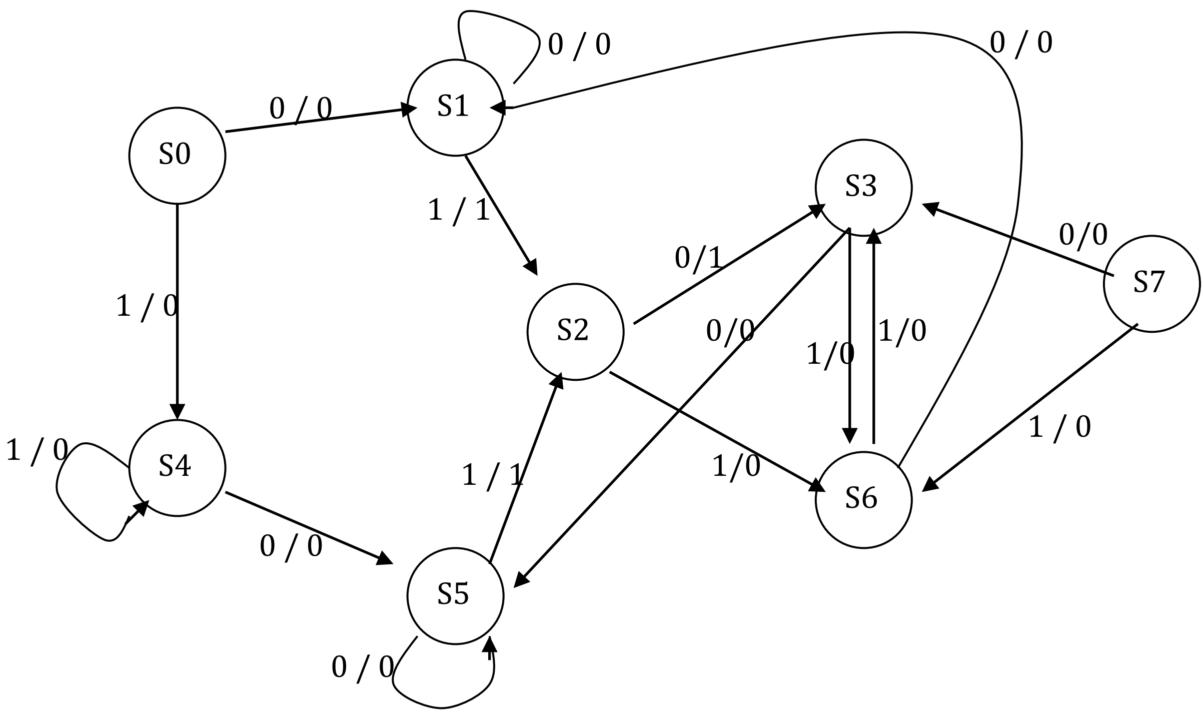
Exercises on the topics of class 21

Exercises with solutions

Ex. 1. Given the following automaton with starting state S0:



Minimize it and then, from the minimal automaton, derive the corresponding sequential net whose combinatorial part is done with a PLA and the sequential part with FFs of kind JK.

SOLUTION:

Notice that S7 is unreachable from S0 and so can be directly deleted. The triangular table is:

S1	X					
S2	X	X				
S3	1,5 4,6	X	X			
S4	1,5	X	X	4,6		
S5	X		X	X	X	
S6	3,4	X	X	1,5	1,5 3,4	X
	S0	**S1**	**S2**	**S3**	**S4**	**S5**

The resulting states are T0 = {S0, S3, S4, S6} , T1 = {S1, S5} , T2 = {S2}, and so the minimal automaton is:

	0	**1**
T0	T1 / 0	T0 / 0
T1	T1 / 0	T2 / 1
T2	T0 / 1	T0 / 0

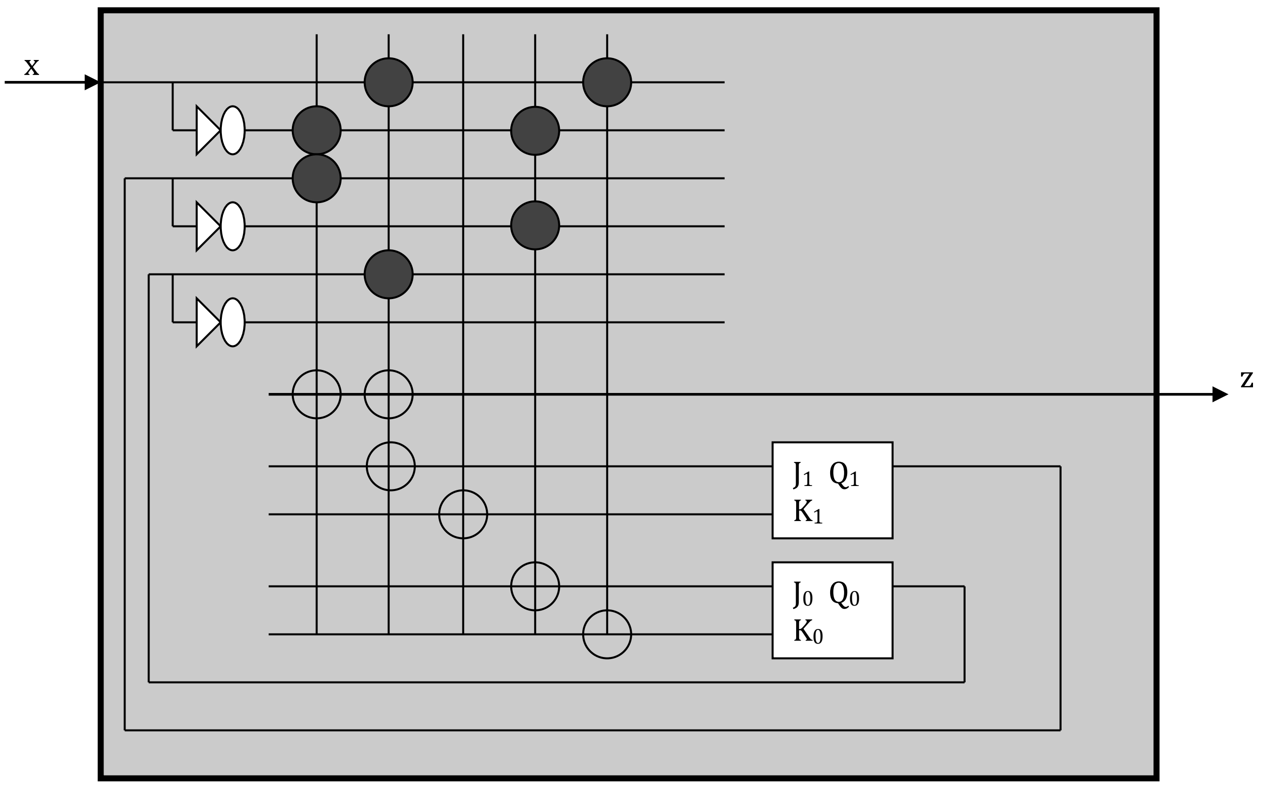
This automaton yields the following future states table (with the encoding Q1Q0 = 00 for T0, Q1Q0 = 01 for T1 and Q1Q0 = 10 for T2) :

x Q1 Q0	**Q1 Q0 (t+1)**	**z**	**J1 K1 J0 K0**
0 0 0	0 1	0	0 - 1 -
0 0 1	0 1	0	0 - - 0
0 1 0	0 0	1	- 1 0 -
0 1 1	- -	-	- - - -
1 0 0	0 0	0	0 - 0 -
1 0 1	1 0	1	1 - - 1
1 1 0	0 0	0	- 1 0 -
1 1 1	- -	-	- - - -

With Karnaugh maps, we obtain the minimal expressions:

$Z = \underline{x}\,Q1 + x\,Q0$ $J1 = x\,Q0$ $K1 = 1$ $J0 = \underline{x}\,\underline{Q1}$ $K0 = x$

and so the circuit is:



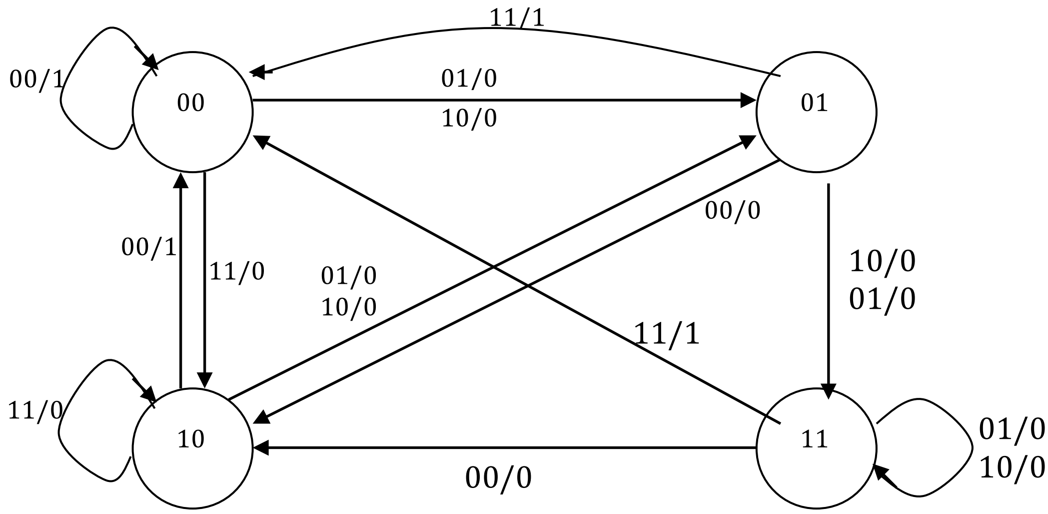
Ex.2. Design a circuit that receives two binary strings and gives in output 1 if and only if the sum of the values associated to the input strings received so far is a multiple of 4. Assume that the first received digit is the most signifying one of the string itselfstringa stessa. For example:

Input: 100100…
100010…
Ouput: 011000…

Work with the minimal automaton and with the resulting minimal expressions (obtained with gates NAND, NOR, XOR and NXOR, if possible). Use flip-flops of kind JK and don't use any module for the sum.

SOLUTION:

States are associated to the last two bits of the binary representation of the sum of the values received so far. Hence:



Its tabular representation is

	00	**01**	**10**	**11**
S0	S0/1	S1/0	S1/0	S2/0
S1	S2/0	S3/0	S3/0	S0/1
S2	S0/1	S1/0	S1/0	S2/0
S3	S2/0	S3/0	S3/0	S0/1

We can easily see that it is minimizable as follows:

	00	**01**	**10**	**11**
T0	T0/1	T1/0	T1/0	T0/0
T1	T0/0	T1/0	T1/0	T0/1

Future states table is:

x	y	Q(t)	Q(t+1)	z	J	K
0	0	0	0	1	0	-
0	0	1	0	0	-	1
0	1	0	1	0	1	-
0	1	1	1	0	-	0
1	0	0	1	0	1	-
1	0	1	1	0	-	0
1	1	0	0	0	0	-
1	1	1	0	1	-	1

With the KMs, we have:

$$z = \underline{x}\,\underline{y}\,\underline{Q} + x\,y\,Q \qquad J = \underline{x}\,y + x\,\underline{y} \qquad K = x\,y + \underline{x}\,\underline{y}$$

The last two expressions can be equivalently written as

$$J = x \oplus y \qquad K = x \otimes y$$

From them, we obtain the final circuit.

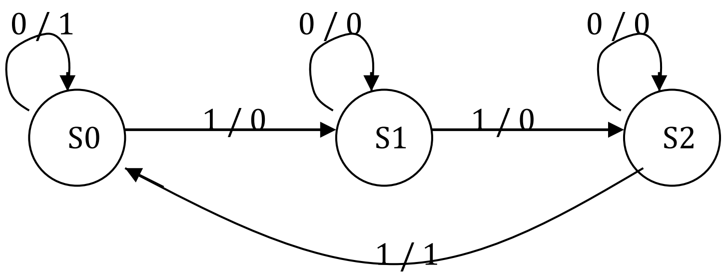
Ex. 3. Design a sequential net that, given in input a binary string, gives in output 1 if and only if the number of 1s received so far is a multiple of 3. Use flip-flops of kind T. From this, draw the net that only uses flip-flops of kind JK (REMARK: you don't have to perform another systhesis!!)

SOLUTION:

The required automaton has to check whether the number *n* of the 1s received so far is:

- $n = 3 \cdot k$
- $n = 3 \cdot k + 1$
- $n = 3 \cdot k + 2$

for some *k*, and only in the first case return 1. The first condition is S0, the second one is S1 and the third one is S2. So, the automaton is



Let's encode S0 as Q1 Q0 = 00, S1 as Q1 Q0 = 01 and S2 as Q1 Q0 = 10 (combination Q1 Q0 = 11 is not used).

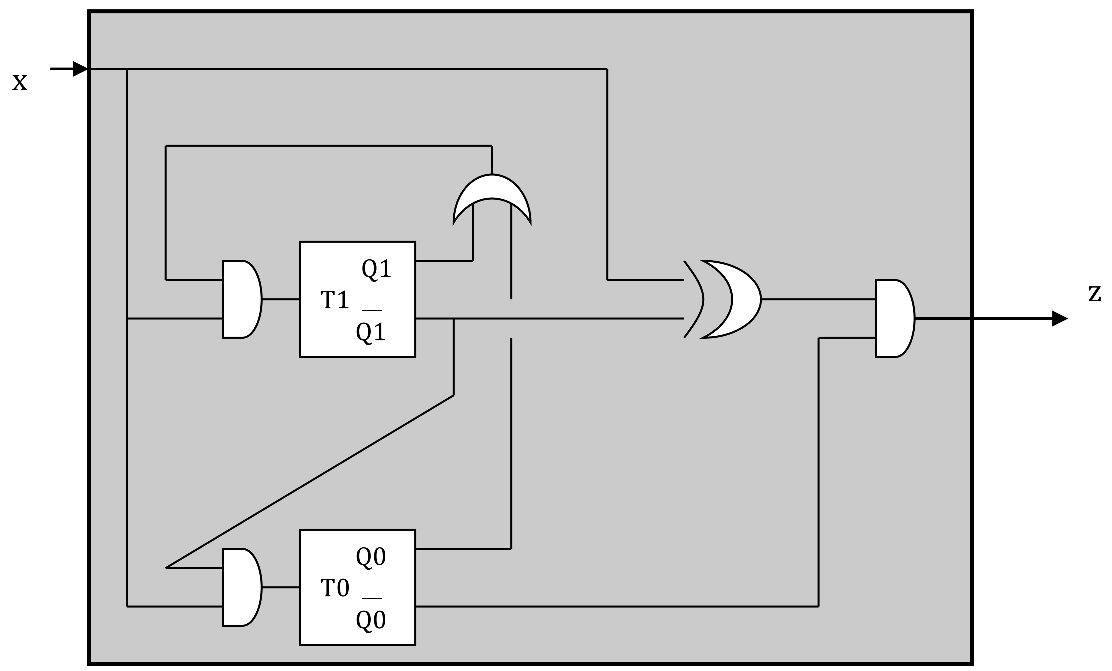
x	Q1 (t)	Q0	Q1	Q0 (t+1)	z (t)	T1	T0 (t)
0	0	0	0	0	1	0	0
0	0	1	0	1	0	0	0
0	1	0	1	0	0	0	0
0	1	1	-	-	-	-	-
1	0	0	0	1	0	0	1
1	0	1	1	0	0	1	1
1	1	0	0	0	1	1	0
1	1	1	-	-	-	-	-

By using KMs, we have that:

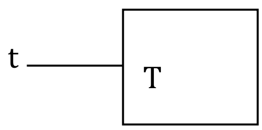
$z = \overline{x}\,\overline{Q1}\,\overline{Q0} + x\,Q1$ $\qquad T0 = x\,\overline{Q1}$ $\qquad T1 = x\,(Q0 + Q1)$

<u>REMARK:</u> we can obtain the more compact expression $\overline{Q0}\,(x \oplus \overline{Q1})$ *for z by considering both don't cares as 0!*

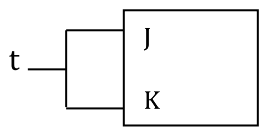
From the BEs, we draw the circuit:



To replace FFs of kind T with FFs of kind JK, it suffices to notice that



is equal to



Exercises without solutions

Ex.1. Design the sequential net that accepts sequences 100 and 111 by using flip-flops of kind D

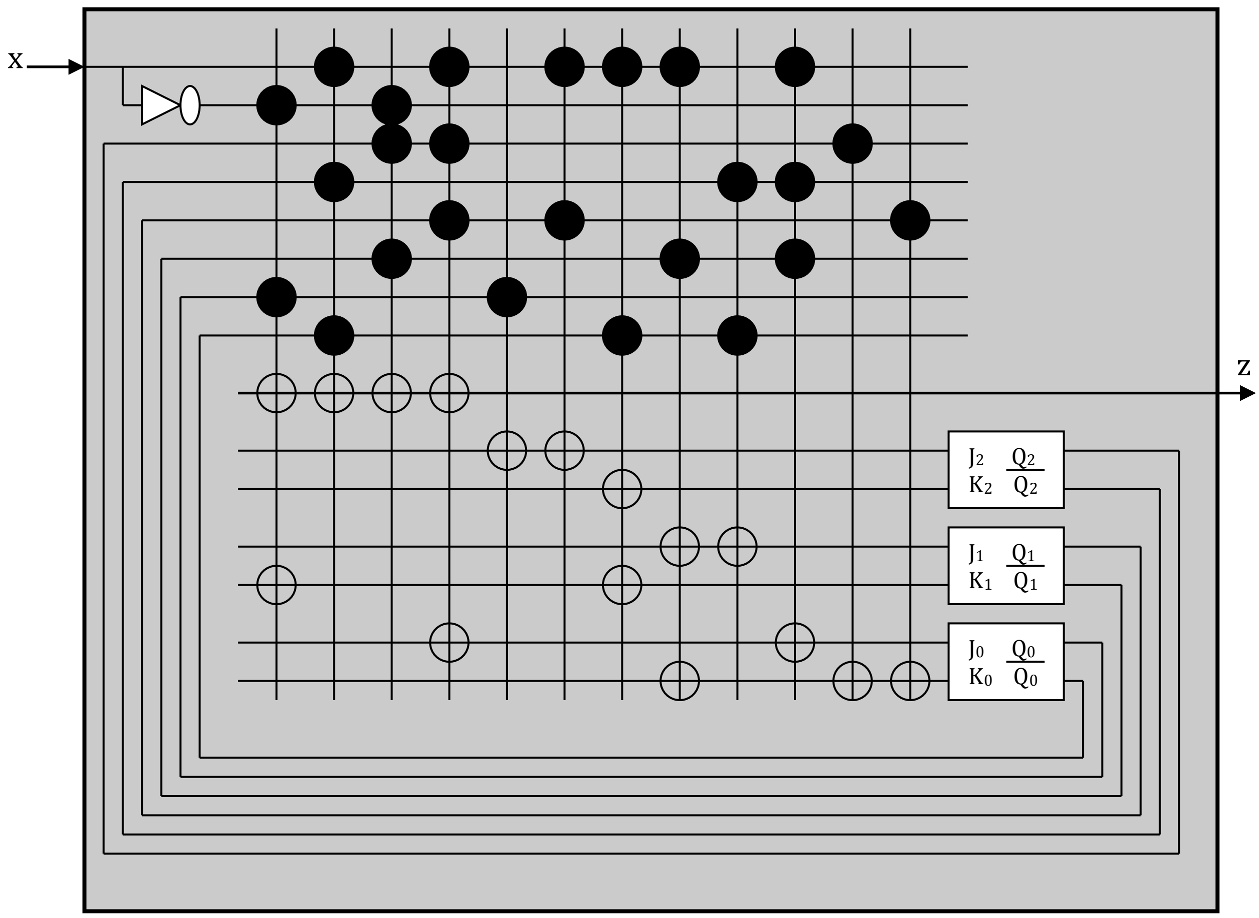
a) **with** superpositions,
b) **without** superpositions.

Ex. 2 (this exercise can be done AFTER studying counters). Design the control circuit of a semaphore. It should work as follows:

- the semaphore keeps the green light on for 10 seconds;
- then, the green light is switched off and the yellow light is on for 6 seconds;
- then, the yellow ligth is switched off and the redo ne is on for 16 seconds;
- finally, the red light is switched off and start again with the green.

Assume to have a clock with frequency at 1 Herz (i.e., with one impulse every second). Use FFs of kind JK.

Ex. 3. Minimize the following sequential net:



(Hint: minimize the automaton associted to the circuit – this can be obtained from an analysis procedure – and from the minimal automaton you can derive the minimal sequential net).