EE 2720 Fall 2011

HW #6 solution

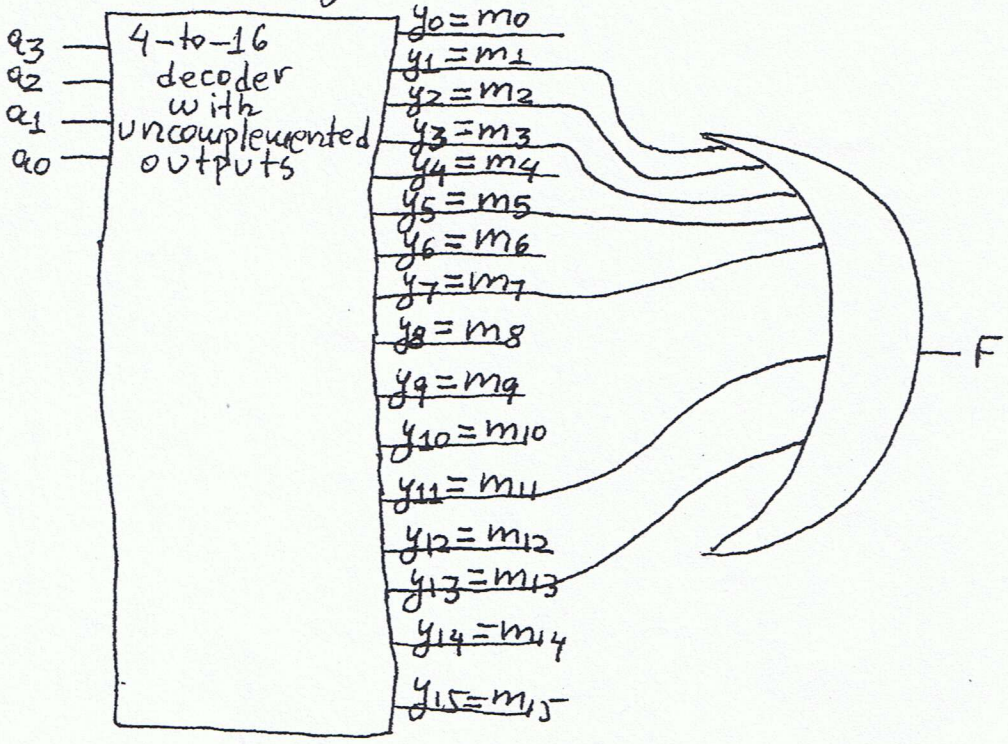
EE 2720 (6)

HW# 6 Solutions

Problem 1: Here, I will denote minterm i by m_i and maxterm i by M_i; (just notations).

(a). We have: $F = \Sigma_{q_3, q_2, q_1, q_0}(1, 2, 3, 5, 7, 11, 13)$.

From this we get:



(b). We have: $F = \Sigma_{q_3, q_2, q_1, q_0}(1, 2, 3, 5, 7, 11, 13) =$

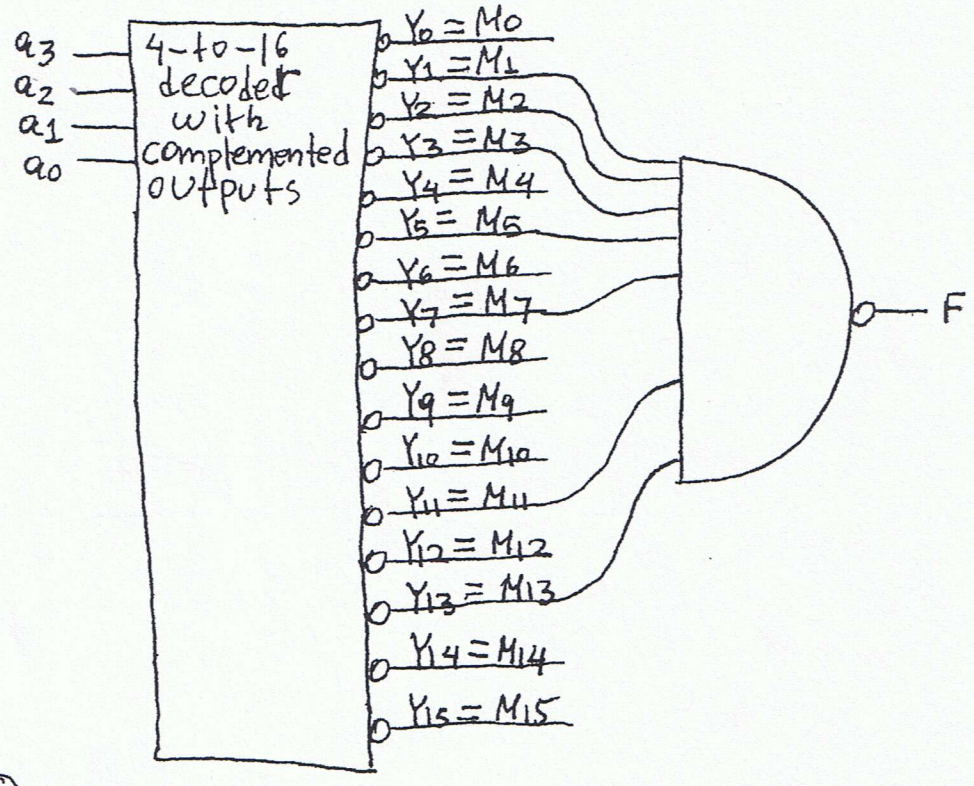
$= m_1 + m_2 + m_3 + m_5 + m_7 + m_{11} + m_{13} =$

$= (m_1' \cdot m_2' \cdot m_3' \cdot m_5' \cdot m_7' \cdot m_{11}' \cdot m_{13}')' =$

$(M_1 \cdot M_2 \cdot M_3 \cdot M_5 \cdot M_7 \cdot M_{11} \cdot M_{13})'$.

From the above we get:

Problem 1 cont:

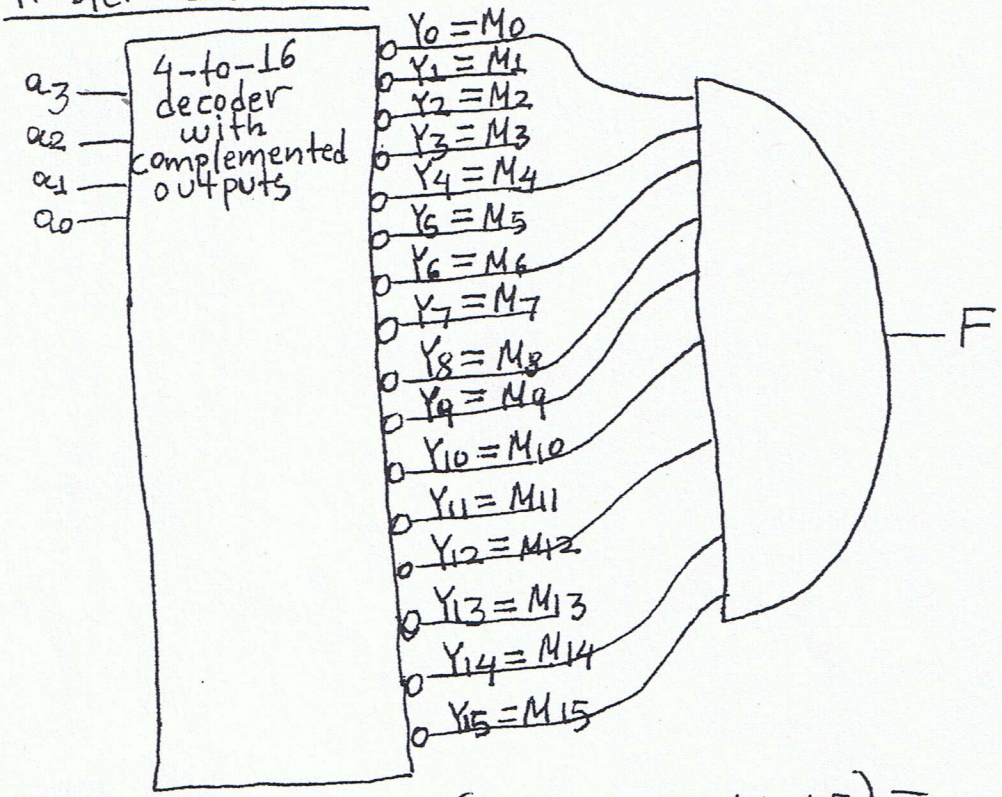


(c) $F = \sum_{a_3, a_2, a_1, a_0}(1, 2, 3, 5, 7, 11, 13) =$

$= \prod_{a_3, a_2, a_1, a_0}(0, 4, 6, 8, 9, 10, 12, 14, 15).$

From the above we get:

Problem 1 cont:



(d) $F = \Sigma_{a_3,a_2,a_1,a_0}(1,2,3,5,7,11,13) =$

$= \Pi_{a_3,a_2,a_1,a_0}(0,4,6,8,9,10,12,14,15) =$

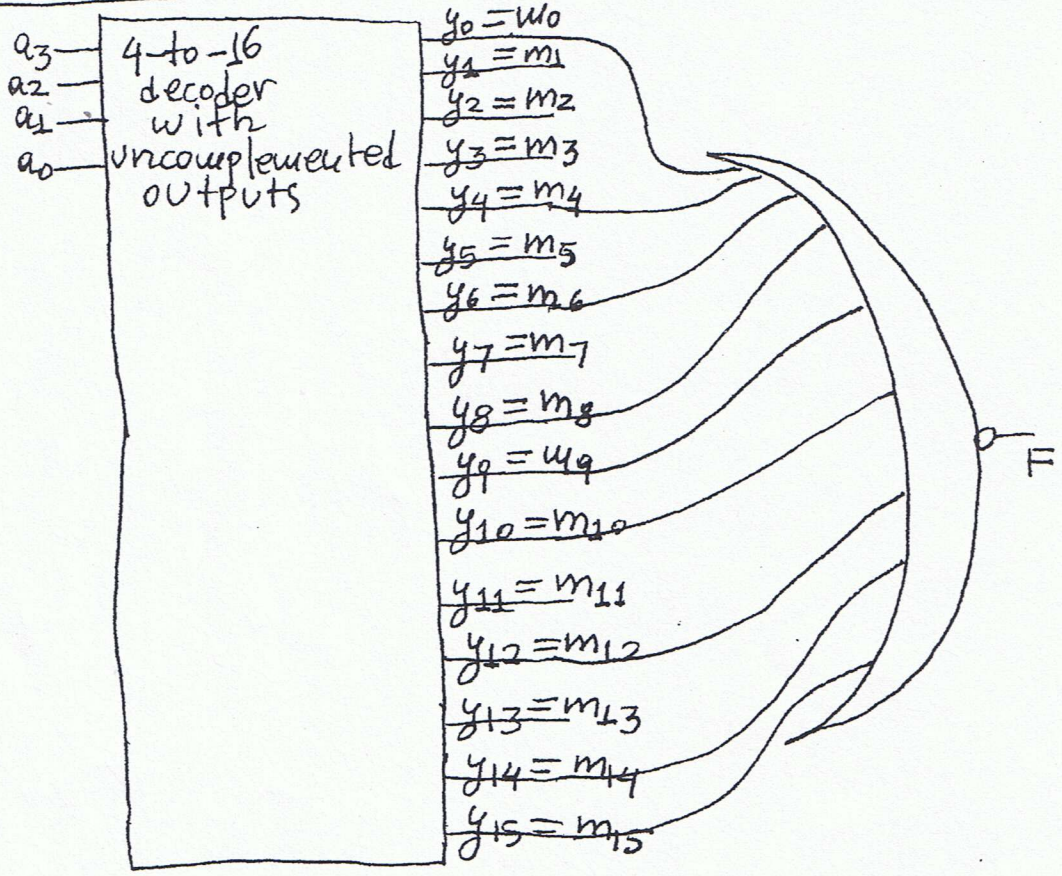
$= M_0 \cdot M_4 \cdot M_6 \cdot M_8 \cdot M_9 \cdot M_{10} \cdot M_{12} \cdot M_{14} \cdot M_{15} =$

$= (M_0' + M_4' + M_6' + M_8' + M_9' + M_{10}' + M_{12}' + M_{14}' + M_{15}')'$

$= (m_0 + m_4 + m_6 + m_8 + m_9 + m_{10} + m_{12} + m_{14} + m_{15})'$

From the above we get:

Problem 1 cont:



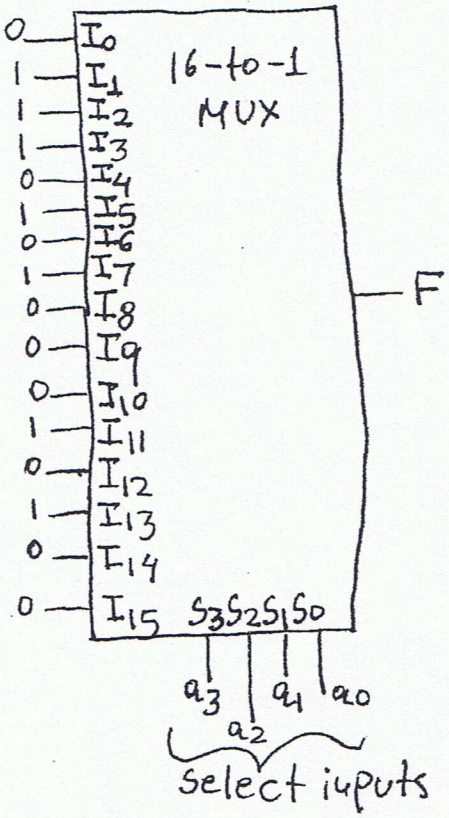
For parts (e) and (f) we need a truth table. The truth table is provided on the next page.

HW # [redacted] solutions cont.

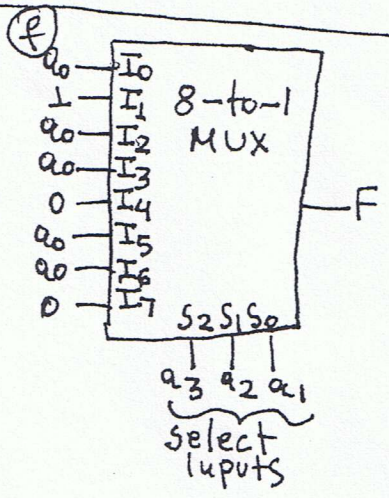
Problem 1 cont:

a_3	a_2	a_1	a_0	F	
0	0	0	0	0	a_0
0	0	0	1	1	
0	0	1	0	1	1
0	0	1	1	1	
0	1	0	0	0	a_0
0	1	0	1	1	
0	1	1	0	0	a_0
0	1	1	1	1	
1	0	0	0	0	0
1	0	0	1	0	
1	0	1	0	0	a_0
1	0	1	1	1	
1	1	0	0	0	a_0
1	1	0	1	1	
1	1	1	0	0	0
1	1	1	1	0	

(e)



(f)



(g). Design of part (f) is better because it relies on a smaller multiplexer and thus it is cheaper and faster.

Problem 2: Since a decoder with uncomplemented outputs generates the minterms, we have to express F as a canonical sum. Since F is given as a product-of-sums expression, it is probably easier to obtain the canonical product from which we can easily get the canonical sum. We have: $F=(a+b+c)\cdot(b+c+d)\cdot(a+c+d)=$

$=(a+b+c+d\cdot d')\cdot(b+c+d+a\cdot a')\cdot(a+c+d+b\cdot b')$

$=(a+b+c+d)\cdot(a+b+c+d')\cdot$

$(b+c+d+a)\cdot(b+c+d+a')\cdot$

$(a+c+d+b)\cdot(a+c+d+b')=$

$=(a+b+c+d)\cdot(a+b+c+d')\cdot(a'+b+c+d)\cdot(a+b'+c+d)$

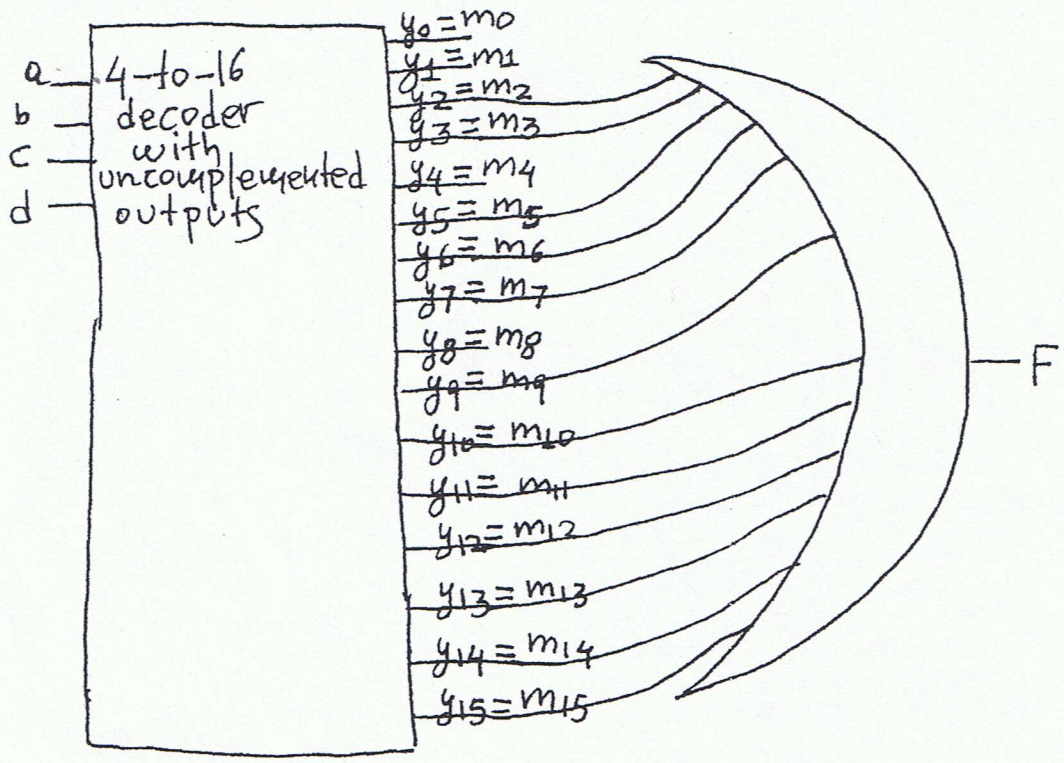
$\quad 0\ 0\ 0\ 0 \qquad 0\ 0\ 0\ 1 \qquad 1\ 0\ 0\ 0 \qquad 0\ 1\ 0\ 0$

$=\Pi_{a,b,c,d}(0,1,4,8)=$

$=\Sigma_{a,b,c,d}(2,3,5,6,7,9,10,11,12,13,14,15)$

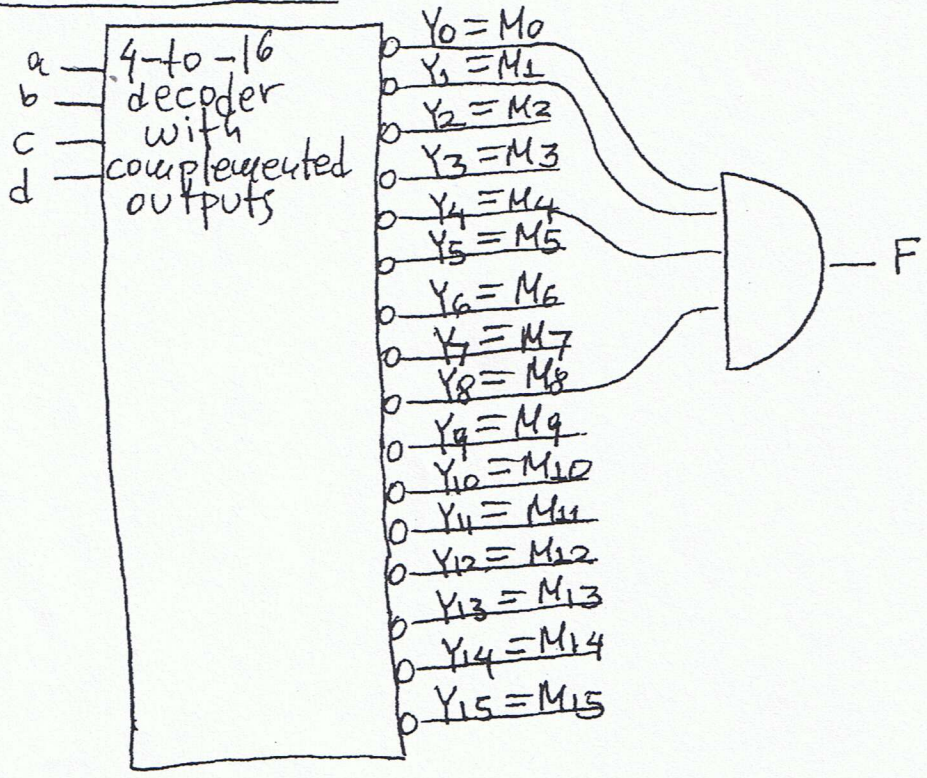
From the above canonical sum expression we get:

Problem 2 cont:



Problem 3: Since a decoder with complemented outputs generates the maxterms, we have to use the canonical product expression for F that we obtained in problem 2. This is $F = \Pi_{a,b,c,d}(0,1,4,8)$. From this canonical product we get:

Problem 3 cont:



Problem 4: Since a decoder with uncomplemented outputs generates the minterms, we have to express F as a canonical sum. We have:

$$F = a\cdot b\cdot c + b\cdot c\cdot d + a\cdot d = a\cdot b\cdot c\cdot(d+d') +$$
$$+ b\cdot c\cdot d\cdot(a+a') + a\cdot d\cdot(b+b')\cdot(c+c') =$$
$$= a\cdot b\cdot c\cdot d + a\cdot b\cdot c\cdot d' + a\cdot b\cdot c\cdot d + a'\cdot b\cdot c\cdot d +$$
$$+ a\cdot d\cdot(b\cdot c + b\cdot c' + b'\cdot c + b'\cdot c') =$$

Problem 4 cont.

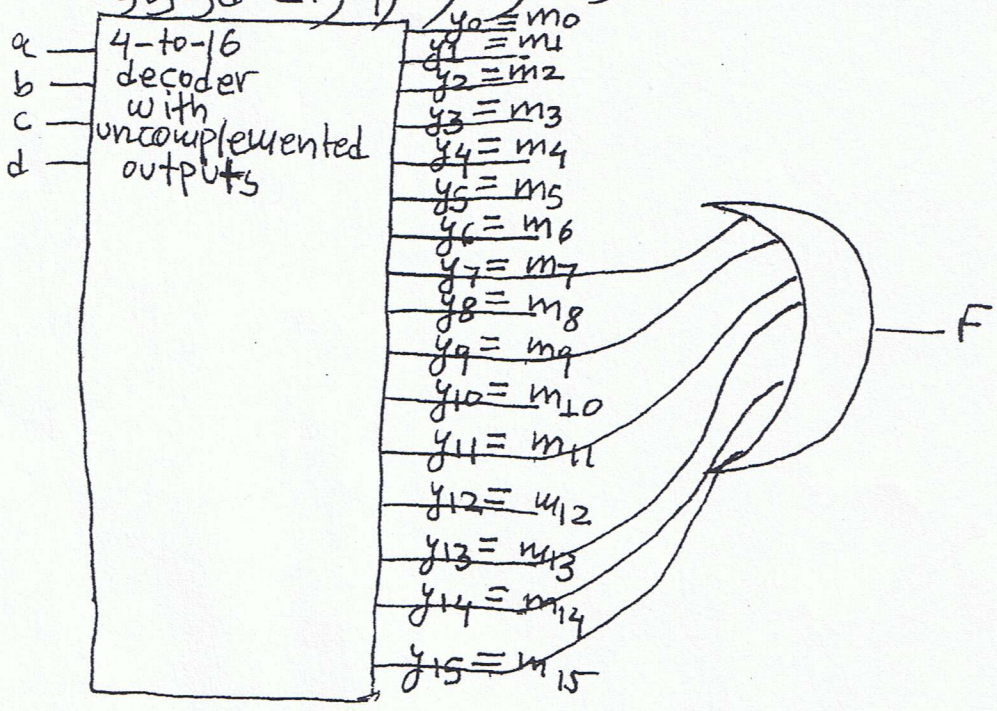
$= a \cdot b \cdot c \cdot d + a \cdot b \cdot c \cdot d' + a \cdot b \cdot c \cdot d + a' \cdot b \cdot c \cdot d$

$+ a \cdot b \cdot c \cdot d + a \cdot b \cdot c' \cdot d + a \cdot b' \cdot c \cdot d + a \cdot b' \cdot c' \cdot d$

$= a \cdot b \cdot c \cdot d + a \cdot b \cdot c \cdot d' + a' \cdot b \cdot c \cdot d$
$\quad 1\ 1\ 1\ 1 \quad 1\ 1\ 1\ 0 \quad 0\ 1\ 1\ 1$

$+ a \cdot b \cdot c' \cdot d + a \cdot b' \cdot c \cdot d + a \cdot b' \cdot c' \cdot d$
$\quad 1\ 1\ 0\ 1 \quad 1\ 0\ 1\ 1 \quad 1\ 0\ 0\ 1$

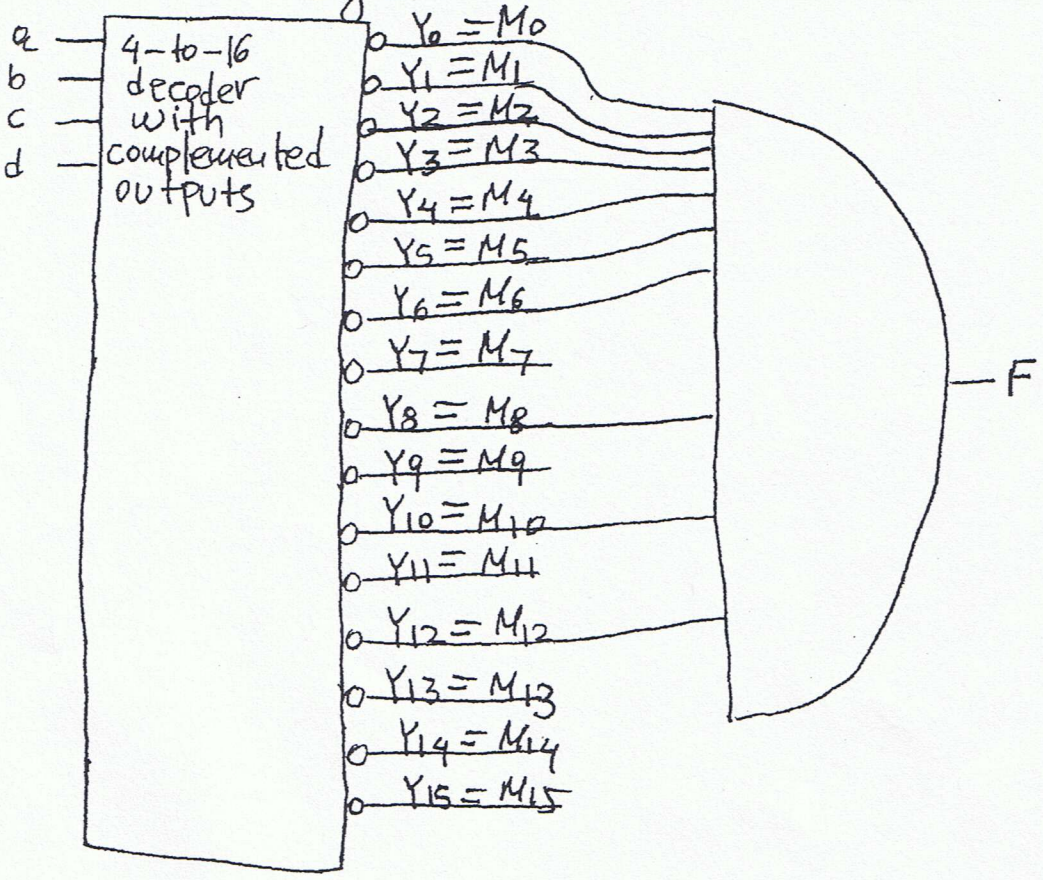
$= \Sigma_{a,b,c,d}(7, 9, 11, 13, 14, 15)$. From this we get:



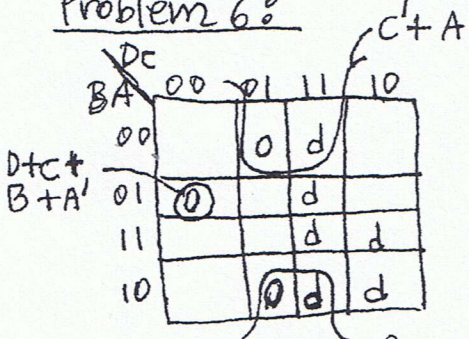
Problem 5: Here we need F expressed as canonical product. From problem 4 we have

$F = \Sigma_{a,b,c,d}(7, 9, 11, 13, 14, 15) =$

$= \Pi_{a,b,c,d}(0, 1, 2, 3, 4, 5, 6, 8, 10, 12)$. From this expression we get:

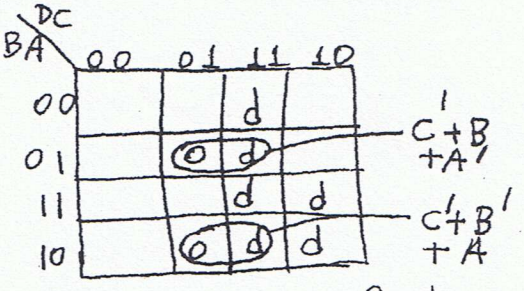


Problem 6:



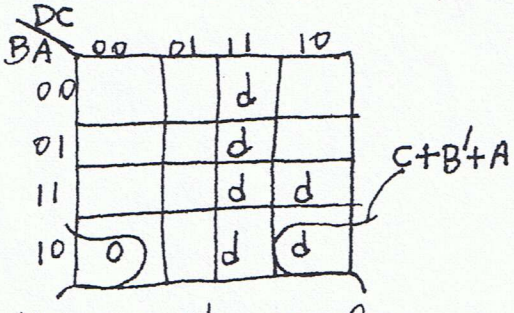
Karnaugh map for a

$a = (D+C+B+A') \cdot (C'+A)$



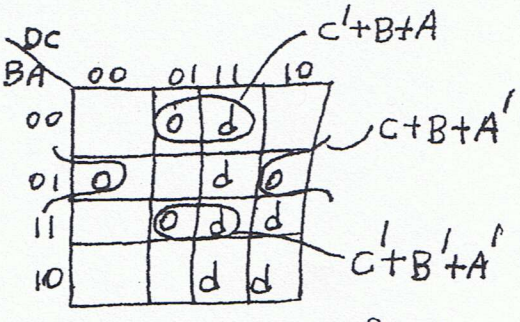
Karnaugh map for b

$b = (C'+B+A') \cdot (C'+B'+A)$



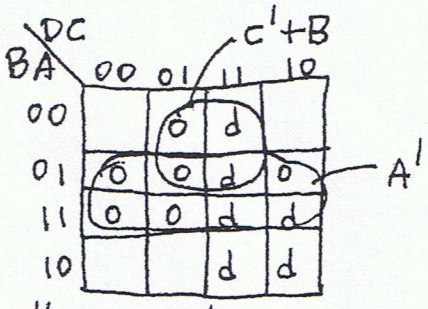
Karnaugh map for c

$c = C+B'+A$



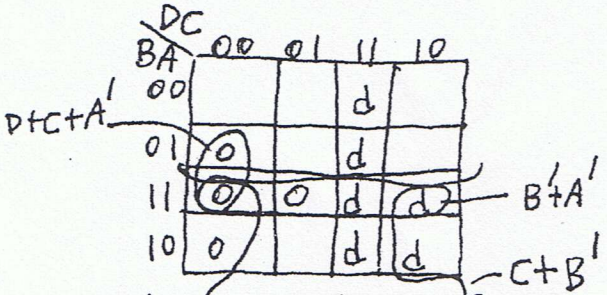
Karnaugh map for d

$d = (C'+B'+A') \cdot (C+B+A') \cdot (C'+B+A)$



Karnaugh map for e

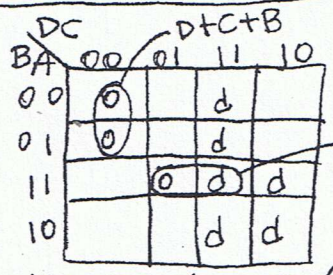
$e = A' \cdot (C'+B)$



Karnaugh map for f

$f = (D+C+A') \cdot (C+B') \cdot (B'+A')$

Problem 6 cont:

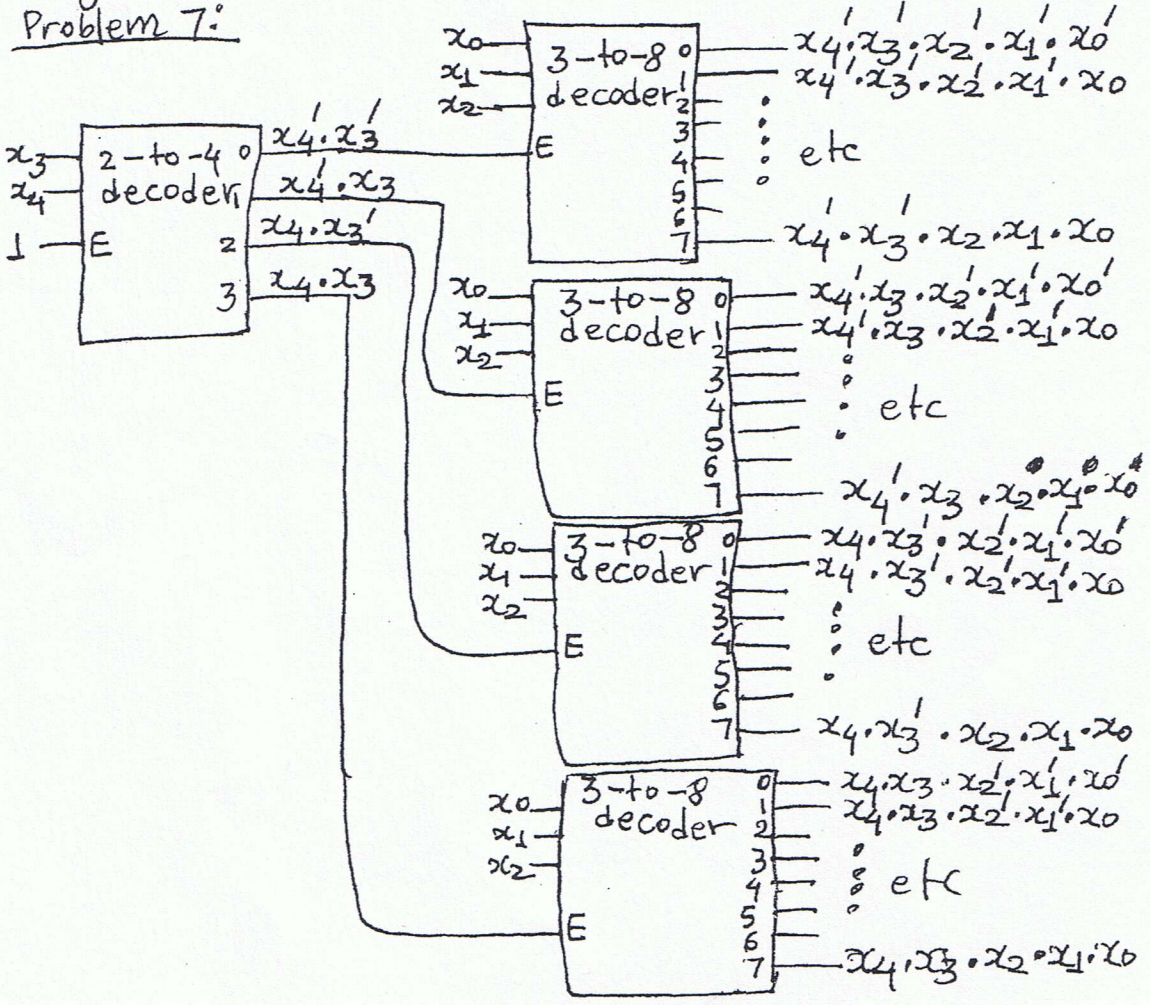


Karnaugh map for g

$g = (D+C+B)\cdot(C'+B'+A')$

Note: I got the karnaugh maps for a, b, c, d, e, f, g from the truth table of page 13 of hand-out #17.

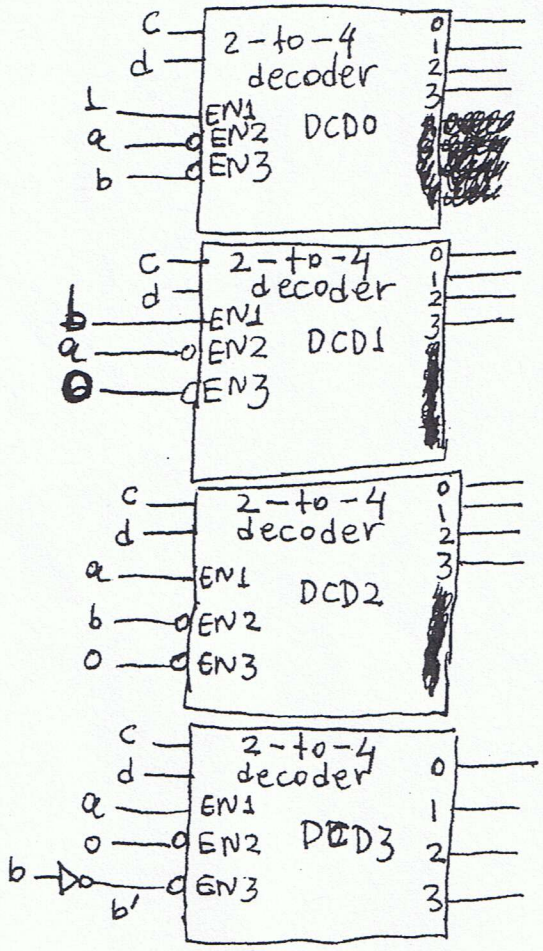
Problem 7:



EE 2720 (6)
HW #6 solutions cont.

Problem 7 cont: For the 5-to-32 decoder of problem 7, x_4 is the MSB of the vector to be decoded while x_0 is the LSB of the vector to be decoded. As seen, the 5-to-32 decoder generates at its outputs all the ~~eight~~ 32 minterms of the variables x_4, x_3, x_2, x_1, x_0.

Problem 8:

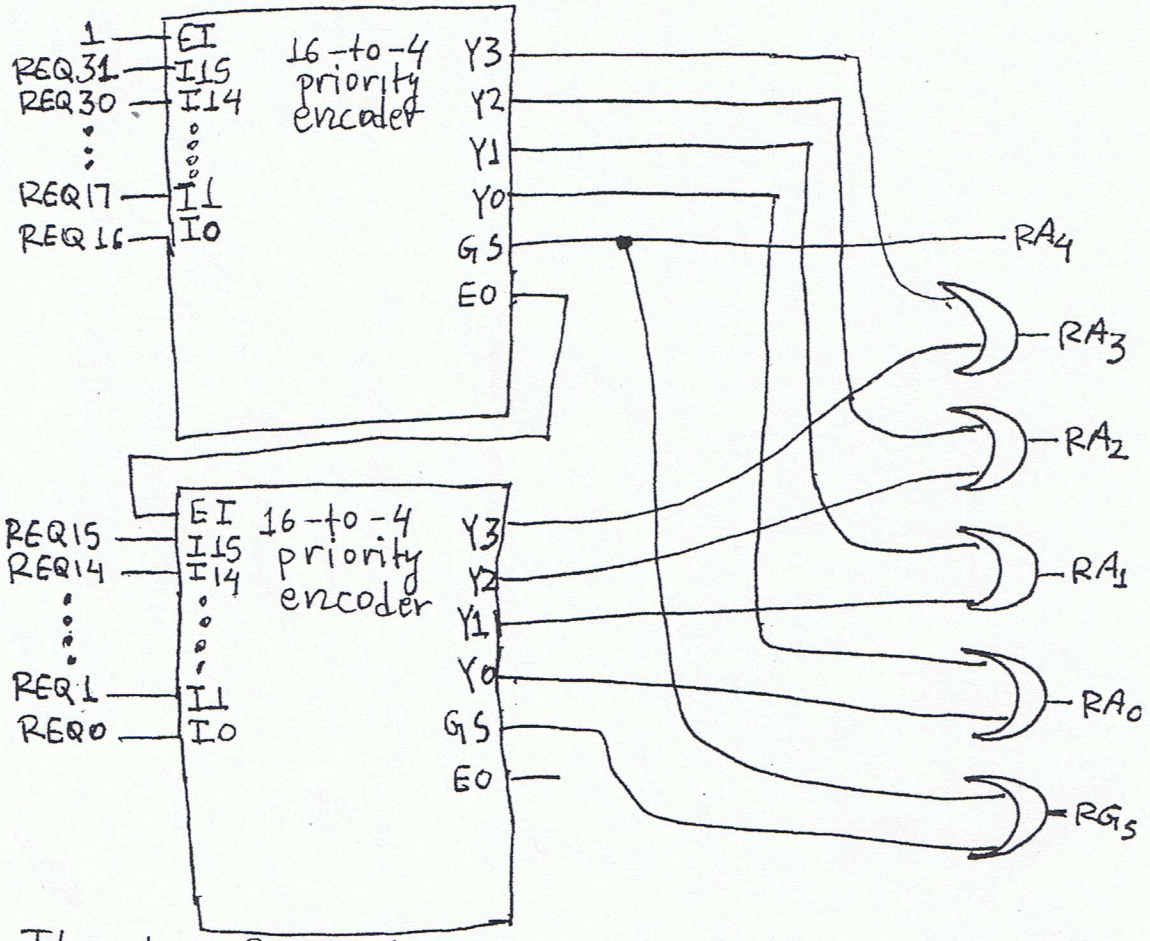


For this 4-to-16 decoder, a is the MSB of the decoded vector, while d is the LSB of the decoded vector.

- If ab=00, then the decoder named DCD0 is enabled and all others disabled.
- If ab=01, then the decoder named DCD1 is enabled and all others disabled.
- If ab=10, then the decoder named DCD2 is enabled and all others disabled.
- If ab=11, then the decoder named DCD3 is enabled and all others disabled.

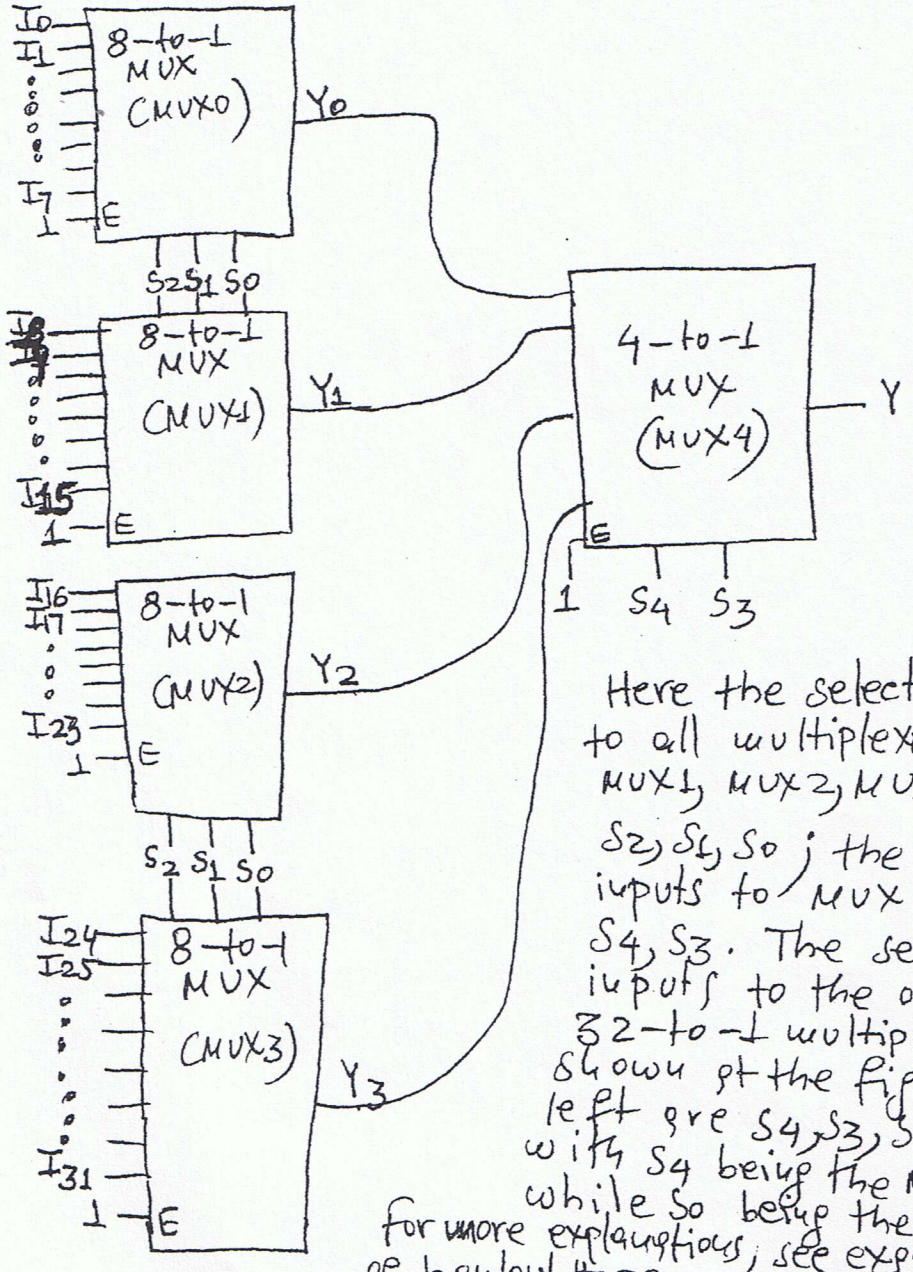
Note: For more explanations, see page 8 of handout #18.

Problem 9:



The above figure shows the 32-to-5 priority encoder. Its data inputs are REQ31, REQ30, ..., REQ1, REQ0. The input REQ31 is the highest priority input, while REQ0 is the lowest priority input. Its data outputs are RA_4, RA_3, RA_2, RA_1, RA_0 with RA_4 being the MSB, while RA_0 being the LSB. For further explanations, see handout# 19 (following fig. 2 of handout# 19).

Problem 10:

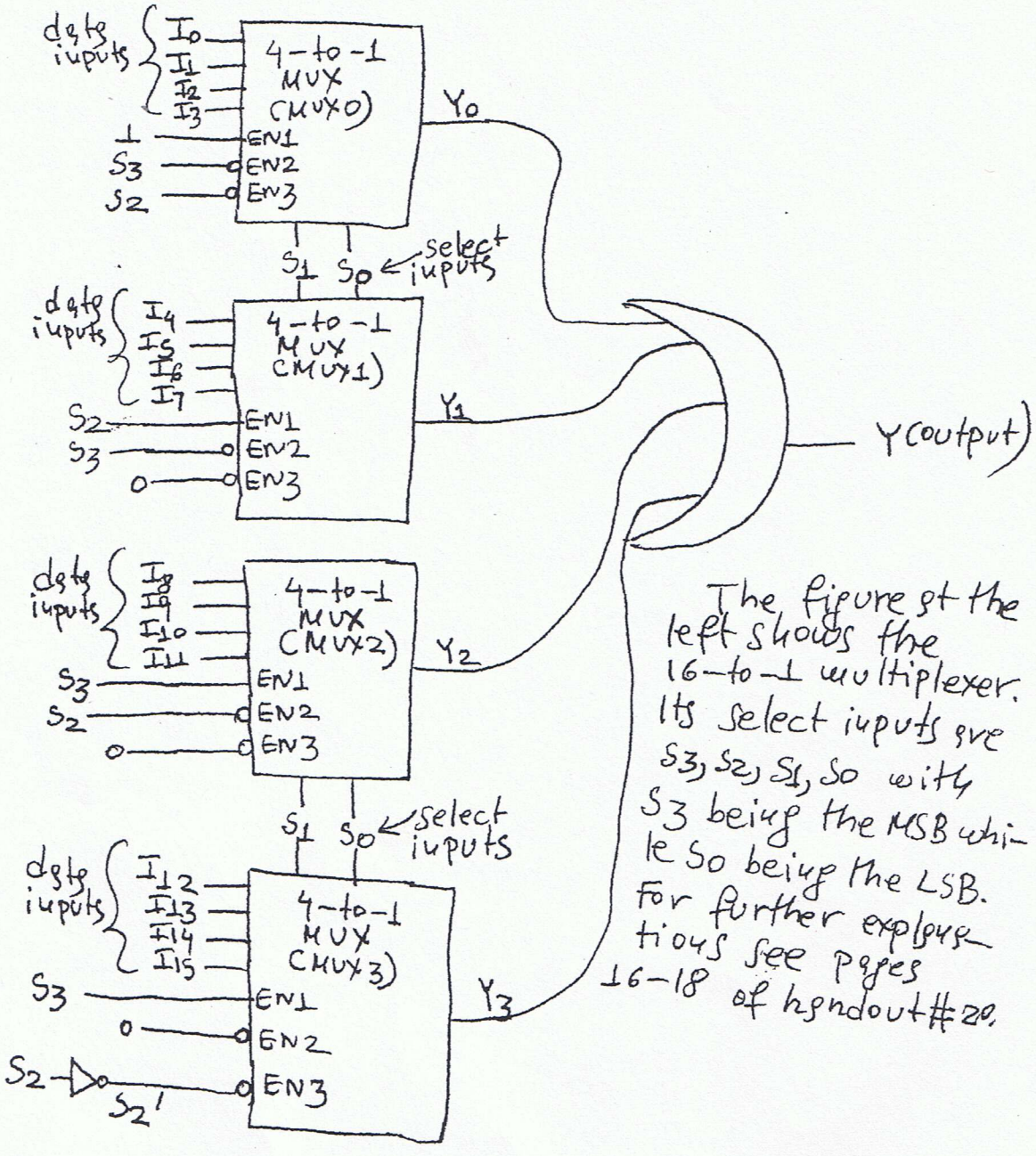


Here the select inputs to all multiplexers MUX0, MUX1, MUX2, MUX3 are S_2, S_1, S_0; the select inputs to MUX4 are S_4, S_3. The select inputs to the overall 32-to-1 multiplexer shown at the fig. at the left are S_4, S_3, S_2, S_1, S_0, with S_4 being the MSB, while S_0 being the LSB. For more explanations, see examples 1, 2 of handout #20.

EE 2720 (6)

HW# 6 solutions cont.

Problem 11:

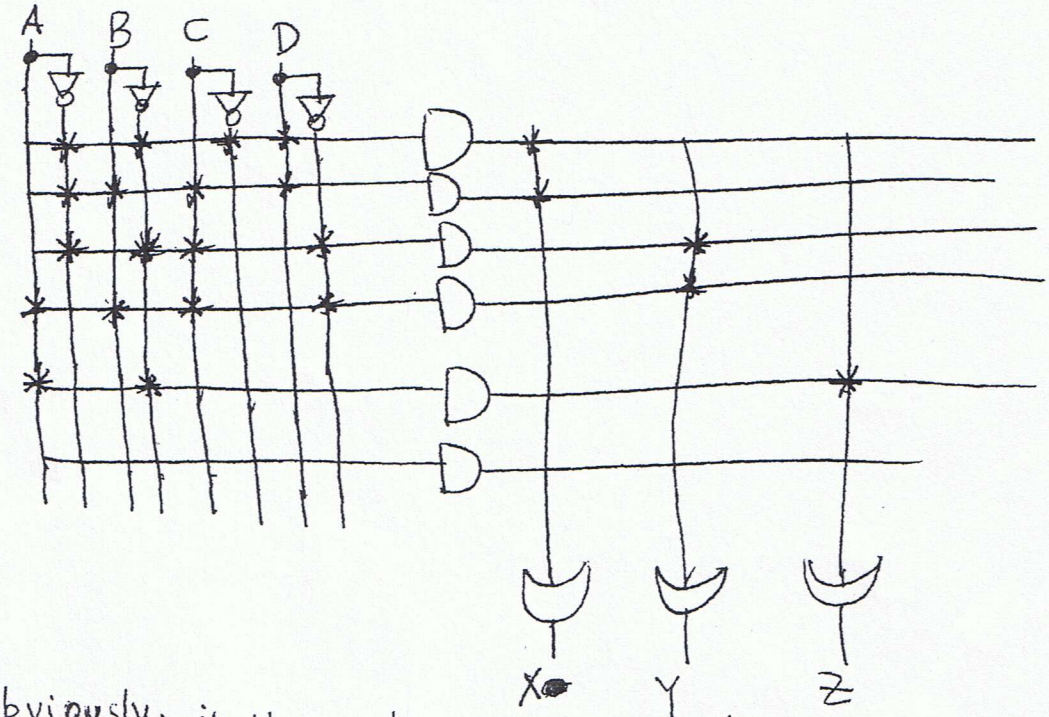


The figure at the left shows the 16-to-1 multiplexer. Its select inputs are S_3, S_2, S_1, S_0 with S_3 being the MSB while S_0 being the LSB. For further explorations see pages 16-18 of handout #20.

EE 2720
HW #1 solutions cont.

Problem 12: Important note: A PLA is a two-level AND-OR device that can realize sum-of-products expressions; (we said that on page 4 of handout #21). Thus, we have to transform the expressions for X, Y, Z into sum-of-products. Here X and Z are in sum-of-products form but Y is not. So, for Y we have:

$Y = \Pi_{A,B,C,D}(0,1,3,4,5,6,7,8,9,10,11,12,13,15) =$
$= \Sigma_{A,B,C,D}(2,14)$. We now have the following PLA implementation:



Obviously, in the above, $X = A' \cdot B' \cdot C' \cdot D + A' \cdot B \cdot C \cdot D$

$Y = A' \cdot B' \cdot C \cdot D' + A \cdot B \cdot C \cdot D'$

$Z = A \cdot B'$

Problem 13: We have to transform the expressions for X, Y, Z into sum-of-products forms. Here, X and Z are in sum-of-products form but Y isn't. for Y we have:

$Y = \Pi_{A,B,C,D}(3,3,4,6,7,8,10,11,12,13,14,15) =$

$= \Sigma_{A,B,C,D}(0,1,5,9)$. We now have:

$X = A' \cdot B' \cdot C' \cdot D' + A \cdot B \cdot C \cdot D$

$Y = A' \cdot B' \cdot C' \cdot D' + A' \cdot B' \cdot C' \cdot D + A' \cdot B \cdot C' \cdot D + A \cdot B' \cdot C' \cdot D$

$Z = A' \cdot B \cdot C' \cdot D + A \cdot B' \cdot C' \cdot D + A \cdot B \cdot C' \cdot D'$

From the above, we have the following PLA implementation: