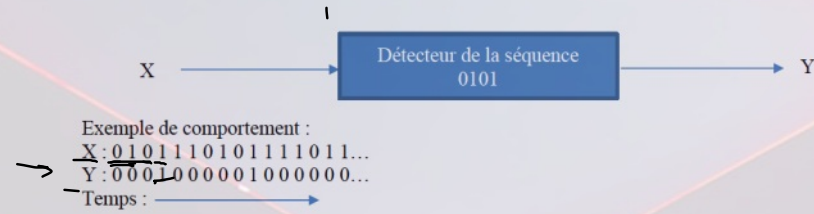


Concevoir un circuit séquentiel qui prend en entrée, séquentiellement, un à un, des bits et met 1 en sortie lorsque les quatre derniers bits lus sont 0101 et qui met 0 sinon. Il faut bien sûr mettre 0 en sortie quand moins de quatre bits ont été lus.



b) Écrire une table de transition d'états ;

États	X	
	X=0	X=1
A	B/0	A/0
B	B/0	C/0
C	D/0	A/0
D	B/0	C/1

c) Écrire une table de vérité en utilisant un encodage binaire en ordre pour les états ;

États	Encodage	<u>État présent</u>			<u>État futur</u>		
		S_1	S_0	X	S_1'	S_0'	Z
A	00	0	0	0	0	1	0
B	01	0	0	1	0	0	0
C	10	0	1	0	1	0	0
D	11	1	0	0	0	1	0
	$S_1 S_0$	1	1	1	1	0	1

d) Simplifier les équations en utilisant la méthode des tables de Karnaugh;

S_1'

$S_1 \backslash S_0$	00	01	11	10
X	0	1	1	1
	0	1	1	1

$$S_1' = S_0 X + S_1 \bar{S}_0 \bar{X}$$

S_0'

$S_1 \backslash S_0$	00	01	11	10
X	0	1	1	1
	0	1	1	1

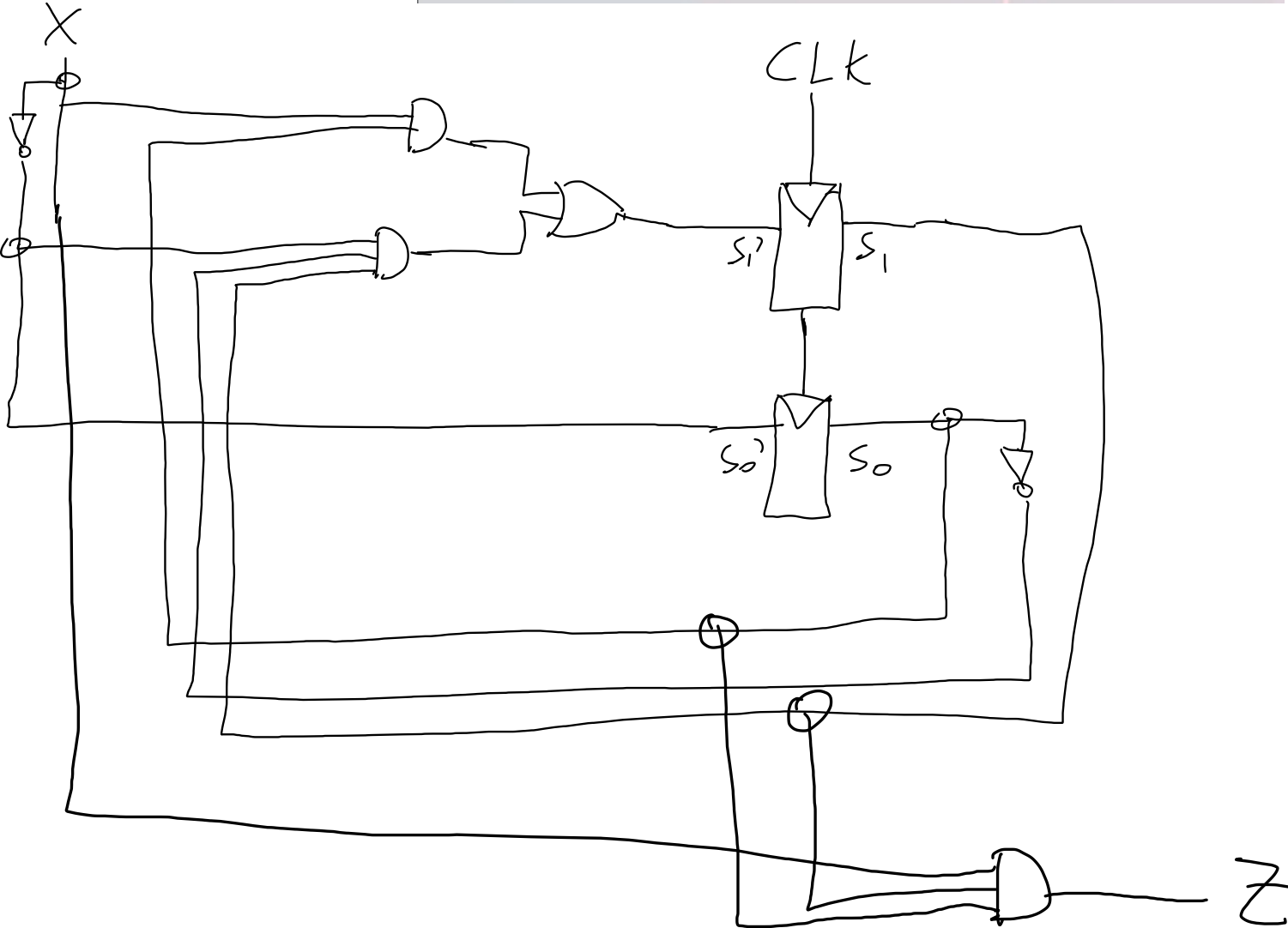
$$S_0' = \bar{X}$$

Z

$S_1 \backslash S_0$	00	01	11	10
X	0	1	1	1
	0	1	1	1

$$Z = S_1 S_0 X$$

e) Dessiner le logigramme du circuit en utilisant les portes OU, ET et NON.



A.32 Créer un circuit qui implémente le comportement suivant. Il faut trier les deux mots binaires entrés en série sur 2 entrées les bits les plus significatifs d'abord, en transférant la valeur du mot plus grand sur la sortie GE et celle du plus petit sur LT.

A _____ LT
B _____ GE

Exemple de comportement :

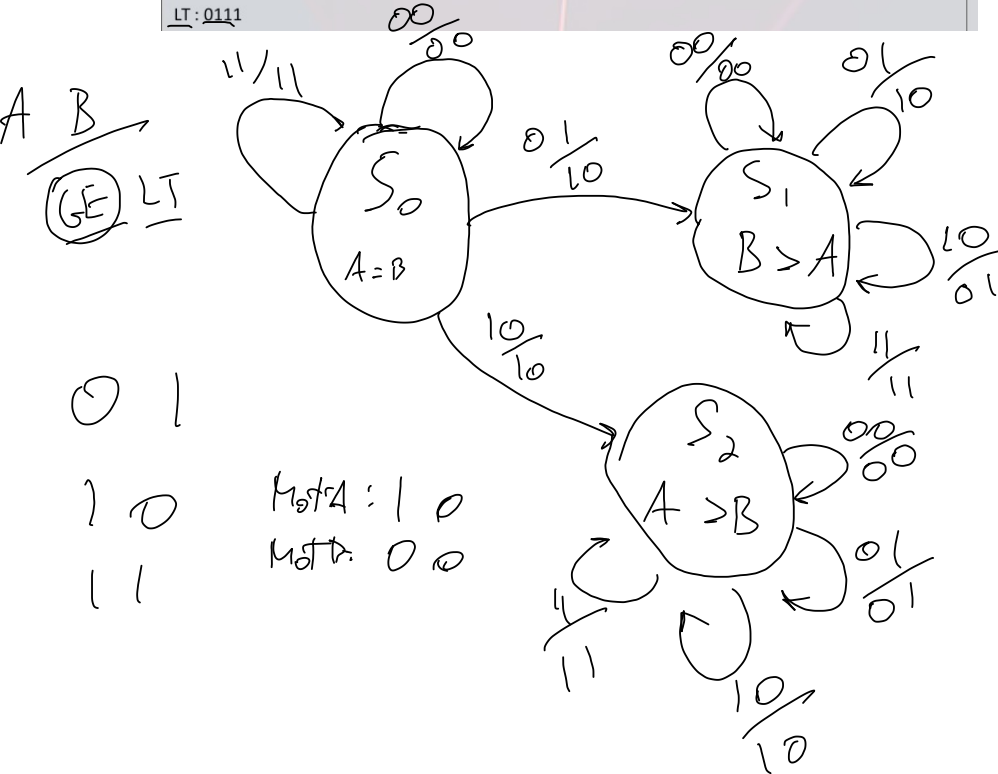
Mot A : 1001

Mots B : 0111

Comme $A > B$ (MSB de $A >$ MSB de B), donc le mot A doit apparaitre sur la sortie GE et mot B sur la sortie LT.

GE : 1001

LT : 0111



↓
Mot A : 0 1
Mot B : 1 1
GE : 1 1
LT : 0 1

		A B			
Effects		00	01	10	11
A=B	S ₀	$\frac{S_0}{00}$	$\frac{S_1}{10}$	$\frac{S_2}{10}$	$\frac{S_0}{11}$
B→A	S ₁	$\frac{S_1}{00}$	$\frac{S_1}{10}$	$\frac{S_1}{01}$	$\frac{S_1}{11}$
A→B	S ₂	$\frac{S_2}{00}$	$\frac{S_2}{01}$	$\frac{S_2}{10}$	$\frac{S_2}{11}$

Effects	Encoding
S ₀	00
S ₁	01
S ₂	10
B ₁ B ₀	
↑ ↑	

B ₁	B ₀	A B	B ₁ '	B ₀ '	GE	LT
0	0	00	0	0	0	0
0	0	01	0	1	1	0
0	0	10	1	0	1	0
0	0	11	0	0	1	1
0	1	00	0	1	0	0
0	1	01	0	1	1	0
0	1	10	0	1	0	1
0	1	11	0	1	1	1
1	0	00	1	0	0	0
1	0	01	1	0	0	1
1	0	10	1	0	1	0
1	0	11	1	0	1	1
1	1	00	0	0	0	0
1	1	01	0	0	1	0
1	1	10	0	0	1	1
1	1	11	0	0	1	1

Truth table for B_1' :

$B_1 B_0$	00	01	11	10
$A B$	00	01	11	10
00				
01				
11				
10				

Red circles highlight the following cells: $(A, B) = (0, 1)$ for $B_1 B_0 = 01$ and $(A, B) = (1, 0)$ for $B_1 B_0 = 10$.

$$B_1' = B_1 + \overline{B_0} A \overline{B}$$

Truth table for B_0' :

$B_1 B_0$	00	01	11	10
$A B$	00	01	11	10
00				
01				
11				
10				

Red circles highlight the following cells: $(A, B) = (0, 1)$ for $B_1 B_0 = 01$ and $(A, B) = (1, 0)$ for $B_1 B_0 = 10$.

$$B_0' = B_0 + \overline{A} B \overline{B_1}$$

Truth table for GE :

$B_1 B_0$	00	01	11	10
$A B$	00	01	11	10
00				
01				
11				
10				

Red circles highlight the following cells: $(A, B) = (0, 1)$ for $B_1 B_0 = 01$ and $(A, B) = (1, 0)$ for $B_1 B_0 = 10$.

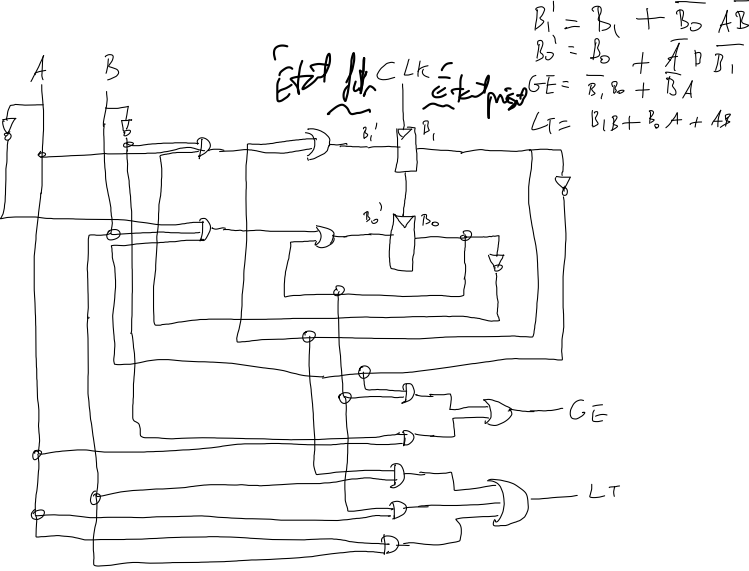
$$GE = \overline{B_1} B + \overline{B_0} A$$

Truth table for LT :

$B_1 B_0$	00	01	11	10
$A B$	00	01	11	10
00				
01				
11				
10				

Red circles highlight the following cells: $(A, B) = (0, 1)$ for $B_1 B_0 = 01$ and $(A, B) = (1, 0)$ for $B_1 B_0 = 10$.

$$LT = B_1 B + B_0 A + A B$$



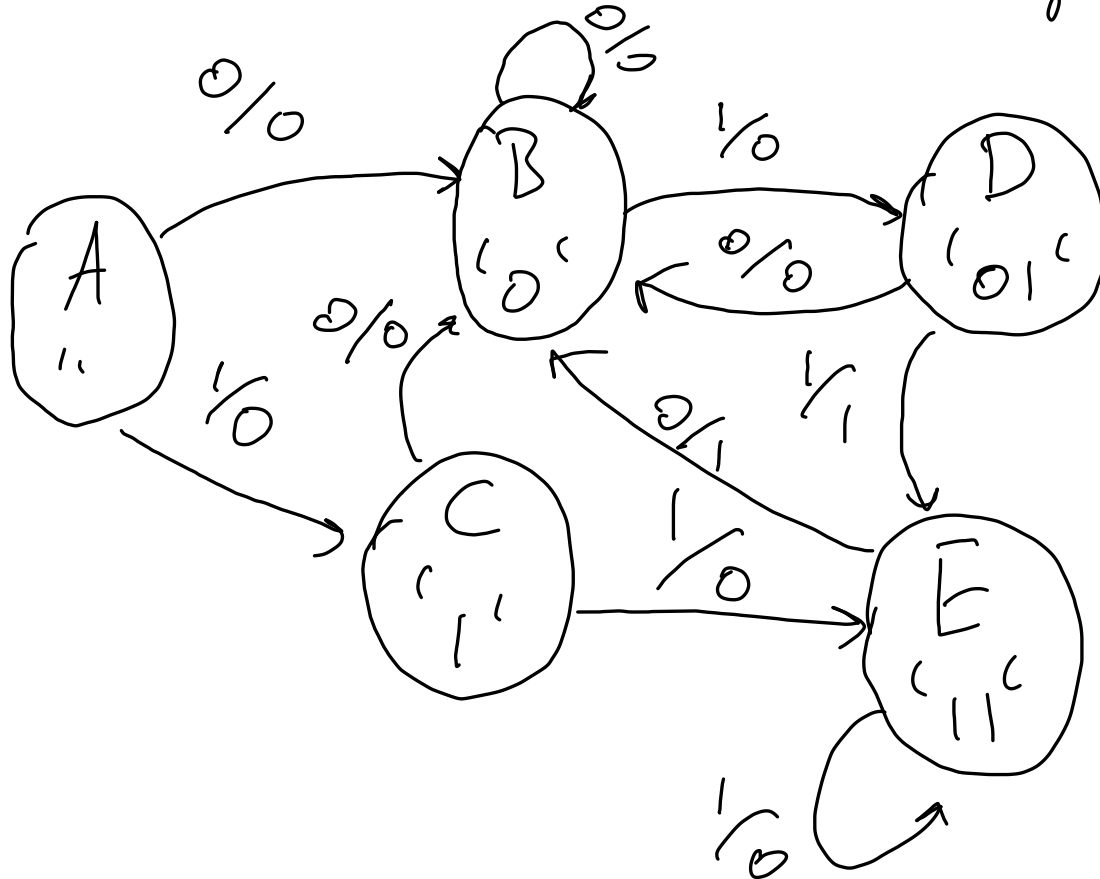
$$B_0 + \overline{A} B \overline{B_1}$$

$$GE = \overline{B_1} B + \overline{B_0} A$$

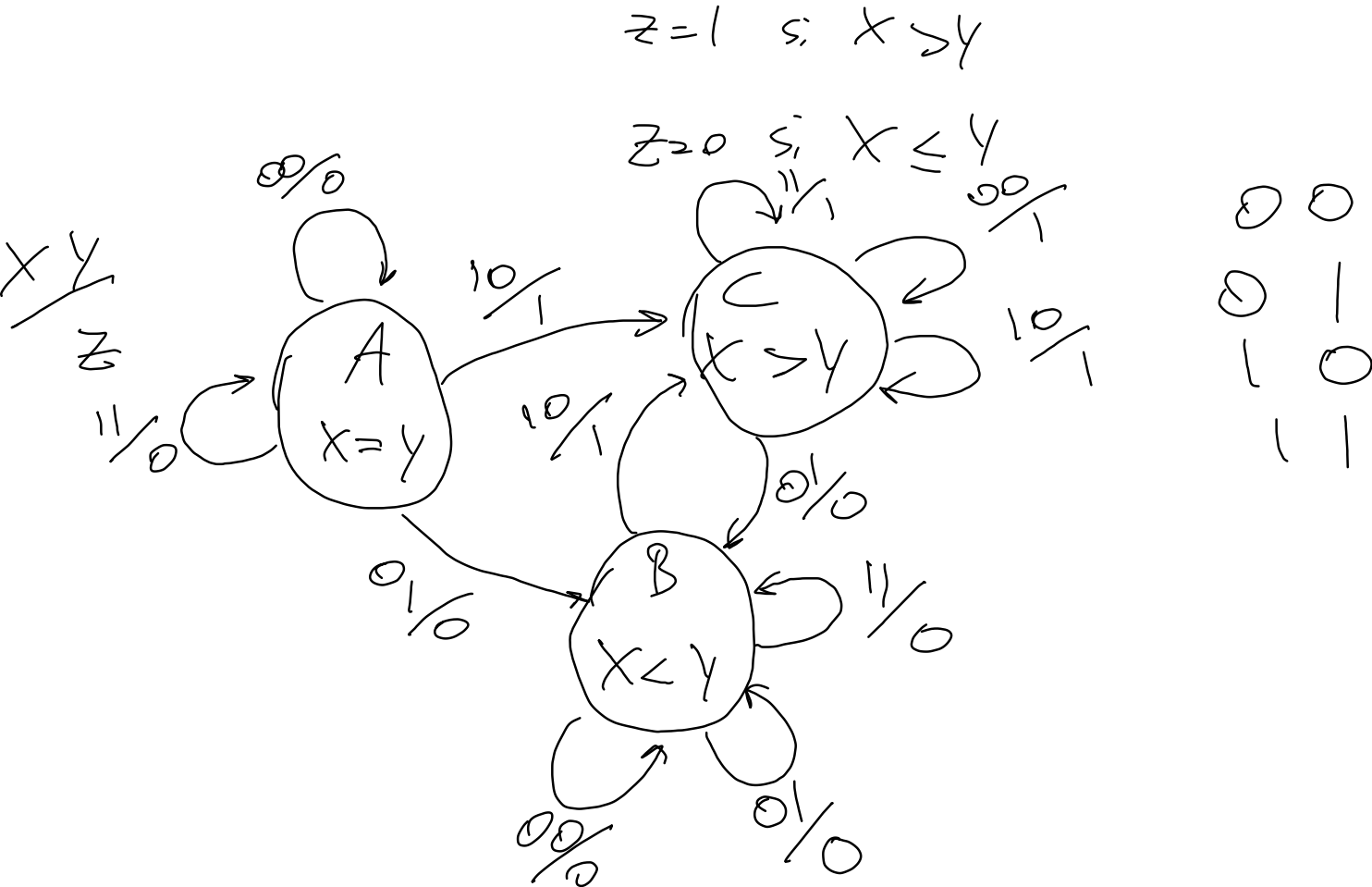
A.34 Concevoir un FSM de détecteur de séquence. La machine doit reconnaître les deux motifs : 011 ou 110.

FSM: Finite State Machine \Rightarrow Diagramme d'états

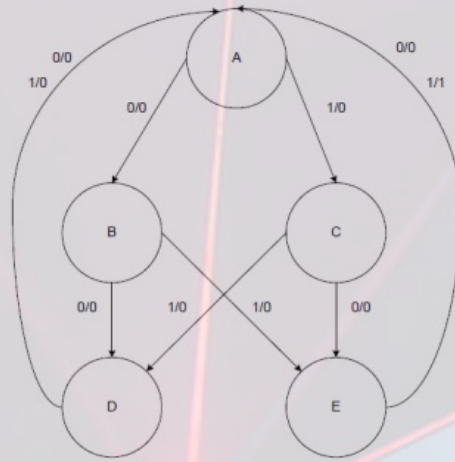
011
110



A.35 Concevoir un FSM qui prend 2 mots binaires entrés en série, les bits les moins significatifs d'abord, et produit une sortie Z. $Z = 1$ lorsque $X > Y$ et $Z = 0$ dans le cas contraire. On présume qu'en démarrage $X = Y$ et $Z = 0$.



A.38 Considérer une machine à états finis suivante :

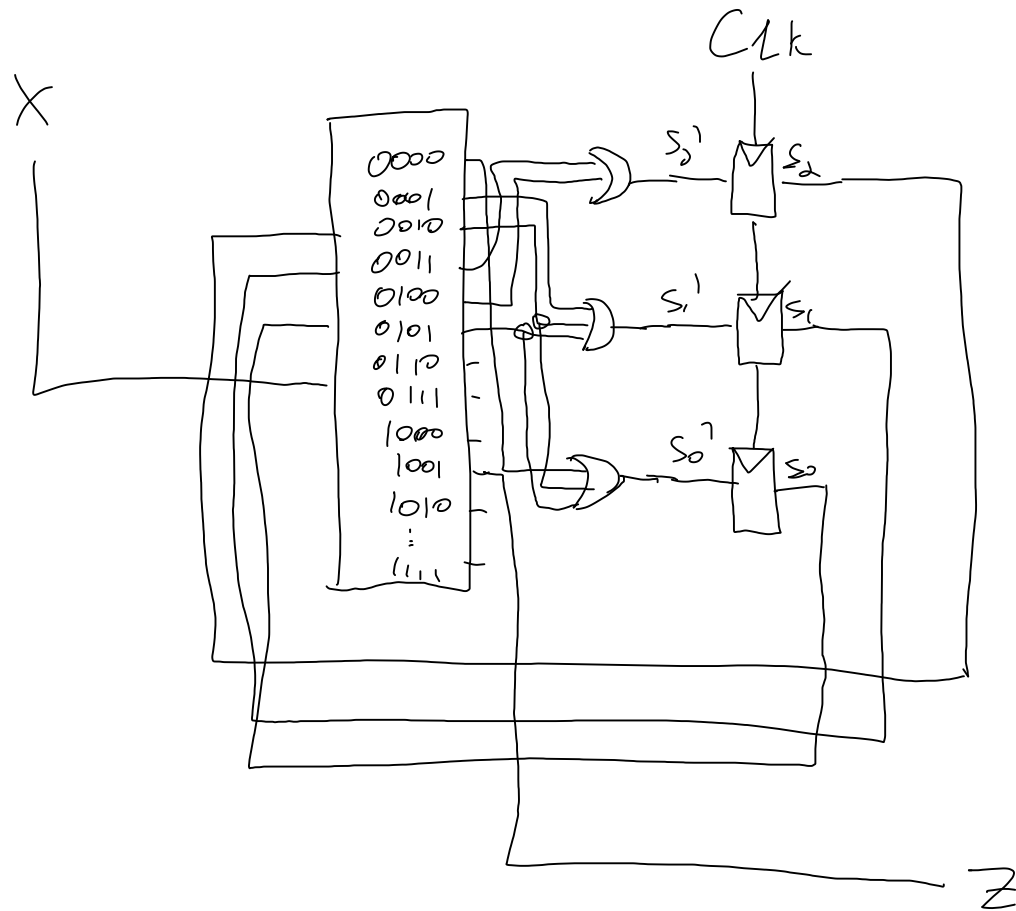


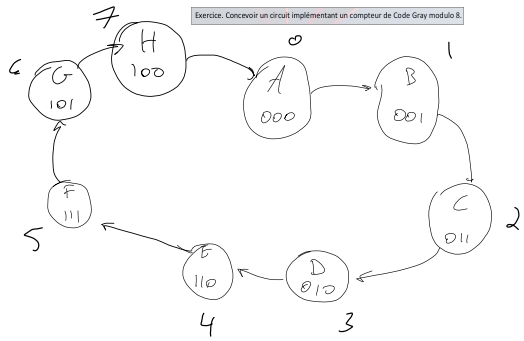
a) Créer une table de transitions d'états

États	x	
	x=0	x=1
A	B/0	C/0
B	D/0	E/0
C	E/0	D/0
D	A/0	A/0
E	A/0	A/1

b) Concevoir un circuit en utilisant un décodeur et les portes OU logiques.

États	Encodage	S_2	S_1	S_0	X	S_2'	S_1'	S_0'	Z
A	000	0	0	0	0	0	0	1	0
B	001	0	0	0	1	0	1	0	0
C	010	0	0	1	0	0	1	1	0
D	011	0	0	1	1	1	0	0	0
E	100	0	1	0	0	1	0	0	0
	$S_2 S_1 S_0$	0	1	0	1	0	1	1	0
		0	1	1	0	0	0	0	0
		0	1	1	1	0	0	0	0
		1	0	0	0	0	0	0	0
		1	0	0	1	0	0	0	1
					
					

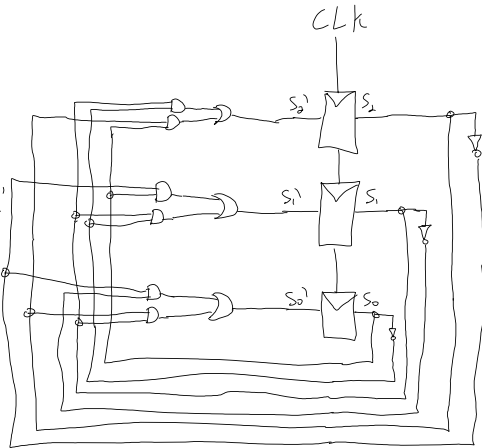




États | Encodage par Code

A	000
B	001
C	011
D	010
E	110
F	111
G	101
H	100
$s_2 s_1 s_0$	

s_2	s_1	s_0	s_2'	s_1'	s_0'
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	1	0	0
1	1	0	1	1	1
1	1	1	0	1	1



s_2'	$s_2 s_1$	s_0	00	01	11	10
s_0	0	0	0	0	1	1
	1	0	0	1	1	1

$$s_2' = s_1 \bar{s}_0 + s_2 s_0$$

s_0'	$s_2 s_1$	s_0	00	01	11	10
s_0	0	0	1	1	1	1
	1	0	1	1	1	1

$$s_0' = \bar{s}_2 \bar{s}_1 + s_2 s_1$$

s_1'	$s_2 s_1$	s_0	00	01	11	10
s_0	0	0	0	1	1	1
	1	0	0	1	1	1

$$s_1' = \bar{s}_2 s_0 + s_1 \bar{s}_0$$