Stony Brook University

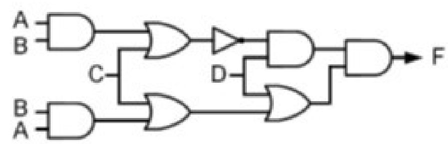
CSE 220: Systems Fundamentals I

Critical Path & Multiplexors Practice Problems

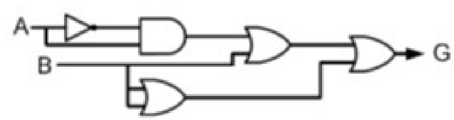
Important Notes:

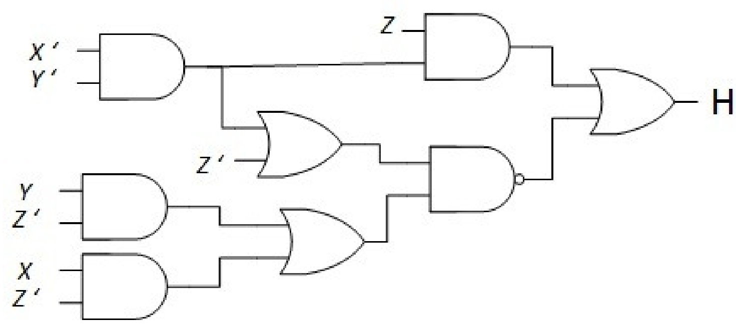
- These practice problems will provide you a sense of the kinds of questions you may encounter on the quizzes and examinations.
- The real quizzes and exams might include questions on material not covered by these practice problems.
- The questions on the real quizzes and exams might not have the same formats as these practice problems.

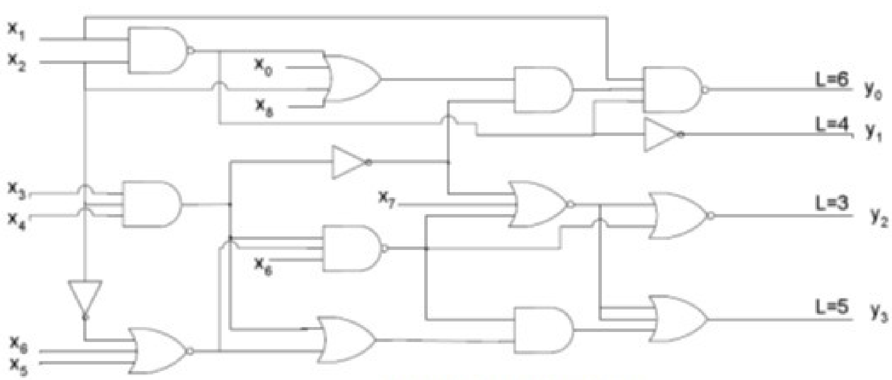
1. Calculate the critical path delay and the critical path for each output of the gate networks.



Gate	Timing
AND	3ns
OR	4ns
NOT	1ns
NAND	2ns
NOR	2ns
XOR	2ns
Multiplexor (any size)	3ns
Decoder	5ns
Encoder	4ns

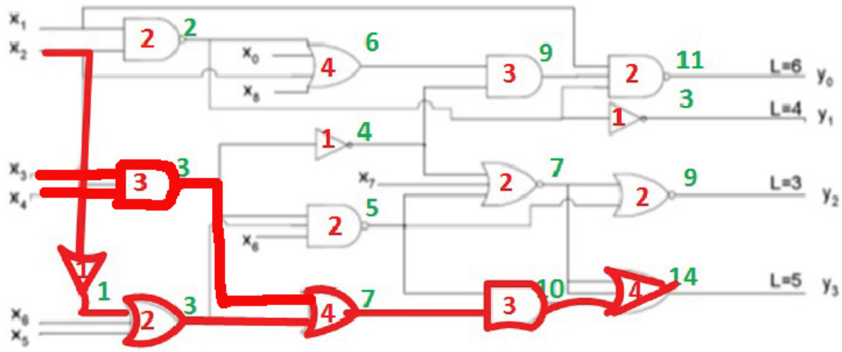


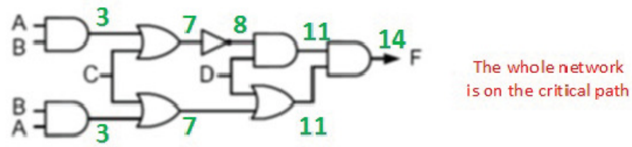


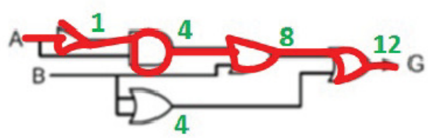


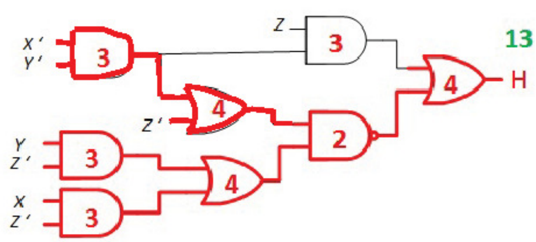
(ignore L numbers in figure)

Solution:



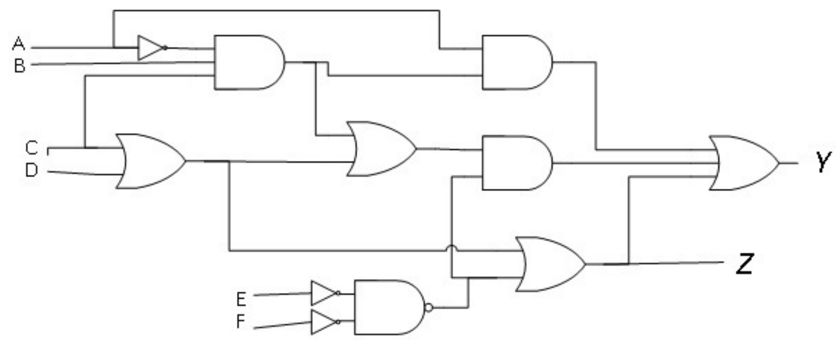






2. Consider the gate network shown.

Gate	Timing
AND	3ns
OR	4ns
NOT	1ns
NAND	2ns
NOR	2ns
XOR	2ns
Multiplexor (any size)	3ns
Decoder	5ns
Encoder	4ns



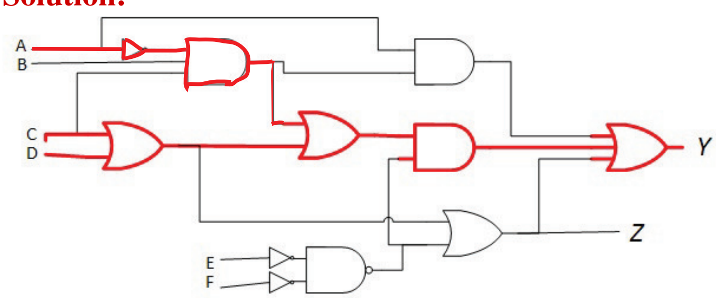
a. Calculate the delay for Y and Z given the gate delays in the table.

Solution:

Y = 15ns, Z = 8ns

b. Mark the critical path of the network. How long is the delay?

Solution:



c. Write the boolean expressions for Y and Z. Simplify.

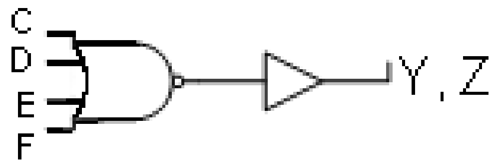
Solution:

Z = (E'F')'+ (C+D)
Z = (E+F)+ (C+D) = E + F+ C + D

Y = (A'BC + C + D)(E'F')' + AA'BC + (E'F')'+(C+D)
Y = ((A'B + 1)C + D)(E'F')' + (E+F)+(C+D)
Y = (C + D)(E+F) + (E+F)+(C+D)
Y = E + F + C + D
Z = Y = E + F + C + D

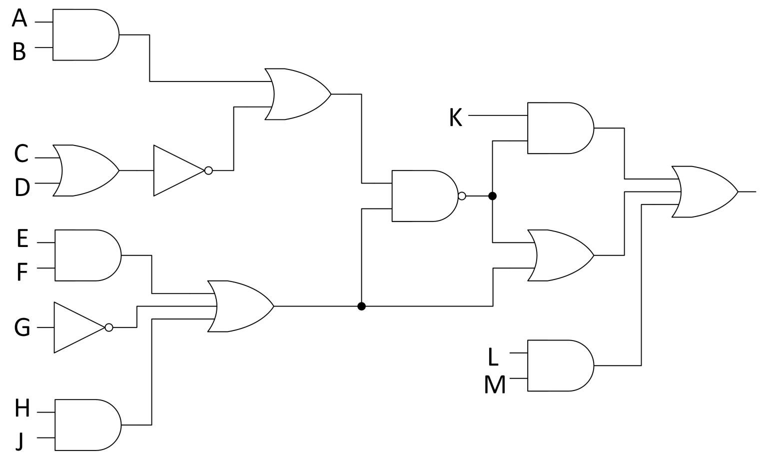
d. Redraw the network with any of the gates in the previous problem, minimizing the critical path and the delays for Y and Z.

Solution:



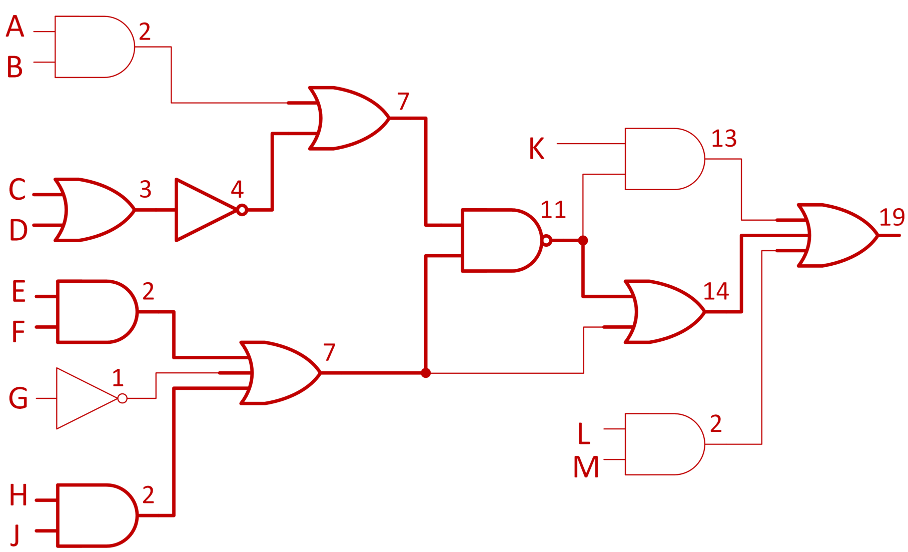
3. Consider the gate network shown below with corresponding timings.

Gate	Delay (ns)
2-input AND	2
2-input OR	3
inverter	1
NAND	4
3-input OR	5



a. Mark the critical path of the circuit on the diagram above.

Solution:



b. What is the delay of the critical path?

Solution:

19 ns

4. Design a black box that has a 3-bit input x, and a 3-bit output y. The function of the unit is y = (5x) % 8.

a. Create the truth table.

Solution:

Decimal x	(5x) mod 8 y	x_2	x_1	x_0	y_2	y_1	y_0
0	0	0	0	0	0	0	0
1	5	0	0	1	1	0	1
2	2	0	1	0	0	1	0
3	7	0	1	1	1	1	1
4	4	1	0	0	1	0	0
5	1	1	0	1	0	0	1
6	6	1	1	0	1	1	0
7	3	1	1	1	0	1	1

b. Write the Boolean expression for each output signal.

Solution:

By writing minterms/maxterms expressions and using boolean algebra to simplify we obtain the following expressions. Minterms and maxterms require same number of gates for y_2

$y_2 = x_2'x_0 + x_2x_0'$

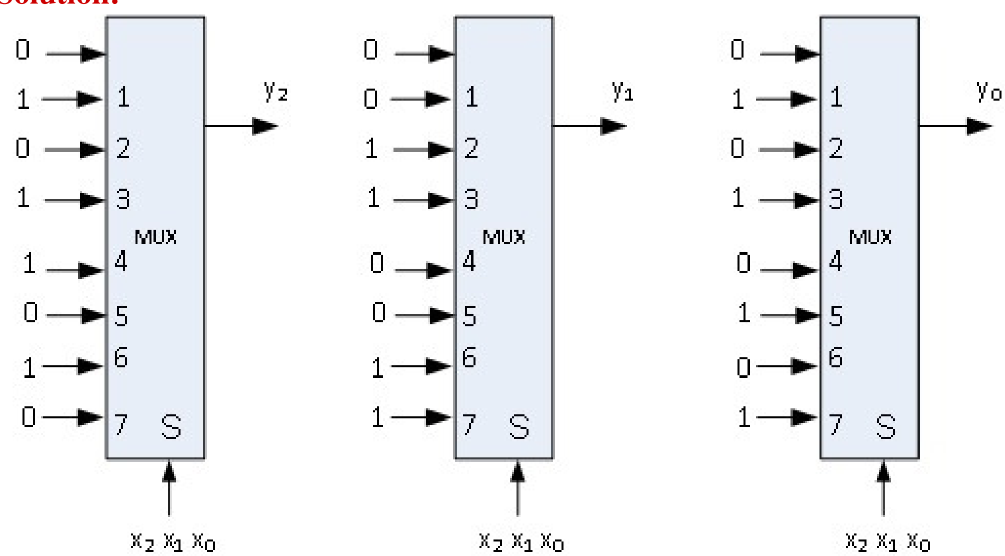
$y_2 = (x_2 + x_0)(x_2' + x_0')$

$y_1 = x_1$

$y_0 = x_0$

c. Implement each output signal with one 8-input (3-bit selector) multiplexor.

Solution:



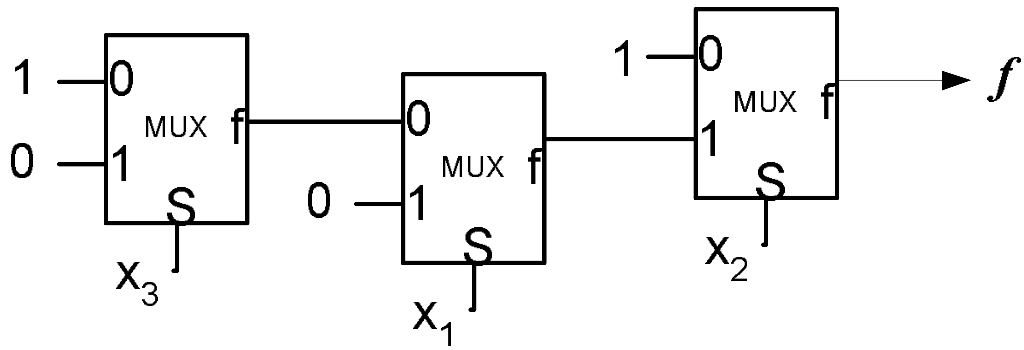
5. Implement f using 2-input multiplexors (1-bit selector)

$$f = x_1'x_2' + x_3'x_2x_1' + x_1x_2'$$

Solution:

$f = x_2'(x_1' + x_1) + x_3'x_2x_1'$
$f = x_2' + x_3'x_2x_1'$

$f = x_2'(1) + x_2(x_3'x_1')$

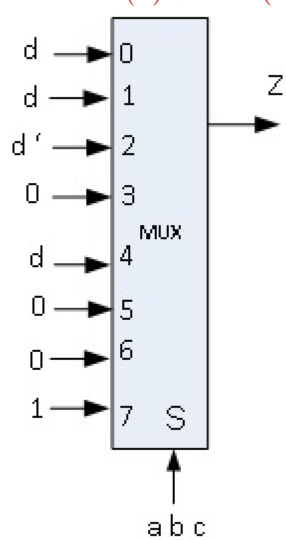


6. Given the boolean expression for Z:

$$Z = a'b'c'd + a'bc'd' + a'b'cd + ab'c'd + abcd + abcd'$$

a. Implement the Z function using a single 8-input multiplexor (3-bit selector)

Solution:
Select 3 variables to be the selector variables: abc
$Z = a'b'c'(d) + a'b'c(d) + a'bc'(d') + a'bc(0) + ab'c'(d) + ab'c(0) + abc'(0) + abc(d + d')$

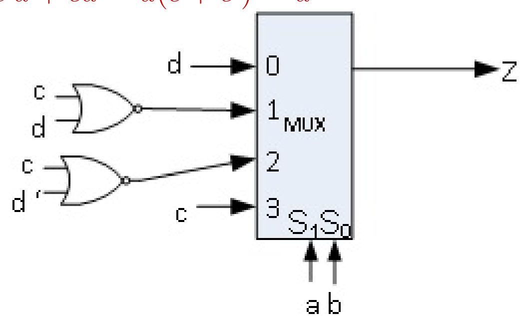


b. Implement the Z function using a single 4-input multiplexor (2-bit selector) and NOR gates.

Solution:
Select 2 variables to be the selector variables: ab

$Z = a'b'(c'd + cd) + a'b(c'd') + ab'(c'd) + ab(cd + cd')$
$Z = a'b'(c'd + cd) + a'b(c'd') + ab'(c'd) + ab(c)(d + d')$
$Z = a'b'(c'd + cd) + a'b(c'd') + ab'(c'd) + ab(c)$

Implement each of the expressions in the () using NOR gates and input into MUX.
$c'd' = (c + d)'$
$c'd = (c + d')'$
$c'd + cd = d(c + c') = d$



7. Design a system that takes a two digit base 3 number as input and produces the corresponding binary value (Z). (Example input: $12_3 = 0101_2$).
   a. Create the truth table.

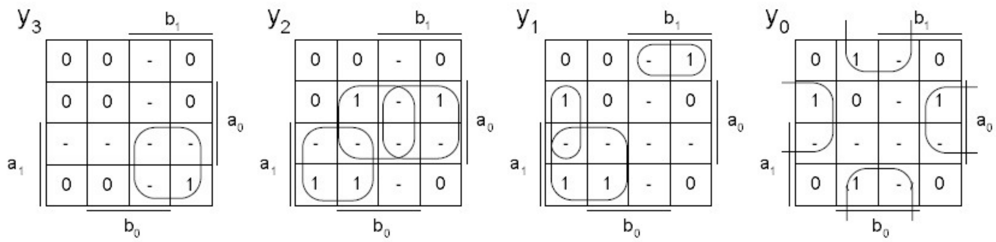
   Solution:

   Since the largest value of the input $22_3 = 8_{10}$, we need 4 bits to represent output.

a_1	a_0	b_1	b_0	y_3	y_2	y_1	y_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	-	-	-	-
0	1	0	0	0	0	1	1
0	1	0	1	0	1	0	0
0	1	1	0	0	1	0	1
0	1	1	1	-	-	-	-
1	0	0	0	0	1	1	0
1	0	0	1	0	1	1	1
1	0	1	0	1	0	0	0
1	0	1	1	-	-	-	-
1	1	0	0	-	-	-	-
1	1	0	1	-	-	-	-
1	1	1	0	-	-	-	-
1	1	1	1	-	-	-	-

   b. Create k-maps for each output signal to simplify each boolean expression.

   Solution:



$y_3 = a_1 b_1$

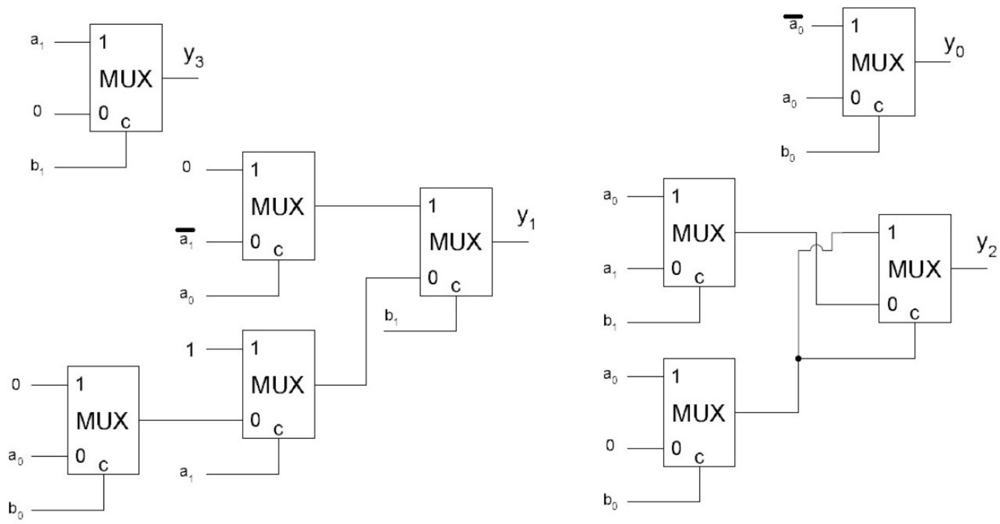
$y_2 = (a_1\overline{b_1} + a_0 b_1) + a_0 b_0$

$y_1 = (a_1 + a_0\overline{b_0})\overline{b_1} + \overline{a_1 a_0} b_1$

$y_0 = (\overline{a_0}) b_0 + a_0\overline{b_0}$

c. Implement the 2 most significant output signals using the minimal number of 2-input multiplexors (selector is 1-bit).

Solution:

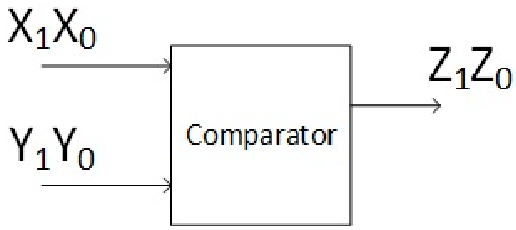


8. Design a black box that compares two 2-bit binary numbers (X_1X_0, Y_1Y_0). The 2-bit output, (Z_1,Z_0) should encode the conditions greater (X>Y), equal (X==Y) and smaller (X<Y) as shown below.

	Z_1	Z_0
Equal	0	0
Smaller	0	1
Greater	1	0

a. Draw the black box with all inputs and outputs signals. Label each signal. How many total input bits? How many total output bits?

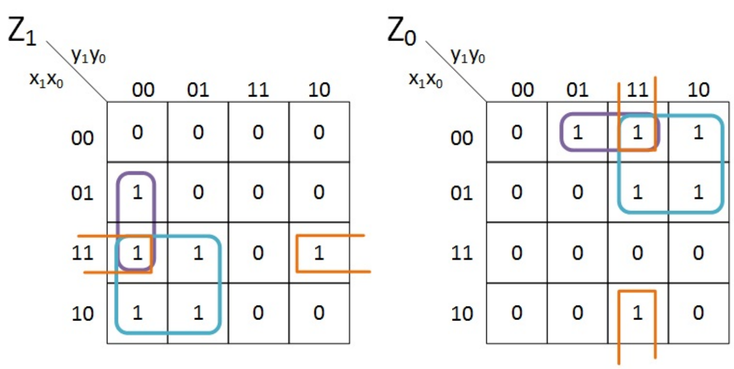
Solution:



b. Create the truth table. Write the boolean expressions and simplify using identities or kmaps.

Solution:

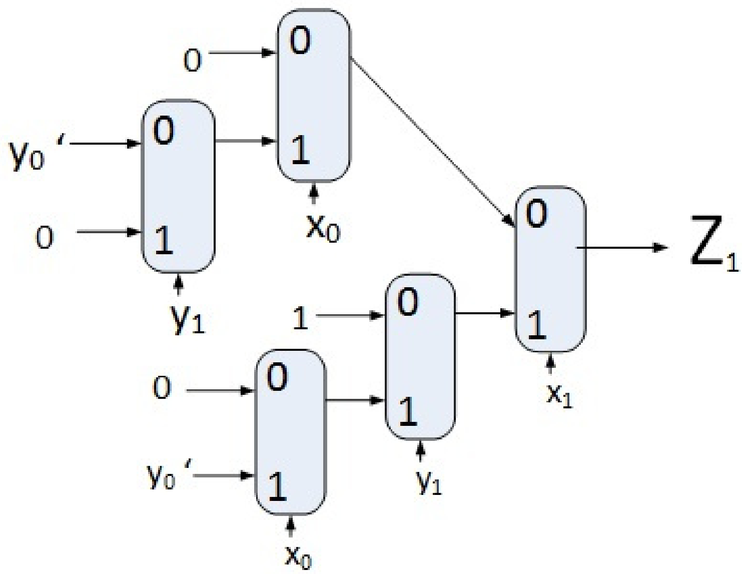
X_1	X_0	Y_1	Y_0	Z_1	Z_0
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	1
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	0	0
0	1	1	0	0	1
0	1	1	1	0	1
1	0	0	0	1	0
1	0	0	1	1	0
1	0	1	0	0	0
1	0	1	1	0	1
1	1	0	0	1	0
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	0	0



$$Z_1 = X_1Y_1' + X_1X_0Y_0' + X_0Y_1'Y_0'$$
$$Z_0 = Y_1Y_0X_0' + X_1'X_0'Y_0 + Y_1X_1'$$

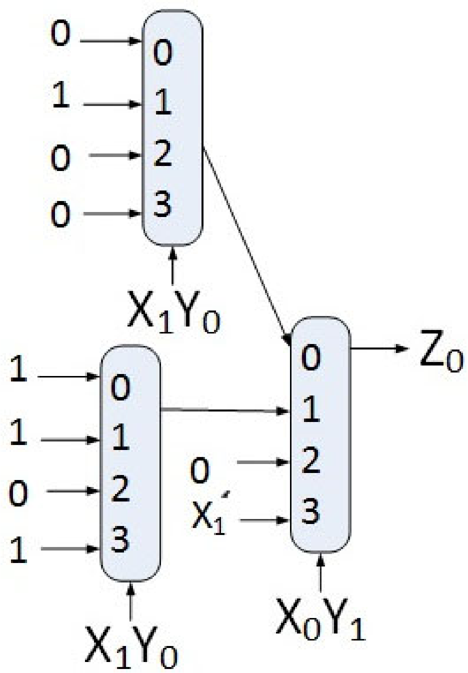
c. Implement Z_1 with 2-input (1-bit selector) Multiplexors.

Solution:



d. Implement Z_0 with 4-input (2-selector) Multiplexors.

Solution:



9. Design a black box that converts a decimal digit (X) in binary to its 4-bit Excess-3 encoding (Z) (http://en.wikipedia.org/wiki/Excess-3). In Excess-3, numbers are represented as decimal digits, and each digit is represented by four bits as the digit value + 3 (the ”excess” amount). Eg. 7 becomes 10, 9 becomes 12.

   a. Create the truth table. Write the boolean expressions and simplify using identities or kmaps.

   Solution:

X3	X2	X1	X0	Y1	Y0	Z1	Z0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	-	-	-	-
1	0	1	1	-	-	-	-
1	1	0	0	-	-	-	-
1	1	0	1	-	-	-	-
1	1	1	0	-	-	-	-
1	1	1	1	-	-	-	-

Y_1

$X_3X_2 \backslash X_1X_0$	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	-	-	-	-
10	1	1	-	-

Y_0

$X_3X_2 \backslash X_1X_0$	00	01	11	10
00	0	1	1	1
01	1	0	0	0
11	-	-	-	-
10	0	1	-	-

Z_1

$X_3X_2 \backslash X_1X_0$	00	01	11	10
00	1	0	1	0
01	1	0	1	0
11	-	-	-	-
10	1	0	-	-

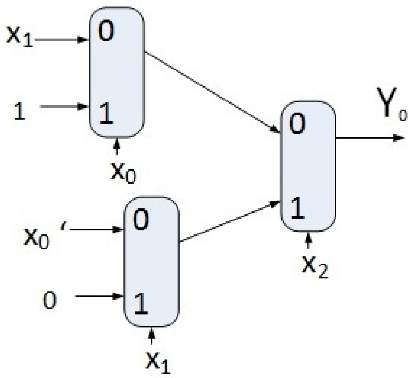
Z_0

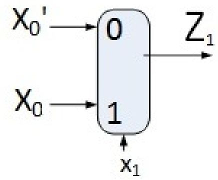
$X_3X_2 \backslash X_1X_0$	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	-	-	-	-
10	1	0	-	1

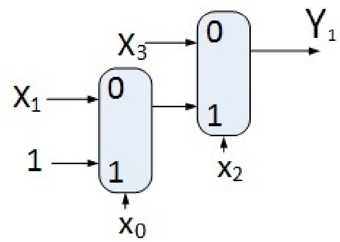
$Z_1 = X_1X_0 + X_1'X_0'$
$Z_0 = X_0'$
$Y_1 = X_0X_2 + X_1X_2 + X_3$
$Y_0 = X_2X_1'X_0' + X_2'X_0 + X_1X_2'$

b. Implement Z_1 with 2-input (1-bit selector) Multiplexors.

Solution:







10. The binary digit vector x_2, x_1, x_0 represents integer x in the range $0 \leq x \leq 7$. Design a digital system using only NOR gates that implements the following function $Z = min(u, w)$ where $u = |x^2 + 2x - 4|$ and $w = |x^2 - 3|$.

a. Create the Truth Table. How many bits are needed to represent Z?

b. Find the minimal boolean expressions using Karnough maps for the most significant bit (MSB) and least significant bit (LSB) of Z.

c. Create the minimal two level NOR-NOR implementation for the outputs in (b).

Solution:

x_2	x_1	x_0	u	w	z	z_5	z_4	z_3	z_2	z_1	z_0
0	0	0	4	3	3	0	0	0	0	1	1
0	0	1	1	2	1	0	0	0	0	0	1
0	1	0	4	1	1	0	0	0	0	0	1
0	1	1	11	6	6	0	0	0	1	1	0
1	0	0	20	13	13	0	0	1	1	0	1
1	0	1	31	22	22	0	1	0	1	1	0
1	1	0	44	33	33	1	0	0	0	0	1
1	1	1	59	46	46	1	0	1	1	1	0

From the K-maps shown below we obtain:

$$z_5 = x_2 x_1 = \overline{\overline{x_2} + \overline{x_1}}$$

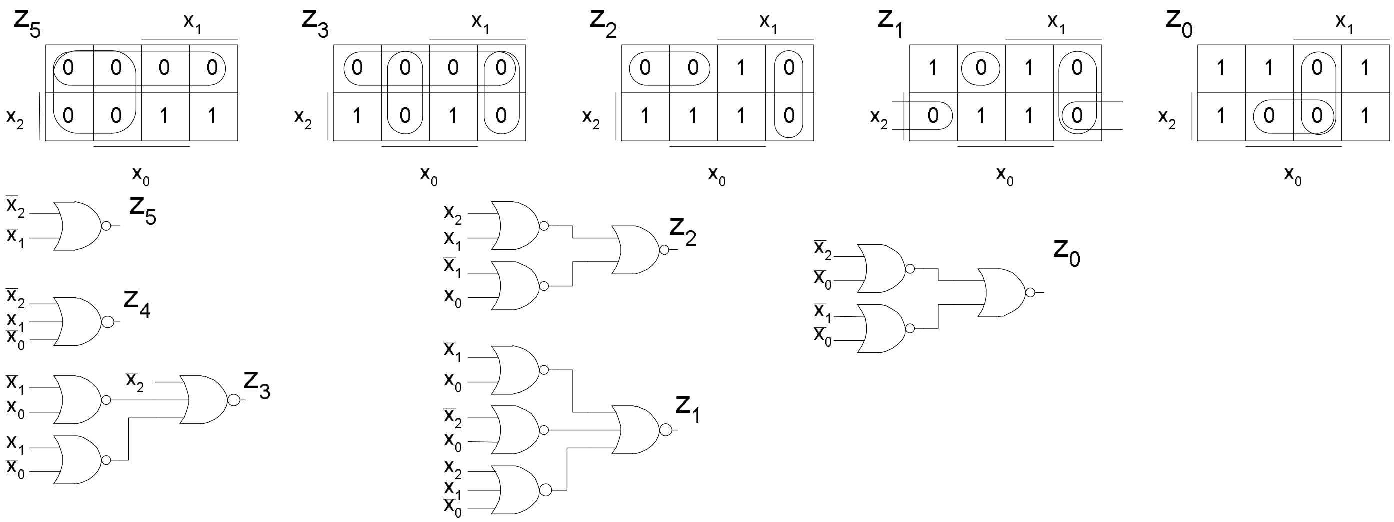
$$z_4 = x_2 \overline{x_1} x_0 = \overline{\overline{x_2} + x_1 + \overline{x_0}}$$

$$z_3 = x_2(x_1 + \overline{x_0})(x_1 + \overline{x_0}) = \\ = \overline{\overline{x_2} + \overline{(\overline{x_1} + x_0)} + \overline{(x_1 + \overline{x_0})}}$$

$$z_2 = (x_2 + x_1)(\overline{x_1} + x_0) = \\ = \overline{\overline{(x_2 + x_1)} + \overline{(\overline{x_1} + x_0)}}$$

$$z_1 = (\overline{x_1} + x_0)(\overline{x_2} + x_0)(x_2 + x_1 + \overline{x_0}) = \\ = \overline{\overline{(\overline{x_1} + x_0)} + \overline{(\overline{x_2} + x_0)} + \overline{(x_2 + x_1 + \overline{x_0})}}$$

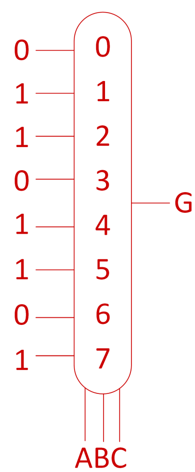
$$z_0 = (\overline{x_2} + \overline{x_0})(\overline{x_1} + \overline{x_0}) = \\ = \overline{\overline{(\overline{x_2} + \overline{x_0})} + \overline{(\overline{x_1} + \overline{x_0})}}$$



11. For the following function: $G = A'BC' + AB' + ABC + B'C$

    a. Implement the function using a single 3-bit selector multiplexer (i.e., an 8:1 multiplexer).

    Solution:

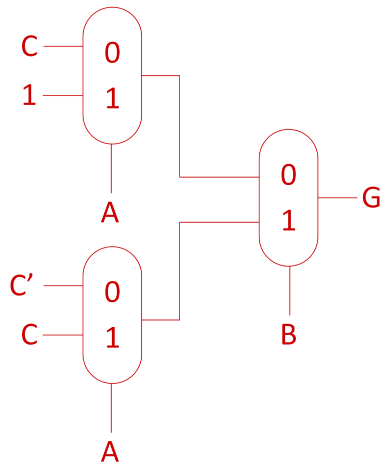
    

    $G = ABC + A'BC' + AB'C + AB'C' + A'B'C$

    b. Implement the function using 1-bit selector multiplexers (i.e., 2:1 multiplexers). Use B as the selector bit for the first level of multiplexers (nearest to the output) and A for the second level of multiplexers (nearest to the inputs).

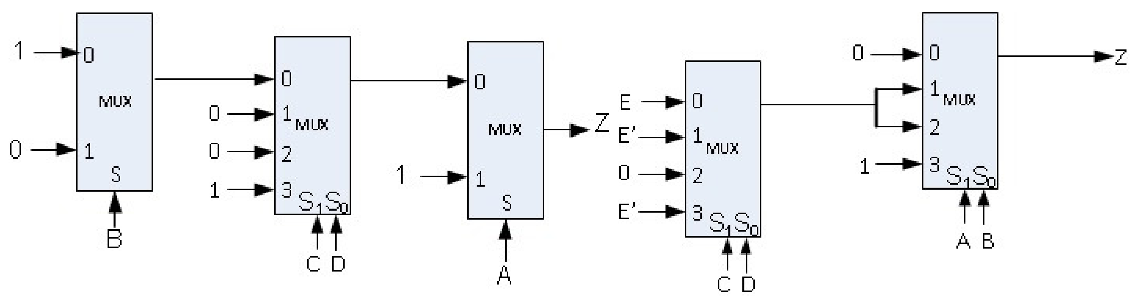
    Solution:

    $G = A'BC' + AB' + ABC + B'C$
    $= B(AC + A'C') + B'(A + C)$
    $AC + A'C' = A(C) + A'(C')$ and $A + C = A(1) + A'(C)$

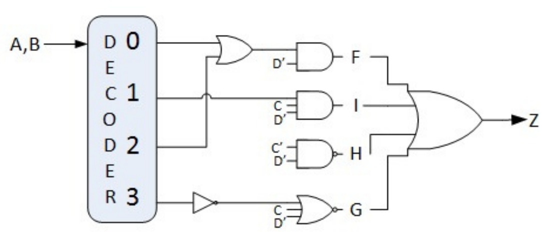


The following questions are from past quizzes. Solutions will not be provided, but you are welcome to discuss them and your own proposed answers to them on Piazza

12. What is the Boolean equation for the circuit below? Do not minimize.



13. Consider the figure shown.

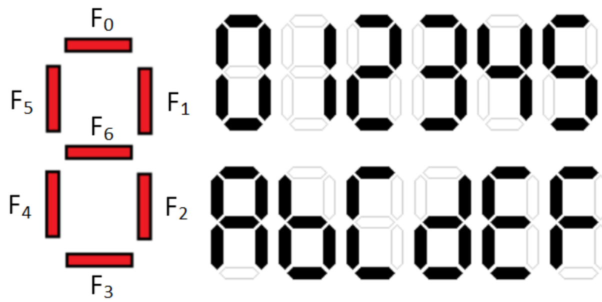


a. Write the Boolean expression for Z(A,B,C,D).

b. Simplify using k-maps. Write the minimal simplified Boolean Expression for Z(A,B,C,D).

c. Is your expression in (c) in SOP or POS form? ____________
Z(A,B,C,D) = $\sum$ m(____)
How many literals are in (b)? ________________________

14. Implement a 7-segment display which outputs 0-5 when input C is 0, and A-F when C is 1. Output is 0 when segment is off and 1 when segment is lit.



Only fill the columns necessary for Problem 5.

C	X_2	X_1	X_0	Letter	F_6	F_5	F_4	F_3	F_2	F_1	F_0
0	0	0	0	0							
0	0	0	1	1							
0	0	1	0	2							
0	0	1	1	3							
0	1	0	0	4							
0	1	0	1	5							
0	1	1	0	-							
0	1	1	1	-							
1	0	0	0	A							
1	0	0	1	B							
1	0	1	0	C							
1	0	1	1	D							
1	1	0	0	E							
1	1	0	1	F							
1	1	1	0	-							
1	1	1	1	-							

a. Using kmaps write the minimal boolean expression for each output, F_6-F_0.

b. Implement each output, F_6-F_0, using 2-input, 4-input, or 8-input multiplexors.