VLSI Testing Homework #1

系所：資訊應用研究所

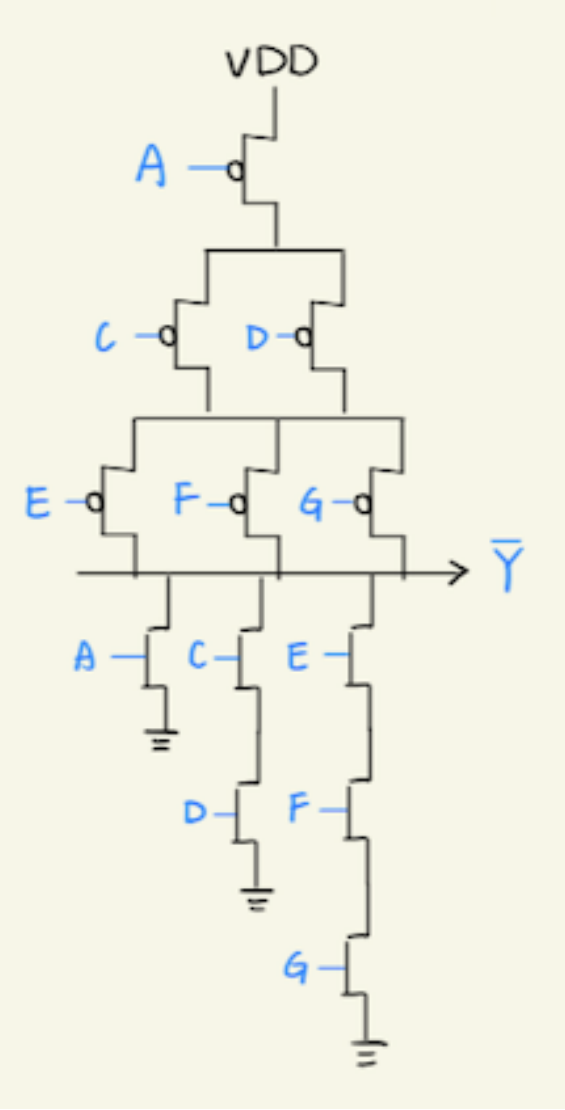
姓名：林元泰

學號：109065520

(a) Draw the transistor schematic for a logic cell with the following functionality:

Y = (A + CD + EFG)’

I use pull-up network (PUN) and pull-down network (PWN) to implement this transistor.



(b) **Write a C or C++ program to generate the truth table of this logic cell**. Your program should print out the response of each of the $2^6 = 64$ input combinations. Also, report the **cardinalities of the ON-SET and the OFF-SET**, respectively.

ON-SET (A, C, D, E, F, G) :

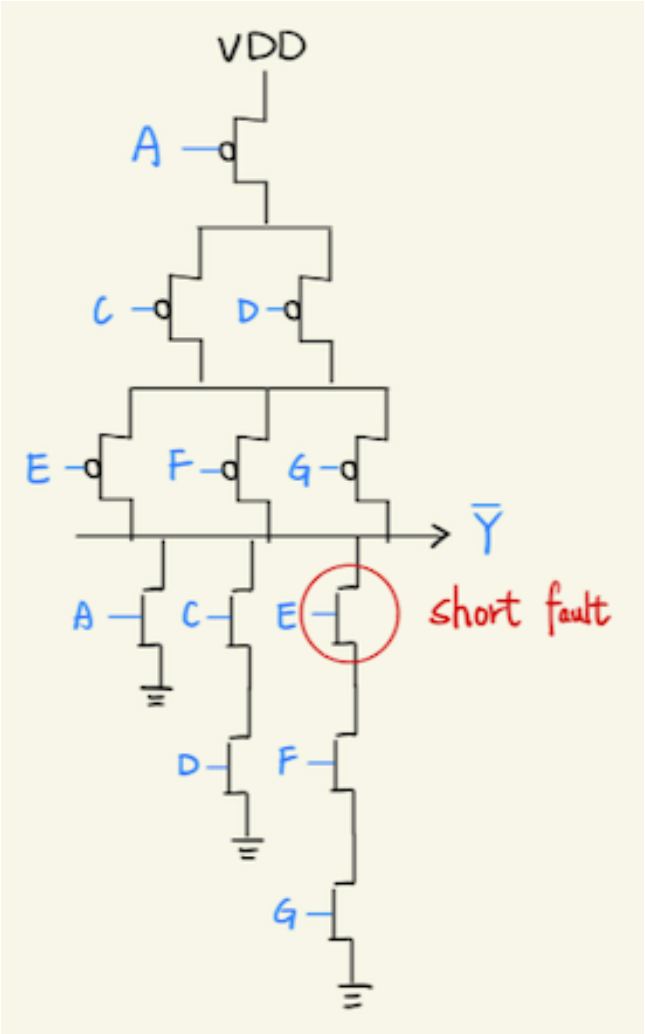
```
( 0, 0, 0, 0, 0, 0 )   ( 0, 0, 1, 1, 1, 0 )
( 0, 1, 0, 0, 0, 0 )   ( 0, 0, 0, 0, 0, 1 )
( 0, 0, 1, 0, 0, 0 )   ( 0, 1, 0, 0, 0, 1 )
( 0, 0, 0, 1, 0, 0 )   ( 0, 0, 1, 0, 0, 1 )
( 0, 1, 0, 1, 0, 0 )   ( 0, 0, 0, 1, 0, 1 )
( 0, 0, 1, 1, 0, 0 )   ( 0, 1, 0, 1, 0, 1 )
( 0, 0, 0, 0, 1, 0 )   ( 0, 0, 1, 1, 0, 1 )
( 0, 1, 0, 0, 1, 0 )   ( 0, 0, 0, 0, 1, 1 )
( 0, 0, 1, 0, 1, 0 )   ( 0, 1, 0, 0, 1, 1 )
( 0, 0, 0, 1, 1, 0 )   ( 0, 0, 1, 0, 1, 1 )
( 0, 1, 0, 1, 1, 0 )
```

OFF-SET (A, C, D, E, F, G) :

(1, 0, 0, 0, 0, 0)	(1, 0, 1, 0, 0, 1)
(1, 1, 0, 0, 0, 0)	(0, 1, 1, 0, 0, 1)
(1, 0, 1, 0, 0, 0)	(1, 1, 1, 0, 0, 1)
(0, 1, 1, 0, 0, 0)	(1, 0, 0, 1, 0, 1)
(1, 1, 1, 0, 0, 0)	(1, 1, 0, 1, 0, 1)
(1, 0, 0, 1, 0, 0)	(1, 0, 1, 1, 0, 1)
(1, 1, 0, 1, 0, 0)	(0, 1, 1, 1, 0, 1)
(1, 0, 1, 1, 0, 0)	(1, 1, 1, 1, 0, 1)
(0, 1, 1, 1, 0, 0)	(1, 0, 0, 0, 1, 1)
(1, 1, 1, 1, 0, 0)	(1, 1, 0, 0, 1, 1)
(1, 0, 0, 0, 1, 0)	(1, 0, 1, 0, 1, 1)
(1, 1, 0, 0, 1, 0)	(0, 1, 1, 0, 1, 1)
(1, 0, 1, 0, 1, 0)	(1, 1, 1, 0, 1, 1)
(0, 1, 1, 0, 1, 0)	(0, 0, 0, 1, 1, 1)
(1, 1, 1, 0, 1, 0)	(1, 0, 0, 1, 1, 1)
(1, 0, 0, 1, 1, 0)	(0, 1, 0, 1, 1, 1)
(1, 1, 0, 1, 1, 0)	(1, 1, 0, 1, 1, 1)
(1, 0, 1, 1, 1, 0)	(0, 0, 1, 1, 1, 1)
(0, 1, 1, 1, 1, 0)	(1, 0, 1, 1, 1, 1)
(1, 1, 1, 1, 1, 0)	(0, 1, 1, 1, 1, 1)
(1, 0, 0, 0, 0, 1)	(1, 1, 1, 1, 1, 1)
(1, 1, 0, 0, 0, 1)	

(c) A transistor could have unexpected **short fault** (meaning that the source and drain are connected at all times independent of the value of its gate). Consider one such type of logic cell with E-controlled nMOS having a short fault. **Modify your program to show the truth table for this faulty cell**.

`E-controlled nMOS have a short fault`

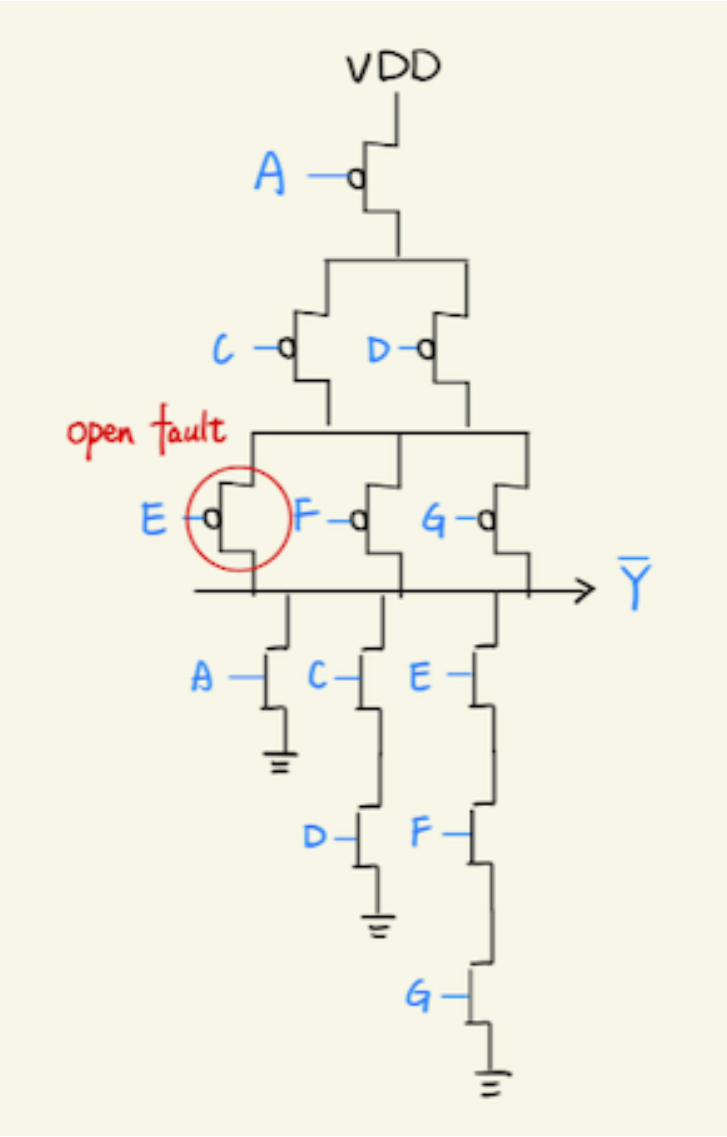


A	C	D	E	F	G	Y
0	0	0	0	0	0	1
1	0	0	0	0	0	0
0	1	0	0	0	0	1
1	1	0	0	0	0	0
0	0	1	0	0	0	1
1	0	1	0	0	0	0
0	1	1	0	0	0	0
1	1	1	0	0	0	0
0	0	0	1	0	0	1
1	0	0	1	0	0	0
0	1	0	1	0	0	1
1	1	0	1	0	0	0
0	0	1	1	0	0	1
1	0	1	1	0	0	0
0	1	1	1	0	0	0
1	1	1	1	0	0	0
0	0	0	0	1	0	1
1	0	0	0	1	0	0
0	1	0	0	1	0	1
1	1	0	0	1	0	0
0	0	1	0	1	0	1
1	0	1	0	1	0	0
0	1	1	0	1	0	0
1	1	1	0	1	0	0
0	0	0	1	1	0	1
1	0	0	1	1	0	0
0	1	0	1	1	0	1
1	1	0	1	1	0	0
0	0	1	1	1	0	1
1	0	1	1	1	0	0
0	1	1	1	1	0	0
1	1	1	1	1	0	0

0	0	0	0	0	1	1
1	0	0	0	0	1	0
0	1	0	0	0	1	1
1	1	0	0	0	1	0
0	0	1	0	0	1	1
1	0	1	0	0	1	0
0	1	1	0	0	1	0
1	1	1	0	0	1	0
0	0	0	1	0	1	1
1	0	0	1	0	1	0
0	1	0	1	0	1	1
1	1	0	1	0	1	0
0	0	1	1	0	1	1
1	0	1	1	0	1	0
0	1	1	1	0	1	0
1	1	1	1	0	1	0
0	0	0	0	1	1	X
1	0	0	0	1	1	0
0	1	0	0	1	1	X
1	1	0	0	1	1	0
0	0	1	0	1	1	X
1	0	1	0	1	1	0
0	1	1	0	1	1	0
1	1	1	0	1	1	0
0	0	0	1	1	1	0
1	0	0	1	1	1	0
0	1	0	1	1	1	0
1	1	0	1	1	1	0
0	0	1	1	1	1	0
1	0	1	1	1	1	0
0	1	1	1	1	1	0
1	1	1	1	1	1	0

(d) A transistor could have unexpected **open fault** as well (meaning that the source and drain are disconnected at all times independent of the value of its gate). Consider one such type of logic cell with E-controlled pMOS having an open fault. **Modify your program to show the truth table for this faulty cell**.

`E-controlled pMOS have an open fault`



A	C	D	E	F	G	Y
0	0	0	0	0	0	1
1	0	0	0	0	0	0
0	1	0	0	0	0	1
1	1	0	0	0	0	0
0	0	1	0	0	0	1
1	0	1	0	0	0	0
0	1	1	0	0	0	0
1	1	1	0	0	0	0
0	0	0	1	0	0	1
1	0	0	1	0	0	0
0	1	0	1	0	0	1
1	1	0	1	0	0	0
0	0	1	1	0	0	1
1	0	1	1	0	0	0
0	1	1	1	0	0	0
1	1	1	1	0	0	0
0	0	0	0	1	0	1
1	0	0	0	1	0	0
0	1	0	0	1	0	1
1	1	0	0	1	0	0
0	0	1	0	1	0	1
1	0	1	0	1	0	0
0	1	1	0	1	0	0
1	1	1	0	1	0	0
0	0	0	1	1	0	1
1	0	0	1	1	0	0
0	1	0	1	1	0	1
1	1	0	1	1	0	0
0	0	1	1	1	0	1
1	0	1	1	1	0	0
0	1	1	1	1	0	0
1	1	1	1	1	0	0

0	0	0	0	0	1	1
1	0	0	0	0	1	0
0	1	0	0	0	1	1
1	1	0	0	0	1	0
0	0	1	0	0	1	1
1	0	1	0	0	1	0
0	1	1	0	0	1	0
1	1	1	0	0	1	0
0	0	0	1	0	1	1
1	0	0	1	0	1	0
0	1	0	1	0	1	1
1	1	0	1	0	1	0
0	0	1	1	0	1	1
1	0	1	1	0	1	0
0	1	1	1	0	1	0
1	1	1	1	0	1	0
0	0	0	0	1	1	Z
1	0	0	0	1	1	0
0	1	0	0	1	1	Z
1	1	0	0	1	1	0
0	0	1	0	1	1	Z
1	0	1	0	1	1	0
0	1	1	0	1	1	0
1	1	1	0	1	1	0
0	0	0	1	1	1	0
1	0	0	1	1	1	0
0	1	0	1	1	1	0
1	1	0	1	1	1	0
0	0	1	1	1	1	0
1	0	1	1	1	1	0
0	1	1	1	1	1	0
1	1	1	1	1	1	0

(e) An open fault is more deterministically testable than the short fault but requires two-pattern tests. Modify your program to **list all 2-pattern tests** for the faulty cell described in part (d). Report the **total number of tests** you derived.

```
2-pattern tests ( A, C, D, E, F, G ) :
( 1, 0, 0, 0, 1, 1 ) => ( 0, 0, 0, 0, 1, 1 )
( 0, 0, 0, 1, 1, 1 ) => ( 0, 0, 0, 0, 1, 1 )
( 1, 1, 0, 0, 1, 1 ) => ( 0, 1, 0, 0, 1, 1 )
( 0, 1, 1, 0, 1, 1 ) => ( 0, 1, 0, 0, 1, 1 )
( 0, 1, 0, 1, 1, 1 ) => ( 0, 1, 0, 0, 1, 1 )
( 1, 0, 1, 0, 1, 1 ) => ( 0, 0, 1, 0, 1, 1 )
( 0, 1, 1, 0, 1, 1 ) => ( 0, 0, 1, 0, 1, 1 )
( 0, 0, 1, 1, 1, 1 ) => ( 0, 0, 1, 0, 1, 1 )

Total number of test : 8
```