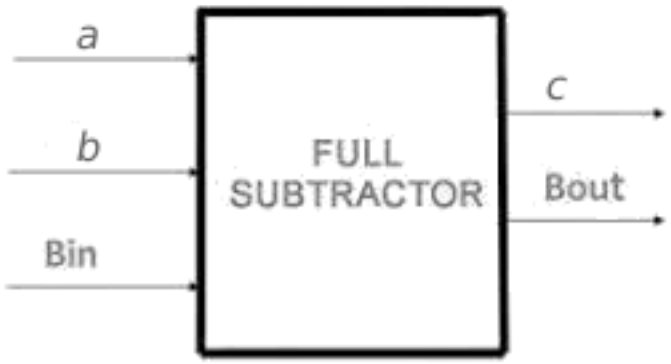
Problem Set – 2 Solutions

CS 230, Spring 2023

Questions

1. Consider a full-binary subtractor that accepts three input bits *a*, *b*, and B_{in}, and outputs *c* and B_{out}. *a* and *b* are the operands for the subtraction operation and B_{in} is the borrow taken by the previous digit. *c* is the resulting bit for the operation *(a - b)* and B_{out} is the borrow taken from the next digit for the subtraction. Then the boolean expression for *c* and B_{out} is:
   a. $c = \bar{a} \oplus b \oplus B_{in}\,;\ B_{out} = \bar{a}B_{in} + a\bar{b} + bB_{in}$
   b. $c = a \oplus \bar{b} \oplus B_{in}\,;\ B_{out} = \bar{a}B_{in} + \bar{a}b + bB_{in}$
   c. $c = a \oplus b \oplus B_{in}\,;\ B_{out} = \bar{a}B_{in} + \bar{a}b + bB_{in}$
   d. $c = a \oplus b \oplus B_{in}\,;\ B_{out} = \bar{a}B_{in} + \bar{a}b + b\overline{B_{in}}$

A full subtractor will look like this:



Truth Table of Full Subtractor:

a	*b*	B_{in}	*c*	B_{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Solving for ***c***

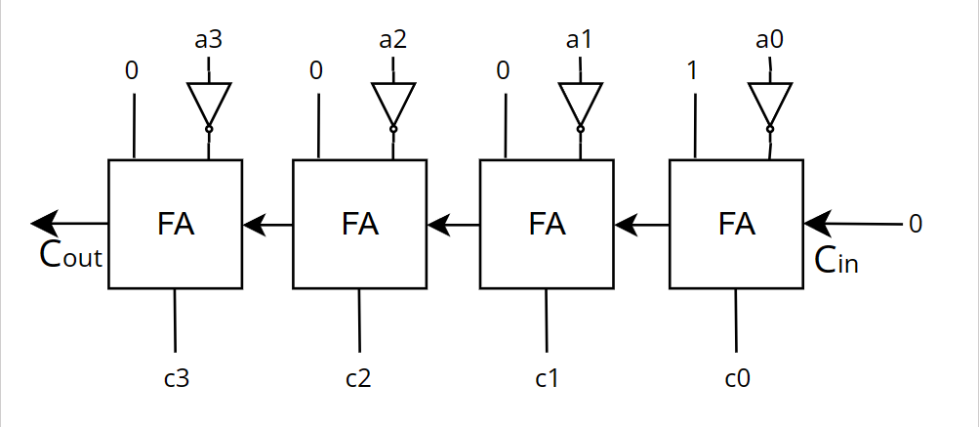
$$
\begin{aligned}
c &= \bar{a}\bar{b}B_{in} + \bar{a}b\overline{B_{in}} + a\bar{b}\overline{B_{in}} + abB_{in} \\
&= B_{in}(ab + \bar{a}\bar{b}) + \overline{B_{in}}(a\bar{b} + \bar{a}b) \\
&= B_{in}\overline{(a \oplus b)} + \overline{B_{in}}(a \oplus b) \\
&= a \oplus b \oplus B_{in}
\end{aligned}
$$

Solving for B_{out} using K-map.

a\b,Bout	00	01	11	10
0	0 (0)	1 (1)	1 (3)	1 (2)
1	0 (4)	0 (5)	1 (7)	0 (6)

$$B_{out} = \bar{a}B_{in} + \bar{a}b + bB_{in}$$

2. Consider the following logic circuit. Suppose binary numbers are represented in 2's complement form.



a. What is the output $C = c_3c_2c_1c_0$ for input $A = 0101$?
b. What does the above circuit do?
   i. Outputs $A + 1$
   ii. Outputs $\bar{A}$
   iii. Outputs $-A$
   iv. Outputs $-A + 1$

a. It is 4 bit adder in which one operand is fixed 0001 and the second operand is $\bar{A}$

We have $A = 0101$

$\bar{A} = 1010$

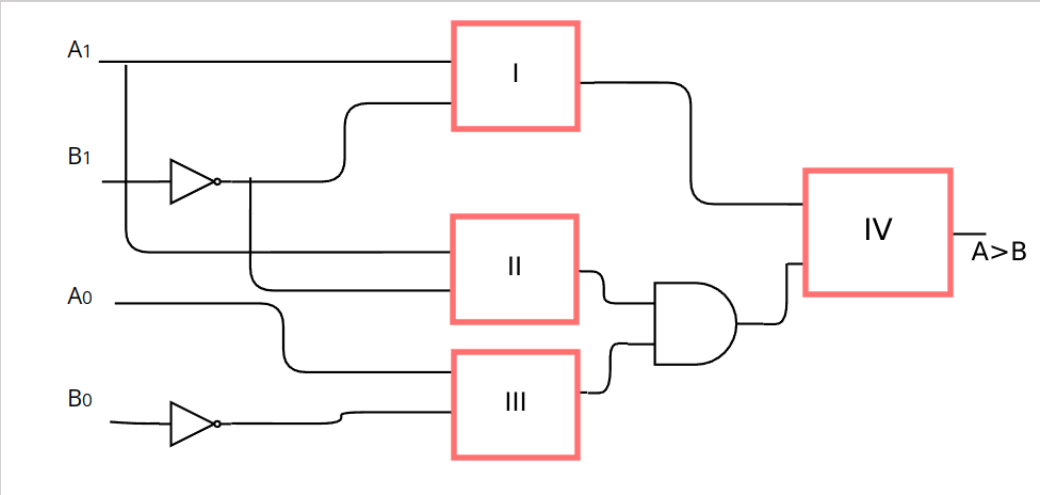
$C = \bar{A} + 0001$

$= 1010 + 0001$

$= \mathbf{1011}$

b. The output of the circuit is $\bar{A} + 1$, which represents $-\boldsymbol{A}$ in 2's complement form.

3. You have to design a two bit greater than comparator logic circuit ($A > B$). Following incomplete circuit is provided to you. Name appropriate logic gate for each red box to complete the circuit.



The truth table for the comparator:

A_1	A_0	B_1	B_0	C
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

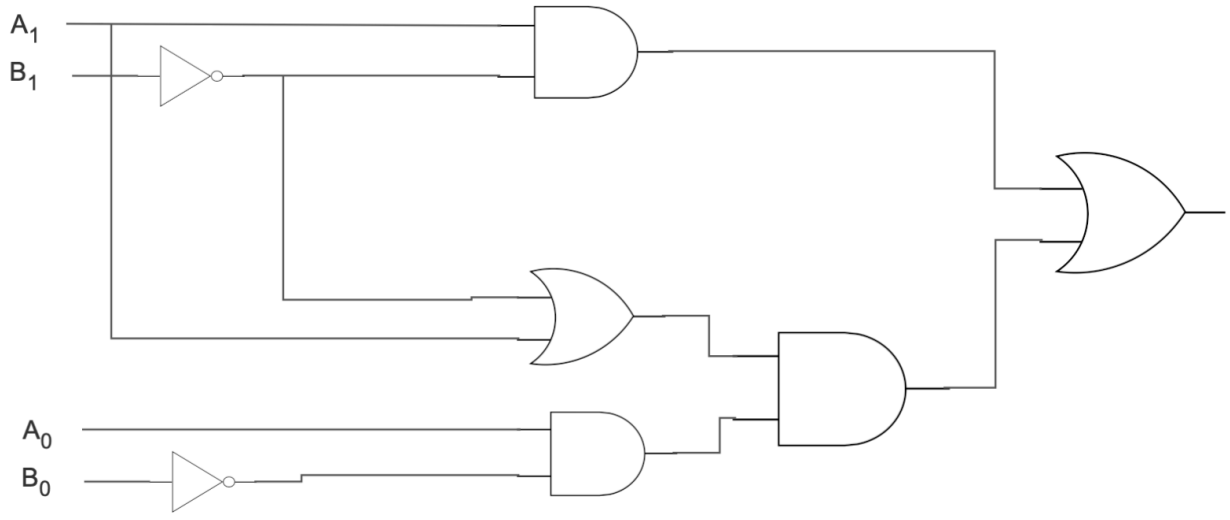
Solving using K-map

A1,A0 \ B1,B0	00	01	11	10
00	0 (0)	0 (1)	0 (3)	0 (2)
01	1 (4)	0 (5)	0 (7)	0 (6)
11	1 (12)	1 (13)	0 (15)	1 (14)
10	1 (8)	1 (9)	0 (11)	0 (10)

$C = A_0\overline{B_1B_0} + A_1A_0\overline{B_0} + A_1\overline{B_1}$
$= (A_1 + \overline{B_1})A_0\overline{B_0} + A_1\overline{B_1}$

Mapping the expression with circuit:

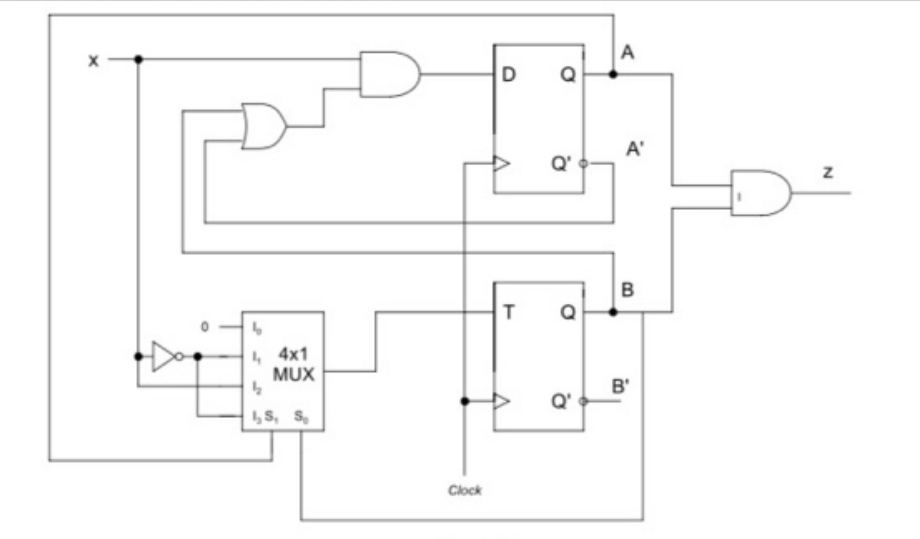
I) AND Gate
II) OR Gate
III) AND Gate
IV) OR Gate



4. A toggle flip-flop (T flip-flop) toggles its output when the input T = 1 during clock signal transition. For input T = 0, the output remains same. The state table of T flip-flop is as follows:

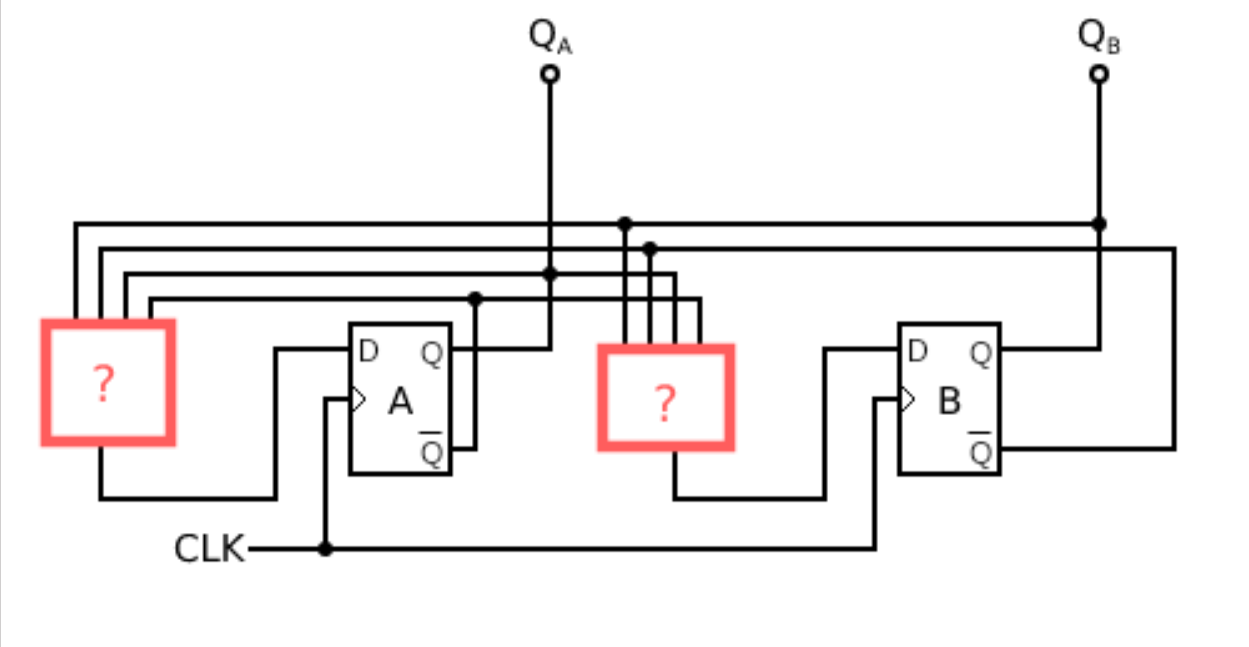
T	Q_n	Q_{n+1}	
0	0	0	Unchanged/hold
0	1	1	Unchanged/hold
1	0	1	Toggle
1	1	0	Toggle

Complete the state table of the following circuit.



Current State		Input	Next State		Output
A	**B**	**x**	**A**	**B**	**z**
0	0	0	0	0	0
0	0	1	1	0	0
0	1	0	0	0	0
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	0	0	0
1	1	1	1	1	1

5. You have to design a 2 bit mod 3 binary counter using D flip-flops. The counter should follow the following sequence:
0, 1, 2, 0, 1, 2, 0……
You are given the following circuit.



Write the Boolean expression for D_A *and* D_B in terms of $Q_A, \overline{Q_A}, Q_B$ *and* $\overline{Q_B}$ to complete the circuit.

The state table will be:

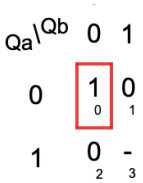
Q_A	Q_B	Q_{A+1}	Q_{B+1}	D_A	D_B
0	0	0	1	0	1
0	1	1	0	1	0
1	0	0	0	0	0
1	1	X	X	X	X

K-map for D_A

Q_a \ Q_b	0	1
0	0 (0)	1 (1)
1	0 (2)	- (3)

$$\boldsymbol{D_A = Q_B}$$

K-map for D_B



$$\boldsymbol{D_B = \overline{Q_A}\ \overline{Q_B}}$$

6. What is the minimum number of D flip-flops to design a counter for the sequence 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 0, 0, 1, 1…?

Since each number is repeated two times, we can assign two different states to each number. The sequence then can be written as: 0_1, 0_2, 1_1, 1_2, 2_1, 2_2, 3_1, 3_2, 4_1, 4_2, 5_1, 5_2, 6_1, 6_2, 7_1, 7_2, 0_1, 0_2….

We have 16 distinct states.

To represent m distinct states, we need a minimum of $\boldsymbol{\lceil log_2(m) \rceil}$ flip-flops, because each flip-flop can toggle between two states.

So, for 16 states, you need **4 flip-flops**.