VIETNAM NATIONAL UNIVERSITY,
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

INTERNSHIP REPORT

SEMESTER 222 – ACADEMIC YEAR: 2022-2023

- **MAJOR: Computer Engineering**
- **EDUCATION PROGRAM: High-quality training program**

- **INTERNSHIP HOST ORGANIZATION:**
  ADTechnology & SNST Viet Nam
- **DIRECT SUPERVISOR FROM THE HOST ORGANIZATION:**
  Mr. Phan Vĩnh Vương - DFT Senior Manager
- **ACADEMIC ADVISOR/INSTRUCTOR/REPORT ASSESSOR FROM THE FACULTY:**
  Mr. Mai Đức Trung
- **INTERNSHIP STUDENT (IS):**
  FULL NAME: Hoàng Đỗ Phương Nguyên
  STUDENT ID: 1952360
  CLASS: CC20KTM1

HO CHI MINH CITY, AUGUST/2023

Công ty TNHH ADTechnology & SNST Việt Nam

Địa chỉ: Tầng 10, Tòa nhà Sacom Chíp Sáng, Khu CNC, Lô T2-4, Phường Tân Phú, Quận 9, HCM

SĐT: +84 28 7109 9400

Website: https://adt-snst.vn

CHƯƠNG TRÌNH
THỰC TẬP TỐT NGHIỆP (TTTN)/ THỰC TẬP NGOÀI TRƯỜNG (TTNT)

DÀNH RIÊNG CHO SINH VIÊN

KHOA KH&KT MÁY TÍNH – TRƯỜNG ĐH BÁCH KHOA – ĐHQG TP.HCM

HỌC KỲ 3/2022-2023 (HK223)

(Thời gian thực tập thực tế từ 12/06-11/08/2023)

I. NỘI DUNG:

1. **Giới thiệu sơ bộ về DN**:

Công ty ADTechnology & Snst Việt Nam là một trong những công ty đi đầu trong lĩnh vực thiết kế vi mạch với hơn 4 năm thành lập từ năm 2018, đến nay đã phát triển mở rộng thành lập hai chi nhánh tại Thủ Đức và Bình Thạnh với số lượng hơn 230 nhân viên.

Hiện tại công ty đang là đối tác chính của Samsung, cung cấp các dịch vụ thiết kế tiên tiến, giải pháp tối ưu đến khách hàng.

2. **Chương trình**:

Ngoài kiến thức chuyên môn về ngành Khoa học và Kỹ thuật Máy Tính - Công nghệ Thông tin, chương trình **đảm bảo/ cam kết** trang bị cho sinh viên (SV) đầy đủ kỹ năng sau:

(DN sẽ chấm điểm kỳ thực tập cho SV theo tiêu chí trong bảng dưới đây – mẫu bảng điểm đính kèm).

Stt	Tiêu chí	Điểm tối đa	
1	**Kỹ năng chuyên môn:** Mức độ hoàn thành công việc.	**20**	
2	**Khả năng làm việc theo nhóm:**	**30**	
a	Khả năng xác định được vai trò, vị trí và công việc của mình trong nhóm. Xác định được quy trình làm việc trong nhóm một cách rõ ràng.	10	
b	Khả năng cộng tác với các thành viên khác trong nhóm. Khả năng trao đổi với các thành viên trong nhóm để phối hợp thực hiện các nhiệm vụ chung và nhiệm vụ của bản thân. Khả năng dung hòa được các hoạt	15	

	động của mình với tiến độ chung. Khả năng giúp đỡ các thành viên khác nếu có thể để cùng đảm bảo hoàn thành kế hoạch nhóm đúng tiến độ.		
c	Mức độ tích cực đóng góp cho các hoạt động của nhóm. SV phải đóng góp ý kiến tích cực trong các buổi họp nhóm nhằm giải quyết vấn đề của nhóm cũng như giúp nhóm phát triển tốt hơn.	5	
3	**Kỹ năng giao tiếp:**	**35**	
a	Mức độ tham gia các hoạt động trainning về giao tiếp của chương trình. SV phải tham gia đầy đủ các buổi trainning được tổ chức.	5	
b	Khả năng trình bày (presentation skill): Khả năng chuẩn bị, tổ chức một buổi thuyết trình theo yêu cầu bao gồm việc chuẩn bị tư liệu, bài trình bày, phương tiện, kế hoạch,…	5	
c	Khả năng sử dụng kỹ năng thuyết trình.	5	
d	Khả năng sử dụng các trang thiết bị phục vụ cho bài presentation như máy tính, phần mềm, máy chiếu,…	5	
e	Khả năng sử dụng kỹ năng viết báo cáo. Khả năng nắm bắt các phương thức trình bày các dạng văn bản phổ biến như: thư, thư điện tử, đề nghị, báo cáo,…	15	
4	**Khả năng hòa nhập với môi trường doanh nghiệp:**	**15**	
a	Mức độ tuân thủ kỷ luật làm việc về: giờ giấc, nội quy, tác phong, mức độ chấp hành yêu cầu của cấp trên.	5	
b	Mức độ hòa nhập với các hoạt động chung của DN, khả năng hòa nhập với văn hóa DN. SV phải tham gia một cách tích cực và hòa đồng với các hoạt động chung.	5	
c	Mức độ hiểu biết về cách tổ chức hoạt động của DN: văn hóa tổ chức, giao tiếp & sinh hoạt của DN, mức độ hòa nhập với văn hóa chung của DN.	5	
Tổng điểm:		**100**	

- Số lượng suất thực tập: 01
- Tổng thời gian làm việc của đợt thực tập: từ ngày **12/06-11/08/2023** (tối thiểu 240 giờ làm việc).
- Thời gian làm việc trong ngày: cụ thể buổi sáng (hoặc chiều) từ 8:00 buổi sáng đến 17:00 buổi chiều. Hoặc theo thỏa thuận. **Tối thiểu SV phải được sắp xếp ngồi làm việc tại trụ sở DN 2,5 ngày trong một tuần**.
- Địa điểm làm việc (SV cần được DN sắp xếp chỗ ngồi làm việc tại DN, để SV có cơ hội hấp thu văn hóa DN trong quá trình thực tập): Chi nhánh công ty – Tầng 4, Tòa nhà Nice Building, Điện Biên Phủ, Phường 25, Bình Thạnh, Tp. HCM
- Ưu đãi (nếu có): Lương thực tập và phụ cấp (theo thỏa thuận công ty)
- Thông tin thêm (nếu có): ……
- Khối lượng/ nội dung công việc cụ thể:

(Nêu rõ tên/ nội dung đề tài, ngôn ngữ lập trình/ framework mà SV được tham gia/ sử dụng)

Week 1	Week 2	Week 3	Week 4
Study Linux environment: directory, file manipulation, basic command, C-Shell language.	Continue to practice C-Shell (awk, grep, sed) Learn Tcl language, tool command.	Overview about IC design flow, DFT flow, LDRC (logic design rule check), Insertion flow, some basic command using during Insertion. Tools: DC (Design-compiler), PT (Prime time)	Practice Insertion with real Project data (Simple block) Tools: DC (Design-compiler), PT (Prime time)
Week 5	**Week 6**	**Week 7**	**Week 8**
Overview about Formal verification flow, ATPG (auto test pattern generation) concepts, some basic command using in ATPG Tools: Test MAX, FM	Practice run formal verification, ATPG with real Project data (Simple block) Tools: FM (Formality), Test MAX	Advance insertion concepts: OCC (on chip clocking), TPI (test point insertion) Tools: DC (Design-compiler), PT (Prime time)	Practice Insertion with real Project data (Block have OCC) Tools: DC (Design-compiler), PT (Prime time)

II. HỒ SƠ, PHỎNG VẤN, LIÊN HỆ (DN tự nhận và xử lý hồ sơ):

1. **Hồ sơ**:
   - Hồ sơ bao gồm: CV
   - Địa điểm/ kênh tiếp nhận: gửi CV qua email: dung.ntp@snstfinger.com hoặc đăng kí tại link: https://fresher.adt-snst.vn
   - Hạn nộp: từ ngày 07/04/2023 ~ 25/04/2023
2. **Phỏng vấn/ xét tuyển**: Thời gian/ lịch trình phỏng vấn/ xét tuyển như thế nào?

   Thời gian phỏng vấn: 18/04/2023 ~ 25/04/2023

   Hình thức phỏng vấn: Test IQ và phỏng vấn trực tiếp
3. **Liên hệ**: Khi có thắc mắc, Khoa/ sinh viên có thể liên hệ với ai? (ghi rõ họ tên – nam/ nữ, điện thoại, e-mail)

- **Nguyễn Thị Phương Dung** – HR Staff
- **Email**: dung.ntp@snstfinger.com
- **Phone**: 0931636192

III. DN CAM KẾT VỚI KHOA:

(Điều này rất quan trọng, rất mong Quý DN lưu ý hỗ trợ Khoa/ SV)

1. DN sẽ gởi chương trình **file scan màu** có ký tên đóng dấu hoàn chỉnh cho Khoa sau khi chương trình được Khoa duyệt trong vòng 03 ngày.
2. DN sẽ gởi **file scan màu** có ký tên đóng dấu hoàn chỉnh công văn xác nhận SV trúng tuyển thực tập cho Khoa (nếu có SV trúng tuyển) – theo mẫu của Khoa, trong vòng 03 ngày sau khi xác nhận với Khoa/ SV.

3. Hướng dẫn SV làm việc theo đúng chương trình đã thống nhất với Khoa. **Nếu DN có yêu cầu SV trúng tuyển ký bất kỳ cam kết gì khác, thì cần nêu rõ trong phần "Thông tin thêm (nếu có)" trong chương trình.**
4. DN có mentor chuyên môn có trình độ Thạc sĩ, hoặc trình độ Đại học với 5 năm kinh nghiệm trở lên làm đại diện hướng dẫn chương trình thực tập.
5. Sắp xếp tiếp cán bộ giám sát do Khoa cử sang DN vào khoảng giữa kỳ thực tập (nếu có). Cán bộ được Khoa phân công sẽ trực tiếp liên hệ với DN để thống nhất lịch tiếp này. Cán bộ giám sát có trách nhiệm quan sát/ đánh giá chương trình thực tập của DN có đạt hay không so với tiêu chí Khoa đề ra, sau đó kết luận/ đề xuất Khoa có nên duyệt gởi SV đi thực tập tại DN nữa hay không?!
6. Gửi hồ sơ thực tập về cho Khoa **trước 16g00 ngày 18/08/2023, theo mẫu của Khoa (bản giấy có ký tên - đóng dấu hoàn chỉnh), bao gồm:**
   - Chương trình thực tập (form 2).
   - Công văn xác nhận kết quả xét tuyển (form 3).
   - Bảng điểm (form 4).
   - Bảng đánh giá SV (form 5).

   Địa chỉ nhận hồ sơ bản giấy có ký tên – đóng dấu hoàn chỉnh (đường BƯU ĐIỆN):

   Khoa KH&KT Máy Tính - Trường Đại học Bách Khoa

   268 Lý Thường Kiệt, Phường 14, Quận 10, TPHCM (Nhà A3)

   ĐT: 028 3863 8912 - Ext: 5830

Tp. HCM, ngày 08 tháng 06 năm 2023

Ban Lãnh đạo

(Ký & ghi rõ họ tên – chức vụ, đóng dấu)

TỔNG GIÁM ĐỐC

Kim Hee Seok

CÔNG TY TNHH ADTECHNOLOGY & SNST VIET NAM

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc
---o0o---

TP.HCM, ngày 08 tháng 06 năm 2023

Kính gửi: **Khoa Khoa học và Kỹ thuật Máy Tính,**
Trường Đại học Bách Khoa - ĐHQG TPHCM

Sau thời gian nhận hồ sơ, phỏng vấn – xét tuyển sinh viên theo lộ trình triển khai chương trình Thực tập ngoài Trường/ Học kỳ 3 năm học 2022-2023 (TTNT/HK223) của Khoa, chúng tôi xin báo đến Khoa danh sách sinh viên trúng tuyển TTNT/HK223 tại Công ty ADTechnology & SNST Việt Nam như sau:

Stt	Mã số sinh viên	Họ lót	Tên	Ghi chú
1	1952360	Hoàng Đỗ Phương	Nguyên	Kết quả phỏng vấn: Đậu

Trân trọng kính chào!

BAN LÃNH ĐẠO
(Ký, ghi rõ họ tên – chức vụ, đóng dấu)

CÔNG TY TNHH ADTECHNOLOGY & SNST VIỆT NAM

TỔNG GIÁM ĐỐC
Kim Hee Seok

Contents

Chapter 1

ACKNOWLEDGEMENT

Prosperous conclusion of any course requires support from various personnel and I was fortunate to have that support, direction and supervision in every aspect from Faculty of Computer Science and Engineering of HCMC University of Technology

First of all, I would like to thank all the teachers in Faculty of Computer Science and Engineering HCMC University of Technology, who taught me carefully and how to learn technical knowledge about computer engineering as well as the way to treat well with everybody.

In particular, I would like to express my gratitude to Mr. Phan Vinh Vuong, DFT Senior Manager of ADTechnology and SNST Vietnam, for his direct guidance and generous assistance in helping me finish my work. Additionally, I want to thank the DFT Team of ADTechnology and SNST Vietnam employees for their assistance over the course of my two-month practice there.

Finally, I want to thank everyone who has been kind and helpful to me. I also hope to get feedback from everyone so I can finish the assignment to the highest standard.

Sincerely,

Ho Chi Minh City, 7th August, 2023

Student's signature & fullname

Chapter 2

INTRODUCTION

2.1 Enterprise introduction

ADTechnology & SNST Vietnam is a foreign-invested enterprise, providing information technology services; engineering and technology research and development; research, development, support and design of functional hardware and software for semiconductor ICs (integrated circuits). As the global partner of the Samsung Electronic Foundry division, ADTechnology & SNST Vietnam develops and aims to be globally competitive in technology that can become a growth engine for our clients' businesses along with other companies. company and various partners.
Main tasks of ADTechnology & SNST Vietnam: Perform all stages in SoC/ASIC/IP design process from functional design testing, logic circuit synthesis, test circuit design, physical design, time and power testing to create a complete and ready design database for semiconductor chip fabrication.
Company's address:

Figure 2.1: Logo of ADT & SNST Vietnam

- **Campus 1:** 10th Floor, Sacom Chip Sang Building, Lot T2-4, D1 Street, Saigon High Tech Park, Thu Duc City, HCMC, Viet Nam
- **Campus 2:** 1st, 4th, 5th Floor, Nice Building, 467 Dien Bien Phu Street, Ward 25, Binh Thanh District, HCMC, Viet Nam

2.2 Mission in internship

During my 2-month internship at the company, I was assigned to learn and practice about:

- Understanding the SoC design process.
- Master in C Shell, Linux and TCL to control tools.
- Make the design to test (Design for testability - DFT).

2.3 Internship schedule

- **Week 1:** Study Linux environment: directory, file manipulation, basic command, C-Shell language.
- **Week 2:** Continue to practice C-Shell (awk, grep, sed) Learn Tcl language, tool command.
- **Week 3:** Overview about IC design flow, DFT flow, LDRC (logic design rule check), Insertion flow, some basic command using during Insertion. Tools: DC (Design-compiler), PT (Prime time).
- **Week 4:** Practice Insertion with real Project data (Simple block). Tools: DC Design-compiler), PT (Prime time).
- **Week 5:** Overview about Formal verification flow, ATPG (auto test pattern generation) concepts, some basic command using in ATPG.Tools: TestMAX, FM.
- **Week 6:** Practice run formal verification, ATPG with real Project data (Simple block). Tools: FM (Formality), TestMAX.
- **Week 7:** Advance insertion concepts: OCC (on chip clocking), TPI (test point insertion). Tools: DC (Design-compiler), PT (Prime time).
- **Week 8:** Practice Insertion with real Project data (Block have OCC). Tools: DC (Design-compiler), PT (Prime time) .

Chapter 3

INTERNSHIP CONTENT

3.1 Overview of SoC & DFT

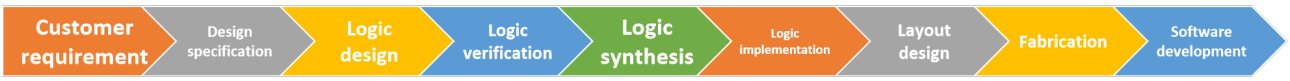

Figure 3.1: VLSI Design Flow

To product one chip, basically, we need to process through 9 steps:

- Step 1 is processed by the Marketing or Selling department.
- Step 2, 3, 4, and 5 are processed by the Design department and are called Frontend Design.
- Step 6 and 7 are processed by the Design department and are called Backtend Design.
- Step 8 is processed by the Manufacture department.
- Step 9 is processed by the Software department.

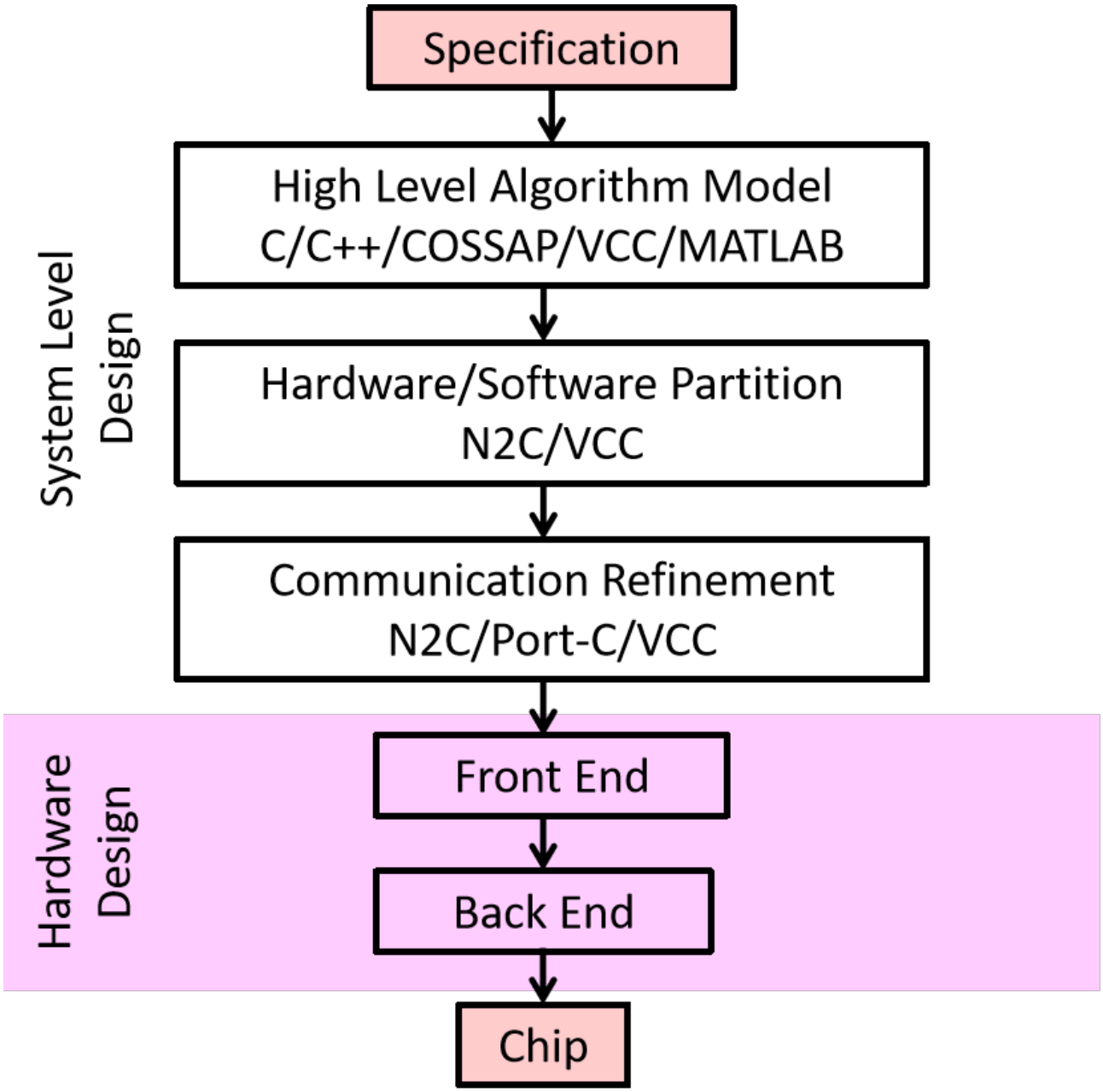

Figure 3.2: SoC Design Flow

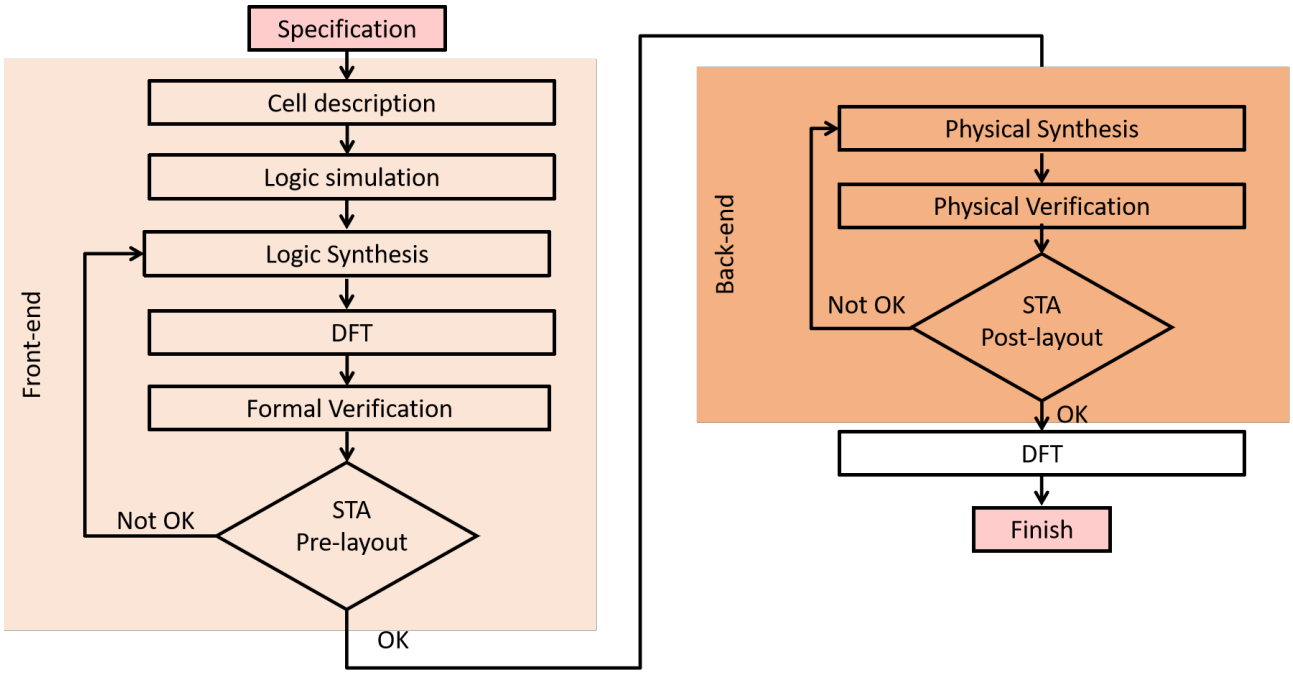

Figure 3.3: Hardware Design Flow

DFT is a design technique that adds test circuits inside the LSI hardware. These added circuits permits to easily apply tests on the designed LSI after manufacturing.
DFT consists of :
- Creating test circuits
- Inserting test circuits into LSI hardware
- Preparing test data for physical failure analysis

3.1.1 Why do we need to integrate DFT circuit into a chip ?

By designing a chip with testability, it becomes easier to identify the structural defects in the chip and fix design errors before shipping products to customers.
It can make the testing process more efficient, reduce the cost and time required for testing. This leads to cost saving during manufacturing process.
It can help to identify any defects early in the process, allow faster repair and improvement in production yield.
It can help accelerate the development cycle by reducing time and effort for testing, debugging.

3.1.2 What is Manufacturing Test and A Physical Defect ?

Manufacturing Test:

- A speck of dust on a wafer is sufficient to kill chip.
- Yield of any chip is < 100%: must test chips after manufacturing before delivery to customers in order to ship only good

parts.

- Automated test equipment (ATE) is used to fix and test packaged IC chip. - Units are either discarded or sent to the failure-diagnosis lab if they fail any program tests.

- Only products that successfully complete all program tests are ever distributed to the final user.

Manufacturing Defects: are physical defects happening during the manufacturing time.

Physical defects:

- Abnormality in MOS transistor: Abnormality in Gate Oxide; Shortage in MOS transistor; Abnormality in wire width, space and thickness; Contamination due to foreign particle.

- Shortage in wiring: Etching residuum; Bridge due to foreign particle; Abnormal wider width and thicker thickness.

- Disconnection in wiring: Abnormal narrower width and thinner thickness; Half disconnection due to foreign particle.

3.1.3 Multiplexed Flip-Flop Scan Style ?

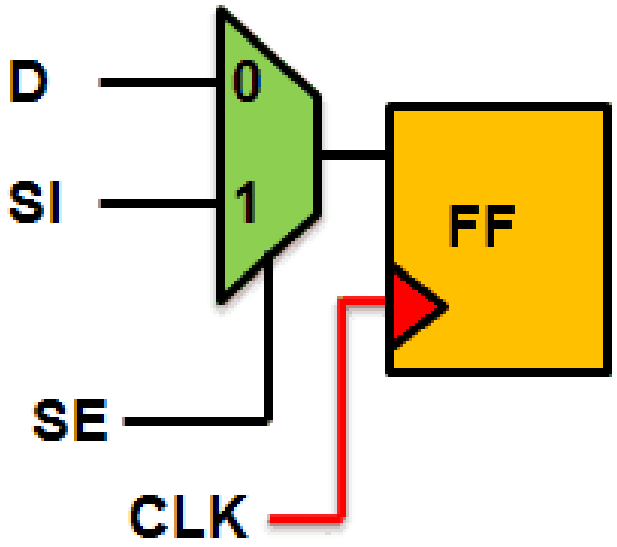

Figure 3.4: A Scan Flipflop

The multiplexed flip-flop scan style uses a multiplexed data input to provide scan shift capability. In functional mode, the scan-enable signal, acting as the multiplexer select line, selects the system data input. During scan shift, the scan-enable signal selects the scan data input. The scan data input comes from either the scan-input port or the scan-output pin of the previous cell in the scan chain.

In normal/capture mode, SE = 0. The value present at the data input DI is captured into the internal D flip-flop when a rising clock edge is applied.

In shift mode, SE = 1. The scan input SI is used to shift in new data to D flip-flop, while the content of D flip-flop is being shifted out.

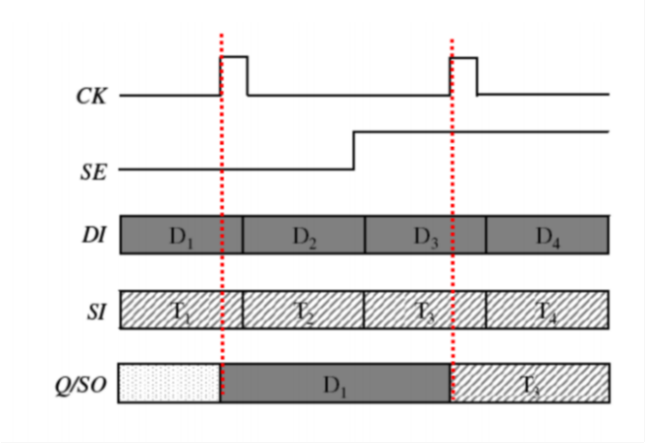

Figure 3.5: A Scan Flipflop function

3.2 Fault Model

3.2.1 What is Stuck-at Fault (SA) concept ?

Stuck-at Fault occurs when a line is permanently stuck to Vdd or ground giving a faulty output. This line may be an input or output to any gate.

When a signal, or gate output, is stuck at a 0 or 1 value, independent of the inputs to the circuit, the signal is said to be “stuck at” and the fault model used to describe this type error is called a “stuck at fault model”. Stuck-at Fault is a functional fault on a Boolean (logic) function implementation.

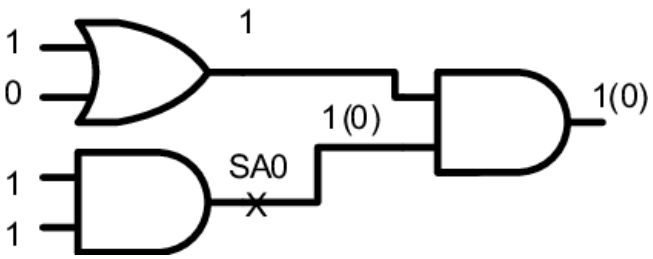

Figure 3.6: Stuck-at Fault Model

3.2.2 What is Transition Delay Fault (TD) concept ?

Defects that cause the faulty timing behavior of a circuit are modeled by delay faults.
The transition fault model captures delay defects that cause a slow-to-rise transition or a slow-to-fall transition at a specific line in the circuit.
Under this fault model, it is assumed that the extra delay caused by a transition fault on a line is large enough so that the delay of every path passing through this line exceeds the clock period.

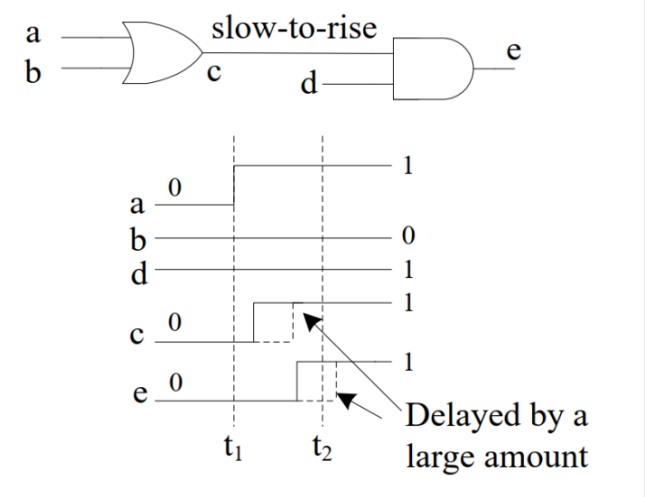

Figure 3.7: TD Fault Model

3.3 Scan Insertion Flow & Test Protocol

3.3.1 DFT Compiler (DC) Scan Insertion Flow ?

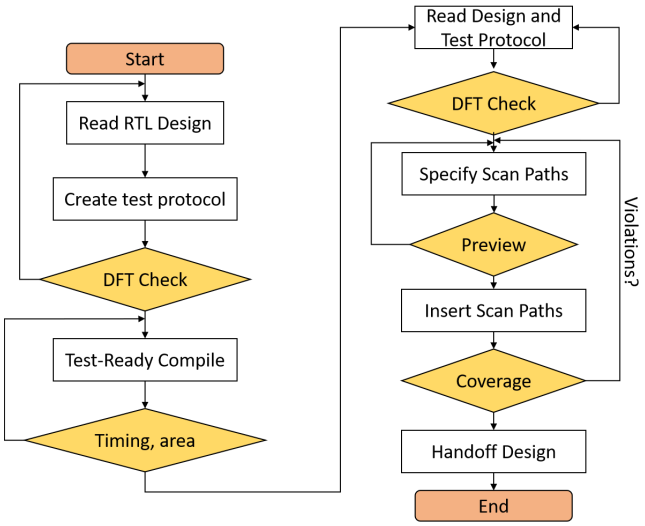

Figure 3.8: DC Scan Insertion Flow for Top-down model

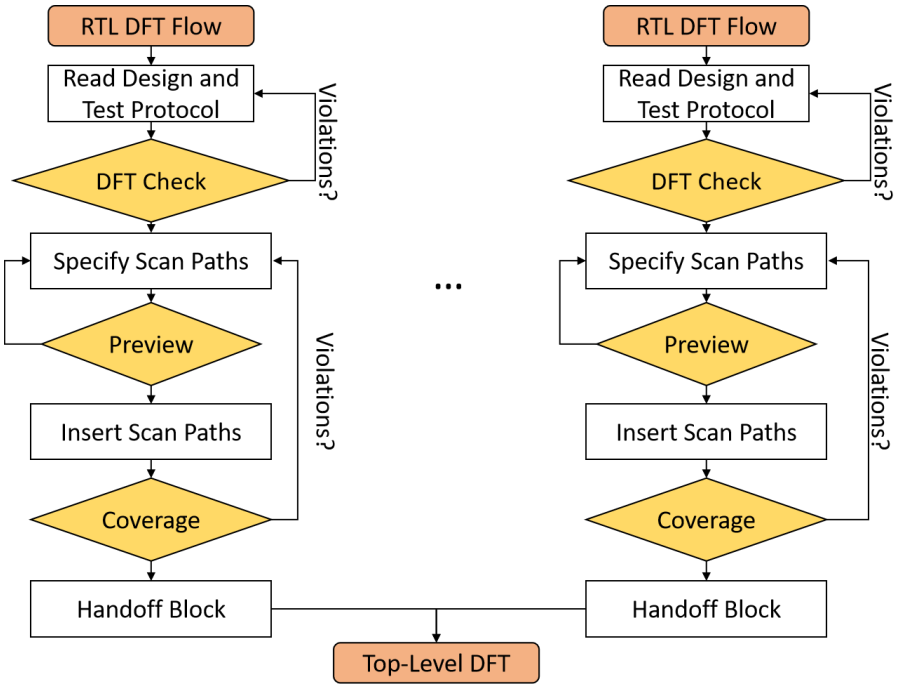

Figure 3.9: DC Scan Insertion Flow for Bottom-up model

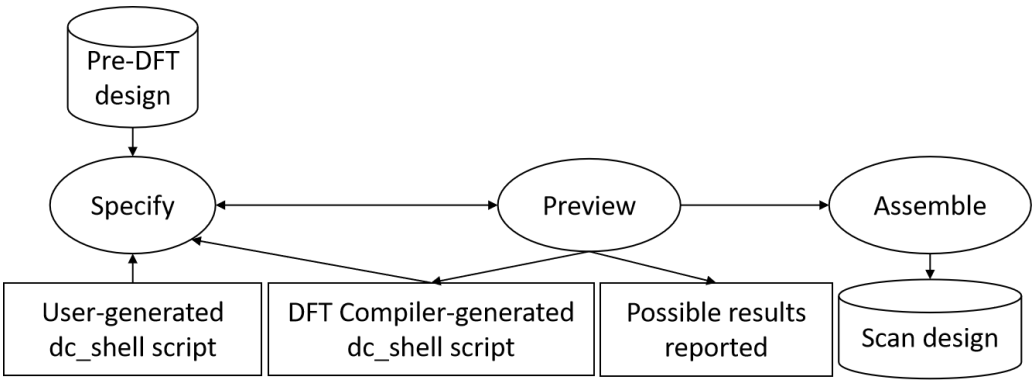

Figure 3.10: Scan Insertion Flow

3.3.2 What is Test Protocol & Steps for creating a Test Protocol ?

A test protocol completely describes the test environment for a design. The test protocol includes:
- The test timing information
- The initialization sequence used to configure the design for scan testing
- The test configuration used to select between scan shift and parallel cycles during pattern application
- The pattern application sequence

A test protocol is the set of specific instructions for scan testing a design.
Steps for creating a Test Protocol:
- Define DFT signals: clocks, sets, resets, scan inputs, scan outputs, scan enables.
- Use create_test_protocol command to create.

3.4 DFT Design Rule Check (DRC)

3.4.1 What is purpose of LDRC on Pre Scan Netlist ?

What is LDRC ?
Pre-DFT test design rule checking (DRC) evaluates unrouted scan design using a set of constraints that apply to the chosen scan type, then generates a set of violations. The design must be modified to get ready for DFT insertion based on violations.
What is Pre Scan Netlist ?
Pre-scan netlist also contains all the gate level info and the connection between these gate, but the flip-flops in this type of netlist are non-scan flip-flops.
Purpose of LDRC on Pre Scan Netlist is a rule so that after scan the circuit can still perform the correct function.

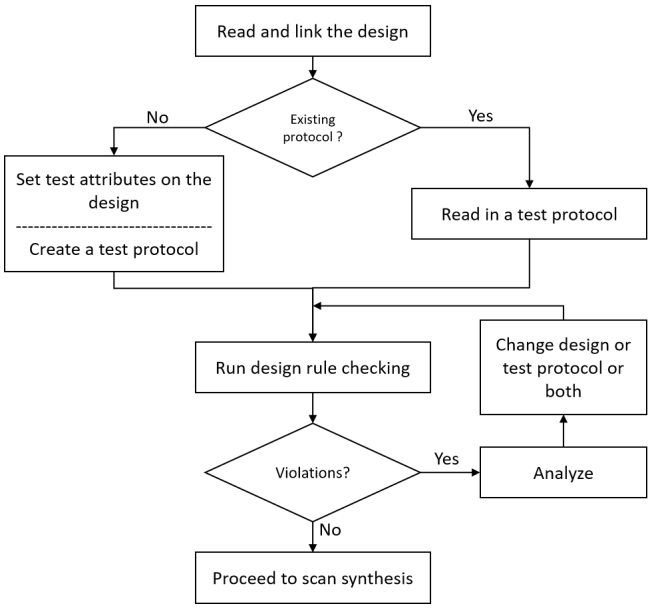

Figure 3.11: Pre DFT Test DRC

3.4.2 What are Types of DRC Runs ?

There are 2 types of DRC Runs that can be performed in a design with command dft_drc:

- Pre-DFT DRC:
  - Runs prior to scan insertion
  - Evaluates unrouted scan design using a set of constraints that apply to the chosen scan type, then generates a set of violations.
  - Based on violations, modify the design.

- Post-DFT DRC:
  - Runs after scan insertion.
  - Perform DRC of the DFT-inserted to ensure that no violations have been introduced into the design by scan insertion process.

3.4.3 What are D-Rule & Why check for DRC violation ?

D-Rules (Category D – DRC Rules) are a category of rules that are only checked during pre-DFT (before scan chains are built).
We check D-rules for:
- Highly-automated structured DFT depends on checking all designs for compliance with a standard set of rules.
- DRC violations can show where to insert ad hoc DFT.
- DRC violations can prevent “scannable” registers from being included in the scan chains.
- Some DRC violations don’t impact scan insertion but may affect test-pattern generation.
There are 4 common methods of correcting DRC violations in DFTC:
- Edit RTL code and resynthesize design with DFT logic.
- Use AutoFix to insert bypass or injection logic.
- Use User Defined Test Point to insert ad hoc test points.
- Edit netlist with create_net and other commands.

3.5 Specify/Preview Scan Configuration and Insertion DFT logic

3.5.1 What Does Specify Scan Architecture Do?

Scan configuration is the specification of global scan properties for the current design.
We use the set_scan_configuration command to specify global scan properties such as:
- Scan style and methodology
- Length and number of scan chains: -chain_count, -max_length, -exact_length, -count_per_domain
- Handling of multiple clocks - Internal and external three-state nets - Bidirectional ports

3.5.2 Why Do Use Lock-up Elements ?

A scan lock-up latch is a retiming sequential cell on a scan path that can address skew problems between adjacent scan cells when clock mixing or clock-edge mixing is enabled.
Lock-up latch is used to reduce the clock skew and follow the hold time constraints. The clock skew occurs mostly in the systems where multiple clocks are used. The clock skew can occur during shift and capture time.
If both scan cells receive a clock edge at the same time, no timing violations occur.
If the CLK2 waveform at FF2 is delayed, perhaps due to higher clock tree latency, a hold violation might result where FF2 incorrectly captures the current cycle's data instead of the previous cycle's data.
A lock-up latch prevents hold violations for scan cells that might capture data using a skewed clock edge. It is a latch cell that is inserted between two scan cells and clocked by the inversion of the previous scan cell's clock. The lock-up latch cell works by holding the previous cycle's scan data while the current cycle's scan data is captured, effectively delaying the output data transition to the next edge of the source clock.

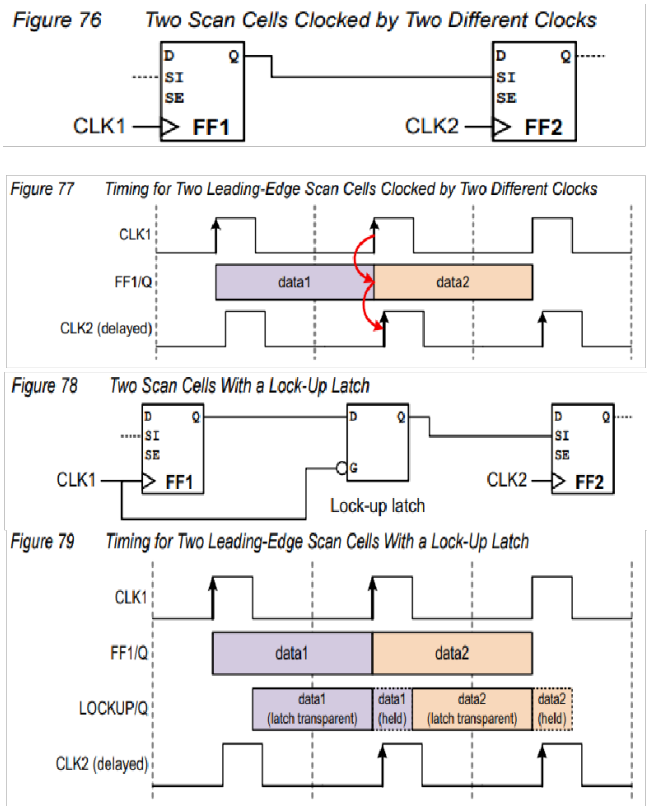

Figure 3.12: Function of Lockup latch

Lock-up latch operation for trailing-edge scan cells is similar to that of leading-edge scan cells, except that the data is held into the next clock cycle.

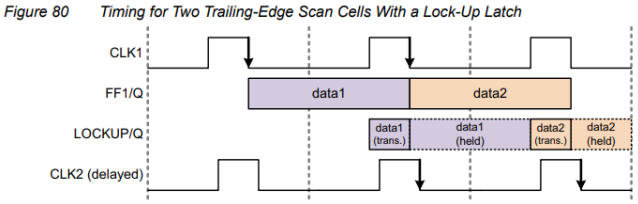

Figure 3.13: Function of Lockup latch (cont.)

3.5.3 What does Scan Preview do ?

Preview additional scan chain information:
- Command: preview_dft –show {…}
- cell – Shows all scan cells and scan segments in each scan chain.
- scan_clocks – Shows scan clock domains along the scan chains.
- scan_signals – Show info about DFT signals and hookup pins associated with each scan chain.
- segments – Show info about scan segments included in scan chains.
- voltages – Shows scan cell operating voltage information along the scan chains.
- power_domains – Shows power domains along the scan chains.
- bidirectionals – Shows information about bidirectional conditioning logic used to enable scan paths.
- tristates – Shows information about all tristate conditioning logic used to prevent tristate contention during scan shift.
- scan_summary – Shows a short summary of the scan chains and their scan clocks.
- all – Show all information about scan chains (equivalent to specifying all keywords except scan_summary).

Preview the DFT design using script commands
- Command: preview_dft –script
- The result is a set of DFT commands that describe the DFT structures to be built.

Preview test mode information:
- It reports the scan chain structures for each test mode.
- It reports the test-mode signals and encodings to be used for test mode selection.

Checks the specification for consistency
Creates a complete specification
Runs AutoFix
Produces a list of test points that are to be inserted into the design, based on currently enabled utilities.

3.5.4 What does Scan Insertion do ?

Scan replacement

Scan replacement the process of remapping nonscan sequential cells to library cells have appropriate test pins for the chosen scan style.

DFT Compiler performs the following scan replacement tasks during the insert_dft command:
- Scan-replaces sequential elements if a scan replacement on the sequential elements was not performed previously, and

the cell does not violate test DRC.
- Converts the scan elements that resulted from a test-ready compile or a previous scan insertion back to nonscan elements if test DRC violations prevent their inclusion in a scan chain.

Scan element allocation and ordering

Allocates scan elements to produce the minimum number of scan chains consistent with clock domain requirements.
Automatically infers existing scan chains both in the current design and in subdesigns.
Does not reroute existing scan chains previously built by the insert_dft command or subdesign scan chains built by the insert_dft command.
In Design Compiler wire load mode, allocates and orders scan elements into scan chains alphanumerically, using the full hierarchical path specification of the scan element name.

DFT Compiler inserts and routes test signals in the following manner

Automatically inserts and routes global test signals to support the specified scan style. (clocks and enable signals)
Allocates ports to carry test signals.
Inserts three-state and bidirectional disabling logic during default scan synthesis.

3.5.5 Why Balanced Scan Chain and How DFT Balances Scan Chains ?

Why balance scan chains ? Always work to create a set of top-level scan chains that is as balanced as you can.
The longest scan chain determines how many ATE cycles are needed for each scan pattern
. Test-cost, test-time saving.

How DFT cab balance scan chains ?

If having multiple clocks in test mode, allow clock mixing in the scan chains.
At the boundary where the clock changes, a lockup latch should be inserted.

3.6 DFTMAX Compression

3.6.1 What is purpose of DFTMAX Compression ?

DFTMAX compression provides synthesis-based scan data compression technology to lower the cost of testing complex design. (DSM design)
DFTMAX compression also provides some beneficial features:
- Test time and test data reduction compared to standard scan.
- Similar ease-of-use as standard scan.
- Simultaneous area, power, timing, physical constraints and test constraint via a synthesis-based implementation.
- Pin-limited test optimizations
- Unknown logic value (X) handling
- Flexible scan channel configurations to support multisite testing and wafer-level burn-in.

3.6.2 DFTMAX Compression Architecture ?

DFTMAX compression architecture includes:
- Compressed scan chains: divided into a larger number of shorter chains from standard scan chains.

- Codec: reduces the amount of test data needed to test the chip.
+ Decompressor: controls the flow of scan data into the scan chains.
+ Compressor: reduces the captured data from the larger number of compressed scan chains.
→ It is a combination of decompressor and compressor wrapped around the scan chains.

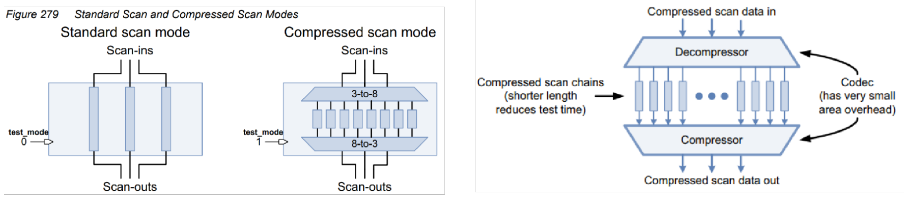

Figure 3.14: Codec Structure

3.6.3 Decompressor Operation

The decompressor outputs are driven by different combinations of scan-in data pins, either directly or through MUXs. One or more scan-in data pins, called load-mode pins, are dedicated to the MUX select signals.
In each shift clock cycle, ATPG can choose load-mode and scan data values that steer these required values into the compressed scan chains.
As the compressed chain count increases, more data steering logic configurations are needed. If the compressed chain count is increased too high, the data steering configurations must repeat, which can reduce the ability of ATPG to steer data into the compressed chains.

3.6.4 Compressor Operation

The compressor outputs are driven by different combinations of compressed scan chains, combined using XOR logic. An incorrect data value (fault) from a compressed scan chain results in a specific signature of incorrect values at the compressor outputs.
The compressor outputs are driven by different combinations of compressed scan chains, combined using XOR logic. An incorrect data value (fault) from a compressed scan chain results in a specific signature of incorrect values at the compressor outputs.

3.6.5 What is a pipeline register and its function ?

Pipeline Register (Pipelined scan data) is a feature provided by the TestMAX DFT tool. In typical scan flows, long wires between the scan chain input and the first flip-flop and between the last flip-flop and the scan chain output can cause delay problems. These delays are reduced by placing pipeline registers at the beginning and end of the scan chains. They divide the long routes between the scan chain terminals into smaller wires between the registers and therefore help reduce the path delay Pipeline registers inserted at the scan-in and scan-out ports are called head and tail pipeline registers respectively.

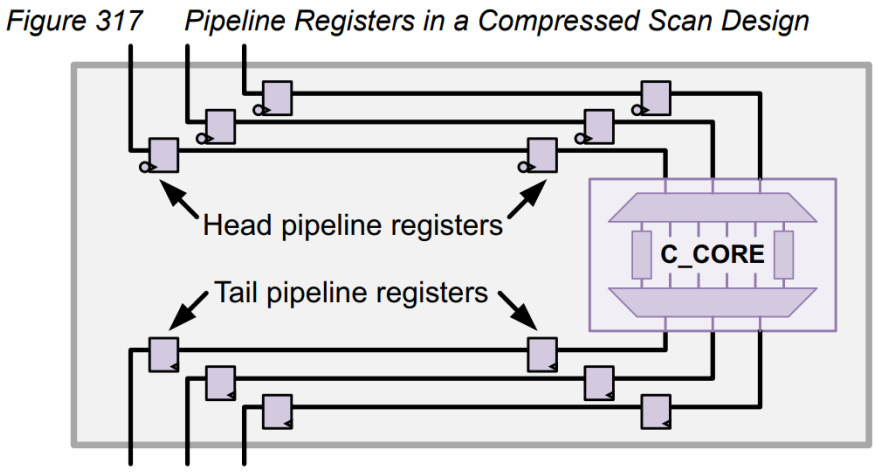

Figure 3.15: Pipelined registers in a Compressed scan design

3.6.6 How to reduce power consumption in DFTMAX design ?

To reduce the power consumption of designs with scan compression, use the following features:
- Reduce compressor power when codec is inactive
- Reduce scan shift power using shift power groups

Reduce compressor power when codec is inactive

In a compressed scan architecture, an XOR compression tree is needed only during scan shifting in codec's compressed scan mode. At other times, the compression logic is not needed, but it will still toggle when the tail FFs of the compressed chains toggle.

→ A particular concern during mission mode, when the flip-flops are clocked at their full operating frequency.

→ The tool can insert gating at the inputs to the XOR compression tree to eliminate this toggling activity and reduce power consumption in other modes of operation.

Example: When enable XOR compressor gating, the tool inserts gating at the inputs to the XOR compression tree. Every compressor input is AND-gated with an active-low pwr_save_n signal before going to the XOR tree. The pwr_save_n gating signal is generated by combining the scan-enable signal and the test control module (TCM) signal for the codec's compression mode.

Reduce scan shift power using shift power groups

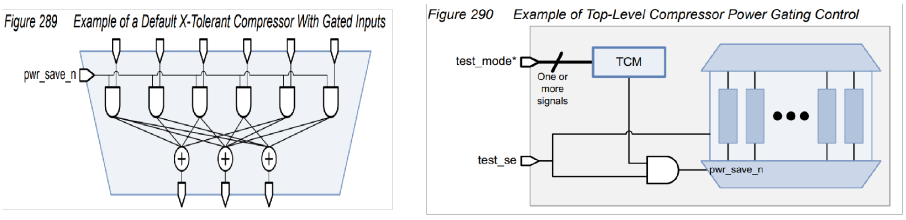

Figure 3.16: Reduce compressor power when codec is inactive

During scan shift, there is significant toggle activity in the scan chains. At high scan shift frequencies, this can result in higher-than-desired shift power consumption. This feature inserts AND gates at the decompressor outputs before each compressed scan chain. The chains are gated in groups that are controlled by a shift power control (SPC) chain. The SPC chain is an external (uncompressed) chain outside the DFTMAX codec. When scan-in completes, the SPC registers

contain the group mask values for the next pattern. The de-asserted scan-enable signal, test_se, latches these bits into shadow latches that retain the mask values for scan-in of the next pattern.

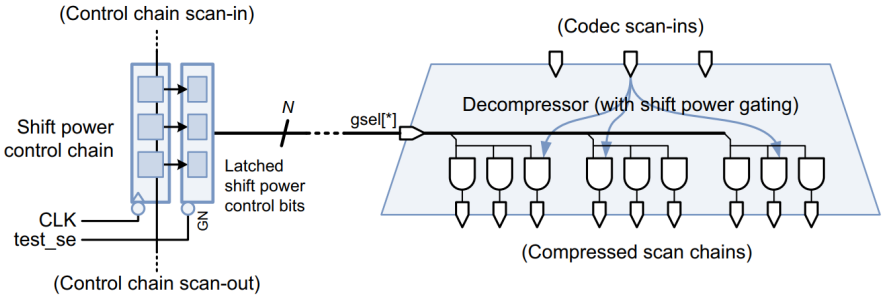

Figure 3.17: Reduce scan shift power using shift power groups

SPC chains must be external chains because a compressed SPC chain would gate itself, preventing it from reliably loading in each pattern. The shift power logic also includes a hardware disable signal that, when asserted, disables the shift power logic by enabling all compressed chains. This signal must be de-asserted or asserted prior to DRC, depending on whether the shift power groups feature is enabled in TestMAX ATPG or not, respectively

Figure 292 Shift Power Disabling Logic

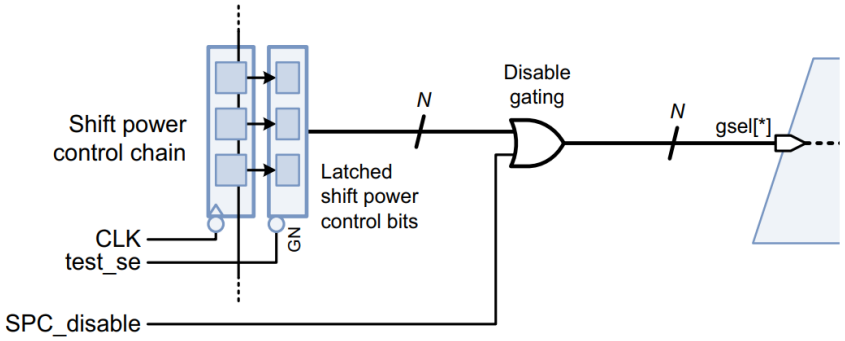

Figure 3.18: Reduce scan shift power using shift power groups

3.6.7 What is The High X-Tolerance Architecture ?

DFTMAX compression provides scan compression using only combinational circuitry. → Can achieve moderate to high compression, minimize the additional cost for DFT implementation. Scan compression can be applied to designs with a large number of X values, whose sources are either (logic-induced) or dynamic (constraint-induced). The high X-tolerance scan compression solution, provided by DFTMAX compression, meets the challenge of coverage loss by implementing new logic that selectively masks circuit response on a per-shift basis. It provides 100 percent X-tolerance without introducing sequential circuitry.

The high X-tolerance architecture provides the following observe modes:

- A full (unmasked) observe mode, which is equivalent to the default X-tolerance mode.
- Additional X-tolerance (masked) observe modes, which can mask X values from selected chains before they reach the XOR compressor.

An existing ScanDataIn port provides the mask enable signal. A mask enable value of zero selects the full observe mode.

A mask enable value of one plus combinations of the mask mode signals selects additional X-tolerance observe modes for unload. The high X-tolerance architecture introduces a combinational path between the scan input and scan output ports. The path travels from the scan input ports, through the decompression MUX mask signal generation logic, through the X-blocking and XOR compactor circuits, then through the scan output ports. → Long routes → use Pipeline Registers.

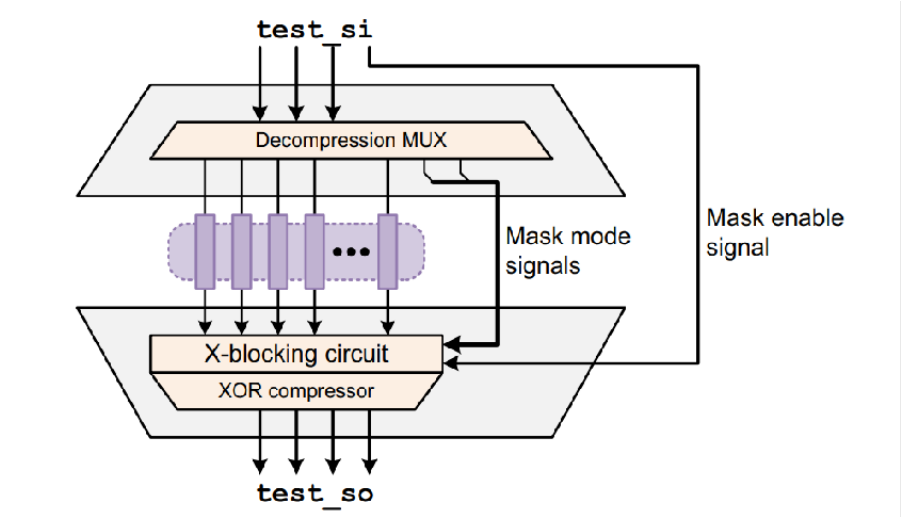

Figure 3.19: High X-Tolerance Architecture

3.7 Wrapping Cores Concept

When a DFT-inserted core is integrated into a top-level design, the core-level scan structures are integrated into the top-level scan structures. To test the core-level logic separately from the top-level logic, the core must be a wrapped core. There are 2 core wrapping flows:

- Simple core wrapping flow: provide basic core wrapping capabilities
- Maximized reuse core wrapping flow: minimize area and timing impact of core wrapping by reusing more existing functional registers.

A wrapped core has a wrapper chain: allows the core to be isolated from the surrounding logic. A wrapper chain is composed of wrapper cells inserted between the I/O ports and the core logic of the design.

A wrapper cell consists of a scan cell and MUX logic. It is designed to provide controllability and observability to the circuit components that need to be tested. Wrapper cell can transparently pass the I/O signal through, or it captures values at its input and/or launch values at its outputs. Wrapper chains are shift chains, which allow known values to be scanned into the wrapper cells and captured values to be scanned out.

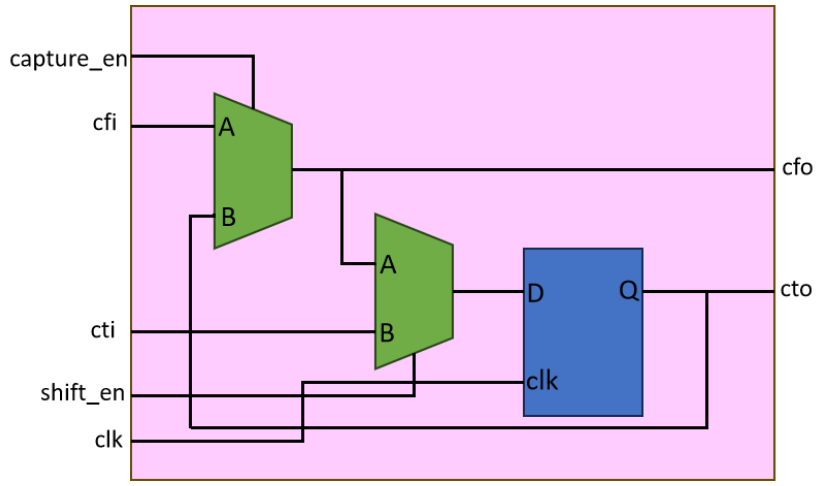

Figure 3.20: Wrapper Cell Structure

3.8 Formality

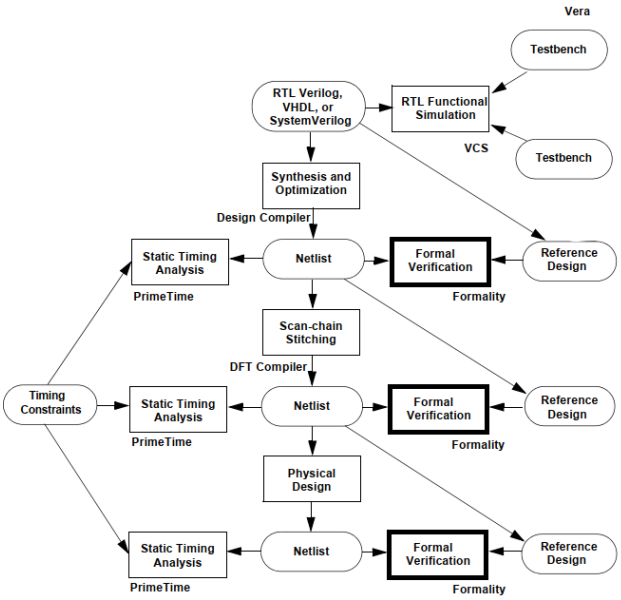

Figure 3.21: ASIC Verification Flow Using Formality

The purpose of Formality is to detect unexpected differences that might have been introduced into a design during development. It uses a formal verification comparison engine to prove or disprove the equivalence of two given designs and presents any differences for follow-on detailed analysis.

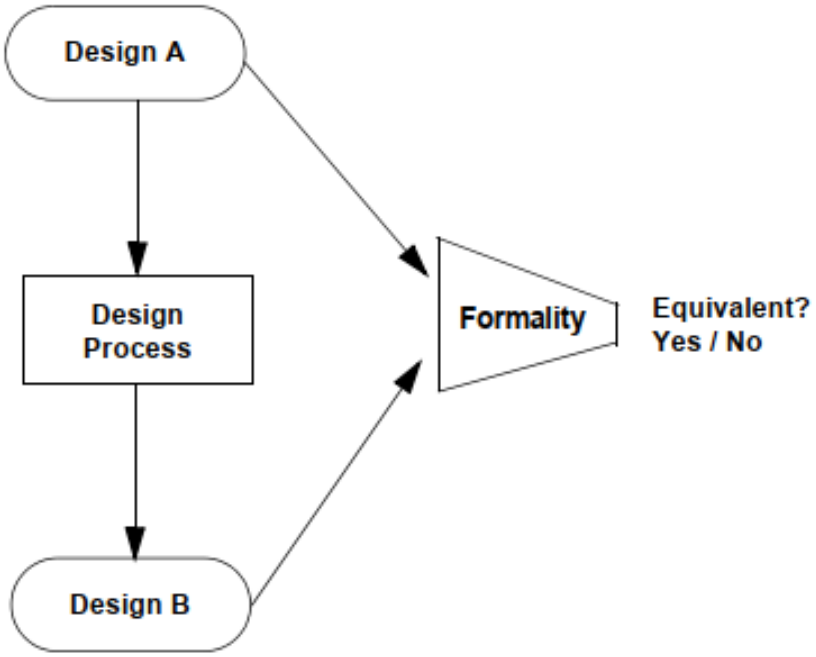

Figure 3.22: ASIC Verification Flow Using Formality

3.9 Automatic Test Pattern Generation (ATPG)

For scan designs, ATPG tools generate input stimulus, for the primary inputs and pseudo-primary inputs, and expected responses, for the primary outputs and pseudo-primary outputs. The set of input stimulus and output response that includes primary inputs, primary outputs, pseudo-primary inputs, and pseudo-primary outputs is called a test patterndefinitionst patterntest pattern orscan pattern scan pattern. **TetraMAX Design Rule Check (DRC)** During DRC, TestMAX ATPG performs a set of checks to ensure that the scan structure is correct and to determine how to use the scan structure for test generation and fault simulation. These checks include ensuring that the scan chains operate properly, identifying scan cells, identifying nonscan cell behavior, and ensuring that clocks obey the required rules.

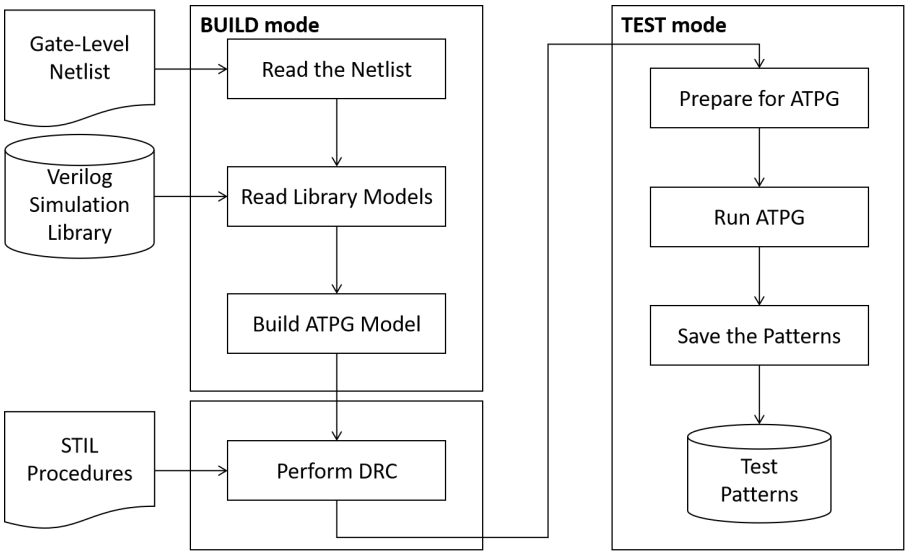

Figure 3.23: Basic ATPG design flow

3.10 On-Chip Clocking Support

On-Chip Clocking (OCC): Complex designs often have many different high-frequency clock domains, and the requirement to deliver a precise launch and capture clock. One common alternative for at-speed testing is to leverage existing on-chip clock generation circuitry.

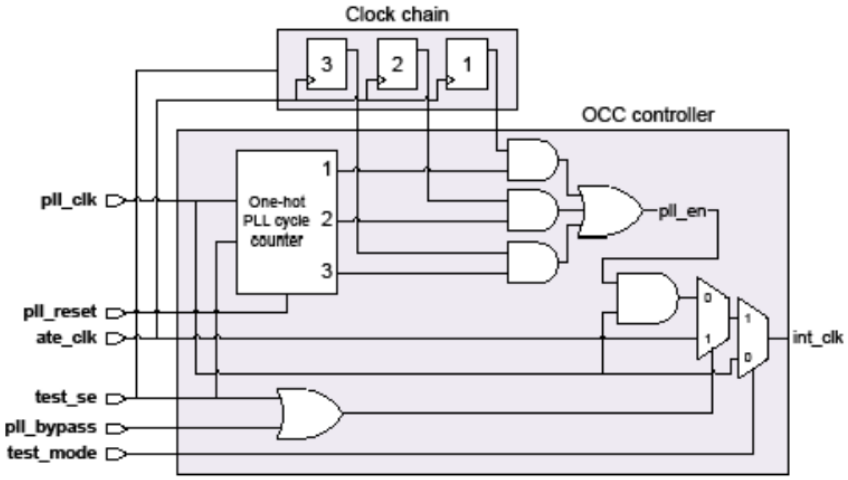

Figure 3.24: OCC Structure

Clock Type Definitions

- Reference clock – The frequency reference to the PLL. Free-running or the PLL may lose synchronization. Reference clocks can not route directly to the clock inputs of FF's.
- PLL clock – The output of the PLL. A free-running source that also runs at a constant frequency which may or may not be the same as it's respective reference clock.
- ATE clock – Shifts the scan chain, ATE clocks are only used and route to the respective OCC controller, they can not route directly to clock inputs of FF's.

- Internal clock – The OCC controller is responsible for gating and selecting between the PLL and ATE clocks, thus creating the internal clock signal to satisfy ATPG requirements.
- External clock – A primary clock input of a design that directly clocks flip-flops through the combinational logic, without the use of a PLL clock. The period for this clock is determined by the test_default_period variable.

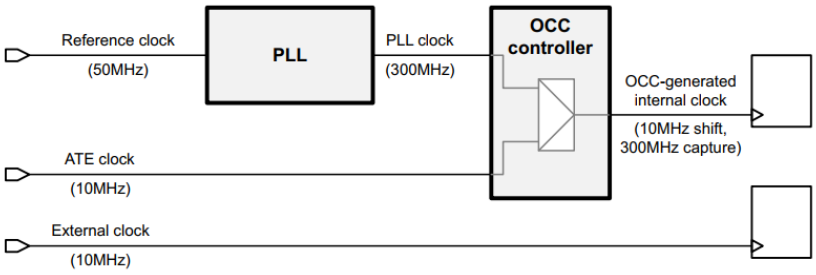

Figure 3.25: Clock Types

The OCC Controller inserted between PLL and DFT design.

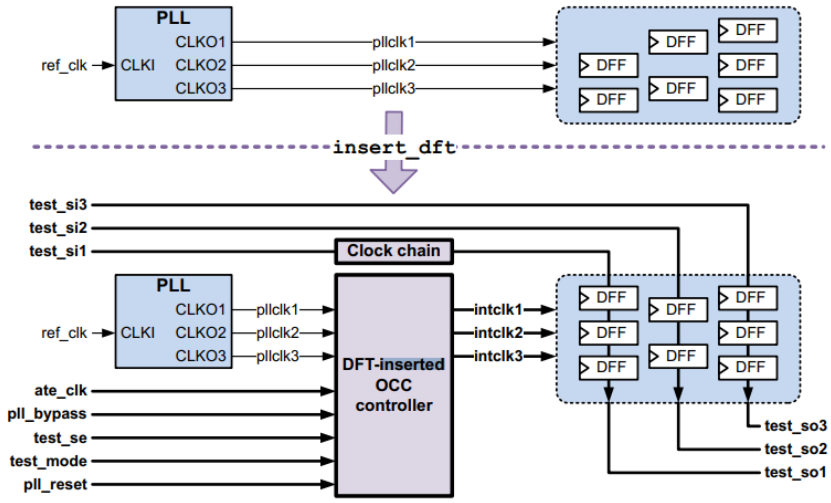

Figure 3.26: OCC's placement

Chapter 4

Achievement

4.1 Knowledge

- Understand the Soc design process, the stages in IC design and DFT
- Learn and run specialized IC design tools, DFT Compiler tool, practice basic work in DFT design.
- Understanding how to work on linux environment.
- Learn more about languages such as: Tcl, Cshell, etc.

4.2 Skills

- Skills to work in a team, communicate and learn with team members and in the company.
- Skills to plan to complete assigned work as required.
- Self-study skills new knowledge and reading English documents.
- Skills to write reports according to business standards and present reports.

4.3 Accumulated experience

- A practical working process in a company environment.
- Communication method so that mentors and team members know the progress of the assigned work.
- Aggressive and attitude serious work.
- How to manage time effectively.