

## UNIT - 6

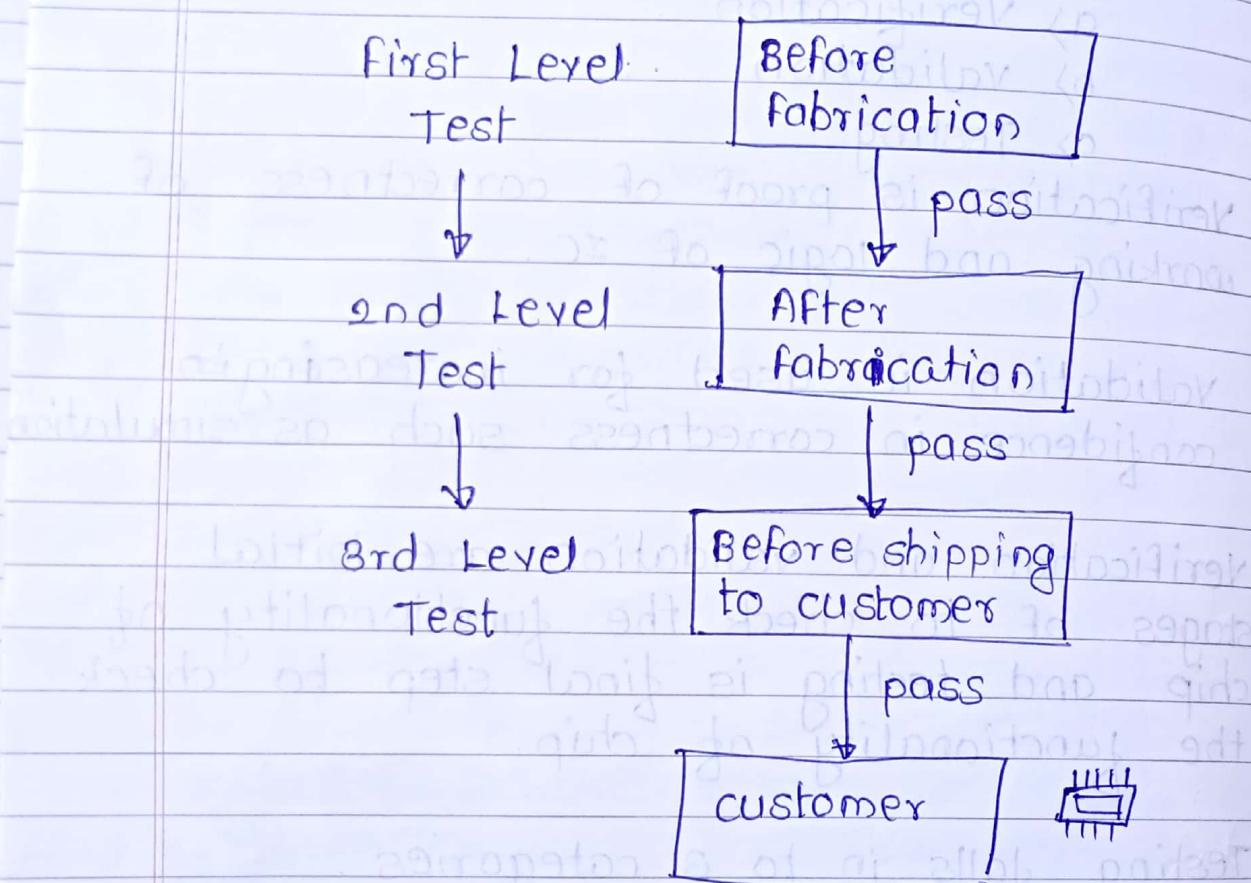
### Testability

#### \* Design for Testability :-

- Testability means checking whether manufactured chip is working properly or not.
- This can be done using
  - a) Verification
  - b) Validation
  - c) Testing
- Verification is proof of correctness of working and logic of IC.
- Validation is used for increasing confidence in correctness such as simulation.
- Verification and validation are initial stages of to check the functionality of chip and testing is final step to check the functionality of chip.
- Testing falls in to 3 categories.
  - ① first test verifies that the chip performs intended function. This test is run out before the tapeout to verify the functionality of chip.
  - ② second test is performed after fabrication process to check functionality of chip.

(3) Third set of test verify that every transistor, gate, and storage element in the chip functions correctly. These tests are conducted before shipping to customer and also called as manufacturing test.

### Test Level in chip process



- Design for testability is used to indicate that we need to build fault models to test the chip which comes off manufacturing line, called as manufacturing test.
- By detecting the malfunctioning earlier,

manufacturing cost can be kept low.

- Testing on the chip can be done at the following level.
  - a) Wafer level testing
  - b) Packaged chip level
  - c) Board level
  - d) System level
  - e) field level.

continued-

Dia fig.a.

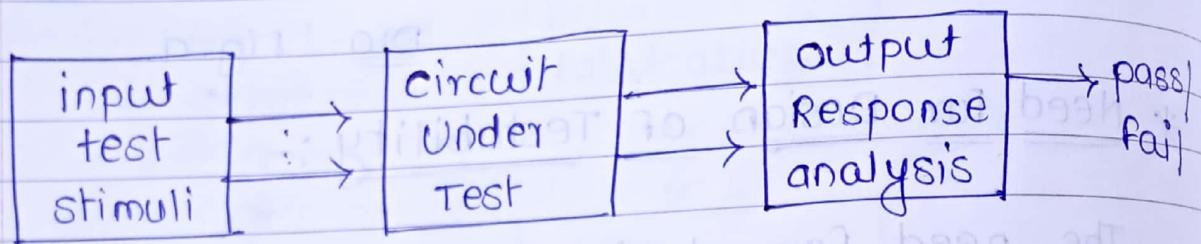
### \* Need for Design of Testability :-

The need for design of testability for chip are given as -

- ① It verify the intended functionality of chip.
- ② If the faults are detected earlier, then manufacturing cost can be reduced.
- ③ Detecting fault at earlier stage reduces time of debugging at chip and system level.
- ④ Assures quality of product ie. chip.
- ⑤ Quality and economy are two main benefits of testing.
- ⑥ The purpose of testing is to resist circulation of bad product to user.

- Testing typically consists of -

- ① Applying set of test stimuli to inputs of circuit under test &
- ② Analyzing output, if incorrect or fails then circuit under test is said to be faulty and if it is correct then circuit is said to be fault free.



- Following dia. shows testing during the VLSI development.

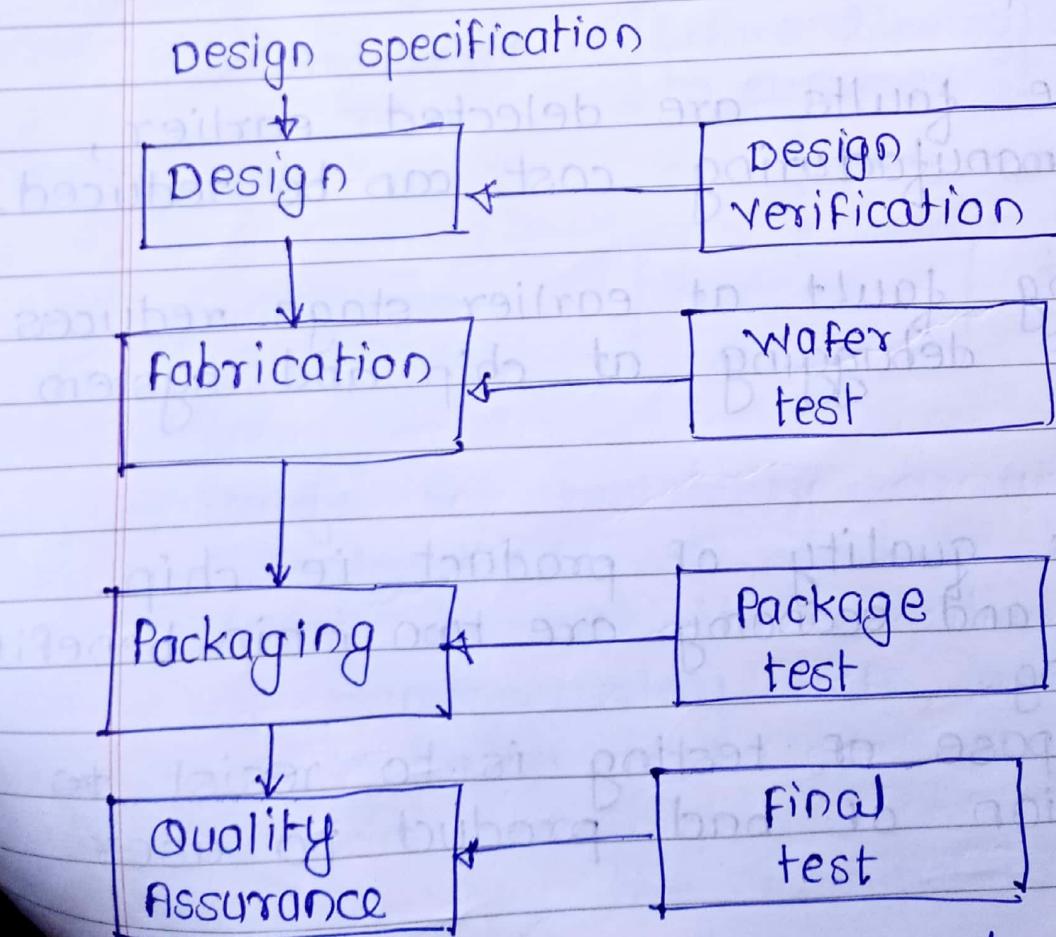
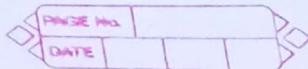


fig. a. testing levels



## \* Types of faults :-

Normally in VLSI circuits following types of faults are occurred.

- a) Physical faults
- b) Electrical faults
- c) Logical faults.

### a) Physical faults :-

These faults are caused by physical structure elements, material, doping, due to fabrication process.

Some of the common faults are

given as -

- ① Defects in silicon substrate ie. size of area of pmos and nmos may differ.
- ② Photolithographic defects
- ③ Mask contamination or scratches
- ④ Excessive etching
- ⑤ Oxide defects

Due to above mentioned physical defects electrical and logical defects are generated.

### (b) Electrical Defects :-

Various electrical defects in VLSI are -

- ① Electrical short ckt
- ② Electrical open ckt
- ③ Change in threshold v<sub>tg</sub> of mos
- ④ Transistor stuck on or stuck open

- ⑤ Excessive steady state current
- ⑥ Resistive short and open

Due to above electrical faults, logical faults are generated.

### ② Logical Defects :-

Logical defects means change in logic.

- ① Stuck at 0
- ② Stuck at 1
- ③ Delay faults
- ④ Bridging faults
  - a) AND bridging
  - b) OR bridging

### \* Fault Models :-

⇒ To deal with existence of good or bad part, it is necessary to propose fault model.

⇒ Fault model propose how faults occur and their impact on circuits.

⇒ Most common fault models are

- a) Stuck at 0 Model } stuck at
- b) Stuck at 1 Model } Models
- c) Short ckt / open ckt model.

## A] stuck at fault model \*

stuck at faults occurs due to metal to metal short or gate oxide short, ie. nmos gate to ground or pmos gate to VDD.

There are two types of stuck at faults where logic sticks to 0 or 1.

- a) stuck at 0 fault
- b) stuck at 1 fault

### a) stuck at 0 Model :-

In this input of gate is shorted to ground.

consider the following circuit in which input terminal of AND1 is shorted to ground. Because of this the output changes to 0 but actual output should be 1.

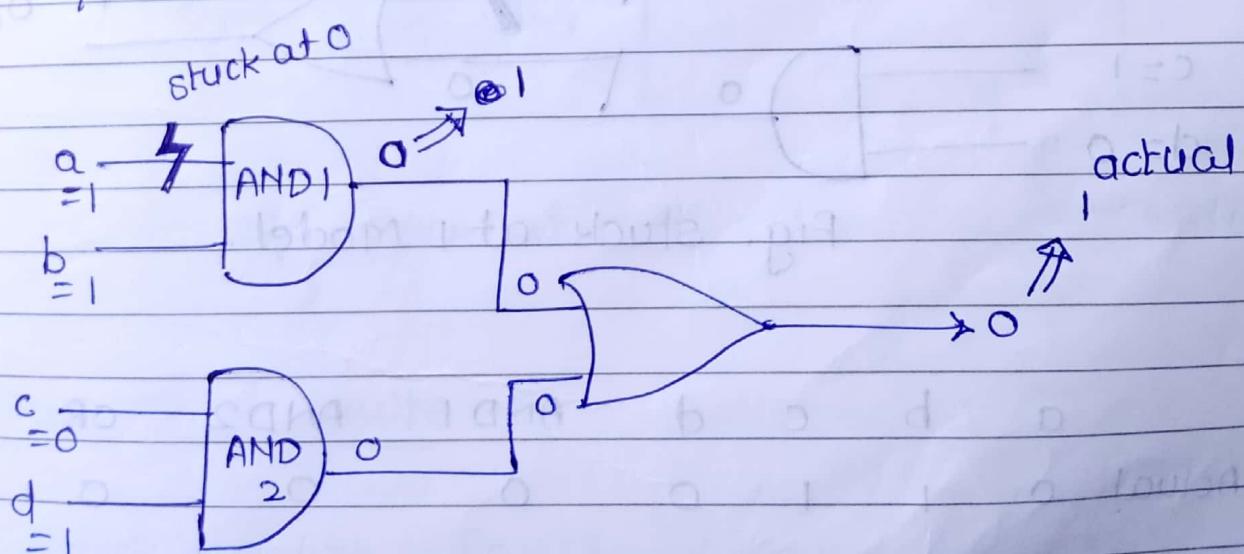


Fig. stuck at 0 fault

	a	b	c	d	AND1	AND2	OR
Actual	1	1	0	1	1	0	1
due to G <sub>a</sub> =0	0	1	0	1	0	0	0 $\Rightarrow$ Fault.

### b) stuck at 1 fault Model :-

If the input of any gate is shorted to positive power supply, stuck at 1 faults appears.

Consider the circuit in which the input terminal of OR gate is shorted to Vdd ie logic 1.

Due to this, for a given set of input the output is different.

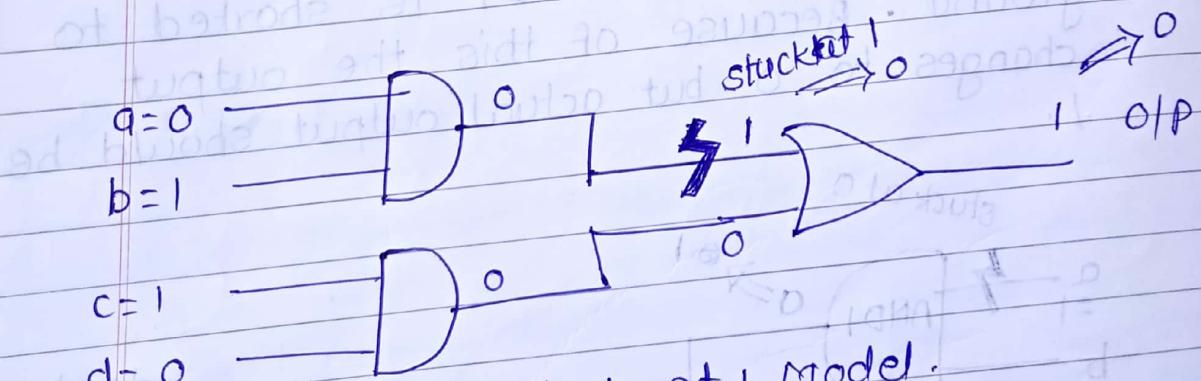


Fig. stuck at 1 Model.

	a	b	c	d	AND1	AND2	OR
Actual	0	1	1	0	0	0	0
Faulty	0	1	1	0	0	0	1 $\Rightarrow$ Fault.

## \* Stuck open ckt faults :-

- stuck open faults creates an unintended high impedance state on output node.
- stuck open faults occurs due to either an incomplete contact of source to drain node.
- stuck open faults may also occurs due to large separation of drain and source diffusion from the gate.
- This fault causes permanent turn off of transistor regardless of input value.

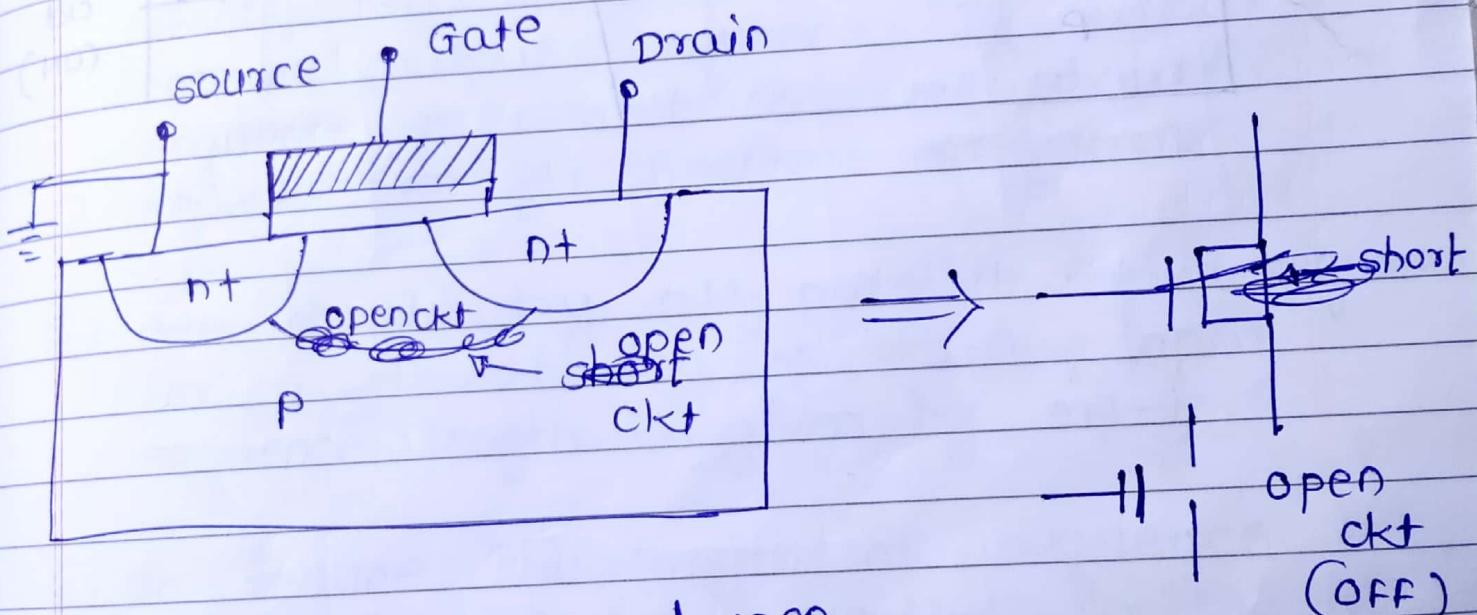


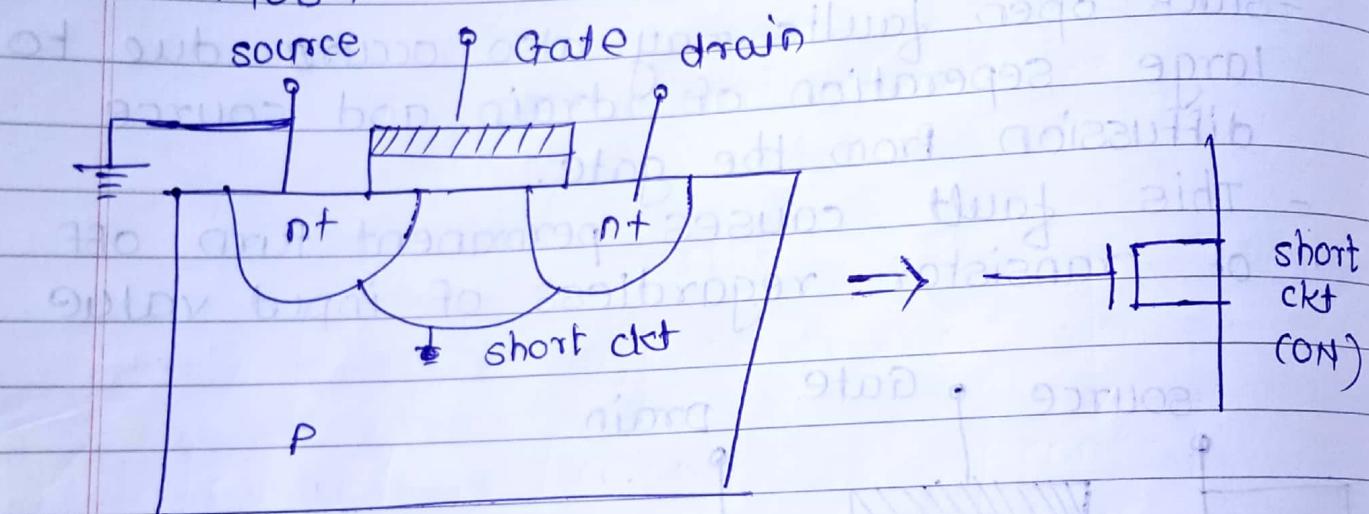
Fig. Stuck at open

## \* Stuck short faults :-

Stuck short fault occurs due to drain and source region short ckt. Stuck short may occur due to metal to metal short of drain and source.

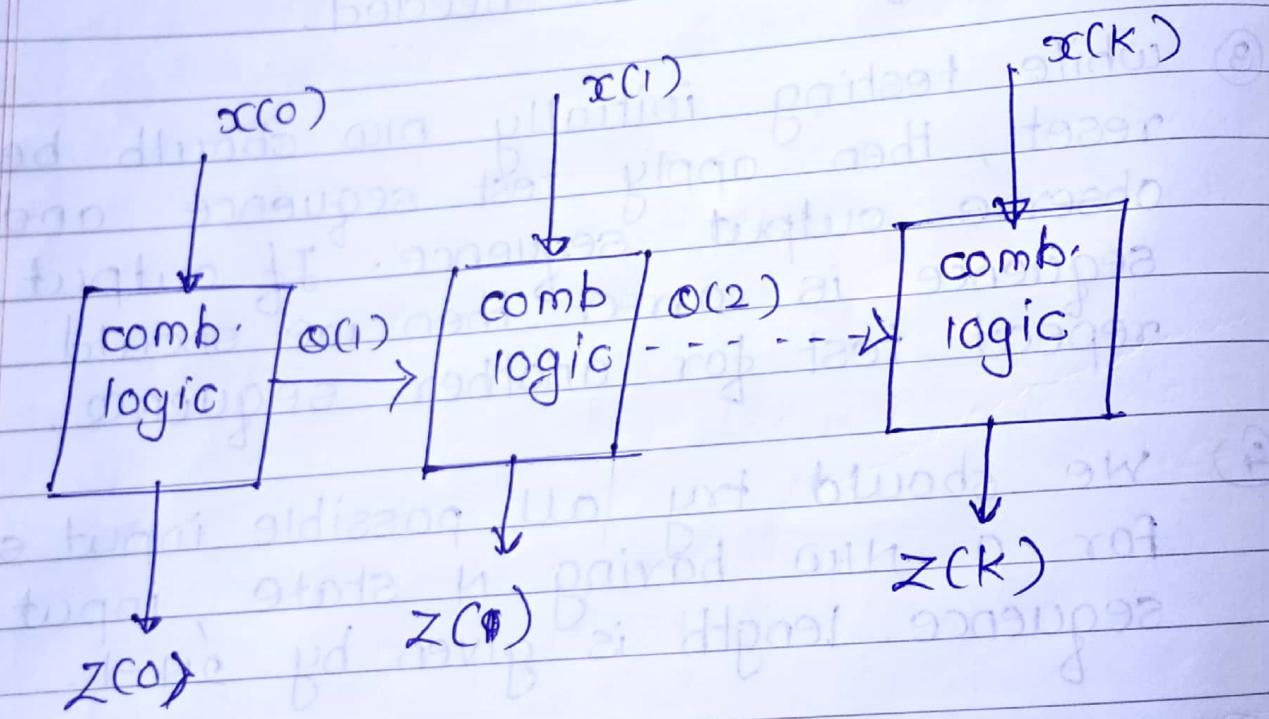
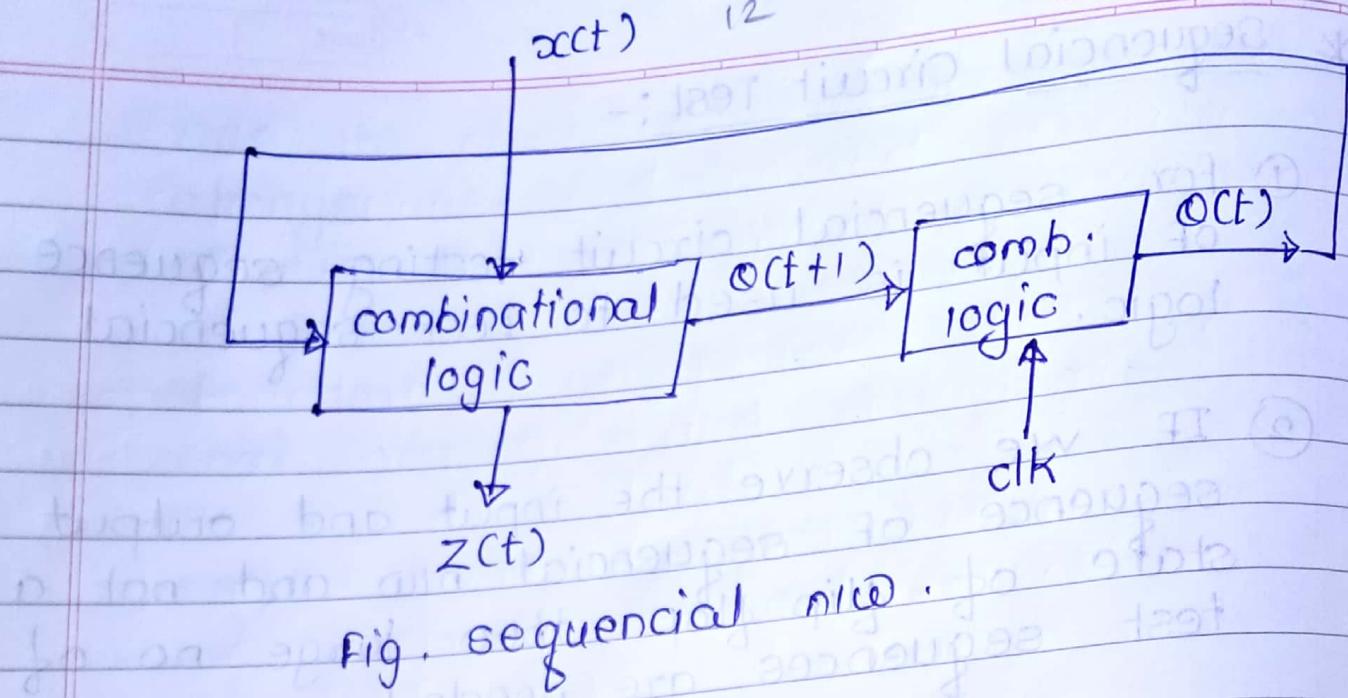
Due to stuck short fault, transistor always remains in ON state regardless of input logic.

Following fig. shows short circuit of drain and source region in MOS which causes permanent turn ON of MOS.



## \* sequential circuit Test :-

- ① for sequential circuit testing sequence of input is used to test sequential logic.
- ② If we observe the input and output sequence of sequential n/w and not a state of flip flop, then large no of test sequences are needed.
- ③ while testing initially n/w should be reset, then apply test sequence and observe output sequence. If output sequence is correct then we should repeat test for another sequence.
- ④ We should try all possible input sequences for a n/w having N state, input sequence length is given by  $2^{N-1}$ .
- ⑤ To reduce the length of sequence, the sequential ckt is converted in to the iterative network.
- ⑥ Iterative n/w is combinational n/w. ∴ sequence length for combinational ckt is reduced.
- ⑦ sequential and iterative n/w are given as,



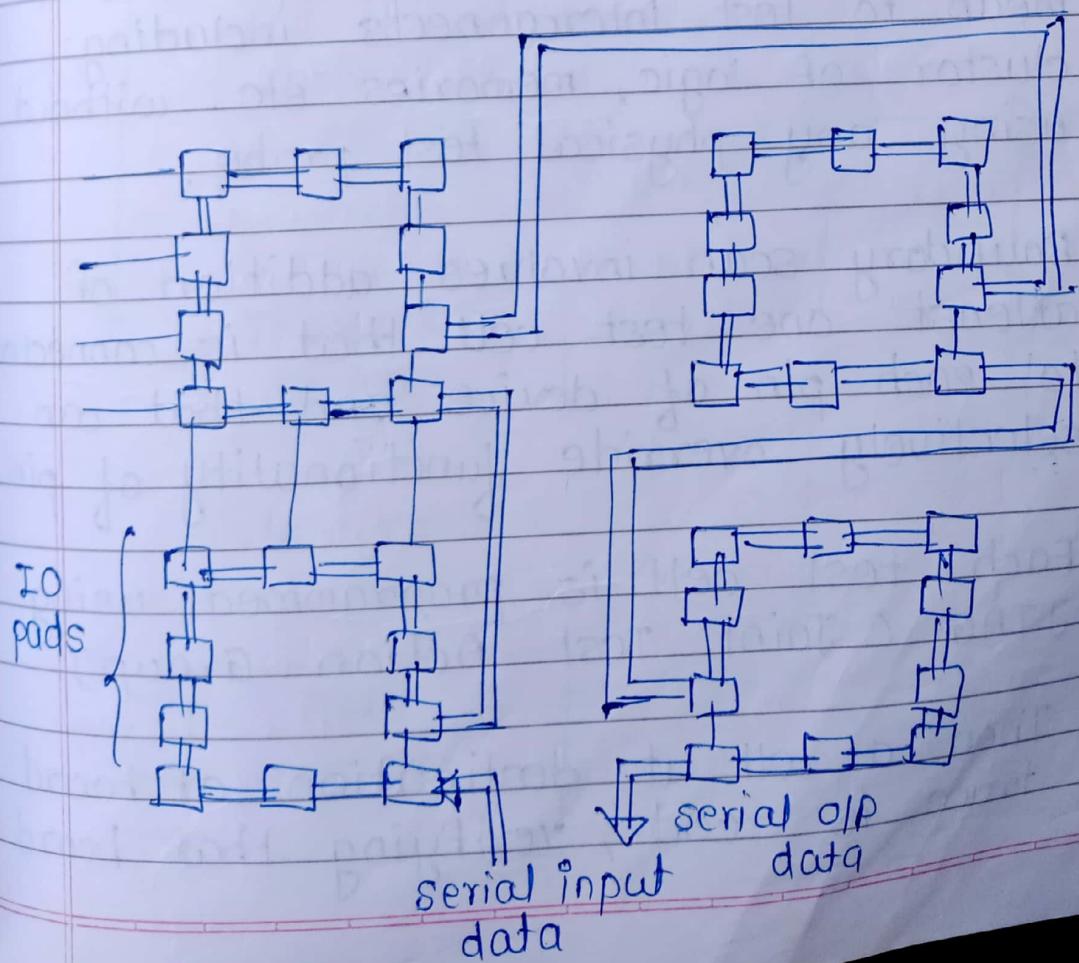
### \* Boundary Scan :-

→ Boundary scan is method of testing the interconnects or subblocks inside an IC.

→ Boundary scan is also widely used as debugging method to watch integrated ckt pin status, measure voltages or analyze subblock inside an integrated ckt.

→ Boundary scan provides a standardize path ie. serial scan path through IO pins of IC.

→ Boundary scan is an IEEE 1149 standard  
The architecture of IEEE 1149 Boundary scan is shown as,



- ⇒ IC to be tested can be connected in series and parallel connections to enable the testing of complete board.
- ⇒ Boundary scan performs following fun:
  - a) Sampling and setting chip input and outputs
  - b) connectivity test between component
  - c) distribution and collection of self test and built in self test result.
- ⇒ Boundary scan Description Language (BSDL) describes the logic content of IEEE 1149.
- ⇒ Boundary scan architecture provides a mean to test interconnects including cluster of logic, memories etc. without using any physical test probe.
- ⇒ Boundary scan involves addition of atleast one test cell that is connected to each pin of device and that can selectively override functionality of pin.
- ⇒ Each test cell is programmed using JTAG (Joint Test Action Group)
- ⇒ Then a cell at destination of board trace is read, verifying that board

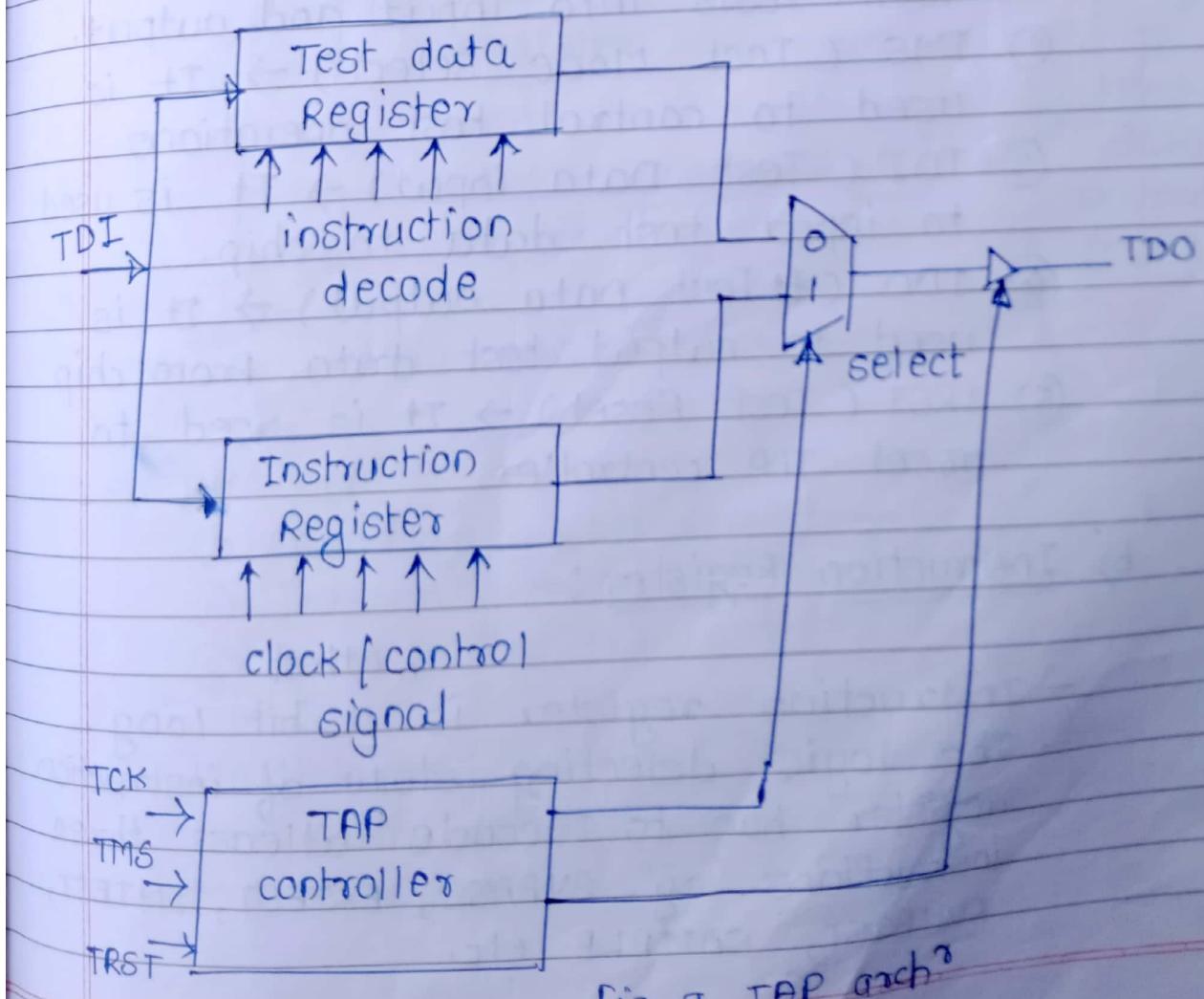
trace properly connects two pins.

→ If the trace is shorted or open circuited, correct signal value does not show at destination pin indicating fault.

### \* Test Access Port (TAP)

→ TAP is interface for boundary scan logic which is included on IC.

→ TAP architecture is shown in fig.a.



→ TAP architecture consists of following components.

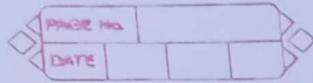
- TAP controller
- Instruction Register
- Test Data Register
- Boundary Scan Registers.

#### a) TAP controller :-

- TAP controller is 16 state finite state machine which proceed from state to state depending upon TCK and TMS signal.
- TAP controller has 5 connections -
  - ① TCK (Test clock input) → used to clock tests into input and output.
  - ② TMS (Test Mode select) → It is used to control test operations
  - ③ TDI (Test Data input) → It is used to input test data to chip.
  - ④ TDO (Test Data output) → It is used to output test data from chip.
  - ⑤ TRST (Test Reset) → It is used to reset TAP controller.

#### b) Instruction Register :-

- Instruction register is 2 bit long.
- The logic detecting state of instruction register has to decode atleast three instructions. e.g. BYPASS, EXTEST, INTEST, RUNBIST, SAMPLE etc.



### c) Test Data Register (DR) :-

⇒ DR are used to set the inputs of modules to be tested and to collect the results of running test.

⇒ The generalized structure of data register is given as,

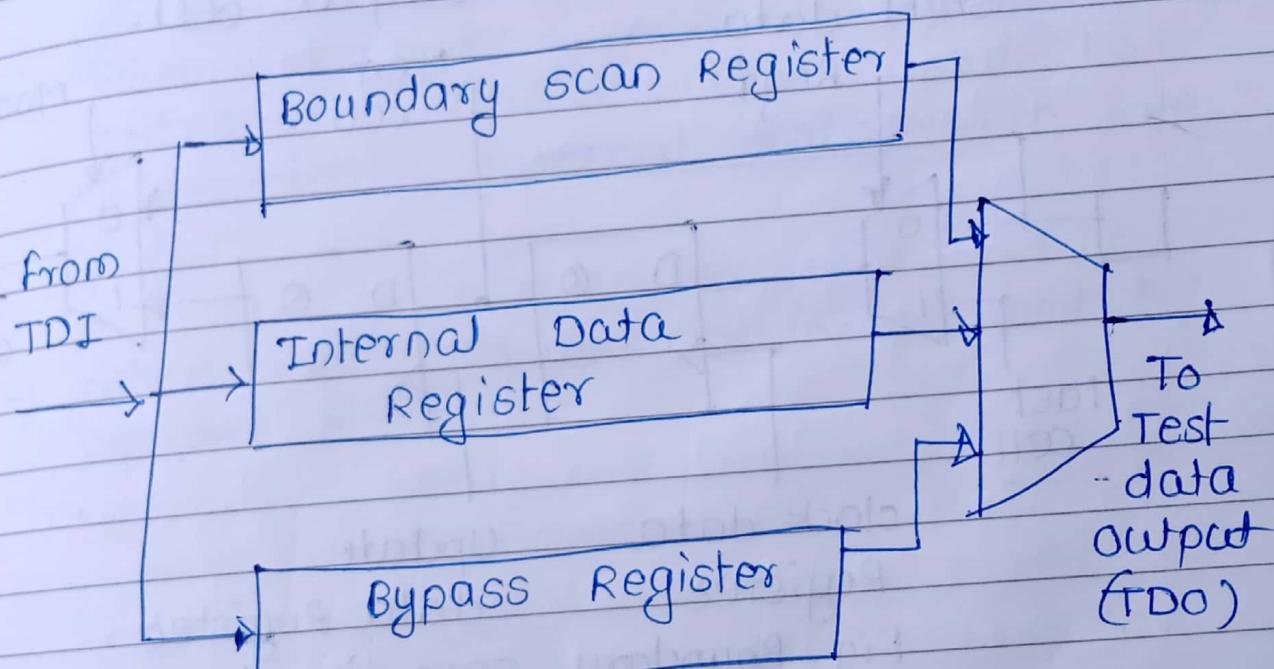


Fig. TAP data register

⇒ DR consists of,-

- ① Boundary scan Register
- ② Internal Data Register
- ③ Bypass Register

⇒ multiplexer is used to select the particular data register to be routed to test data output.

### d) Boundary scan Registers :-

⇒ Boundary scan registers are used to test circuit board interconnection, external components and state of chips digital input and outputs to be sampled.

⇒ Following fig. shows typical boundary scan input and output cell.

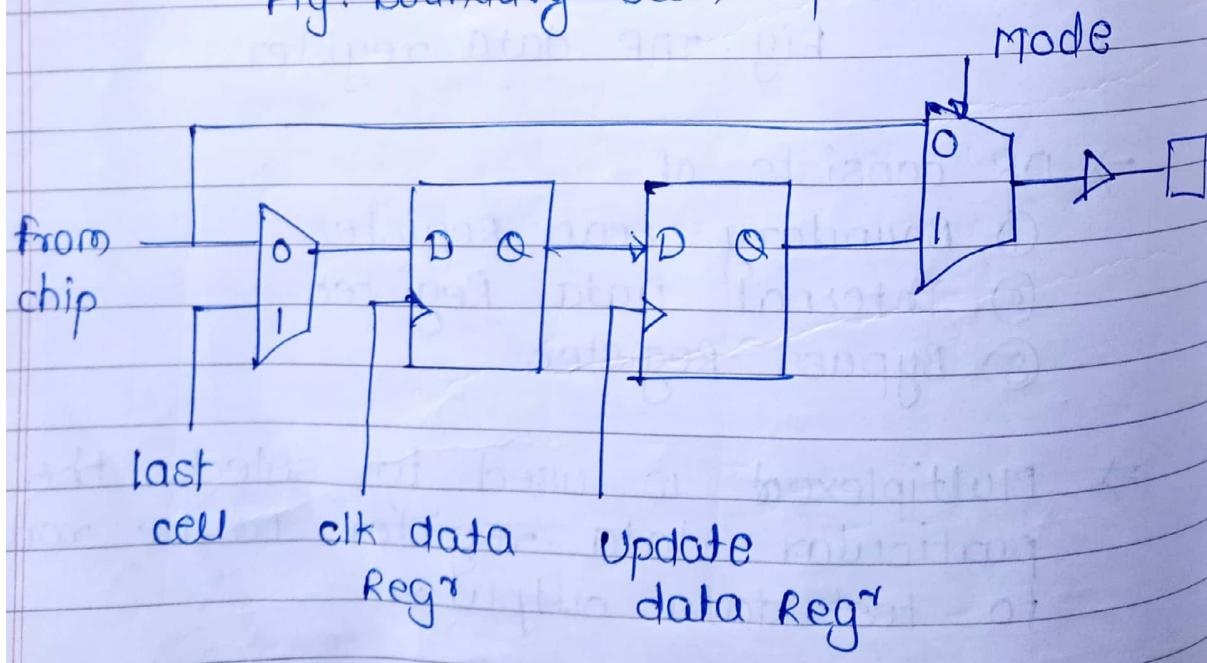
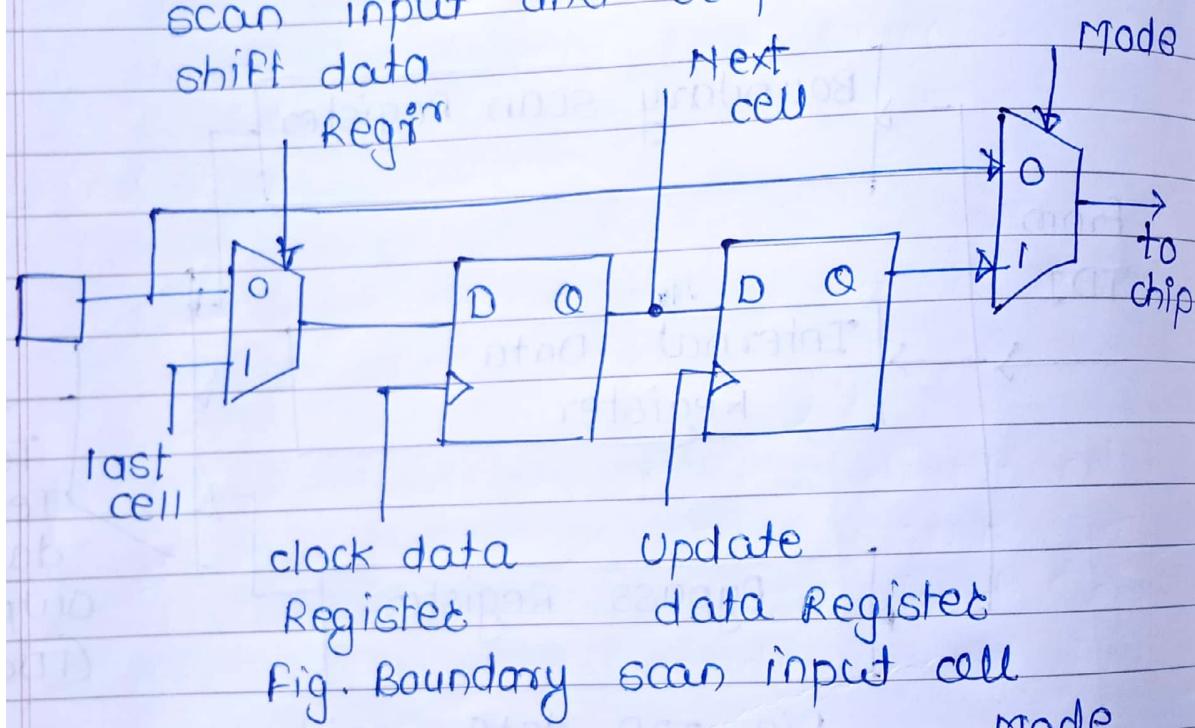
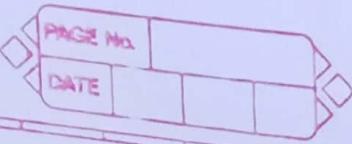


Fig. boundary scan output cell



### for input cell

- 19
- ⇒ When mode is 1, the data bit may be directed to internal circuitry in INTEST or RUNBIST mode.
  - ⇒ When mode is 0, the cell is in EXTEST or SAMPLE/RELOAD mode.

### for output cell

- ⇒ When mode is 1, cell is in EXTEST or INTEST or RUNBIST mode, used to communicate internal data with o/p pad.
- ⇒ When mode is 0, cell is in SAMPLE/ PRELOAD mode.

## \* Boundary Scan Test :-

### \* Need of boundary scan

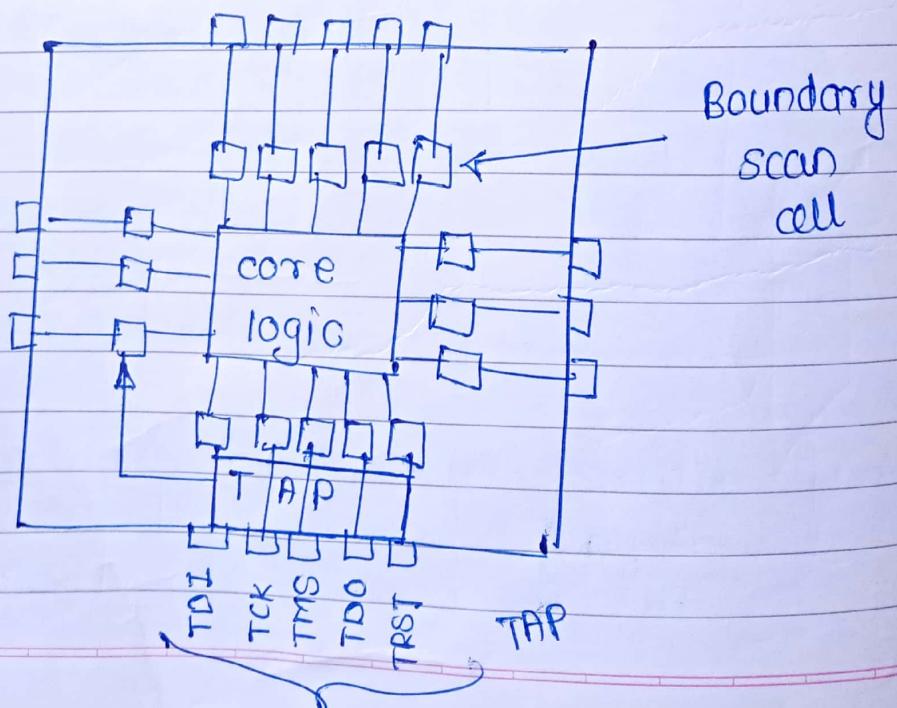
⇒ IC become more complex for testing when no of pins are more.

⇒ Physical probe testing is not possible for high density boards having multiple layers and very fine traces.

### \* Boundary scan :-

⇒ Boundary scan test is used to test complex PC boards.

⇒ A standard for BST is developed by Joint Test Action Group (JTAG) as IEEE 1149.1 which is "Standard Test Access Port and Boundary Scan Arch".



→ one cell of boundary scan register is placed bet' each input and output pin and internal core logic.

→ TAP controller and additional test logic are also added to core logic.

TDI → Test data input

TCK → Test clock

TMS → Test mode select

TDO → Test data output

TRST → Test reset

→ A typical boundary scan cell is given in following fig.

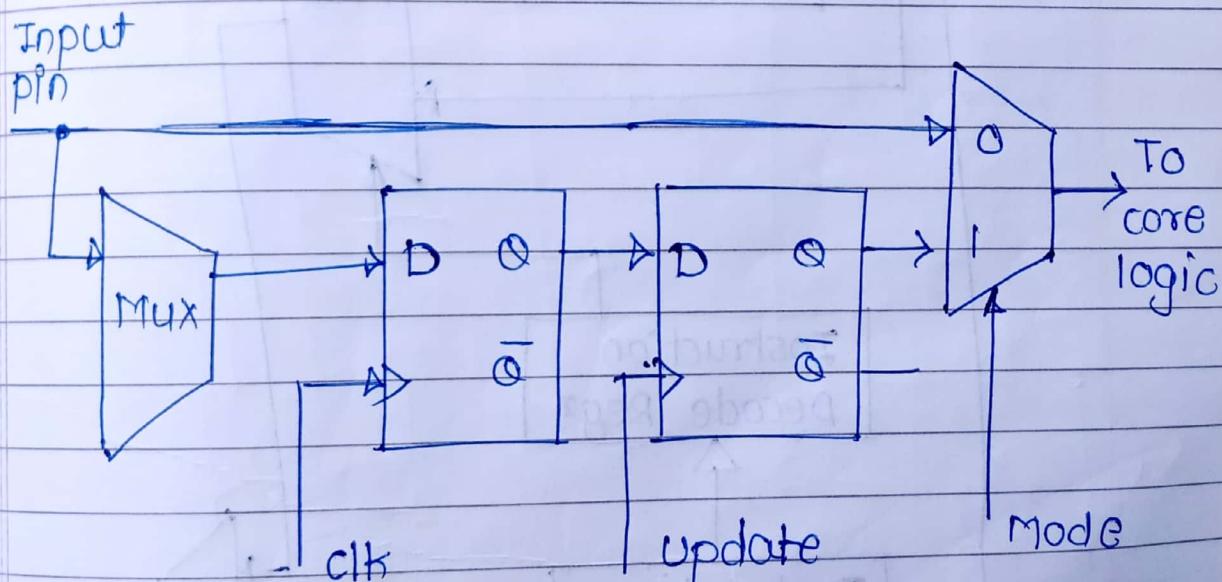
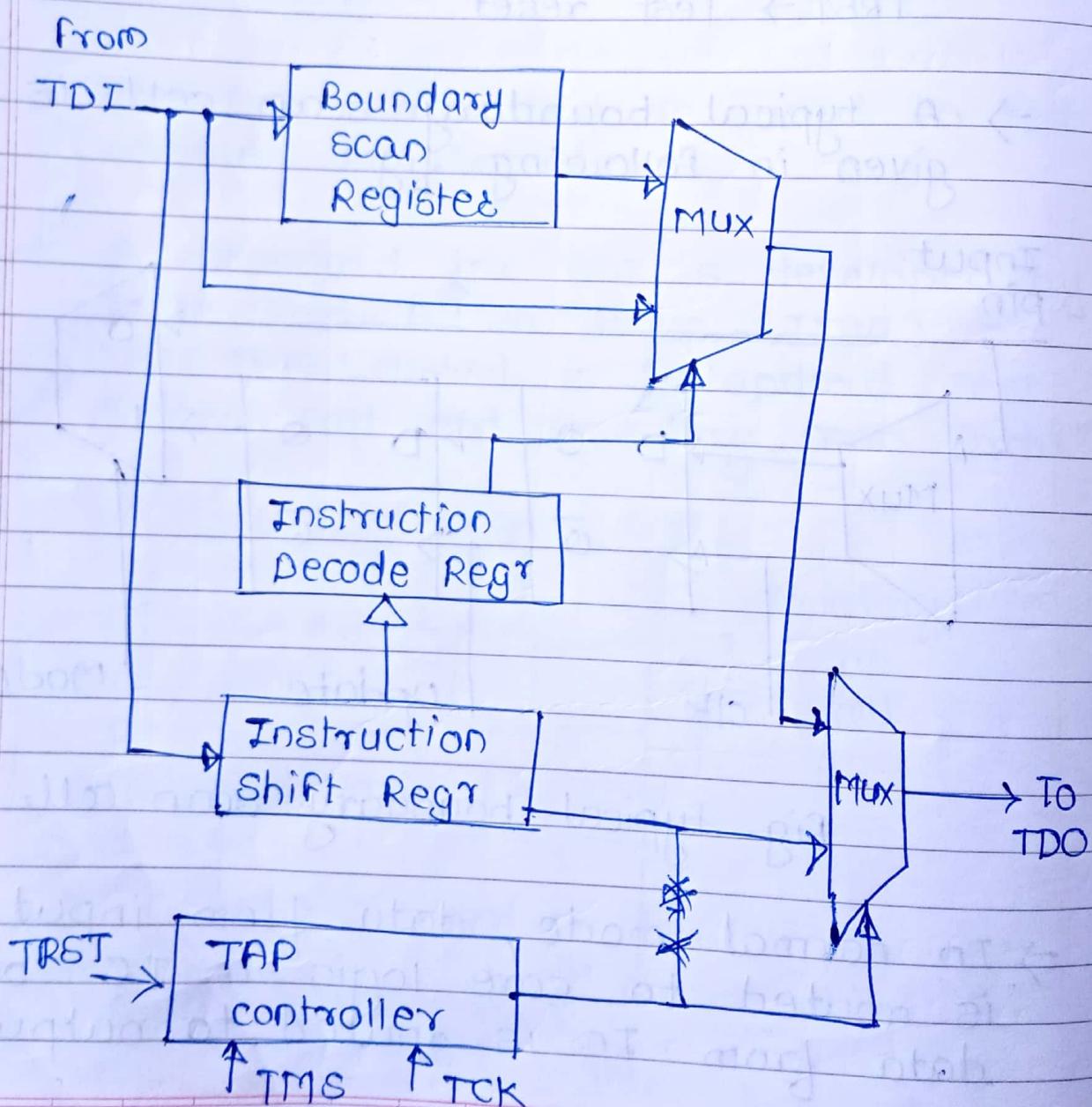
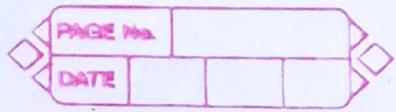


Fig. typical boundary scan cell

→ In normal mode, data from input pin is routed to core logic in IC or data from IC is routed to output pin.

- ⇒ In shift mode, serial data from previous cell is clocked to flip flop  $Q_1$ , at the same time data stored in  $Q_1$  is clocked in to next boundary cell.
- ⇒ After  $Q_2$  is updated, test data is supplied to internal logic or output pin.
- ⇒ Boundary scan architecture is shown as,





⇒ TAP controller is 16 state finite state machine and it proceed from state to state.

⇒ There are two boundary scan registers BSR1 and BSR2.

BSR1 ⇒ shifts logic from input pin to core logic

BSR2 ⇒ shifts logic from core logic to output pins

## Joint Test Access Group

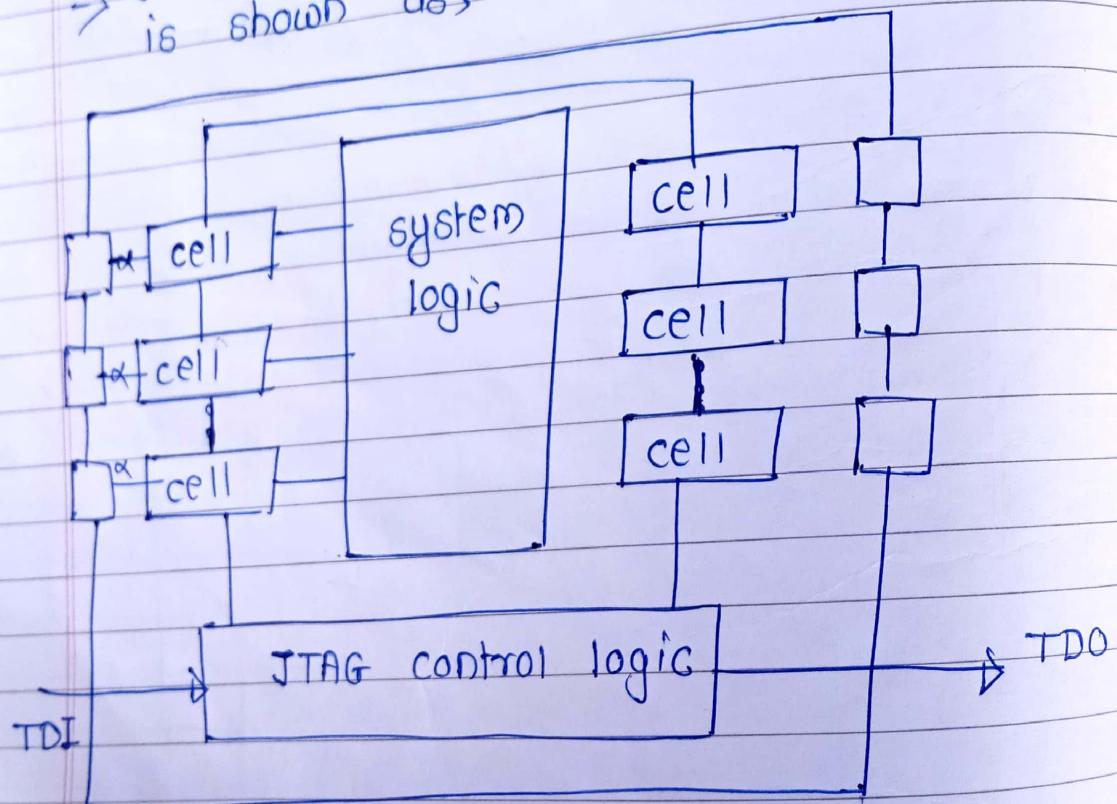
\* JTAG:-

⇒ JTAG is IEEE 1149.1 standard.

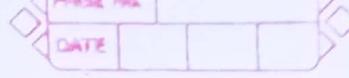
⇒ It is normally used to verify that the circuit has been mounted on the circuit board properly.

⇒ JTAG tests device functionality and connection using test access port and boundary scan.

⇒ JTAG used to test internal circuitry is shown as,



⇒ JTAG also specifies built in self test mode. BIST is used to limit no of vector that need to be clocked through scan path.

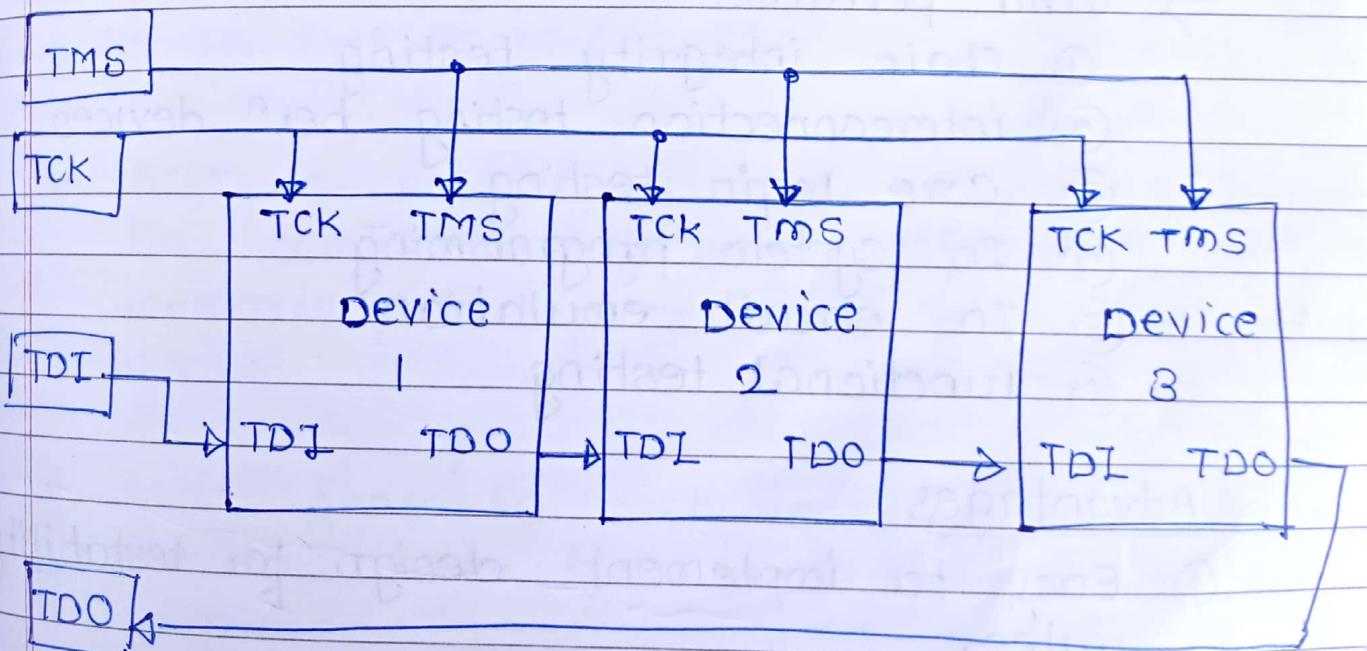


⇒ JTAG implements standard for on chip instrumentation in Electronics Design Automation (EDA) as a complementary tool for digital simulation.

⇒ When combined with built in self test (BIST) the JTAG scan chain enables a low overhead, embedded solution to testing IC to detect certain faults like short, open and logic errors.

⇒ This scan chain mechanism does not help to diagnose the fault.

### JTAG 1149.1 ckt dia



⇒ JTAG 1149.1 has following connector pins

- ① TDI → test data input
- ② TDO → test data output
- ③ TCK → test clock

- 24
- ④ TMS  $\rightarrow$  test mode select
  - ⑤ TRST  $\rightarrow$  test reset.

$\Rightarrow$  Since only one data line is available,  
the data is serial data.

$\Rightarrow$  one bit of data transferred in from  
TDI, out to TDO per TCK clock rising  
edge.

$\Rightarrow$  maximum operating clock frequency for  
TCK is typically 10-100 MHz but it  
may vary depending upon all chips  
in chain.

$\Rightarrow$  JTAG performs -

- ① chain integrity testing
- ② interconnection testing bet<sup>n</sup> devices
- ③ core logic testing
- ④ In system programming
- ⑤ In circuit emulation.
- ⑥ functional testing.

### Advantages

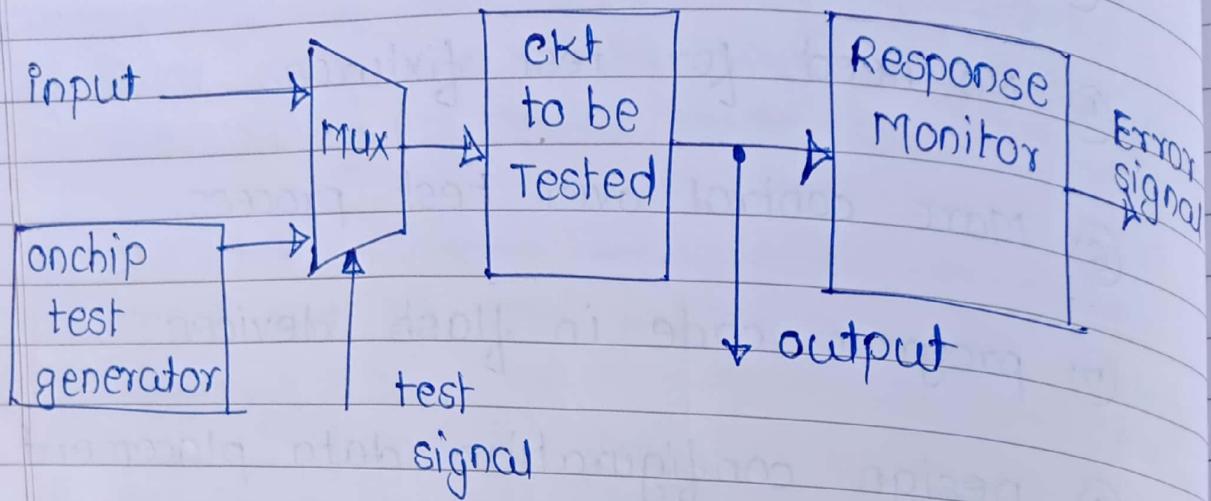
- ① Easy to implement design for testability rules.
- ② Design analysis prior to PCB layout to improve testability.
- ③ Packaging problems are found prior

- to PCB layout.
- ④ Little need for test points
- ⑤ No need for test fixtures.
- ⑥ More control over test process.
- ⑦ program code in flash devices
- ⑧ design configuration data placement in to CPLD's.
- ⑨ JTAG emulation & source level debugging.

### \* Built In Self Test (BIST)

- ⇒ BIST is set of structured test techniques for combinational, sequential logic, memories, multipliers and other embedded blocks.
- ⇒ BIST generate test vector, apply it to circuit under test and then check the response.
- ⇒ All these three steps are done by itself therefore called as built in self test.

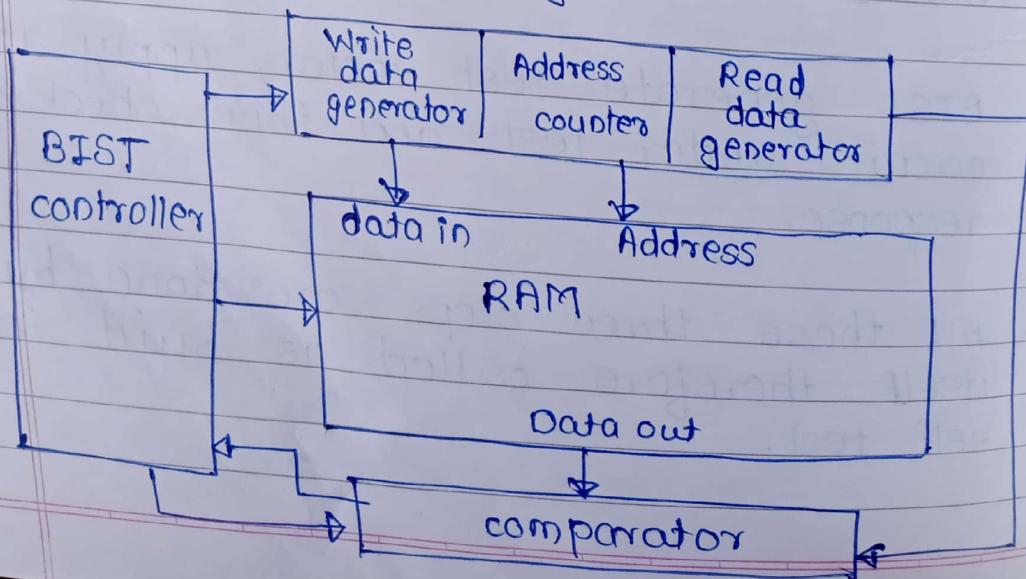
⇒ General BIST structure is shown as,



⇒ When a test mode is selected by the test select signal, on chip test generator applies test pattern to circuit under test.

⇒ The resulting output is observed by response monitor which produces an error signal if an incorrect output pattern is detected.

e.g. RAM testing using BIST



⇒ Normally test pattern are generated using

① LFSR

(Linear Feedback Shift Registers)

② MISR

(multiple Input shift Register)

① LFSR

⇒ LFSR used to generate test pattern which consists of flip flop. A four bit LFSR is shown in fig.

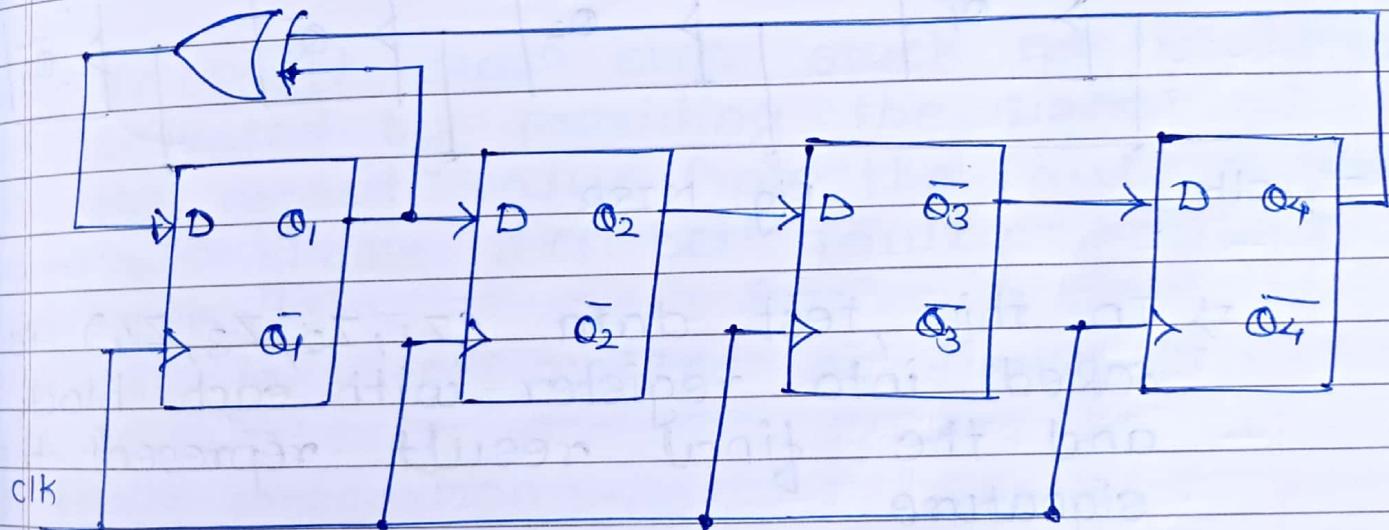


Fig. 4 bit LFSR

⇒ The output from 1st and 4th flip flop are XORed together and fed back to input D of 1st flip flop.

⇒ A general form of LFSR is shift register with two or more outputs from flip flop XORed together.

⇒ XOR is equivalent to addition of addition is linear. ∴ name is Linear shift reg?

## (2) MISR

MISR can be constructed by modifying a LFSR by adding XOR gates as shown in following fig.

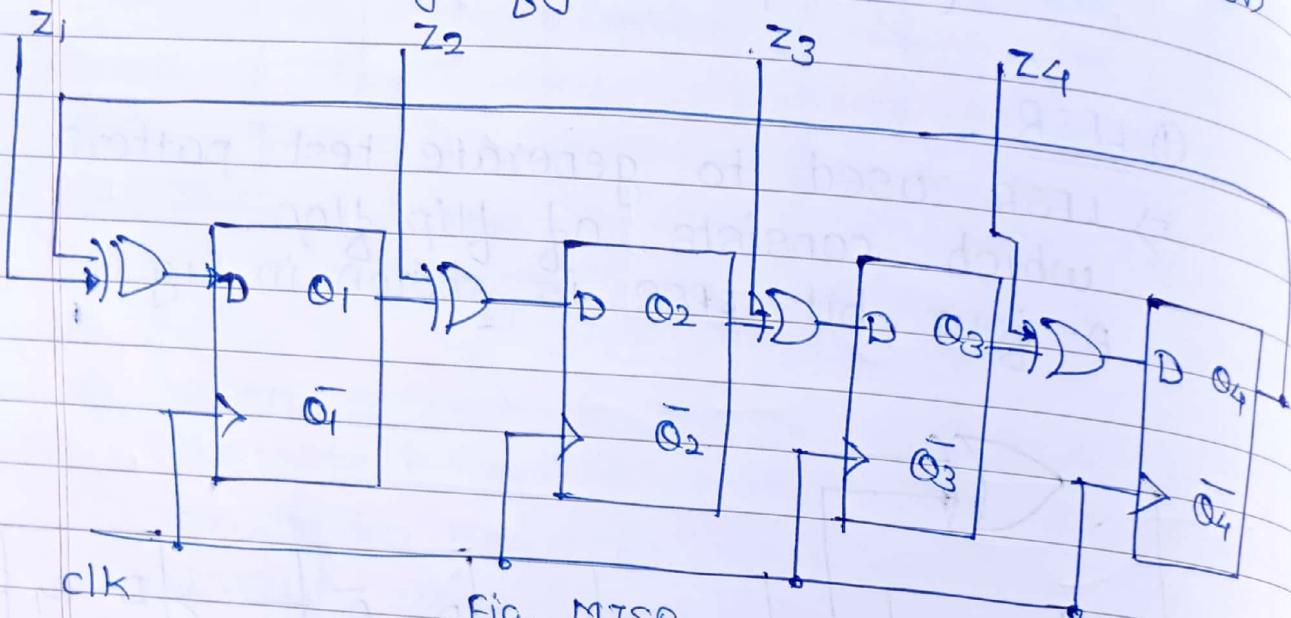


Fig. MISR

⇒ In this, test. data ( $z_1, z_2, z_3, z_4$ ) is XORed into register with each block and the final result represent the signature.

⇒ This signature is compared with the signature of correctly functioning unit.

⇒ An  $n$  bit signature register maps all possible inputs in to one of the  $2^n$  possible signatures.

## \* Path sensitizing

- Path sensitization is automatic test pattern generation method.

- Path sensitization consists of 3 steps -

- 1) Fault sensitization
- 2) Fault propagation
- 3) Line justification

### ① Fault sensitization :-

- In fault sens' step stuck at fault is activated by providing the signal at the opposite value from the fault value.
- It shows diff betn faulty & good circuit.
- It is also called as fault activation.

### ② Fault propagation :-

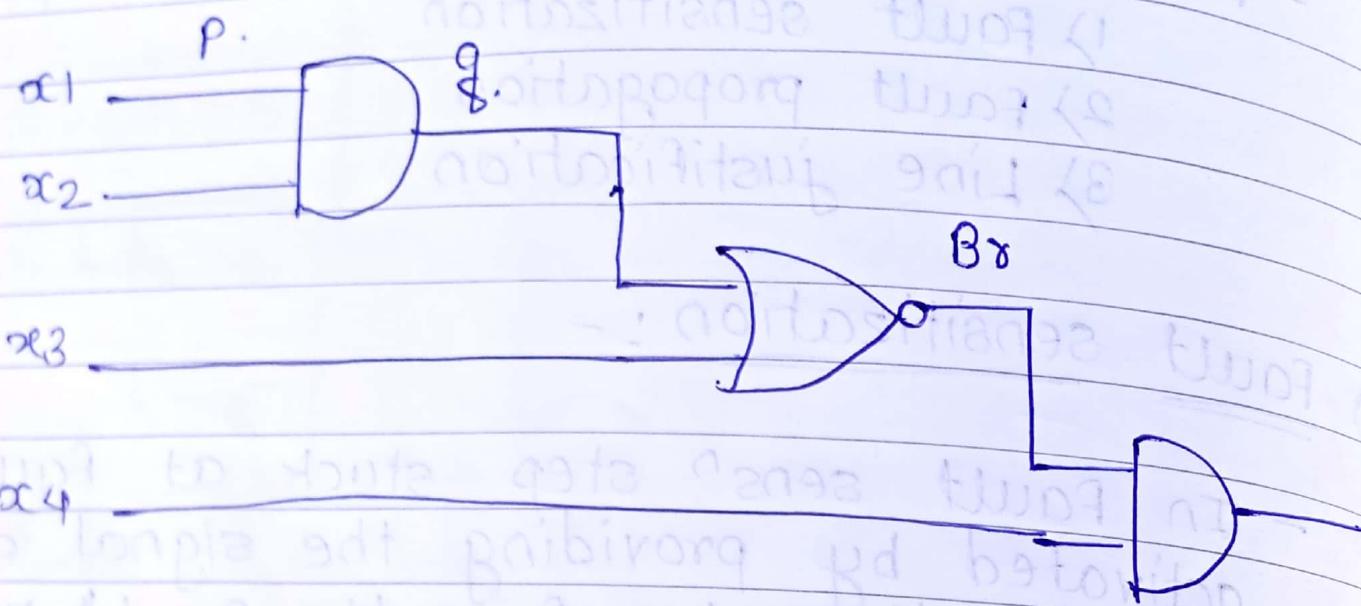
In this fault is propagated through the path towards the primary output of circuit.

### ③ Line justification :-

- In line just' the fault activation of path sensitization is justified through internal circuitry of logic gate.
- If the line just' shows conflicts in signal assignment then the back track is used.

which discard the previous assignment and makes alternative assignment in the signal.

→ following fig shows path sens<sup>n</sup>.



- The path from primary input  $x_1$  to primary output  $s$  is activated which consisted of wires  $P, Q, R, S$ .
- The primary i/p are activated are  $x_2 = x_3 = 0$ , and  $x_4 = 1$ .

When  $x_1 = 0$ , output  $s = 1$

When  $x_1 = 1$ , output  $s = 0$ .

which clearly shows that path from  $x_2$  is  $s$  is sensitized.