

5.24

- (iv) Average Short - Circuit Current Flowing Through The Gate (I_s)

The short circuit current can be reduced by considering the following two points.

- Reduction of unavoidable current
- Avoidence of unnecessary current.

5.2 CMOS TESTING

5.2.1 Need for Testing

Q36. What is the need for testing and explain the two categories of testing.

Model Paper-III, Q11

(or)

Explain in detail the need for testing and the two groups of testing.

Ans:

Need for Testing

Generally, the production of a particular IC is the ratio of number of good die to the total number of dies per wafer. All the dies on a wafer may not be operated correctly due to the complexity of manufacturing process. That incorrect operation of a chip is due to miniature defects in a processing steps.

The above all reasons are responsible for malfunctioning of chip. Thus, it is the aim of a test procedure to determine which die (chip) is good should be used in end systems. The different levels at which testing occurs is,

- Wafer level
- Packaged - chip level
- Board level
- System level
- In the field

By knowing the defects of a chip at wafer level is leads to the low fabrication cost.

The following information shows the relative cost to recognize the fault of their respective levels.

Wafer level	-	\$0.01 - \$0.1
Packaged chip level	-	\$0.1 - \$11
Board level	-	\$1 - \$10
System level	-	\$10 - \$100
Field level	-	\$100 - \$1000

From the above information, it is clear that manufacturing cost is kept low, if the faults are detected at wafer levels. For the testing at the packaged-chip level or board level, following parameters are needed.

- Expenditure to develop suitable tests at the wafer level.
- Mixed signal conditions
- Speed consideration.

The wafer level and system levels tests can be done by component vendor and satellite-borne electronics respectively.

Categories of Testing

There are two categories of testing available. They are,

- Functionality test
- Manufacturing test.

1. Functionality Test

In functionality testing, the whole design process of construction is verified to check whether the circuit is working as per given function or not. It is required to justify the functionality equivalent to some specification.

The specification is any one of the following,

- Verbal definition or explanation
- Plain - language text
- Pascal, C, FORTRAN
- VHDL, verilog etc.
- Tables contains inputs and outputs.

The specification which is used is comprises of any two descriptions of chip. It also involves to check the equivalence to inputs and outputs at available checkpoints in time.

The better exact checking is performed on a cycle-by-cycle basis. There is no good theory on how to ensure that good functional tests be written. The best solution is to stimulate the chip or system as closely as possible to the way it will be used in real world.

It is impossible in several cases due to following reasons,

- Reluctant and slow simulation times.
- Very lengthy information sequence.

One approach is to divide the simulation hierarchy into a small modules at lower levels.

2. Manufacturing Tests

The manufacturing testing methods are used to detect the manufacturing defects in a chip during their fabrication. This is done by checking the operation of gates.

The following are defects in manufacturing tests.

- Layer-to layer shorts (i.e., metal to metal)
- Discontinuous wires
- Thin oxide shorts to substrate or well
- Nodes shorted to power or ground
- Nodes shorted to each other.

To detect the above defects, manufacturing tests is needed through verifying the gates and registers. These tests are normally carried out at the wafer level to cull out bad dies, and then on the packaged parts. The length of the tests at wafer level might be shortened to reduce test time based on experience with test sequence.

Apart from verification of internal gates, I/O integrity is also tested through completing the following tests,

- I/O-level test
- Speed test
- I_{DD} test.



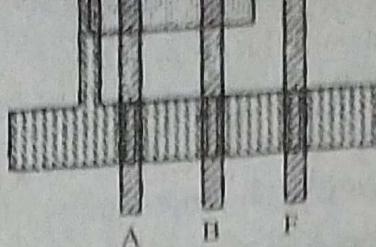


Figure (c): A Defect that Causes Static I_{DD} Current

Q40. Explain the terms controllability, observability and fault coverage.

Model Paper-II, Q11

Ans:

(a) Observability

Observability of any internal node is a degree to which it can be observed that node at output operates correctly. It is very useful to a tester/engineer to measure the output of a gate with in a complex circuit to check its correct operation. The main goal of any design engineer is that easy observation of gate outputs. Higher observability indicates the less number of cycles required to measure output node value.

(b) Controllability

The controllability is defined as ability of setting a particular logic signal to '0' or '1'. It is used to analyze the difficulty of testing a particular circuit. These value setting can be done by the input pads [IP's]. The main goal of any designer is to design the easily controllable nodes. So it is possible to control easily by using IP's.

(c) Fault Coverage

Fault coverage is defined as the percentage of fault that can be founded by the applied test vector. It gives the measure of goodness of test program. The calculation involves the following steps,

- (i) Every individual circuit node set to logic '0' one by one.
- (ii) Then the output is compared with the 'good machine'.
- (iii) If the output doesn't accurate, then it comes under 'faulty machine'.
- (iv) Then above steps are repeated by set the circuit node to logic '1'.
- (v) Percentage of fault coverage is,

$$\% \text{ fault coverage} = \frac{\text{Number of faults detected}}{\text{Total nodes in the circuit}}$$

Print on the TITLE COVER before you buy

'Failure', or PDCF, is a set of inputs that sensitizes a path for a particular fault within a circuit. The 'Propagation D Cube', or PDC, is a set of inputs that propagates a D from the inputs to the output.

The D-algorithm picks all possible PDCF's for the circuit under test and applies them to the circuit with their corresponding PDC's to propagate various faults to the output. While the PDCF's and PDC's are being applied, the 'implied' values for other circuit nodes are tested for consistency, rejecting sets of inputs that cause a circuit violation. The application and testing of various PDCF's and PDC's for a circuit is done repeatedly and recursively, until the minimal set of input patterns necessary to test the circuit for the specified faults is determined.

Q42. Explain about different fault models in VLSI testing with examples.

April-11, Set-3, Q3(b)

(or)

Explain the following simulation techniques,

- (a) Serial and parallel fault simulation**
- (b) Concurrent fault simulation.**

Ans:

Fault Simulation

Fault simulation is a process of simulation of circuit with existence of faults. That fault simulation results are analyzed with fault free simulation results of same circuit, then respective faults are detected.

UDENTS

SIA GROUP 

5.2.8

The fault simulation techniques are described as follows,

- (i) Serial and parallel fault simulation
 - (ii) Concurrent fault simulation.
- (a) **Serial Fault Simulation**
- It is one of the easiest and impractical simulation technique.
 - It is determined by considering two versions or copies of circuit i.e., good circuit and faulty circuit.
 - It first select a fault and include to faulty circuit.
 - Then repeat the simulating process at a time.
 - It is slow process.
- (b) **Parallel Fault Simulation**
- The main benefit of a parallel fault simulation is multiple bits of words. It takes only binary (0 and 1) values. It uses 32 bit machine word, an integer which consists of 32 bit binary vector. It is possible to simulate 32 copies at time by using parallel fault simulation.
- It is 32 times faster than serial fault simulation. Here it consider one good copy and include that into other copies.
- (ii) **Concurrent Fault Simulation**
- It is one of the advantageous fault simulation technique which it does not simulate the whole circuit. It simulates only differential parts of the whole circuit. At first, it completely simulates the good circuit. After that the only fault is simulated.
- It is very critical to find the fault parts of circuit. But it is used to save memory and also possible to do concurrent simulation. It is segmented into different modules which each module is simulated separately.

3. Adding multiplexers
4. Providing for easy state reset.
- Structured Design Methods**
- Structured design method incorporates serial scan and parallel scan.
- For answer refer Unit-V, Q45, Topics: Serial Scan, Parallel Scan.

(ii) **Self-test**

For answer refer Unit-V, Q46.

Q45. Explain the Scan-based Test Techniques.

Ans:

There are three types of Scan-based Techniques. They are as follows,

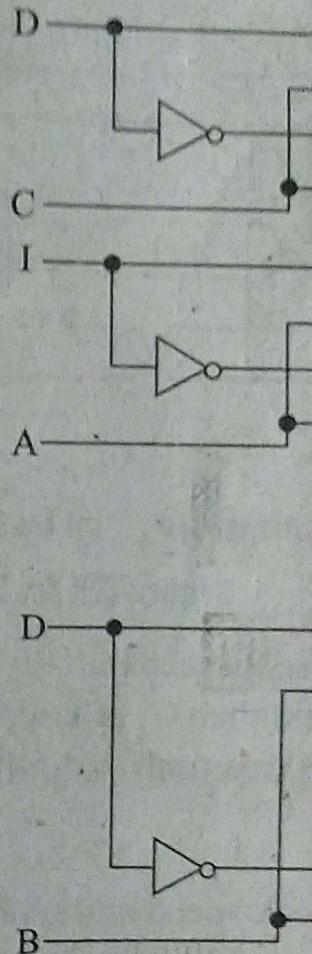
1. Level Sensitive Scan Design (LSSD)
2. Serial Scan
3. Parallel Scan.

1. **Level Sensitive Scan Design (LSSD)**

The most widely used Scan-based test technique is level sensitive scan design, or the LSSD. This is based on two principles. First that the circuit is level sensitive and the second principle of LSSD is that each register may be converted to a serial shift register.

From the first principle, any logic system is level sensitive if and only if it satisfies the following rule,

- (a) Its steady state response of any allowed input change is independent of circuit delays.
- (b) The response is independent of the order in changing the inputs.



Fig

SPECTRUM ALL-IN-ONE JOURNAL FOR ENGINEERING STUDENTS

The block level and gate level implementation of Shift Register Latch (SRL) is shown in figures (a), (b) and (c) respectively. The block level arrangement consists of,

- (a) Two latches L_1 and L_2 where L_1 consists of serial data port, enable, data port and enable C.
- (b) L_2 consists of data port D and signal B.

If enable 'A' is high value of T_1 is high. If L_1 is set by D, then 'C' becomes high. It is not possible for 'A' and 'C' at a time become high.

If signal 'B' in ' L_2 ' is high, then T_1 is transmitted to T_2 .

In gate level approach, 'D' is normal input to register and T_2 is the output signal while ' L_1 ' and ' L_2 ' acts as master and slave respectively.

The typical and expanded view of a LSSD system is shown in figures 2 (a) and 2(b) respectively. In these, the inputs are taken from preceding stages and outcomes are QA1, QA2 and QA3. These outcomes are connected to logic circuits. These logic circuit outcomes are connected to second rank of SRL. The outcomes of these are QB1 and QB2 and QB3.

In this testing process, serial data-in is clocking to right point which is controlled and serial data out is for observation. Hence its concluded that input is controlled and output is observed.

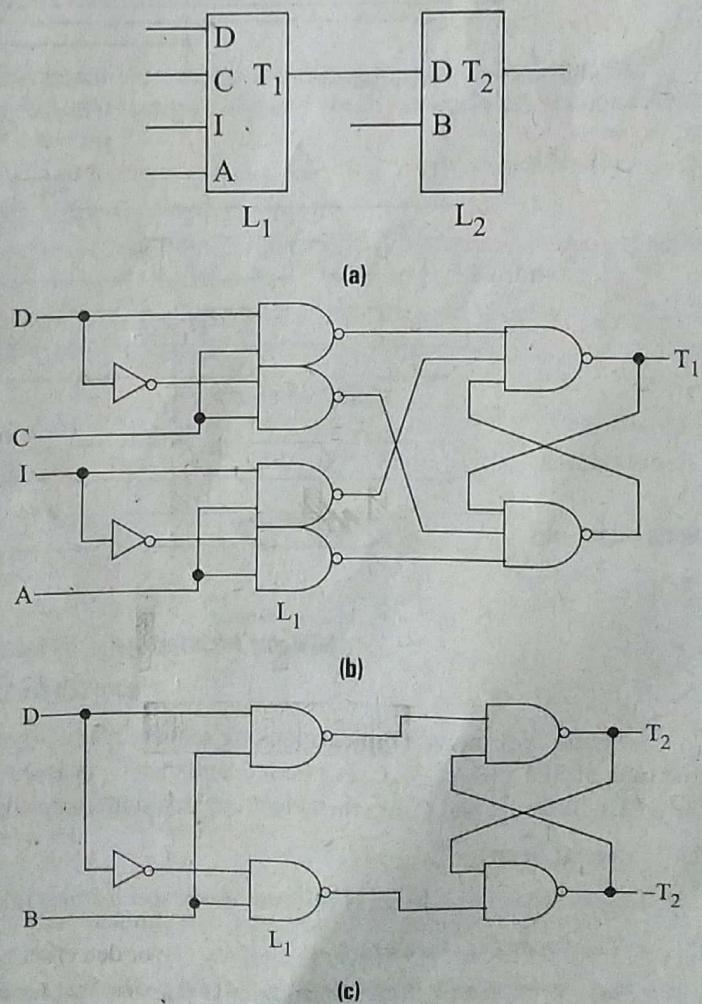
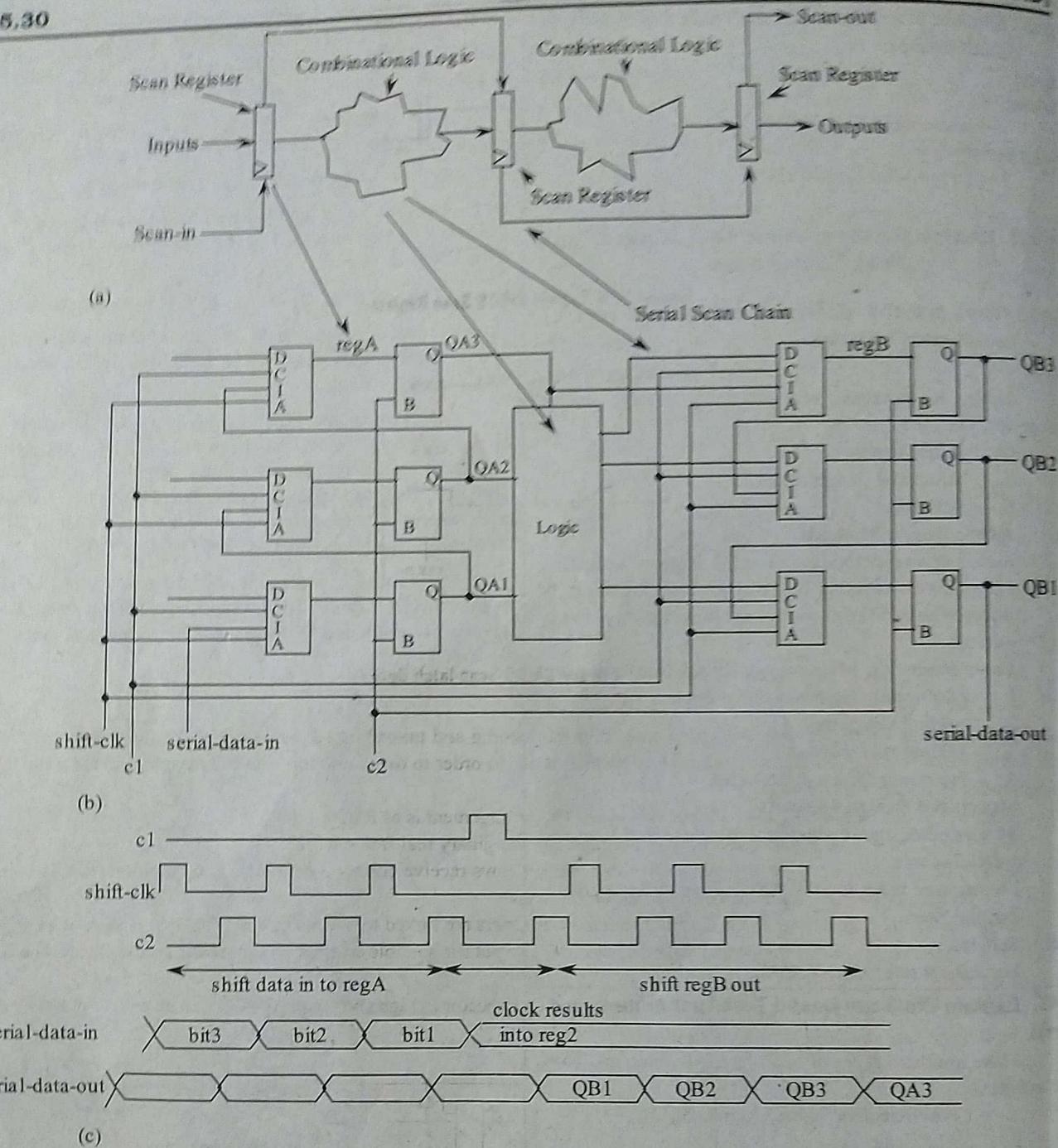


Figure (1): A Shift Register Latch



(a) Basic Architecture ; (b) Example Circuit ; (c) Example Timing

Figure (2): An LSSD Scan Chain

Figure 2(c) shows a typical clocking sequence. Initially the shift-clk and C_1 are clocked three times to shift data in to the first rank of SRLs (QA1-3). C_1 is asserted, and then C_2 is asserted, clocking the output of the logic block into the second rank of the SRLs. Shift-clk and C_2 are then clocked three times to shift QB1, QB2, and QB3 out via the serial - data - outline.

2. Serial Scan

Serial scan has a features of faster clock speed and small registers structure which is shown in figure (3).

The figure (3) consists of,

- Multiplexer which consists of D register, Test Input pin (TI) and Test Enable pin (TE).
- If TE is high, the TI undergoes into rising edge clock.

The following diagrams shows various implementations of circuit diagram of CMOS SRL implementations.

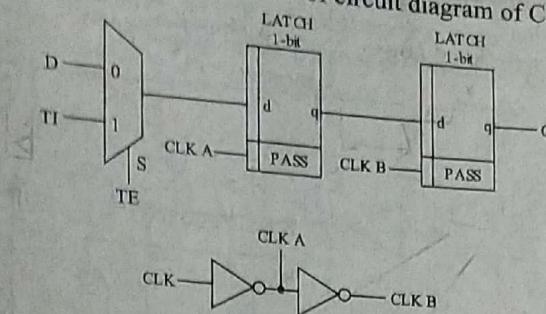


Figure (3): A Typical CMOS Scan Register

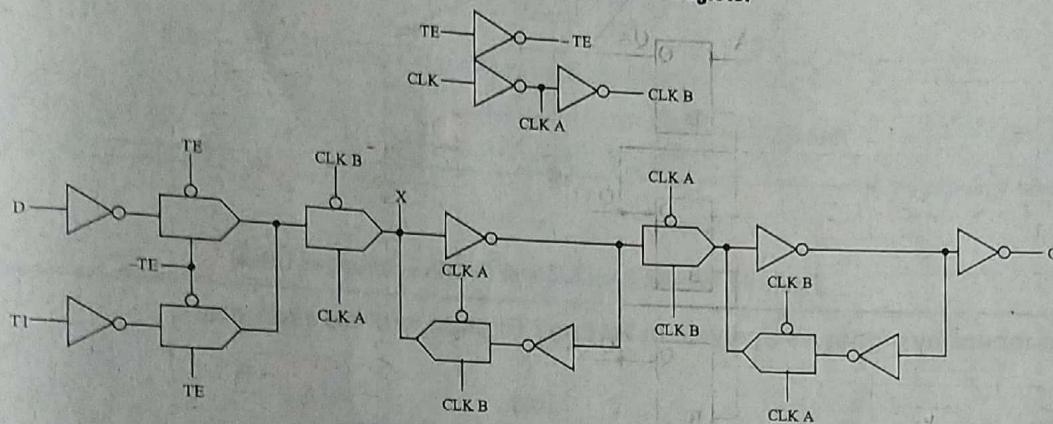


Figure (4): Various CMOS Scan-latch Options

Parallel Scan

The chains of serial-scan become quite long, and the loading and unloading sequence can dominate testing time. An extension of serial scan is called random-access or parallel scan. In order to overcome this disadvantage by reducing the long chains, we can use parallel scan.

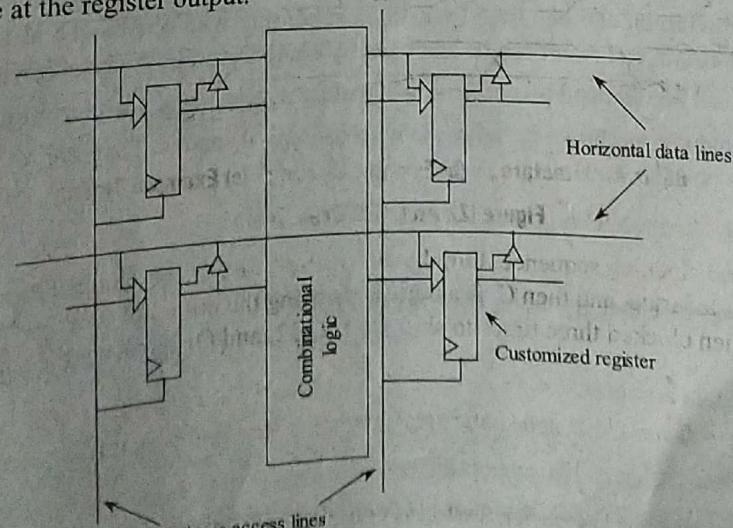
Figure (5) shows basic structure of a parallel scan. The arrangement is as follows.

- (a) Every individual register in design is packaged on imaginary real frame work.
 - (b) Within these registers, the registers with common rows receive common data lines and common read and write control signals are received by the common columns.

From the figure, it is clear that the D and Q signals of registers are linked to normal circuit. The observation of a register output can be possible by enable the proper column read line and put the suitable address on output data multiplexer. The data to be written to any register correspondingly.

Similarly, if test-write enable is low or 0. Probe [J] and CLK becomes high and sense [i] value can be driven towards node available at the register output.

These changes are visible at the register output.



Figure(5): Parallel Scan-basic Structure

5.32

Figure (6) shows a D register implementation called a cross-controlled catch. It consists of a normal CMOS master-slave edge triggered register augmented by two small n-transistors, N_1 , and N_2 . When test-write-enable is high, probe[j] is high, and CLK is low, the value of node Y(D) may be sensed on sense[i] via transistor N_2 .

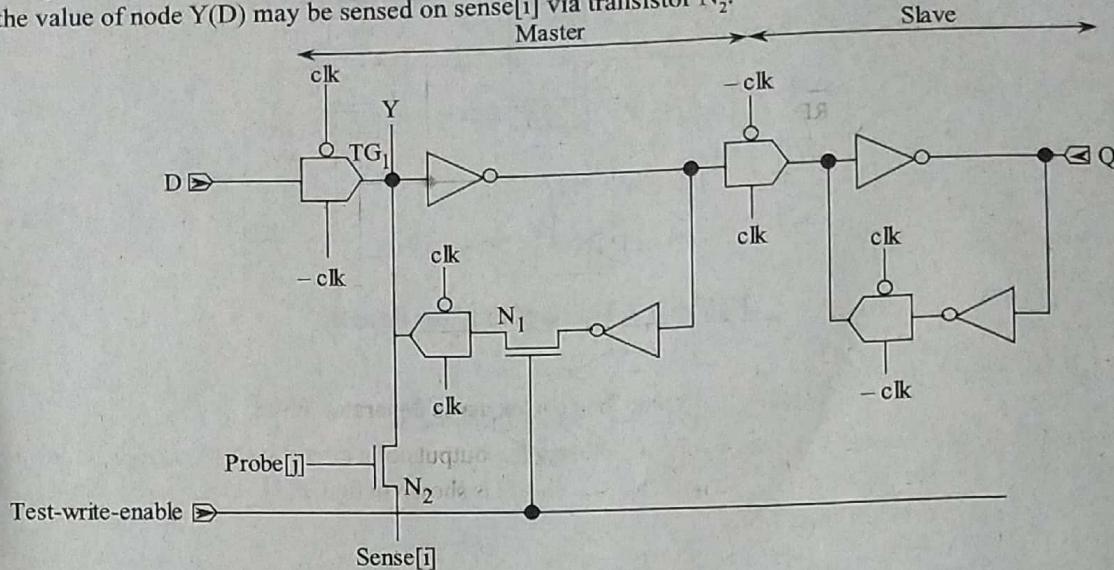


Figure (6): Parallel Scan Register (A Cross-controlled Latch)

Q46. What is meant by signature analysis in testing? Explain with an example.

April-11, Set-4, Q8(b)

(or)

Explain the self-test technique 'Signature Analysis and BILBO'.

Ans:

Self-test, as the name indicates, rely on growing circuits to allow them to perform operations on themselves that prove correct operation. One method of incorporating the self-test (or Built-in test) is to use signature analysis or cyclic-redundancy checking.

Signature Analysis and BILBO

The approach of a signature analysis and BILBO comprises of,

1. Pseudo random sequence generator (PRSG) to produce the input signal.
2. That input signal is applied to the combinational circuitry.
3. Then the output signals are perceived by the signal analyzer.

Figure (1) shows the PRSG which comprises a polynomial of length 'N'. This design is established by using a linear feedback shift register (LFSR). In this design, 1-bit registers are connected serially. The outputs of these bits are undergoes to XOR operation and passed to the input of LFSR to determine the appropriate polynomial.

After this operation, the circuit can be tested by using a syndrome which is placed at LFSR and sequence. It contains present and past outputs. The circuit test result can be analyzed by comparing this syndrome with the correct syndrome.

The BILBO structure can be obtained by combining signature analysis and scan technique.

The complete arrangement is as shown in figure (3) with contains 4 modes. They are,

Mode D

$C_0 = C_1 = 1$ it performs like parallel registers.

Mode C

$C_0 = 1, C_1 = 0$ it behaves as either signature analyzer or PRSG.

Mode B

$C_0 = 0, C_1 = 1$ the registers are reset.

Mode A

$C_0 = 0, C_1 = 0$ the registers behaves like scan registers.

For example, in figure (1), the 3-bit shift register is computing the polynomial $f(x) = 1 + x + x^3$. For an n-bit LFSR, the output will cycle through $2^n - 1$ states before repeating the sequence.

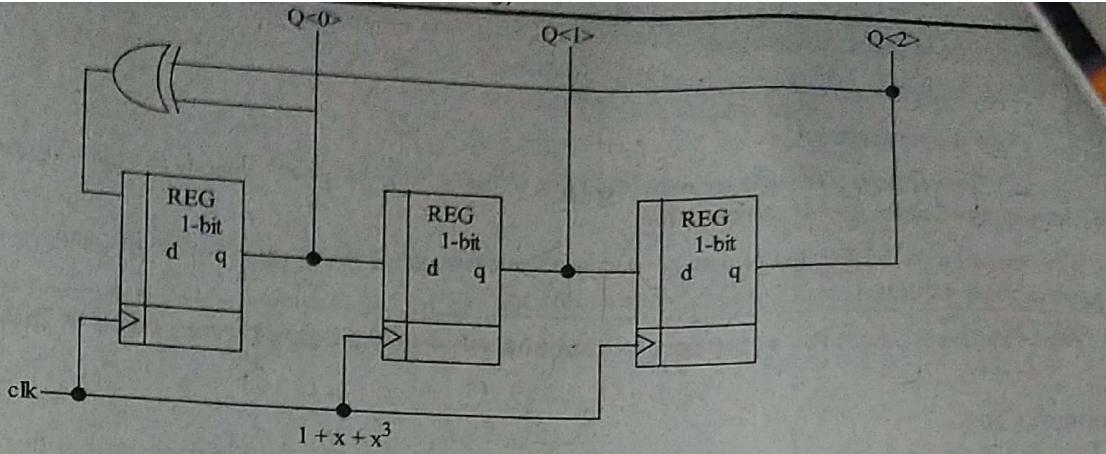


Figure (1): Pseudo Random Sequence Generator (PRSG)

A signature analyzer is constructed by cyclically adding the outputs of a circuit to shift register or an LFSR if successive logic blocks are to be tested in a like manner. A characteristic circuit is shown in figure (2). As each test vector is run, the incoming data is XORed with the contents of the LFSR.

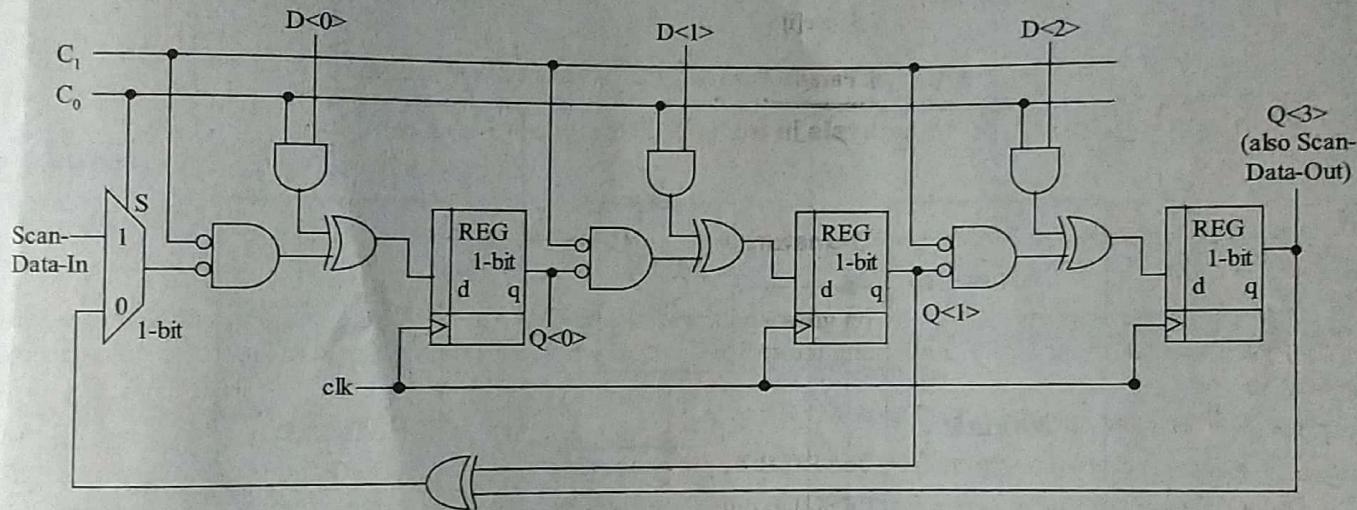


Figure (2): Built-in Logic Block Observation (BILBO) (Individual Register)

MODE	C ₀	C ₁	
A	0	0	Scan Mode
B	0	1	Reset
C	1	0	PRSG or Signature Analyzer
D	1	1	Parallel Register

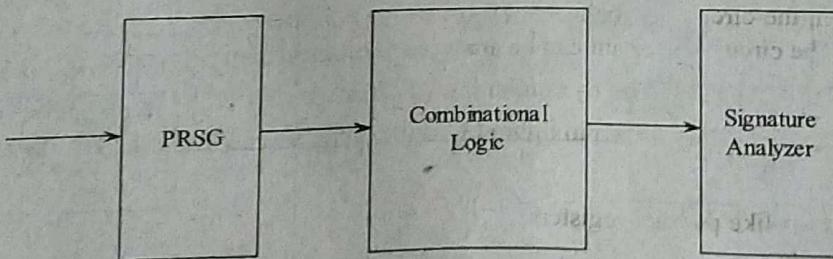


Figure (3): Use in a System

Q47. How IDDQ testing is used to test the bridge faults?

Ans:
DDQ Testing

The CMOS bridging faults can be tested by using an increasingly popular method called IDDQ (V_{DD} supply current quiescent) or current supply monitoring "when a complementary CMOS logic gate is not switching draws no D.C current" is the principle on which IDDQ testing depends. A measurable D.C. I_{DD} will flow, when a bridging fault occurs for some combination of input conditions. IDDQ testing consists of following steps,

Model Paper-I, Q11(a)

Q52. What type of testing techniques are suitable for the following,

- (i) **Memories**
- (ii) **Random logic**
- (iii) **Data path?**

Ans:

(i) Memories

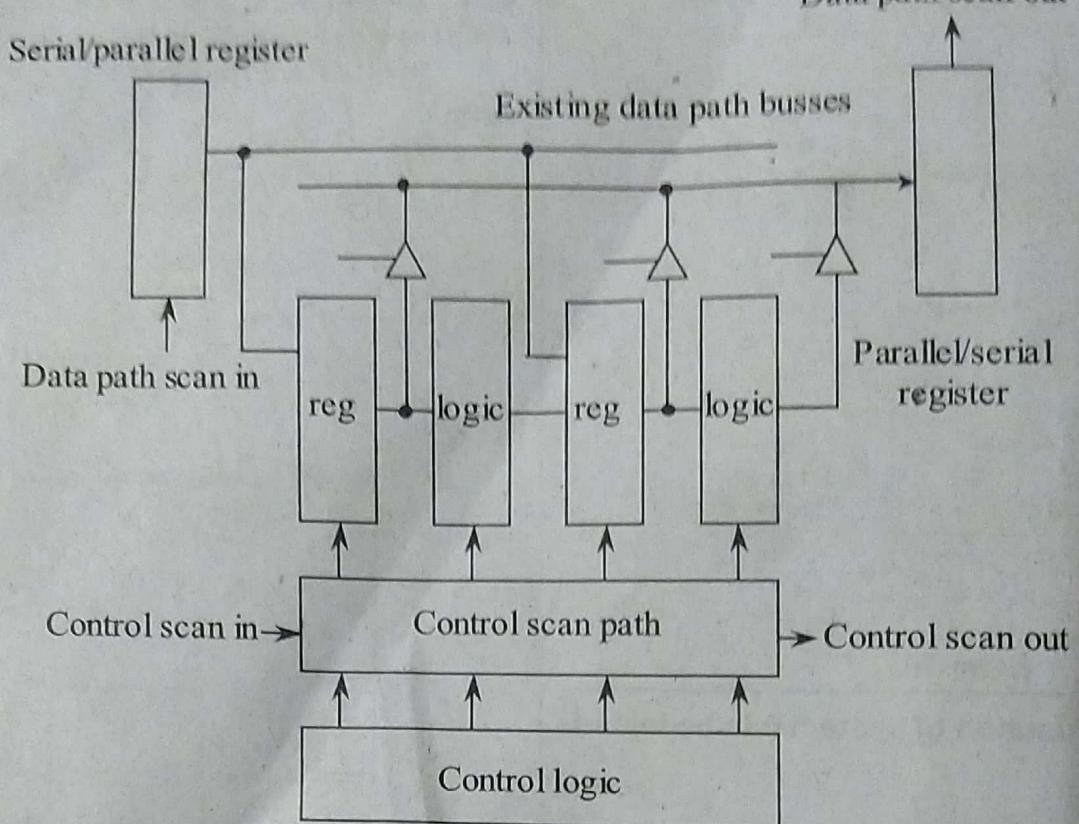
- (a) Self testing techniques are suitable for memories.
- (b) Memory self test is suitable for embedded memories by providing the multiplexers on data inputs and addresses.
- (c) The indirect address sequencing of memories is a mistake.
- (d) The reason is memories are tested exhaustively.
- (e) The writing and reading in a indirect address sequencing can increases the test time which increases the complexity.

(ii) Random Logic

The random logic can be tested by using full serial or parallel scan approach.

(iii) Data Path

The best technique for testing the structures such as data paths is partial serial scan or parallel scan. Figure shows the one approach that has been used in a lisp microprocessor.

**Figure**

Here, a serially loaded register is used to drive the input buses. These in turn may be used to load the internal registers. The bus on to which data path registers may be sourced, is to be loaded into a register that may be serially. All of the control signals to the data path are also made scannable.