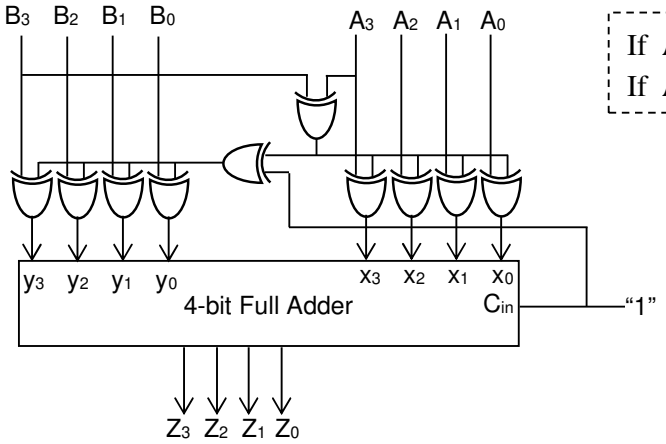
İTÜ Computer Engineering Dep.
Assoc. Prof. Feza BUZLUCA
Asst. Prof. Sanem KABADAYI
Prof. Mustafa KAMAŞAK
January 10, 2024

BLG 231E DIGITAL CIRCUITS FINAL EXAM SOLUTIONS

QUESTION 1 (35 Points):
Note that Parts (a) and (b) below are not related.

a) [25 Points]
i. [15]



If $A_3 \oplus B_3 = 0,\ Z = A + \overline{B} + 1$
If $A_3 \oplus B_3 = 1,\ Z = \overline{A} + B + 1$

Notes:
- Ten 2-input XOR gates are sufficient.
- Negation requires only one XOR gate.
- It is also possible to use 3-input XOR gates for the input B. In this case, the extra XOR gate for negation is not necessary.

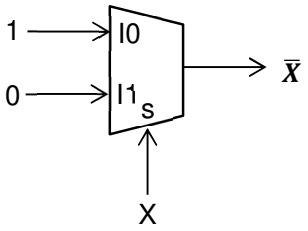
ii. [10]
During subtraction, overflow can occur only if A and B have opposite signs. Therefore, for overflow, the operation must be `Z = B-A`.
There are only two cases that can generate overflow:
B: negative, A: positive, Z: positive
B: positive, A: negative, Z: negative

Overflow: Z_3 \ B_3A_3	00	01	11	10
0	0	0	0	1
1	0	1	0	0

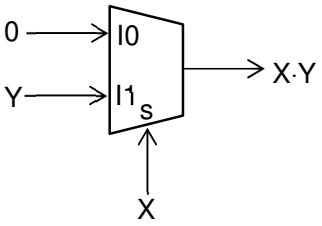
$$\textbf{Overflow} = B_3\overline{A_3}\,\overline{Z_3} + \overline{B_3}A_3Z_3$$

b) [10 points]

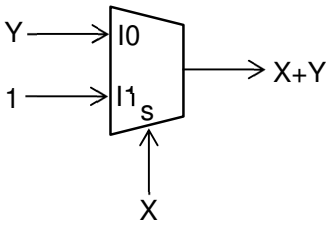
{NOT, AND} and {NOT, OR} are two functionally complete sets. If a 2:1 multiplexer (MUX) is a universal gate, we must be able to realize one of these sets using only 2:1 multiplexers.
It is possible to realize all operators of the Boolean algebra.



NOT gate

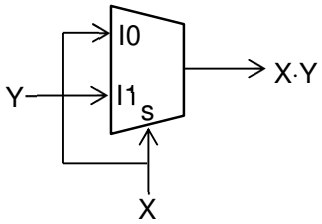


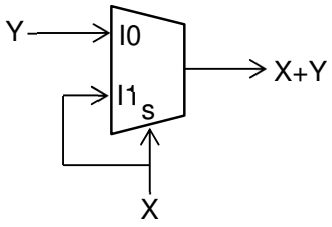
AND gate



OR Gate

Other possible solutions:





Equation of 2:1 MUX:

$Z = \bar{s}I0 + sI1$

NOT: s = X, I0 = 1, I1 = 0: $Z = \overline{X} \cdot 1 + x \cdot 0 = \overline{X}$

AND: s = X, I0 = X, I1 = Y: $Z = \overline{X} \cdot X + X \cdot Y = X \cdot Y$

OR : s = X, I0 = Y, I1 = X: $Z = \overline{X} \cdot Y + X \cdot X = \overline{X} \cdot Y + X = X + Y$

The answer is YES. We can use a 2:1 multiplexer (MUX) as a universal gate because we can realize functionally complete sets using only 2:1 multiplexers.

Note:

- The statement "Since a mux contains AND, OR, NOT gates, it is a universal gate" is not correct.
- Implementing only one of the sets {NOT, AND} or {NOT, OR} is sufficient.

QUESTION 2 (35 Points):
Note that Parts (a) and (b) below are not related.

a) [20 Points]
i.

Q+ Q / XY	0	1
00	0	1
01	0	1
11	0	0
10	1	1

Q+ = X'Q + XY'

ii.
Required transitions ($Q \rightarrow Q^+$) for the flip-flop to be implemented

QQ+ / XY	0	1
00	00 (0)	11 (1)
01	00 (0)	11 (1)
11	00 (0)	10 (β)
10	01 (α)	11 (1)

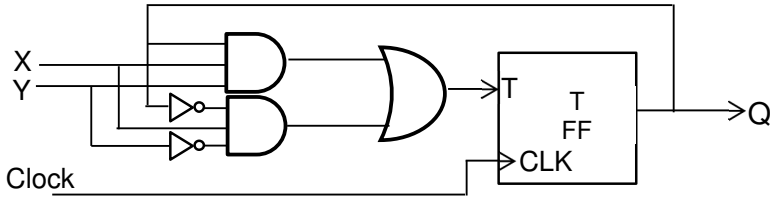
Transition Table for T flip-flop:

QQ+	T
00 (0)	0
01 (α)	1
10 (β)	1
11 (1)	0

Karnaugh map to determine the equation of the circuit to drive the input of the T flip-flop:

T Q / XY	0	1
00	0	0
01	0	0
11	0	1
10	1	0

T = XYQ + XY'Q'



Note:
You cannot implement this circuit connecting some gates to the output of the flip-flop only. In this way, you cannot change (control) the value (state) of the flip-flop.

b) [15 Points]
Remember: NMOS: if VGS>0, ON; if VGS=0, OFF
The pull-down network will make Z Low when T1 is ON AND (T2 OR T3 is ON).
Hence,

A	B	C	T1	T2	T3	Z
L	L	L	OFF	OFF	OFF	H
L	L	H	OFF	ON	OFF	H
L	H	L	OFF	OFF	ON	H
L	H	H	OFF	ON	ON	H
H	L	L	ON	OFF	OFF	H
H	L	H	ON	OFF	ON	L
H	H	L	ON	ON	OFF	L
H	H	L	ON	ON	ON	L

Using positive logic

Z C / AB	0	1
00	1	1
01	1	1
11	0	0
10	1	0

Z = A' + B'C'

QUESTION 3 (30 Points):

a) [10 points]

i. [5 points]

Moore because Z = Q1. The output depends only on the current state, and not the current input. $O = g(S)$

ii. [5 points]

Since the circuit contains two flip-flops, there are two state variables, i.e., Q1 and Q0. This FSM has 4 states because Q1Q0 = 00, 01, 10, 11

b) [10 points]

Q1	Q0	X	U1	U0	Q1+	Q0+	Z
0	0	0	0	0	1	1	0
0	0	1	0	1	1	0	0
0	1	0	0	0	1	0	0
0	1	1	1	1	0	1	0
1	0	0	0	0	0	1	1
1	0	1	1	1	1	0	1
1	1	0	1	0	1	0	1
1	1	1	1	1	1	1	1

(2) (1) (3) (5) (4)

Step 1:Since the input of Q0 is just U0=X, filling in the column for U0 is straightforward.
Step 2: We have the equation for U1:
U1= XQ1+XQ0+Q0Q1
This means that at least two out of Q1, Q0, and X have to be 1 for U1 to be 1. We will in the U1 value for each (Q1, Q0, X) combination.
Step 3: In the problem statement, all the combinations of Q, U, and Q+ have been provided.

Q	U	Q+
0	0	1
0	1	0
1	1	1
1	0	0

This table indicates that Q+=Q'U'+QU

c) [10 points]

The truth table has been filled in above.

Step 4: We fill in Q1+ and Q0+ columns based on the Q-U-Q+ table we found in Part(b).
Step 5: Z = Q1, so we copy the column for Q1 into the Z column.