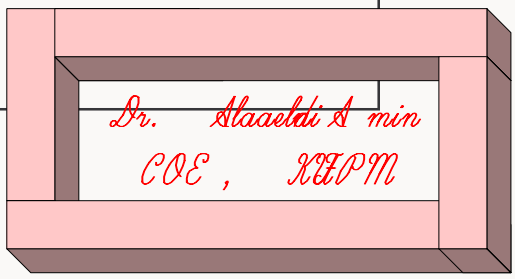
DIGITAL SYSTEM TESTING
COE -545

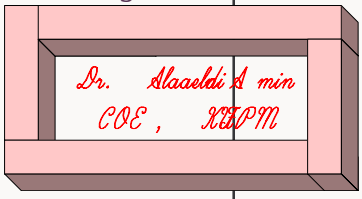
Lecture – 17

Built-In Self-Test (BIST)



Introduction

- **DFT** will Improve CUT Testability (Controllability & Observability) → Still requires expensive ATE to Perform the Test
- To Reduce Test Cost → CUT is Designed to Test Itself → Include On-Chip Logic to Generate TV's and Evaluate Response →Built-In Self Test (**BIST**)
- *V. Agrawal defines BIST as a DFT technique in which testing (test generation and test application) is accomplished through built-in hardware features*
- BIST = BIT (Built in test) + self test



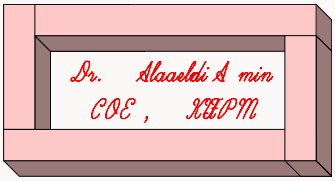
BIST Merits

Advantages

- **Accessability of Internal Nodes (not Possible with External ATE)**
- **At-Speed Test** → Test is applied @ Chip Own Speed
- **Reduced Test Cost** (No Expensive ATE)
- **In-Field Testing** is Possible

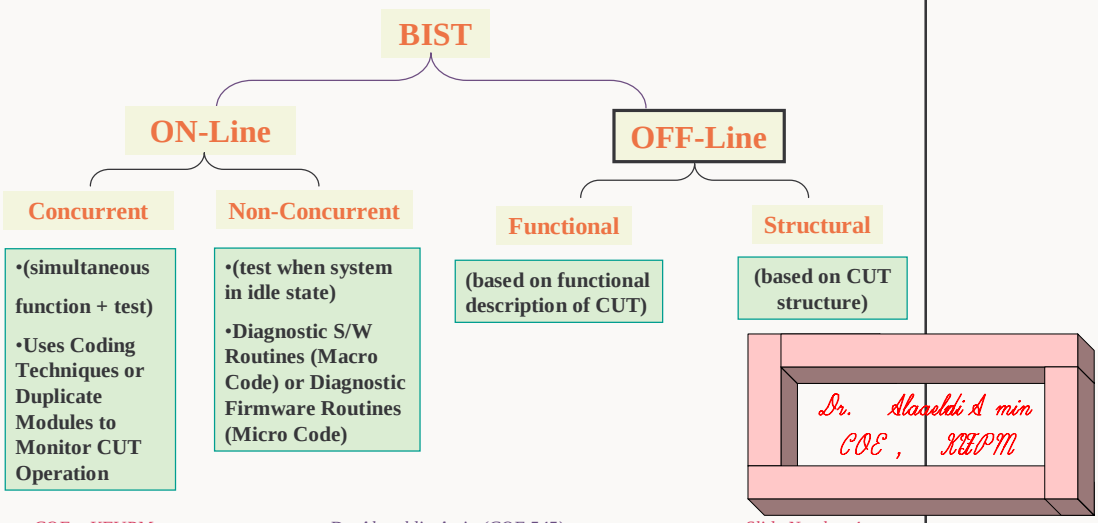
Disadvantages

- **Area Overhead**
- **Performance Overhead**

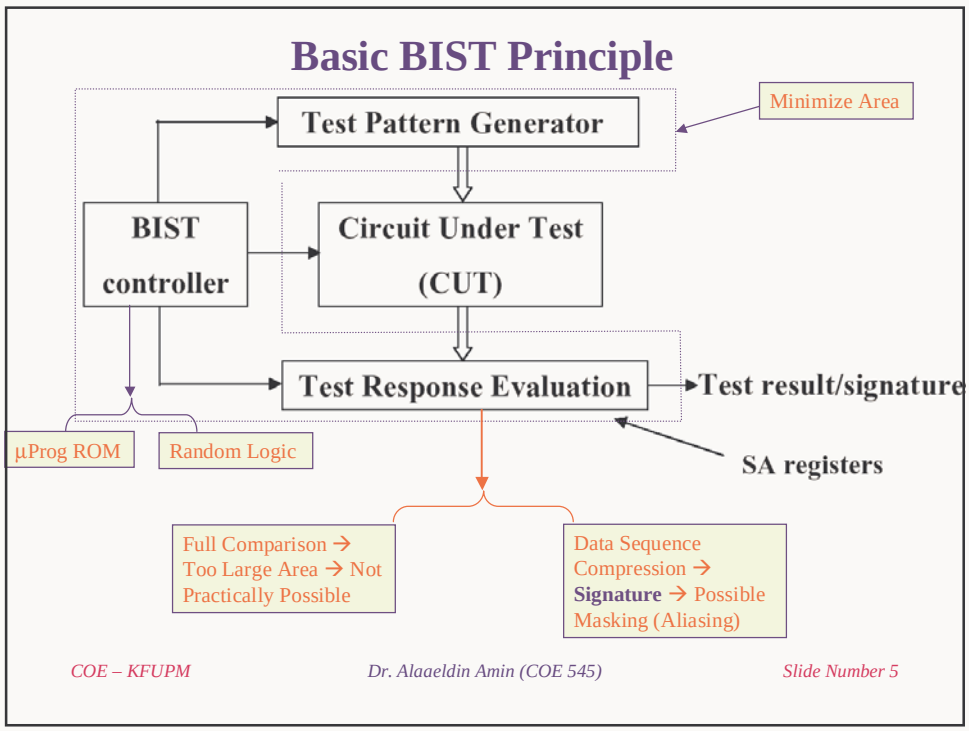


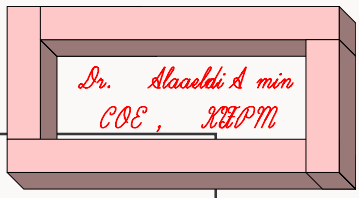
COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 3

BIST Classification



COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 4





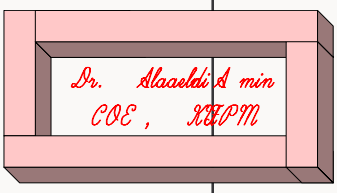
Possible TPG & ORA

Test Pattern Generator (TPG):

1. Counter (Exhaustive)
2. Linear feedback shift register (**LFSR**) → *Pseudo Random*
3. ROM with pre-Stored test data
4. Specially-designed FSM to generate test vectors

Output Response Analyzer (ORA):

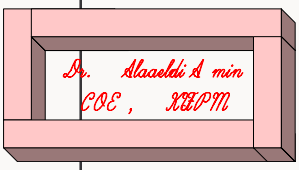
1. One's Counting
2. Transition Counting
3. **LFSR** → Signature Analyzer
4. **MISR** → Multiple Input Signature Register



COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 6

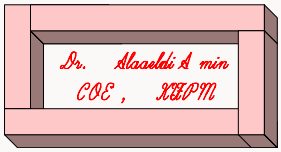
Linear Feedback Shift Register

- **Linear** → Preserves (Follows) the Principle of *SuperPosition*
- Response to a Linear Composition of Stimuli = Linear Combination of Individual Responses to each Stimuli
- Let **Ri** = Response of a Linear CUT to Input Stimuli Si, and
- Let **P** = The Linear CUT Function
- Let **a1, a2** = Constants
- Thus **R1 = P(S1)** and **R2 = P(S2)**
- **a1R1 + a2R2 = a1 P(S1)** + a2 **P(S2) = P(a1S1 + a2S2)**
- **A Linear Logic Circuit Can be Constructed from the following basic Components:-**

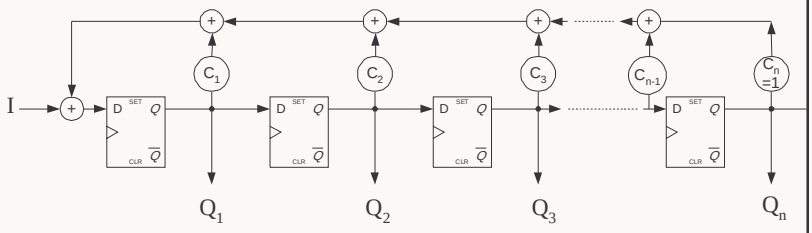


Linear Feedback Shift Register (LFSR)

- A *Linear* Logic Circuit Can be Constructed from the following basic Components:-
  - Unit Delay Elements (D-FFs)
  - Modulo-2 Adders (XOR Gate)
  - Modulo-2 Constant Multiplier (0 → No Connection, 1 → Connection)
- Analyzing these circuits, all *operations* are "***Modulo 2***"

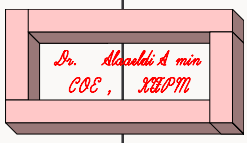


LFSR Type 1 (External XOR)

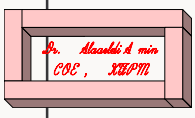


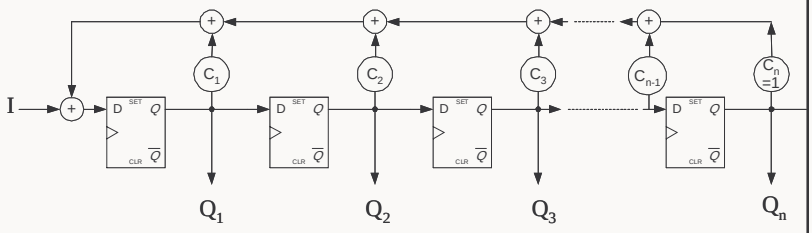
Initial State At mth Clock	I_0	a_0	a_{-1}	a_{-2}	a_{-3}	a_{-n}
	I_m	a_m	a_{m-1}	a_{m-2}	a_{m-3}	a_{m-n}

- $a_m = I_m + c_1a_{m-1} + c_2a_{m-2} + c_3a_{m-3} + ... + c_na_{m-n}$
- Generally $$a_m = I_m + \sum_{i=1}^{n} C_i a_{m-i}$$ Recurrence Relation Describing LFSR



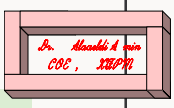
LFSR Type 1 (External XOR)





- The Output Sequence of an LFSR can be represented by a Polynomial G(x), where:

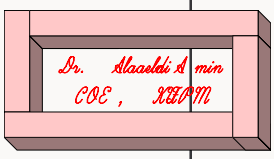
$$G(x) = \sum_{m=0}^{\infty} \{a_m\} x^m$$



- Finally $$G(x) = \frac{\sum_{m=0}^{\infty} I_m x^m + \sum_{i=1}^{n} c_i x^i (a_{-i} x^{-i} + ... + a_{-1} x^{-1})}{1 + \sum_{i=1}^{n} c_i x^i}$$

LFSR Type 1 (External XOR)

- Thus, G(x) is Function of:
  - The Input Sequence
  - The Initial State of the Register
  - The Denominator Polynomial P(x) = $1+\sum_{i=1}^{n} c_i x^i$
  - P(x) is Called the Characteristic Polynomial of the LFSR



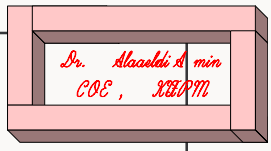
SPECIAL CASES

1. LFSR as Pseudo Random Test Pattern Generator (PRTPG)

- By Initializing the LFSR to 0000001 with no Input Sequence (I_m) -→ We get a Pseudo Random Pattern Output

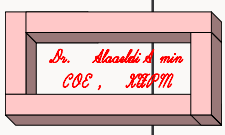
2. LFSR as a Signature Analyzer (SA)

- By Initializing the LFSR to All 0's and Applying the Output Sequence of the CUT to the Input of the LFSR (I_m) → The Sequence is Compressed into One Signature



LFSR Type 1 (External XOR)

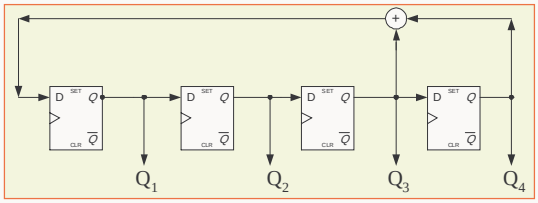
- The pattern produced by an LFSR is Periodic with a Max Period of 2^n-1, where *n* is the number off FFs in the LFSR = The degree of the C/C. Polynomial C/C. Polynomial P(x)
- C/C. Polynomials which have maximal length are known as primitive polynomials
- Autonomous LFSRs with no I/P sequence (with Im=0), Serve as PRTPG.
- PRTPG Produce Sequences that are deterministic (not random) but the generated sequence has properties that have some properties of Random signals, e.g.:
  - Has (2^{n-1}) 1's and (2^{n-1}-1) 0's
  - One Run of *n* Consecutive 1's and one run of (*n-1*) 0's



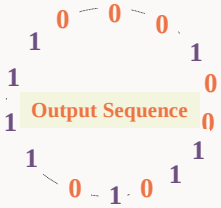
- By Initializing the LFSR to All 0's and Applying the Output Sequence of the CUT to the Input of the LFSR (I_m) → The Sequence is Compressed into One Signature

LFSR Type 1 (External XOR)

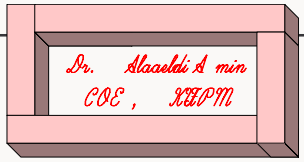
	$Q_1 = Q_3 \oplus Q_4$	Q_2	Q_3	Q_4
1	0	0	0	1
2	1	0	0	0
3	0	1	0	0
4	0	0	1	0
5	1	0	0	1
6	1	1	0	0
7	0	1	1	0
8	1	0	1	1
9	0	1	0	1
10	1	0	1	0
11	1	1	0	1
12	1	1	1	0
13	1	1	1	1
14	0	1	1	1
15	0	0	1	1
	0	0	0	1



- $P(x) = 1 + x^3 + x^4$
- Produced Sequence is Maximal → Period = 15
- Each Output has 8-1's & 7-0's
- Single Run of 4-1's, && Single Run of 3-0's
- Single run of 2-1's & Single Run of 2-0's
- Two Runs of a Single 1 && Two Runs of Single 0
- Output Sequence is the Same for all Q's



COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 13



LFSR Type 1 (External XOR)

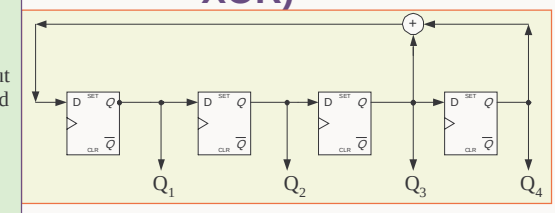
- $P(x) = 1 + x^3 + x^4$
- Output Sequence **G(x) = 1 / P(x)**

$= x^{-4} + x^{-5} + x^{-6} + x^{-7} + x^{-9} + x^{-4} + x^{-11} + x^{-12} + x^{-15}$

Note x^{-2} is an output bit Coming out Earlier than x^{-1} by onel Clock Period

- **G(x) = 100110101111000**

x^{-15} x^{-14} x^{-4}



$x^{-4} + x^{-5} + x^{-6} + x^{-7} + x^{-9} + x^{-4} + x^{-11} + x^{-12} + x^{-15}$

$1 + x^3 + x^4$ | 1

$1 + x^{-1} + x^{-4}$

$x^{-1} + x^{-4}$

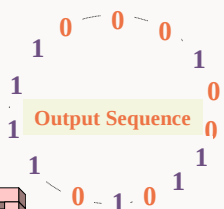
$x^{-1} + x^{-2} + x^{-5}$

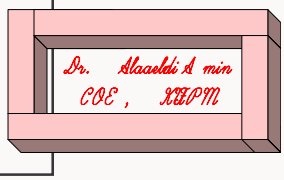
$x^{-1} + x^{-4}$

$x^{-1} + x^{-2} + x^{-5}$

$x^{-2} + x^{-4} + x^{-5}$

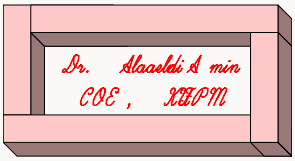
$x^{-2} + x^{-3} + x^{-6}$





COE – KFUPM Dr. Alaaeldin Amin (COE 545) Slide Number 14

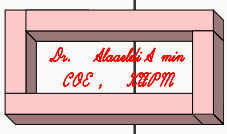
LFSR's



Theorem:

- An LFSR PRTPG Produces a *Periodic* Sequence $\{a_n\}$, such that the **Period** is the *smallest* integer ***k*** for which P(x) *Divides* $(1 + x^k)$.

Previous Example:



$(1+ x^{15})/(1 + x^3 + x^4) = x^{11}+x^{10}+x^9+x^8 + x^6 + x^4 + x^3 + 1$

Thus k=15 is the Period for the LFSR with

$$P(x)=(1 + x^3 + x^4)$$

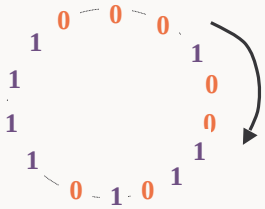
Exercise:

Prove that an LFSR with $P(x)=(1 + x^2 + x^4)$ is not Primitive → i.e. Period < 15

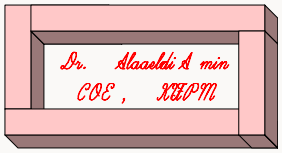
Reciprocal Polynomials

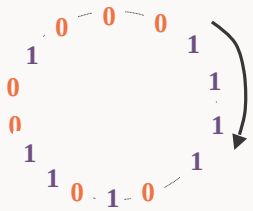
- Let P(x) be a C/C Polynomial; Define the Reciprocal Polynomial $\mathbf{P^*(x) = x^nP(1/x)}$
- An LFSR with **P*(x)** C/C Polynomial Generates the Sequence $\{b_n\}$ which is the same as $\{a_n\}$ Generated by P(x) but in Reverse Order → $b_i = a_{n-I}$
- **Example** P(x) = $(1 + x^3 + x^4)$ → **P*(x)** = $(1 + x + x^4)$
- G*(x) = $1/(1 + x + x^4)$

$$= x^{-4} + x^{-7} + x^{-8} + x^{-10} + x^{-12} + x^{-13} + x^{-14} + x^{-15}$$



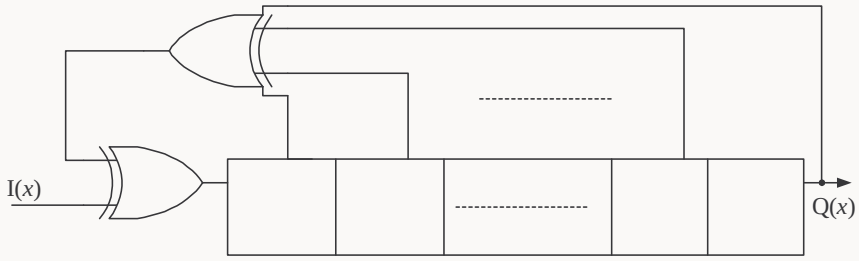
P (x) Output Sequence



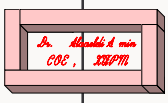


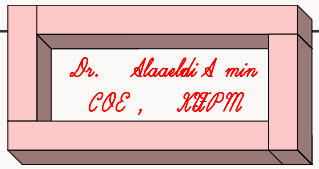
P*(x) Output Sequence

LFSR as a Signature Analyzer

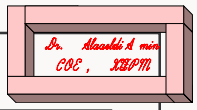


- Initial Stat = ALL 0's
- The Signature of a Sequence is the Pattern Left in the LFSR after the Last bit of Input I(x) has been Sampled G(x) = I(x)/P(x) under Zero Initial State
- **I(x) = Q(x). P(x) + R(x)** → Where *R(x) is the Signature of I(x)*
- Let's Assume Having Two Input Sequences I(x) and I'(x), where
  - I(x) = Error Free Sequence
  - I'(x) = Faulty Sequence = I(x) + e(x) where e(x) is the Error Sequence (Polynomial)



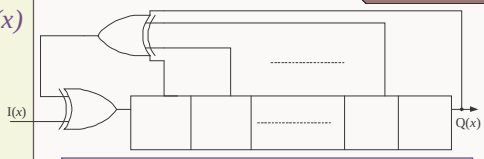


LFSR as a Signature Analyzer



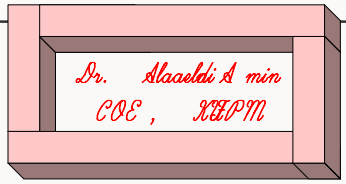
- The error will not be Detected if *I(x) & I'(x) Have the Same Signature*
- I(x) = Q(x). P(x) + **S(x)**
- I'(x) = Q'(x). P(x) + **S(x)**
- **Thus e(x) = I'(x) – I(x)**
  = P(x)(Q'(x) – Q(x))
- **For the error Signal not to be Detected it must be a Multiple of the C/C Polynomila of the LFSR P(x) → (*i.e., P(x) Divides e(x)*)**
- For an Input Sequence of k-Bits, Assuming all Possible Patterns to be Equally Likely → The Probability of Failing to Detect an Error (Aliasing) is given by:

$$\frac{\#\text{ of Undetectable errors}}{\text{Total Number of Possible Errors}}$$

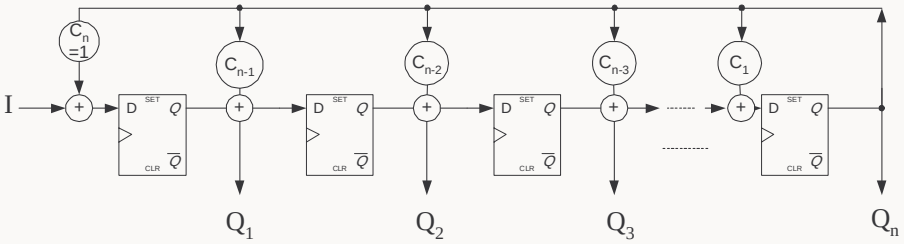


Only 1 Seq is Correct → $= \frac{2^{k-n}-1}{2^{k}-1}$

- For Long Sequences, i.e. ***k>>n***
- Aliasing Probability ≈ 2^{-n}
- Aliasing Probability Only Depends on the Length of the LFSR
- For an LFSR of 16-Bits→ Aliasing Probability ≈ 0.0015%
- For an LFSR of 8-Bits→ Aliasing Probability ≈ 0.4%

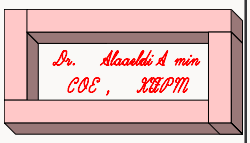


LFSR Type 2 (Intternal XOR)



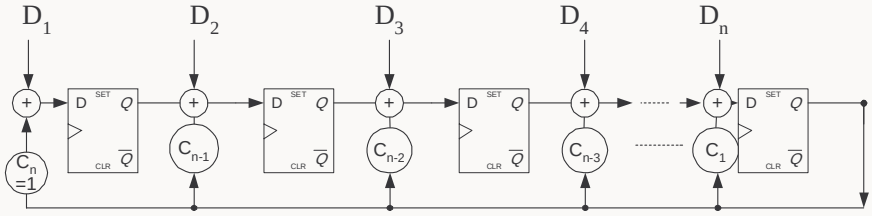
- The C/C Polynomial of this LFSR is Given by

$$P(x) = 1+\sum_{i=1}^{n} c_i x^i$$

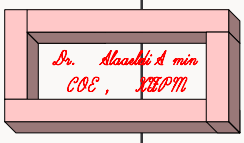


- In This Case the Signature Remainder is Given By Dividing the Input Sequence by the Reciprocal Polynomial P*(x)
- G(x) = Q(x). P*(x) + S(x) → where S(x) is the Signature

Multiple Input Signature Register (MISR)

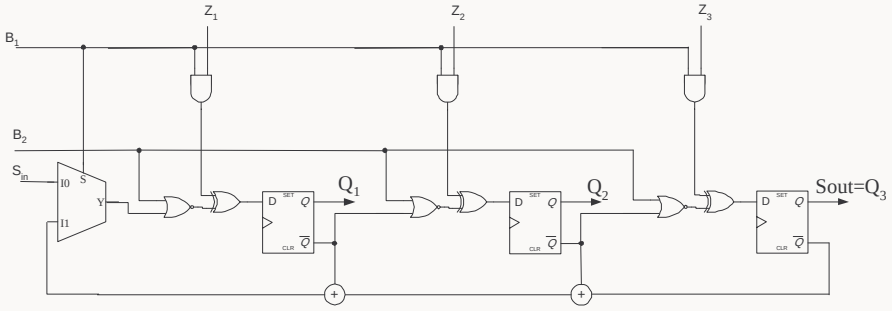


- Used to Test CUTs wit Multiple Outputs (m)
  - Use One LFSR per Output → Costly (m LFSRs)
  - Use one LFSR m Times → Long Test Application Time (Costly!!)
  - Use a MISR → Parallel Signature Analyzer

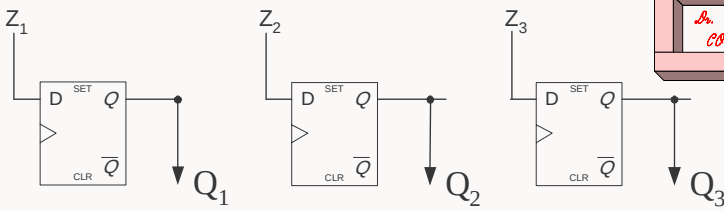


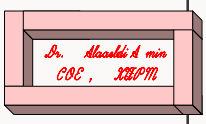
- A MISR of n-Stages Has a Probability of Aliasing (error Masking) of 2-n for Equally Likely Patterns and Long Data Sequences
- An Error Sequence e(x) is Masked if it is Multiple of the C./C Polynomial P(x)

Built-In Logic Block Observer

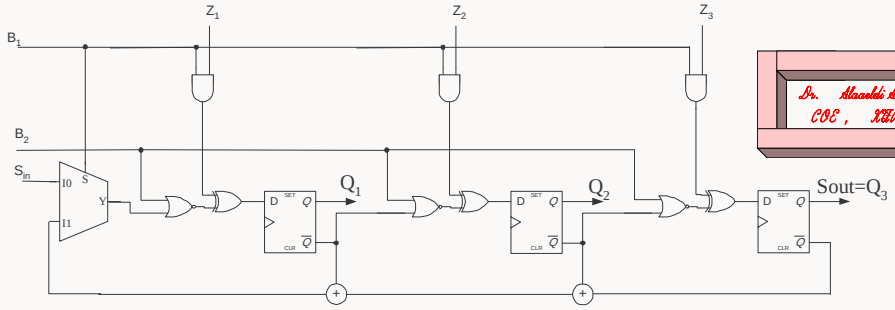


- B_1B_2=11 → Parallel Load of Input Data → Q_i ← Z_i

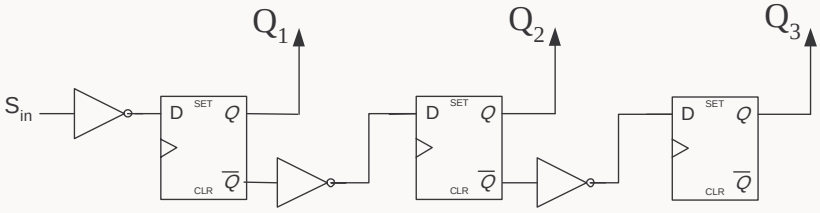




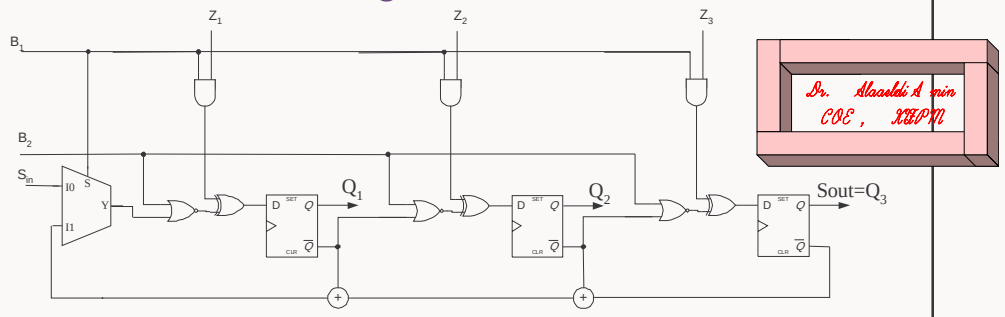
Built-In Logic Block Observer



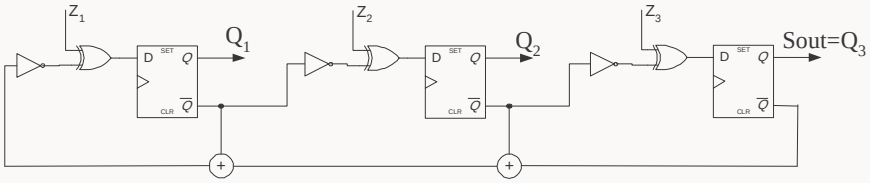
- B1B2=00 Acts a Shift Register



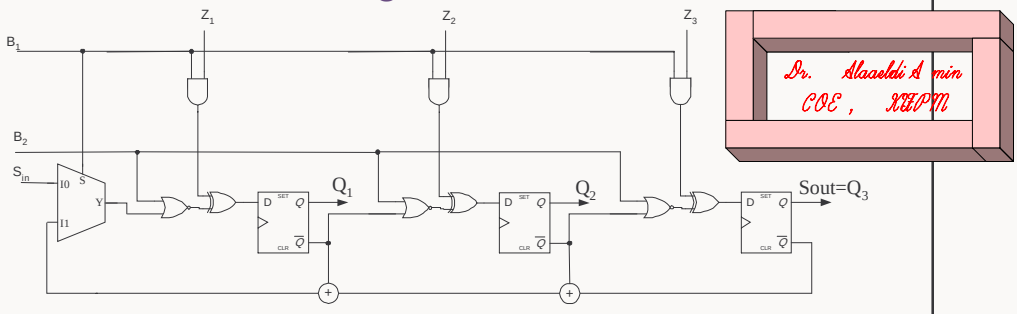
Built-In Logic Block Observer



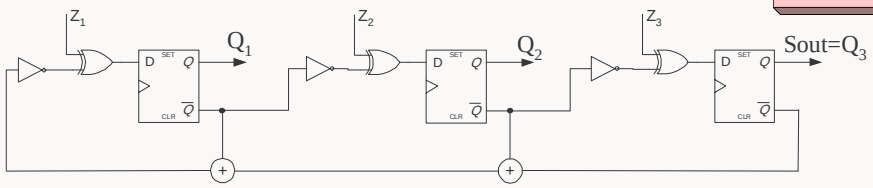
- B_1B_2=10 → Acts as a MISR if



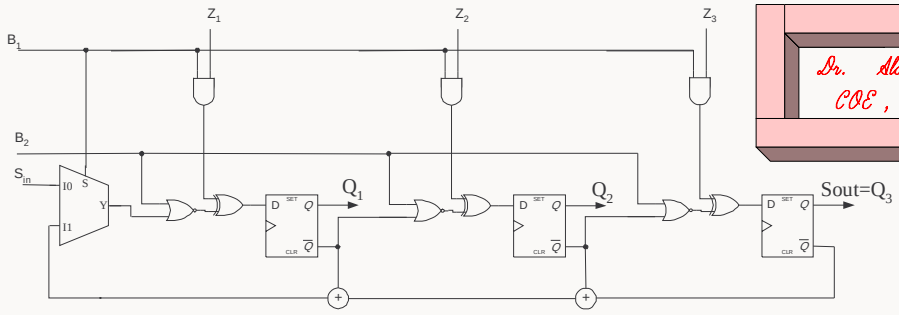
Built-In Logic Block Observer

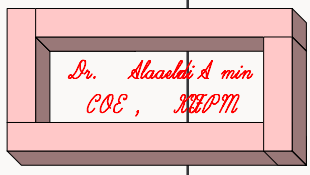


- B1B2=10 → Acts as a PRTPG if **All Z_i=0's** Or **All Z_i=1's**

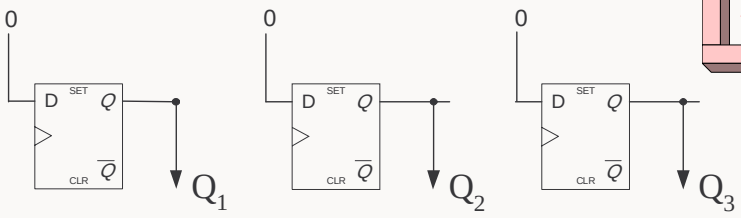


Built-In Logic Block Observer

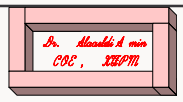




- B1B2=01 → Clears The SR (All Qi = 0)



BILBO Schemes

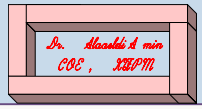


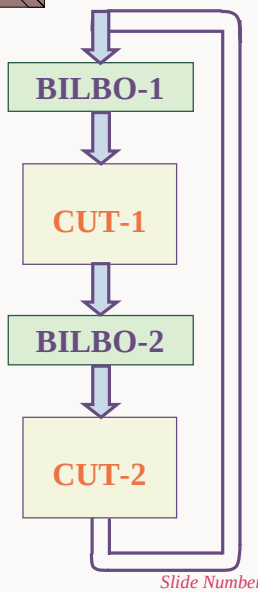
Testing CUT-1

- BILBO-1 → PRTPG
- BILBO-2→ Output Response Analyzer (ORA)

Testing CUT-2

- BILBO-2 → PRTPG
- BILBO-1→Output Response Analyzer (ORA)

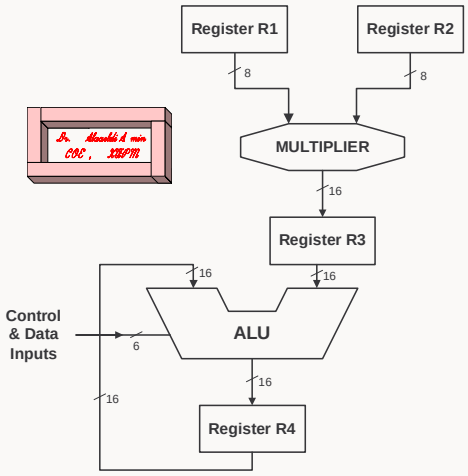




BIST CASE STUDY
Partial TMS32010 DSP Chip

OBJECTIVE:
Incorporation of BIST Structure

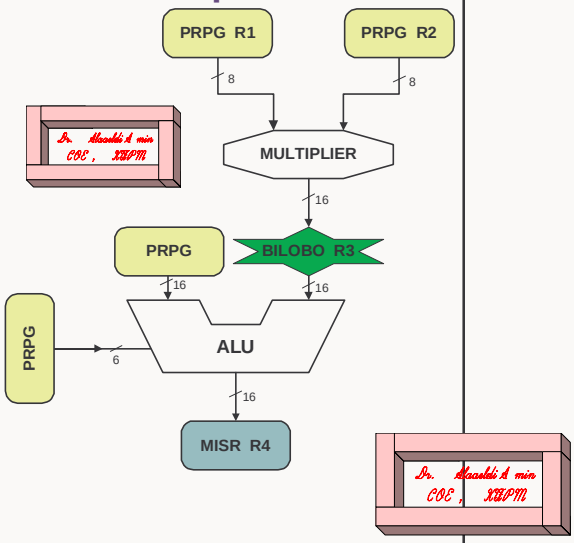
- **8 X 8 Bit Multiplier ⇒ 880 Gates**
- **16-Bit ALU (5 Control Inputs + Carry_In) ⇒ 354 Gates**
- **1255 SSF in the Multiplier**
- **574 SSF in the ALU**



Portion of the TMS32010 DSP Chip

BIST CASE STUDY
Partial TMS32010 DSP Chip

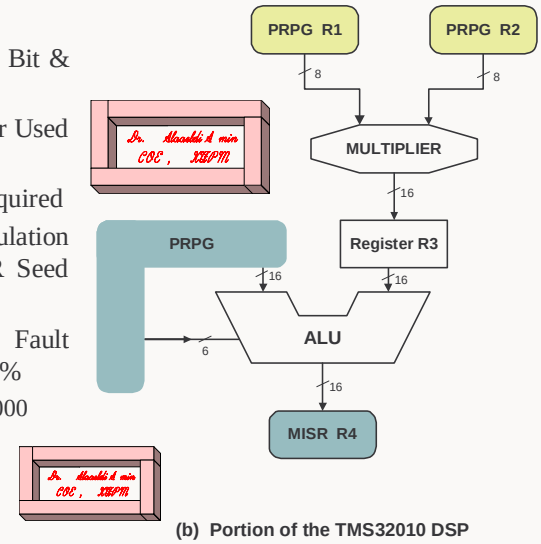
- Two Test Sessions Required
  - Multiplier Test Sessions (BILBO = MISR)
  - ALU Test Sessions (BILBO = PRPG)
- Addition of Two PRPG For ALU (16 Bit & 6-Bit)
- 20 Different Fault Simulation Runs with Different LFSR Seed Values
- Simulation Stopped After Fault Coverage Reaches 100%
  - Average # of Patterns 2177
  - Minimum # of Patterns 830
  - Maximum # of Patterns 3619
  - Fault Coverage = 100%



(a) BILBO BIST STRUCTURE

BIST CASE STUDY
Partial TMS32010 DSP Chip

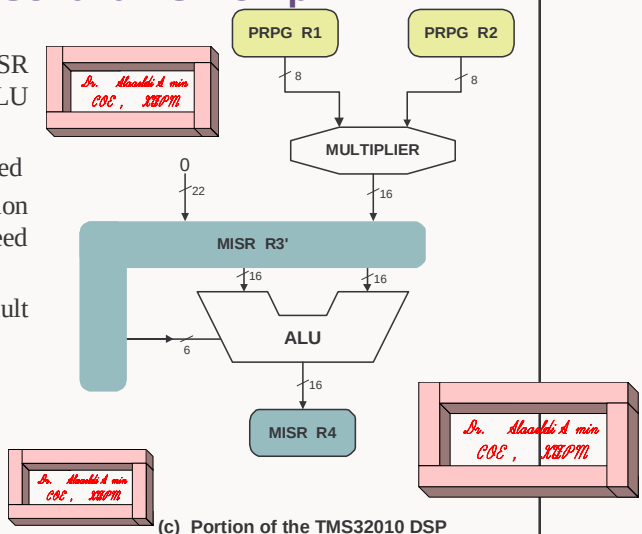
- R3 Unmodified
- 22-Bit PRPG For ALU (16 Bit & 6-Bit)
- Response Data of Multiplier Used As Test Input to ALU
- Only ONE Test Session Required
- 20 Different Fault Simulation Runs with Different LFSR Seed Values
- Simulation Stopped After Fault Coverage Saturated At 64.5%
  - Average # of Patterns > 3000
  - Fault Coverage = 64 %



(b) Portion of the TMS32010 DSP Chip

BIST CASE STUDY
Partial TMS32010 DSP Chip

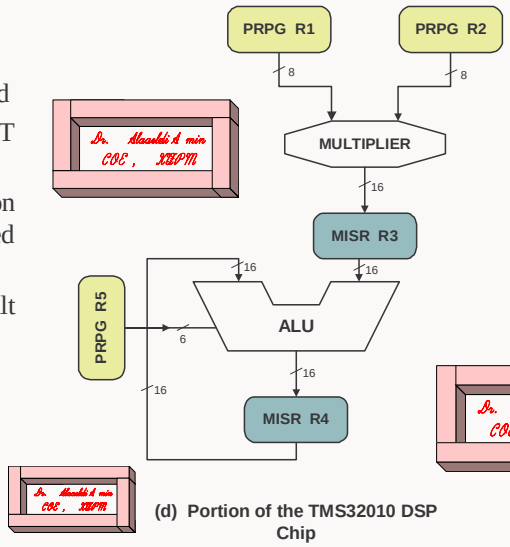
- R3 Extended as 38-Bit MISR (First 22_Bits Inputs =0) or ALU (16 Bit & 6-Bit)
- Only ONE Test Session Required
- 20 Different Fault Simulation Runs with Different LFSR Seed Values
- Simulation Stopped After Fault Coverage Reaches 100%
  - Average # of Patterns 1457
  - Minimum # of Patterns 634
  - Maximum # of Patterns 2531
  - Fault Coverage = 100%



(c) Portion of the TMS32010 DSP Chip

BIST CASE STUDY
Partial TMS32010 DSP Chip

- R3 & R4 Are MISR
- Only ONE Test Session Required
- Least Hardware BIST Configuration
- 20 Different Fault Simulation Runs with Different LFSR Seed Values
- Simulation Stopped After Fault Coverage Reaches 100%
  - Average # of Patterns 1387
  - Minimum # of Patterns 721
  - Maximum # of Patterns 2121
  - Fault Coverage = 100%



(d) Portion of the TMS32010 DSP Chip