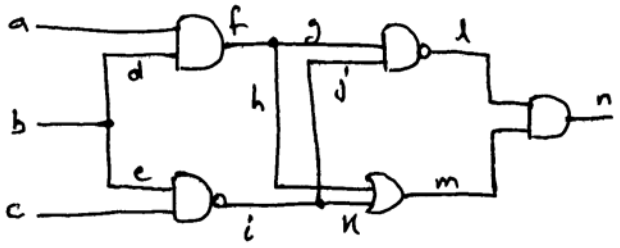
CoE 571 Dr. Aiman El-Maleh

HW#4 Solution

Q1.



(i) Controllability Costs:

line	C_0	C_1
a	0	0
b	1	1
c	0	0
d	1	1
e	1	1
f	1	2
g	1	2
h	1	2
i	2	1
j	2	1
k	2	1
l	3	1
m	3	1
n	3	2

Observability Costs:

line	O
n	0
m	1
l	1
k	2
j	3
i	2
h	3
g	2
f	2
e	2
d	2
c	3
b	2
a	3

(ii) FAN Algorithm

b s-a-0

Current obj.	Proc. Entry	Stem obj.	Head obj	Assign.	Implicat.	D-F	J-F
					b=1, d=D, e=D	{i,f} select f	
(a,1)	(a,1)		(g,1)	a=1	f=D, g=D, h=D	{c,l,m} select l	
(j,1)		(l,1)					
(l,1)	(i,1)						
(c,0)	(c,0)		(c,0)	c=0	i=1, j=1, k=1, l=$\bar{D}$, m=1, n=$\bar{D}$ Success		

Generated Test Sequence (abc) = {110}.

Decision Tree:

Verification of Generated Test:

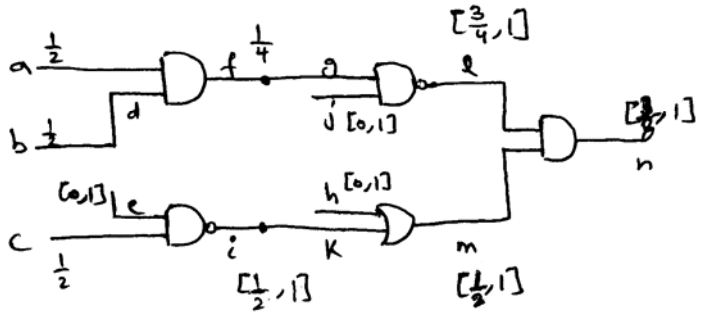
Circuit Netlist	**Signal Names**
INPUT(a) INPUT(b) INPUT(c) OUTPUT(n) f = AND(a,b) i = NAND(b,c) l = NAND(f,i) m = OR(f,i) n = AND(l,m) END	1 a 2 b 3 c 4 f 5 i 6 l 7 m 8 n 9 n_$OUTPUT
Fault b s-a-0 Fault Representation: 2 0 0 Simulated Vector: 110 Detection Status: user time 0.000000 sec sys time 0.000000 sec det faults 1 tot faults 1 coverage 1.000000	

(iii) The propagation path used in (ii) for detecting the fault b s-a-0 is $\{f, l, n\}$.

Thus, the auxiliary gate G is :

and P_G is the prob. of detecting the fault b s-a-0.

Using the cutting algorithm:



Thus, $P_G = \frac{1}{2} \times [0,1] \times [\frac{3}{4},1] \times \frac{1}{2}$

$= [0, \frac{1}{4}]$

(iv) To compute the exact fault detection probability, we need to find the number of test vectors that detect the fault b s-a-0.

$n = \overline{(f \cdot i)} \cdot (f + i) = \bar{f} i + \bar{i} f$

$= \overline{(ab)}\,\overline{(bc)} + (bc)(ab) = \bar{b} + \bar{a}\bar{c} + abc$

$n_f = 1$

The set of test vectors that detect the fault b s-a-0 is $n \oplus n_f = 1$

$$\Rightarrow \bar{b} + \bar{a}\bar{c} + abc \oplus 1 = 1$$

$$\Rightarrow \overline{[\bar{b} + \bar{a}c + abc]} = 1$$

$$\Rightarrow \bar{a}bc + ab\bar{c} = 1$$

Thus, there are two test vectors that detect the fault b s-a-0 : {011, 110}.

Hence, the exact detection probability for the fault is $\frac{2}{8} = \frac{1}{4}$.

The length of the random sequence to detect the fault depends on the confidence level required.

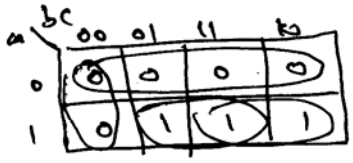
$$N = \left\lceil \frac{\ln(1-c)}{\ln(1-d_f)} \right\rceil$$

Assuming a confidence level of 95%, then

$$N = \left\lceil \frac{\ln(1-0.95)}{\ln(1-0.25)} \right\rceil = \lceil 10.41 \rceil = 11$$

As we can see, this is more than the total number of input combinations.

Q2. $F = ab + ac$



$F = ab + ac$

$\bar{F} = \bar{a} + \bar{b}\bar{c}$

Thus, the primitive cubes are:

a	b	c	F
1	1	x	1
1	x	1	1
0	x	x	0
x	0	0	0

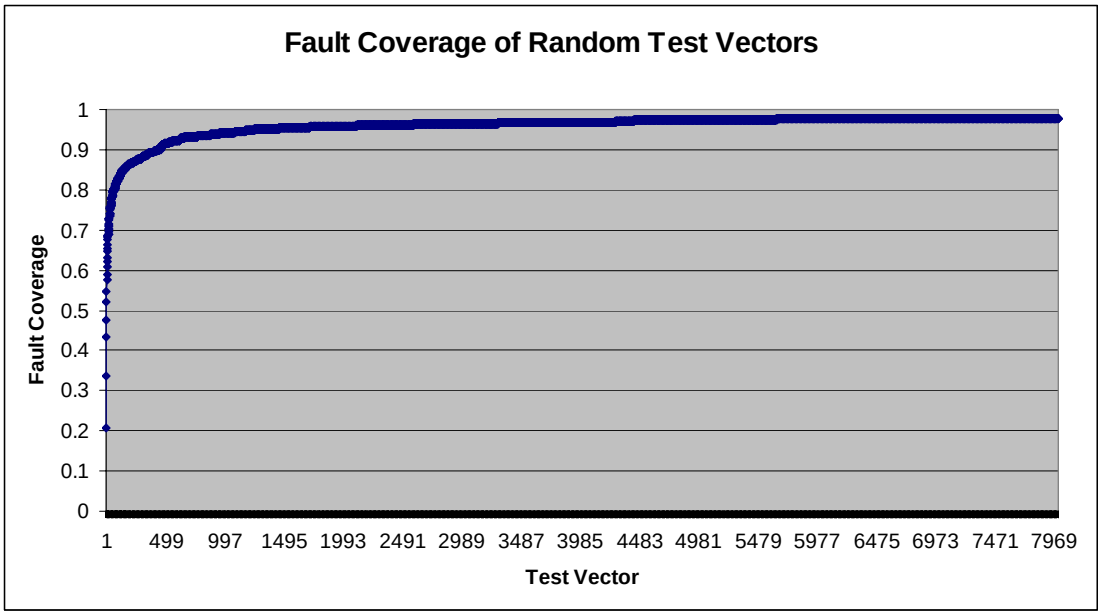
The propagation D-cubes are:

a	b	c	F
D	1	x	D
D	x	1	D
1	D	0	D
1	0	D	D
D	D	x	D
D	x	D	D
D	D	D	D
1	D	D	D

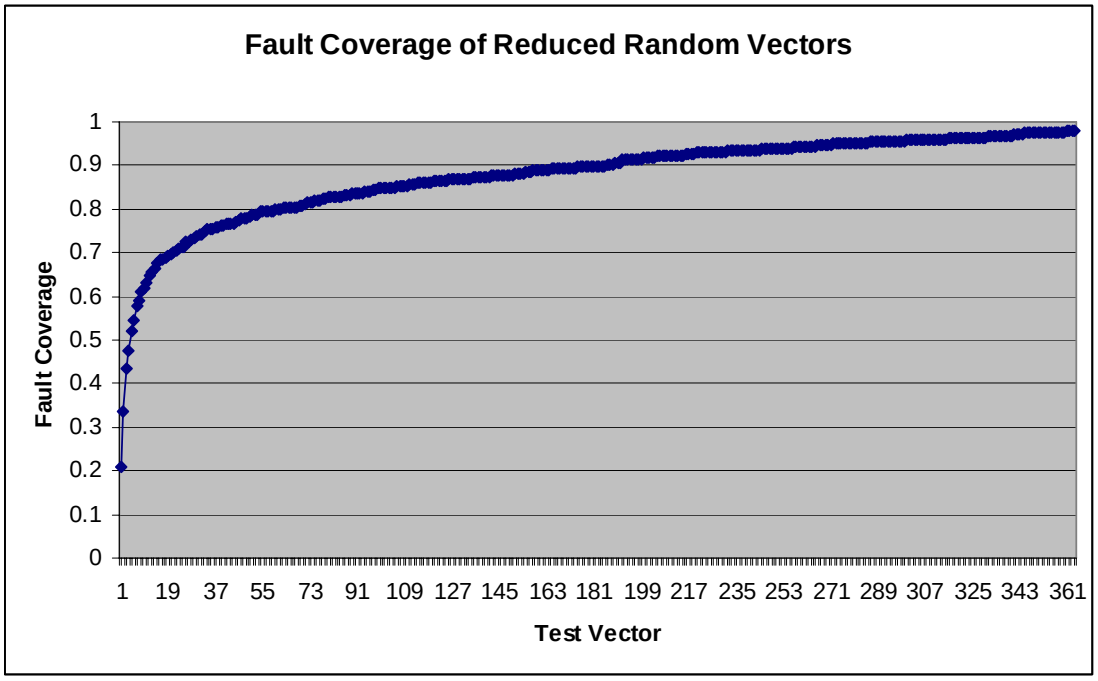
Additional propagation D-cubes are also obtained by replacing every D by $\bar{D}$.

Q3.

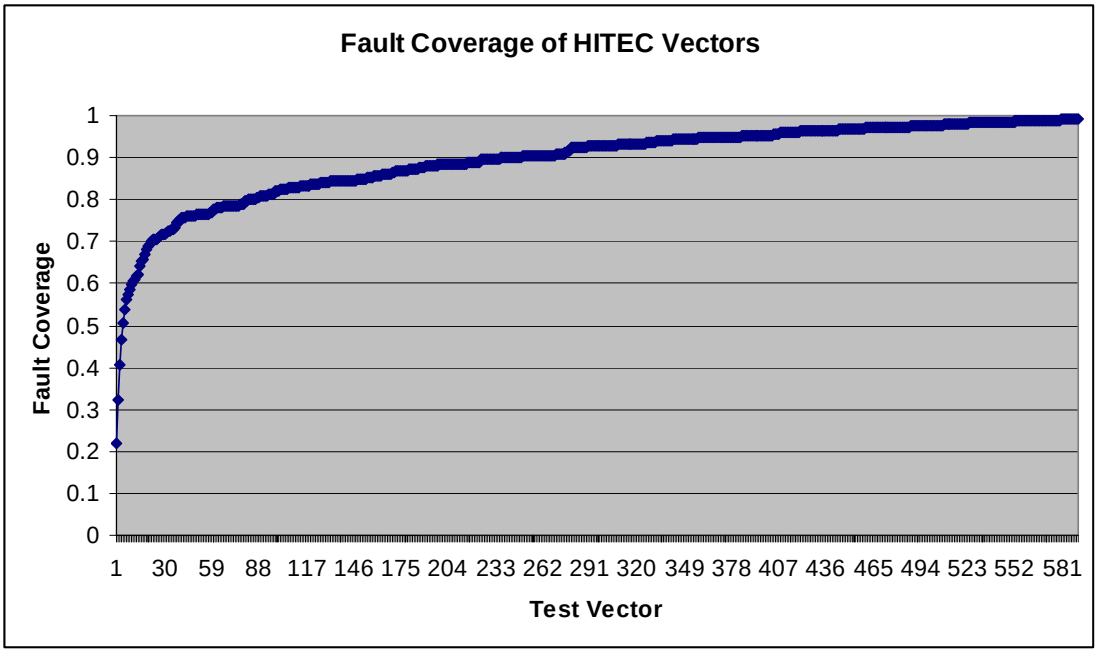
(i) Fault Coverage of 8000 random test vectors:



(ii) After eliminating test vectors that do not detect any fault, the test set is reduce from 8000 test vectors to 364 test vectors. The fault coverage of the reduced test vectors is given below:



(iii) After running HITEC for two iterations, 590 vectors were generated with a fault coverage of 99.13%. The fault coverage of the HITEC generated test vectors is given below:



(iv) Comparison of parts (ii) & (iii):

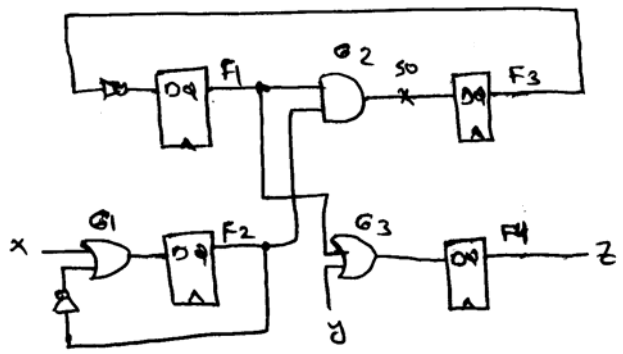
	Reduced Random Test set	HITEC Test Set
Test Length	364	590 (359)
Fault Coverage	97.87	99.13
CPU Time	7s	3.9s

Based on the results, we can see that with 8000 random test vectors generated, only 364 were useful in detecting additional undetected faults and the fault coverage achieved is 97.87. Fault simulating 8000 vectors took 7s. More fault coverage is achieved by HITEC with less CPU time. However, in general, ATPG takes more CPU time than fault simulation. It should be observed that the test vectors generated by HITEC are not reduced and after reducing them the test length becomes 359 test vectors.

(v) Two phase approach:

By fault simulating 4000 random test vectors before running deterministic ATPG using HITEC, the fault coverage achieved is 99.13, the test length is 4122 and the CPU time is 1.45s. After reducing the test set, the test length is 413. Thus, we can see that the two-phase ATPG approach achieved the same fault coverage as the deterministic ATPG with less CPU time but with more vector count than the reduced test set generated by HITEC.

Q4.



T=0

To excite the fault, we need to set G2=1. This implies that we need to set F1=1 and F2=1 at the previous frame. This also propagates a D at the input of F3.

T=1

At this frame, F3 = D and this propagates $\bar{D}$ at the input of F1.

T=2

F1 = $\bar{D}$. We can either propagate across G2 or G3. We choose G3 since it has better observability. This requires that y=0 and propagates $\bar{D}$ at the input of F4

T=3

F4 = $\bar{D}$ and z = $\bar{D}$ and the fault is observed.

Next, we justify the values backward in time frames.

T=-1

F1=1 ⇒ F3=0. F2=1 is justified by x=1

T = -2

F3 = 0 can be justified through either F1 = 0 or F2 = 0. We select F2 = 0

T = -3

F2 = 0 is justified through x = 0 and F2 = 1

T = -4

F2 = 1 is justified throug x = 1 and we succeded to find an initializing test sequence.

The generated test sequence is:

(xy) = {1x, 0x, xx, 1x, xx, xx, x0, xx}.

Verification of Generated Test:

Circuit Netlist	**Signal Names**
INPUT(x) INPUT(y) OUTPUT(z) F1 = DFF(F3b) F2 = DFF(G1) F3 = DFF(G2) F4 = DFF(G3) G1 = OR(x, F2b) G2 = AND(F1, F2) G3 = OR(F1, y) F2b = NOT(F2) F3b = NOT(F3) z = BUF(F4)	1 x 2 y 3 F1 4 F2 5 F3 6 F4 7 G2 8 G3 9 F2b 10 F3b 11 z 12 z_$OUTPUT 13 G1
Fault G2 s-a-0 Fault Representation: 7 0 0 Simulated test sequence: {1x,0x,xx,1x,xx,xx,x0,xx}	Detection Status: user time 0.000000 sec sys time 0.000000 sec det faults 1 tot faults 1 coverage 1.000000

(ii) Test Generation using HITEC:

2
10
00
01
11
01
00
00
10
END