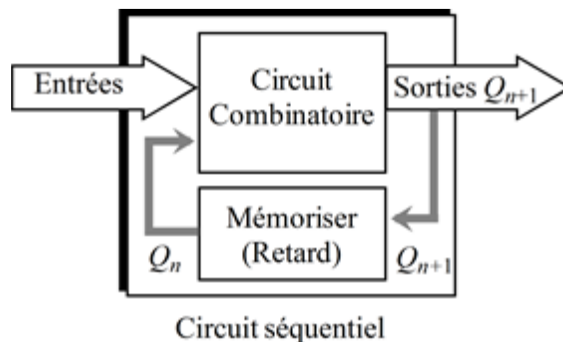


## Chapitre 5 : Les bascules

### 5-1 Introduction

On a vu que dans un système combinatoire les sorties dépendent uniquement des entrées, pour les mêmes entrées on a toujours les mêmes sorties. Dans un système séquentiel, les sorties dépendent non seulement des entrées mais aussi des sorties précédentes, donc pour les mêmes entrées on n'a pas toujours les mêmes sorties.

Tout circuit séquentiel est constitué d'un circuit combinatoire couplé à une mémoire comme indique par la figure suivante :



Dans la table de vérité on trouve, en plus des entrées, les sorties à l'état précédent.

$$Q_{n+1} = f(\text{Entrées}, Q_n)$$

On distingue deux types de systèmes logiques séquentiels :

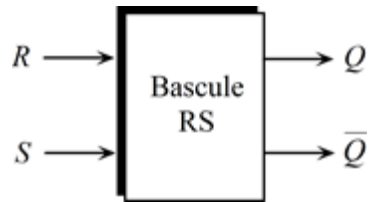
1. Les circuits séquentiels asynchrones, dans lesquels les sorties évoluent dès qu'il y a un changement sur l'une des entrées ;
2. Les circuits séquentiels synchrones, dans lesquels les sorties ne peuvent évoluer que si un signal d'horloge est actif.

Les bascules sont les circuits logiques de base de la logique séquentielle, elles possèdent deux sorties complémentaires  $Q$  et  $\overline{Q}$ . On note  $Q_{n+1}$  la sortie à l'état actuel et  $Q_n$  la sortie à l'état précédent (mémorisée). Il existe des bascules asynchrones (sans horloge) et des bascules synchrones (avec horloge).

### 5-2 Bascules asynchrones

#### 5 -2 -1 Bascule RS

Une bascule RS comporte deux entrées : Une entrée R (Reset) de mise à zéro et une entrée S (Set) de mise à un. Schématisée sous la forme suivante :



Son fonctionnement se résume ainsi :

R	S	$Q_{n+1}$	Etat
0	0	$Q_n$	maintient (état mémoire)
0	1	1	mise à 1
1	0	0	mise à 0
1	1	$\phi$	indéterminé (cas interdit)

La table de vérité est donc :

R	S	$Q_n$	$Q_{n+1}$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	$\phi$
1	1	1	$\phi$

Les équations logiques correspondant,

– En utilisant les mintermes :

$S \ Q_n$	00	01	11	10
R				
0	0	1	1	1
1	0	0	$\phi$	$\phi$

→ S

↓ R  $Q_n$

$$Q_{n+1} = S + R \cdot Q_n$$

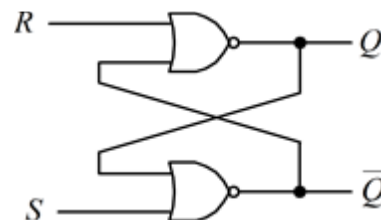
– En utilisant les maxtermes :

$S \ Q_n$	00	01	11	10
R				
0	0	1	1	1
1	0	0	$\phi$	$\phi$

$$Q_{n+1} = \overline{R} \cdot (S + Q_n)$$

Le logigramme avec uniquement les portes NOR :

$$\begin{aligned} Q_{n+1} &= \overline{\overline{R} \cdot (S + Q_n)} \\ &= \overline{\overline{R} \cdot (S + Q_n)} \\ &= \overline{\overline{R}} + \overline{(S + Q_n)} \\ &= \overline{R} + \overline{(S + Q_n)} \end{aligned}$$



### 5-2-2 Bascule JK

La bascule JK vient prendre en charge le cas indéterminé de la bascule RS, son fonctionnement est défini par la table suivante:

J	K	$Q_{n+1}$	Etat
0	0	$Q_n$	maintient (état mémoire)
0	1	0	mise à 0
1	0	1	mise à 1
1	1	$\overline{Q_n}$	basculement (changement)

Sa table de vérité est alors :

J	K	$Q_n$	$Q_{n+1}$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

L'équation logique simplifiée par le tableau de Karnaugh en prenant les 1 :

	K $Q_n$	00	01	11	10
J					
0		0	1	0	0
1		1	1	0	1

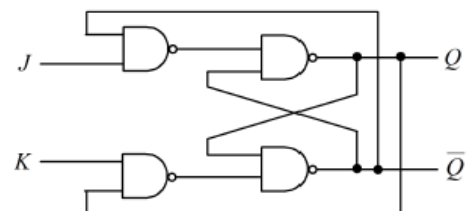
$$Q_{n+1} = J \cdot \overline{Q_n} + \overline{K} \cdot Q_n$$

Si on prend les 0, on aura :

$$Q_{n+1} = (J + Q_n) \cdot (\overline{K} + \overline{Q_n}) \Rightarrow \overline{Q_{n+1}} = \overline{J + Q_n} \cdot \overline{\overline{K} + \overline{Q_n}} = \overline{J} \cdot \overline{Q_n} + K \cdot Q_n$$

Le logigramme utilisant seulement les portes NAND :

$$\begin{aligned} Q_{n+1} &= J \cdot \overline{Q_n} + \overline{K} \cdot Q_n \\ &= J \cdot \overline{Q_n} + \overline{K} \cdot Q_n + Q_n \cdot \overline{Q_n} \\ &= \overline{\overline{J \cdot \overline{Q_n} + Q_n \cdot (\overline{K} + \overline{Q_n})}} \\ &= \overline{\overline{J \cdot \overline{Q_n} + Q_n \cdot (\overline{K} \cdot Q_n)}} \\ &= J \cdot \overline{Q_n} \cdot Q_n \cdot (\overline{K} \cdot Q_n) \end{aligned}$$



### 5 -2-3 Bascule D

La bascule D possède une seule entrée. Son principe de fonctionnement est donné par les tables suivantes :

D	$Q_n$	$Q_{n+1}$
0	0	0
0	1	0
1	0	1
1	1	1

 $\Rightarrow$ 

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

 $\Rightarrow Q_{n+1} = D$ 

Le logigramme correspondant :

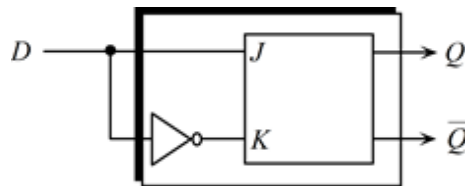
$$Q_{n+1} = D \cdot (Q_n + \overline{Q_n})$$

$$= D Q_n + D \overline{Q_n}$$

Par identification à l'équation de la bascule JK :

$$Q_{n+1} = J \cdot \overline{Q_n} + \overline{K} \cdot Q_n \Rightarrow \begin{cases} J = D \\ K = \overline{D} \end{cases}$$

Par suite, le schéma logique de la bascule D est le suivant :



Sous cette forme, la bascule D n'a pas grand intérêt puisqu'il s'agit d'un bloc fonctionnel qui ne fait que recopier son entrée, en permanence. On verra plus loin que son utilité apparaît avec le signal "d'horloge" (version synchrone).

### 5 -2 - 4 Bascule T

La bascule T tire son nom du terme anglais 'toggle'. Si son entrée T est active, elle bascule à chaque impulsion d'horloge d'où son nom. Si son entrée T est inactive, elle conserve son état, comme indiqué par la table de fonctionnement suivante :

T	$Q_n$	$Q_{n+1}$	Etat
0	0	0	maintient
0	1	1	maintient
1	0	1	basculement
1	1	0	basculement

 $\rightarrow$ 

T	$Q_{n+1}$
0	$Q_n$
1	$\overline{Q_n}$

L'équation caractéristique est :

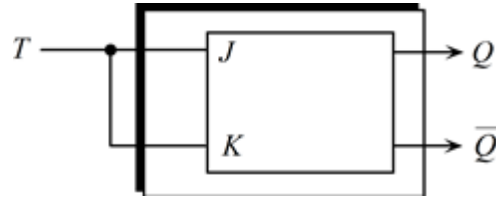
$$Q_{n+1} = \overline{T} \cdot Q_n + T \cdot \overline{Q_n}$$

$$Q_{n+1} = T \oplus Q_n$$

Par identification à l'équation de la bascule JK, on obtient :

$$Q_{n+1} = J \cdot \bar{Q}_n + \bar{K} \cdot Q_n \longrightarrow J = T \text{ et } K = T$$

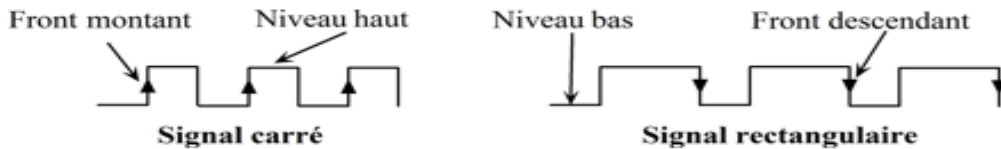
Le logigramme de la bascule T est :



### 5-3 Horloge et bascules synchrones

#### 5-3-1 Horloge

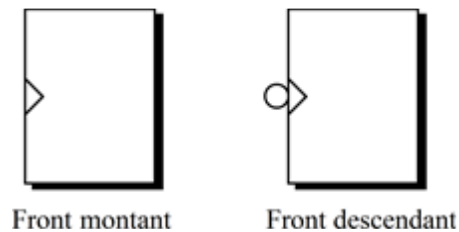
L'horloge génère des impulsions périodiques, qui se présentent sous forme de signaux carrés ou rectangulaires.



Les bascules sont conçues pour changer d'état, soit sur front montant, soit sur front descendant du signal d'horloge, et restent stables entre deux impulsions successives.

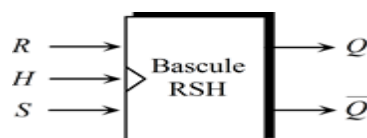
- Le passage du niveau bas (état 0) au niveau haut (état 1) est appelé : front montant ;
- Le passage du niveau haut au niveau bas est appelé : front descendant.

La représentation de l'entrée d'horloge est telle qu'illustrée par la figure suivante :



#### 5-3-2 Bascule RS synchrone (RST ou RSH)

On rajoute une entrée d'horloge H (validation) à la bascule RS. Schématiser pour le cas front montant sous la forme suivante :



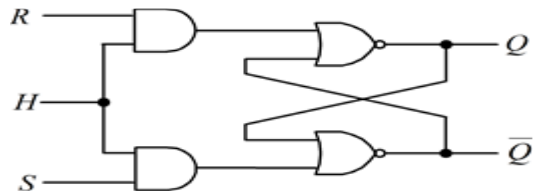
Pour son fonctionnement :

- Si  $H=0 \forall R$  et  $\forall S \Rightarrow Q_{n+1} = Q_n$  mémorisation ( $\forall$  quoi qu'il soit) ;
- Si  $H=1 \Rightarrow$  retournement au fonctionnement normal de la bascule RS.

Son fonctionnement se résume ainsi :

H	R	S	$Q_{n+1}$	Etat
0	$\forall$	$\forall$	$Q_n$	maintient (mémorisation)
1	0	0	$Q_n$	maintient (mémorisation)
	0	1	1	mise à 1
	1	0	0	mise à 1
	1	1	$\emptyset$	indéterminé (cas interdit)

Le logigramme correspondant

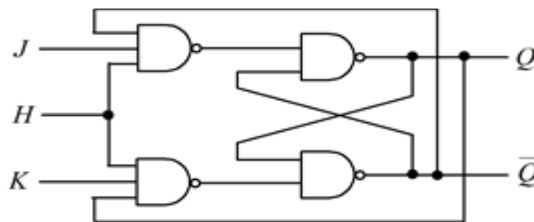


### 5-3-3 Bascule JK synchrone (JKT ou JKH)

De même on peut avoir une bascule JK qui fonctionne avec un signal d'horloge. Pour son fonctionnement :

- Si  $H=0 \forall J$  et  $\forall K \Rightarrow Q_{n+1} = Q_n$  mémorisation ;
- Si  $H=1 \Rightarrow$  retournement au fonctionnement normal de la bascule JK.

Le logigramme de JKH utilisant seulement les portes NAND est le même que celui de JK multiplié par l'entrée  $H$ , comme l'indique le schéma suivant :



La limitation de la JKT est quand  $J = 1$  et  $K = 1$  : La sortie oscille en 0 et 1 pendant toute la durée de l'état haut du signal d'horloge, pour cela on utilise une bascule JK avec d'enclenchement sur front (montant ou descendant).

Une autre solution consiste à utiliser une bascule JK Maître -Esclave.

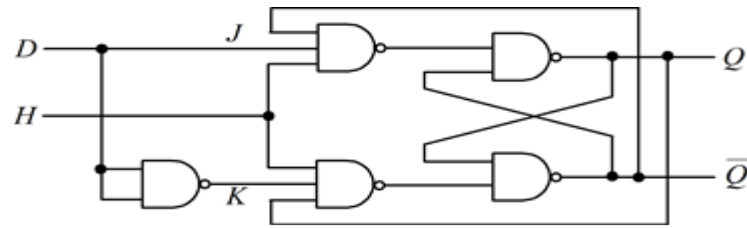
### 5-3-4 Bascule D à verrou (D latch)

La bascule recopie l'entrée  $D$  en sortie  $Q$  quand l'horloge est active  $H = 1$ . Quand l'horloge est inactive  $H = 0$ , la bascule garde l'état précédent. Sur niveau 1 (haut) on peut écrire :

H	$Q_{n+1}$
0	$Q_n$
1	D

 $\longrightarrow Q_{n+1} = \overline{H} \cdot Q_n + H \cdot D$

Le schéma logique de la bascule « D latch » à partir d'une bascule JKH (utilisant les portes NAND uniquement) est le suivant :

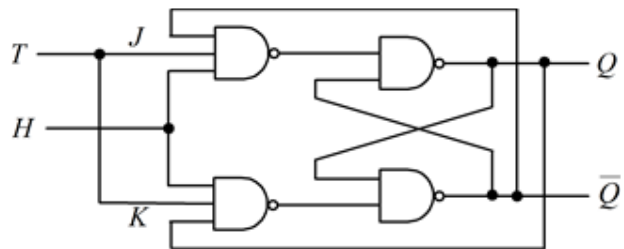


**Remarque :** La bascule « D latch » est très utilisée dans les compteurs synchrones.

### 5-3-5 Bascule T synchrone

En rajoutant un signal d'horloge à la bascule T, la sortie bascule (change d'état) dans le cas où l'horloge est active  $H = 1$  ainsi que  $T = 1$ .

Le logigramme de la bascule T synchrone à partir d'une bascule JKH (avec des portes NAND uniquement) est :



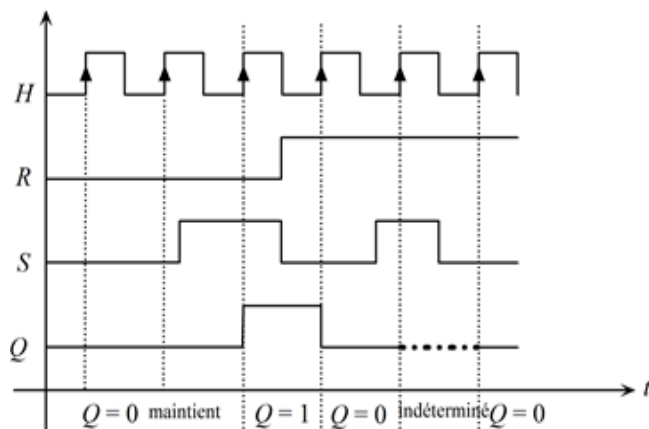
### 5-4 Synchronisation sur front et exemples de chronogrammes

Les bascules synchrones sur front changent d'état uniquement sur un front du signal d'horloge ; en dehors de ces fronts, elle fonctionne en mémoire.

Ce mode de fonctionnement protège d'éventuels parasites sur les entrées car les entrées ne sont prises en compte que pendant la durée d'un front, qui est très courte.

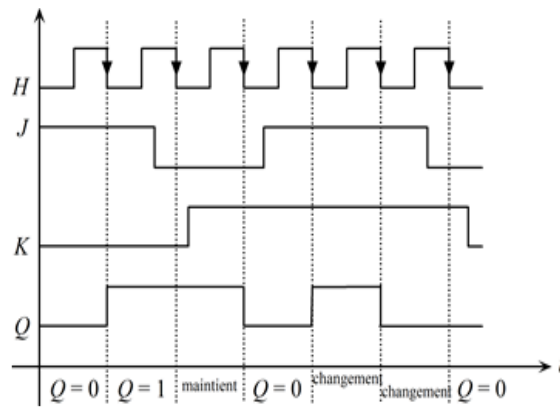
**Exemple 1 :**

Le chronogramme de fonctionnement d'une bascule RSH sur front montant, avec à l'état initial  $Q = 0$  est illustré par la figure suivante :



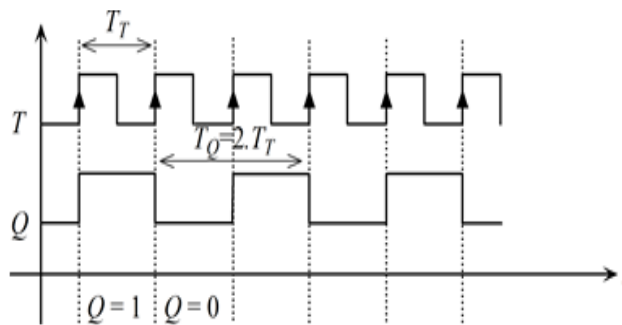
**Exemple 2 :**

Le chronogramme suivant est un exemple de fonctionnement d'une bascule JKH sur front descendant, avec au départ  $Q = 0$  :



**Exemple 3 :**

Le chronogramme de fonctionnement d'une bascule T sur front montant, avec initialement  $Q = 0$  :



On remarque que la fréquence du signal en sortie ( $Q$ ) est divisée par deux par rapport à celle de la bascule T ( $f_Q = f_T / 2$ ), puisque la période de  $Q$  représente le double de la période de la bascule T ( $T_Q = 2 \cdot f_T$ ).

**5-5 Détection des fronts**

Le circuit suivant permet de générer une impulsion de courte durée, par exploitation du temps de propagation des portes logiques :

