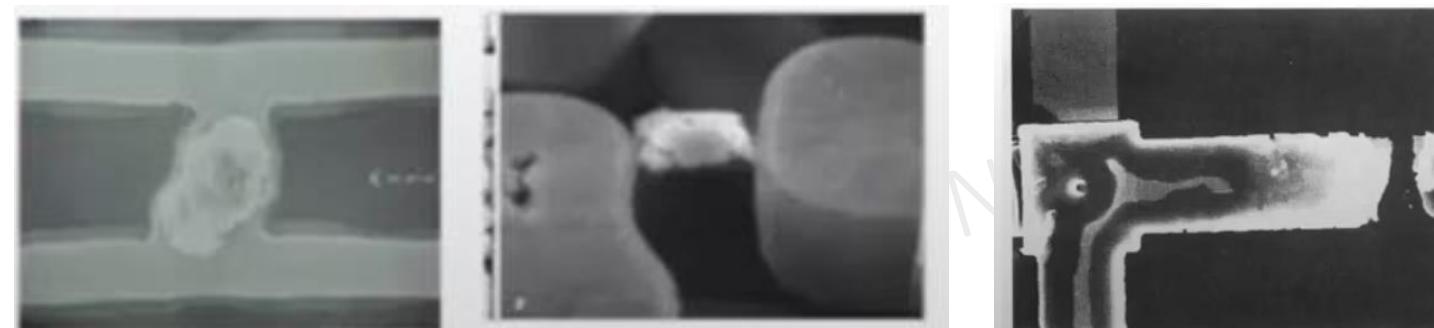
LEC: Logic-Equivalent-Check is the step to check whether output of a step (RTL/gate) and input (RTL/gate) are equivalence. It means, optimization. In this case, the LEC tool will check the equivalence between the RTL (input of logic synthesis) and gate netlist (output of logic synthesis) to see whether they are equivalence

ASIC Design Process

BACK-END DESIGN PROCESS – DFT



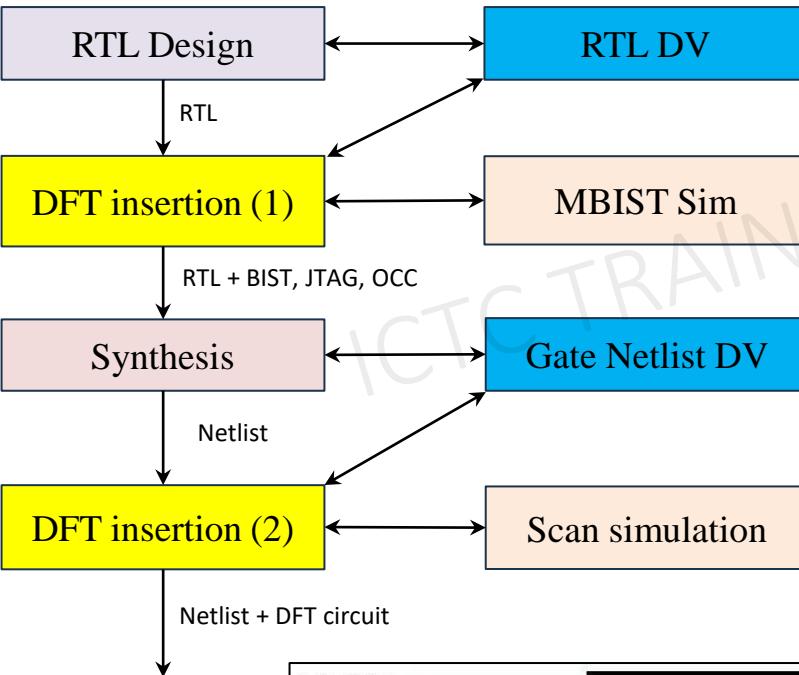
DFT: Design-For-Test is a set of techniques used in digital circuit design to facilitate the testing and verification of the manufactured integrated circuits (ICs). The purpose of DFT is to find failure chips caused by manufacturing process.



DFT insertion is a process to insert additional logics for circuit testing only and does not change the circuit functionalities.

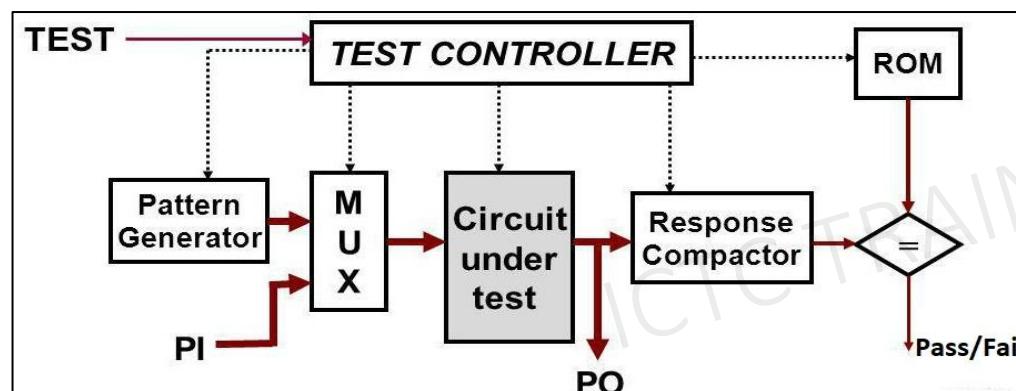
ASIC Design Process

BACK-END DESIGN PROCESS – DFT

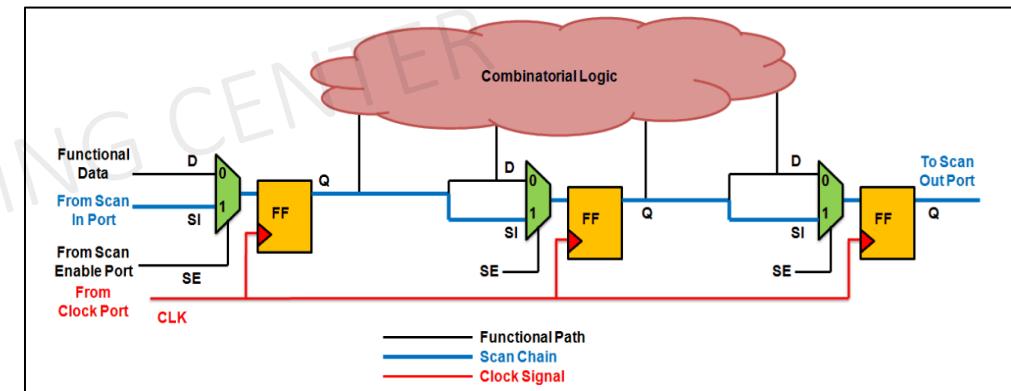


Actually, DFT insertion contributes in 2 steps:

- **Step 1:** after RTL Design, some test logic (RTL) are inserted to the RTL. These are BIST, MBIST, JTAG, OCC. DV must ensure the insertion does not change RTL functionalities.
- **Step 2:** after synthesis, DFT logics are inserted to the netlist. These are scan F.F, scan mux, scan chain ... DV must ensure the insertion does not change the circuit functionalities.



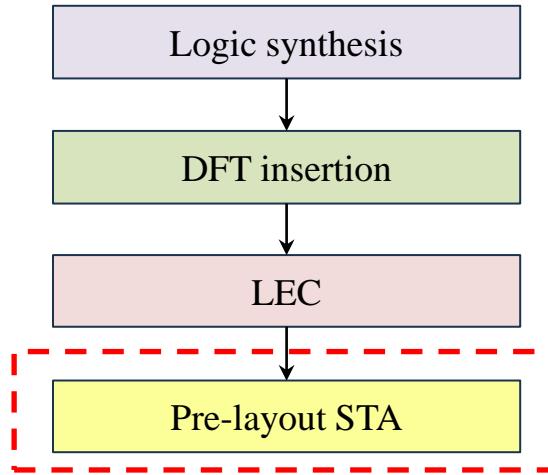
Built-In-Self-Test Basic Structure



Scan Chain

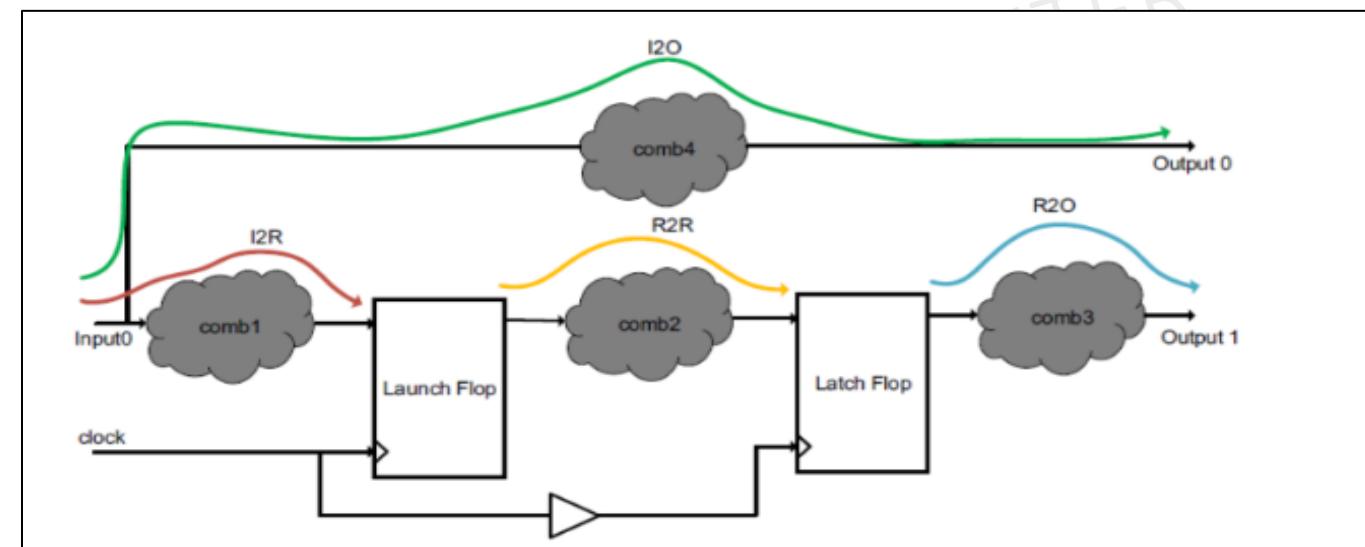
ASIC Design Process

BACK-END DESIGN PROCESS – STA



STA: Static Timing Analysis is a method of validating the timing performance of a design by checking all possible paths for timing violations. This includes 3 main steps:

- The design is broken down into timing paths
- The delay of each path is calculated
- All path delays are checked against timing constraints to determine if the constraints have been met or not (violation) and suggest the fixing if violation occurs.



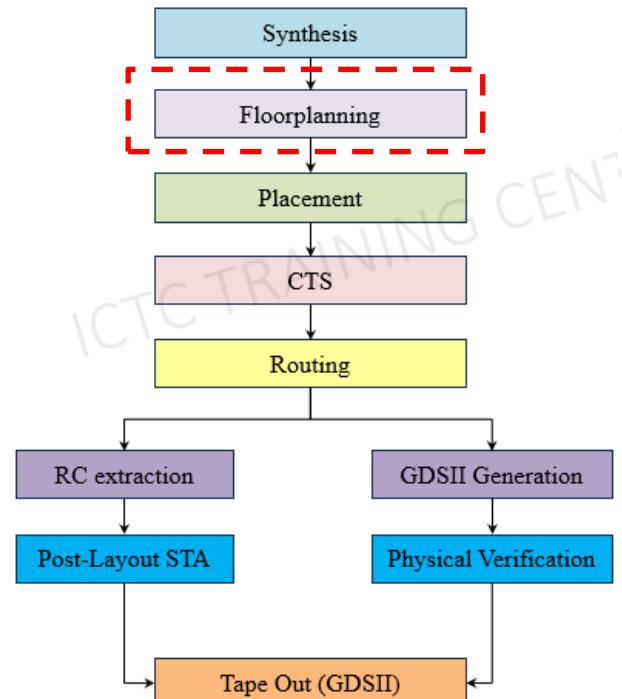
Timing analysis



ASIC Design Process

BACK-END DESIGN PROCESS - FLOORPLAN

VLSI Physical Design is the process of translating circuit schematic (gate level netlist) into silicon physical layout (GDSII).



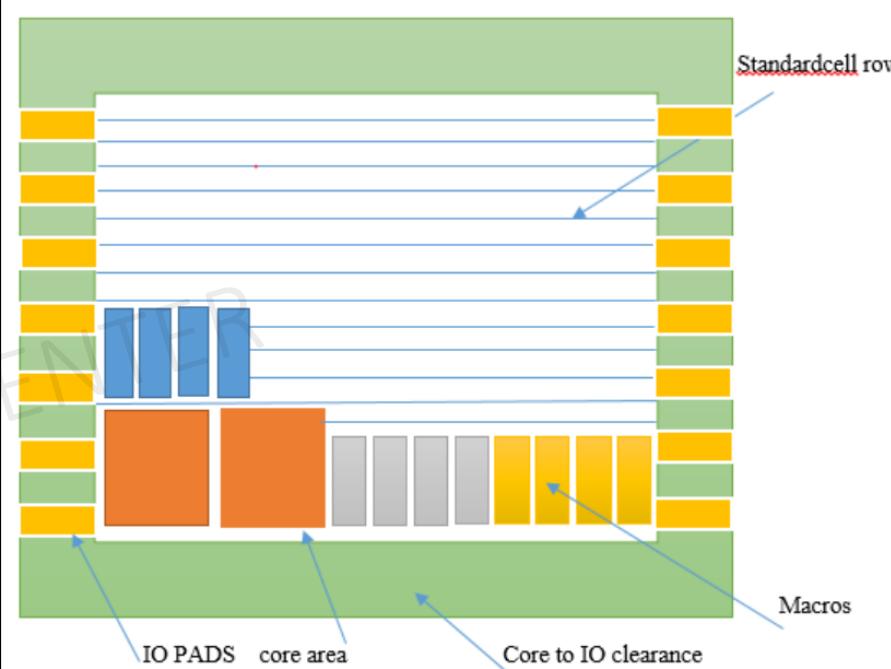
Floorplanning: is the initial stage where designers define the approximate locations and dimensions of various functional blocks, macros, and other important elements on the chip's silicon die.

Work:

- Define size, area
- Finalizing the size, shape, area of subsystems.
- Placement of macros/blocks
- Power planning
- I/O PAD placement

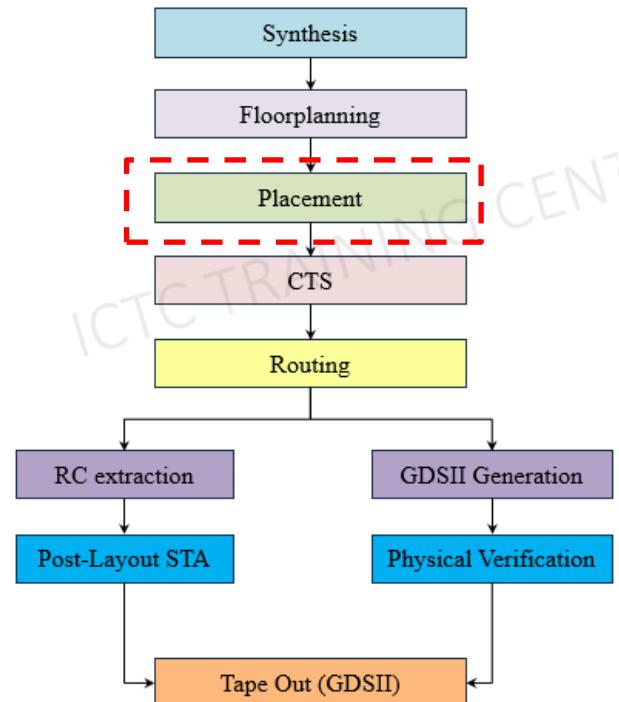
Objective:

- Minimize chip area
- Reducing the wire length
- Making routing easy
- Minimizing the delay
- Less IR drop



ASIC Design Process

BACK-END DESIGN PROCESS - Placement



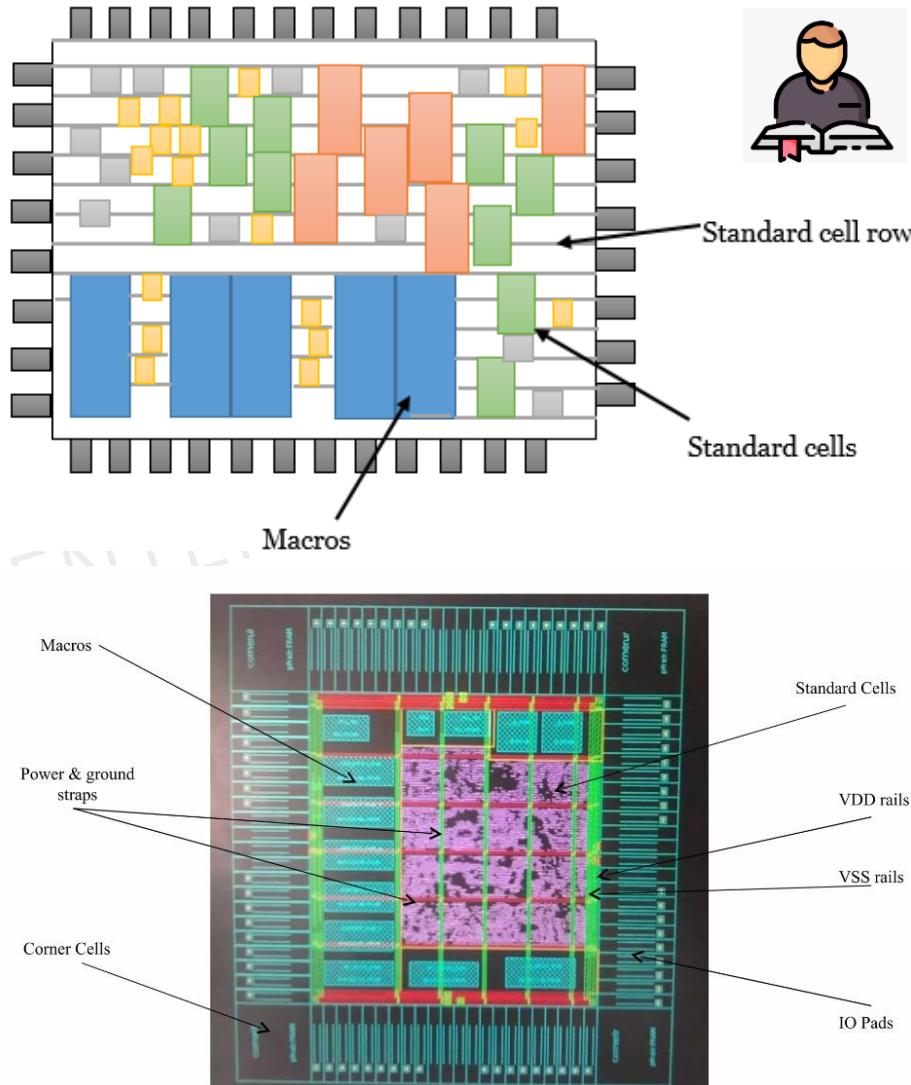
Placement: the individual components of the chip are positioned within the chip's layout area, consider factors such as timing, power, and area constraints

Work:

- Global placement: functional blocks are placed on the chip's floorplan.
- Detailed placement: refine the positions of standard cells within functional blocks.
- Timing/power/congestion optimization

Objective:

- Minimize total area
- Minimize power consumption
- Reduce interconnect and complexity



ASIC Design Process

BACK-END DESIGN PROCESS - CTS



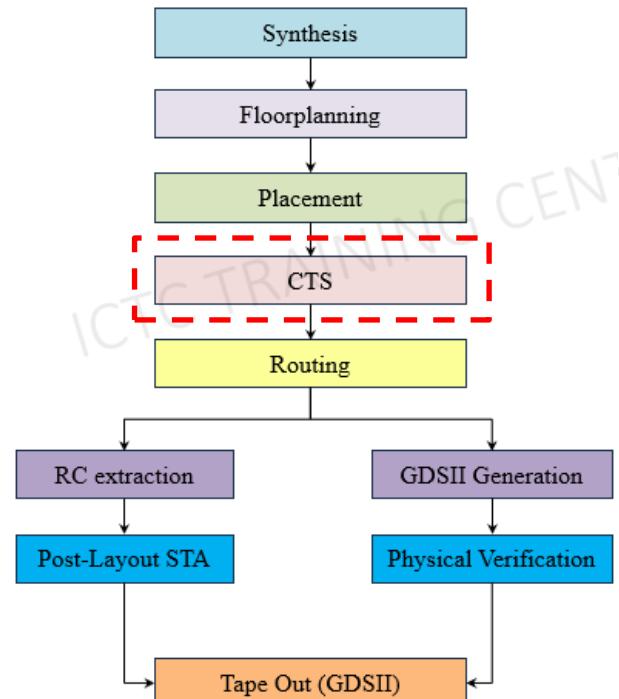
CTS: Clock-Tree-Synthesis is a process of connecting the clocks to the clock pins of sequential elements by using buffers/inverters

Work:

- Clock tree construction: construct a hierarchical tree structure of clock distribution networks.
- Clock buffer insertion: to buffer and amplify clock signals as they propagated through the chip
- Clock routing: clocks are routed through the chip.
- Clock optimization: balance clock tree branches, optimize buffer placement and sizing
- Clock tree verification: using STA to verify clock path timing or simulation and formal verification to validate functionality of the clock network.

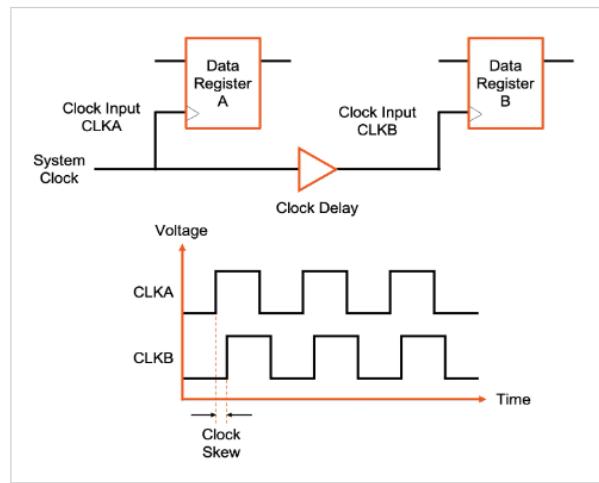
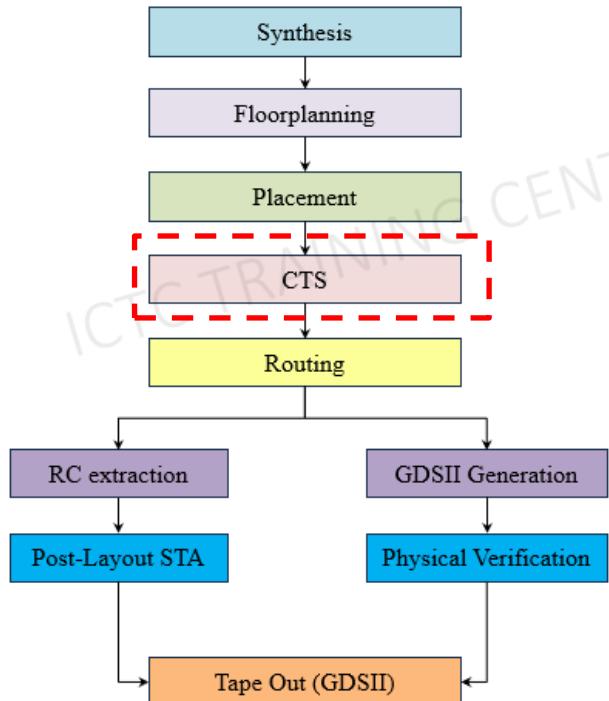
Objective:

- Minimize the insertion delay
- Minimize clock routing congestion
- Minimize power consumption
- To have minimal skew

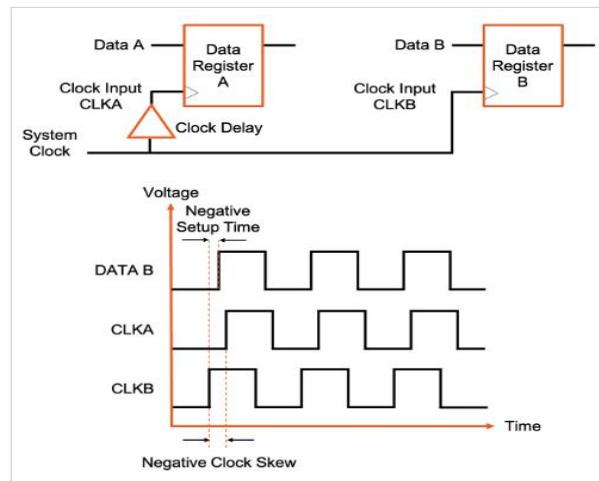


ASIC Design Process

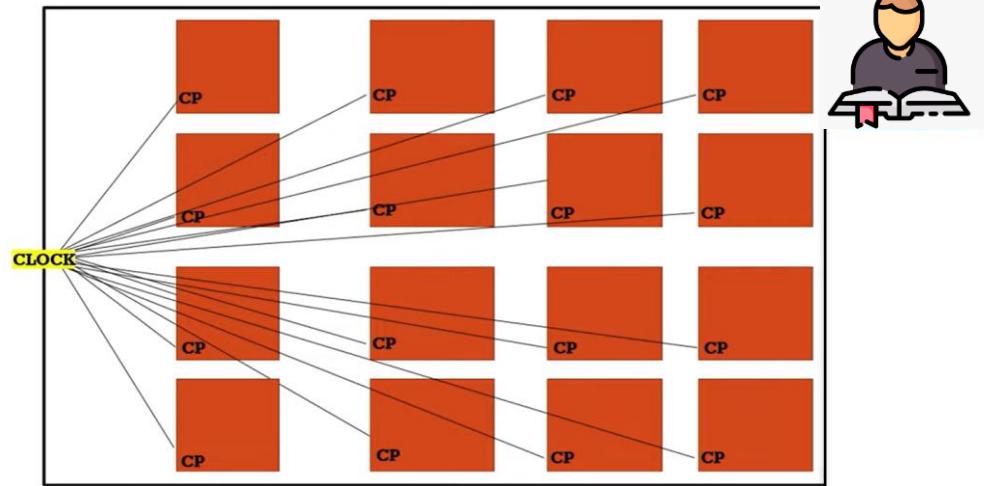
BACK-END DESIGN PROCESS - CTS



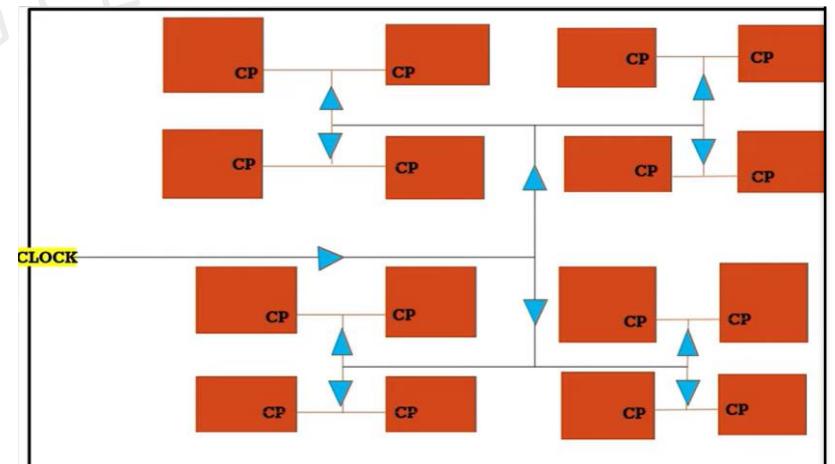
Positive clock skew



Negative clock skew



Before CTS



After CTS



ASIC Design Process

BACK-END DESIGN PROCESS - ROUTING



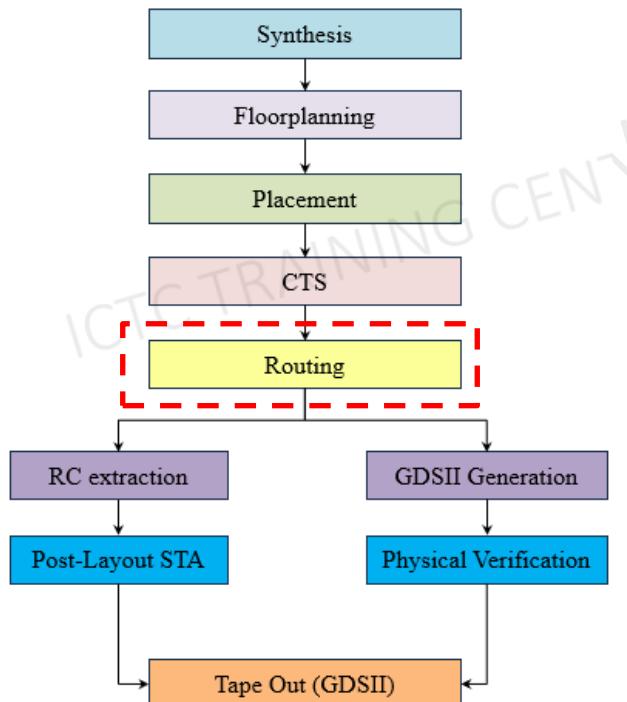
Routing: to create interconnections between the various components of an IC layout

Work:

- **Global routing**: determines the approximate paths for the interconnections between major components of the circuit.
- **Detailed routing**: to establish precise connections between individual components at the transistor level.
- **Clock routing**: distributing clock signals throughout the circuit.
- **Power routing**: distribute power and ground connections throughout the circuit to ensure reliable operation.
- **Routing optimization**: apply some special techniques such as congestion-driven routing or timing-driven routing to improve the quality of routing.

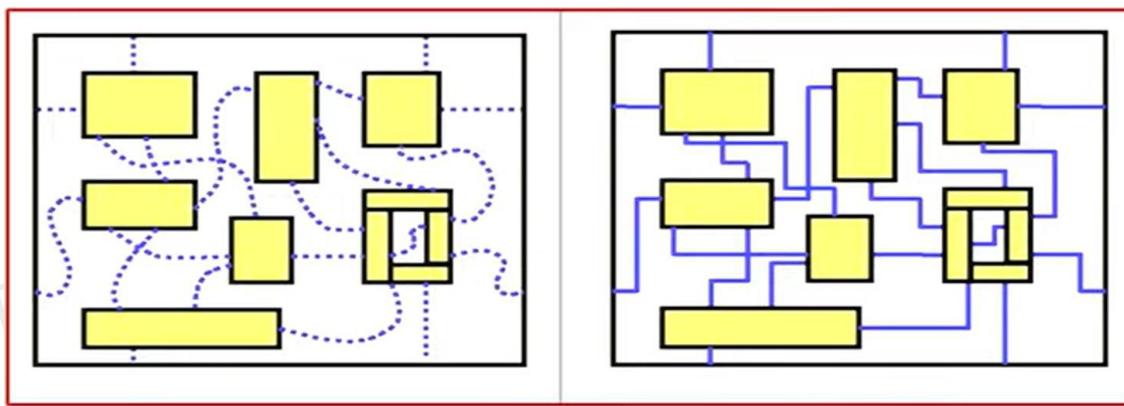
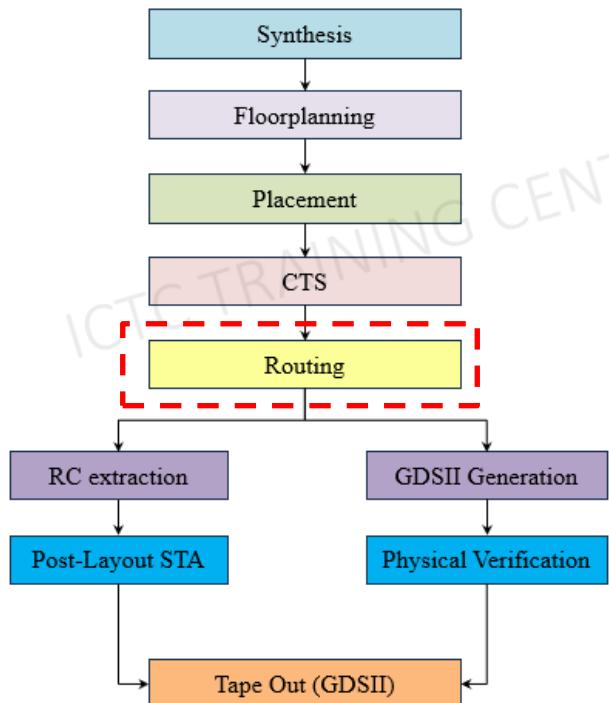
Objective:

- **Signal connectivity**: ensure signals can propagate from source to destination
- **Signal integrity**: minimize signal distortion during transmission
- **Timing closure**: met timing constraints defined by timing requirements.
- **Power distribution**: ensure reliable operation and minimize voltage drops.
- **Area efficiency**: minimize area occupied by routing resources (such as metal layers).
- **Manufacturability**: follows design rule contrains and ensuring design is suitable for the chosen technology node and manufacturing process



ASIC Design Process

BACK-END DESIGN PROCESS - ROUTING



Global Routing

Detailed Routing

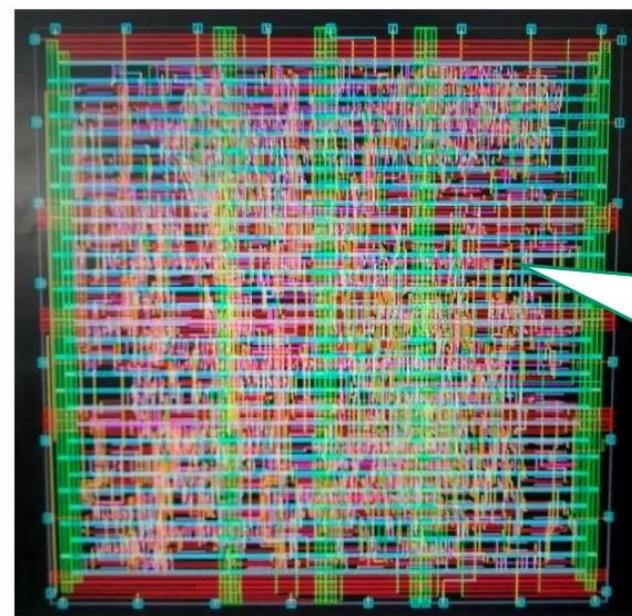


Figure 7 Block level routing

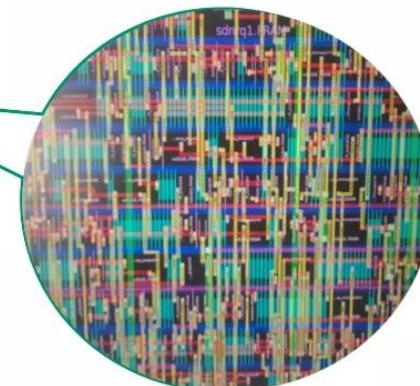


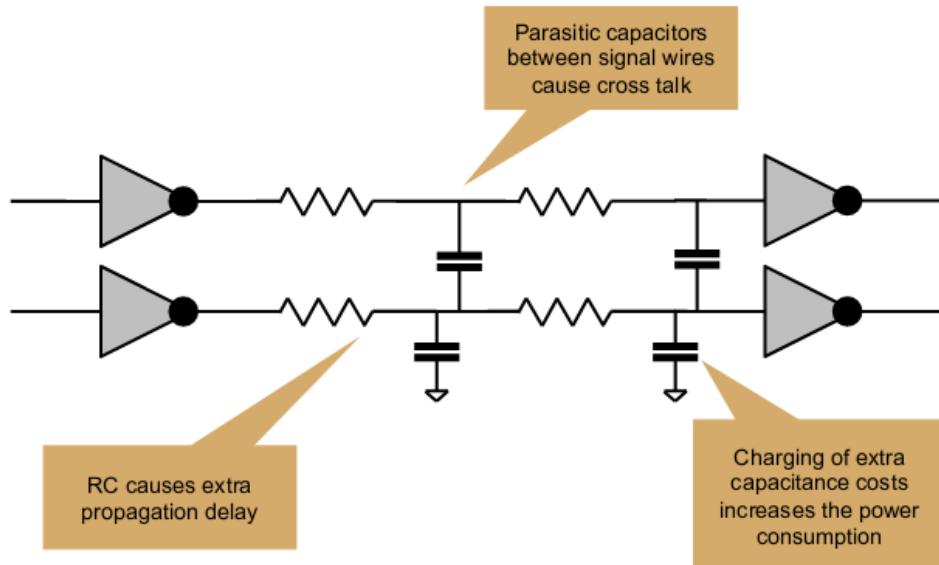
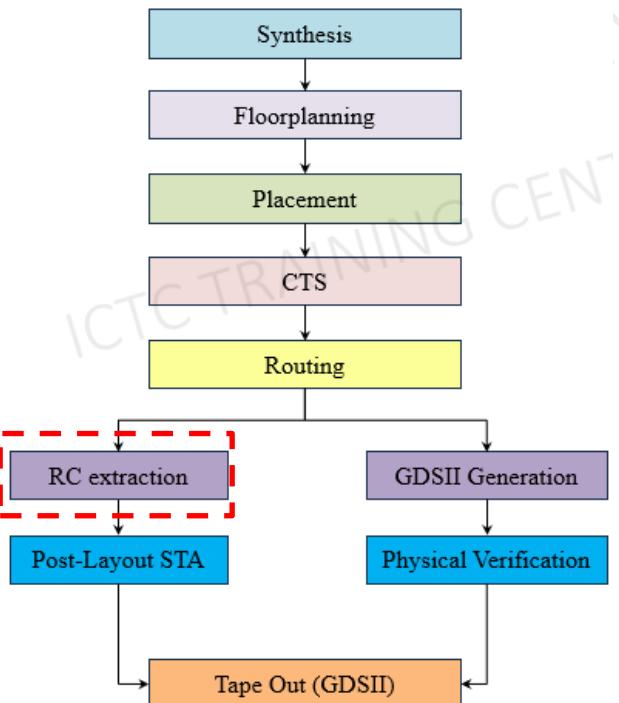
Figure 8 Magnified portion of the block

ASIC Design Process

BACK-END DESIGN PROCESS - ROUTING



RC extraction: a the process of extracting parasitic resistance (R) and capacitance (C) values from the layout of an integrated circuit. These RC values are due to the physical properties of the materials used in the fabrication process and the geometric layout of the circuit components.

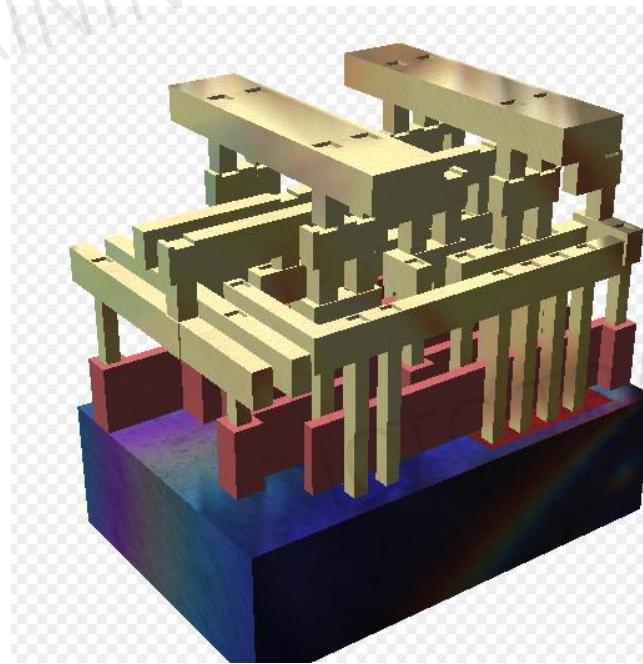
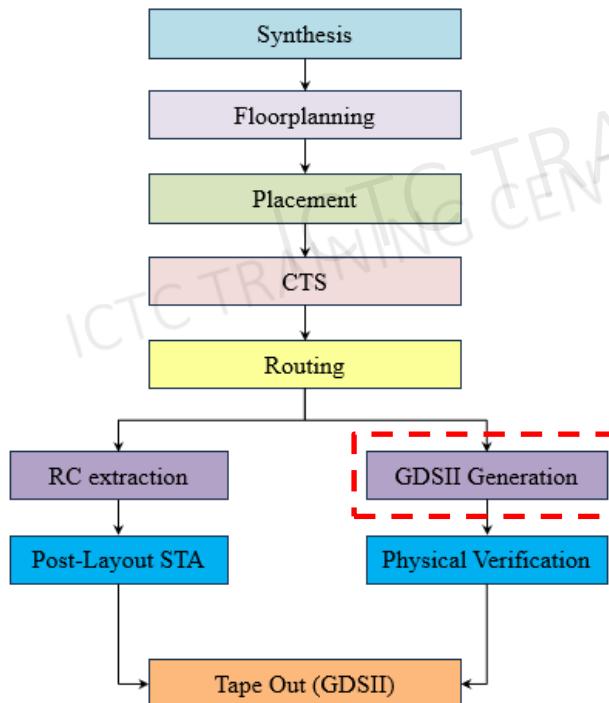


ASIC Design Process

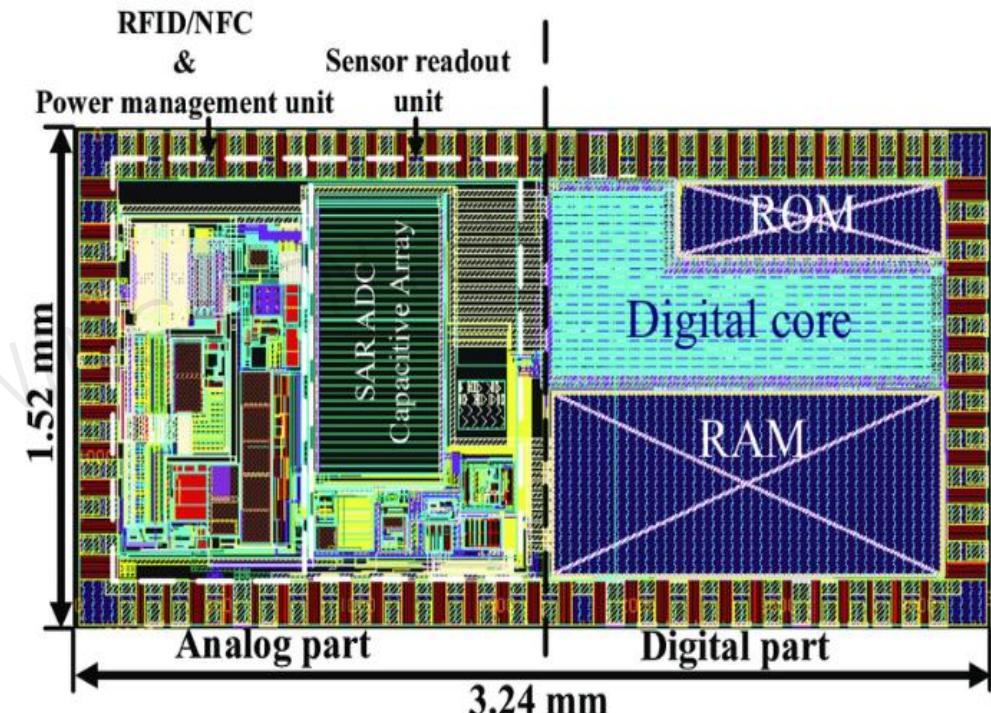
BACK-END DESIGN PROCESS - ROUTING



GDSII Generation: A GDS II (Graphic Data System II) file is a standard file format used in the semiconductor industry to describe the geometric layout of integrated circuit (IC) designs. It is a binary file format that represents the physical layout of a chip in terms of polygons, layers, and other geometric primitives.



A rendering of a small GDSII standard cell with 3 small metal layer



GDS file after layout

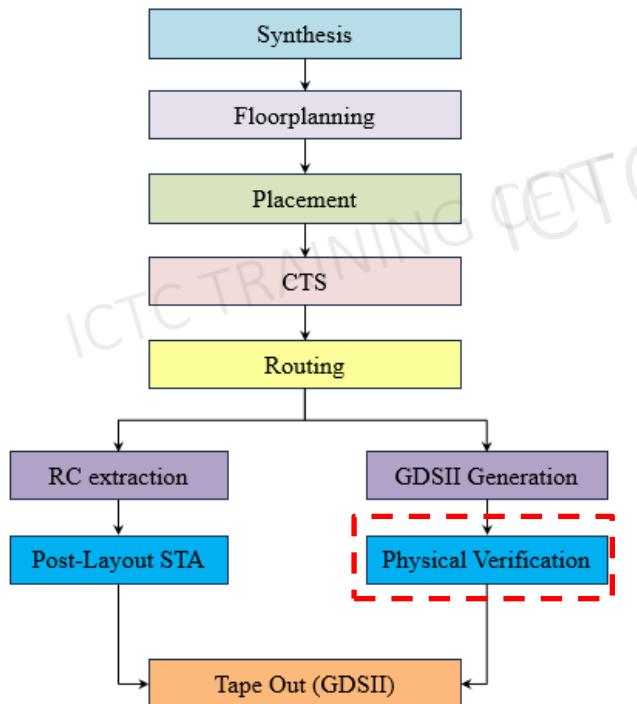
ASIC Design Process

BACK-END DESIGN PROCESS – PHYSICAL VERIFICATION



Physical Verification: to check the layout file (GDSII file) by EDA tools to verify layout characteristic satisfied product requirement.

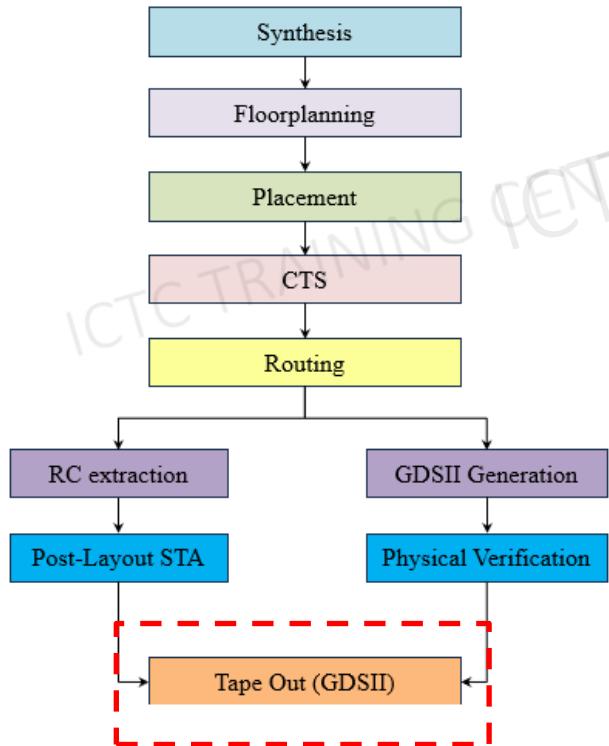
Work and Objective



- **DRC (Design Rule Check):** to check the layout satisfied design rules provide by the foundry.
- **LVS (Layout versus Schematic Check):** to check the consistency between the layout and schematic represents the circuit, to prevent connectivity errors and ensure functional correctness
- **Antenna Rule Check:** to ensure every transistors in the layout are protected from oxide breakdown during fabrication. This is called “attenna effect” during dry-etching phase of fabrication and can be avoided during routing, then re-check at the PV stage
- **ERC (Electrical Rule Check):** check potential issue like short circuit, open circuits, floating nodes, and unintended connections that could affect the electrical performance of the chip.
- **Density Check:** ensure the appropriate density levels across the layout to prevent manufacturing issues.
- **Manufacturability Check:** to check metal density, spacing rule specific to the semiconductor manufacturing process being used. Ensure the layout is compatible with the fabrication process and can be successfully manufactured without costly errors or yield loss.

ASIC Design Process

BACK-END DESIGN PROCESS – TAPE OUT



TAPE OUT: The tape-out process is the final stage in the design flow of an integrated circuit (IC) before it is sent for fabrication. It involves preparing the design files, verifying the design, and generating the final set of data files needed to manufacture the semiconductor chips.

ASIC Design Process

BACK-END DESIGN PROCESS – QUESTIONS



Question 1: The purpose of DFT is to find functional bugs?

Question 2: What's the basic difference between Static Timing Analysis and Dynamic Timing Analysis?

Question 3: Which phases in the PD design flow to ensure the clock connectivity and minimize clock skew?

Question 4: Does Physical Verification step belongs to the Physical Design flow ?

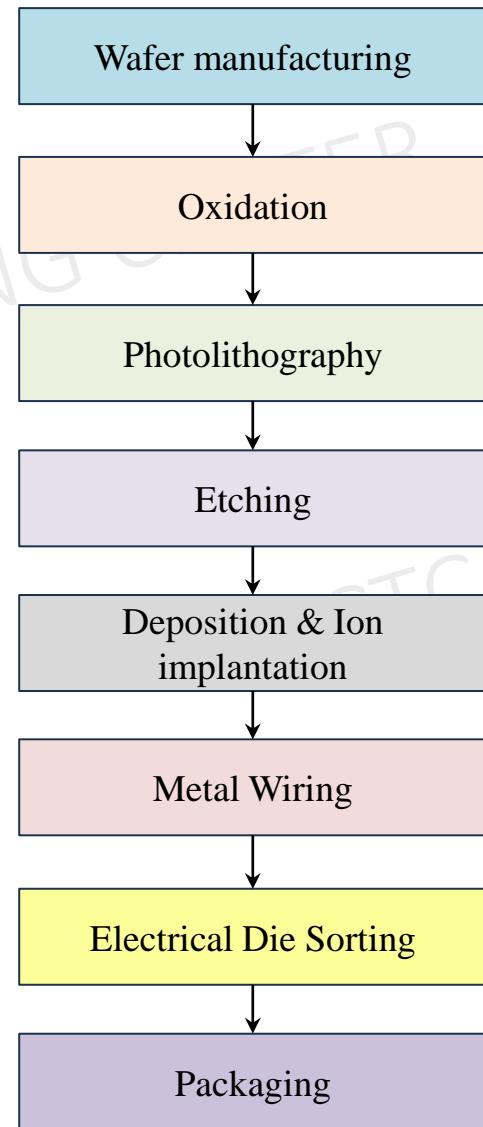
Question 5: How many STA steps do we need in the Back-end Design Flow ?



ASIC Design Process

FABRICATION PROCESS (SELF-LEARNING)

Watch following video for detail
[Manufacturing process](#)



ASIC Design Process

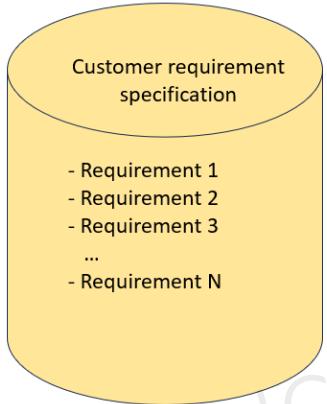
CHIP TESTING (VALIDATION)

Test machine works with ATE (Automatic Test Equipment) to test the chip characteristics and ensure there is no manufacture error.

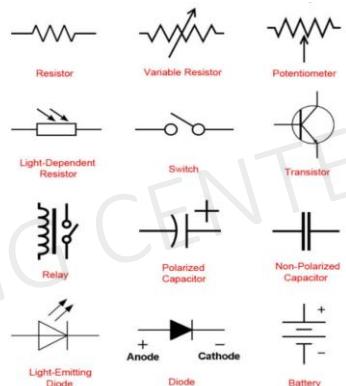


ASIC Design Process

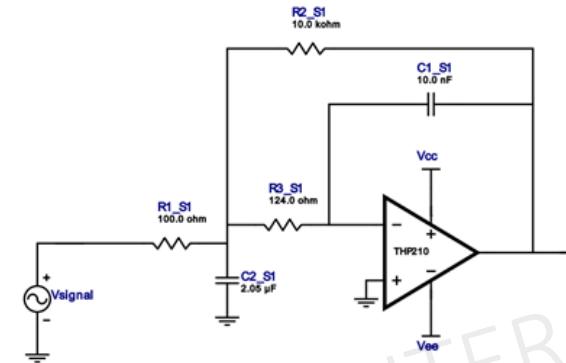
ANALOG DESIGN PROCESS



Requirement collection & analysis



Component selection



Circuit design

The screenshot shows the Cadence Schematic Editor interface. It features a toolbar at the top, a 'File Hierarchy' panel on the left, and a main workspace displaying a circuit schematic. The schematic includes an operational amplifier (AD8251) with various resistors (R1, R2, R3) and capacitors (C1, C2) connected to form a specific circuit topology. A red box highlights the toolbar and the 'Place Part' dialog box in the center-right. The 'Component Selection' tab is visible above the workspace.

Simulation

The screenshot shows the Cadence Layout Editor interface. It features a toolbar at the top, a 'View Tools' panel on the left, and a main workspace displaying a green PCB layout with various components and interconnects. A red box highlights the 'General Tools' and 'Analysis and NETLIST Tools' tabs at the top. Another red box highlights the 'Workbench Tools' panel on the left and the 'General de Camadas' panel on the right. The 'Layout and Layers' tab is visible on the far right.

Layout

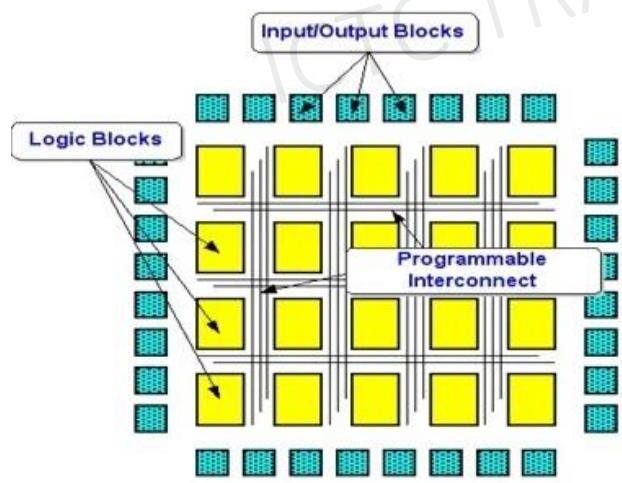
ASIC Design Process

FPGA

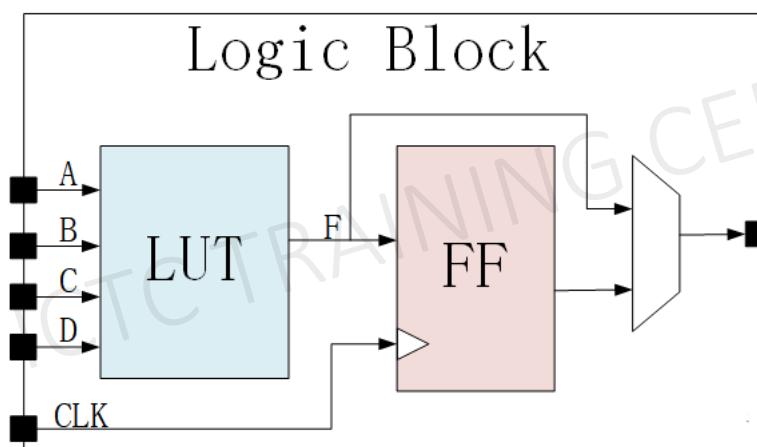


FPGA stands for **Field Programmable Gate Array**, is a hardware reconfigurable architecture. In other words, it is a chip that can be programmed to change its internal structure. FPGAs have been used for many years as a low-volume replacement for application-specific chips (ASICs).

The FPGA architecture mainly includes four parts: configurable logic block (CLB), input and output block (IOB), internal wiring (Interconnect) and other embedded units. The internal structure of the FPGA device is shown below.



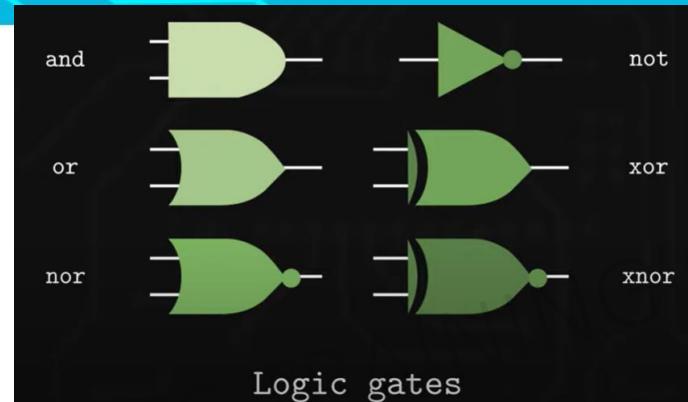
FPGA structure



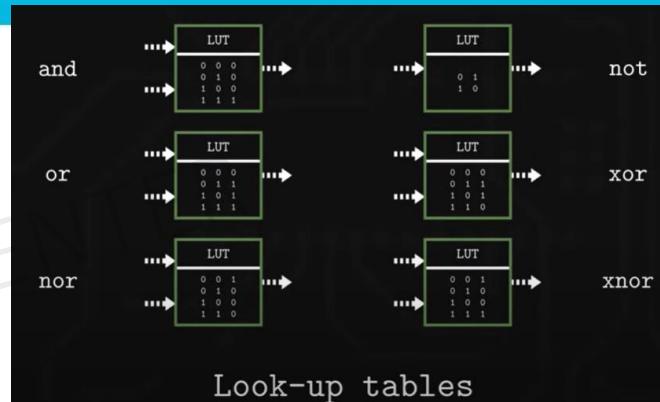
CLB: Configuration Logic Block

ASIC Design Process

FPGA



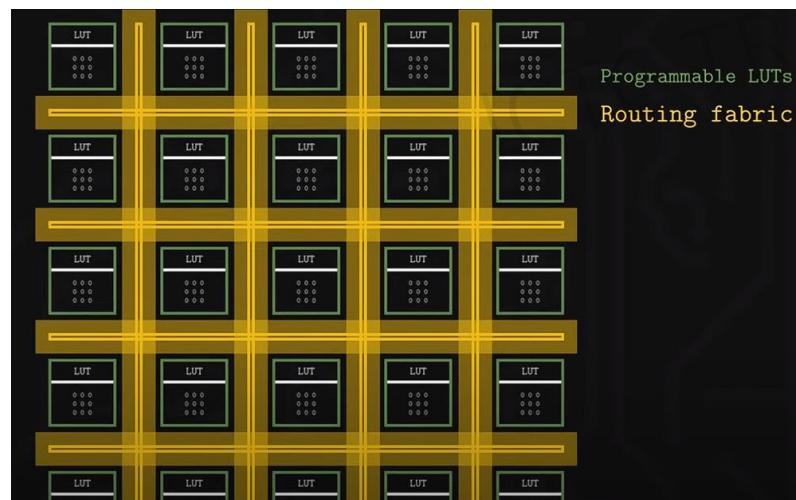
(1) Logic gates



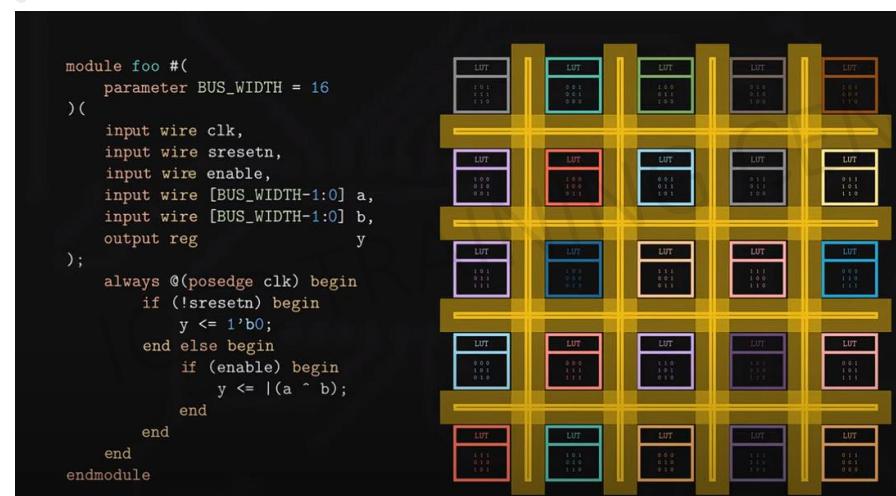
(2) Logic gates are implemented using LUT



(3) Make the LUT programmable



(4) Connect the LUTs together



(5) Allow user to re-program the LUTs to whatever they want

ASIC Design Process

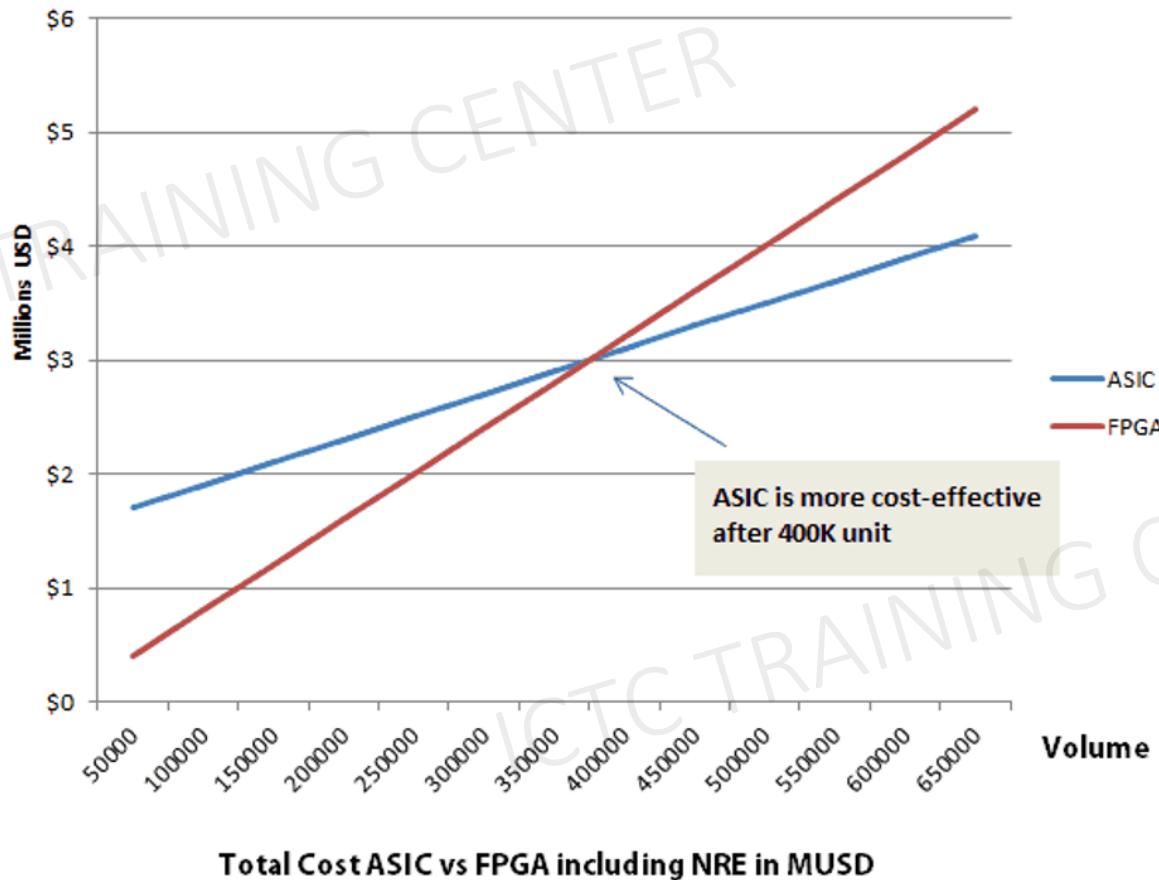
ASIC or FPGA

Criteria	FPGA	ASIC
Time to Market	Fast	Slow
NRE*	Low	High
Design flow	Simple	Complex
Unit Cost	High	Low
Performance	Medium	High
Power Consumption	High	Low
Flexibility	Can change	Fixed

*Non-recurring engineering (NRE) cost refers to the one-time cost to research, design, develop.

ASIC Design Process

ASIC or FPGA



SESSION 4

SUMMARY

SUMMARY:



- ❑ ASIC stands for Application Specific Integrated Circuit
- ❑ ASIC Digital Design Flow has 3 main stages
 - System specification
 - Design (Front-End & Back-End)
 - Front-End Design includes 2 main steps: RTL Design & Design Verification
 - Back-End Design includes 3 main steps: Design For Test, Static Timing Analysis and Physical Design.
 - Fabrication & Testing
- ❑ Analog Design Flow is very different from Digital Design Flow.
- ❑ FPGA stands for Field Programmable Gate Array.
- ❑ FPGA is famous for its programmable ability, faster Time-to-market and low cost at low volume, but not well-optimized in terms of power, area and application specific compare to ASIC

HOMEWORK

Homework:

- Access below link and do the quiz:

<https://forms.gle/GMTa8oim4dTKWMj37>

Snapshot the result and attached to your homework submit form.



This quiz has 20 questions corresponds to 20 points

- First 10 points for standard level
- Second 10 points for advanced level

For example: if you get 15 points, you will have 10 points standard level + 5 points advanced level.

COMMON KNOWLEDGE

CPU, MCU and SoC



- **Central Processing Unit (CPU)** or Processor is the logic circuitry that responds to and processes the basic instructions that drive a computer.
- A processor performs arithmetical, logical, input/output (I/O) and other basic instructions that are passed from an operating system (OS). Most other processes are dependent on the operations of a processor.

- **Microcontroller Unit (MCU)** is an intelligent semiconductor IC that consists of a processor unit, memory modules, communication interfaces and peripherals. The MCU is used across a broad range of applications, including washing machines, robots, drones, radio and game controllers.
- In modern terminology, a microcontroller is similar to, but less sophisticated than, a system on chip (SoC)

- **System on Chip (SoC)** is an integrated circuit system comprising all necessary components integrated on a single chip. These components may include digital circuits, analog circuits, and the integration of both (mixed-signal).
- In modern semiconductor industry, ASIC or SoC systems can be understood as nearly equivalent.



COMMON KNOWLEDGE

MCU vs SOC

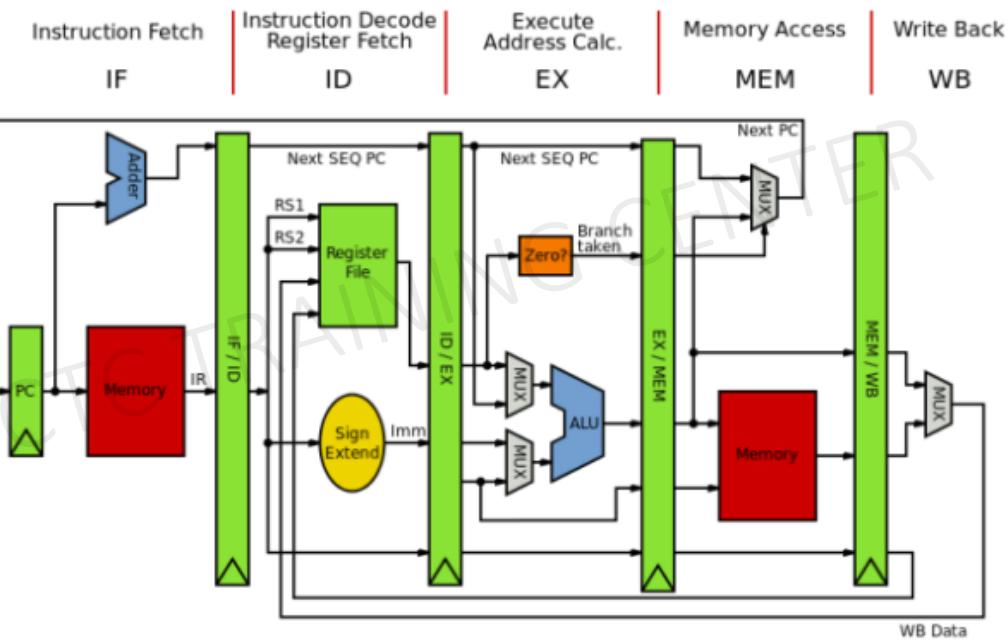
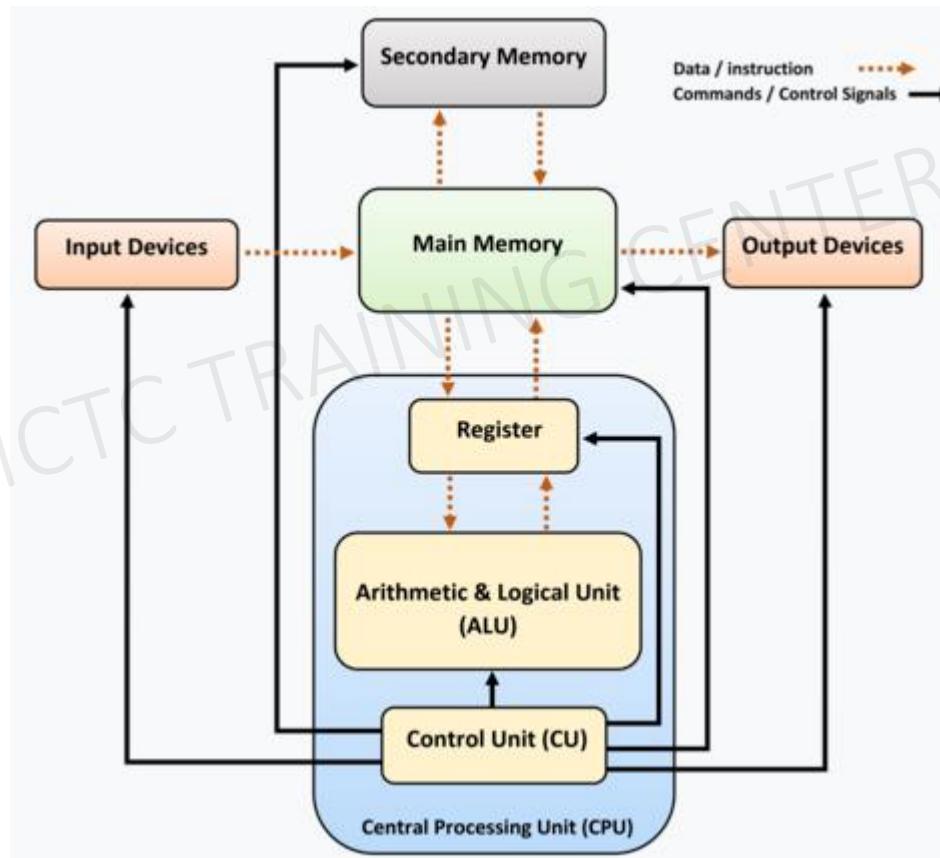


Nowadays, MCU and SOC are very similar in terms of architecture. Below are some basic differences.

Criteria	MCU	SOC
Die Size	Smaller	Bigger
Power	Smaller	Bigger
Complexity	More simple, less peripherals	More complicated, more peripherals
Cost	Lower	Higher
Performance	Lower	Higher
CPU core	1 or 2	1 or 2 or more
Application	General purpose	Application Specific

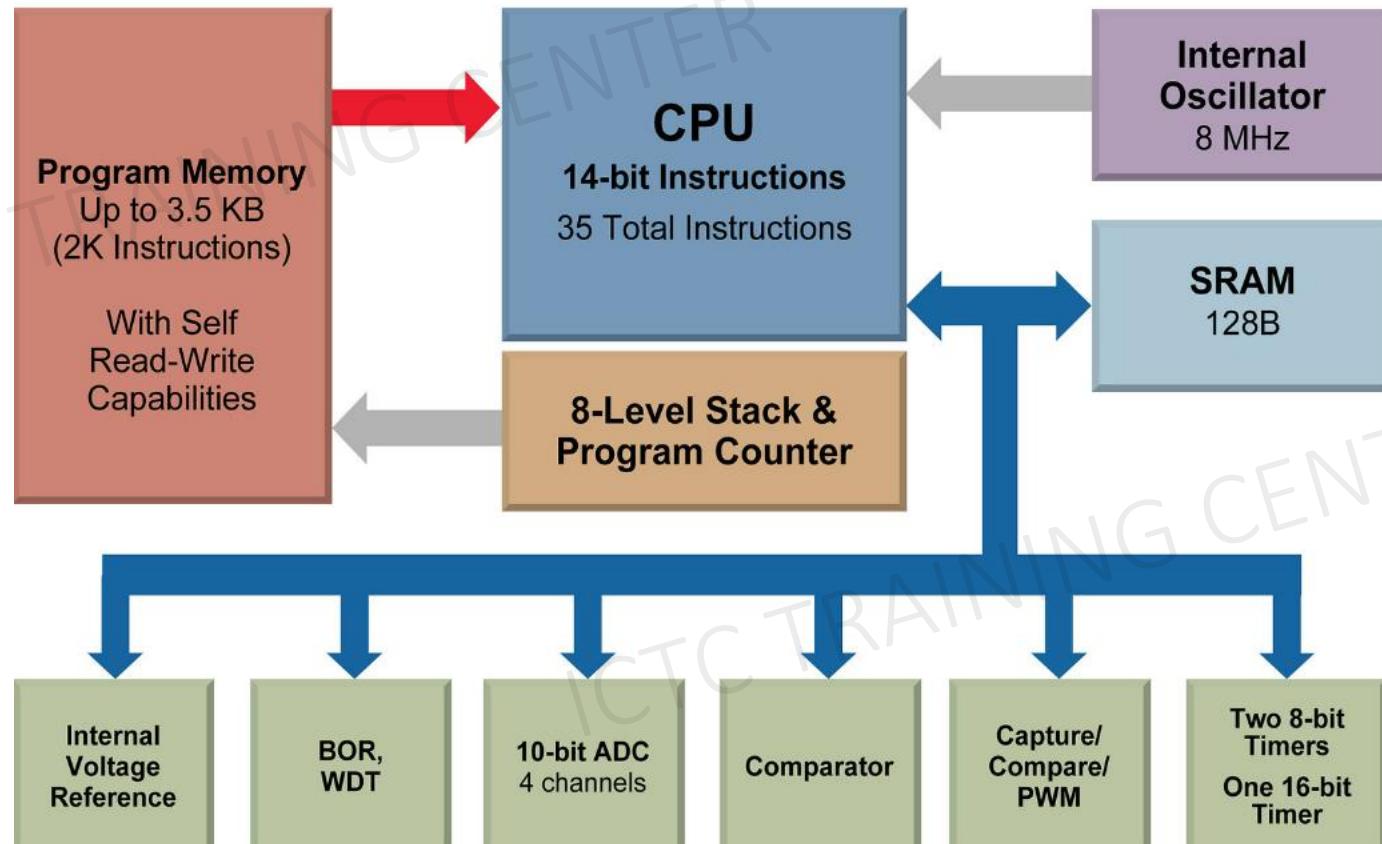
COMMON KNOWLEDGE

CPU BLOCK DIAGRAM



COMMON KNOWLEDGE

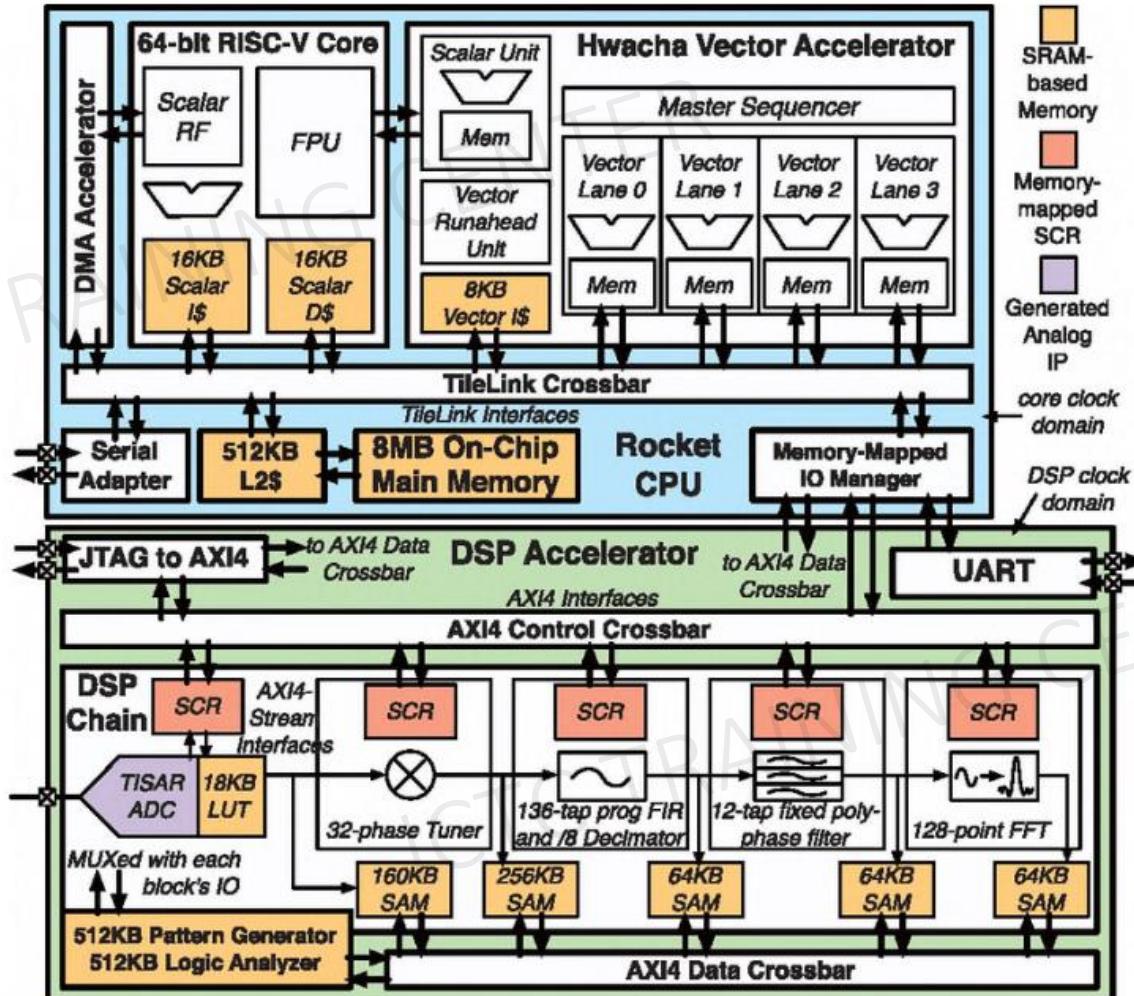
MCU BLOCK DIAGRAM



Simple MCU block diagram

COMMON KNOWLEDGE

SOC BLOCK DIAGRAM



SOC block diagram