

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR**  
**ET DE LA RECHERCHE SCIENTIFIQUE**



**Université Larbi Ben M'hidi Oum el Bouaghi**  
**ISTA Ain M'lila**  
**Département Réseaux et Télécommunications**



**1ère Année Licence**  
**Module : Informatique 1**

**Logique combinatoire et séquentielle**

© **Dr BENACER Imad Jan 2021**

**Année Universitaire : 2020 / 2021**



# Logique séquentielle

## Objectifs

Apprendre la structure de quelques circuits séquentielles souvent utilisés ( Bascules, registres, compteurs.....).

Apprendre comment utiliser des circuits séquentielles pour concevoir d'autres circuits plus complexes.



## II. Les Circuits séquentielles

La mémoire de l'ordinateur stocke les données et les instructions du programme en cours d'exécution. Pour réaliser cette mémoire, des circuits sont utilisés dits circuits de mémorisation. Pour mémoriser un bit (0 ou 1), il faut utiliser un circuit capable de se souvenir de la valeur qu'il a enregistrée.

Ces circuits sont appelés circuits séquentiels.

Un circuit séquentiel est un système dont les sorties à l'instant  $(t)$  dépendent à la fois de l'état des entrées et aussi de l'état précédent de la sortie. On peut représenter un système séquentiel par le schéma suivant

## II. Les Circuits séquentielles

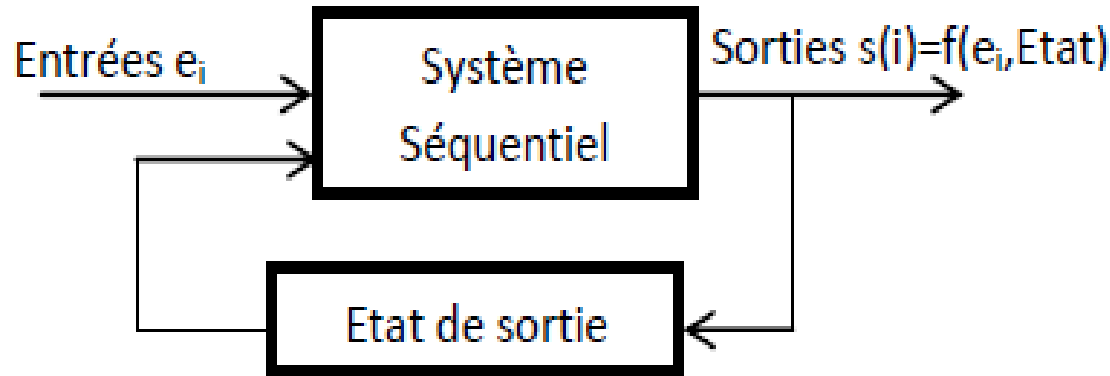


Schéma d'un système séquentiel

Les systèmes séquentiels sont classés en 2 catégories :

### + Circuits séquentiels asynchrones

Dans les circuits séquentiels asynchrones, les sorties changent d'états dès qu'il y a changements des états des entrées.

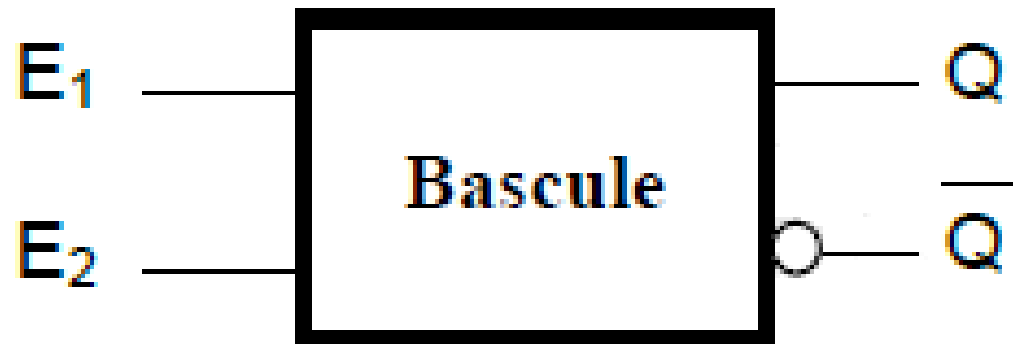
### + Circuits séquentiels synchrones

Dans ce type de circuits les sorties changent d'états après avoir eu une autorisation d'un signal de synchronisation appelé souvent signal « Horloge » noté H ou CLK.

# Les bascules

Une bascule (**flip-flops**) est un élément capable de stocker un bit et le restituer au temps voulu. Les bascules sont utilisées pour réaliser des circuits de mémorisation.

Elle est un système séquentiel constitué par une ou deux entrées et deux sorties complémentaires.



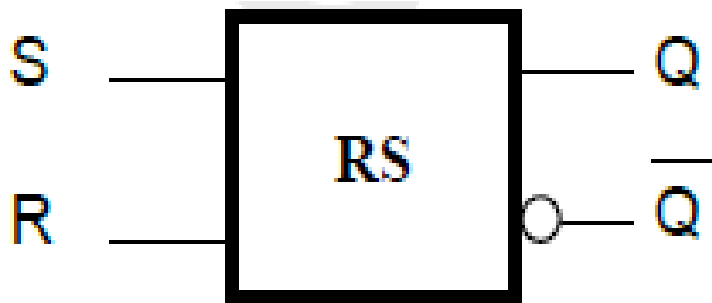
$Q$  : est la variable de sortie de la bascule. Elle fournit l'état de la bascule. Si  $Q=1$ , on dit que l'état de la bascule est à 1. Sinon, l'état de la bascule est 0.

$\overline{Q}$ : C'est l'inverse de la variable de sortie  $Q$ .

# Les bascules

On l'appelle ainsi « bascule bistable » car elle possède deux états stables. On distingue 4 types de bascules : RS, D, JK, et T.

## 1. Bascule RS



S mise à 1 (marche) implique que  $Q=1$  (Q est forcé à un par S)

R mise à 0 (Arrêt) implique que  $Q=0$  (Q est forcé à zéro par R)

## 1. Bascule RS

Pour comprendre le fonctionnement de la bascule R-S, on va étudier le comportement des variables de sortie ( $Q$  et  $\overline{Q}$ ) en fonction des variables d'entrée (R-S).

Pour cela, on désigne par :

- $Q_t$  : la variable de sortie à l'instant  $t$  (l'état présent de la variable).
- $Q_{t+1}$  : la variable de sortie à l'instant  $t+1$  (l'état futur de la variable).

# Les bascules

## 1. Bascule RS

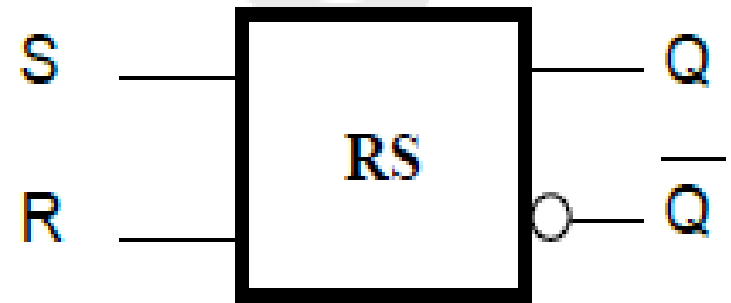
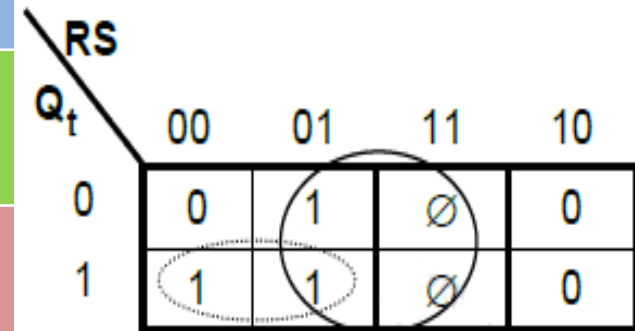


Table de vérité

S	R	Q <sub>t</sub>	Q <sub>t+1</sub>	$\overline{Q_{t+1}}$	
0	0	0	0	1	Q <sub>t</sub> = Q <sub>t+1</sub> (Mémoire) Garder l'état précédent
0	0	1	1	0	
1	0	0	1	0	Q <sub>t+1</sub> = 1 (Mise à 1)
1	0	1	1	0	
0	1	0	0	1	Q <sub>t+1</sub> = 0 (Remise à 0)
0	1	1	0	1	
1	1	0	x	x	Etats indéterminés
1	1	1	x	x	



$$Q_{t+1} = \overline{R}Q_t + S$$



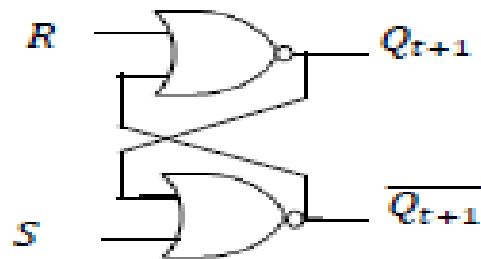
# Les bascules

## 1. Bascule RS

Schéma logique de bascule RS

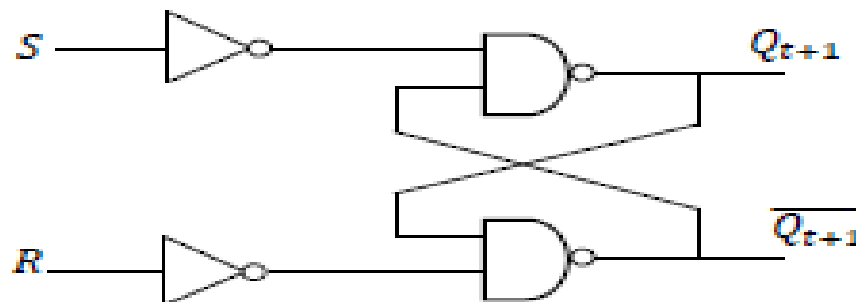
A) Représentation par les portes logique NOR

$$Q_{t+1} = R(Q_t + S) = \overline{R + \overline{(Q_t + S)}}$$



B) Représentation par les portes logique NAND

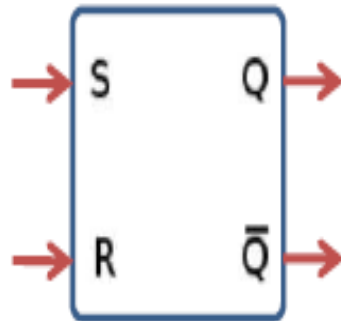
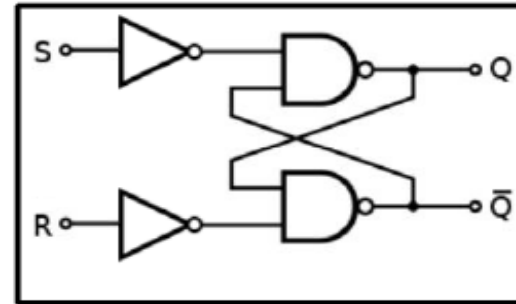
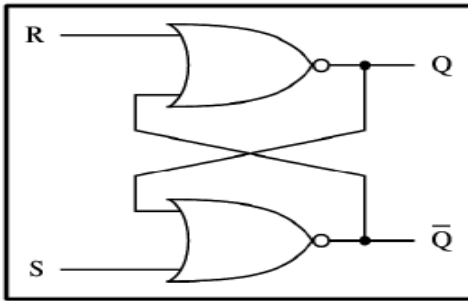
$$Q_{t+1} = \overline{\overline{R(Q_t + S)}} = \overline{(\overline{R}Q_t).S}$$



# Les bascules

## 1. Bascule RS

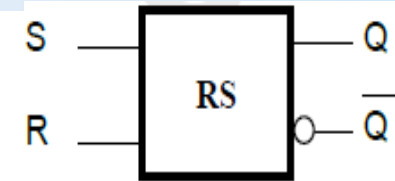
Qu'il soit à base de portes NOR ou de portes NAND, le bistable RS asynchrone sera désormais représenté par son schéma bloc et sa table de vérité, comme suit:



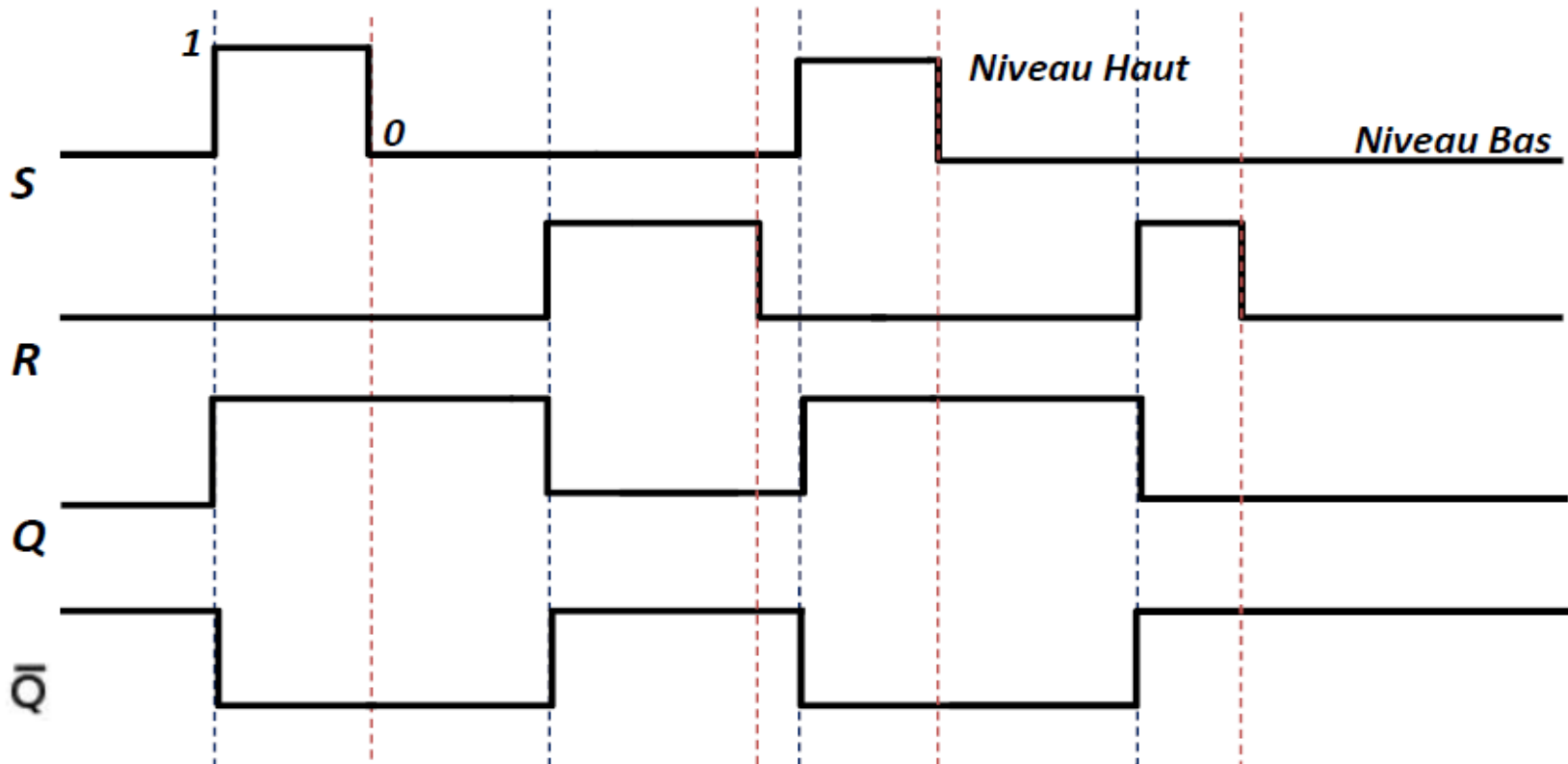
R	S	$Q_{t+1}$
0	0	$Q_t$
0	1	1
1	0	0
1	1	<i>interdit</i>

# Les bascules

## 1. Bascule RS



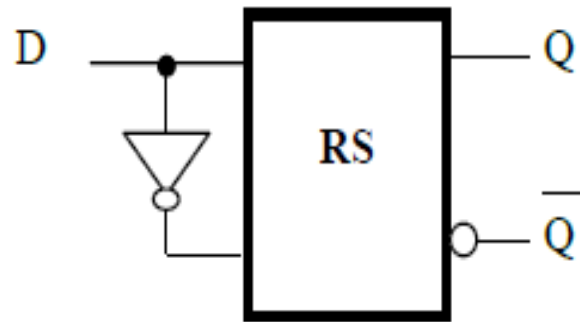
**Notion de chronogramme:** graphique qui montre l'évolution des signaux, en entrée comme en sortie, dans le temps.



# Les bascules

## 2. Bascule D (Latch)

Bascule D est utilisée pour éliminer l'état indéterminé (interdit), il est résumé les deux entrées R et S par une seule entrée D telle que :  $S = D$  et  $R = \overline{D}$



Symbole	Explication
	<ul style="list-style-type: none"><li>• Si <math>D=1</math>, alors la bascule passe à l'état 1.</li><li>• Si <math>D=0</math>, alors la bascule passe à l'état 0.</li></ul>

D	$Q_{t+1}$
0	0
1	1

# Les bascules

## 2. Bascule D

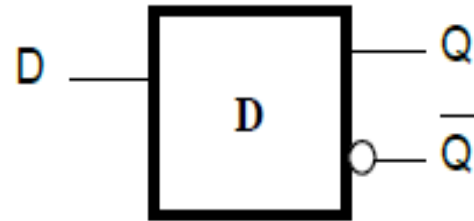
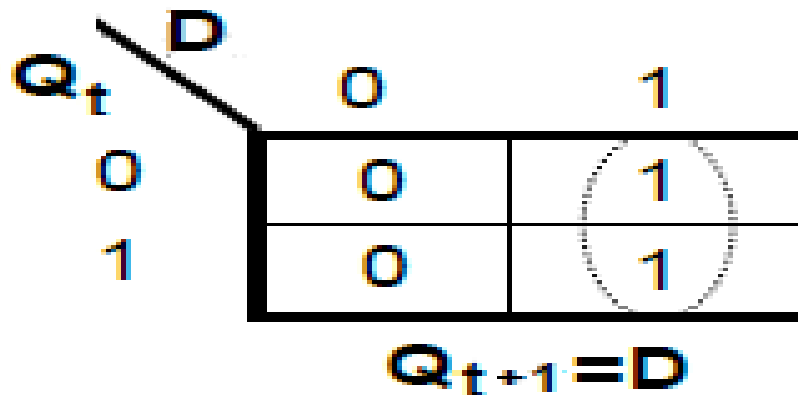


Table de vérité

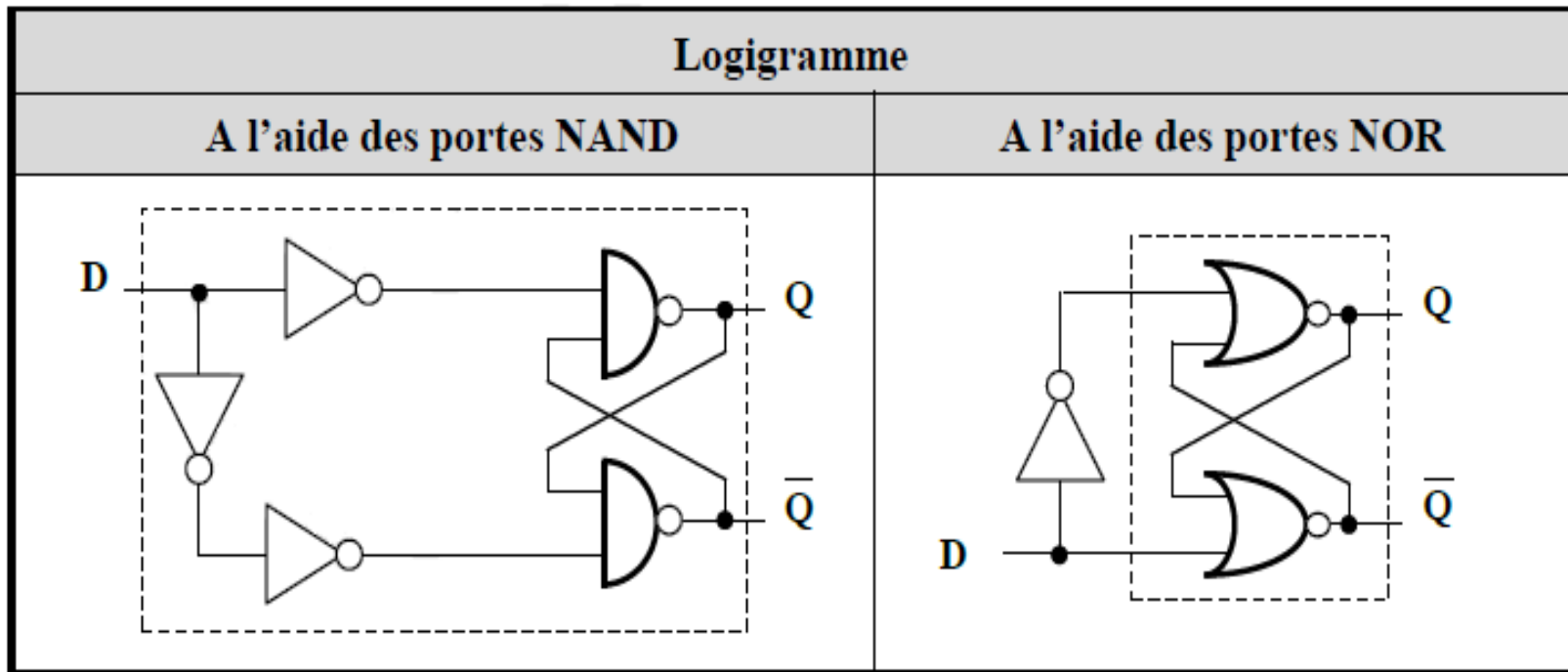
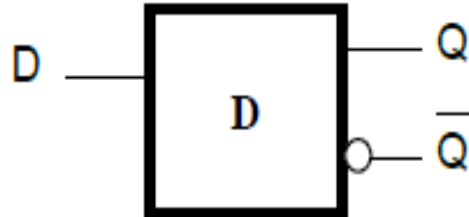
D	Q <sub>t</sub>	Q <sub>t+1</sub>	
0	0	0	Q <sub>t+1</sub> = D = 0 (D qui passe)
0	1	0	
1	0	1	Q <sub>t+1</sub> = D = 1 (D qui passe)
1	1	1	



# Les bascules

## 2. Bascule D

Table de vérité



Remarque : En mettant  $S=D$  et  $R=\bar{D}$  dans l'équation de la bascule RS on aura  $Q_{n+1}=DQ_n + \bar{D} = D(1+Q_n) = D$ .

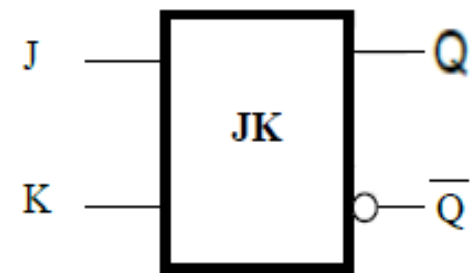
# Les bascules

## 3. Bascule JK

Contrairement à la bascule RS, la condition  $J=K=1$ , ne donne pas lieu à une condition indéterminée, mais par contre la bascule passe à l'état opposé.

Table de vérité

j	k	Qt	Qt+1	
0	0	0	0	Qt+1 = Qt (Etat précédent)
0	0	1	1	
0	1	0	0	Qt+1=0 (Remise à 0)
0	1	1	0	
1	0	0	1	Qt+1=1 (Mise à 1)
1	0	1	1	
1	1	0	1	Qt+1 = Qt
1	1	1	0	

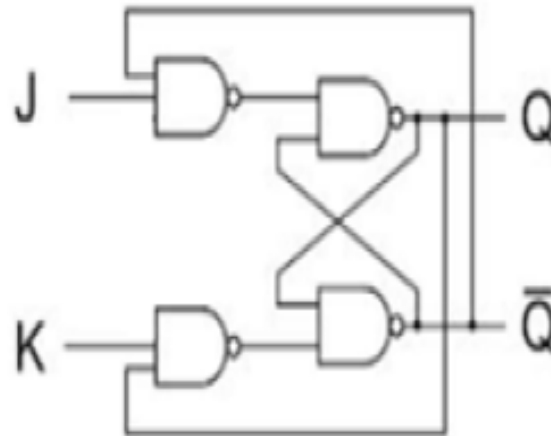
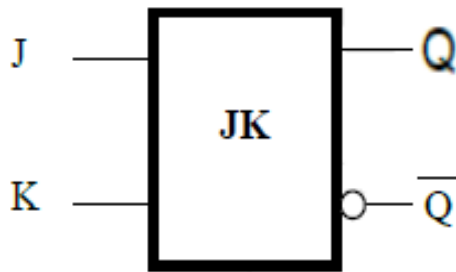


Qt \ JK	00	01	11	10
0	0	0	1	1
1	1	0	0	1

$$Q_{t+1} = J\bar{Q}_t + \bar{K}Q_t$$

# Les bascules

## 3. Bascule JK



<i>J</i>	<i>K</i>	$Q_{t+1}$
<i>0</i>	<i>0</i>	$Q_t$
<i>0</i>	<i>1</i>	<i>0</i>
<i>1</i>	<i>0</i>	<i>1</i>
<i>1</i>	<i>1</i>	$\bar{Q}_t$

Le bistable JK possède une autre table appelée table de transition.

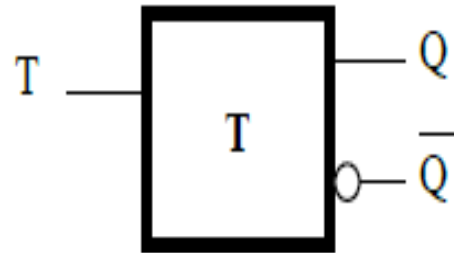
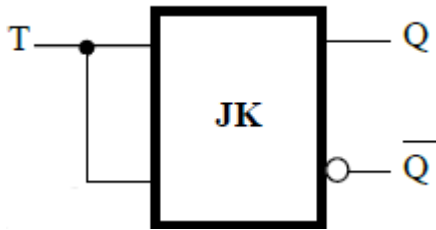
$Q_t$	$Q_{t+1}$	<i>J</i>	<i>K</i>
<i>0</i>	<i>0</i>	<i>0</i>	<i>X</i>
<i>0</i>	<i>1</i>	<i>1</i>	<i>X</i>
<i>1</i>	<i>0</i>	<i>X</i>	<i>1</i>
<i>1</i>	<i>1</i>	<i>X</i>	<i>0</i>



# Les bascules

## 4. Bascule T

La bascule T est obtenue en reliant les entrées J et K d'une bascule JK.



### Table de vérité

T	Q <sub>t</sub>	Q <sub>t+1</sub>	
0	0	0	Q <sub>t+1</sub> = Q <sub>t</sub> (pas de changement)
0	1	1	
1	0	1	Q <sub>t+1</sub> = $\overline{Q_t}$ (complémentation)
1	1	0	

A Karnaugh map for the T flip-flop. The vertical axis is labeled 'Q<sub>t</sub>' and the horizontal axis is labeled 'T'. The cells contain the values of Q<sub>t+1</sub> for each combination of T and Q<sub>t</sub>. The cells (0,1) and (1,0) contain '1' and are circled.

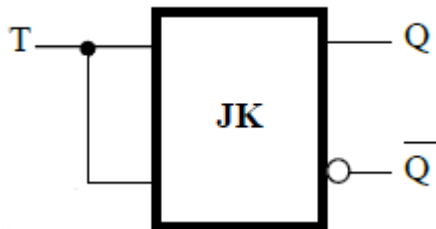
Q <sub>t</sub> \ T	0	1
0	0	1
1	1	0

$$Q_{t+1} = \overline{T}Q_t + T\overline{Q}_t = T \oplus Q_t$$

# Les bascules

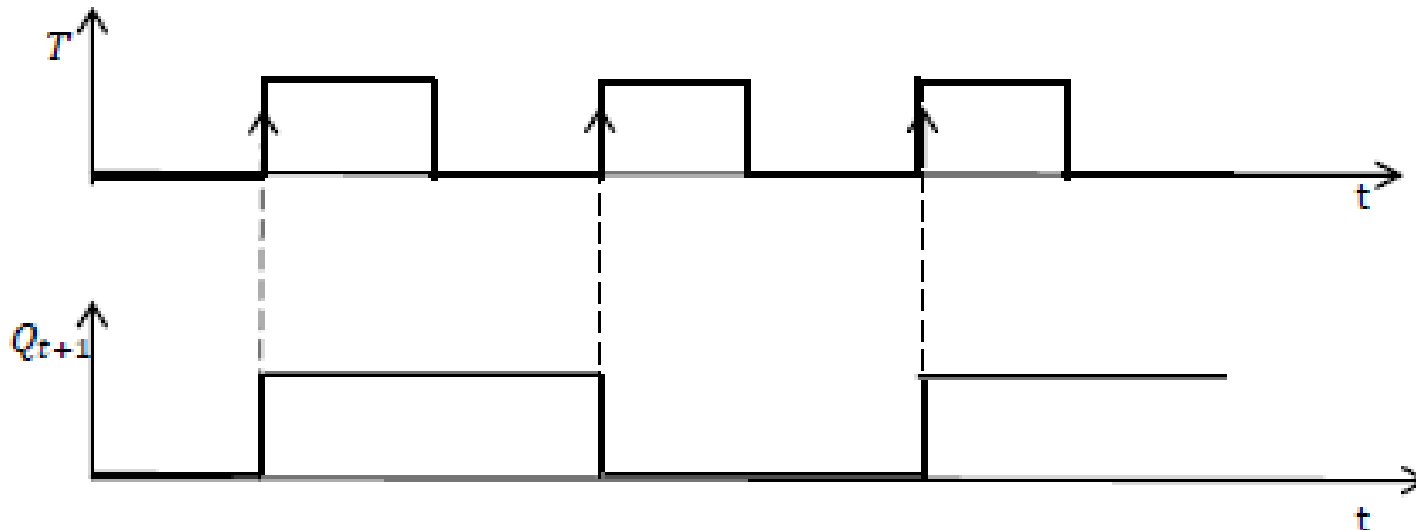
## 4. Bascule T

La bascule T est obtenue en reliant les entrées J et K d'une bascule JK.



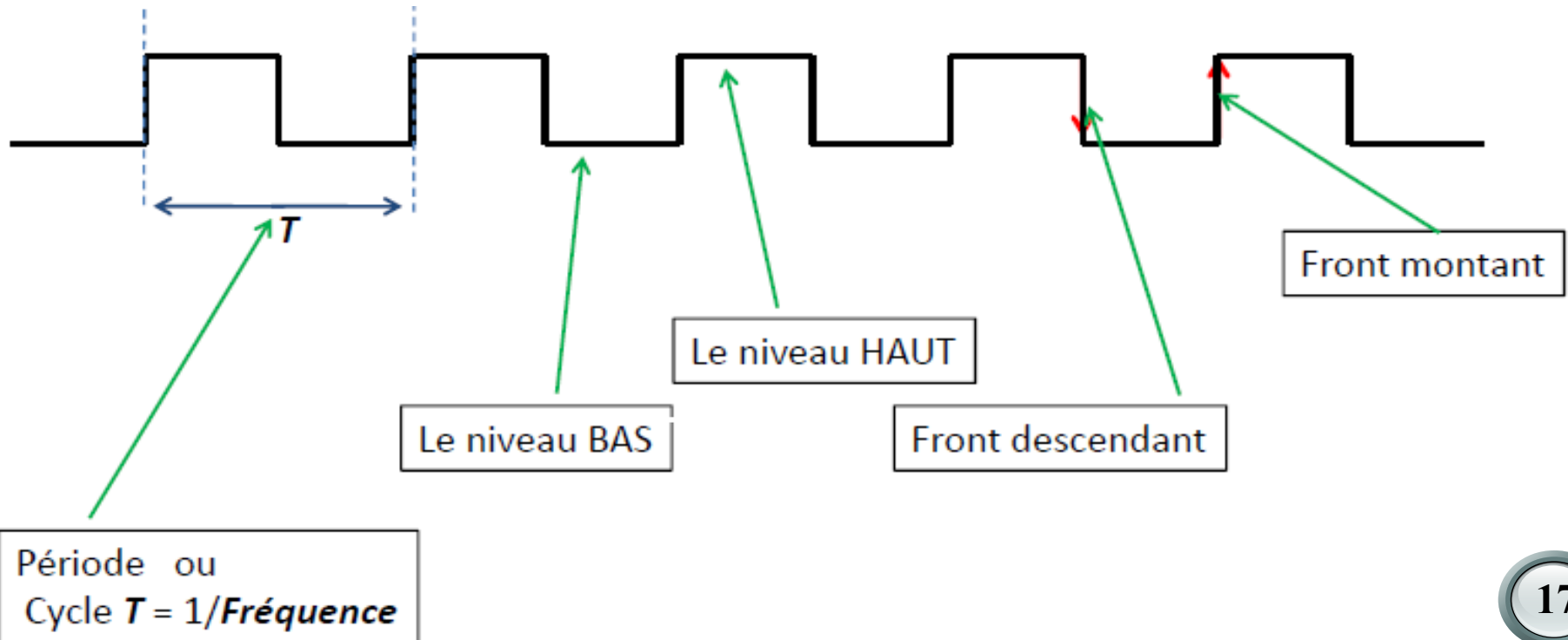
Si  $T=0$  → mémorisation de l'état précédent

Si  $T=1$  → la sortie de la bascule sera changé tel que  $0 \rightarrow 1$  et  $1 \rightarrow 0$

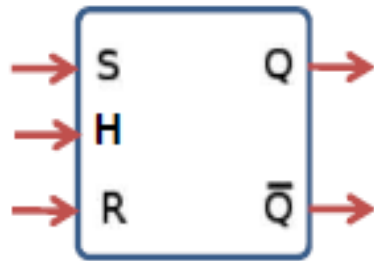


# Horloge (Clock).

- Une bascule est synchrone quand ses sorties ne changent d'état que si un signal supplémentaire est appliquée sur une entrée, dite entrée d'horloge (notée H ou CLK).
- L'horloge est souvent représentée par un train d'impulsion, comme l'illustre la figure suivante, qui définit aussi quelques termes clés associés au signal d'horloge:

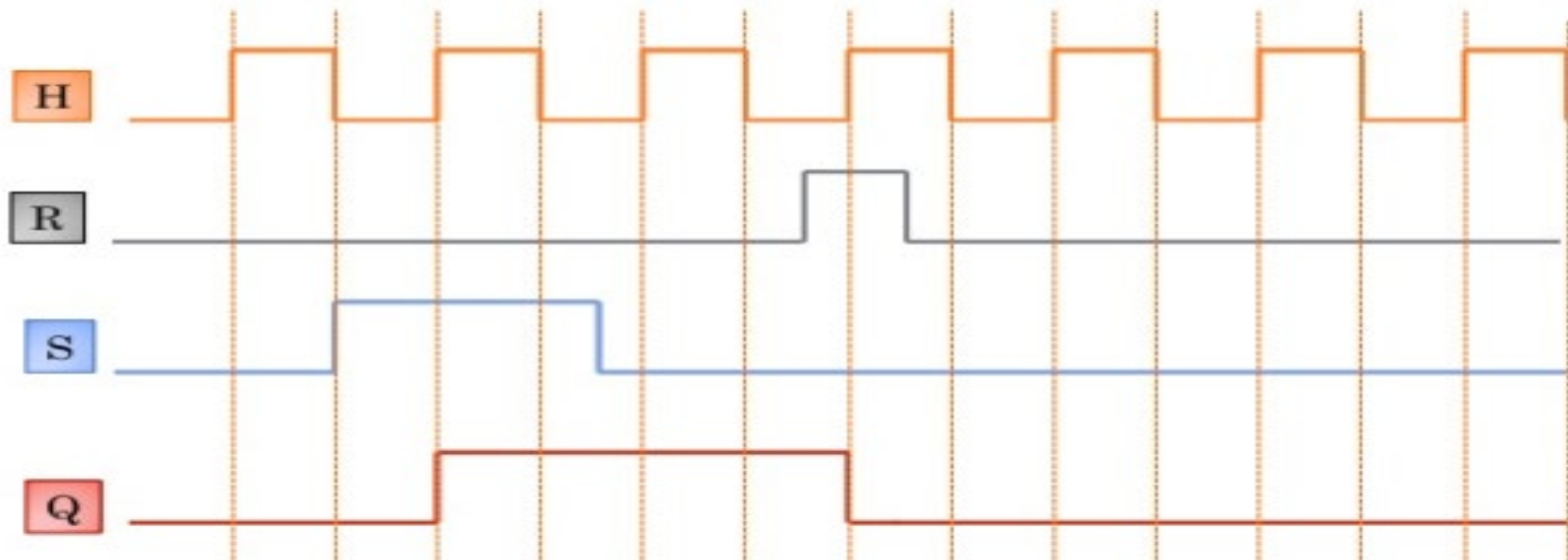


# Horloge (Clock).

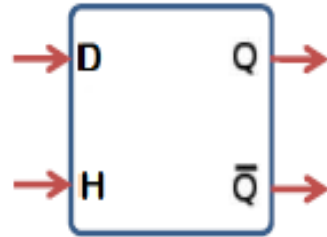


$H$	$R$	$S$	$Q_{t+1}$
0	X	X	$Q_t$
1	0	0	$Q_t$
1	0	1	1
1	1	0	0
1	1	1	<i>interdit</i>

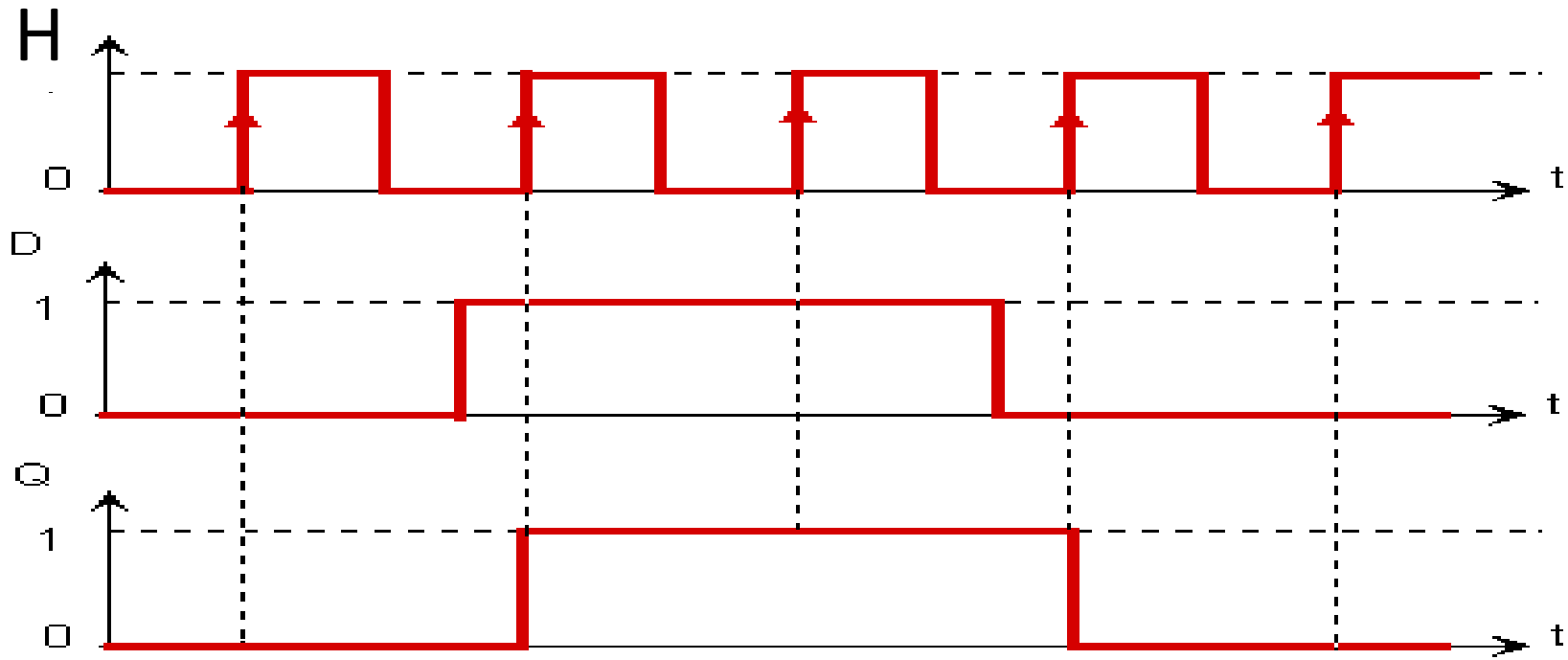
## CHRONOGRAMME



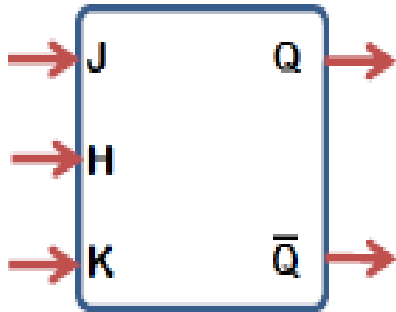
# Horloge (Clock)



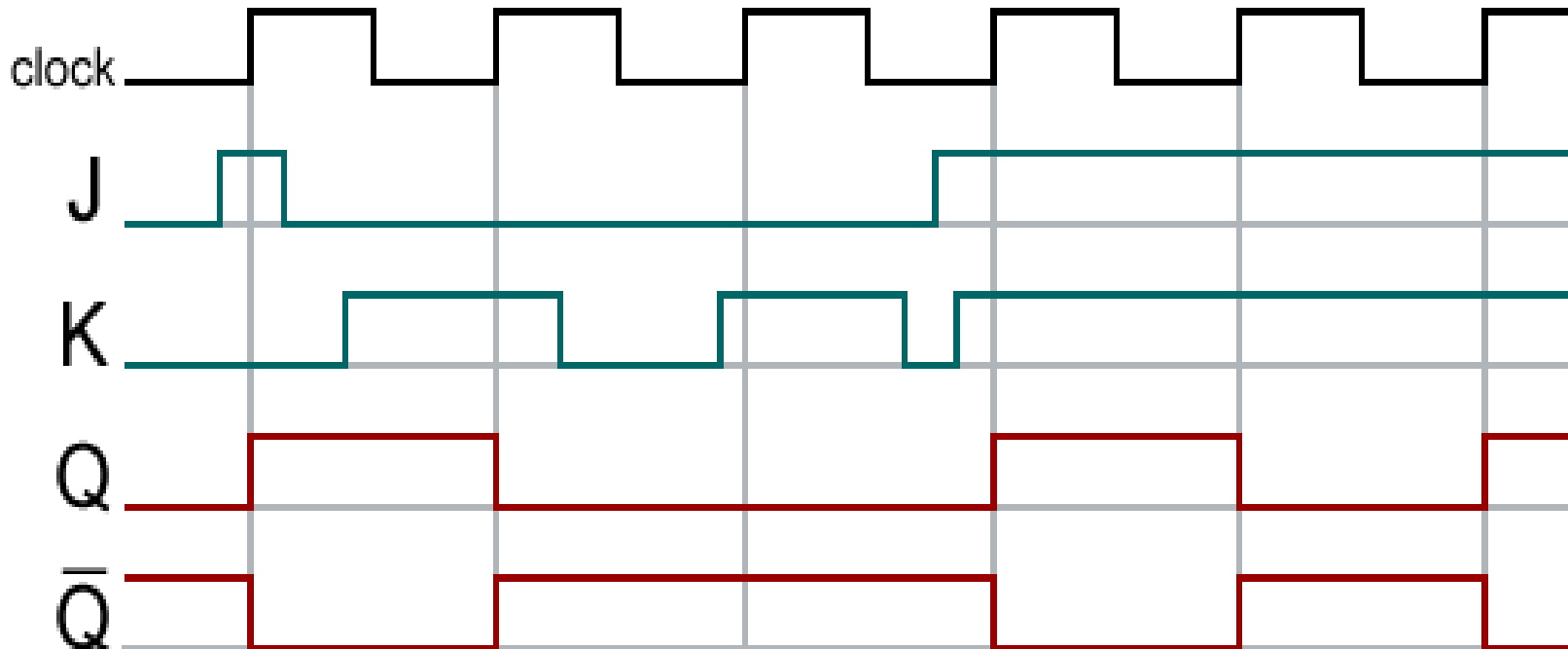
$H$	$D$	$Q_{t+1}$
0	X	$Q_t$
1	0	0
1	1	1



# Horloge (Clock)

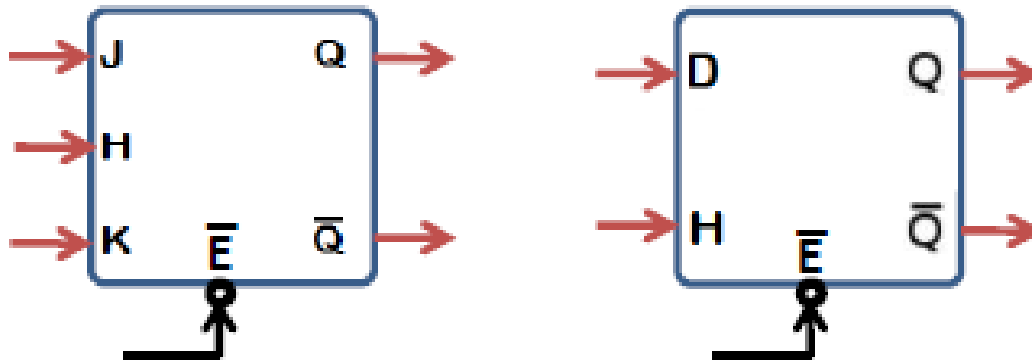


$H$	$K$	$J$	$Q_{t+1}$
0	X	X	$Q_t$
1	0	0	$Q_t$
1	0	1	1
1	1	0	0
1	1	1	$\bar{Q}_t$



# Forçage des bascules & Enable

- L'entrée **Enable**, lorsqu'elle est présente, permet au circuit globalement d'être actif ou inactif



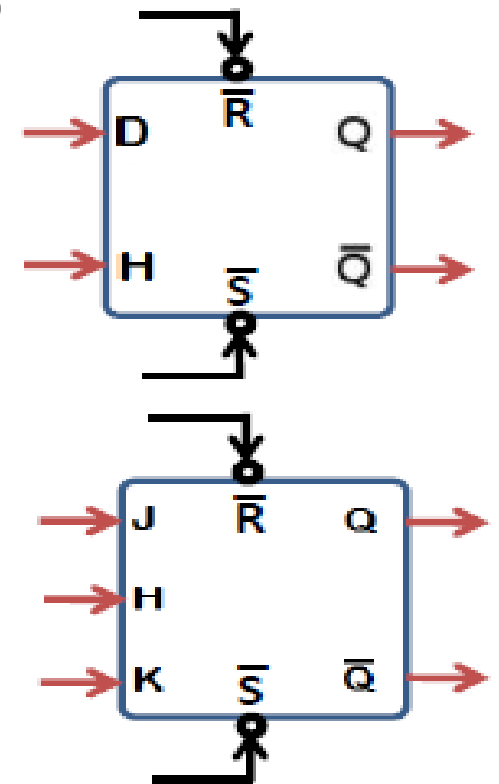
On applique le même raisonnement pour les bascules D, T et JK.

# Forçage des bascules & Enable

- Certains bistables possèdent des entrées asynchrones de mise à 1 y et de mise à 0. Ces entrées sont nommées de diverses manières (R et S pour Reset et Set; PR et CLR pour PReset et CLear ...). Par ailleurs elles peuvent être actives au niveau bas ou au niveau haut (souvent bas, auquel cas leurs entrées sont annotées avec un petit rond)

Table de vérité

Entrées		Sorties		Mode de fonctionnement
PRESET	CLEAR	$Q_{t+1}$	$\overline{Q}_{t+1}$	
0	0	$Q_t$	$\overline{Q}_t$	Mémorisation
0	1	0	1	Forçage à 1
1	0	1	0	Forçage à 0
1	1	$\emptyset$	$\emptyset$	Interdit



On applique le même raisonnement pour les bascules D, T et JK.





# Synchronisation sur niveau et sur front

Un bistable synchrone, donc piloté par une horloge, peut changer son état interne soit sur niveau ( haut ou bas) de l'horloge, soit sur un front (montant ou descendant) du même signal.

## Synchronisation sur niveau

- ⚡ Si  $H=1$  :  $Q$  garde l'état précédent.
- ⚡ Si  $H=0$  : Fonctionnement normal de la bascule.

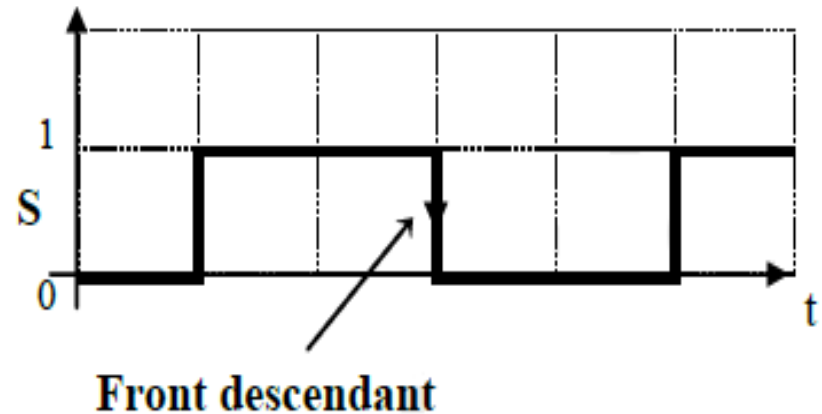
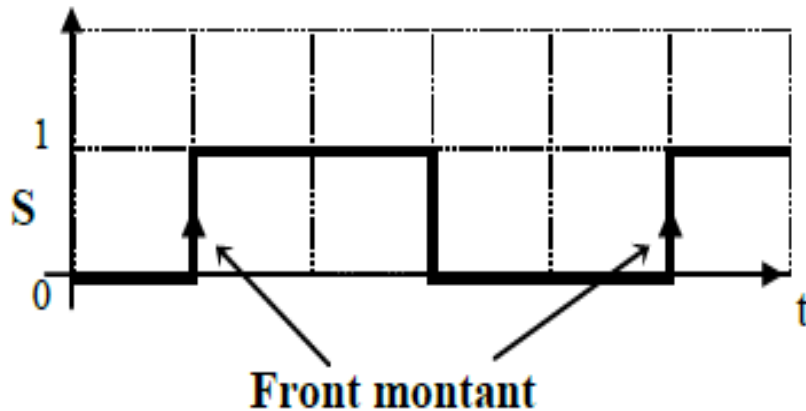
## Synchronisation sur front

Une variable logique  $S$  peut avoir deux niveaux : le niveau haut (Vrai) ou le niveau logique bas (Faux). Quand elle passe du niveau bas vers le niveau haut, elle définit le front montant. Dans le cas contraire, elle définit le front descendant.

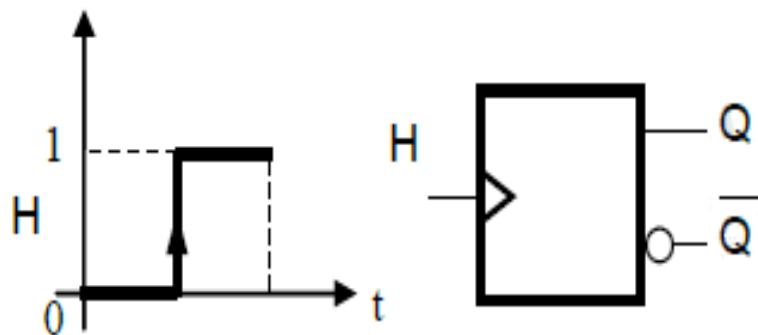


# Synchronisation sur niveau et sur front

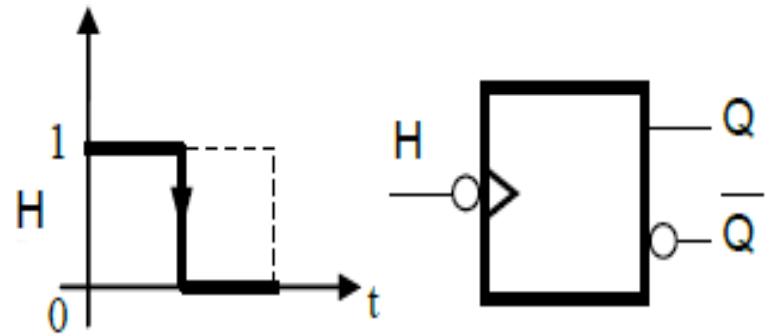
## Synchronisation sur front



### Front montant



### Front descendant

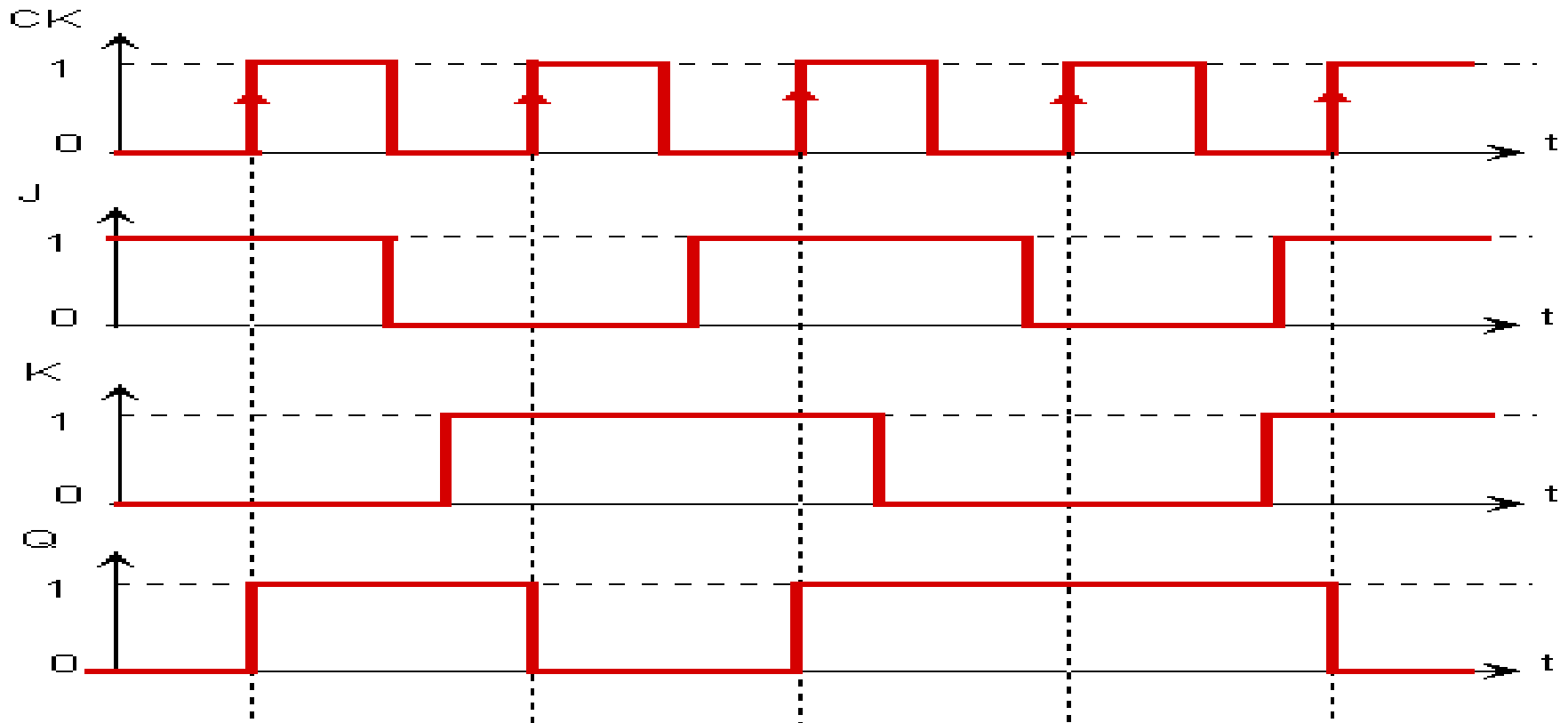




# Synchronisation sur niveau et sur front

## Synchronisation sur front

\* condition de départ  $Q = 0$ .





# Application des bistables

Les applications des bistables sont nombreuses. Nous allons examiner deux grandes familles: les **registres** et les générateurs de séquences et particulièrement les **compteurs/décompteurs**.

## Les registres

- ❖ Un registre est un ensemble de cellules mémoire de base
- ❖ Le nombre de bits du registre correspond au nombre de cellules mémoire (nombre de bascule D ou JK) du registre.
- ❖ On peut se représenter un registre comme un ensemble de mémoires élémentaires susceptibles de stocker chacune un bit
- ❖ Les données peuvent être écrites/lues en même temps (parallèle) ou une après l'autre (série).



# Les registres

- ❖ De la même façon la présentation des informations sur les sorties peut se faire soit en série soit parallèle.
- ❖ Les registres sont classées par :
  - Le nombre de bits.
  - Le mode de fonctionnement (unique ou multiple).

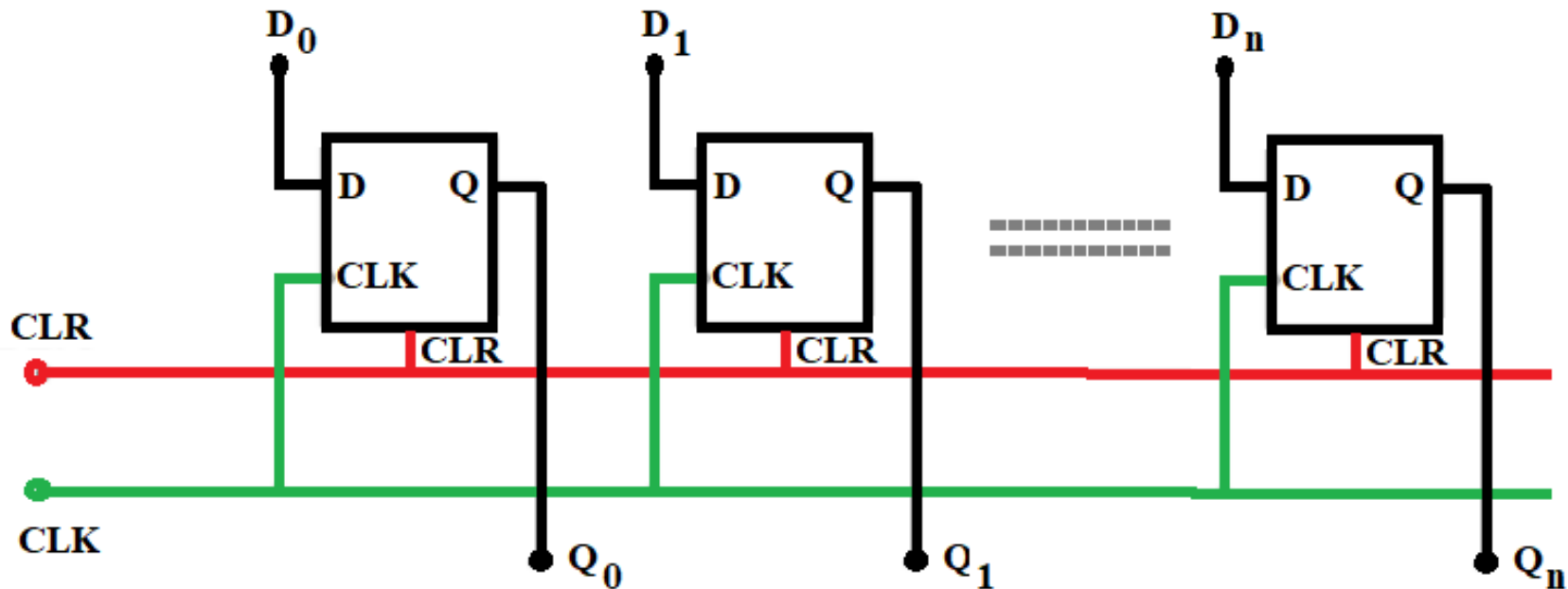
La classification des modes de fonctionnement est la suivante :

- ✓ Des registres à entrées parallèles et sorties parallèles : PIPO (Parallel IN-Parallel OUT).
- ✓ Des registres à entrées parallèles et sorties séries : PISO (Parallel IN-Serial OUT).
- ✓ Des registres à entrées séries et sorties parallèles : SIPO (Serial IN- Parallel OUT).
- ✓ Des registres à entrées séries et sorties séries : SISO (Serial IN- Serial OUT).

# Les registres

## 1. Registres de mémorisation tampon (Buffer)

Les registres de mémorisation ou registres tampon sont des registres de type parallèle-parallèle constitués de  $n$  bascules de type D commandées par une même horloge.

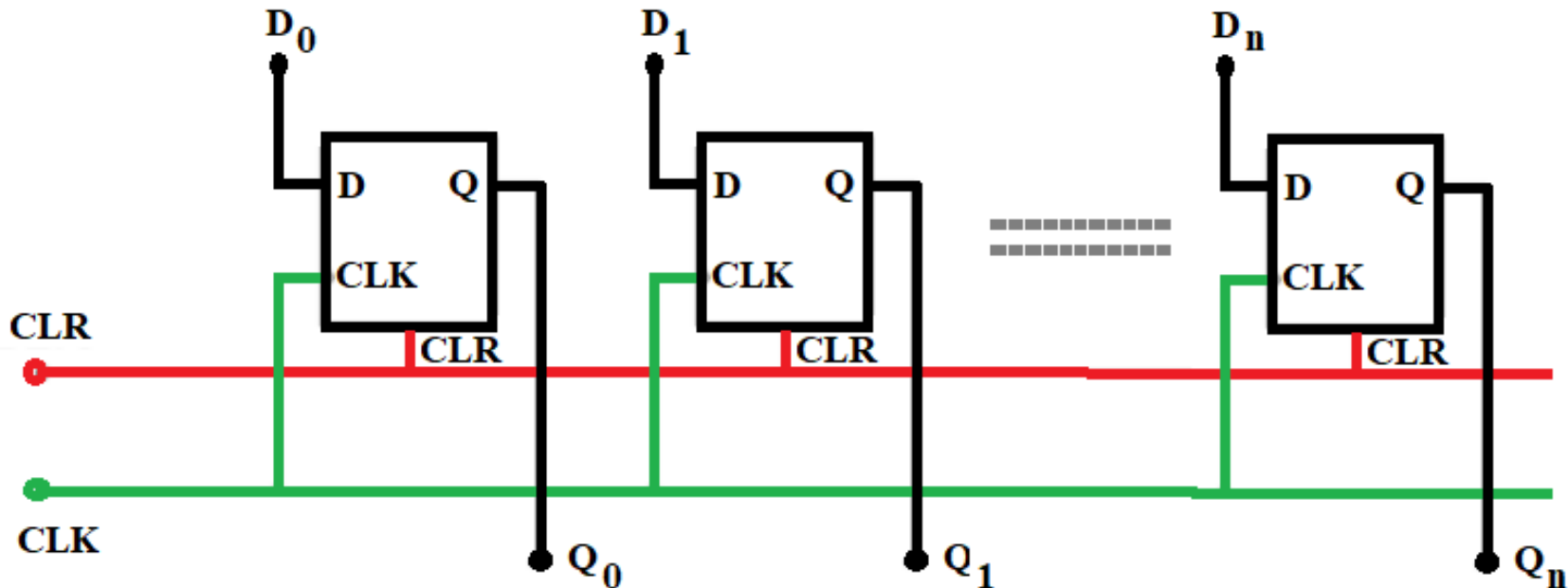


# Les registres

## Registres de mémorisation tampon (Buffer)

Au signal l'horloge (impulsion sur CLK) les entrées D sont recopiées sur les sorties  $Q_i$ .

Une entrée asynchrone CLR permet, de façon prioritaire, d'effacer le contenu du registre et d'écrire  $Q_i = 0$ .



# Les registres

## Registres de mémorisation tampon (Buffer)

Les  $n$  bascules sont chargées en parallèle et lues en parallèle en synchrone avec le signal d'écriture CLK. Ce type de registre est appelé aussi registre PIPO.

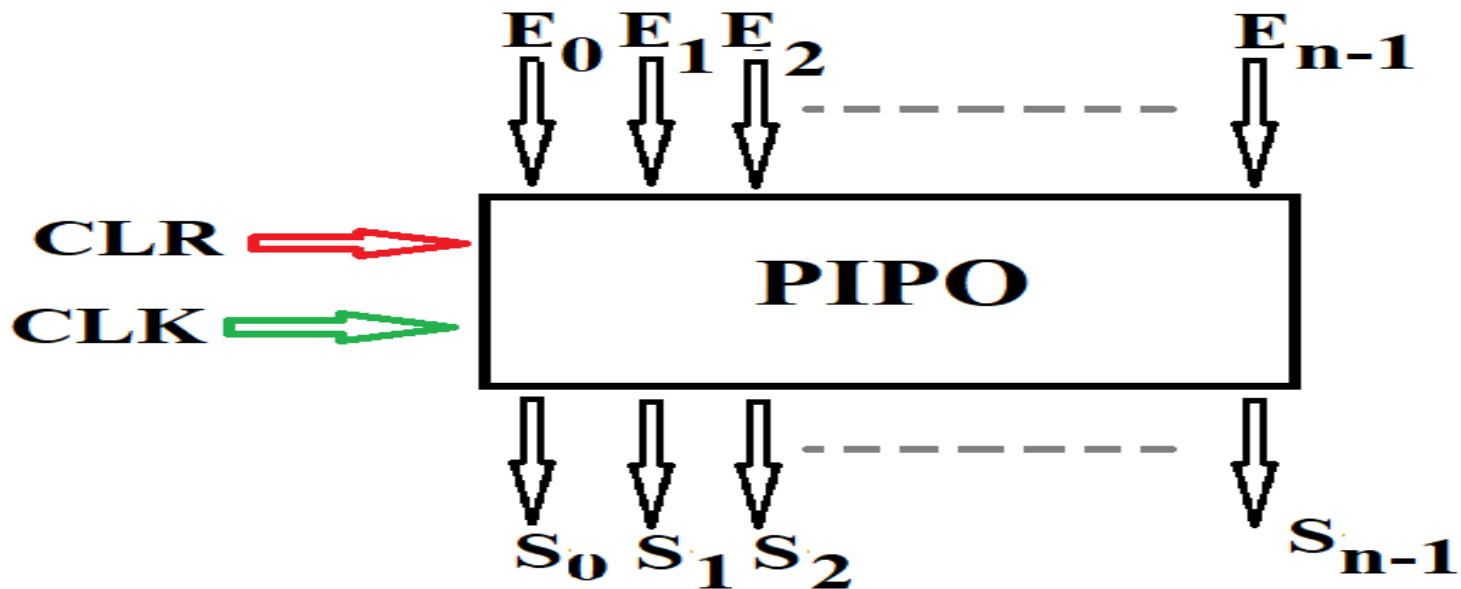


Schéma fonctionnel d'un registre PIPO

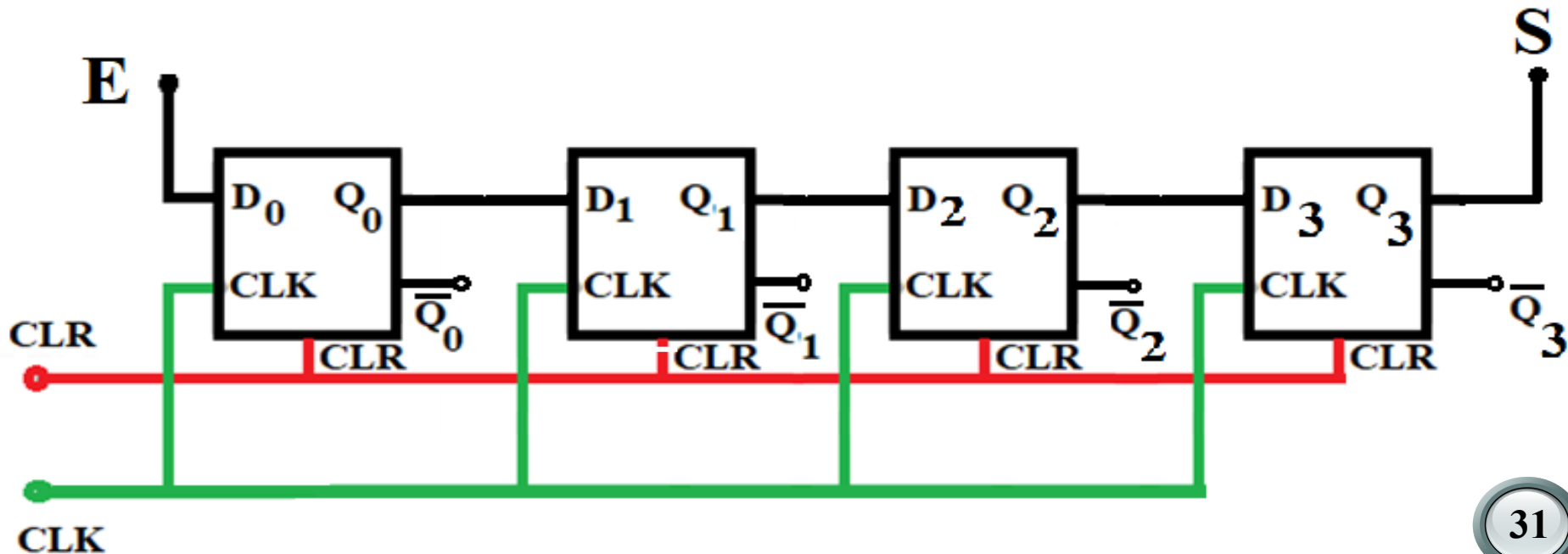


# Les registres

## 2. Les Registres à décalage

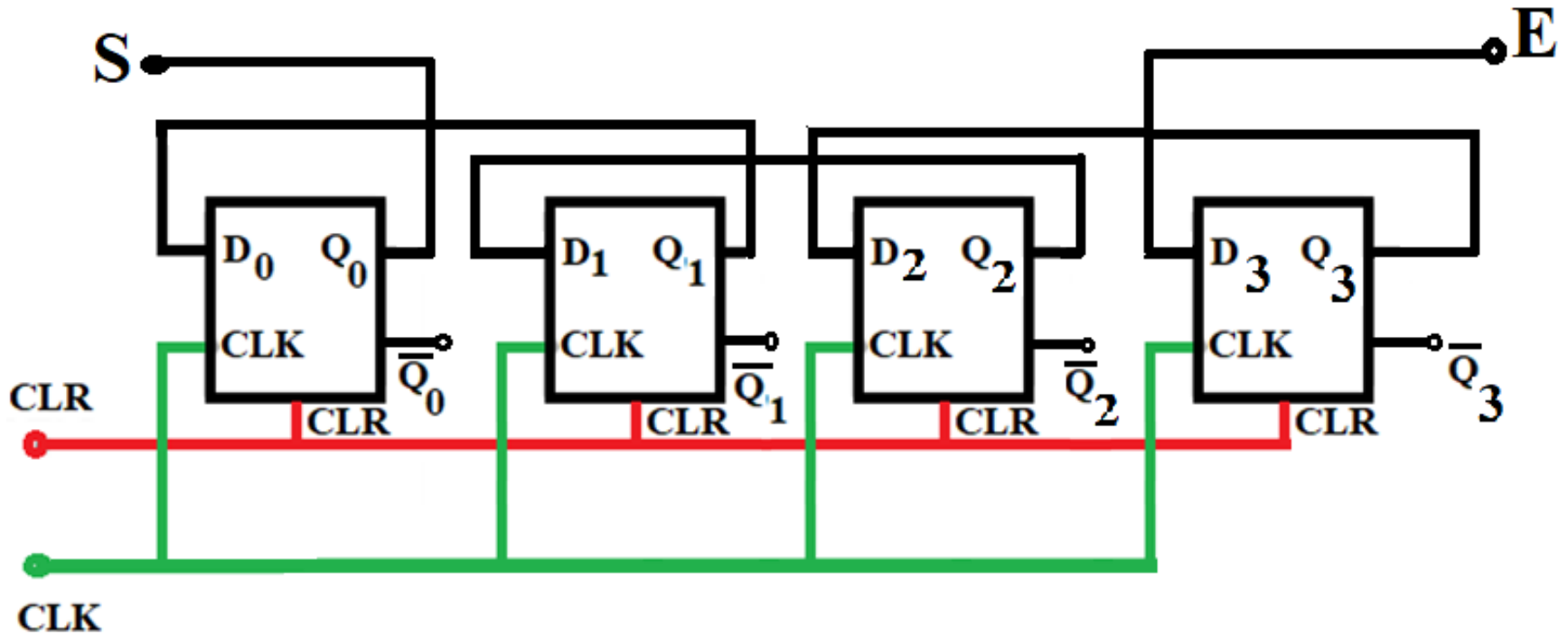
Les registres à décalage sont des registres de type série-série (SISO) ou série parallèle (SIPO), dans lesquels les informations sont décalées d'une bascule vers la suivante au rythme des impulsions d'une horloge

### 2.1 Décalage a droite



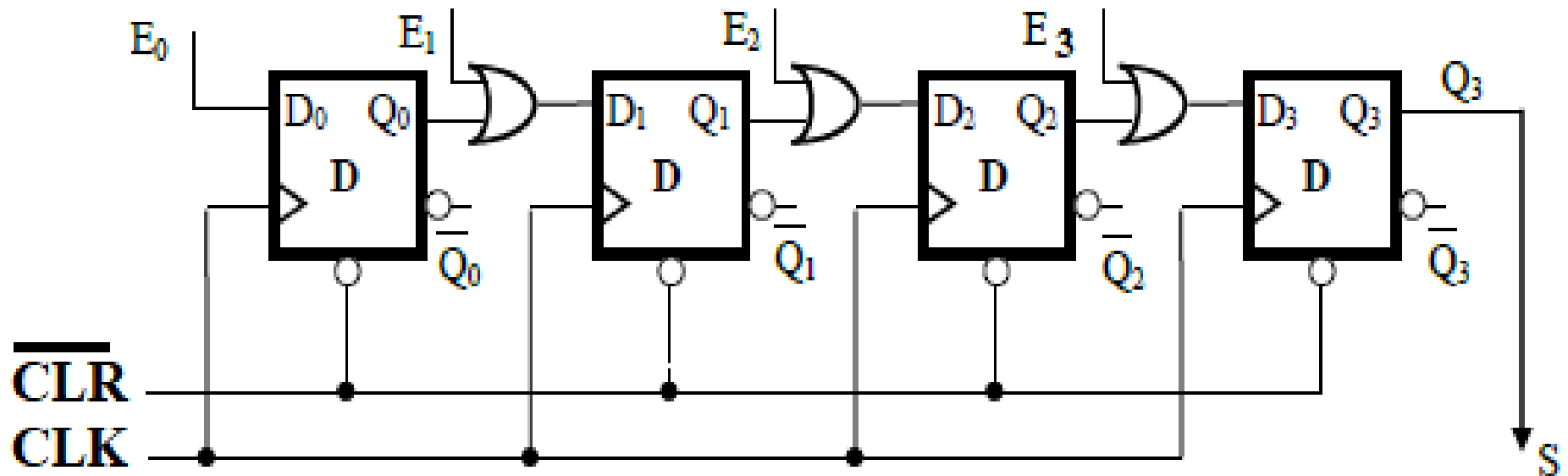
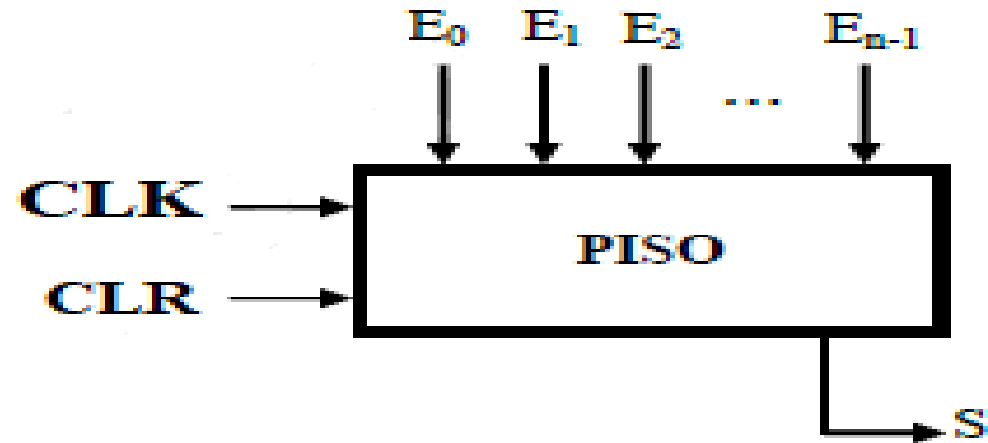
# Les registres

## 2.2 Décalage a gauche



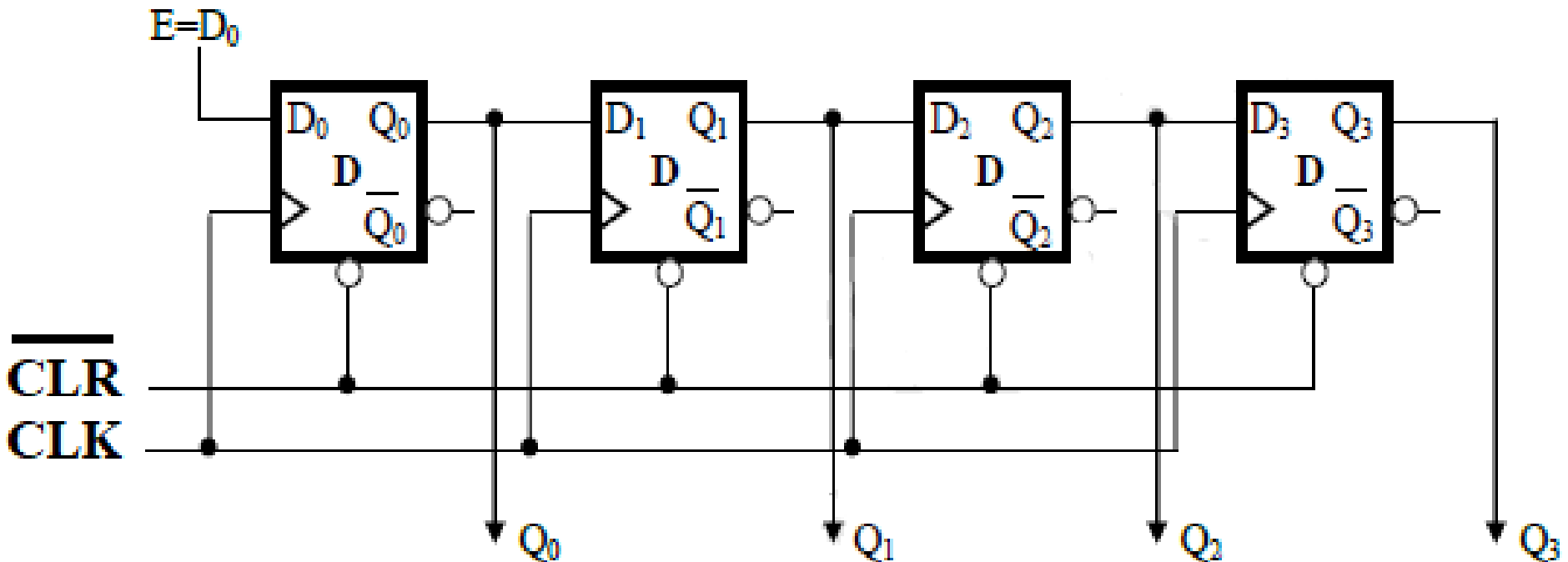
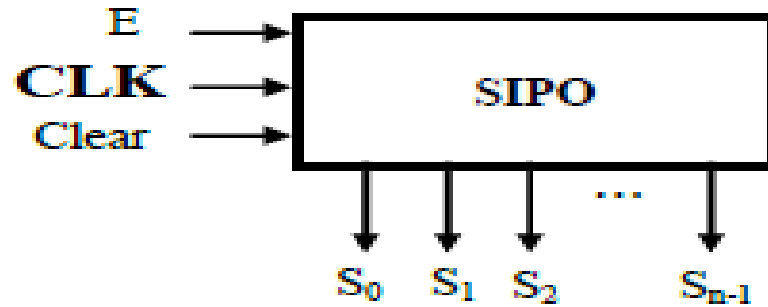
# Les registres

## 2.3 Registre PISO



# Les registres

## 2.4 Registre SIPO





# Les compteurs

- ❖ Les compteurs sont des ensembles des bascules montées en série ou en parallèle pour compter soit incrémenter ou décrémente, on distingue deux types : les compteurs Asynchrone et Synchrone.
- ❖ Un compteur modulo  $n$  peut compter de zéro jusqu'à  $(n-1)$ , tel que : Si  $n=8$  donc est compté de 0 à 7, et si  $n=10$  donc est compté de 0 à 9.
- ❖ L'élément de base des compteurs est une bascule à entrée d'horloge (bascule synchrone), soit de type D, T ou JK.
- ❖ Le modulo est le nombre d'état distincts occupés par un compteur avant son recyclage à l'état initial. Le nombre maximal d'états possibles, ou modulo maximal, d'un compteur est égal à  $2^n$ , où  $n$  représente le nombre des bascules dans le compteur.



# Les compteurs

## Exemples

### Compteur Asynchrone décimal (modulo 10)

Ce compteur est compté de 0 à 9, le nombre des bascules utilisées pour compter de 0 à 9 est déterminé par l'opération suivante:

$$2^0 = 1 < 10, \quad 2^1 = 2 < 10,$$

$$2^2 = 4 < 10, \quad 2^3 = 8 < 10$$

$$2^4 = 16 > 10$$

Donc le nombre des bascules utilisées au compteur décimal (modulo 10) est 4

compteur modulo 5 (càd compté de 0 à 4)

$$2^0 = 1 < 5, \quad 2^1 = 2 < 5,$$

$$2^2 = 4 < 5, \quad 2^3 = 8 > 5$$

Donc le nombre des bascules utilisées pour le compteur modulo 5 est trois (3).



# Les compteurs

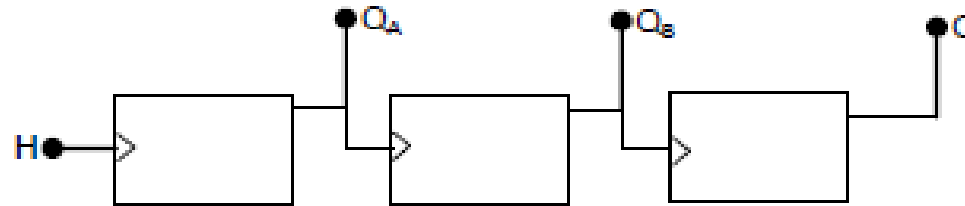
## 1. Compteurs Asynchrone

Le terme asynchrone signifie que les évènements ne possèdent aucune relation temporelle entre eux. Les bascules formant un compteur asynchrone ne changent pas d'état en même temps, car elles ne sont pas reliées au même signal d'horloge.

Ce type de compteur étant généralement d'une réalisation simple et présente l'inconvénient de générer des aléas de fonctionnement (retard de propagation).

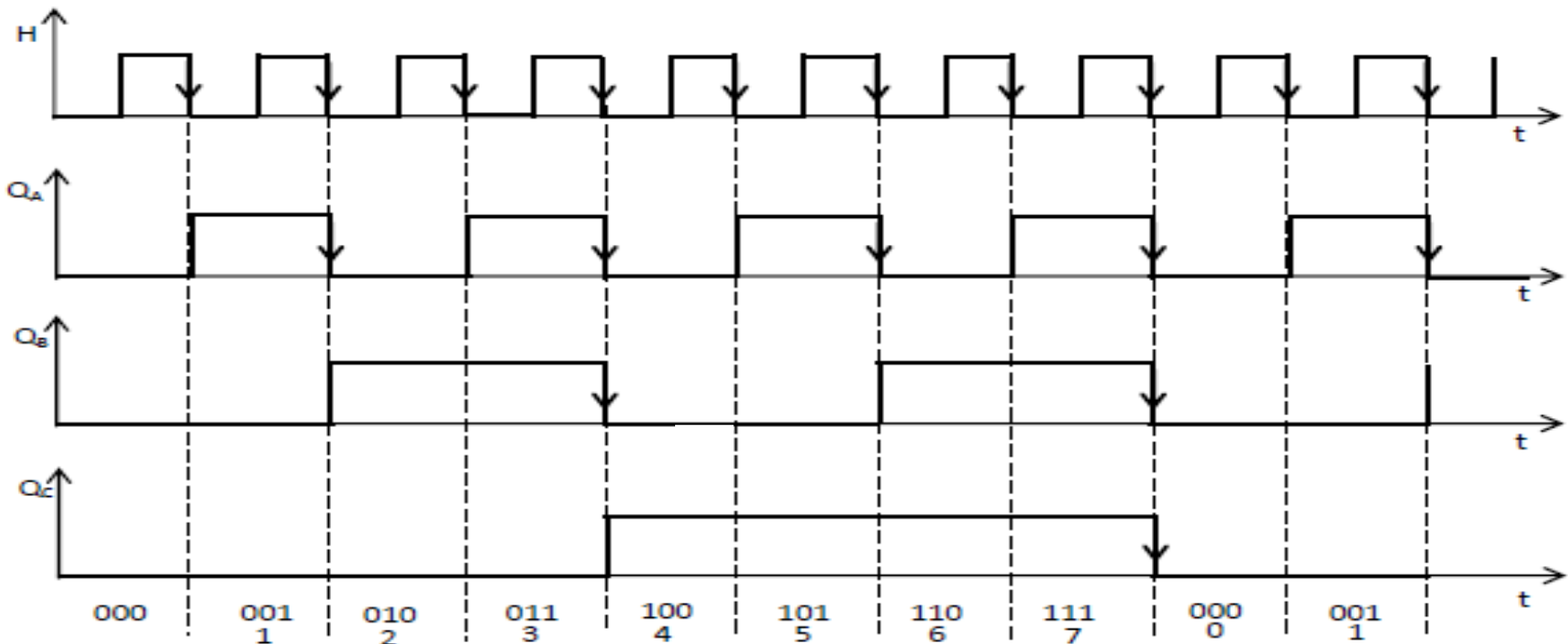
# Les compteurs

## 1. Compteurs Asynchrone



*Schéma d'un compteur Asynchrone*

**Chronogramme**

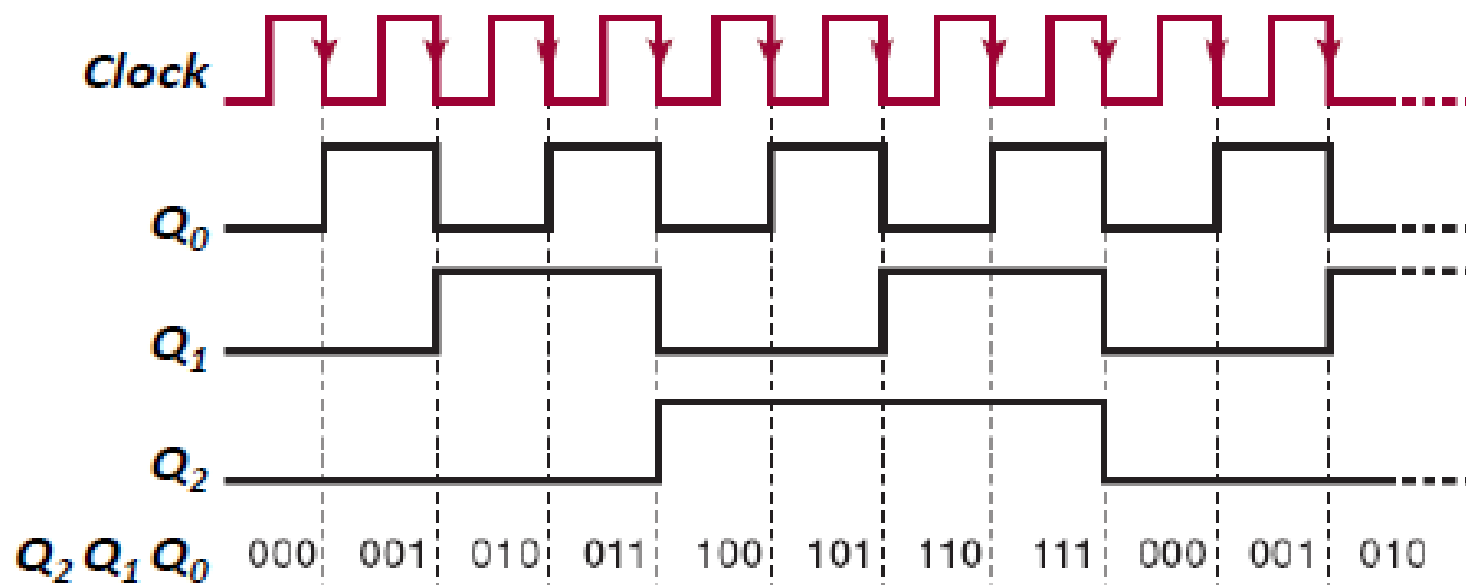
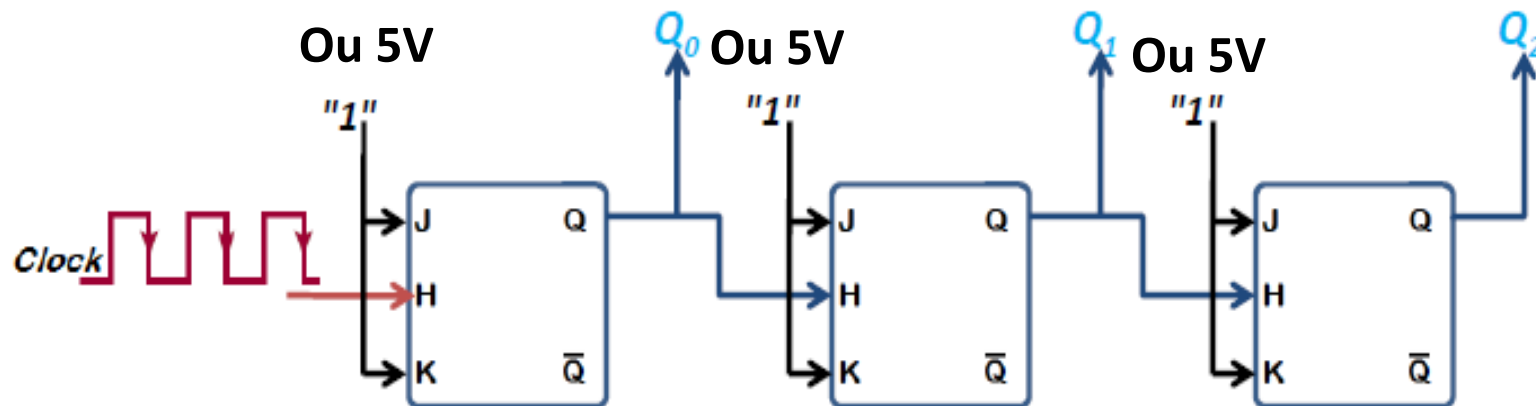




# Les compteurs

## 1. Compteurs Asynchrone

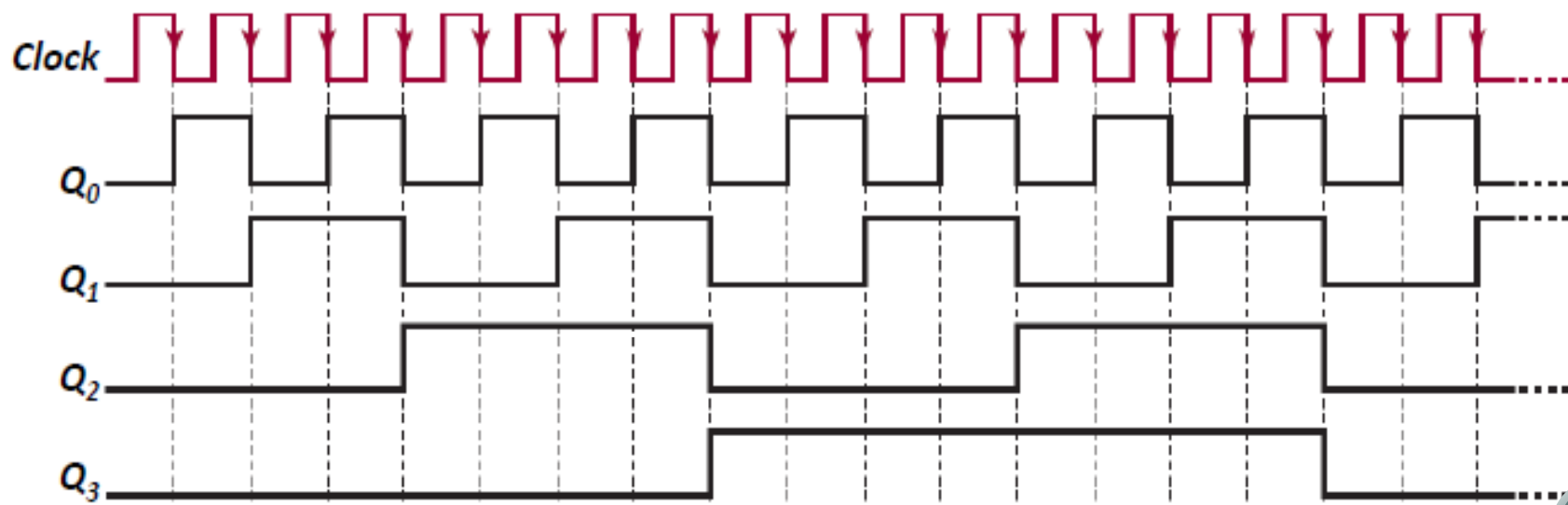
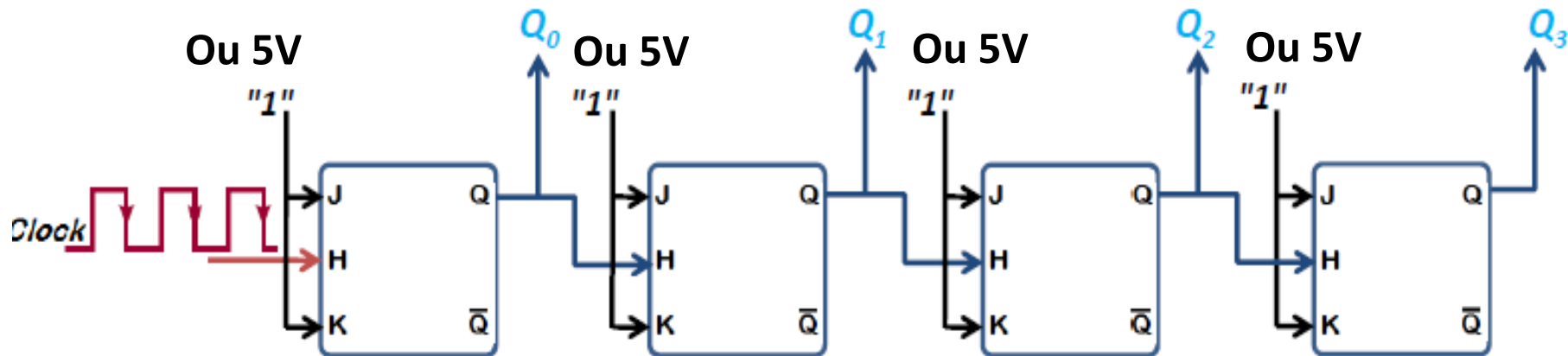
Compteur asynchrone modulo 8. (Le bistable JK est sensible sur front descendant de H).



# Les compteurs

## 1. Compteurs Asynchrone

Compteur asynchrone modulo 16. (Le bistable JK est sensible sur front descendant de H).

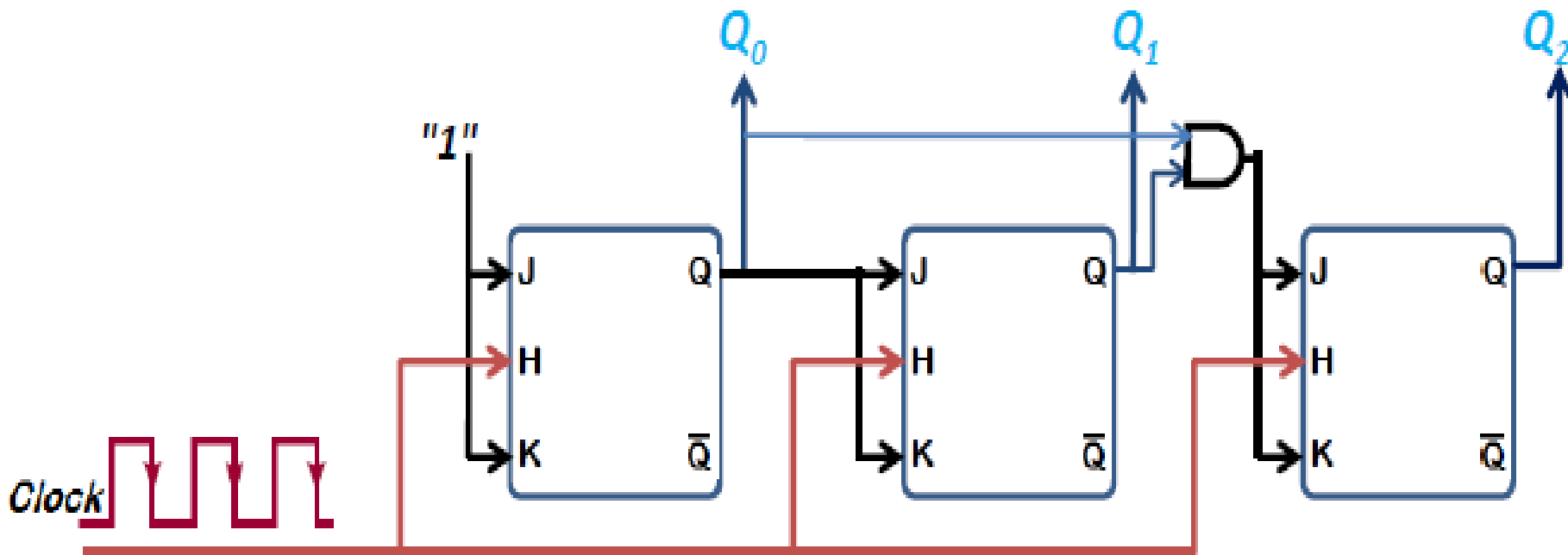


$Q_3 Q_2 Q_1 Q_0$  0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 1100 1101 1110 1111 0000 0001 0010

# Les compteurs

## 2. Compteurs synchrone

En termes de fonctionnement d'un compteur, le mot synchrone désigne que toutes les bascules du compteur sont synchronisées sur le même signal d'horloge. Le problème de retard de propagation donc résolu.

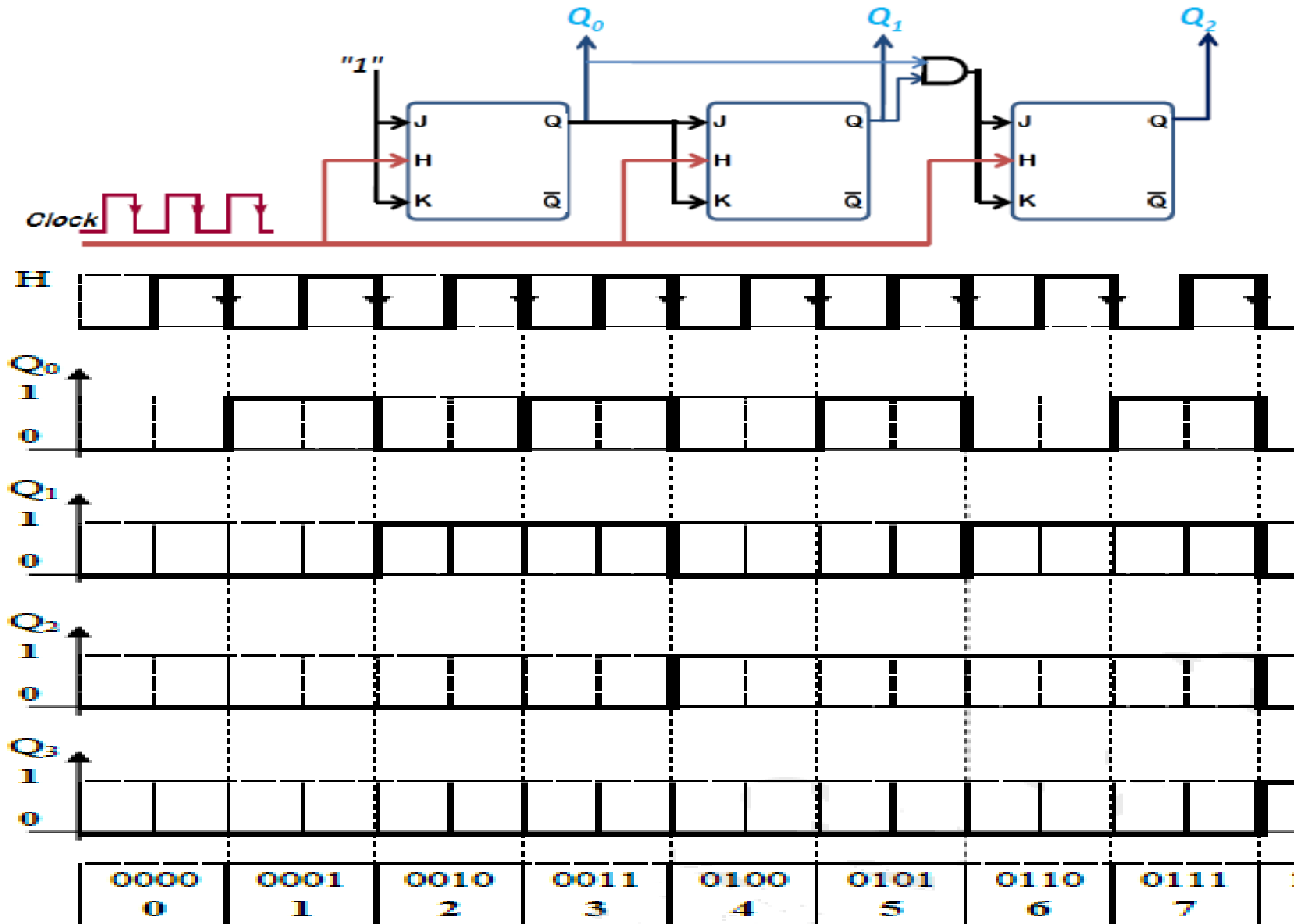


*Compteur synchrone modulo 8. (En utilisant le bistable JK).*



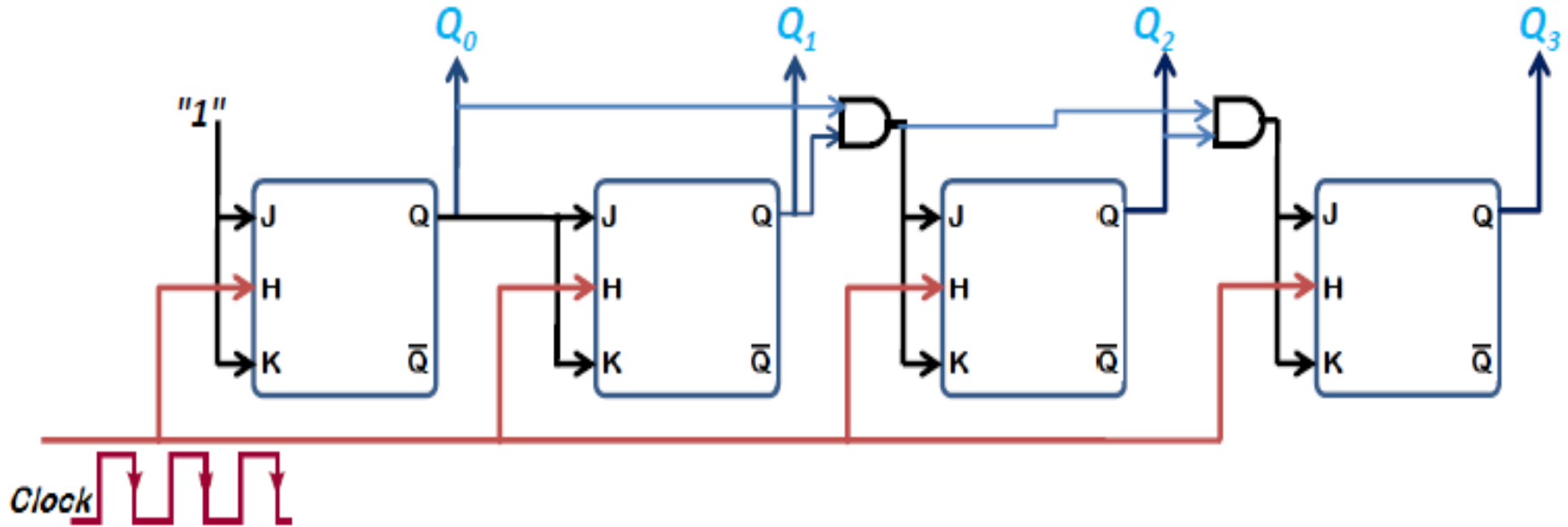
# Les compteurs

## Compteur synchrone modulo 8.



# Les compteurs

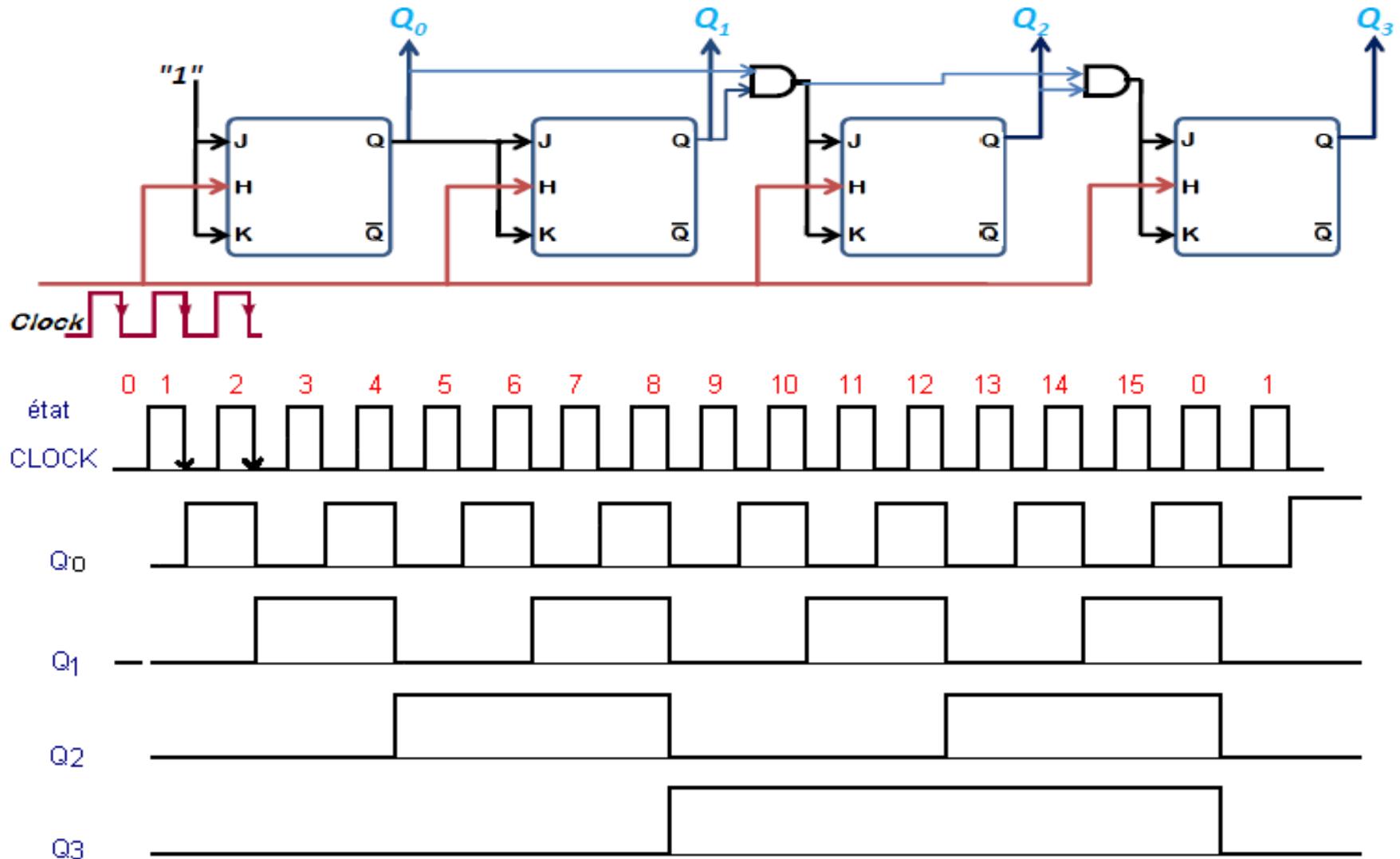
## Compteurs synchrone modulo 16



Le bistable JK est sensible sur front descendant de H

# Les compteurs

## Compteurs synchrone modulo 16



Chronogramme du compteur de module 16.

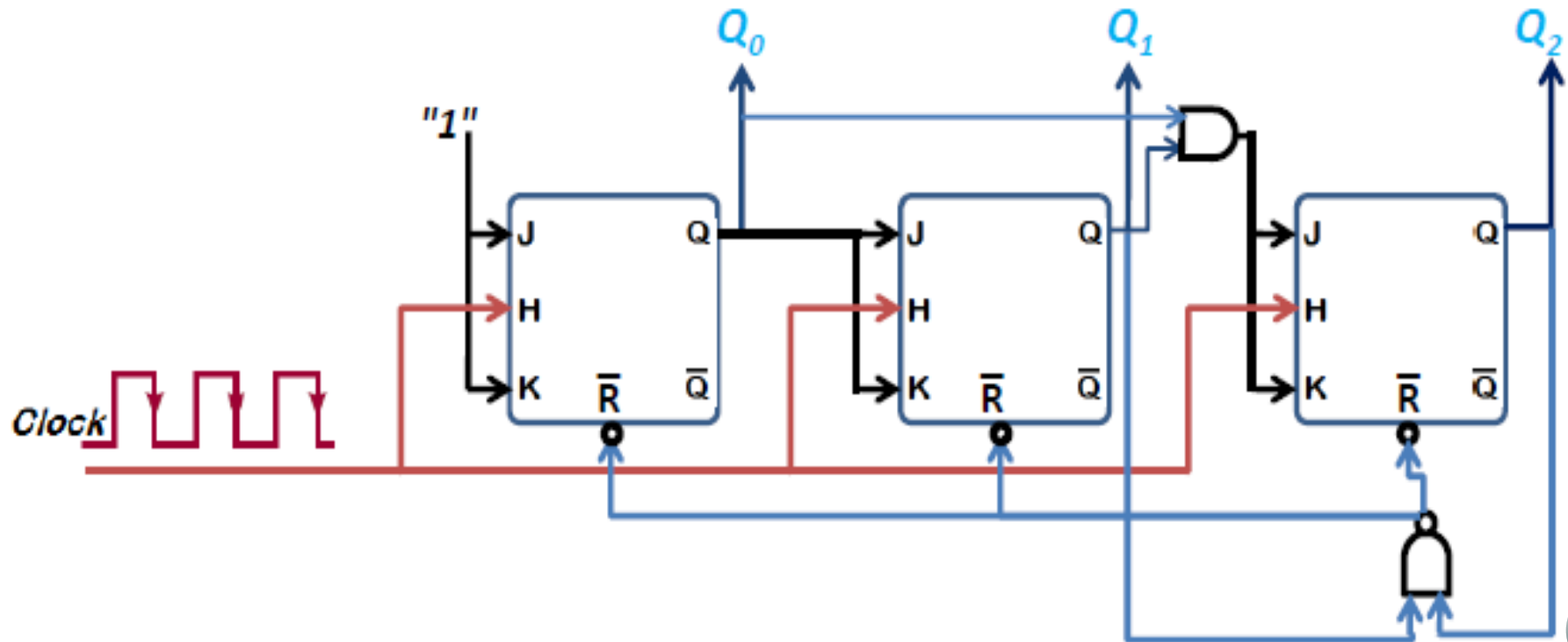


# Les compteurs

## Compteurs synchrone modulo 6

L'idée est de Prendre un compteur modulo 8 et le remettre à zéro dès qu'il atteint l'état 110 ( $Q_2 Q_1 Q_0$ ).

Lorsque  $Q_2 Q_1 = 11$  la porte NAND génère une sortie à 0 permettant  $Q_2 Q_1$  de remettre les bistables à l'état 000



# Les compteurs

## Compteurs synchrone modulo 10

L'idée est de Prendre un compteur modulo 16 et le remettre à zéro dès qu'il atteint l'état 1010 ( $Q_3 Q_2 Q_1 Q_0$ ).

Lorsque  $Q_3 Q_1 = 11$  la porte NAND génère une sortie à 0 permettant  $Q_3 Q_1$  de remettre les bistables à l'état 0000

