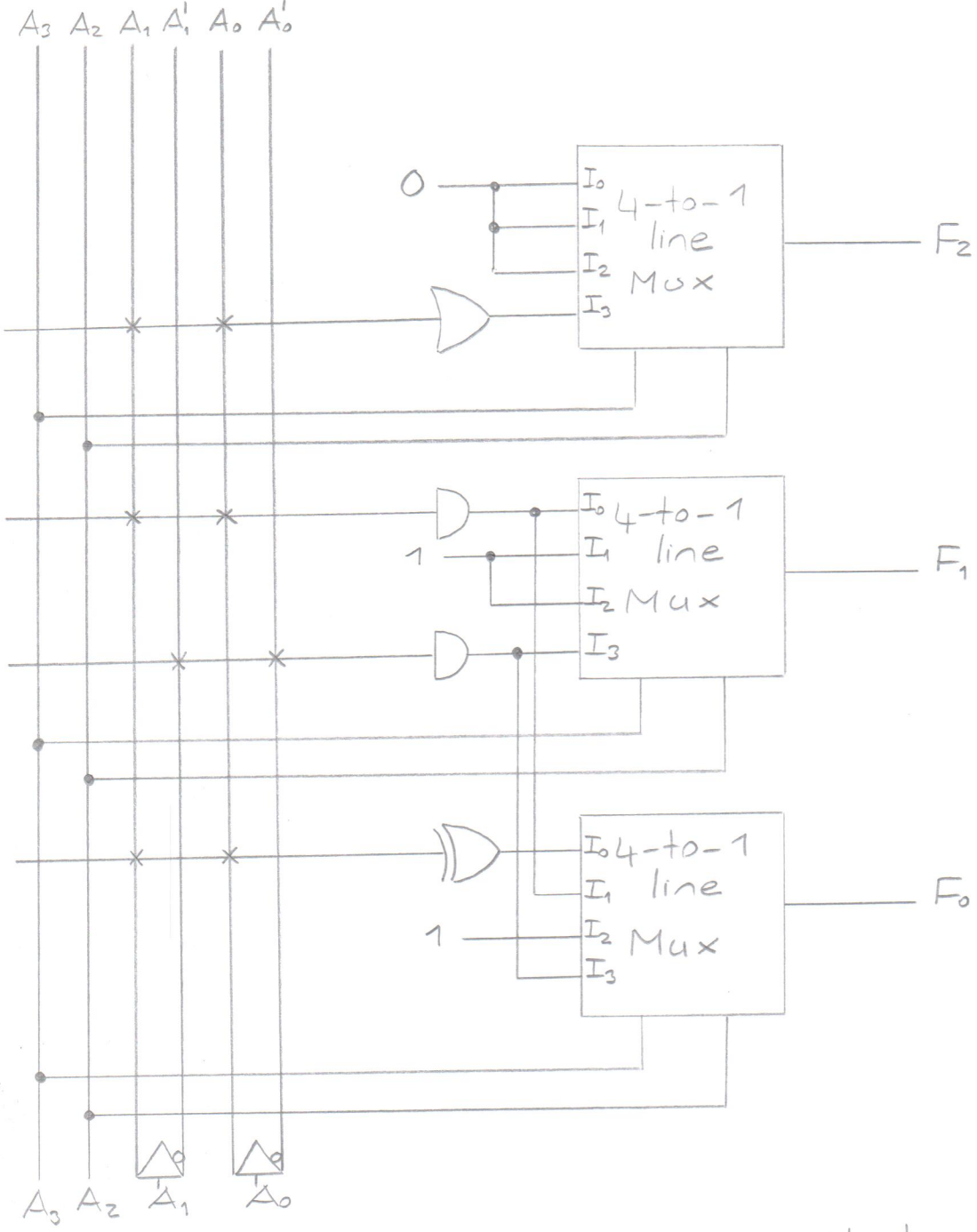
Miscellaneous Problems

Problem 1) Design a combinational circuit that accepts 4-bit binary number and generates a 3-bit binary number output that approximates the square root of the number. For example, if the square root is 3.5 or larger, it gives a result of 4. If the square root is less than 3.5 and greater than or equal to 2.5, give a result of 3. Implement the circuit using three 4-to-1 line multiplexers and external logic gates if needed.

Solution. We tabulate the truth table as follows

A_3	A_2	A_1	A_0	F_2	F_1	F_0		
0	0	0	0	0	0	0	$F_2 = 0$	I_0
0	0	0	1	0	0	1	$F_1 = A_1 A_0$	
0	0	1	0	0	0	1	$F_0 = A_1' A_0 + A_0' A_1$	
0	0	1	1	0	1	0		
0	1	0	0	0	1	0	$F_2 = 0$	I_1
0	1	0	1	0	1	0	$F_1 = 1$	
0	1	1	0	0	1	0	$F_0 = A_1 A_0$	
0	1	1	1	0	1	1		
1	0	0	0	0	1	1	$F_2 = 0$	I_2
1	0	0	1	0	1	1	$F_1 = 1$	
1	0	1	0	0	1	1	$F_0 = 1$	
1	0	1	1	0	1	1		
1	1	0	0	0	1	1	$F_2 = A_1 + A_0$	I_3
1	1	0	1	1	0	0	$F_1 = A_1' A_0'$	
1	1	1	0	1	0	0	$F_0 = A_1' A_0'$	
1	1	1	1	1	0	0		

- we choose A_3, A_2 as the select (control) inputs of the 4-to-1-line multiplexers
- then we can represent F_2, F_1, F_0 in terms of A_1 and A_0



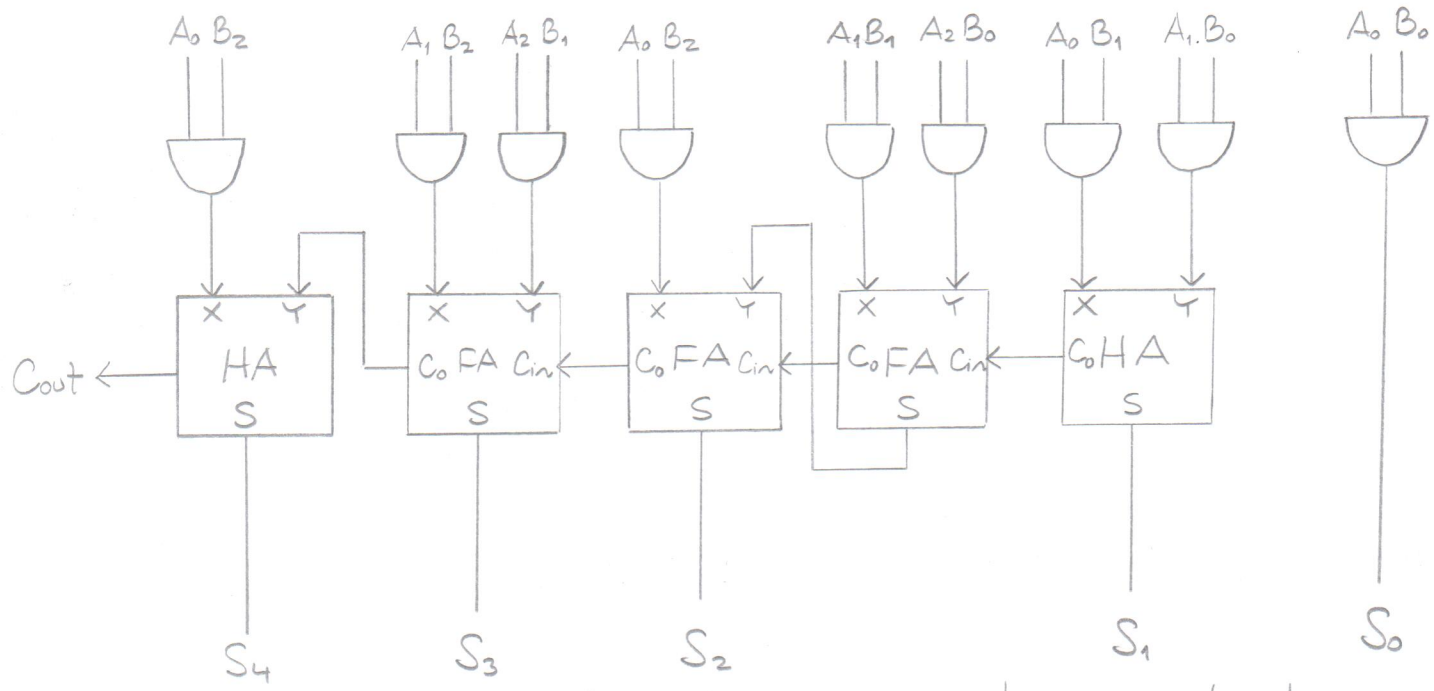
Problem 2) Design a binary multiplier circuit that can multiply two three-bit binary numbers $A_2A_1A_0$ and $B_2B_1B_0$. The only available logic circuit elements are AND gates, half adders and full adders.

a. Draw the overall logic circuit diagram.

b. If one needs to multiply two two-bit binary numbers, is it possible to employ the three-bit binary multiplier circuit? How? Explain.

Solution. We consider

$$
\begin{array}{cccc}
 & A_2 & A_1 & A_0 \\
 & B_2 & B_1 & B_0 \\
\times & \multicolumn{3}{c}{\hrulefill} \\
 & B_0A_2 & B_0A_1 & B_0A_0 \\
 & B_1A_2 & B_1A_1 & B_1A_0 \\
B_2A_0 & B_2A_1 & B_2A_0 & \\
+ & \multicolumn{3}{c}{\hrulefill} \\
S_4 & S_3 & S_2 & S_1 \quad S_0
\end{array}
$$



b. The answer is YES. We only need to set A_2 and B_2 as 0 or just connect to ground.

Problem 3) Consider a combinational circuit that squares a two-bit binary number.

a. Minimize the number of the product terms and specify the minimum possible size of the PLA that can implement the two-bit square generator.

b. Derive the PLA programming table for the two-bit square generator.

c. Draw the logic circuit diagram for the PLA that implements the two-bit square generator.

Solution. We have

c.

A_1	A_0	S_3	S_2	S_1	S_0
0	0	0	0	0	0
0	1	0	0	0	1
1	0	0	1	0	0
1	1	1	0	0	1

- we find that $S_1 = 0$

$$S_3 = A_1 A_0 \quad , \quad S_3' = A_1' + A_0'$$

$$S_2 = A_1 A_0' \quad , \quad S_2' = A_1' + A_0$$

$$S_0 = A_1' A_0 + A_1 A_0$$

$$= (A_1' + A_1) A_0$$

$$= A_0 \quad , \quad S_0' = A_0'$$

- we do NOT need to implement S_0 because it is already equal to A_0.

Hence;

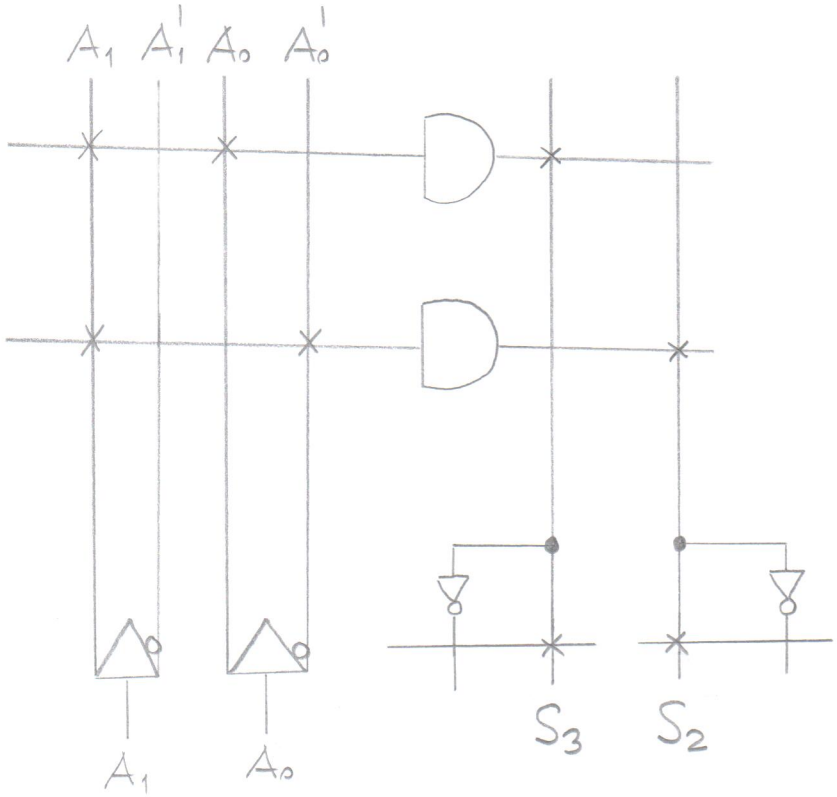
- we only need 2 product terms, that is

PLA size should be $\underset{\text{(inputs)}}{2} \times \underset{\text{(products)}}{2} \times \underset{\text{(outputs)}}{2}$

b.

Product terms	A_1	A_0	S_3 (T)	S_2 (T)
$A_1 A_0$	1	1	1	–
$A_1 A_0'$	1	0	–	1

C. The PLA circuit diagram is drawn as follows:



Problem 4) Given that

$$A \cdot B = 0 \quad \text{and} \quad A + B = 1$$

prove using algebraic manipulation for a general Boolean algebra with axioms/theorems that

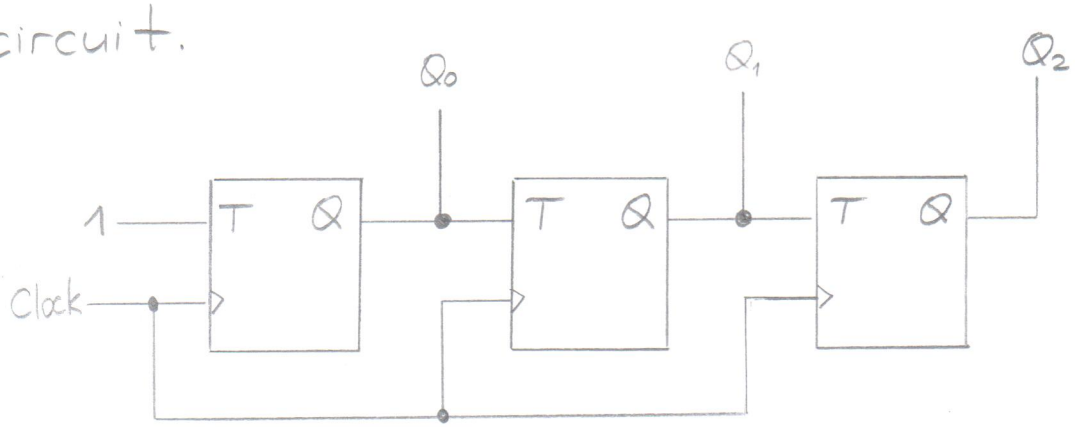
$$AC + A'B + BC = B + C$$

(Note: A, B, C can have values other than 0 and 1)

Solution. We have

$$\begin{aligned} AC + A'B + BC &= AC + A'B + BC + 0 \\ &= AC + A'B + BC + A \cdot B \\ &= (A+B)C + (A'+A)B \\ &= 1 \cdot C + 1 \cdot B \\ &= C + B = B + C \end{aligned}$$

Problem 5) Consider the following sequential circuit.



a. Identify the task that the cascaded T flip-flops implements.

b. Design a sequential circuit that repectedly counts $0 \to 7 \to 2 \to 4 \to 0$ and $1 \to 6 \to 3 \to 5 \to 1$ using JK type flip-flops.

Solution.

a. We know that for a T type flip-flop

$$Q(t+1) = Q(t) \oplus T$$

Hence;

$$Q_0(t+1) = Q_0(t) \oplus 1$$
$$= Q_0(t) \cdot 1' + Q_0'(t) \cdot 1$$
$$= Q_0'(t)$$

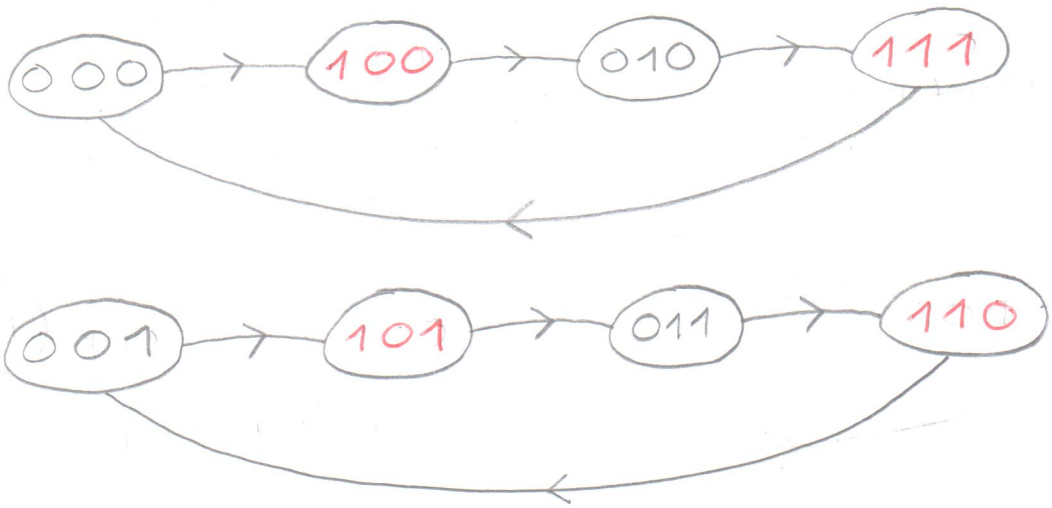
$$Q_1(t+1) = Q_1(t) \oplus Q_0(t)$$

$$Q_2(t+1) = Q_2(t) \oplus Q_1(t)$$

- we now derive the state table as follows

$Q_0(t)$	$Q_1(t)$	$Q_2(t)$	$Q_0(t+1)$	$Q_1(t+1)$	$Q_2(t+1)$
0	0	0	1	0	0
0	0	1	1	0	1
0	1	0	1	1	1
0	1	1	1	1	0
1	0	0	0	1	0
1	0	1	0	1	1
1	1	0	0	0	1
1	1	1	0	0	0

and the state diagram is obtained as



- the circuit justifies to be a counter with two different counting sequences:

$0 \rightarrow 4 \rightarrow 2 \rightarrow 7$

and

$1 \rightarrow 5 \rightarrow 3 \rightarrow 6$

MP 1.7

b. We tabulate the extended state table employing three JK type flip-flops A, B, C

y_A	y_B	y_C	Y_A	Y_B	Y_C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	1	1	1	1	d	1	d	1	d
0	0	1	1	1	0	1	d	1	d	d	1
0	1	0	1	0	0	1	d	d	1	0	d
0	1	1	1	0	1	1	d	d	1	d	0
1	0	0	0	0	0	d	1	0	d	0	d
1	0	1	0	0	1	d	1	0	d	d	0
1	1	0	0	1	1	d	1	d	0	1	d
1	1	1	0	1	0	d	1	d	0	d	1

yY	00	01	11	10
	0,d	1,d	d,0	d,1

$J_A = K_A = 1$

J_B:

y_A \ y_C	0	1
0	1	1
1	.	.

$J_B = y_A'$

K_B:

y_A \ y_C	0	1
0	1	1
1	.	.

$K_B = y_A'$

J_C:

y_A \ y_B	0	1
0	1	.
1	.	1

$J_C = y_A' y_B' + y_A y_B$

$= (y_A \oplus y_B)'$

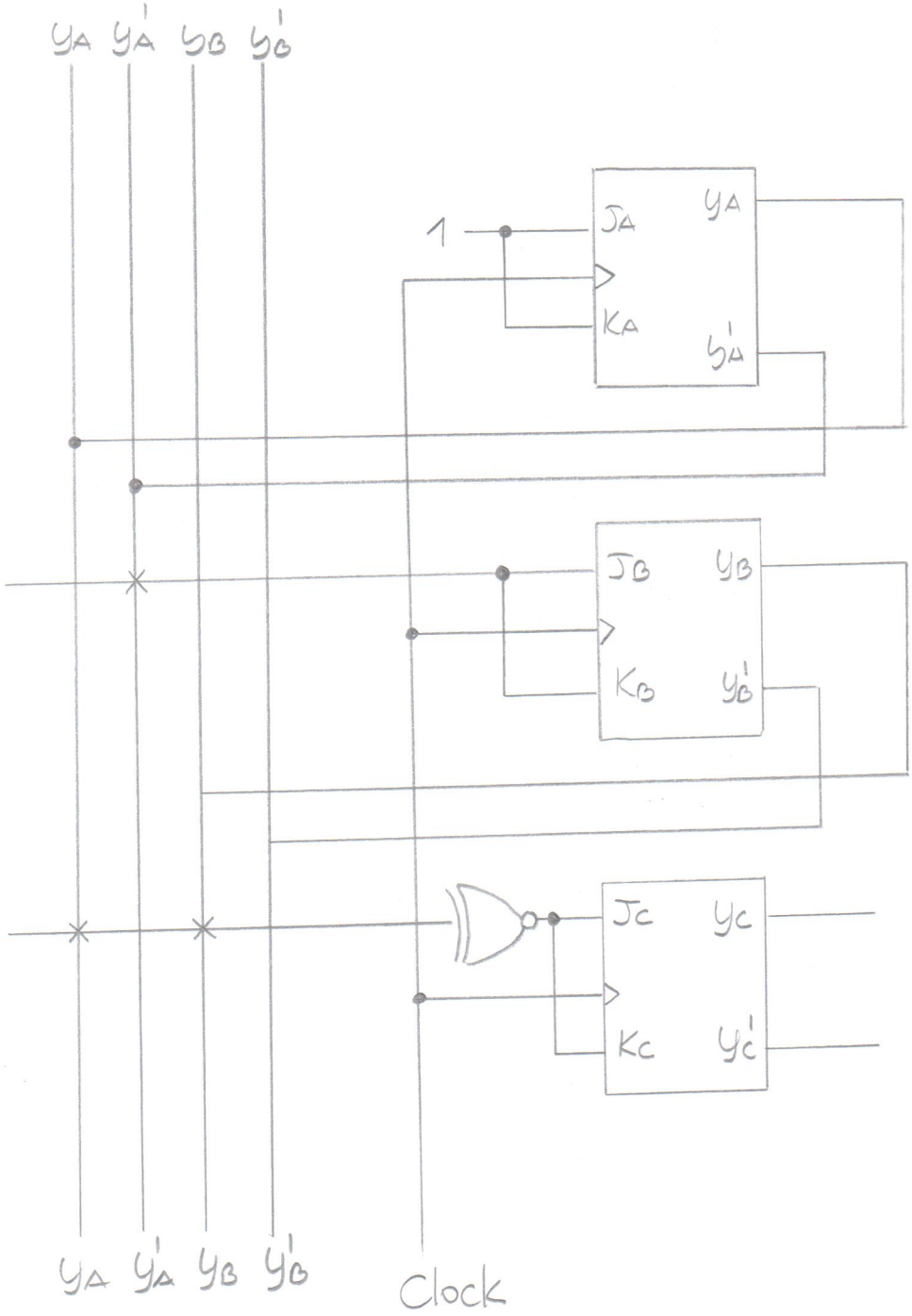
K_C:

y_A \ y_B	0	1
0	1	.
1	.	1

$K_C = y_A' y_B' + y_A y_B$

$= (y_A + y_B)'$

MP 1.8



MP 1.9