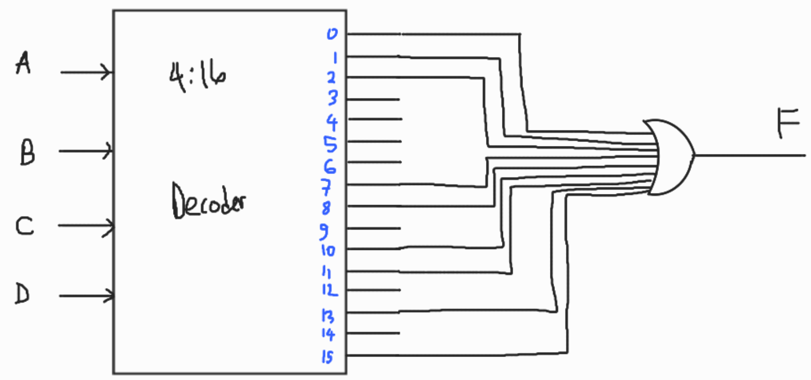
Q: Are all of these enough to get full marks in the exam?

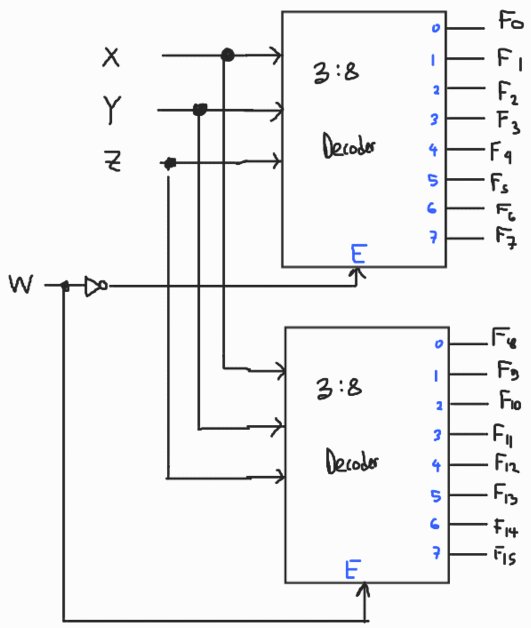
A: NO. This is a practice sheet. Meaning, you can practice all you want using the questions from this sheet. However, doing well in exams depends upon your ability to understand a question, formulate an answer, and express it correctly. You see, these are humane skills which cannot be guaranteed from completing a practice sheet only. But yeah, Best of luck anyways.

Decoders Related Problems:

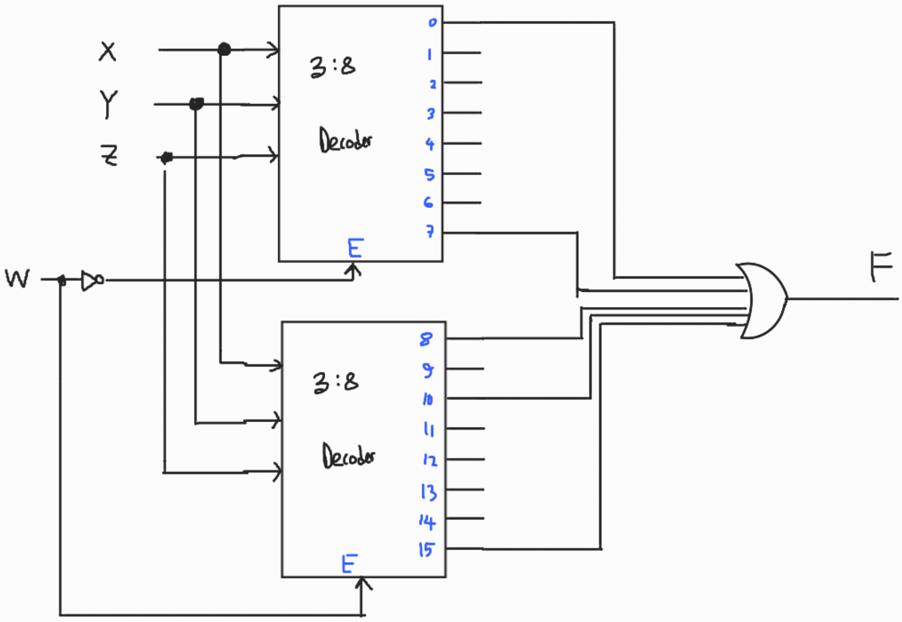
Implement F(A,B,C,D) = ∑(0,1,2,7,8,10,11,13, 15) using 4:16 Decoders.



Implement a 4:16 Decoder using 3:8 Decoders.



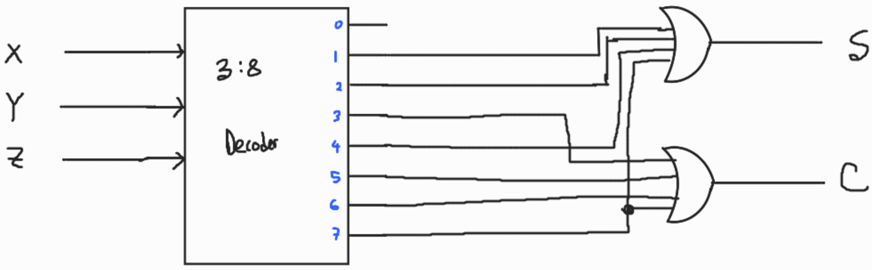
Implement F(w,x,y,z) = ∑(0,7,8,10, 15) using 3:8 Decoders.



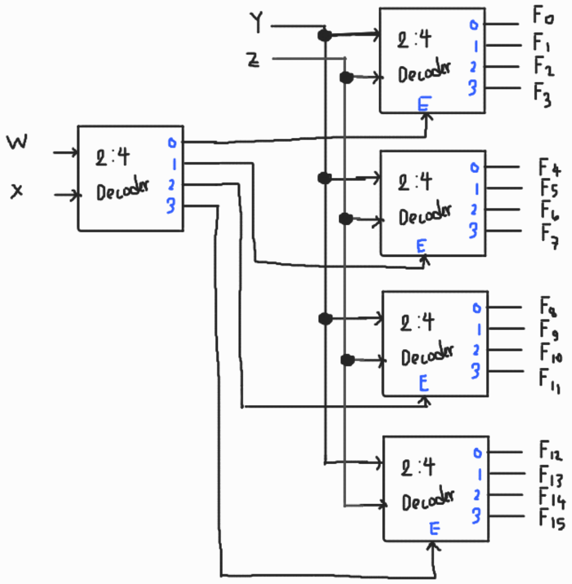
Implement Full Adder using 3:8 Decoders

x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

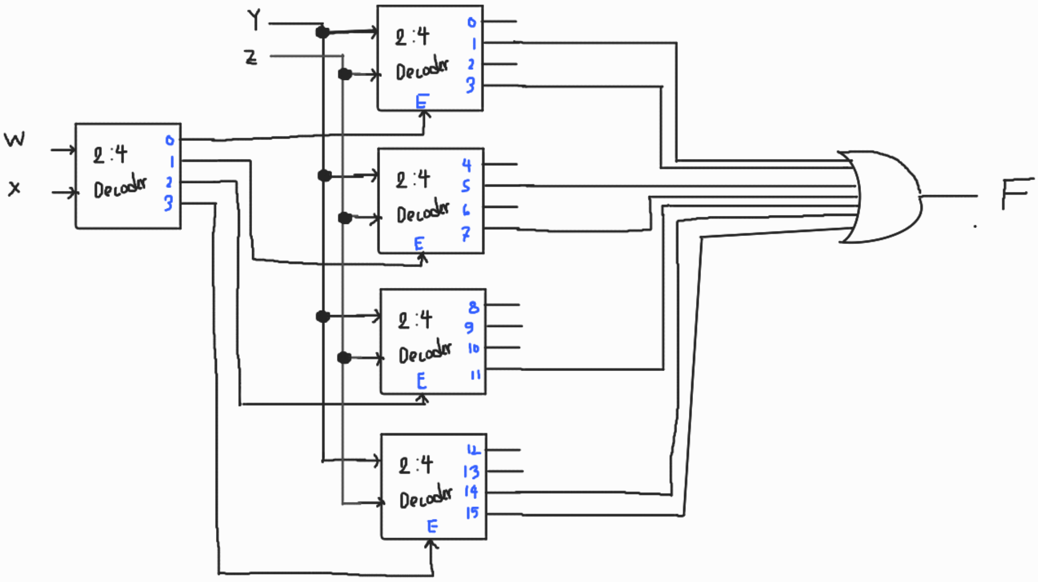
From the the truth table of Full Adder, we get,
S = ∑(1,2,4,7)
C = ∑(3,5,6,7)



Implement a 4:16 Decoder using 2:4 Decoders.

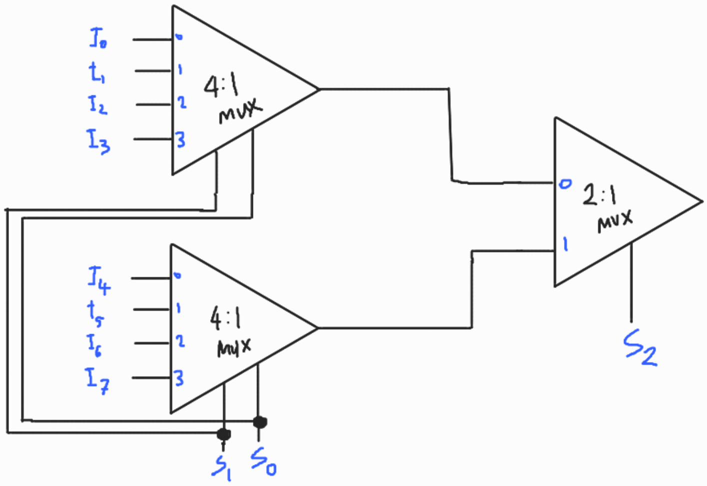


Implement F(a,b,c,d) = ∑(1,3,5,7,11,14,15) using only 2:4 Decoders.



MUX Related Problems:

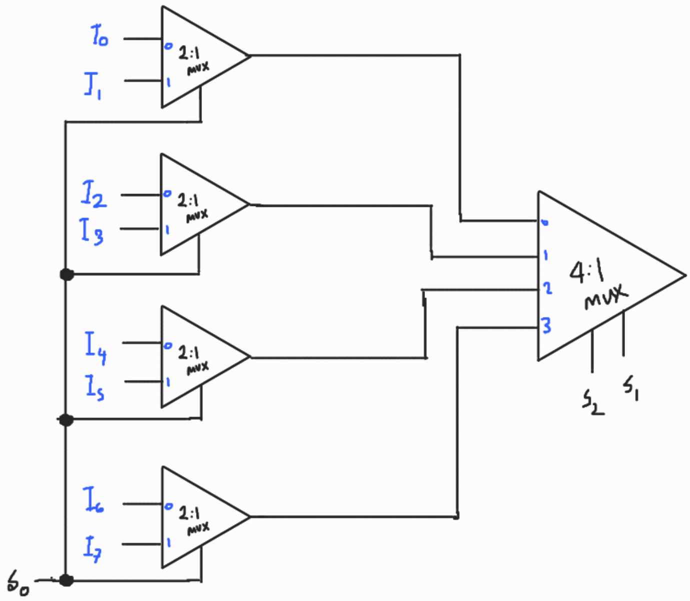
Implement 8x1 MUX using 4x1 MUX and 2x1 MUX



(Method 1)

Between selectors, assign MSB selectors in the rightmost MUX. S_0, S_2 is in the rightmost & S_1 S_0 in the left MUX.

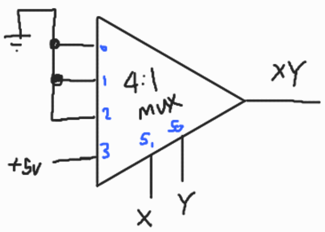
Method 2



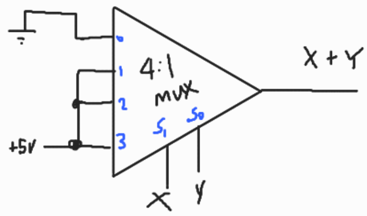
Use 4:1 MUX to design AND, NOT and OR Gate

AND:

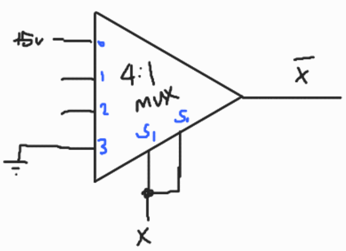
AND

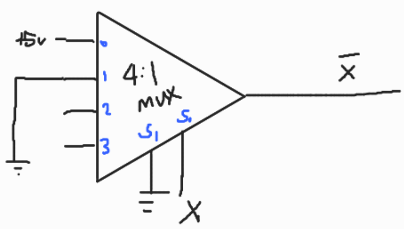


OR



NOT

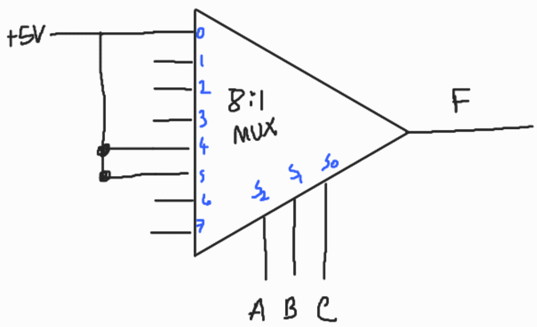




Build F = $\sum(0,4,5)$ using a 8:1 MUX

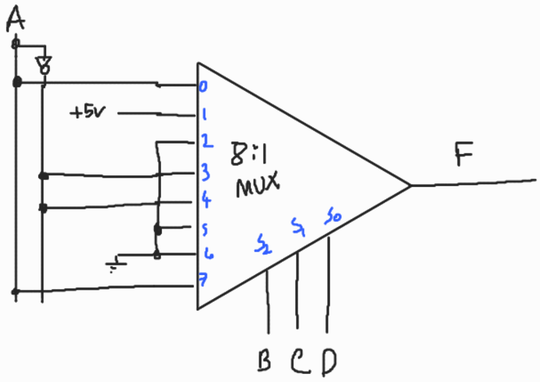
To represent 5, we need 3 bit. So, given function has 3 variables. Let, A,B, C are the variables.

8:1 MUX:



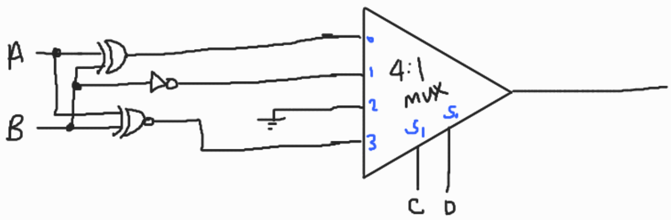
LARGER FUNCTION WITH SMALLER MUX:

Build F(A,B,C,D) = $\sum(1,3,4,8,9,15)$ using a 8:1 MUX



	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{A}$	0	1	2	3	4	5	6	7
A	8	9	10	11	12	13	14	15
	A	1	0	$\bar{A}$	$\bar{A}$	0	0	A

Build F(A,B,C,D) = $\sum(1,3,4,8,9,15)$ using a 4:1 MUX

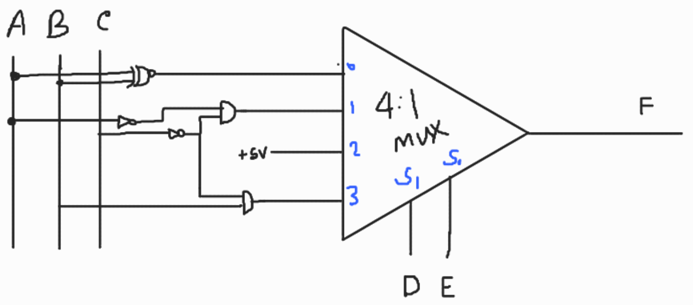


	I_0	I_1	I_2	I_3
A'B'	0	1	2	3
A'B	4	5	6	7
AB'	8	9	10	11
AB	12	13	14	15
	$= A'B + AB' = A \oplus B$	$= A'B' + AB' = B'(A+A') = B'$	0	$= A'B' + AB = A \odot B$

Implement the following boolean function using one 4:1 MUX only.
$F(A,B,C,D,E) = \sum(0,1,2,4,6,9,10,11,14,18,22,24,26,27,28,30)$. Use external gates if required.

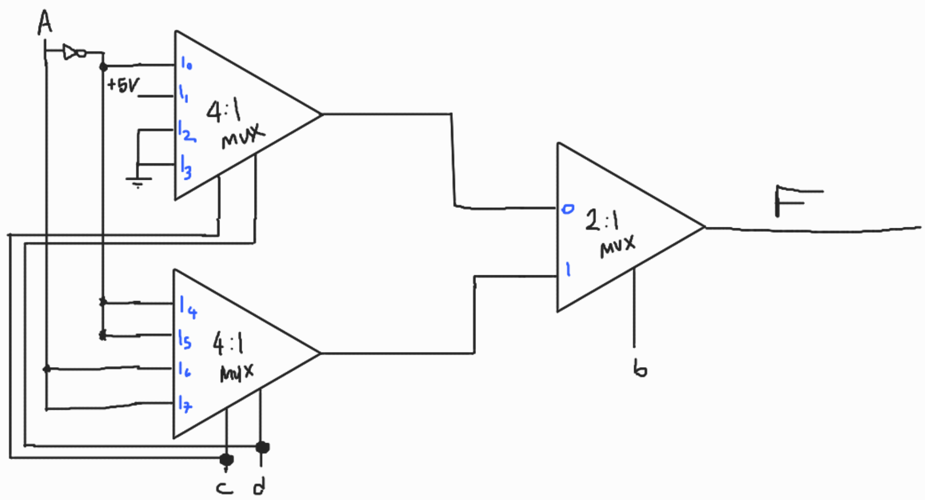
	I0	I1	I2	I3
A'B'C'	0	1	2	3
A'B'C	4	5	6	7
A'BC'	8	9	10	11
A'BC	12	13	14	15
AB'C'	16	17	18	19
AB'C	20	21	22	23
ABC'	24	25	26	27
ABC	28	29	30	31

$I0 = A'B'C' + A'B'C + ABC' + ABC = A'B'(C'+C) + AB(C+C') = A'B' + AB = A \odot B$
$I1 = A'B'C' + A'BC' = A'C'(B+B') = A'C'$
$I2 = 1$
$I3 = A'BC' + ABC' = BC'(A'+A) = BC'$



Implement the following function using 4:1 and 2:1 mux (both) only.
$F(A,B,C,D) = \sum(0,1,4,5,9,14,15)$

	I0	I1	I2	I3	I4	I5	I6	I7
A'	0	1	2	3	4	5	6	7
A	8	9	10	11	12	13	14	15
	A'	1	0	0	A'	A'	A	A



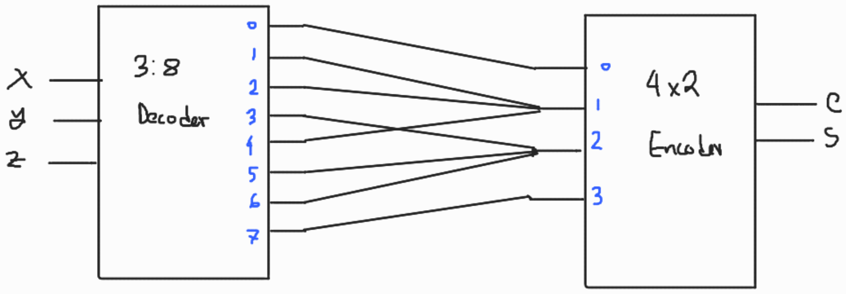
Decoder Encoder Related Problems:

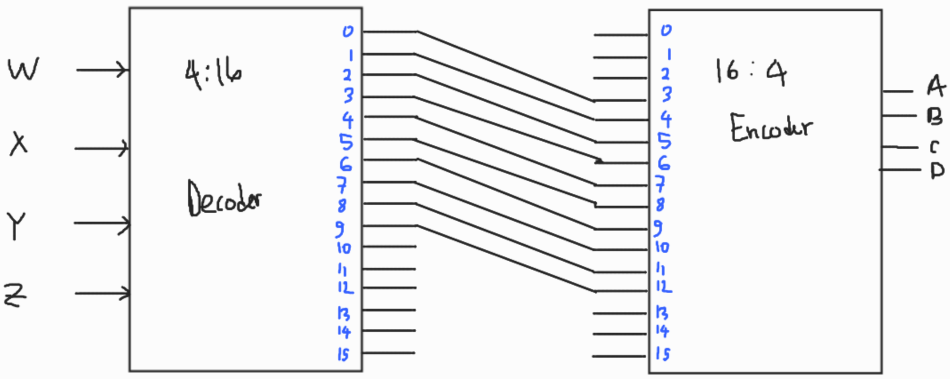
	x	y	z	C	S
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	1

C	S	
0	0	→ 0
0	1	→ 1, 2, 4
1	0	→ 3, 5, 6
1	1	→ 7

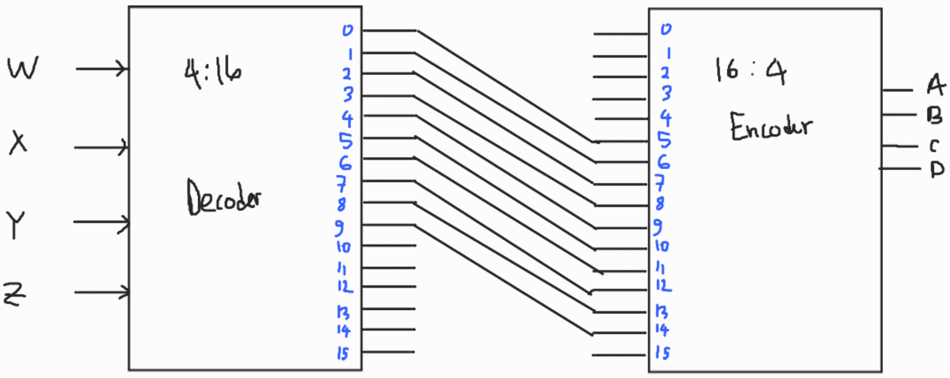
Design a Full Adder using 3x8 Decoder using 4x2 Encoder

From the truth table, we build combination for C and S





BCD to Excess 3 code converter using 4x16 Decoder and 16x4 encoder



BCD to Excess 5 code converter using 4x16 Decoder and 16x4 encoder