

# ELEN0040 – REPETITION 6

---

*Synthèse de circuits  
séquentiels*

# Synthèse

Point de départ : système à modéliser

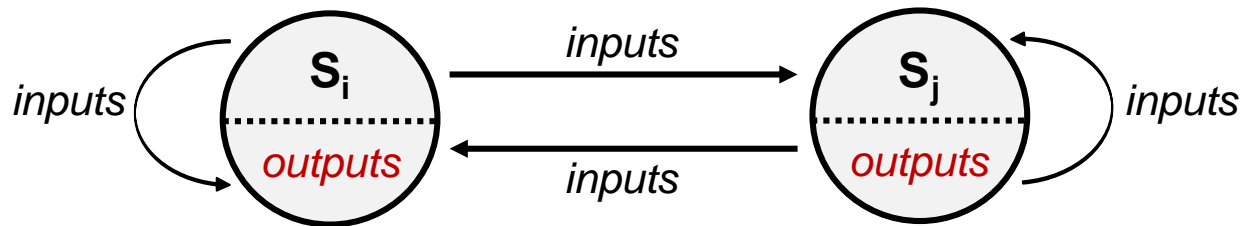
- ➡ 1. Définir et optimiser les **entrées/sorties**
- ➡ 2. Etablir le **diagramme d'états** optimal  
( < lois de fonctionnement )
- ➡ 3. Etablir la **table d'états** ( < diagramme d'états )
- ➡ 4. Exprimer et simplifier les **fonctions logiques**
- ➡ ( ( 5. Implémenter / Représenter le **schéma logique** ) )

# Diagramme d'états - Rappels

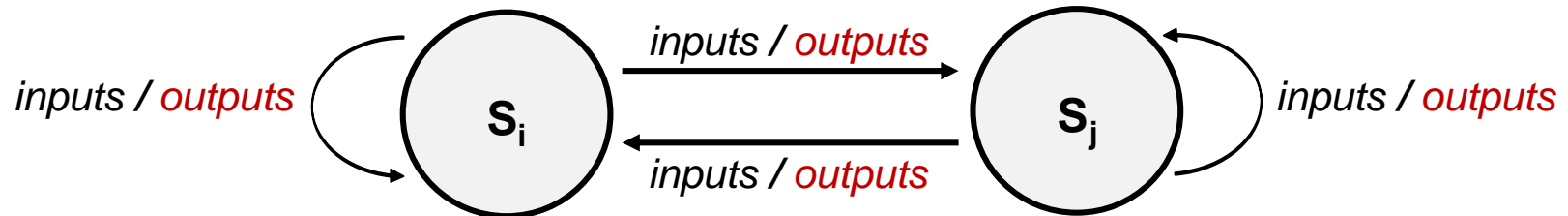
Chaque **état** est numéroté par une combinaison de valeurs des variables d'état (1 variable /FF)

$$n \text{ FF} \rightarrow 2^n \text{ états}$$

Modèle de Moore:  $\text{outputs} = F(\text{state})$



Modèle de Mealy:  $\text{outputs} = F(\text{state}, \text{inputs})$



## Ex 16

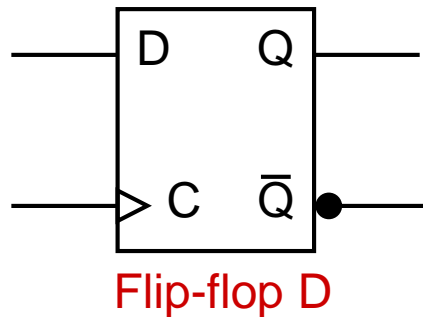
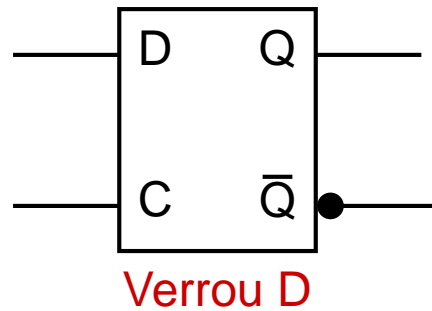
Réaliser le design d'un circuit qui a le comportement suivant :

- Si l'*entrée* = 1 : le circuit reste dans le même état ;
- Si l'*entrée* = 0 : il génère la séquence :  
**00 – 11 – 01 – 10 – 00 – ...**

Utiliser des flip-flops D, la sortie est égale à l'état présent.

# Verrous et flip-flops - Rappels

- Verrou : actif sur un **niveau** d'horloge
- Flip-flop : actif sur une **transition** d'horloge



D(t)	Q(t+1)
0	0
1	1

## Ex 16

Réaliser le design d'un circuit qui a le comportement suivant :

- Si l'*entrée* = 1 : le circuit reste dans le même état ;
- Si l'*entrée* = 0 : il génère la séquence :  
**00 – 11 – 01 – 10 – 00 – ...**

Utiliser des flip-flops D, la sortie est égale à l'état présent.

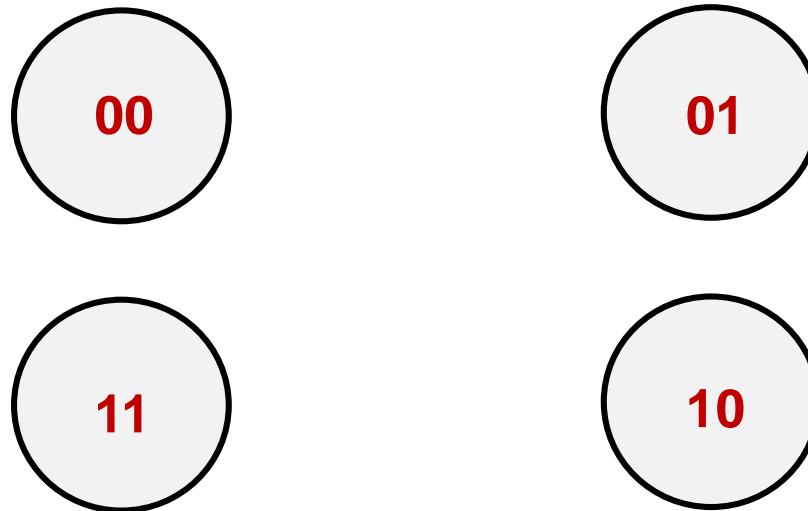
## Ex 16

1a. Entrée(s): **1 bit X**

1b. Sortie(s): **= état présent** (→ Modèle de Moore)

2. Lois de fonctionnement → Diagramme d'états :

- Si l'*entrée* = 1 : le circuit reste dans le même état ;
- Si l'*entrée* = 0 : il génère la séquence : **00 – 11 – 01 – 10 – 00 – ...**  
→ 4 états (00, 01, 10, 11) → 2 variables d'état (2 FFD): **AB**



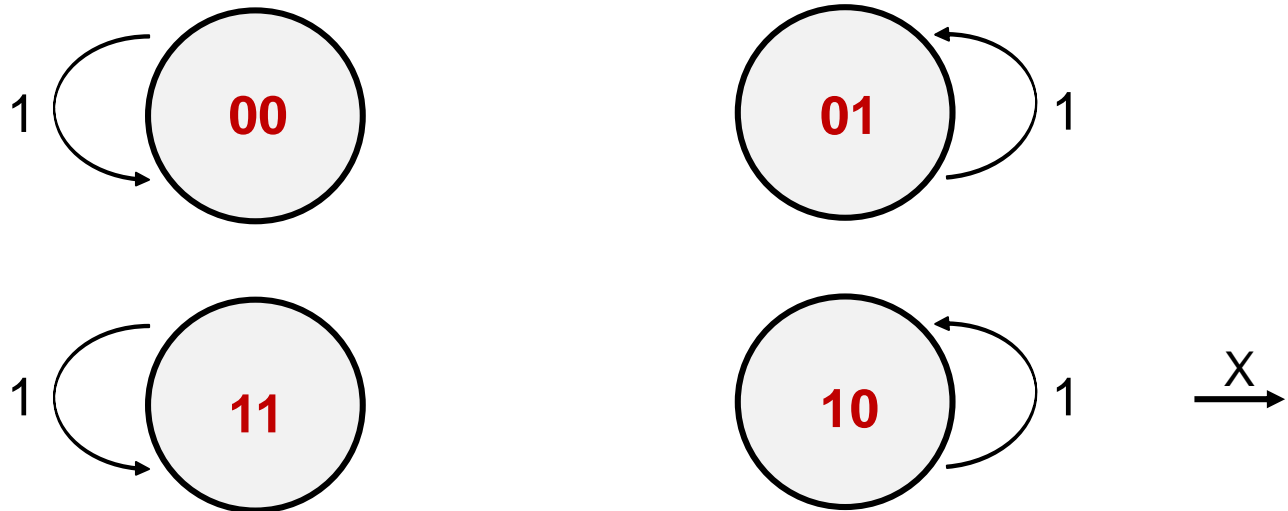
# Ex 16

1a. Entrée(s): 1 bit X

1b. Sortie(s): = état présent (→ Modèle de Moore)

2. Lois de fonctionnement → Diagramme d'états :

- Si l'entrée = 1 : le circuit reste dans le même état ;
- Si l'entrée = 0 : il génère la séquence : **00 – 11 – 01 – 10 – 00 – ...**  
→ 4 états (00, 01, 10, 11) → 2 variables d'état (2 FFD): **AB**





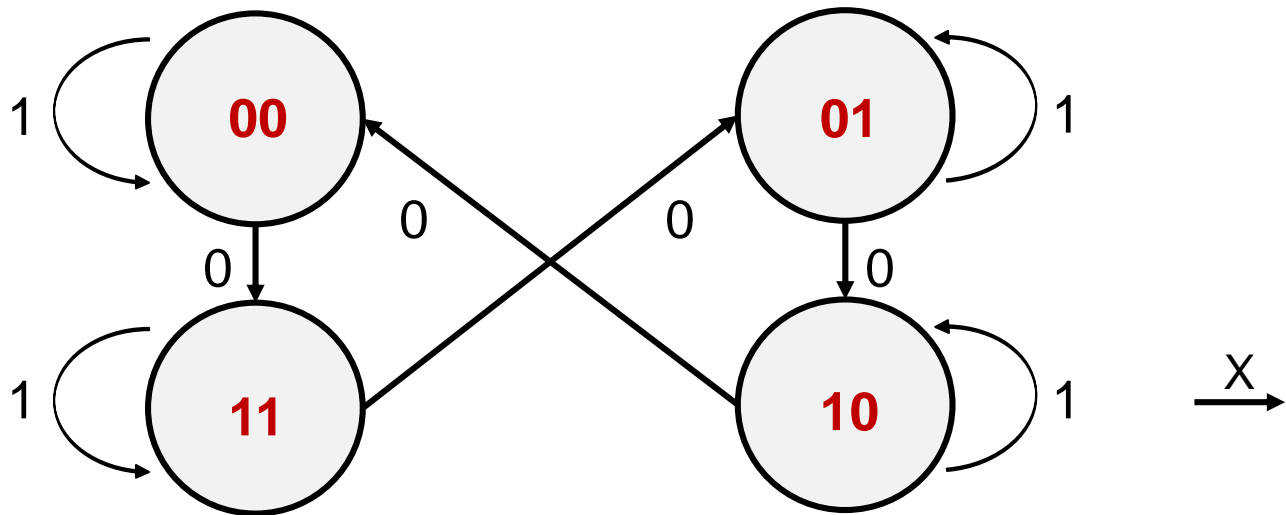
## Ex 16

1a. Entrée(s): 1 bit **X**

1b. Sortie(s): = état présent (→ Modèle de Moore)

2. Lois de fonctionnement → Diagramme d'états :

- Si l'entrée = 1 : le circuit reste dans le même état ;
- Si l'entrée = 0 : il génère la séquence : **00 – 11 – 01 – 10 – 00 – ...**  
→ 4 états (00, 01, 10, 11) → 2 variables d'état (2 FFD): **AB**



# Ex 16

$X = 1 \rightarrow$  Mode mémoire

3. Table d'états:

<b>A(t)</b>	<b>B(t)</b>	<b>X</b>	<b>A(t+1)</b>	<b>B(t+1)</b>	<b>S<sub>1</sub></b>	<b>S<sub>2</sub></b>
0	0	0				
0	0	1	0	0		
0	1	0				
0	1	1	0	1		
1	0	0				
1	0	1	1	0		
1	1	0				
1	1	1	1	1		

## Ex 16

### 3. Table d'états:

$X = 0 \rightarrow$  Séquence: 00 – 11 – 01 – 10 – 00 ...

<b>A(t)</b>	<b>B(t)</b>	<b>X</b>	<b>A(t+1)</b>	<b>B(t+1)</b>	<b>S<sub>1</sub></b>	<b>S<sub>2</sub></b>
0	0	0	1	1		
0	0	1	0	0		
0	1	0	1	0		
0	1	1	0	1		
1	0	0	0	0		
1	0	1	1	0		
1	1	0	0	1		
1	1	1	1	1		

# Ex 16

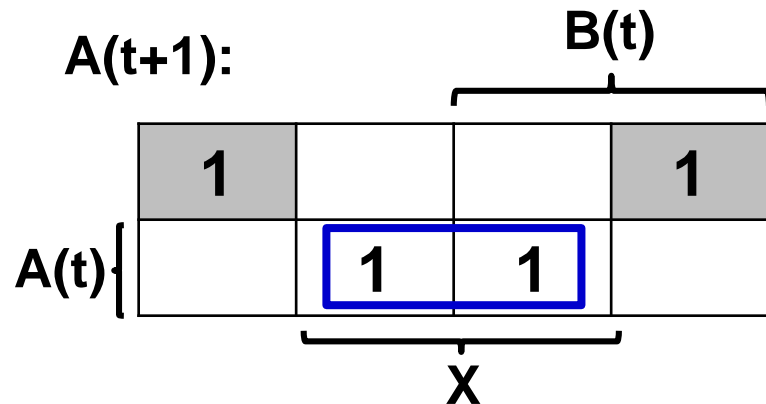
Sorties = état présent =  $A(t) B(t)$

3. Table d'états:

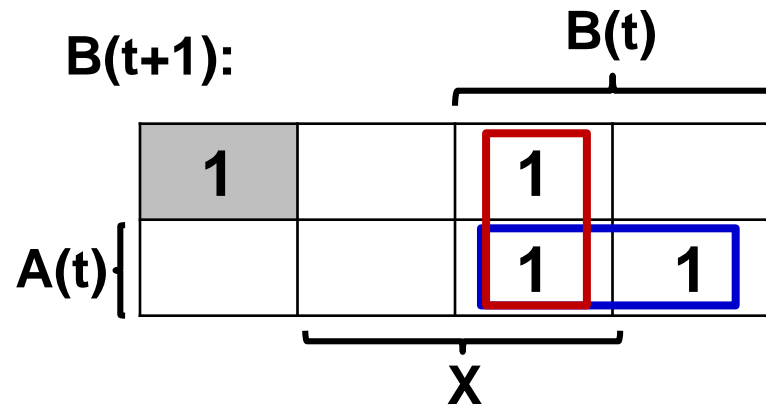
$A(t)$	$B(t)$	X	$A(t+1)$	$B(t+1)$	$S_1$	$S_2$
0	0	0	1	1	0	0
0	0	1	0	0	0	0
0	1	0	1	0	0	1
0	1	1	0	1	0	1
1	0	0	0	0	1	0
1	0	1	1	0	1	0
1	1	0	0	1	1	1
1	1	1	1	1	1	1

## Ex 16

### 4. Fonctions logiques des flip-flops (et sorties):



$$A(t+1) = X.A(t) + \bar{X}.\bar{A}(t) \\ \equiv D_A(t)$$



$$B(t+1) = X.B(t) + A(t).B(t) \\ + \bar{X}.\bar{A}(t).\bar{B}(t) \\ \equiv D_B(t)$$

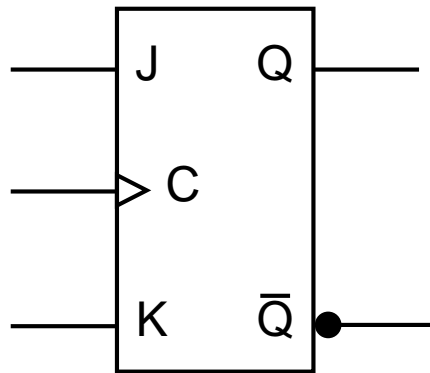
## Ex 18

Réaliser le design d'un circuit qui possède 2 entrées et qui a le comportement suivant :

- Si  $E = 0$  : le circuit reste dans le même état sans regarder la valeur de  $X$  ;
- Si  $E = 1$  et  $X = 1$  : le circuit passe dans les états suivants **00 – 01 – 10 – 11 – 00 ...**
- Si  $E = 1$  et  $X = 0$  : le circuit passe dans les états **00 – 11 – 10 – 01 – 00...**

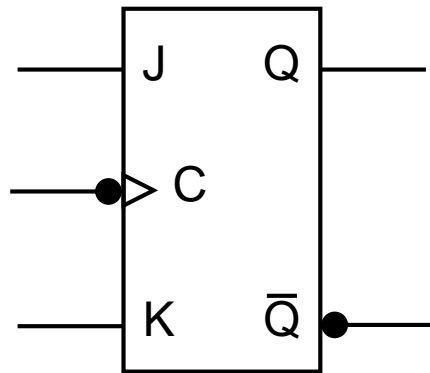
Utiliser des flip-flops JK, associer les sorties aux états.

# Flip-flops JK - Rappels



FF JK

Actif sur flanc  
montant



FF JK

Actif sur flanc  
descendant

J(t)	K(t)	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	$\overline{Q(t)}$

$$Q(t+1) = J.\overline{Q(t)} + \overline{K}.Q(t)$$

## Ex 18

Réaliser le design d'un circuit qui possède 2 entrées et qui a le comportement suivant :

- Si  $E = 0$  : le circuit reste dans le même état sans regarder la valeur de  $X$  ;
- Si  $E = 1$  et  $X = 1$  : le circuit passe dans les états suivants **00 – 01 – 10 – 11 – 00 ...**
- Si  $E = 1$  et  $X = 0$  : le circuit passe dans les états **00 – 11 – 10 – 01 – 00...**

Utiliser des flip-flops JK, associer les sorties aux états.



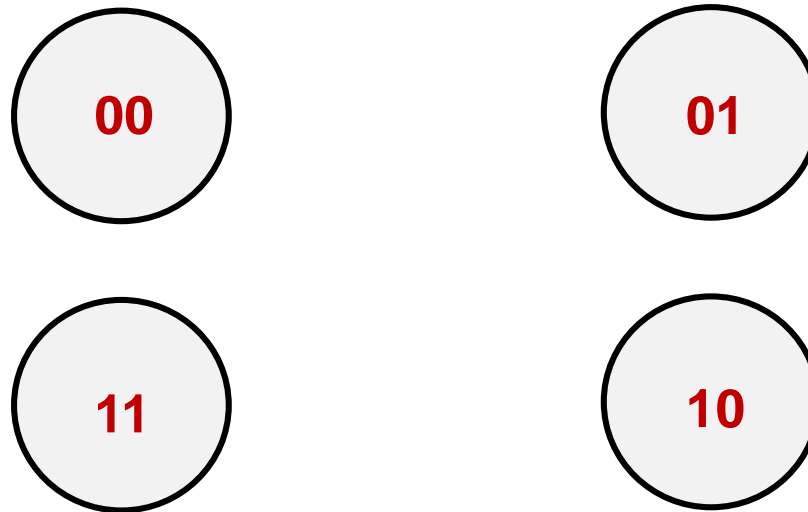
# Ex 18

1a. Entrée(s): 2 bits **E** et **X**

1b. Sortie(s): = état présent ( $\rightarrow$  Modèle de Moore)

2. Lois de fonctionnement  $\rightarrow$  Diagramme d'états :

- Si **E** = 0 : le circuit reste dans le même état ( $\forall X$ );
- Si **E** = 1 et **X** = 1 : il génère la séquence : **00** – **01** – **10** – **11** – **00** – ...
- Si **E** = 1 et **X** = 0 : il génère la séquence : **00** – **11** – **10** – **01** – **00** – ...  
 $\rightarrow$  4 états (00, 01, 10, 11)  $\rightarrow$  2 variables d'état (2 FFJK): **AB**



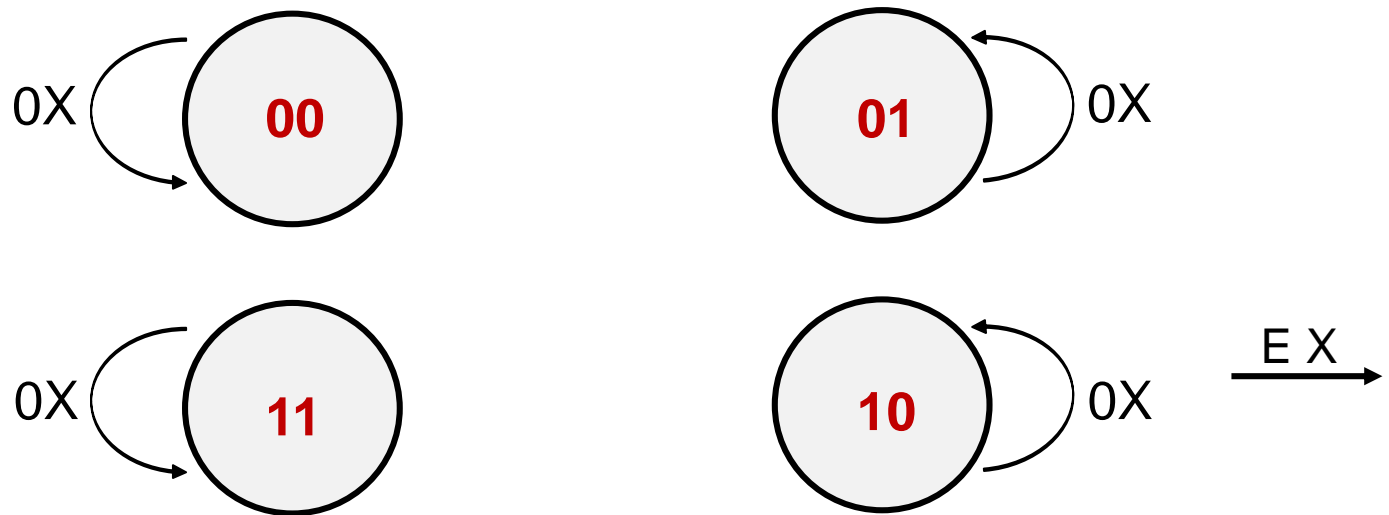
# Ex 18

1a. Entrée(s): **2 bits E et X**

1b. Sortie(s): **= état présent** ( $\rightarrow$  Modèle de Moore)

2. Lois de fonctionnement  $\rightarrow$  Diagramme d'états :

- Si  **$E = 0$**  : le circuit reste dans le même état ( $\forall X$ );
  - Si  **$E = 1$**  et  **$X = 1$**  : il génère la séquence : **00 – 01 – 10 – 11 – 00 – ...**
  - Si  **$E = 1$**  et  **$X = 0$**  : il génère la séquence : **00 – 11 – 10 – 01 – 00 – ...**
- $\rightarrow$  4 états (00, 01, 10, 11)  $\rightarrow$  2 variables d'état (2 FFJK): **AB**



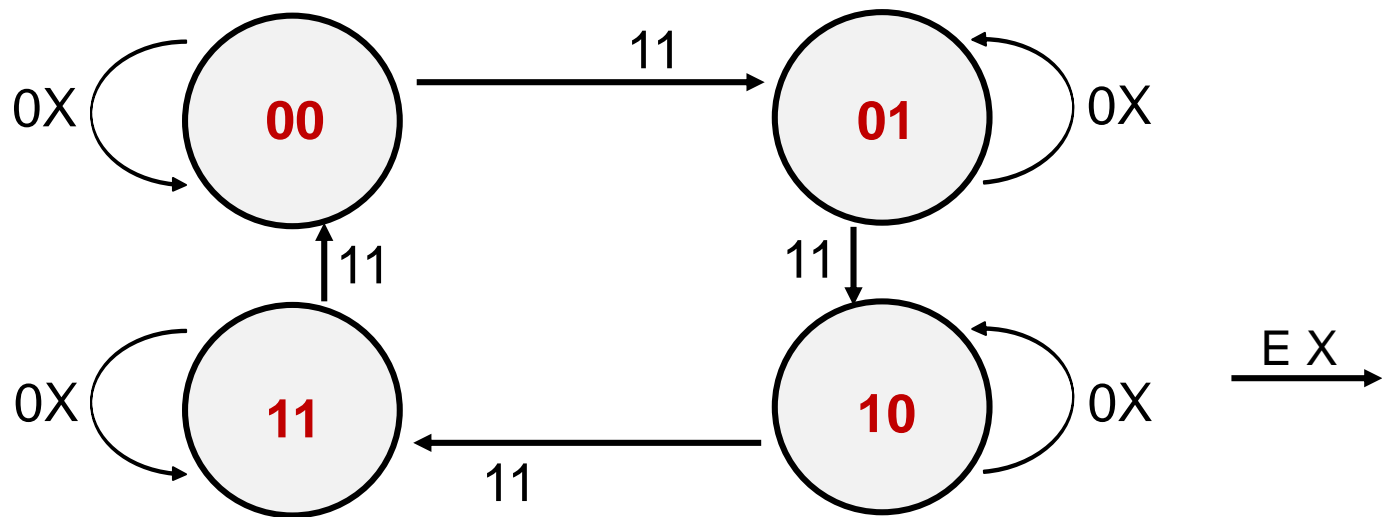
# Ex 18

1a. Entrée(s): **2 bits E et X**

1b. Sortie(s): **= état présent** ( $\rightarrow$  Modèle de Moore)

2. Lois de fonctionnement  $\rightarrow$  Diagramme d'états :

- Si  **$E = 0$**  : le circuit reste dans le même état ( $\forall X$ );
  - Si  **$E = 1$  et  $X = 1$**  : il génère la séquence : **00 – 01 – 10 – 11 – 00 – ...**
  - Si  **$E = 1$  et  $X = 0$**  : il génère la séquence : **00 – 11 – 10 – 01 – 00 – ...**
- $\rightarrow$  4 états (00, 01, 10, 11)  $\rightarrow$  2 variables d'état (2 FFJK): **AB**



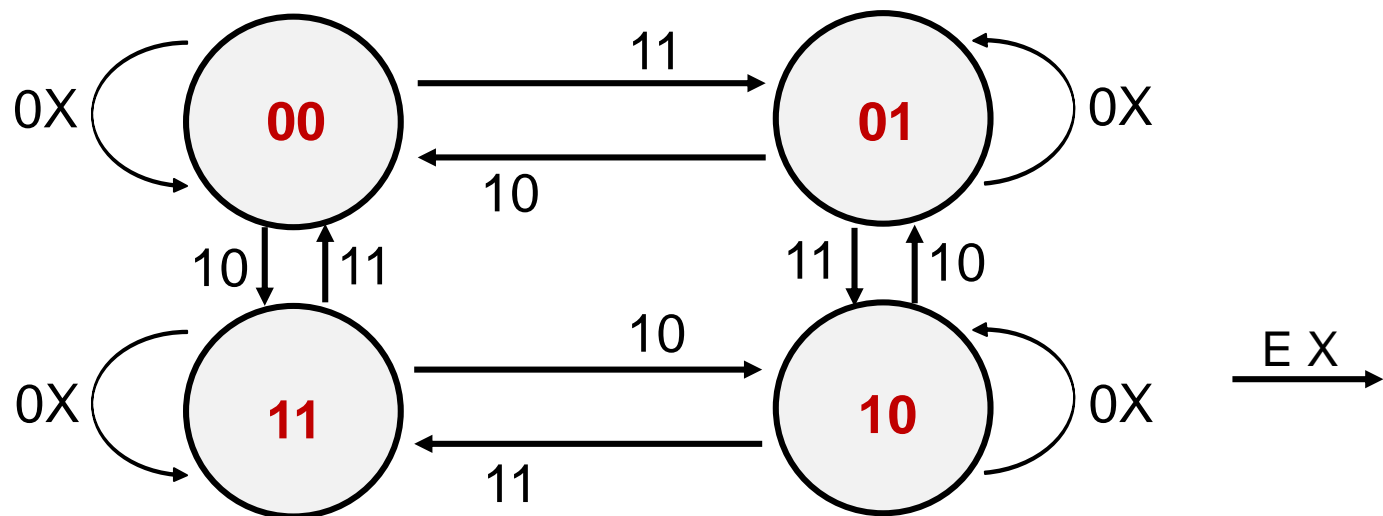
# Ex 18

1a. Entrée(s): **2 bits E et X**

1b. Sortie(s): **= état présent** ( $\rightarrow$  Modèle de Moore)

2. Lois de fonctionnement  $\rightarrow$  Diagramme d'états :

- Si **E = 0** : le circuit reste dans le même état ( $\forall X$ );
  - Si **E = 1** et **X = 1** : il génère la séquence : **00 – 01 – 10 – 11 – 00 – ...**
  - Si **E = 1** et **X = 0** : il génère la séquence : **00 – 11 – 10 – 01 – 00 – ...**
- $\rightarrow$  4 états (00, 01, 10, 11)  $\rightarrow$  2 variables d'état (2 FFJK): **AB**



A(t)	B(t)	E	X	A(t+1)	B(t+1)	S <sub>1</sub>	S <sub>2</sub>		
0	0	0	0	0	0				
0	0	0	1	0	0				
0	0	1	0						
0	0	1	1						
0	1	0	0	0	1				
0	1	0	1	0	1				
0	1	1	0						
0	1	1	1						
1	0	0	0	1	0				
1	0	0	1	1	0				
1	0	1	0						
1	0	1	1						
1	1	0	0	1	1				
1	1	0	1	1	1				
1	1	1	0						
1	1	1	1						

EX = 11 → Séquence: 00 – 01 – 10 – 11 – 00 ...

A(t)	B(t)	E	X	A(t+1)	B(t+1)	S <sub>1</sub>	S <sub>2</sub>		
0	0	0	0	0	0				
0	0	0	1	0	0				
0	0	1	0						
0	0	1	1	0	1				
0	1	0	0	0	1				
0	1	0	1	0	1				
0	1	1	0						
0	1	1	1	1	0				
1	0	0	0	1	0				
1	0	0	1	1	0				
1	0	1	0						
1	0	1	1	1	1				
1	1	0	0	1	1				
1	1	0	1	1	1				
1	1	1	0						
1	1	1	1	0	0				

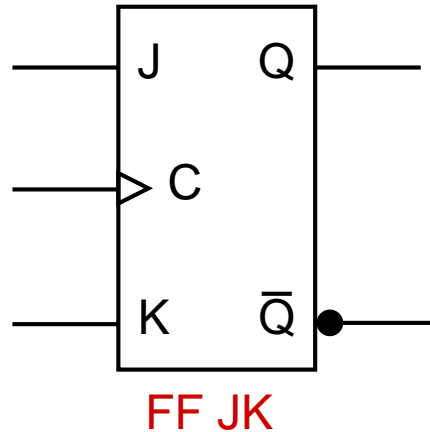
EX = 10 → Séquence: 00 – 11 – 10 – 01 – 00 ...

A(t)	B(t)	E	X	A(t+1)	B(t+1)	S <sub>1</sub>	S <sub>2</sub>		
0	0	0	0	0	0				
0	0	0	1	0	0				
0	0	1	0	1	1				
0	0	1	1	0	1				
0	1	0	0	0	1				
0	1	0	1	0	1				
0	1	1	0	0	0				
0	1	1	1	1	0				
1	0	0	0	1	0				
1	0	0	1	1	0				
1	0	1	0	0	1				
1	0	1	1	1	1				
1	1	0	0	1	1				
1	1	0	1	1	1				
1	1	1	0	1	0				
1	1	1	1	0	0				

A(t)	B(t)	E	X	A(t+1)	B(t+1)	S <sub>1</sub>	S <sub>2</sub>		
0	0	0	0	0	0	0	0		
0	0	0	1	0	0	0	0		
0	0	1	0	1	1	0	0		
0	0	1	1	0	1	0	0		
0	1	0	0	0	1	0	1		
0	1	0	1	0	1	0	1		
0	1	1	0	0	0	0	1		
0	1	1	1	1	0	0	1		
1	0	0	0	1	0	1	0		
1	0	0	1	1	0	1	0		
1	0	1	0	0	1	1	0		
1	0	1	1	1	1	1	0		
1	1	0	0	1	1	1	1		
1	1	0	1	1	1	1	1		
1	1	1	0	1	0	1	1		
1	1	1	1	0	0	1	1		



# Flip-flops JK - Rappels



J(t)	K(t)	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	$\overline{Q(t)}$

Q(t) $\rightarrow$ Q(t+1)	J(t)	K(t)
0 $\rightarrow$ 0	0	X
0 $\rightarrow$ 1	1	X
1 $\rightarrow$ 1	X	0
1 $\rightarrow$ 0	X	1

A(t)	B(t)	E	X	A(t+1)	B(t+1)	S <sub>1</sub>	S <sub>2</sub>	J <sub>A</sub>	K <sub>A</sub>
0	0	0	0	0	0	0	0	0	X
0	0	0	1	0	0	0	0	0	X
0	0	1	0	1	1	0	0	1	X
0	0	1	1	0	1	0	0	0	X
0	1	0	0	0	1	0	1	0	X
0	1	0	1	0	1	0	1	0	X
0	1	1	0	0	0	0	1	0	X
0	1	1	1	1	0	0	1	1	X
1	0	0	0	1	0	1	0	X	0
1	0	0	1	1	0	1	0	X	0
1	0	1	0	0	1	1	0	X	1
1	0	1	1	1	1	1	0	X	0
1	1	0	0	1	1	1	1	X	0
1	1	0	1	1	1	1	1	X	0
1	1	1	0	1	0	1	1	X	0
1	1	1	1	0	0	1	1	X	1

A(t)	B(t)	E	X	A(t+1)	B(t+1)	S <sub>1</sub>	S <sub>2</sub>	J <sub>B</sub>	K <sub>B</sub>
0	0	0	0	0	0	0	0	0	X
0	0	0	1	0	0	0	0	0	X
0	0	1	0	1	1	0	0	1	X
0	0	1	1	0	1	0	0	1	X
0	1	0	0	0	1	0	1	X	0
0	1	0	1	0	1	0	1	X	0
0	1	1	0	0	0	0	1	X	1
0	1	1	1	1	0	0	1	X	1
1	0	0	0	1	0	1	0	0	X
1	0	0	1	1	0	1	0	0	X
1	0	1	0	0	1	1	0	1	X
1	0	1	1	1	1	1	0	1	X
1	1	0	0	1	1	1	1	X	0
1	1	0	1	1	1	1	1	X	0
1	1	1	0	1	0	1	1	X	1
1	1	1	1	0	0	1	1	X	1

# Ex 18

## 4. Fonctions logiques des flip-flops (et sorties):

$J_A$ :

		$E$	
			1
		1	
$A(t)$ {	X	X	X
	X	X	X

$B(t)$

$X$

$$J_A = K_A = E.X.B(t) + E.\bar{X}.\bar{B}(t)$$

$K_A$ :

		$E$	
	X	X	X
	X	X	X
$A(t)$ {			

$B(t)$

$X$

# Ex 18

## 4. Fonctions logiques des flip-flops (et sorties):

$J_B$ :

		$E$	
		1	1
X	X	X	X
X	X	X	X
		1	1

$A(t)$  {  $B(t)$

$X$

$$J_B = K_B = E$$

$K_B$ :

		$E$	
X	X	X	X
		1	1
		1	1
X	X	X	X

$A(t)$  {  $B(t)$

$X$

## Ex 20

Des nombres compris entre 0 et 7 sont transmis sous forme binaire sur une ligne série.

L'envoi de chaque nombre commence toujours par le bit de poids fort.

On demande de réaliser un circuit dont la sortie  $Z = 1$  lors de la détection du bit de poids faible d'un 0 ou d'un 7.

Une nouvelle détection commencera après le passage de 3 bits.

Déterminer la table d'états.

## Ex 20

$$\begin{aligned}X(t) &= b_2 \\X(t+1) &= b_1 \\X(t+2) &= b_0\end{aligned}$$



Entr e:

**X** = bit sur la ligne (un   la fois)

Sortie:

**Z** = 1 si "000" / "111"

On demande de r aliser un circuit dont la sortie  $Z = 1$  lors de la d tection du bit de poids faible d'un 0 ou d'un 7.

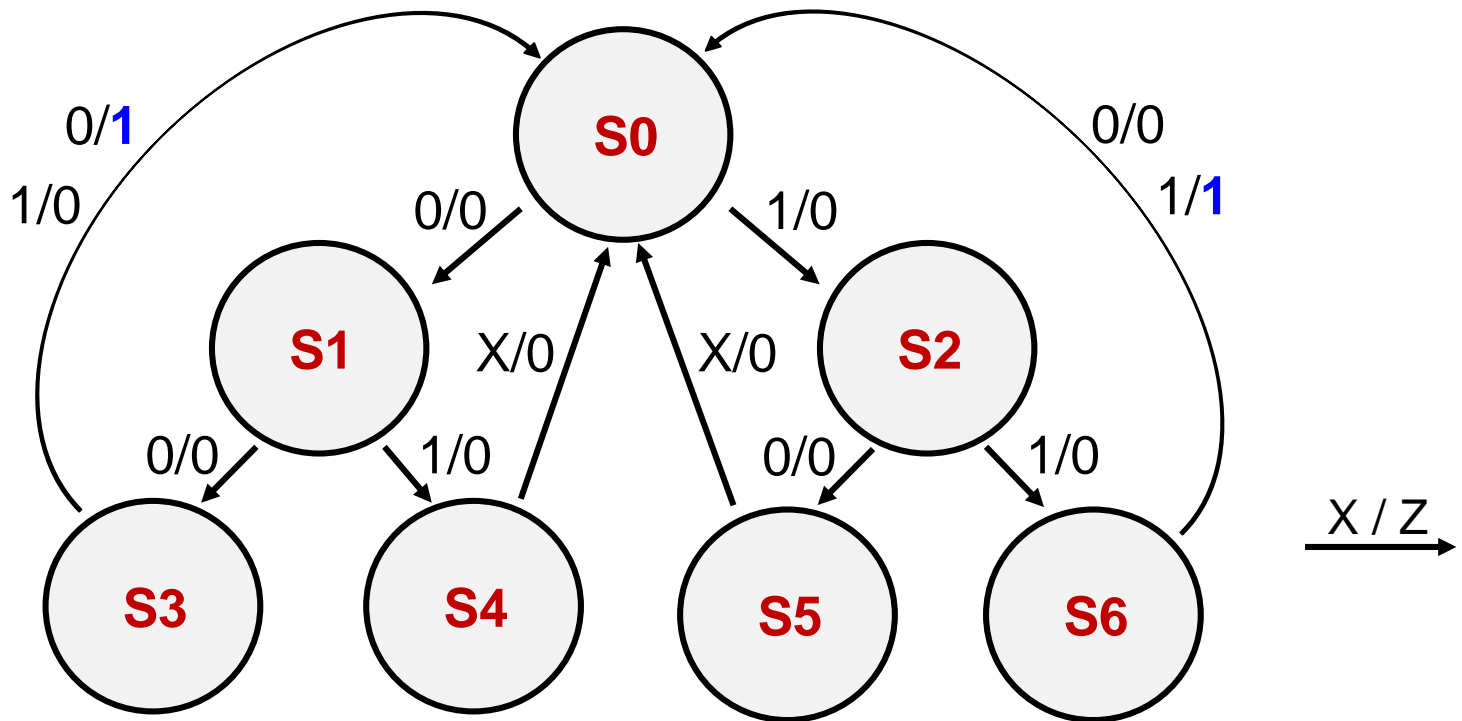
Une nouvelle d tection commencera apr s le passage de 3 bits.

D terminer la table d' tats.

## Ex 20

2. Lois de fonctionnement → Diagramme d'états :

- 1) nouvelle détection (**S0**) → 1<sup>er</sup> bit détecté **X = 0** (**S1**) ou **1** (**S2**);
- 2) 2<sup>ème</sup> bit détecté **X = 0** (**S1**→**S3**, **S2**→**S5**) ou **1** (**S1**→**S4**, **S2**→**S6**);
- 3) 3<sup>ème</sup> bit détecté **X = 0** ou **1** → **Z = 0** ou **1** → nouvelle détection (**S0**);

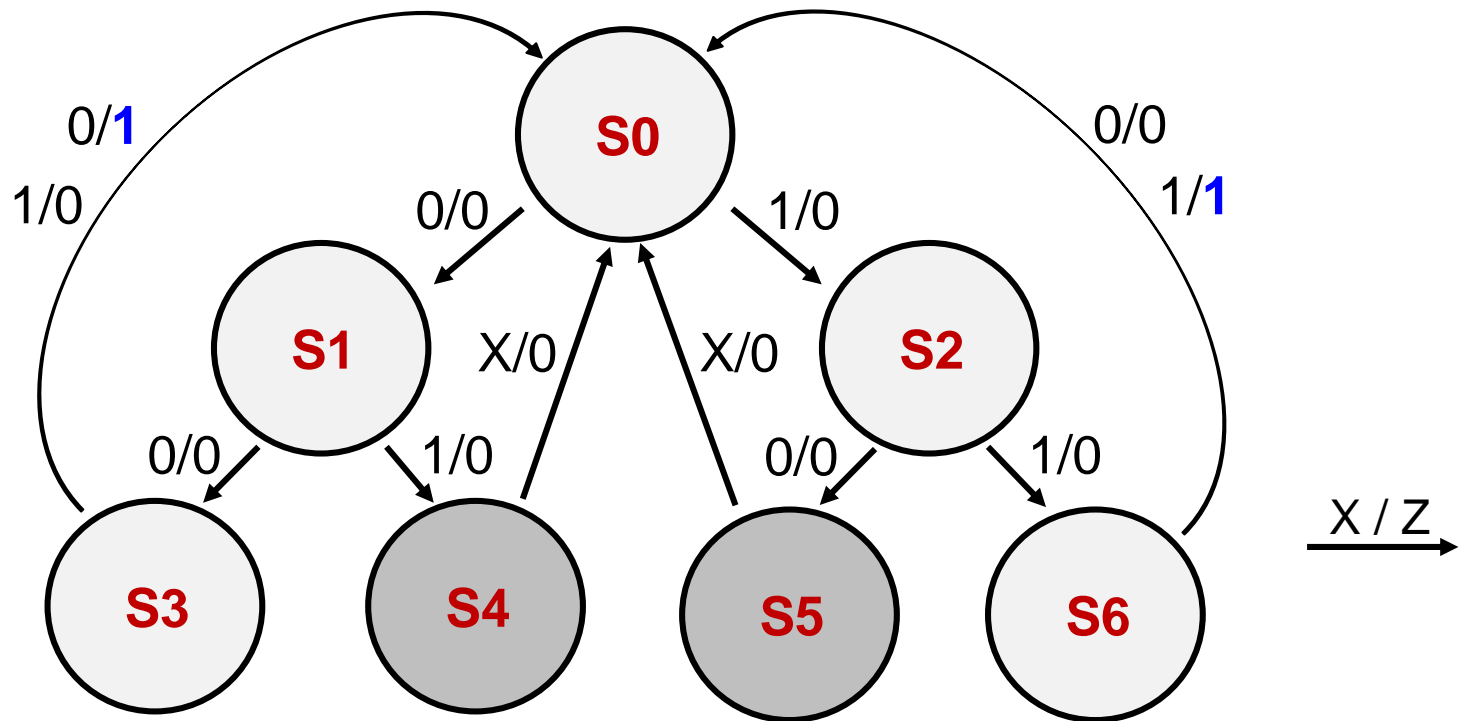




## Ex 20

2. Lois de fonctionnement  $\rightarrow$  Diagramme d'états :

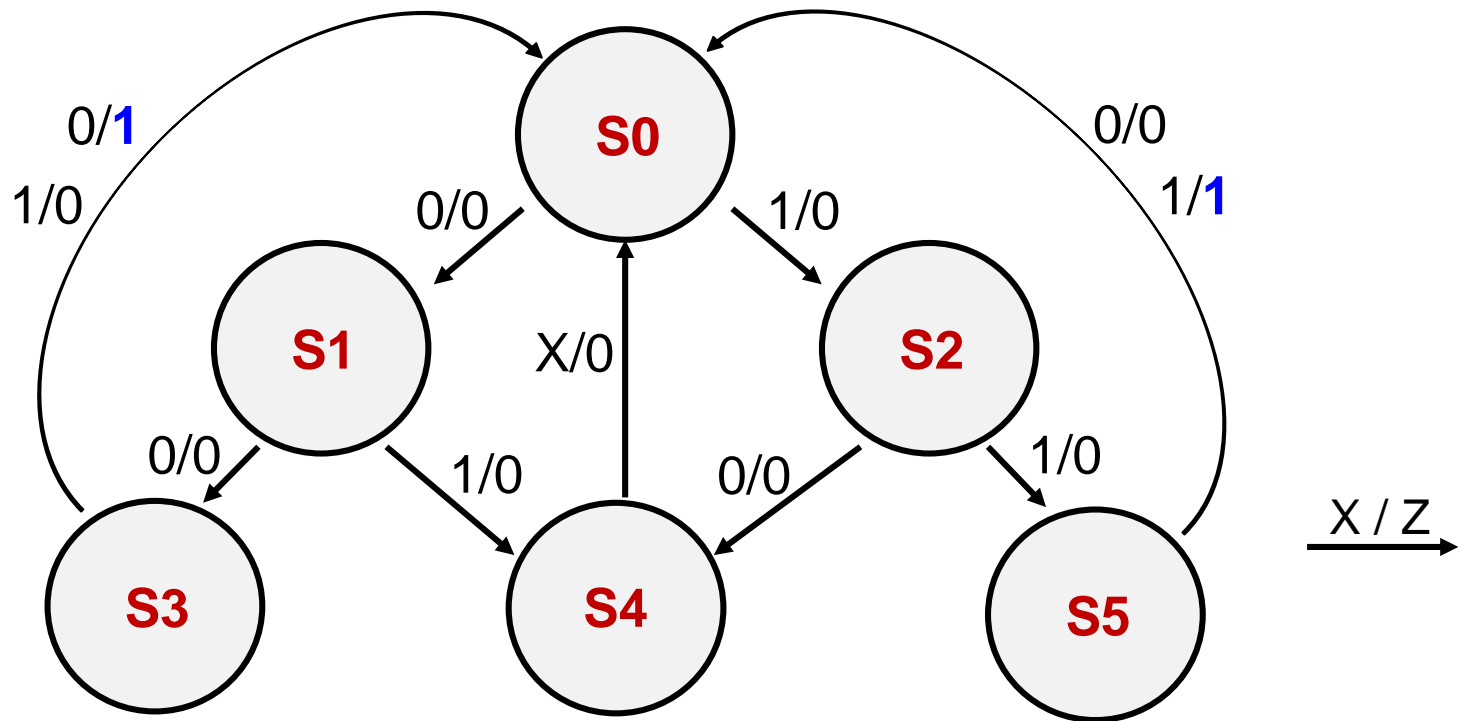
- 1) nouvelle détection (**S0**)  $\rightarrow$  1<sup>er</sup> bit détecté **X = 0** (**S1**) ou **1** (**S2**);
- 2) 2<sup>ème</sup> bit détecté **X = 0** (**S1** $\rightarrow$ **S3**, **S2** $\rightarrow$ **S5**) ou **1** (**S1** $\rightarrow$ **S4**, **S2** $\rightarrow$ **S6**);
- 3) 3<sup>ème</sup> bit détecté **X = 0** ou **1**  $\rightarrow$  **Z = 0** ou **1**  $\rightarrow$  nouvelle détection (**S0**);



## Ex 20

2. Lois de fonctionnement  $\rightarrow$  Diagramme d'états :

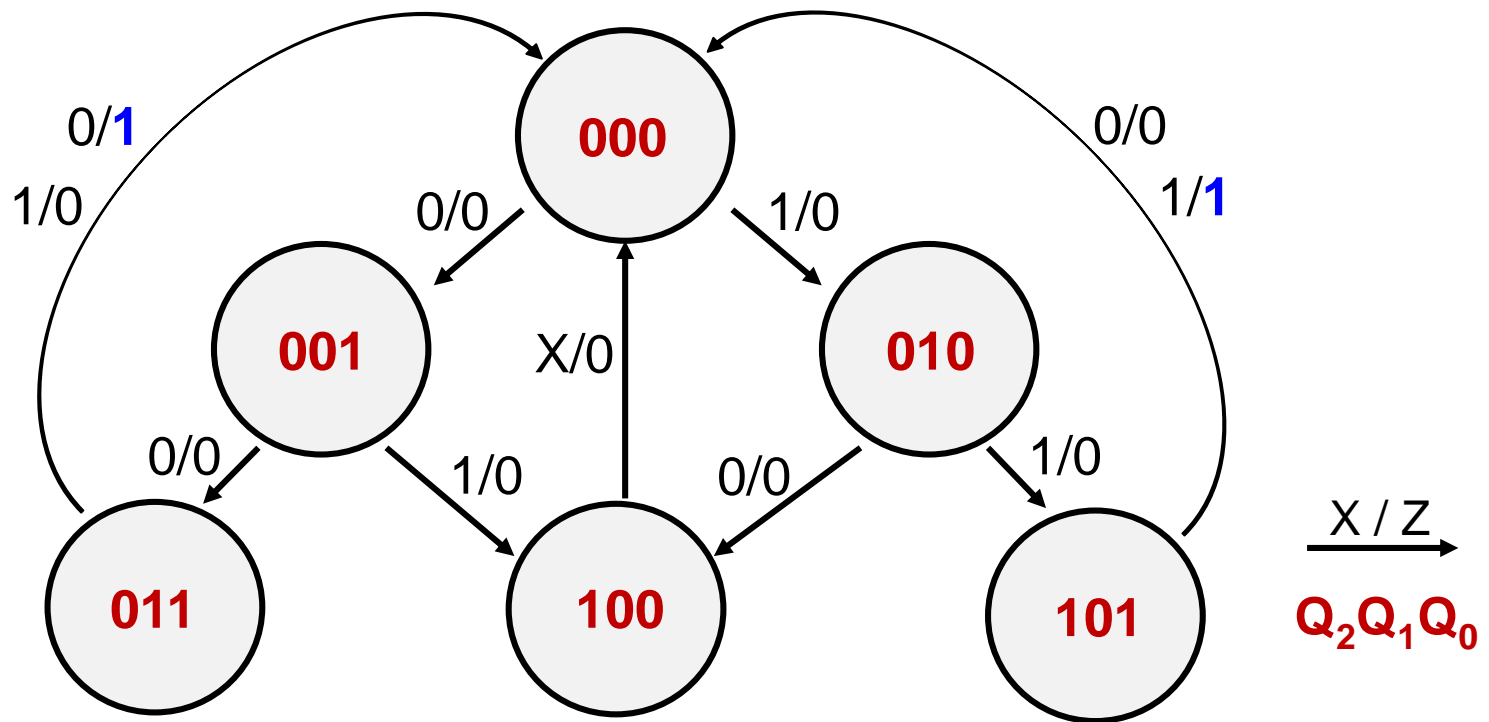
- 1) nouvelle détection (**S0**)  $\rightarrow$  1<sup>er</sup> bit détecté **X = 0 (S1)** ou **1 (S2)**;
- 2) 2<sup>ème</sup> bit détecté **X = 0 (S1  $\rightarrow$  S3, S2  $\rightarrow$  S4)** ou **1 (S1  $\rightarrow$  S4, S2  $\rightarrow$  S5)**;
- 3) 3<sup>ème</sup> bit détecté **X = 0** ou **1  $\rightarrow$  Z = 0** ou **1  $\rightarrow$  nouvelle détection (S0)**;



## Ex 20

2. Lois de fonctionnement  $\rightarrow$  Diagramme d'états :

- 1) nouvelle détection (**000**)  $\rightarrow$  1<sup>er</sup> bit détecté **X = 0** (**001**) ou **1** (**010**);
- 2) 2<sup>ème</sup> bit détecté **X = 0** (**001** $\rightarrow$ **011**, **010** $\rightarrow$ **100**) ou **1** (**001** $\rightarrow$ **100**, **010** $\rightarrow$ **101**);
- 3) 3<sup>ème</sup> bit détecté **X = 0** ou **1**  $\rightarrow$  **Z = 0** ou **1**  $\rightarrow$  nouvelle détection (**000**);



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

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

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

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

## Ex 21

On vous demande de réaliser la synthèse d'un système séquentiel régi par une horloge à 0.05 Hz et dédié à la surveillance de la température des lignes haute tension.

Tant qu'il y a du vent (vitesse supérieure ou égale à 1m/s) et que la valeur du courant reste inférieure ou égale à 1200A, il n'y a rien à craindre.

Lorsque la vitesse du vent tombe à moins d'1m/s et que la valeur du courant reste inférieure ou égale à 1200A, le système passe en mode de surveillance et informe l'utilisateur par un témoin lumineux. De même, si le courant dépasse 1200 A et que la vitesse du vent reste supérieure ou égale à 1m/s. Ce mode reste actif tant que l'une de ces deux conditions est vérifiée.

L'état d'alerte doit être enclenché si la vitesse du vent chute à moins d'1m/s et que la valeur du courant est supérieure à 1200A pendant 20s ou plus. L'alerte est donnée à l'utilisateur au moyen d'une alarme. On maintient l'alerte tant que la valeur du courant reste supérieure à 1200A.

Donnez le nombre de bit(s) d'entrée(s)/sortie(s) et leur signification. Etablissez la machine d'états, les équations des FF-D et des sorties de ce système.



## Ex 21

On vous demande de réaliser la synthèse d'un système séquentiel régi par une horloge à 0.05 Hz et dédié à la surveillance de la température des lignes haute tension.  $\rightarrow T_{clk} = 20s$

Tant qu'il y a du vent (vitesse supérieure ou égale à 1m/s) et que la valeur du courant reste inférieure ou égale à 1200A, il n'y a rien à craindre.

Lorsque la vitesse du vent tombe à moins d'1m/s et que la valeur du courant reste inférieure ou égale à 1200A, le système passe en mode de surveillance et informe l'utilisateur par un témoin lumineux. De même, si le courant dépasse 1200 A et que la vitesse du vent reste supérieure ou égale à 1m/s. Ce mode reste actif tant que l'une de ces deux conditions est vérifiée.

L'état d'alerte doit être enclenché si la vitesse du vent chute à moins d'1m/s et que la valeur du courant est supérieure à 1200A pendant 20s ou plus. L'alerte est donnée à l'utilisateur au moyen d'une alarme. On maintient l'alerte tant que la valeur du courant reste supérieure à 1200A.  $\rightarrow \text{durée critique} = T_{clk}$

Donnez le nombre de bit(s) d'entrée(s)/sortie(s) et leur signification. Etablissez la machine d'états, les équations des FF-D et des sorties de ce système.

# Ex 21

## 1a. Entrées:

- **V** = 1 si la vitesse du vent ( $v_{\text{vent}}$ ) est  $< 1\text{m/s}$  ; 0 si  $v_{\text{vent}} \geq 1\text{m/s}$
- **C** = 1 si la valeur du courant ( $I$ ) est  $> 1200\text{ A}$  ; 0 si  $I \leq 1200\text{A}$

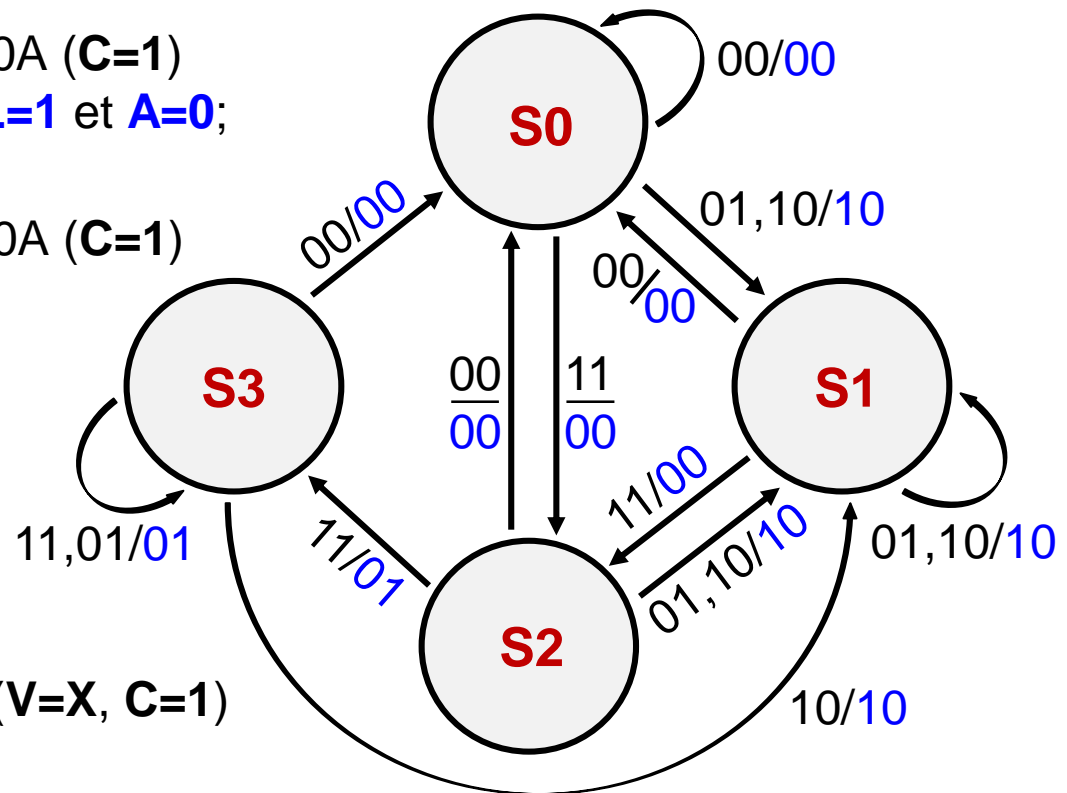
## 1b. Sorties:

- **L** = 1 si le témoin lumineux du mode surveillance est actif ; 0 sinon
- **A** = 1 si l'alarme de l'état d'alerte est déclenchée ; 0 sinon

# Ex 21

## 2. Lois de fonctionnement → Diagramme d'états :

- 1) Si  $v_{vent} \geq 1 \text{ m/s}$  ( $V=0$ ) et  $I \leq 1200 \text{ A}$  ( $C=0$ ) → tout est OK (**S0**):  $L=A=0$ ;
- 2) Si  $v_{vent} < 1 \text{ m/s}$  ( $V=1$ ) et  $I \leq 1200 \text{ A}$  ( $C=0$ ) → mode surveillance (**S1**):  $L=1$  et  $A=0$ ;
- 3) Si  $v_{vent} \geq 1 \text{ m/s}$  ( $V=0$ ) et  $I > 1200 \text{ A}$  ( $C=1$ )  
→ mode surveillance (**S1**):  $L=1$  et  $A=0$ ;
- 4) Si  $v_{vent} < 1 \text{ m/s}$  ( $V=1$ ) et  $I > 1200 \text{ A}$  ( $C=1$ )  
pendant durée  $< T_{clk}$   
→ mode pré-alerte (**S2**):  
 $L=0$  et  $A=0$ ;  
pendant durée  $\geq T_{clk}$   
→ mode alerte (**S3**):  
 $L=0$  et  $A=1$ ;  
maintenu tant que  $I > 1200 \text{ A}$  ( $V=X$ ,  $C=1$ )



VC/LA → State

$Q_1(t)$	$Q_0(t)$	V	C	$Q_1(t+1)$	$Q_0(t+1)$	L	A
0	0	0	0	0	0	0	0
0	0	0	1				
0	0	1	0				
0	0	1	1				
0	1	0	0	0	0	0	0
0	1	0	1				
0	1	1	0				
0	1	1	1				
1	0	0	0	0	0	0	0
1	0	0	1				
1	0	1	0				
1	0	1	1				
1	1	0	0	0	0	0	0
1	1	0	1				
1	1	1	0				
1	1	1	1				

$Q_1(t)$	$Q_0(t)$	V	C	$Q_1(t+1)$	$Q_0(t+1)$	L	A
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	0
0	0	1	0	0	1	1	0
0	0	1	1				
0	1	0	0	0	0	0	0
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	0
0	1	1	1				
1	0	0	0	0	0	0	0
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	0
1	0	1	1				
1	1	0	0	0	0	0	0
1	1	0	1	!!!	!!!	!!!	!!!
1	1	1	0	0	1	1	0
1	1	1	1				

$Q_1(t)$	$Q_0(t)$	V	C	$Q_1(t+1)$	$Q_0(t+1)$	L	A
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	0
0	0	1	0	0	1	1	0
0	0	1	1	1	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	0
0	1	1	1	1	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	0
1	0	1	1				
1	1	0	0	0	0	0	0
1	1	0	1				
1	1	1	0	0	1	1	0
1	1	1	1				

$Q_1(t)$	$Q_0(t)$	V	C	$Q_1(t+1)$	$Q_0(t+1)$	L	A
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	0
0	0	1	0	0	1	1	0
0	0	1	1	1	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	0
0	1	1	1	1	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	0
1	0	1	1	1	1	0	1
1	1	0	0	0	0	0	0
1	1	0	1				
1	1	1	0	0	1	1	0
1	1	1	1				

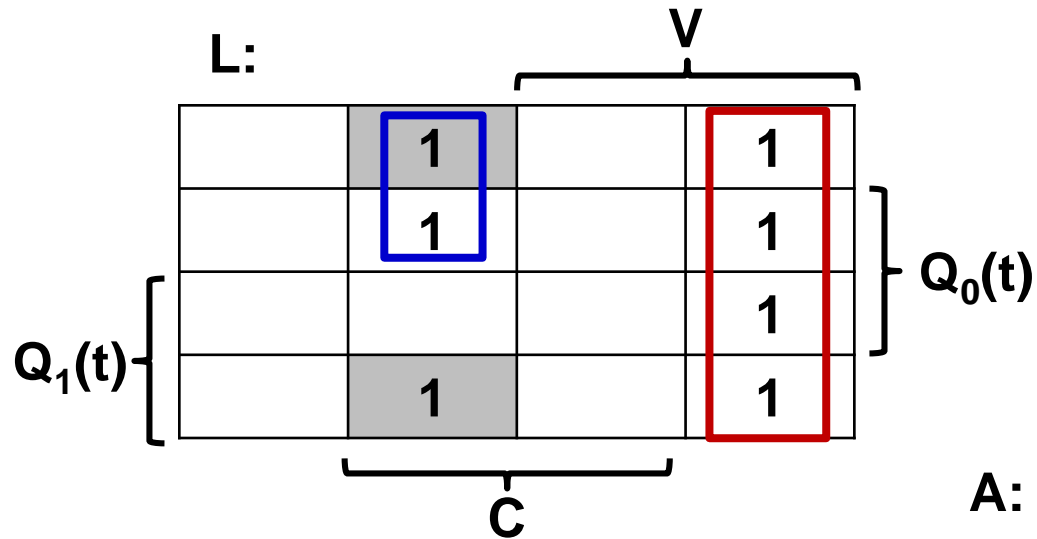
$Q_1(t)$	$Q_0(t)$	V	C	$Q_1(t+1)$	$Q_0(t+1)$	L	A
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	0
0	0	1	0	0	1	1	0
0	0	1	1	1	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	0
0	1	1	1	1	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	0
1	0	1	1	1	1	0	1
1	1	0	0	0	0	0	0
1	1	0	1	1	1	0	1
1	1	1	0	0	1	1	0
1	1	1	1	1	1	0	1



$Q_1(t)$	$Q_0(t)$	V	C	$Q_1(t+1)$	$Q_0(t+1)$	L	A
0	0	0	0	0	0	0	0
0	0	0	1	0	1	1	0
0	0	1	0	0	1	1	0
0	0	1	1	1	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	0
0	1	1	1	1	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	0
1	0	1	1	1	1	0	1
1	1	0	0	0	0	0	0
1	1	0	1	1	1	0	1
1	1	1	0	0	1	1	0
1	1	1	1	1	1	0	1

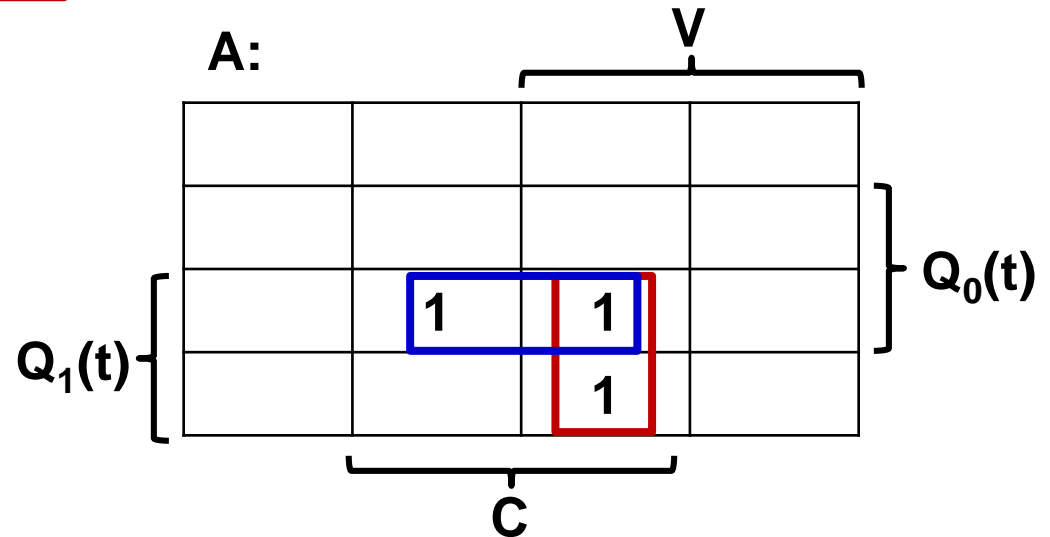
# Ex 21

## 4. Fonctions logiques des flip-flops et sorties:



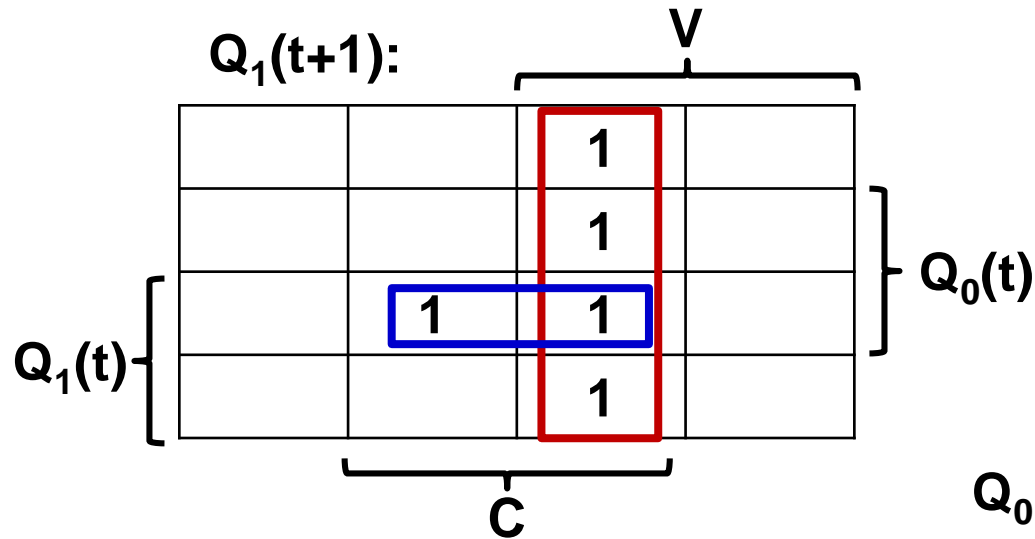
$$L = \overline{Q_1(t)} \overline{V} C + V \overline{C} + Q_0(t) \overline{V} C$$

$$A = Q_1(t) Q_0(t) C + Q_1(t) V C$$



# Ex 21

## 4. Fonctions logiques des flip-flops et sorties:



$$\begin{aligned}
 Q_1(t+1) &= Q_1(t) Q_0(t) C \\
 &\quad + V C \\
 &= D_1(t)
 \end{aligned}$$

$$\begin{aligned}
 Q_0(t+1) &= Q_1(t) V \\
 &\quad + \overline{V} C + V \overline{C} \\
 &= D_0(t)
 \end{aligned}$$

