Digital Logic Design - Chapter 3

1S. For each value of N (when N=2, 3, 4 or 5):

a) Determine number of unique functions possible with N independent variables?
b) If the function and its complement are considered as only one function, determine number of unique functions possible with N independent variables?

Solution:

a) n=2

# of different Functions = 2^{2^n} = 16 functions
# of different Functions excluding complements = 8 functions

b) n=3

# of different Functions = 2^{2^n} = 256 functions
# of different Functions excluding complements = 128 functions

c) n=4

# of different Functions = 2^{2^n} = 65,536 functions
# of different Functions excluding complements = 32,768 functions

d) n=5

# of different Functions = 2^{2^n} = 4,294,967,296 functions
# of different Functions excluding complements = 2,147,483,648 functions

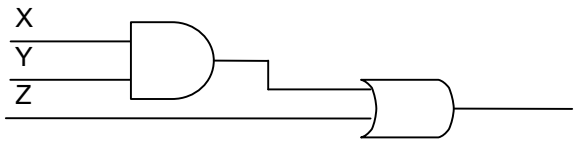
1U. For each value of N (when N=6 or 7):

a) Determine number of unique functions possible with N independent variables?
b) If the function and its complement are considered as only one function, determine number of unique functions possible with N independent variables?

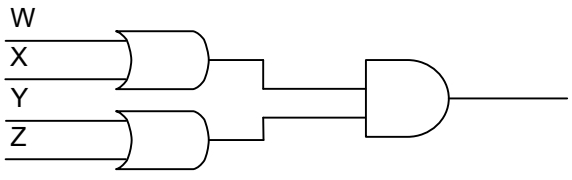
Solution:

2S. Analyze the logic circuits shown below in order to obtain the logic function for each circuit.

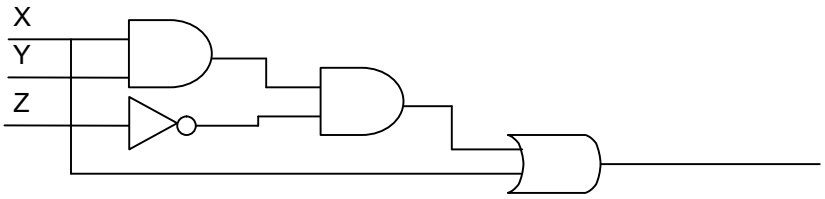
a)



b)

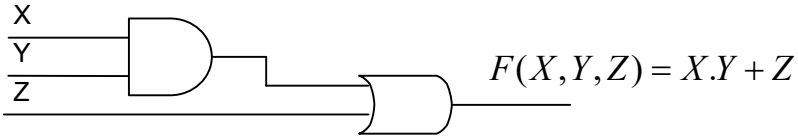


c)

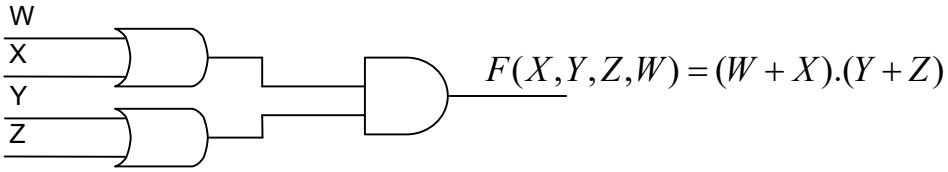


Solution:

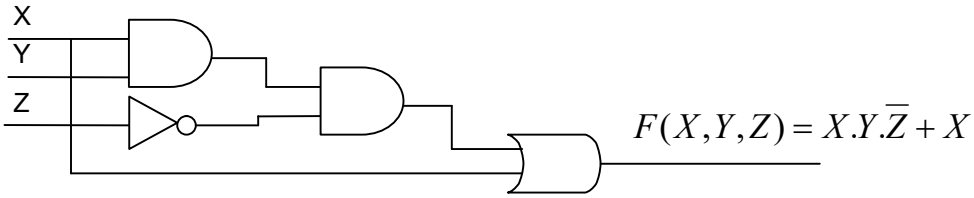
a)



b)

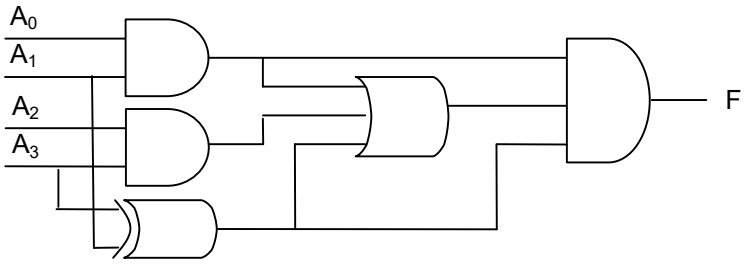


c)



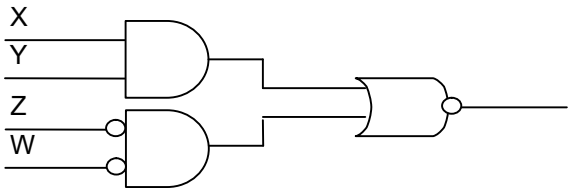
<Can simplify to X.>

2U. Analyze the logic circuits shown below in order to obtain the logic function for each circuit. Write the minimized logic function implemented by the following circuit:

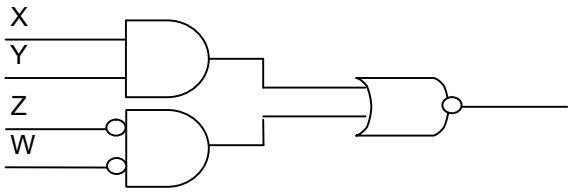


Solution:

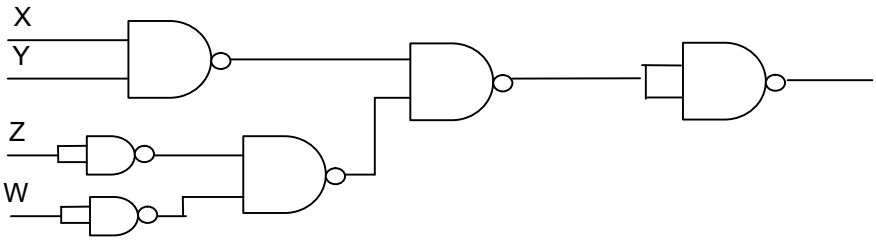
3S. Draw the following circuit using only NAND gates.



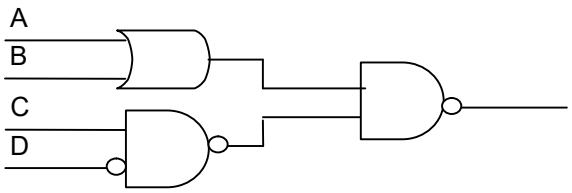
Solution:



$$F(X,Y,Z) = \overline{X.Y + \overline{Z}.\overline{W}}$$



3U. Draw the following circuit using only NOR gates.

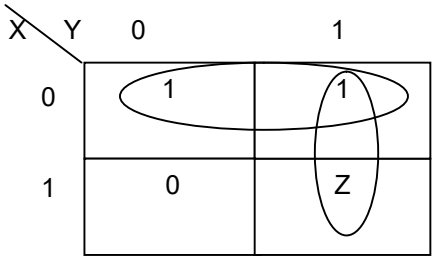


Solution:

4S. Obtain minimum SOP expression for the following function using a compressed K-map.

$F(X,Y,Z) = \overline{X} + X.Y.Z \quad where \quad m = m(X,Y)$ *"Compressed around Z"*

Solution:



$$F(X,Y,Z) = \overline{X} + Z.Y$$

4U. Obtain minimum SOP expression for the following function using a compressed K-map.

$F(X,Y,Z,W) = \overline{X} + X.Y.Z + X.Y.\overline{Z}W + \overline{X}.Y.\overline{Z}W \quad where \quad m = m(X,Y,Z)$ *"Compressed around W"*

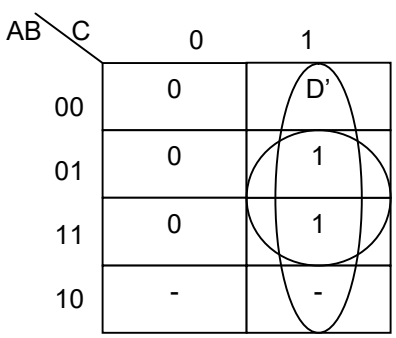
Solution:

5S. The function F1 specified by the following truth table was modified three times. Find a minimum SOP expression for each of the functions F2 through F4. (A dash in the table represents a "don't care" output.)

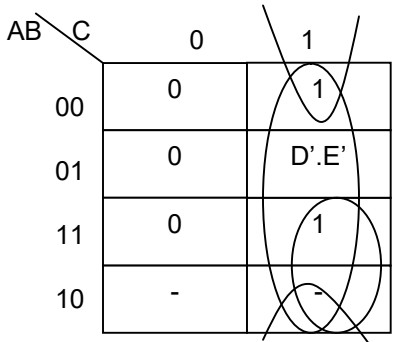
A	B	C	F1	F2	F3	F4
0	0	0	0	0	0	0
0	0	1	1	D'	1	1

0	1	0	0	0	0	0
0	1	1	1	1	D'.E'	1
1	0	0	-	-	-	-
1	0	1	-	-	-	-
1	1	0	0	0	0	0
1	1	1	1	1	1	D'+E'

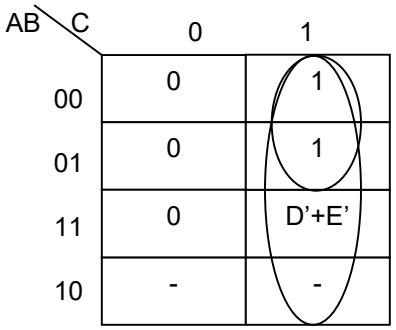
Solution:



F_2= C.D' + B.C



F_3= C.D'.E' + A.C + B'.C



F_4= C.D' + C.E'+ A'.C

5U. Find minimum SOP expression for each of the output functions F1 through F4. (A dash in the table represents a "don't care" output.)

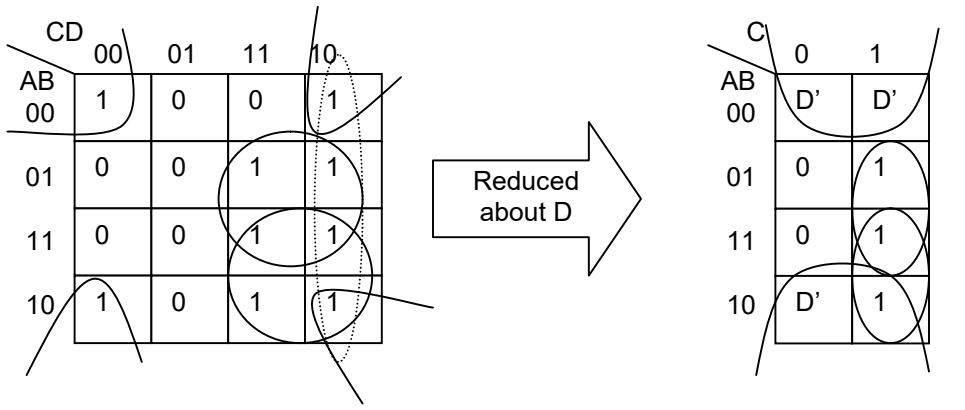
X	Y	Z	F1	F2	F3	F4
0	0	0	0	W'	0	0
0	0	1	1	-	W.U	1
0	1	0	0	0	0	0
0	1	1	1	1	-	1
1	0	0	-	-	-	-
1	0	1	-	-	-	-
1	1	0	0	0	0	0
1	1	1	1	1	1	W+U'

Solution:

6S. For the following K-map, show a reduced K-map one size smaller. Write the minimum SOP expression for the normal and reduced map (compressed around D).

AB \ CD	00	01	11	10
00	1	0	0	1
01	0	0	1	1
11	0	0	1	1
10	1	0	1	1

Solution:



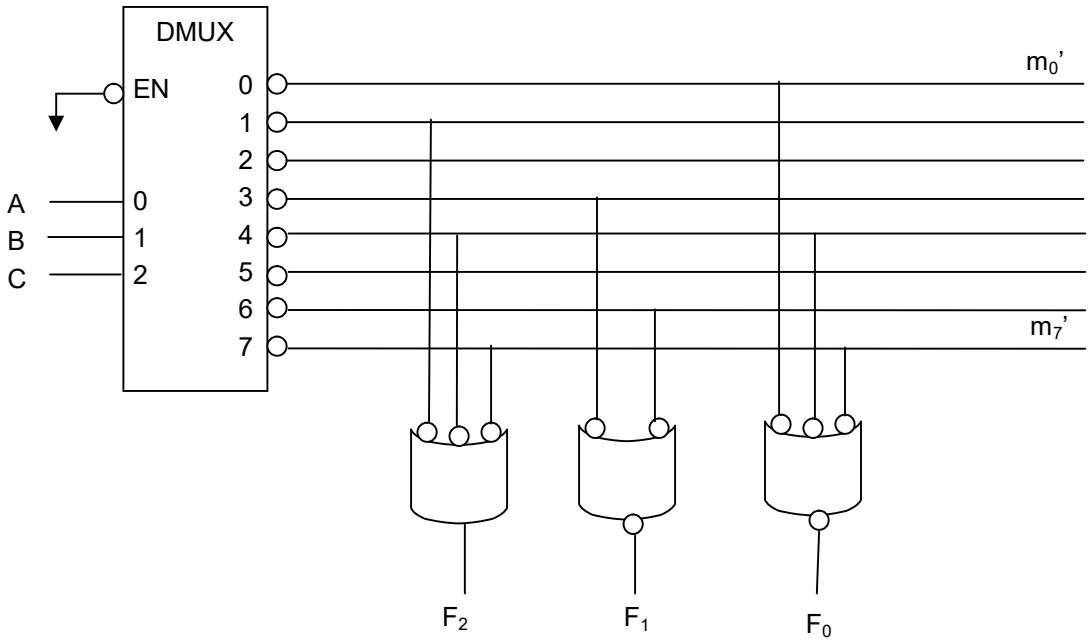
$F_3(A,B,C,D) = B'.D' + B.C + A.C$ $F_3(A,B,C) = B'.D' + B.C + A.C$

6U. For the following K-map, show a reduced K-map one size smaller. Write the minimum SOP expression for the normal and reduced map(compressed around D).

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	0	1	1	1
10	0	0	0	0

Solution:

7S. Analyze the following design to obtain the output Boolean functions,f(A,B,C) in compact min term form.



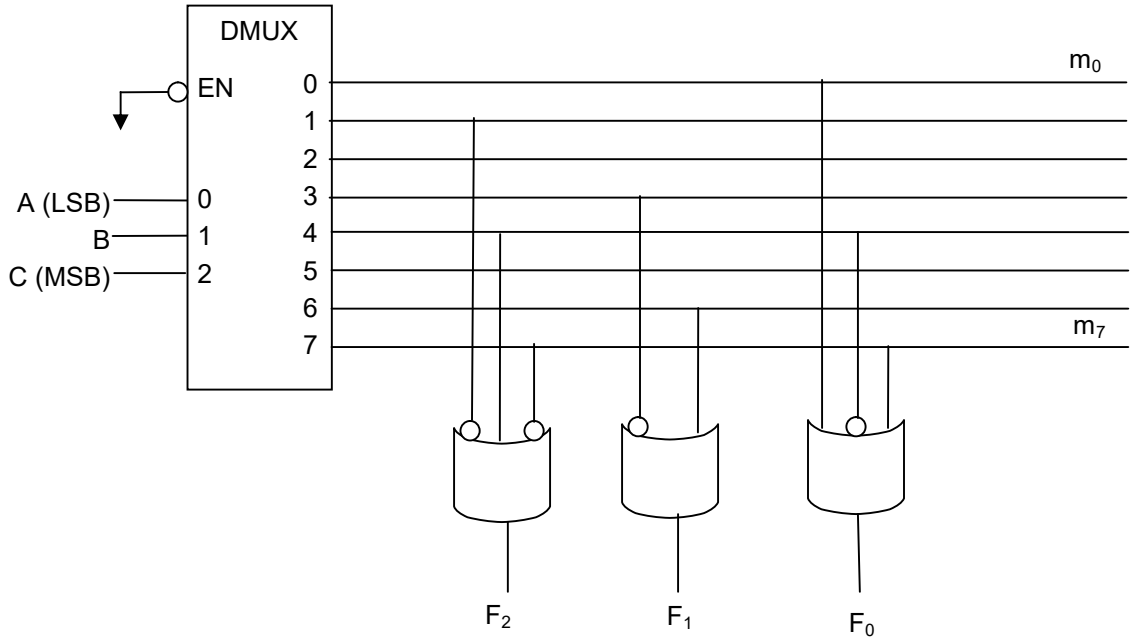
Solution:

- F0'=Σ(0,4,7)
  F0=Σ(1,2,3,5,6)

- F1'=Σ(3,6)
  F1=Σ(0,1,2,4,5,7)

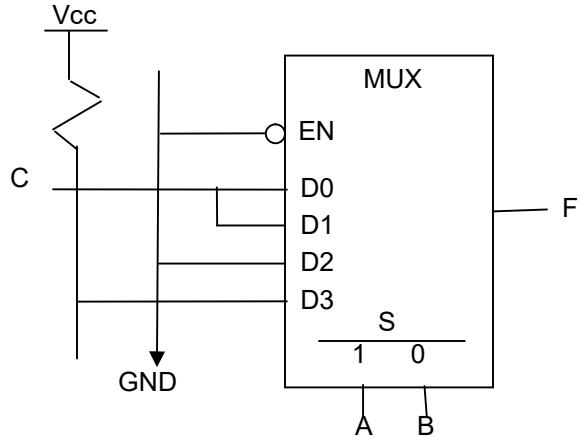
- F2=Σ(1,4,7)

7U. Analyze the following design to obtain the output Boolean functions in compact min term form.



Solution:

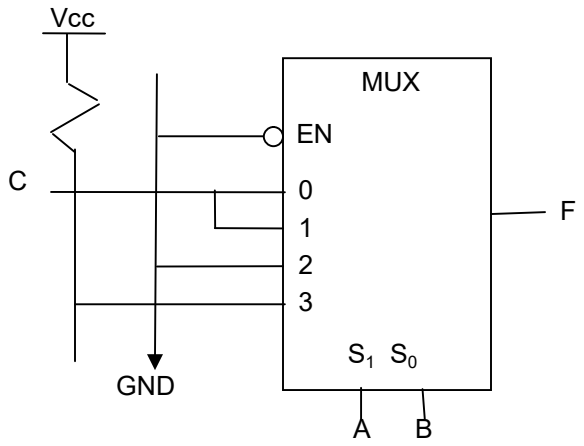
8S. Analyze the multiplexer circuit shown below to obtain its output function, f(A,B,C), in compact minterm form.



MUX Functional Truth Table:

A	B	F
0	0	D0
0	1	D1
1	0	D2
1	1	D3

Solution:

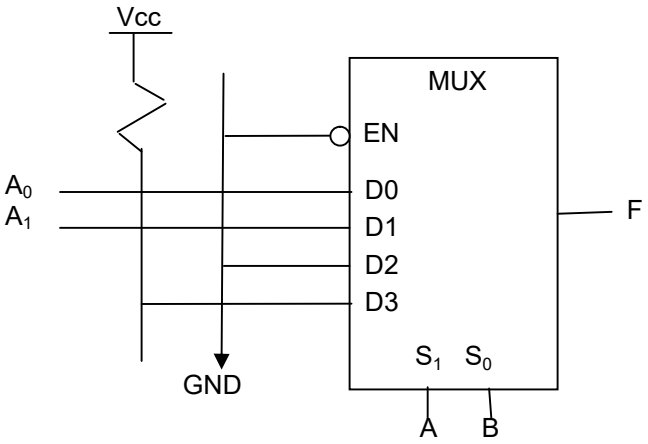


S_1	S_0	F
0	0	C
0	1	C
1	0	0
1	1	1

A	B	C	F	minterm
0	0	0	0	0
0	0	1	1	1
0	1	0	0	2
0	1	1	1	3
1	0	0	0	4
1	0	1	0	5
1	1	0	1	6
1	1	1	1	7

$F = \Sigma(1,3,6,7)$

8U. Analyze the multiplexer circuit shown below to obtain its output function, $f(A,B,A_1,A_0)$, in compact min-term form.



MUX Functional Truth Table:

S_1	S_0	F
0	0	D0
0	1	D1
1	0	D2
1	1	D3

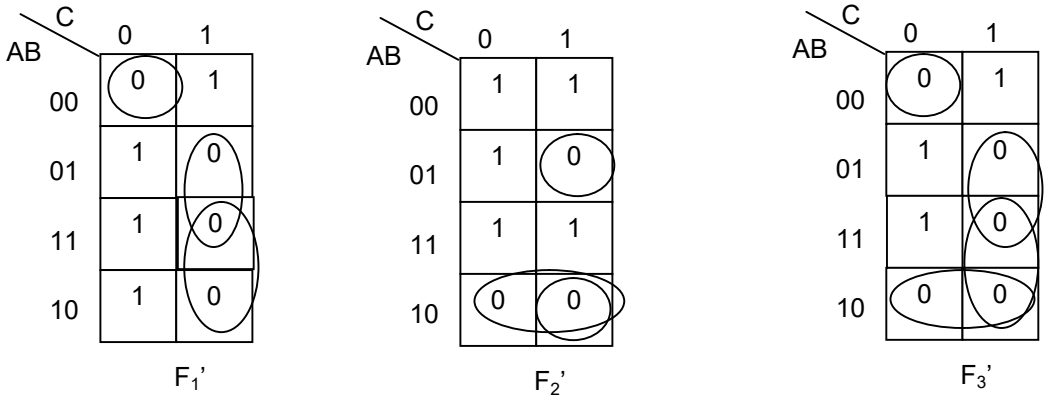
Solution:

9S. Obtain the simplest collective set (lowest number of literals) of SOP expressions for the 0s of the following functions:

$F_1(A,B,C) = \Sigma(1,2,4,6)$
$F_2(A,B,C) = \Sigma(0,1,2,6,7)$
$F_3(A,B,C) = \Sigma(1,2,6)$

Note: In addition to minimizing each function, also look for shared terms between the functions.

Solution:



$F_1' = A'.B'.C' + B.C + A.C$ (7 literals)
$F_2' = A.B' + A'.B.C$ (5 literals)
$F_3' = A.B' + + B.C + A.C + A'.B'.C'$ (9 literals)
Total literal count without term-sharing = 21
Total literal count with term-sharing = 12

9U. Obtain the simplest collective set (lowest number of literals) of POS expressions for the following functions:

$F_1(A,B,C,D) = \Sigma(1,2,3,4,6,7)$
$F_2(A,B,C,D) = \Sigma(0,1,2,3,4,6,7,14,15)$
$F_3(A,B,C,D) = \Sigma(1,2,6, 14, 15)$

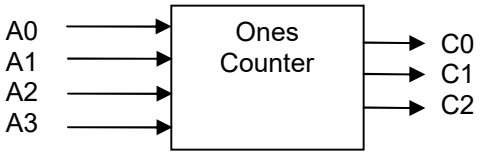
Note: In addition to minimizing each function, also look for shared terms between the functions.

Solution:
Not Provided.

10S. Show fuse map of a PLA that outputs number of 1's in a 4-bit input.

Solution:

Step 1: System Diagram



Step 2: the truth table

Input				Output		
A_1	A_0	B_1	B_0	C_2	C_1	C_0
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	0
0	1	1	1	0	1	1
1	0	0	0	0	0	1
1	0	0	1	0	1	0
1	0	1	0	0	1	0
1	0	1	1	0	1	1
1	1	0	0	0	1	0
1	1	0	1	0	1	1
1	1	1	0	0	1	1
1	1	1	1	1	0	0

Step 3: Minimize each of the output bits ($C_0 - C_2$)

A_1A_0 \ B_1B_0	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	0	1	0
10	0	0	0	0

$C_2 = A_1A_0B_1B_0$

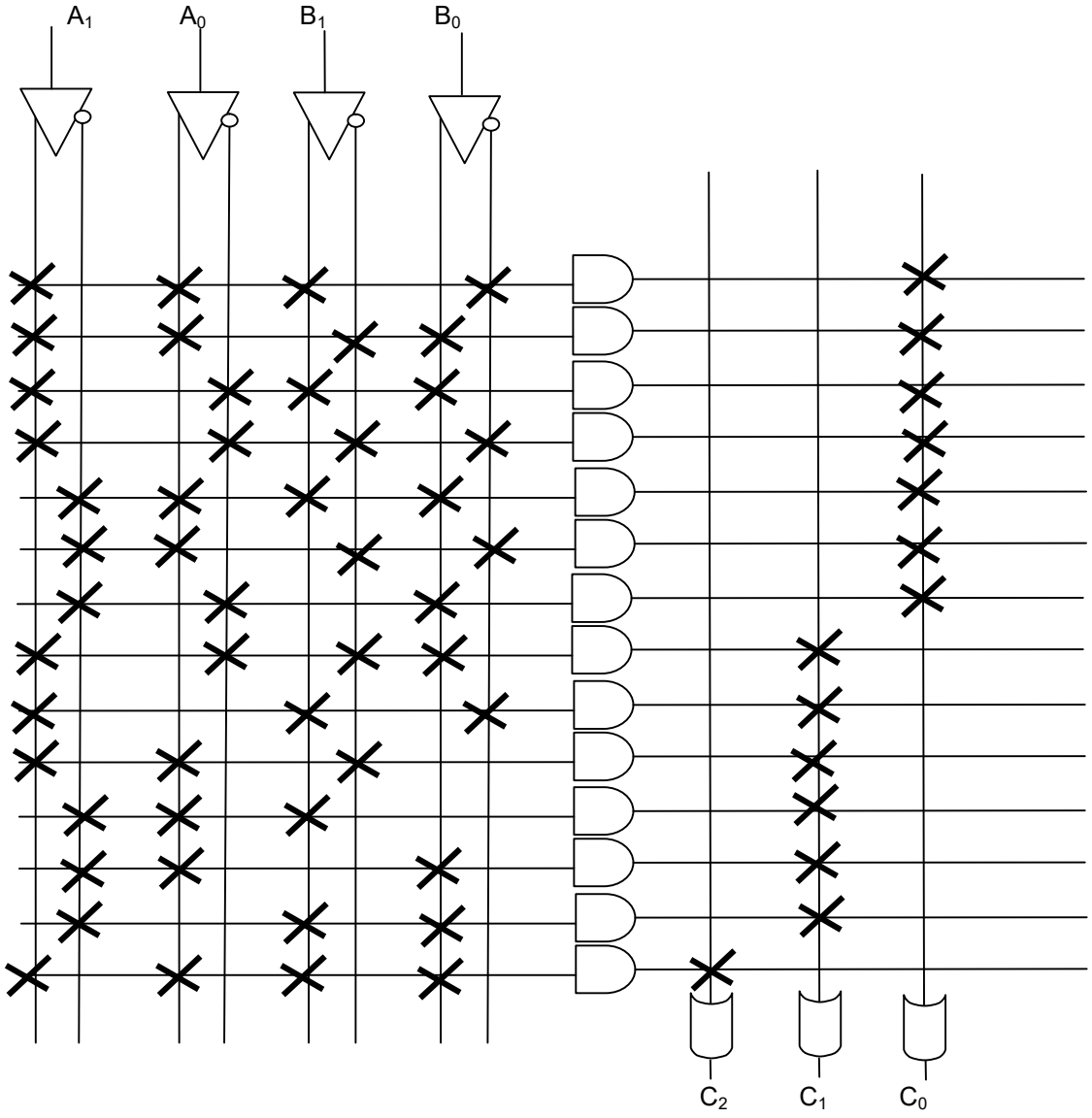
A_1A_0 \ B_1B_0	00	01	11	10
00	0	0	1	0
01	0	1	1	1
11	1	1	0	1
10	0	1	1	1

$C_1 = A_1'B_1B_0 + A_1'A_0B_0 + A_1'A_0B_1 + A_1A_0B_1 + A_1B_1B_0' + A_1A_0'B_0$

A_1A_0 \ B_1B_0	00	01	11	10
00	0	1	0	1
01	1	0	1	0
11	0	1	0	1
10	1	0	1	0

$C_0 = \sum (1,2,4,7,8,11,13,14)$

Step 4: Draw PLA Fuse Map



Note: only show the min-terms used.

10U. Show the fuse map for an 8 input lock that unlocks (output is 1) only when input are set to "10100100", "X0100X01" or "X1100X11". Implement the lock control using PLA.
Note: "X" represent don't care.

Solution:

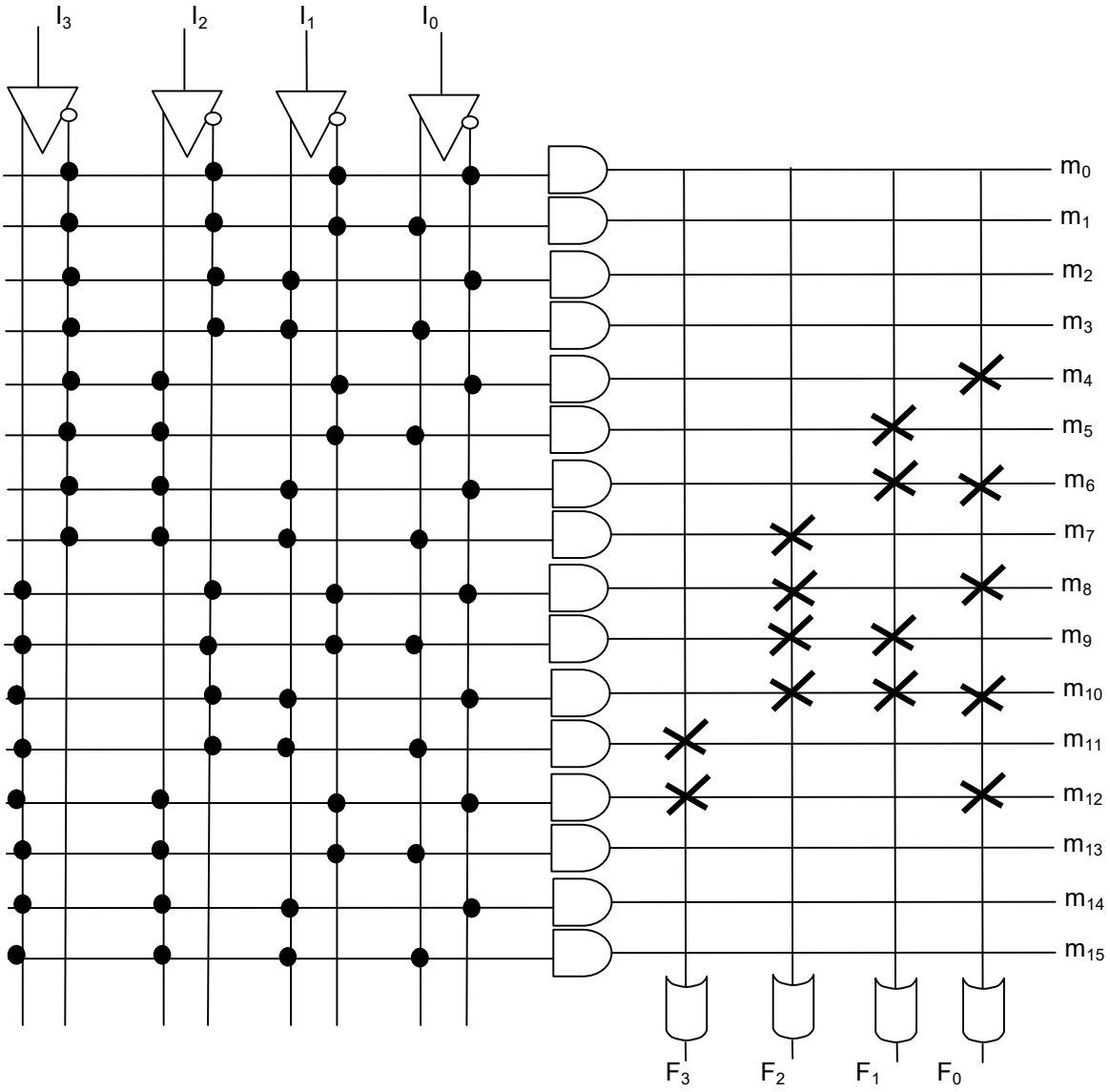
11S. Show the fuse map for an XS3-to-BCD code (see table for definition) converter with inputs I_3, I_2, I_1, I_0 and outputs F_3, F_2, F_1, F_0 using a simple PROM with 4 inputs and 8 outputs. (The min-terms not listed are 0 for all functions.)

XS3 Code				XS3	BCD Code			
I_3	I_2	I_1	I_0	minterm #	F_3	F_2	F_1	F_0
0	0	1	1	3	0	0	0	0
0	1	0	0	4	0	0	0	1
0	1	0	1	5	0	0	1	0

0 1 1 0	6	0 0 1 1
0 1 1 1	7	0 1 0 0
1 0 0 0	8	0 1 0 1
1 0 0 1	9	0 1 1 0
1 0 1 0	10	0 1 1 1
1 0 1 1	11	1 0 0 0
1 1 0 0	!2	1 0 0 1

Solution:

Note that for PROM, output need to be in Canonical form so there is not need to minimize.
For each output term, pick the XS3 minterms that make it one (Intact Fuse, X)



11U . Show the fuse map for an 8 input lock that unlocks (output is 1) only when input are set to “10100100”, “X0100X01” or “X1100X11”. Implement the lock control using PROM.
Note: “X” represent don’t care. Do not draw unused “and” terms.

Solution: