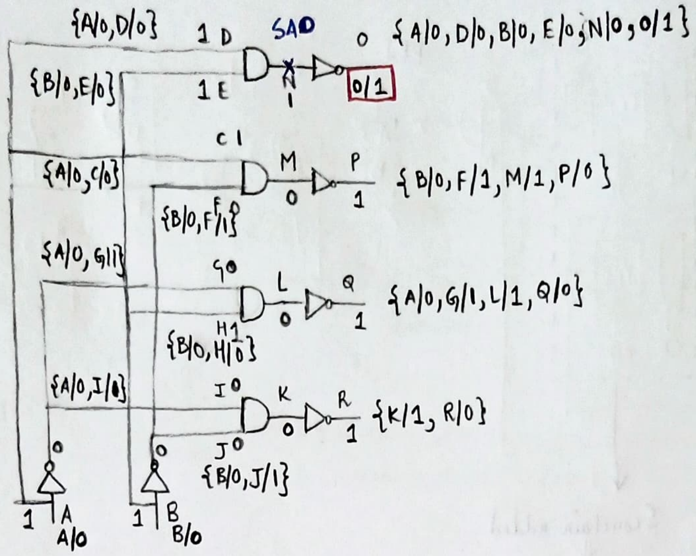
HARSHRAJ CHAUDHARI
21D040060

1.



a) Fault will always be propogated as well have inverter. We only need to excite the Fault ⇒ N ≡ 1
Set N ≡ 1, then we can work backwards to get A & B ⇒ $\boxed{\text{Test Vector } (A,B) = (1,1)}$

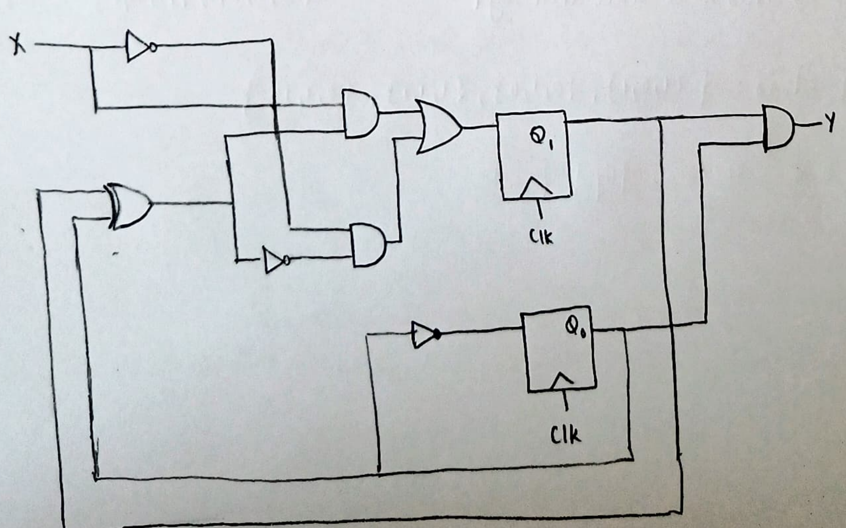
b) This can be done using deductive Fault Simulator.
The Test vector detects the following test's : {A/0, B/0, R/0, D/0, E/0, F/1, G/1, K/1, L/1, M/1, N/0, O/1, P/0, Q/0}
where X/a means line X-SAa

c) Lets do the same for different test vectors:

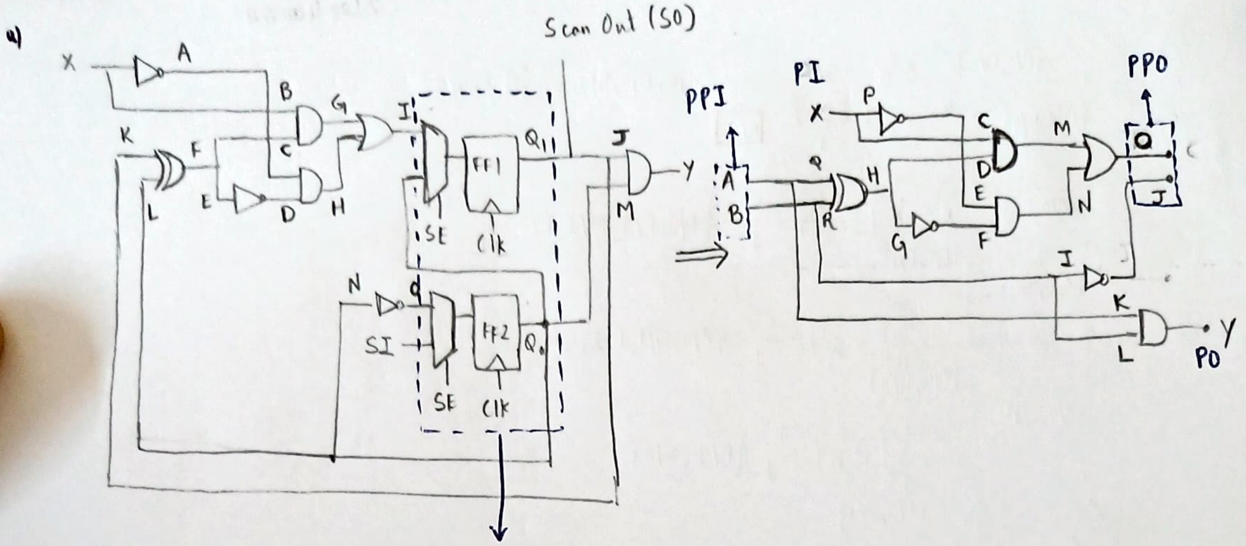
For (A,B) = (0,0) :- {A/1, B/1, C/1, H/1, I/0, J/0, K/0, L/1, M/1, N/1, O/0, P/0, Q/0, R/1} ← Still Faults are left
For (A,B) = (0,1) :- {A/1, B/0, D/1, G/0, H/0, J/1, K/1, L/0, M/1, N/1, O/0, P/0, Q/1, R/0} ← Still faults are left
⇓
We will need all 4 Test vectors for complete Fault-Coverage

$\boxed{\text{Minimum Test Vector Set :- } \{(0,0), (0,1), (1,0), (1,1)\}}$

2.



a)



Scanchain added
⇓
Converts to a Combinational circuit

b) We have 17 lines: {A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, X, Y, P, Q, R} ⇒ 40 Faults
Let take the input vector (X, A, B) as (1, 0, 1)
The faults detected by:

(X,A,B) := (1,0,1) = {X/0, A/1, B/0, C/0, D/0, H/0, I/0, J/1, K/1, L/0, M/0, O/0, Y/1, P/0, Q/1, R/0} → 16 Faults
⇓
24 Faults left

- Now lets try (X,A,B) := (0,1,0) = {X/1, A/0, B/1, C/1, D/0, F/1, G/0, H/0, I/1, J/0, M/1, L/1, N/1, O/1, Y/1, P/0, Q/0, R/1}
⇓
9 Faults remaining
- Now lets try (X,A,B) := (1,1,1) = {X/0, A/0, B/0, D/1, E/1, H/1, I/0, J/1, K/0, L/0, M/1, N/1, O/1, Y/0}
⇓
4 Faults remaining
↓
{E/0, F/0, G/1, N/0}
- Now let's try (X,A,B) := (0,1,1) ⇒ This will cover them left

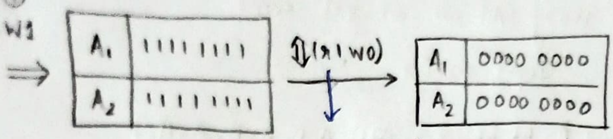
Hence the test set is: {{1,0,1}, {0,1,0}, {1,1,1}, {0,1,1}}

To detect AND-decoder and Stuck-at-all faults, we can use the following march sequence.

$\{ \Updownarrow w1, \Uparrow(r1, w0), \Downarrow(r0), \Uparrow(w1), \Downarrow(r1, w0), \Uparrow(r0))$

①

$\Updownarrow w1 \Rightarrow$



$\Updownarrow(r1\,w0)$ $\Updownarrow(r0)$

① If we donot read 1 then we have a SA0 fault

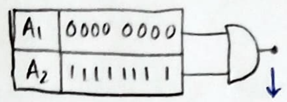
② AND-Decoder fault is also detected here

$\Downarrow$

① SA1 faults are detected

$\rightarrow$ $\Uparrow$: checks in acending order

$\Downarrow$: checks in descending order



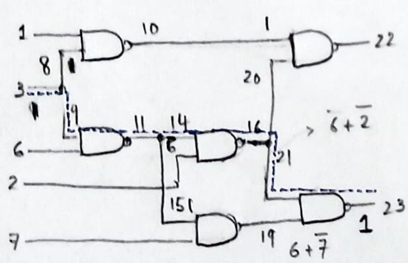
Fault detected

For the "10101010" and "(01010101" fault we will add another element in the march test:

$\{\Updownarrow(w10101010), \Updownarrow(r10101010, w01010101), \Updownarrow r01010101\}$

$\boxed{\text{The complete March test sequence is : } \{\Updownarrow w1, \Uparrow(r1w0), \Downarrow(r0), \Uparrow(w1), \Downarrow(r1, w0); \Uparrow(r0; w10101010), \Downarrow(r10101010, w01010101); \Uparrow r01010101\}}$

4.



$11 = \bar{3} + \bar{6}$

$16 = \bar{2} + \overline{11} = \bar{2} + 3\cdot6$

$23 = \overline{16} + \overline{19} = 2\cdot(\bar{3}+\bar{6}) + \overline{19} = 2(\bar{3}+\bar{6}) + 3\cdot6 + \bar{7}$

The path is: $3\uparrow, 9, 11, 14, 16, 21, 23\downarrow$ $\Rightarrow$ To detect fault all signals (on path) should flip on change of input

So when $\Rightarrow 3 \equiv "0" \rightarrow 11 = "1"$

$16 = \bar{2}$

$\Downarrow$ $23 = 2 + \bar{7} = "1"$

and

when $\Rightarrow 3 \equiv "1" \rightarrow 11 = \bar{6}$

$16 = \bar{2} + 6$

$23 = 2\cdot\bar{6} + 6 + \bar{7} = (2+\bar{7})\bar{6}$

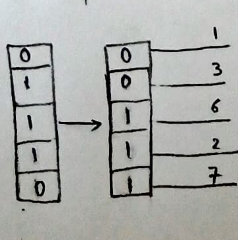
For signal toggle $\Rightarrow$ ① $\bar{6} = "0" \Rightarrow 6 = "1"$

② $\bar{2} \oplus (\bar{2}+6) = 1 \Rightarrow \bar{2} \oplus 6 = 1, 2 = "1"$

③ $23 = "1" \rightarrow \bar{7} = X$

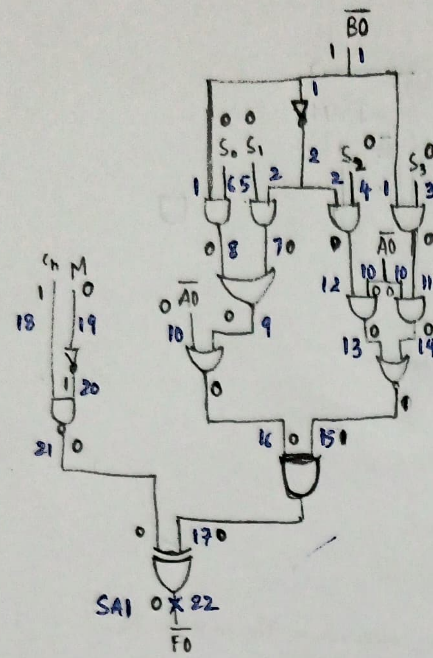
$23 = "0" \rightarrow$ already satisfied by $6 = "1"$

Therefore to excite the path these two input sequences are required.



Bonus Parts

1. Since $\overline{F0}$ is an output, it is observable already, we only need to excite it.
We don't need to consider the entire circuit ⇒



↓

Using SAT we can get some Input vector

$T = (C_n, M, S_0, S_1, S_2, S_3, A_0, B_0, A_1, B_1, A_2, B_2, A_3, B_3)$

⇓

$(1, 0, 0, 1, 0, 0, 0, 0, X, X, X, X, X, X)$

2. Like the last part we only need to excite both the faults:
Here we would have to consider the entire circuit so for brevity sake, I am not drawing the circuit.

↓

we use SAT to get our test input := $T = (1,0,1,1,1,1,0,0,0,0,0,0,0,1)$