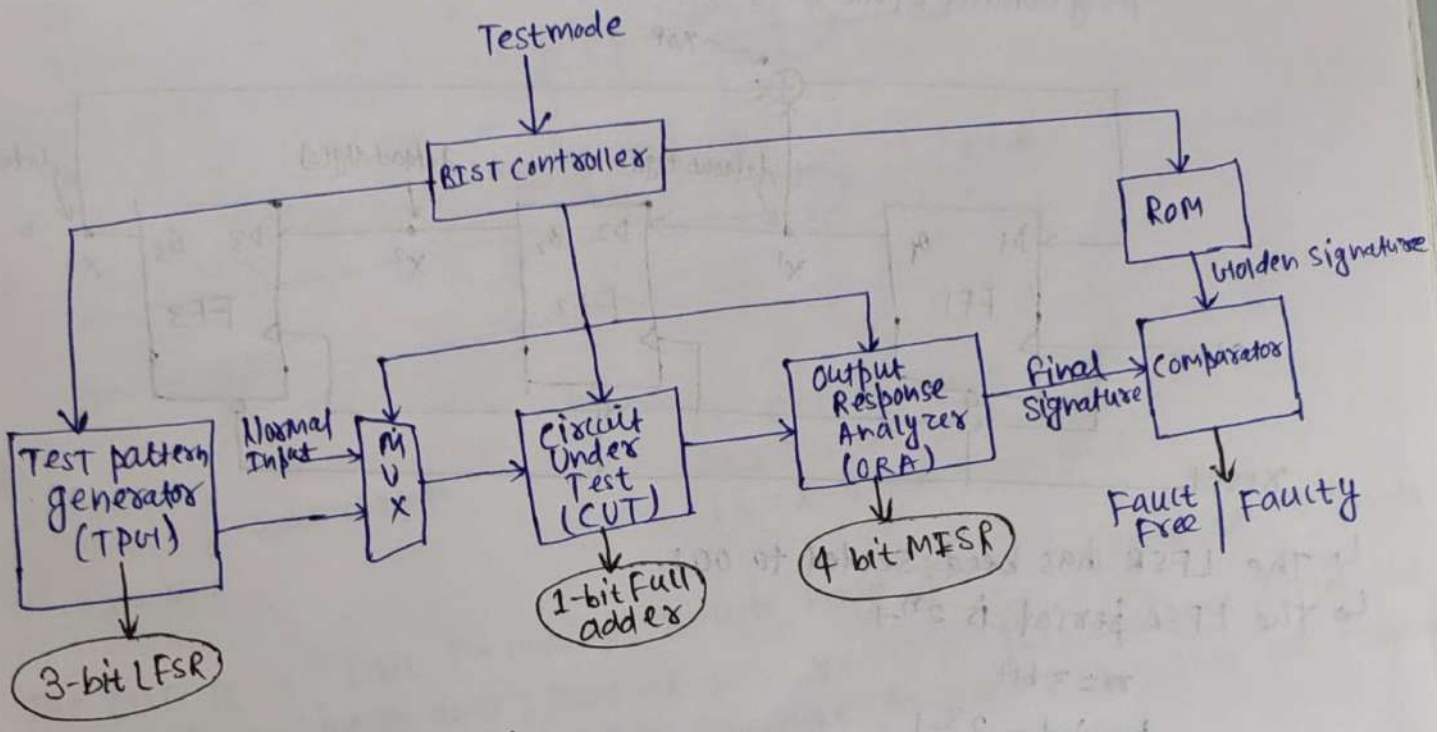
Basic Architecture Of BIST



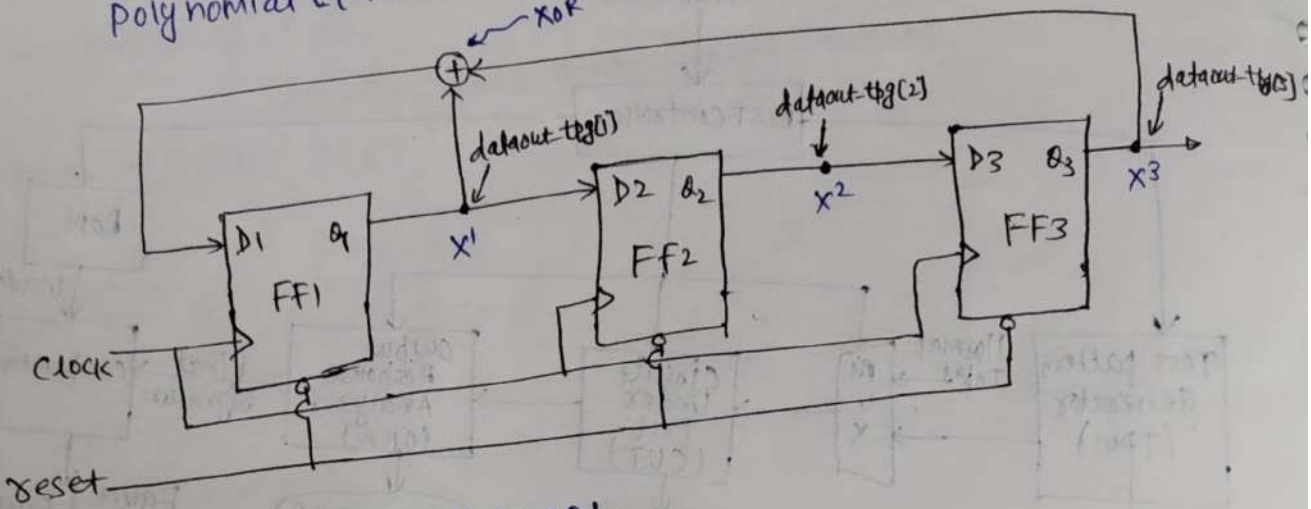
* Testmode : 0 → Normal mode
  1 → Testmode

* LFSR : Linear Feedback shift Register
* TPG : Test pattern Generator
* CUT : circuit Under Test
* ORA : Output Response Analyzer
* MISR: Multi Input Signature Register

3 bit LFSR for Test pattern Generator (TPG)

↳ 3 bit LFSR Using primitive Polynomial

Polynomial Equation : $x^3+x'+1$



↳ The LFSR has beed seeded to 001.

↳ The LFSR period is 2^n-1

$n=3$ bit

period $= 2^3-1$

$= 7$ (7 – Unique state)

* initial, dataout-tpg[3] dataout-tpg[2] dataout-tpg[1] = 001

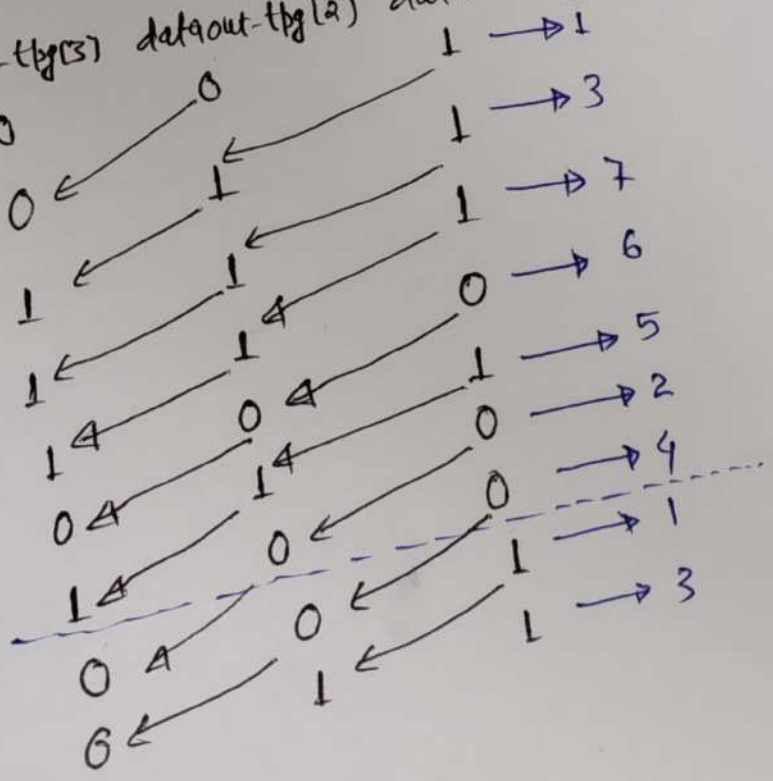
* Next state,

dataout-tpg[3] = dataout-tpg[2]

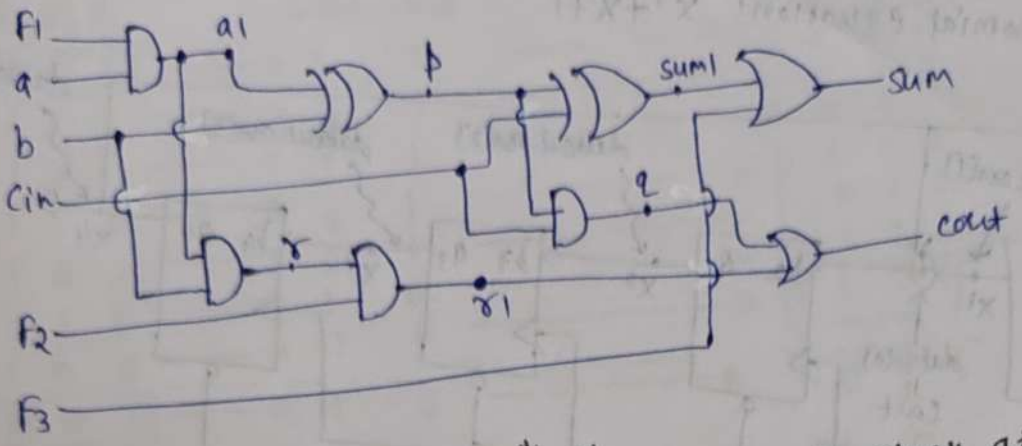
dataout-tpg[2] = dataout-tpg[1]

dataout-tpg[1] = dataout-tpg[3] ⊕ dataout-tpg[1]

* dataout-tpg[3] dataout-tpg[2] dataout-tpg[1]



1-bit Fulladder for circuit under test (CUT)



* at $f1 = 1$, $f2 = 1$, $f3 = 0$ } Normal mode

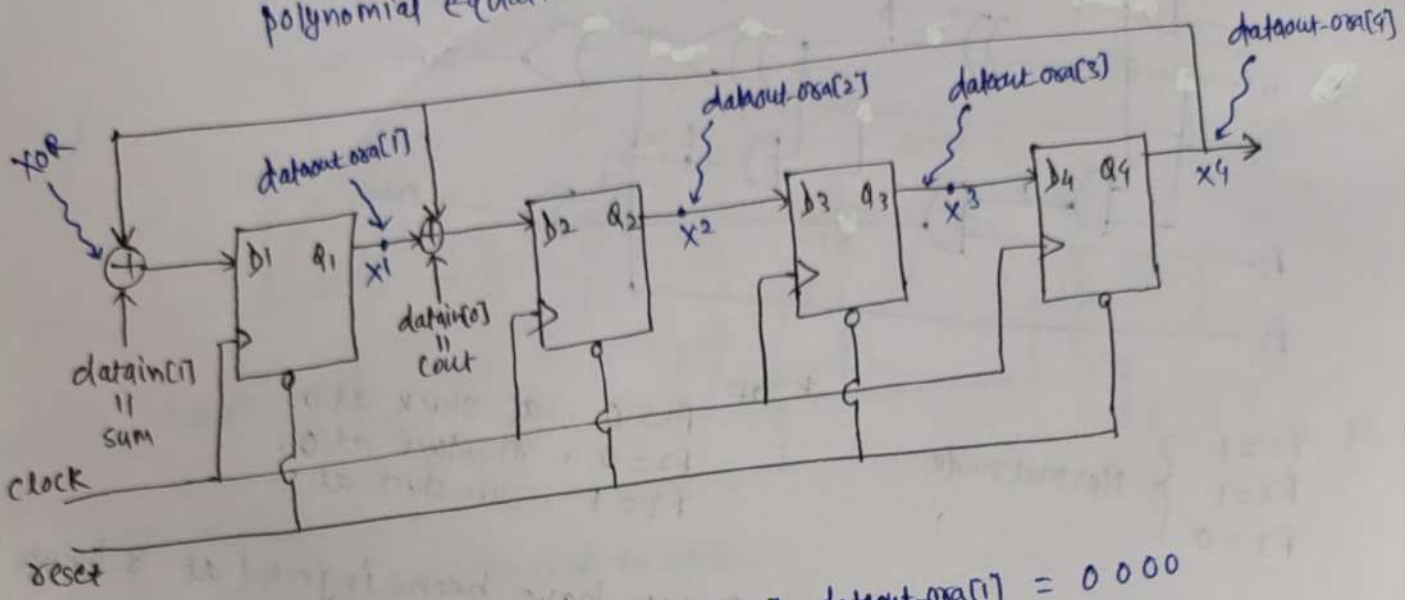
* at $f1 = 0$, a1 stuck at 0.
$f2 = 0$, r1 stuck at 0.
$f3 = 1$, sum stuck at 1.

↳ The CUT is a 1-bit fulladder. Faults have been injected at 3 points in the degin i.e a1 at 0, sum at 1, r1 at 0. The response from the MISR is compared with golden signature to check whether the fault is detected or goes undetected.

4 bit MISR for output Response Analysis (ORA)

↳ 4 bit MISR using primitive polynomial

polynomial equation: $x^4+x'+1$



* initial, dataout-ora[4] dataout-ora[3] dataout-ora[2] dataout-ora[1] = 0 0 0 0

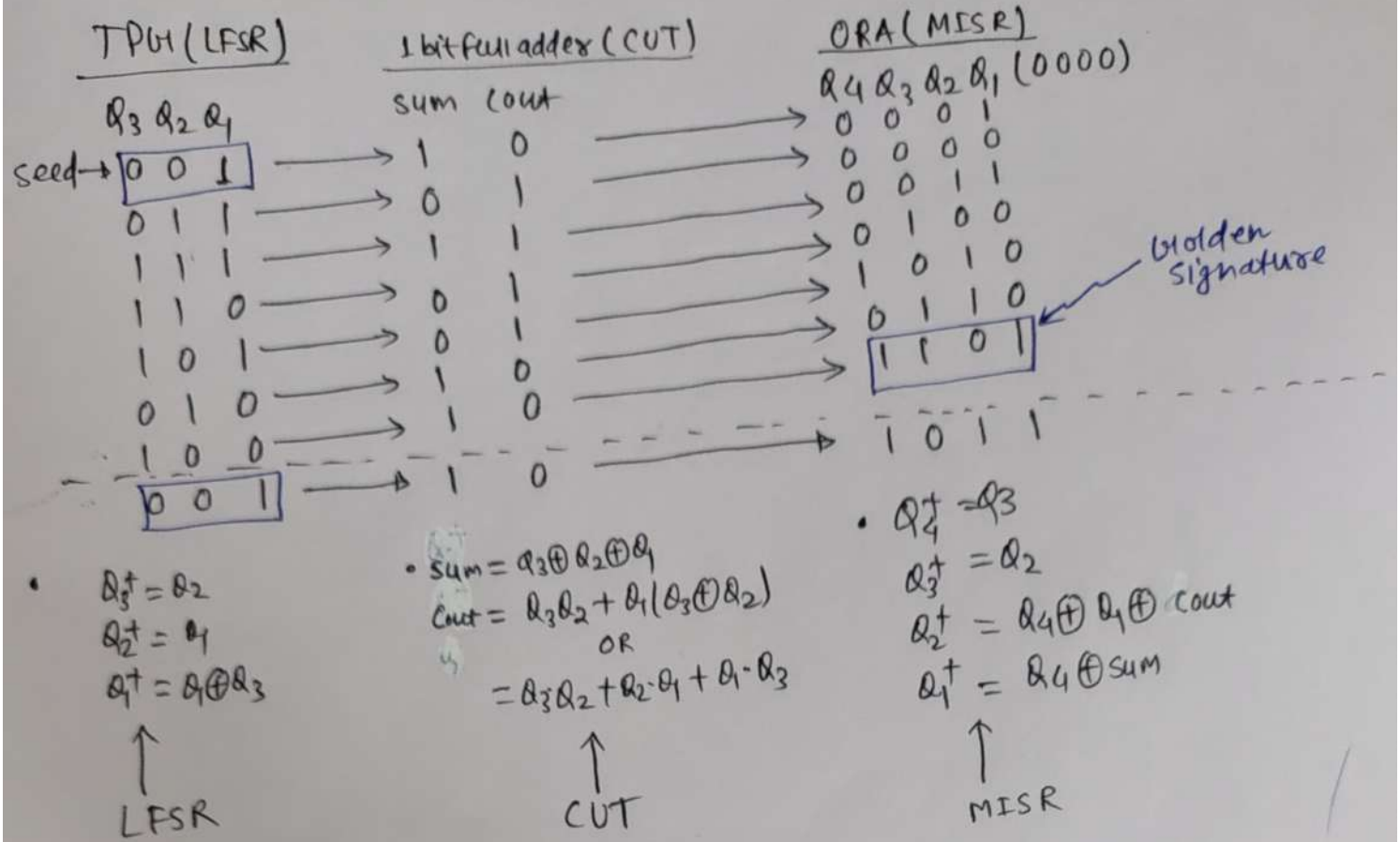
* Next state,

dataout-ora[4] = dataout-ora[3]

dataout-ora[3] = dataout-ora[2]

dataout-ora[2] = dataout-ora[4] ⊕ dataout-ora[1] ⊕ datain[0]

dataout-ora[1] = dataout-ora[4] ⊕ datain[1]


TPG (LFSR)
1 bit full adder (CUT)
ORA (MISR)
$Q_3\ Q_2\ Q_1$
seed
sum cout
$Q_4\ Q_3\ Q_2\ Q_1$ (0000)
Golden signature
• $Q_3^+ = Q_2$
$Q_2^+ = Q_1$
$Q_1^+ = Q_1 \oplus Q_3$
LFSR
• sum = $Q_3 \oplus Q_2 \oplus Q_1$
cout = $Q_3Q_2 + Q_1(Q_3 \oplus Q_2)$
OR
$= Q_3Q_2 + Q_2 \cdot Q_1 + Q_1 \cdot Q_3$
CUT
• $Q_4^+ = Q_3$
$Q_3^+ = Q_2$
$Q_2^+ = Q_4 \oplus Q_1 \oplus$ cout
$Q_1^+ = Q_4 \oplus$ sum
MISR