

# Logique séquentielle

Chapitre IV

# Electronique numérique 2

## Prérequis

Electronique numérique 1

## Volume horaire

CM, TD et TP

## Notation

Contrôle continu : 1 contrôle + TP

## Contact

TD1 : Maha Ben Rhouma ([maha.ben-rhouma@univ-eiffel.fr](mailto:maha.ben-rhouma@univ-eiffel.fr) Copernic : 2B035)

TD2 : Benoit Poussot ([benoit.poussot@univ-eiffel.fr](mailto:benoit.poussot@univ-eiffel.fr) Copernic: 2B025)

TD3 : Maha Ben Rhouma ([maha.ben-rhouma@univ-eiffel.fr](mailto:maha.ben-rhouma@univ-eiffel.fr) Copernic : 2B035)

TD4 : Shermila Mostarshedi ([shermila.mostarshedi@univ-eiffel.fr](mailto:shermila.mostarshedi@univ-eiffel.fr) Copernic: 2B025)

TD5 : Mohammed Elliq ([m.elliq@ens-louis-lumiere.fr](mailto:m.elliq@ens-louis-lumiere.fr))

# Plan

- **Introduction**
  - ◆ Séquentiel et combinatoire
  - ◆ Synchrone et asynchrone
- **Bascules**
  - ◆ Bascules RS
    - ▶ Analyse temporelle
    - ▶ Analyse statique
  - ◆ Bascule RST
  - ◆ Bascule JK
    - ▶ Synchrone
    - ▶ Avec des entrées asynchrones
  - ◆ Bascule D
    - ▶ Synchrone
    - ▶ A verrouillage
- **Compteurs et décompteurs**
  - ◆ synchrones et asynchrones
- **Registres**

# Séquentiel et combinatoire

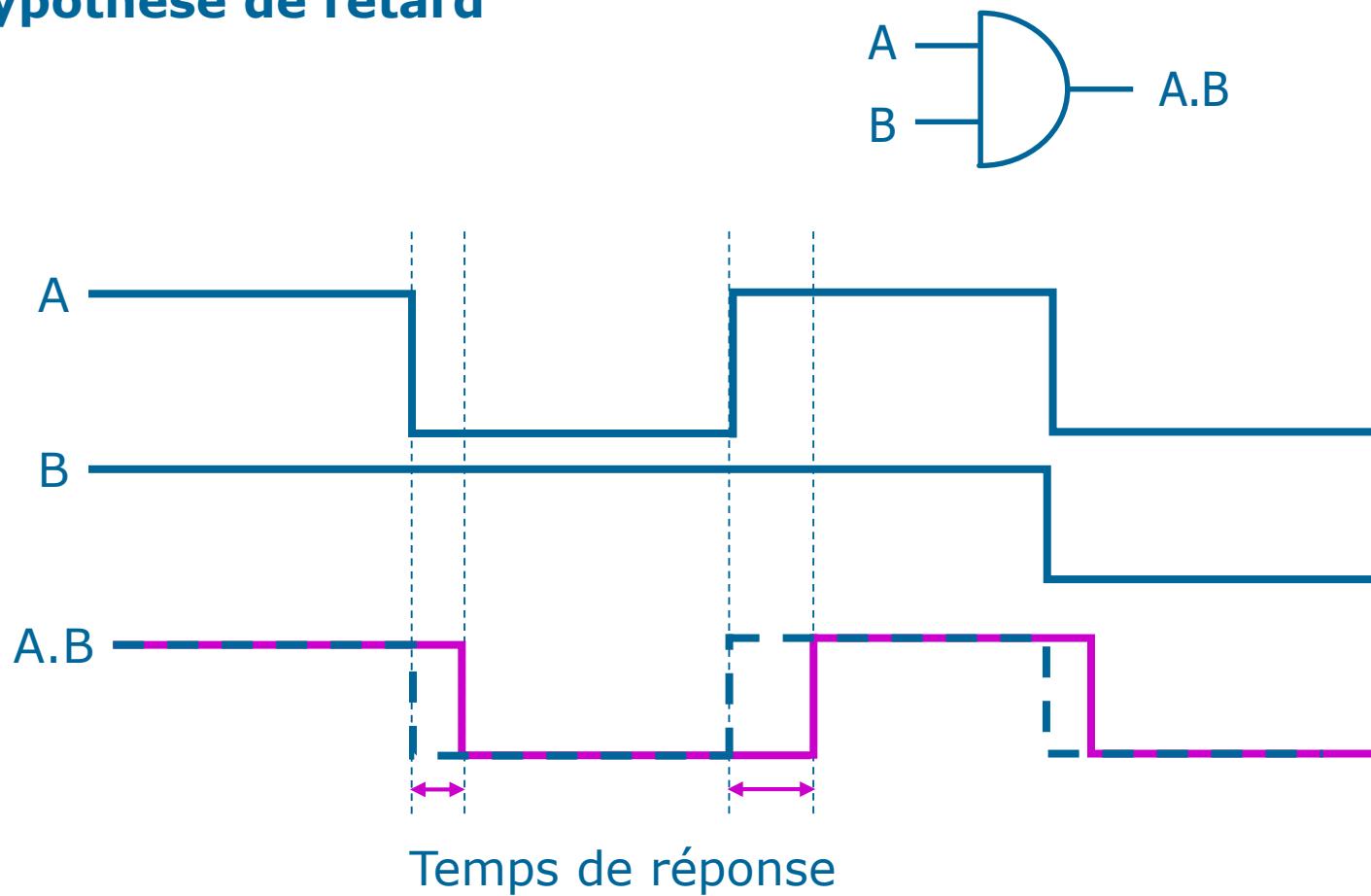
- **Logique combinatoire :**
  - ◆ La sortie à l'instant  $t$  ne dépend que du vecteur d'entrées à l'instant  $t$
  - ◆ L'état du système ne dépend pas de l'état interne du système
  - ◆ Pas de mémorisation de l'état du système
- **Exemples des systèmes plus complexes :**
  - ◆ Un appui sur un bouton allume la lumière, un deuxième appui sur le même bouton l'éteint
    - ▶ Le système sait si c'est la première ou la deuxième fois qu'on appuie sur le bouton.
  - ◆ Une impulsion démarre les essuie-glaces qui s'arrêtent en position de repos
    - ▶ Le système sait quand les essuie-glaces doivent s'arrêter.
  - ◆ Même vecteur d'entrées → 2 états différents dans la sortie : non réalisable en combinatoire
  - ◆ Le système dispose d'autres informations.
  - ◆ Le système connaît lui-même son état courant.

# Séquentiel et combinatoire

- **Deux hypothèses:**
  - Le temps nécessaire à l'établissement de la valeur de sortie d'une porte logique est vu comme un simple retard
  - La connaissance de l'évolution fine du niveau de signal logique n'est pas nécessaire
- **Dans ces conditions, nous allons pouvoir remplacer les variables d'une expression booléenne par des fonctions du temps.**
- **Dans la majorité des circuits numériques, on trouve à la fois des circuits combinatoires et des dispositifs à mémoire**

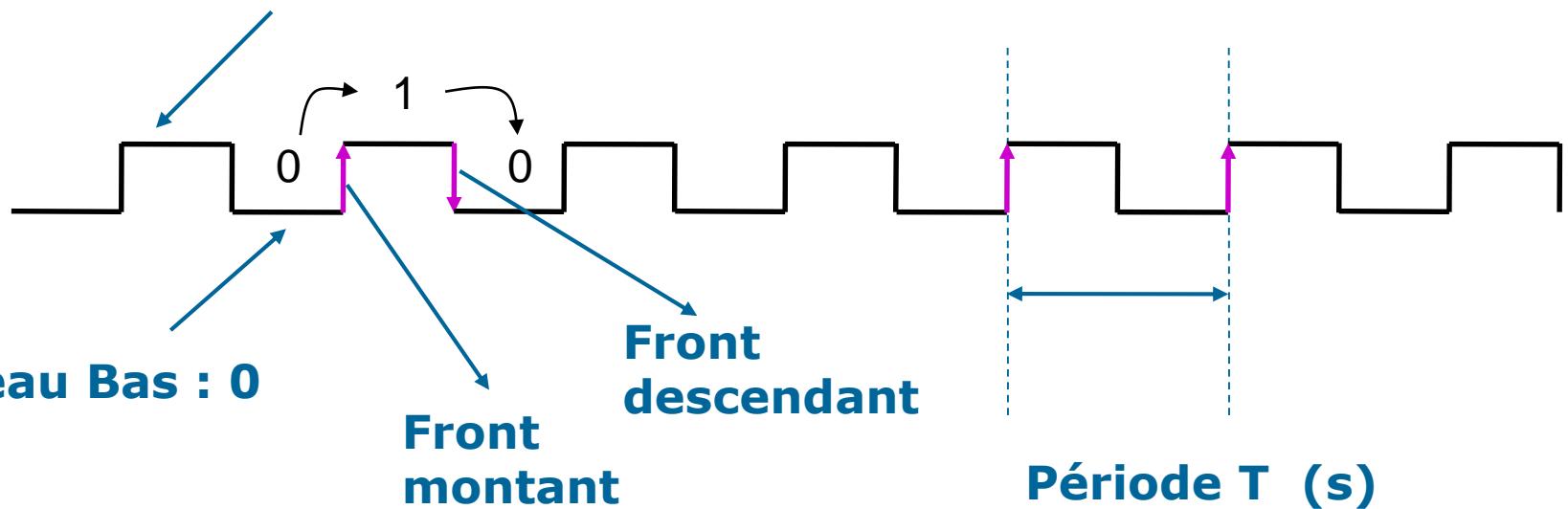
# Temps de réponse des portes logiques

- Hypothèse de retard



# Horloge

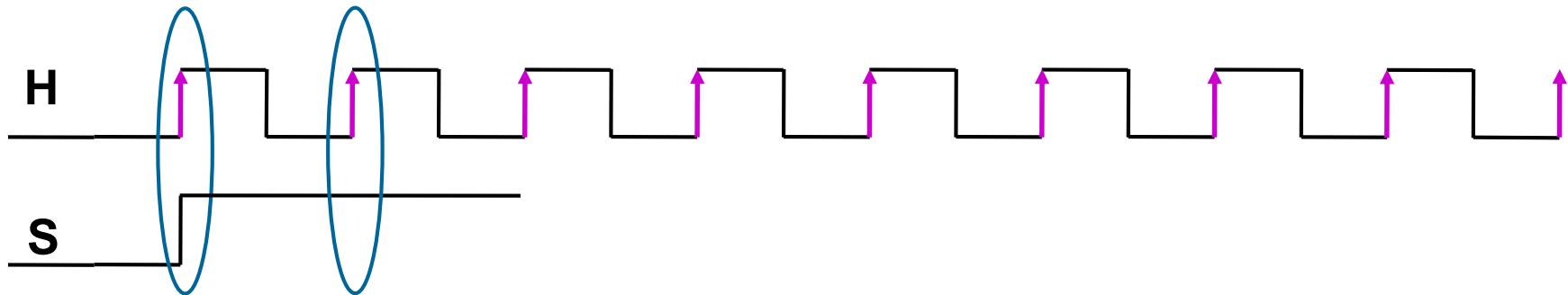
Niveau Haut: 1



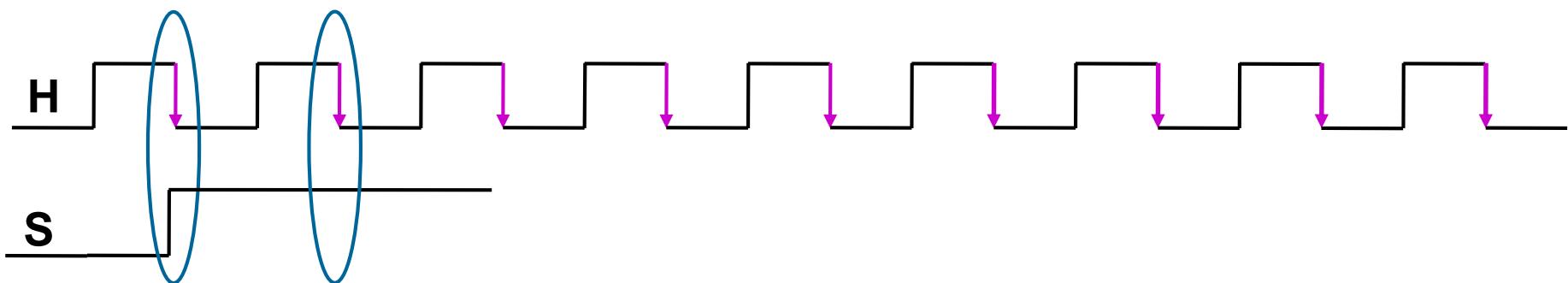
Fréquence  $f = \frac{1}{T}$

# Horloge

**Synchronisation sur front montant**

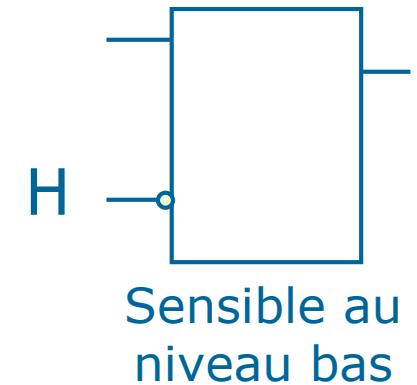
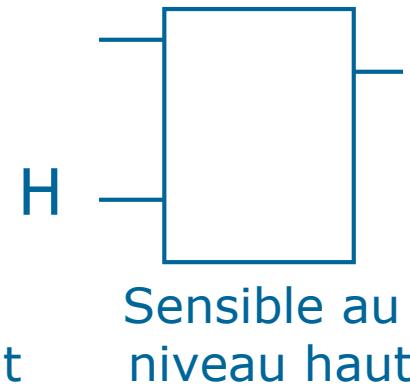
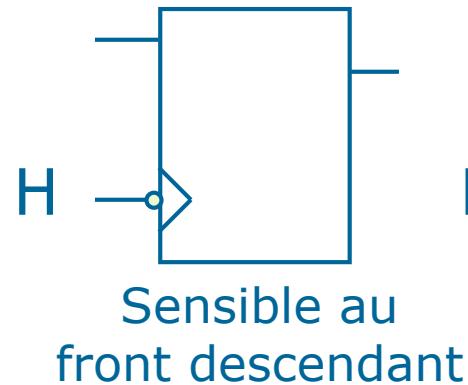
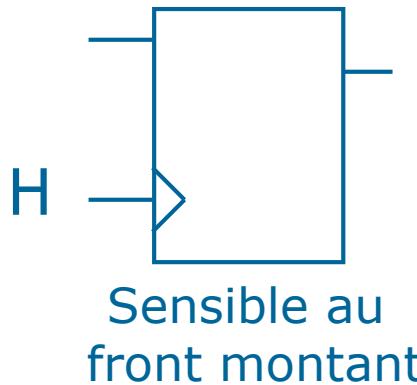


**Synchronisation sur front descendant**



# Synchrone et asynchrone

- **Circuit asynchrone : La sortie peut changer d'état à tout moment.**
  - Le circuit séquentiel n'a pas d'horloge comme variable d'entrée
  - Le circuit fonctionne indépendamment de cette horloge.
- **Circuit synchrone : La sortie change d'état à des instants régulièrement espacés.**
  - L'évolution de la sortie est synchronisée avec un évènement
  - Plus simple à concevoir.
- **Symboles :**



# Bascule

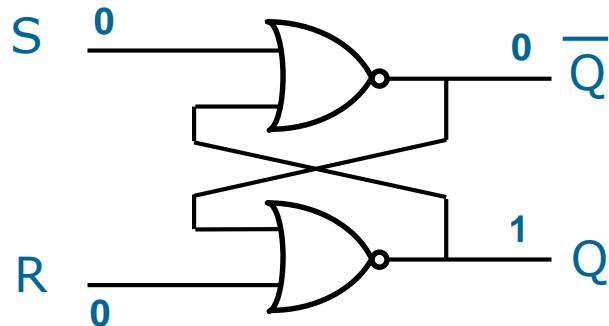
- **Définitions :**
  - ◆ Les bascules sont les circuits de bases de la logique séquentielle → Circuit de mémorisation le plus répandu
  - ◆ Une bascule peut être synchronisée avec une horloge (synchrone) ou non (asynchrone)
  - ◆ Chaque bascule possède une ou plusieurs entrées et deux sorties complémentaires
  - ◆ Les entrées déterminent le passage de la bascule d'un état à l'autre → Basculement
  - ◆ Quand une impulsion est appliquée à une entrée pour imposer un certain état de la bascule, la bascule demeure dans cet état même après le retrait de l'impulsion. C'est ce que l'on appelle la mémoire de la bascule.
  - ◆ Types de bascules :
    - ▶ RS
    - ▶ RST
    - ▶ JK
    - ▶ D
    - ▶ ...

# Entrées asynchrones

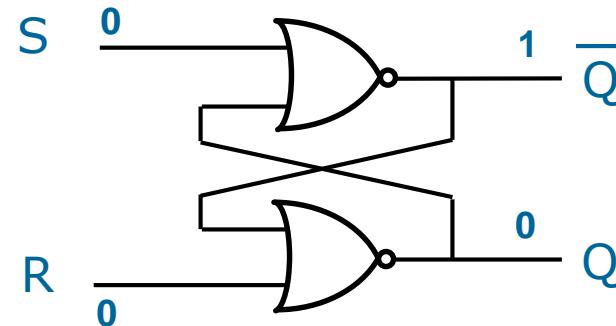
- **Agissent indépendamment des entrées synchrones et du signal d'horloge.**
- **Utilisées pour forcer en tout temps mise à 1 ou à 0. (Réinitialisation)**
- **Entrée prioritaire : imposent un état quel que soient les autres entrées**
- **Remarques :**
  - ◆ Logique négative (active à l'état bas)
  - ◆ RAU (Remise A Un) = DC SET = PRESET
  - ◆ RAZ (Remise A Zéro) = DC CLEAR = RESET

# Bascule RS – Fonctionnement

- Scénario : Les entrées S et R sont au niveau BAS. Momentanément on en met une au niveau HAUT pour changer l'état de sortie de la bascule.
- $S=R=0$



Soit  $Q=1$  et  $\bar{Q}=0$



Soit  $Q=0$  et  $\bar{Q}=1$

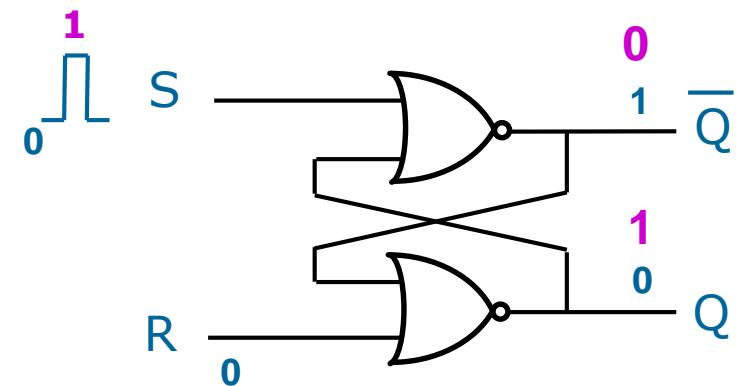
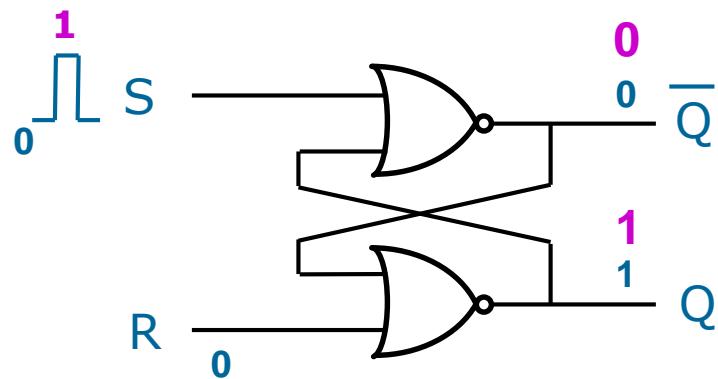
- Les sorties sont verrouillées dans leur état initial.

# Bascule RS – Fonctionnement

- **Basculement : S est momentanément amenée à 1**

■ État initial (N)

■ Nouvel état (N+1)



- Quelque soit l'état initial, l'impulsion à l'entrée S met toujours la bascule dans l'état :

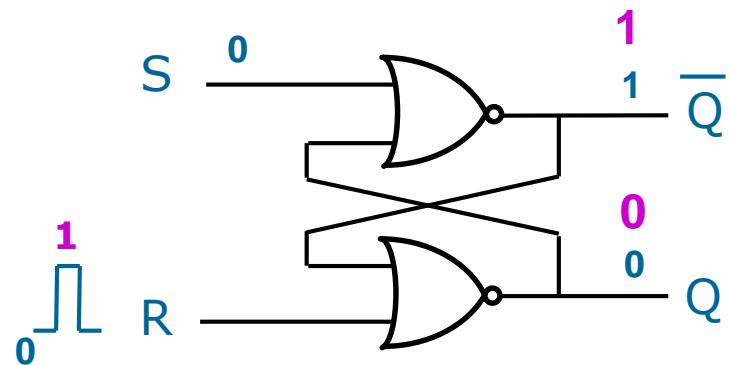
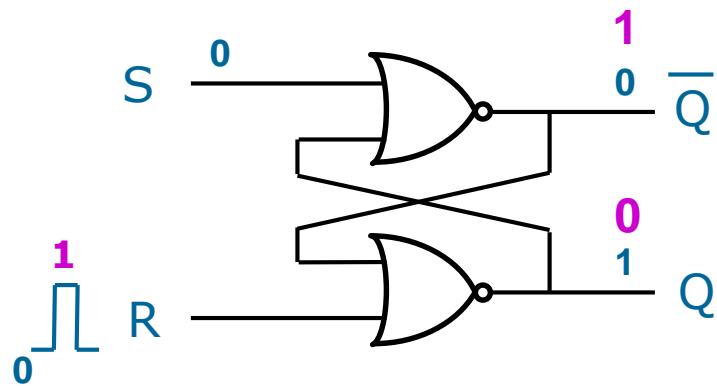
$$Q_{N+1} = 1 \text{ et } \bar{Q}_{N+1} = 0$$

# Bascule RS – Fonctionnement

- **Basculement : R est momentanément amenée à 1**

■ État initial (N)

■ Nouvel état (N+1)



- Quelque soit l'état initial, l'impulsion à l'entrée R met toujours la bascule dans l'état :

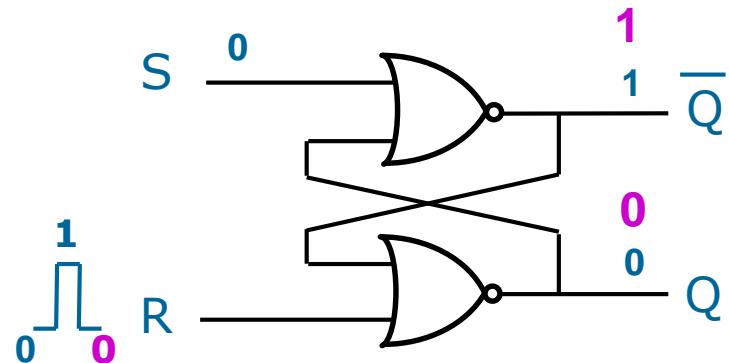
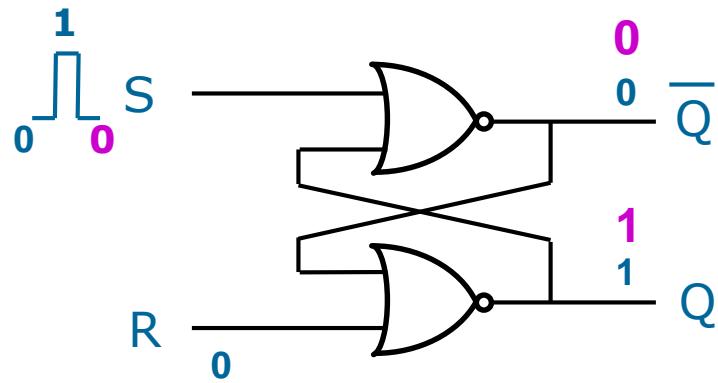
$$Q_{N+1} = 0 \text{ et } \bar{Q}_{N+1} = 1$$

# Bascule RS – Fonctionnement

- **Basculement : S ou R revient à 0**

■ État initial (N)

■ Nouvel état (N+1)



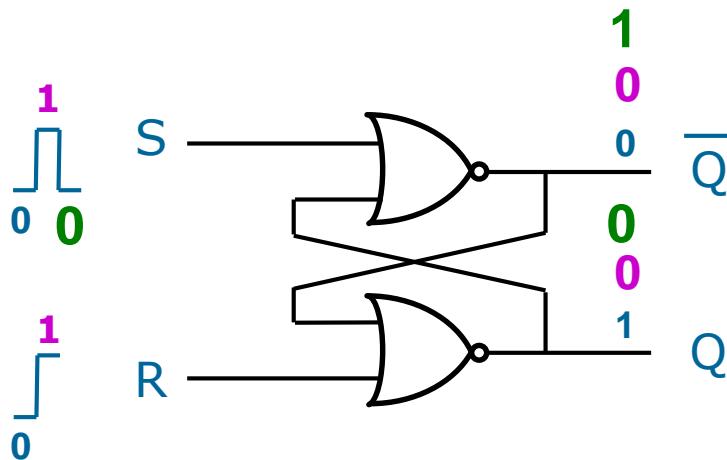
- La bascule mémorise l'état de la sortie même si un changement d'état a eu lieu dans ses entrées :

$$Q_{N+1} = Q_N \text{ et } \overline{Q}_{N+1} = \overline{Q}_N$$

# Bascule RS – Fonctionnement

- **Condition indésirable : S=R=1**

■ État initial (N)      ■ Nouvel état (N+1)

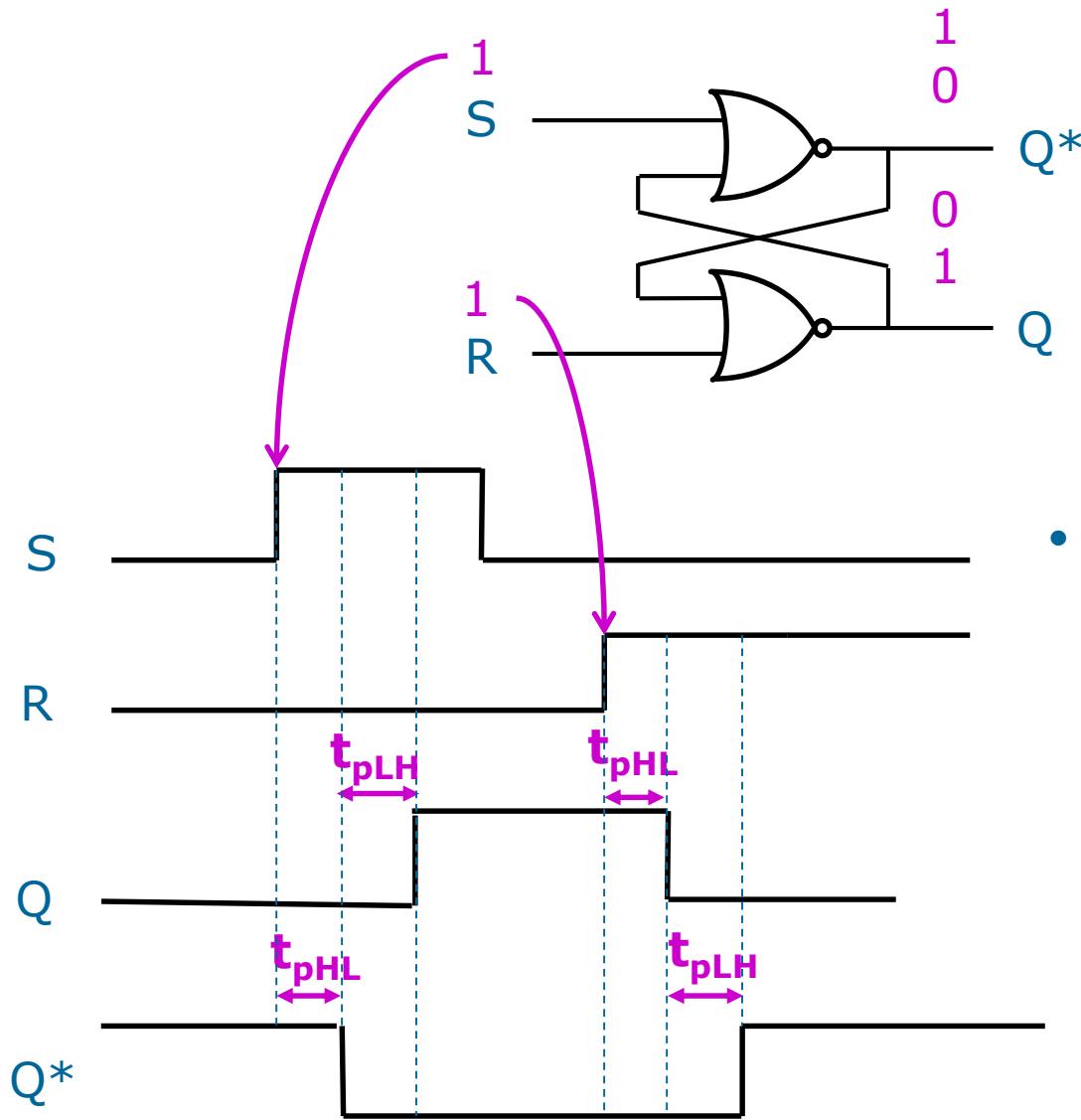


- Cette condition est censée ne jamais se reproduire.

$$Q_{N+1} = \bar{Q}_{N+1} \rightarrow \text{impossible !}$$

- Si entre temps une entrée change d'état (ici  $S \rightarrow 0$ ), la sortie sort de l'état impossible.

# Bascule RS – Analyse temporelle



- **Dissymétrie dans le positionnement des sorties, donc  $Q^* = Q$  n'est vérifié que d'un point de vue statique.**

# Bascule RS – Analyse statique

- **Table de vérité**

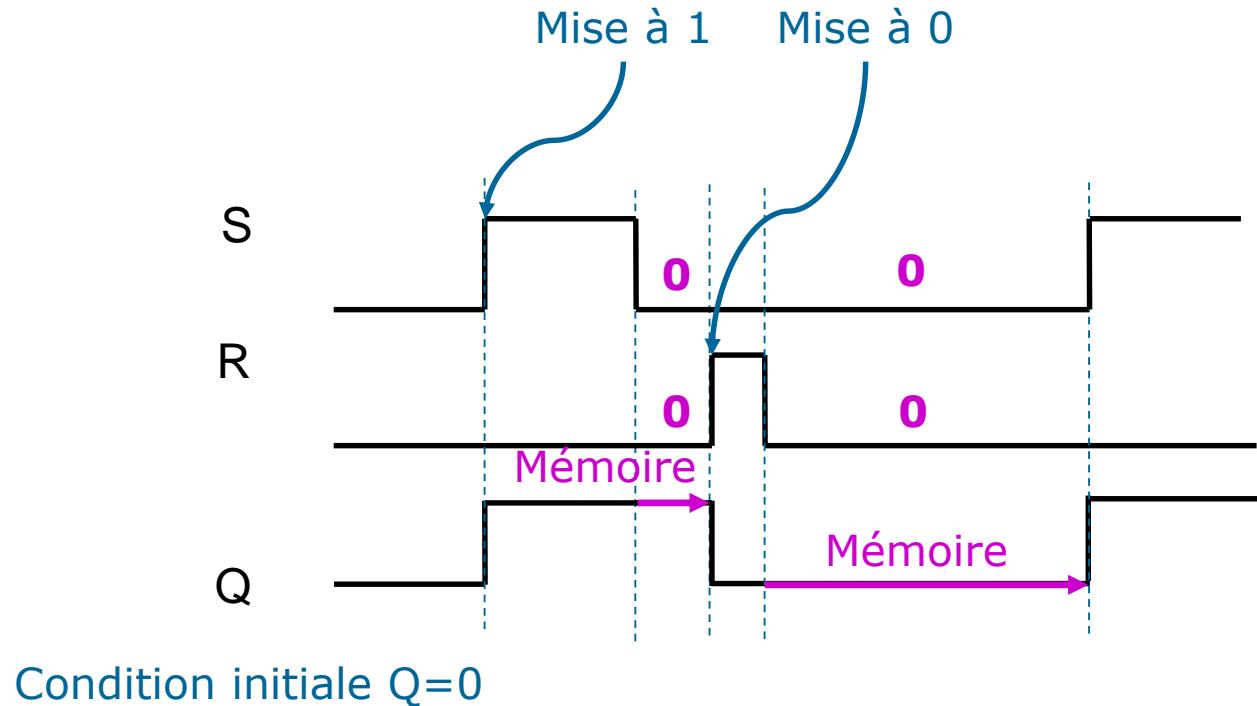
S	R	Q	$\bar{Q}$
0	0	Inchangé	
0	1	0	1
1	0	1	0
1	1	Indésirable (Ambigu)	

- **S représente SET et R représente RESET**

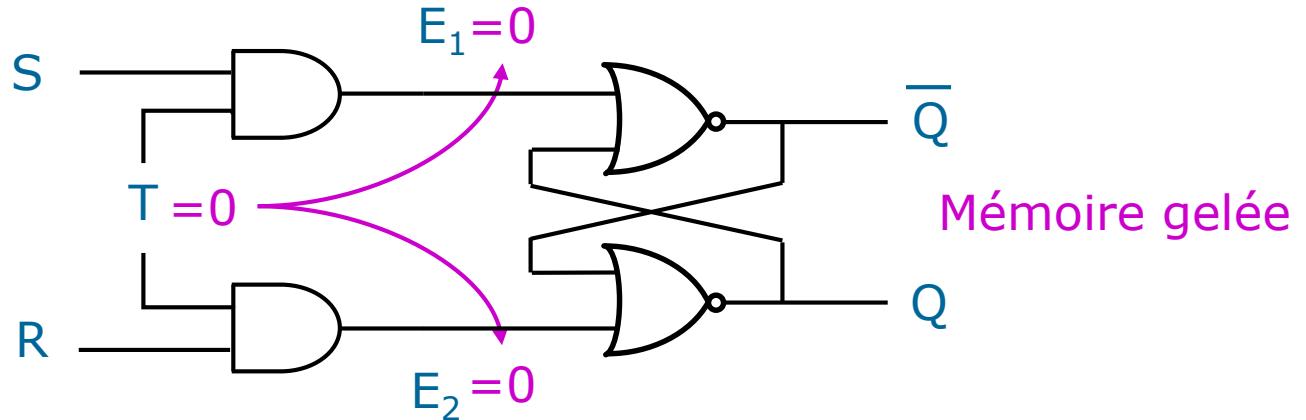
- SET et RESET simultanément actifs ( $S=R=1$ ) → Etat indésirable
- SET actif ( $S=1$ ) →  $Q=1$  (mise à 1)
- RESET actif ( $R=1$ ) →  $Q=0$  (mise à 0)
- SET et RESET simultanément inactifs ( $S=R=0$ ) → Mémoire

# Bascule RS – Analyse statique

- **Chronogramme**



# Bascule RST



$T = 1$  : Mémoire classique

$T = 0$  :  $E_1 = E_2 = 0 \rightarrow$  Mémoire «gelée»

- **Avantages**

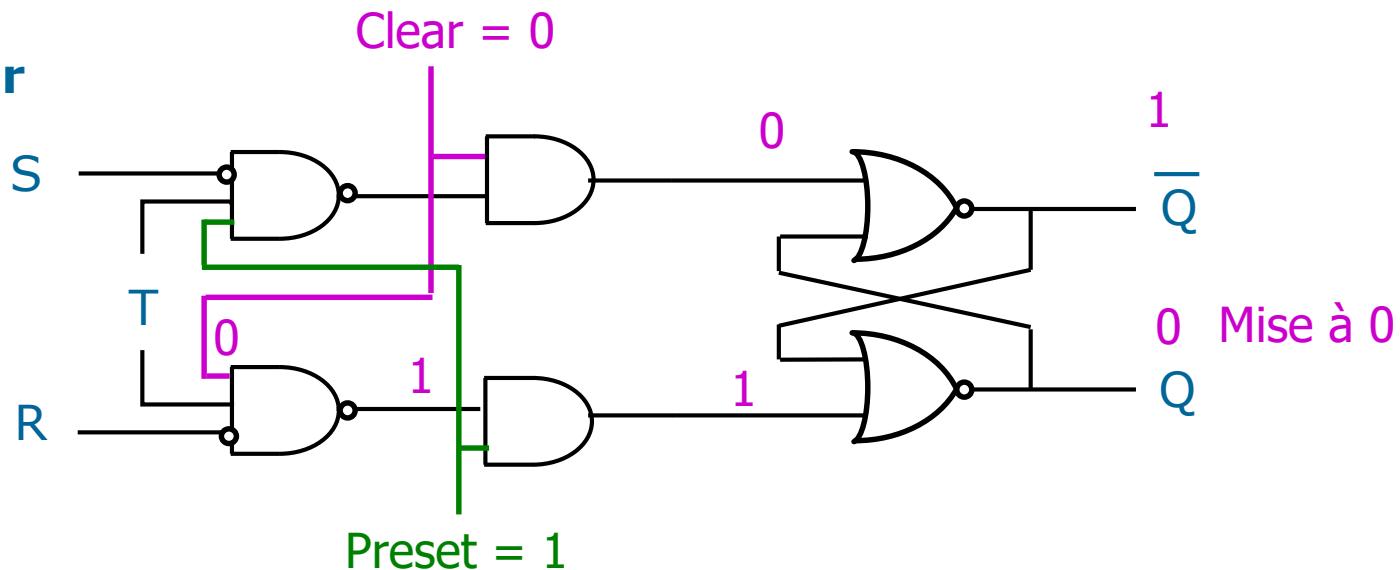
- Préparer S et R, ensuite T = 1
- Isoler la mémoire des parasites

- **Inconvénients**

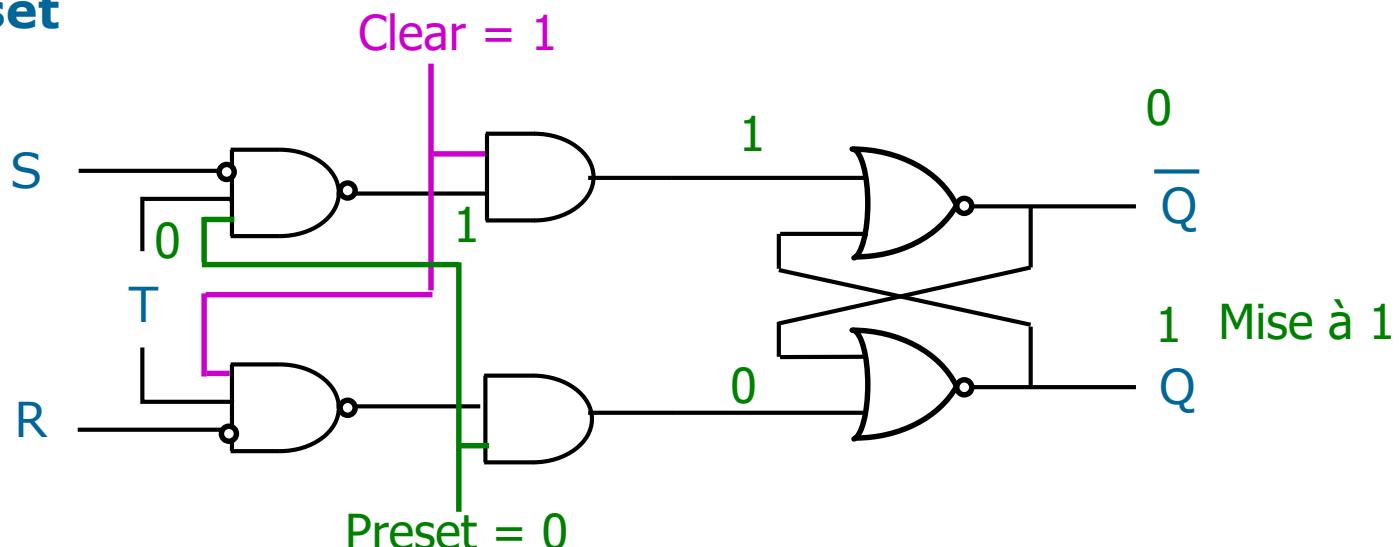
- **R = S = 1 toujours gênant**
- **Il faut avoir T = 1 pendant un certain temps**

# Bascule RST avec Clear/Preset (initialisation)

- **Clear**



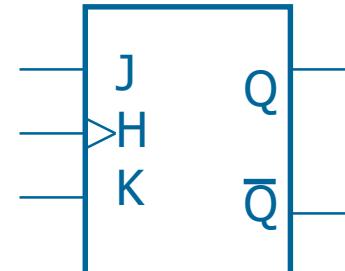
- **Preset**



- Etat interdit : CLEAR = PRESET = 0

# Bascule JK synchrone

- Bascule avec deux entrées J et K et une horloge H (front montant ou descendant)**



The symbol shows a square box with three input lines and two output lines. The top input is labeled J, the bottom input is labeled K, and the middle input is labeled H with a triangle symbol indicating it is an active-high clock input. The rightmost output is labeled Q, and the bottom-right output is labeled  $\bar{Q}$ .



A small diagram of a rising edge waveform is shown below the symbol.

J	K	$Q_{N+1}$	$\bar{Q}_{N+1}$
0	0	$Q_N$	$\bar{Q}_N$
1	0	1	0
0	1	0	1
1	1	$\bar{Q}_N$	$Q_N$

Mémorisation

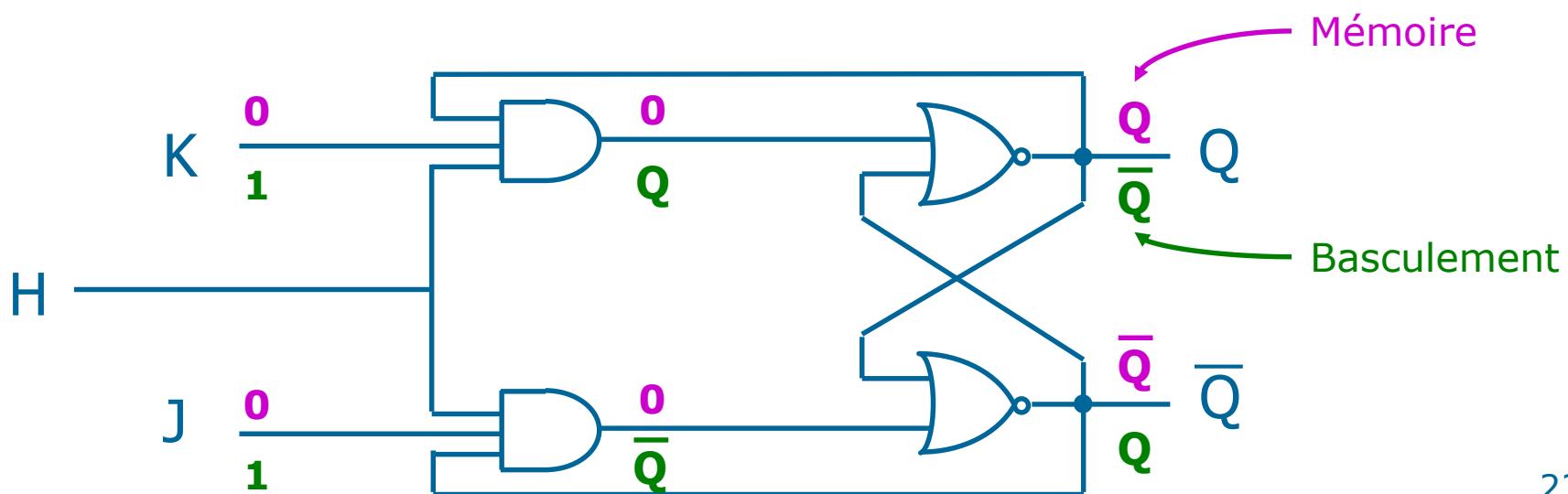
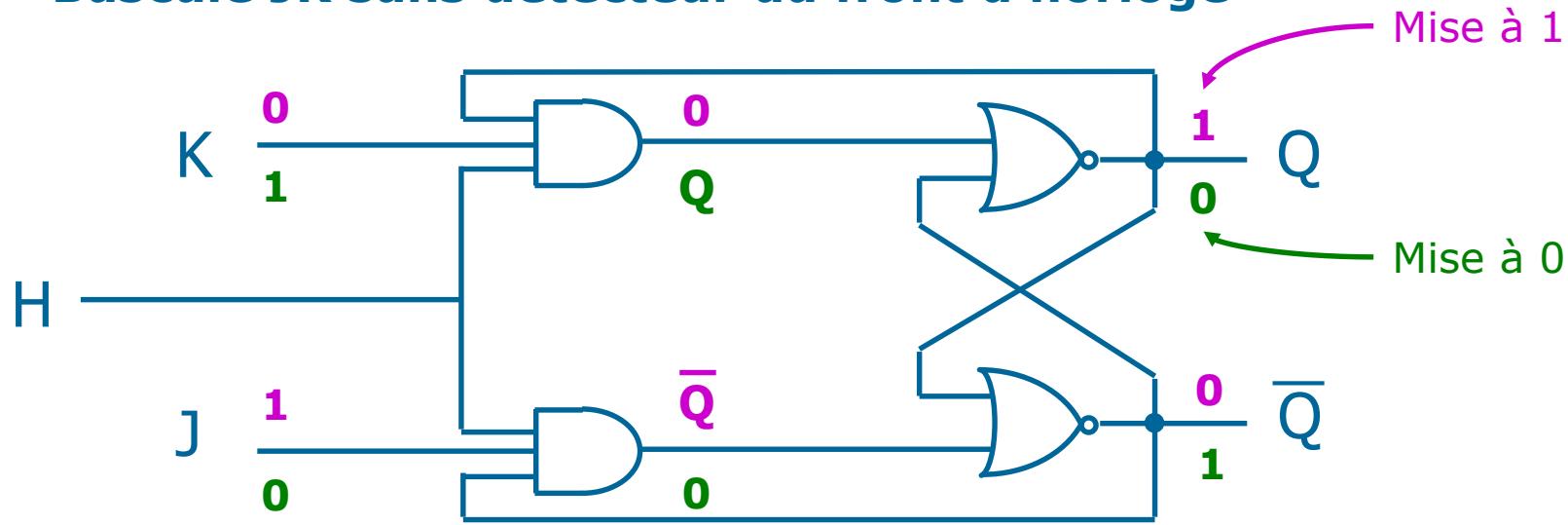
Mise à 1

Mise à 0

Toggle (Basculement)

# Bascule JK synchrone – Réalisation

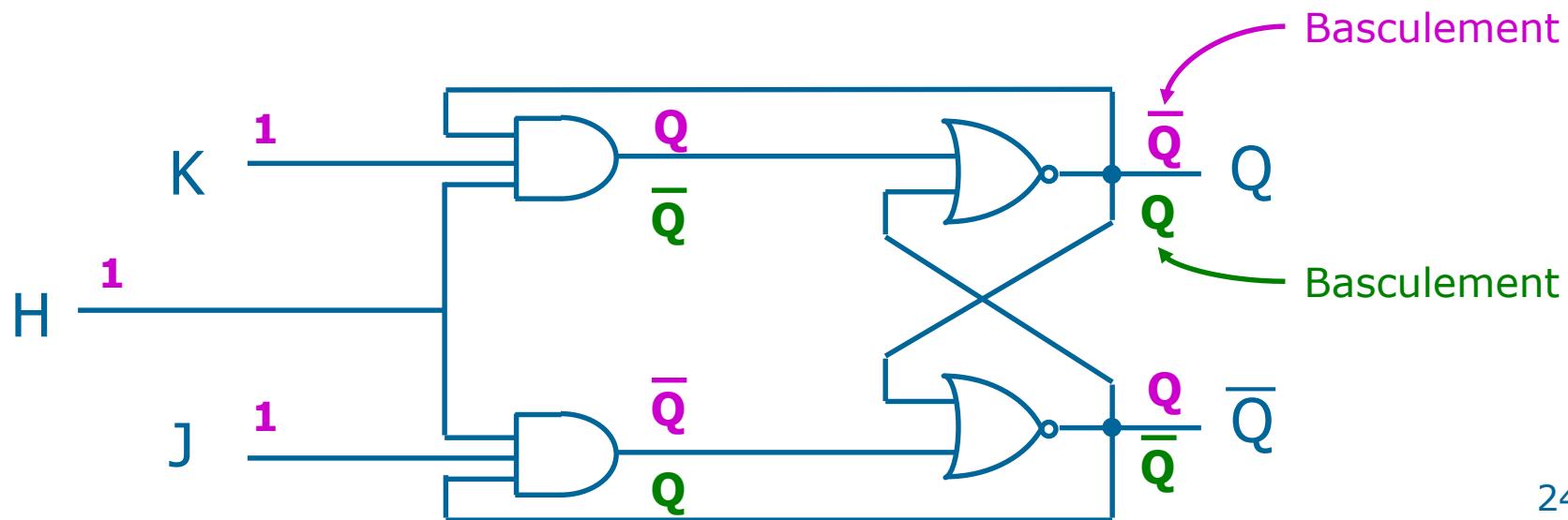
- **Bascule JK sans détecteur du front d'horloge**



# Bascule JK- Réalisation

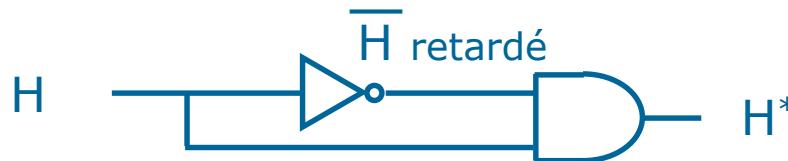
- **Remarque :**

Avec le montage présenté, tant que  $J=K=1$  et que l'horloge est au niveau haut, le basculement continue. (Il suffit de recommencer l'analyse avec les nouveaux états.) Le basculement s'arrête quand l'horloge passe de 1 à 0. Pour que ce montage fonctionne correctement, il faut que la durée du niveau haut du signal d'horloge soit plus courte que le temps de propagation de la porte pour que l'horloge passe à 0 avant que la sortie soit établie. Ce défaut se corrige en utilisant un circuit du détecteur du front d'horloge.

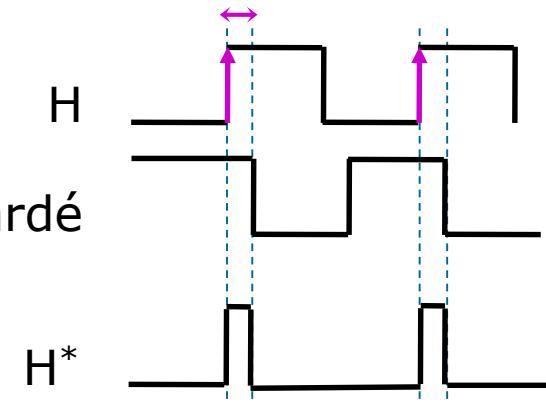


# Circuit détecteur de front d'horloge

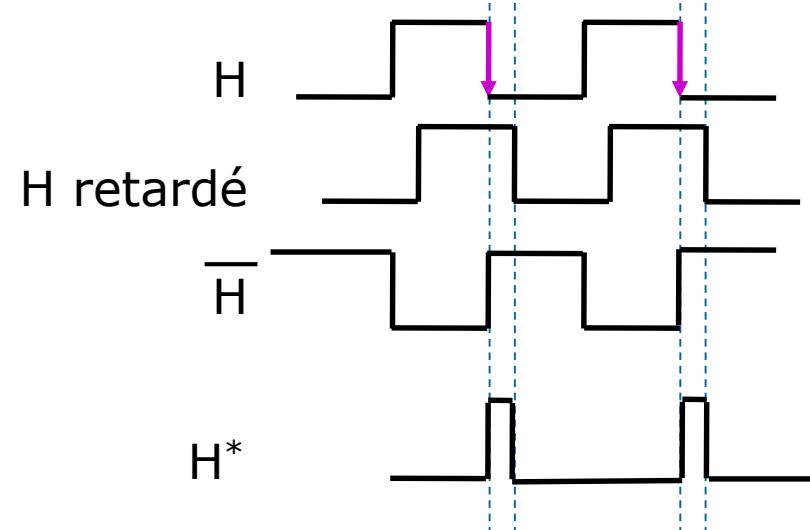
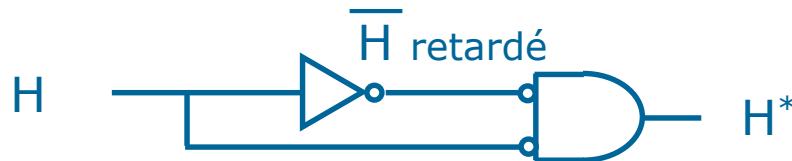
- **Détecteur de front montant ( $\uparrow$ )**



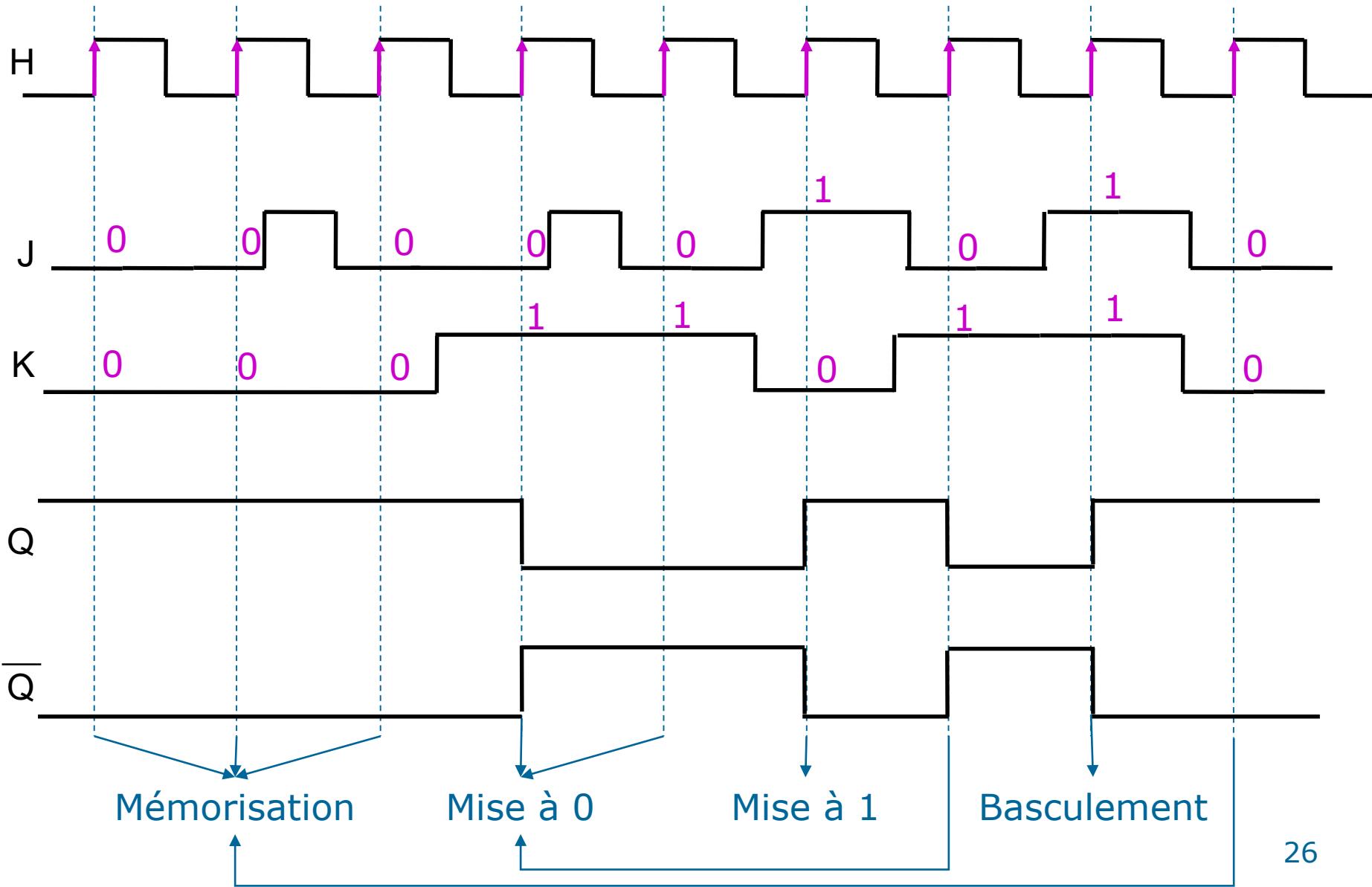
Quelques ns du retard



- **Détecteur de front descendant ( $\downarrow$ )**

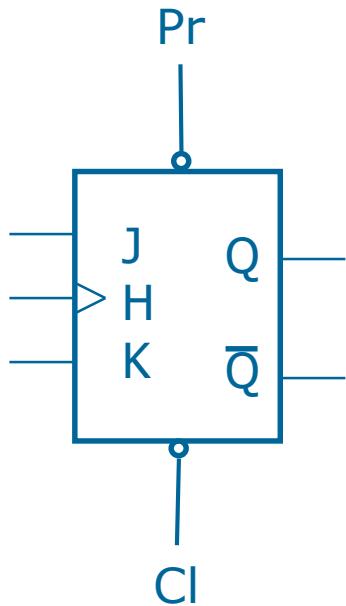


# Bascule JK synchrone – Chronogramme



# Bascule JK asynchrone

- **Deux entrées asynchrones Pr (Preset) et Cl (Clear)**
  - ◆ Plus prioritaires que l'horloge
  - ◆ Fonctionnement avec la logique négative



# Bascule JK

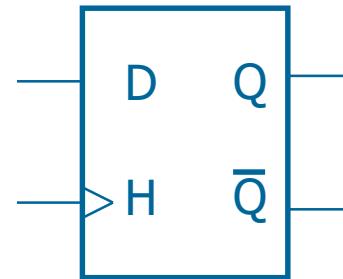
- **Table de vérité**

	Pr	Cl	H	J	K	$Q_{N+1}$	
Mode asynchrone	0	0	X	X	X	X	État interdit
	0	1	X	X	X	1	Remise à 1
	1	0	X	X	X	0	Remise à 0
Mode synchrone	1	1	0/1	X	X	$Q_N$	Etat mémoire
	1	1	↑	0	0	$Q_N$	Etat mémoire
	1	1	↑	0	1	0	Remise à 0
	1	1	↑	1	0	1	Remise à 1
	1	1	↑	1	1	$\bar{Q}_N$	Basculement

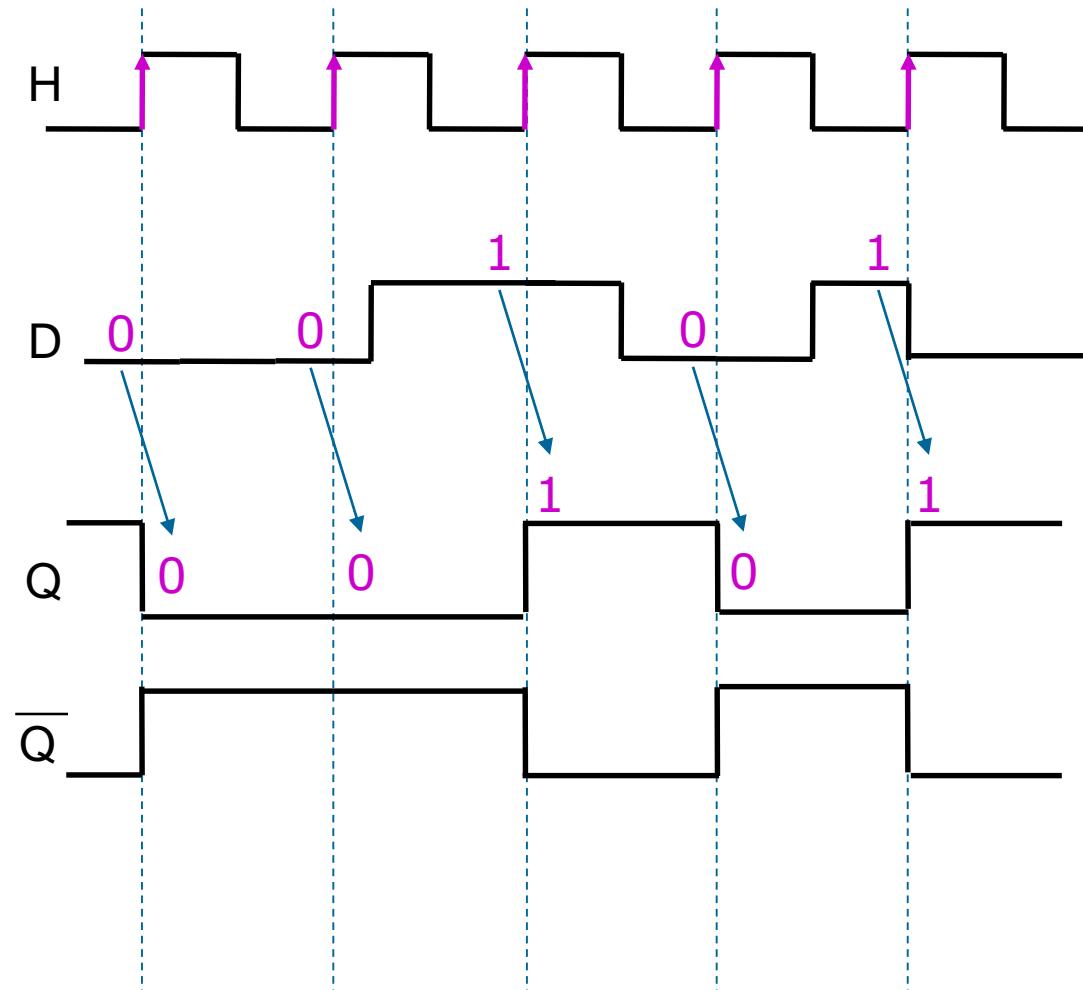
# Bascule D synchrone

- **Bascule avec une entrée D et un horloge H synchronisée sur le front montant ou descendant**

H	D	$Q_{N+1}$	$\bar{Q}_{N+1}$
0/1	X	$Q_N$	$\bar{Q}_N$
↑	0	0	1
↑	1	1	0

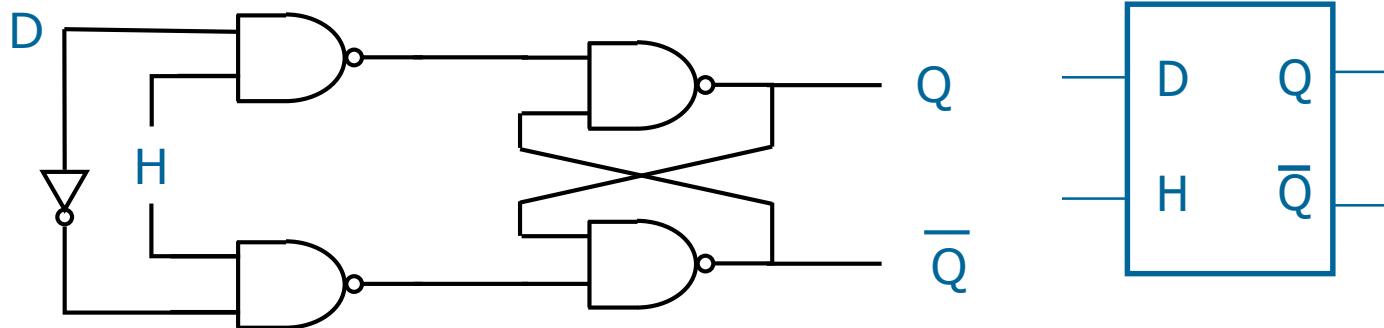


# Bascule D synchrone – Chronogramme



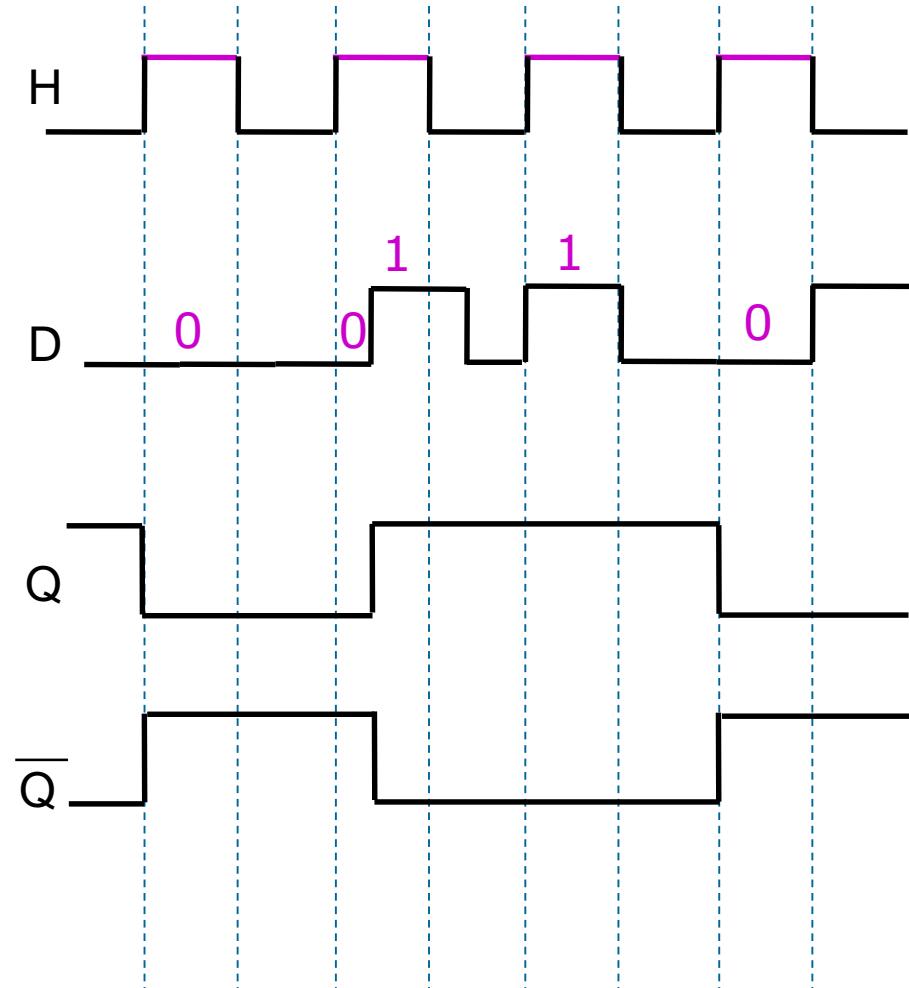
# Bascule D à verrouillage (Latch)

- **Bascule synchrone sur les niveaux de l'horloge**
- **On supprime le circuit détecteur du front de l'horloge dans le bascule D synchrone.**

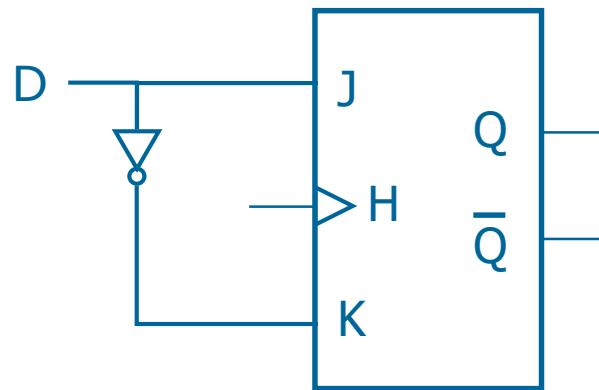


H	D	$Q_{N+1}$	$\bar{Q}_{N+1}$
0	X	$Q_N$	$\bar{Q}_N$
1	0	0	1
1	1	1	0

# Bascule D à verrouillage – Chronogramme



# Bascule JK à bascule D



H	J	K	$Q_{N+1}$
0/1	X	X	$Q_N$
↑	0	1	0
↑	1	0	1

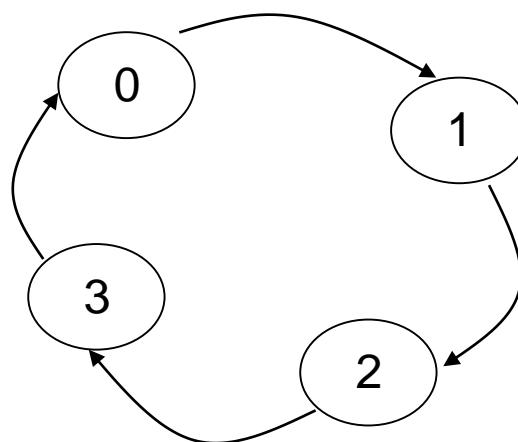
# Table de transition JK

- **Problème : On connaît les valeurs des sorties, comment déterminer les valeurs des entrées ?**

$Q_N$	$Q_{N+1}$	J	K	
0	0	0	X	<b>Remise à 0 ou état mémoire</b>
0	1	1	X	<b>Remise à 1 ou basculement</b>
1	0	X	1	<b>Remise à 0 ou basculement</b>
1	1	X	0	<b>Remise à 1 ou état mémoire</b>

# Exercice

- Réaliser le circuit qui permet de reproduire le cycle 0, 1, 2, 3 à l'aide de bascules JK.



# Solution

Etat précédent

Etat suivant

$Q_{1N}$	$Q_{0N}$	$J_1$	$K_1$	$J_0$	$K_0$	$Q_{1N+1}$	$Q_{0N+1}$
0	0	0	X	1	X	0	1
0	1	1	X	X	1	1	0
1	0	X	0	1	X	1	1
1	1	X	1	X	1	0	0

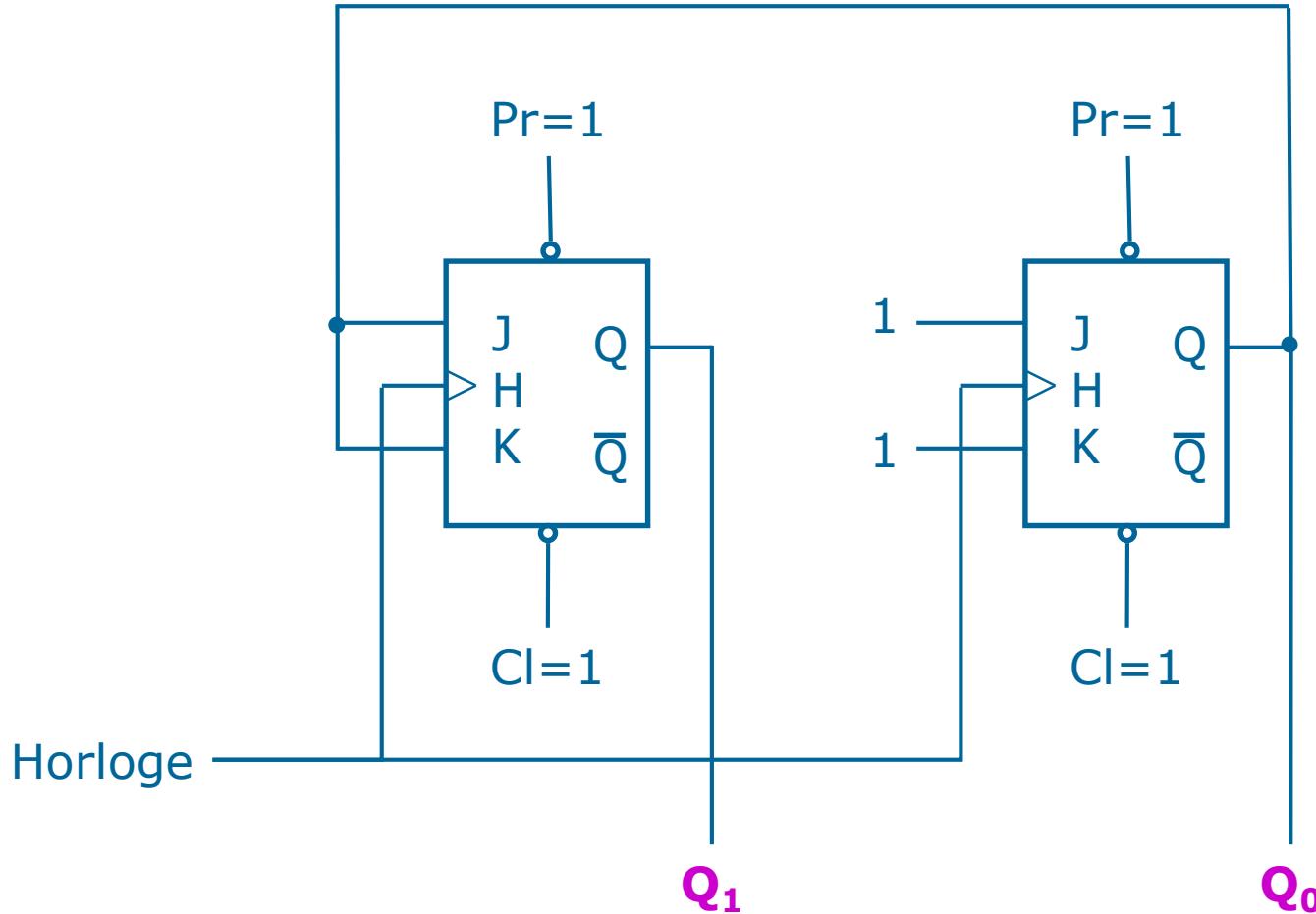


$$J_1 = K_1 = Q_{0N}$$

$$J_0 = K_0 = 1$$

# Solution – Schéma

$$J_1 = K_1 = Q_{0 \text{ N}} \quad J_0 = K_0 = 1$$



# Compteurs

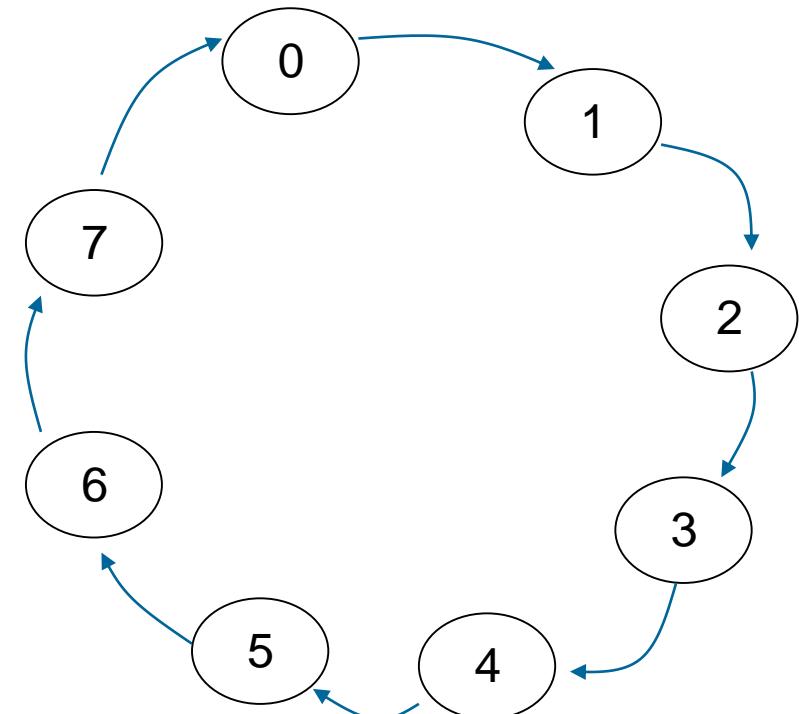
Asynchrones  
Synchrones

# Table de transition

Etats précédents

Etats suivants

$Q_2$	$Q_1$	$Q_0$		$Q_2^+$	$Q_1^+$	$Q_0^+$
0	0	0		0	0	1
0	0	1		0	1	0
0	1	0		0	1	1
0	1	1		1	0	0
1	0	0		1	0	1
1	0	1		1	1	0
1	1	0		1	1	1
1	1	1		0	0	0



Un cycle (succession des états)

# Qu'est-ce qu'un compteur ?

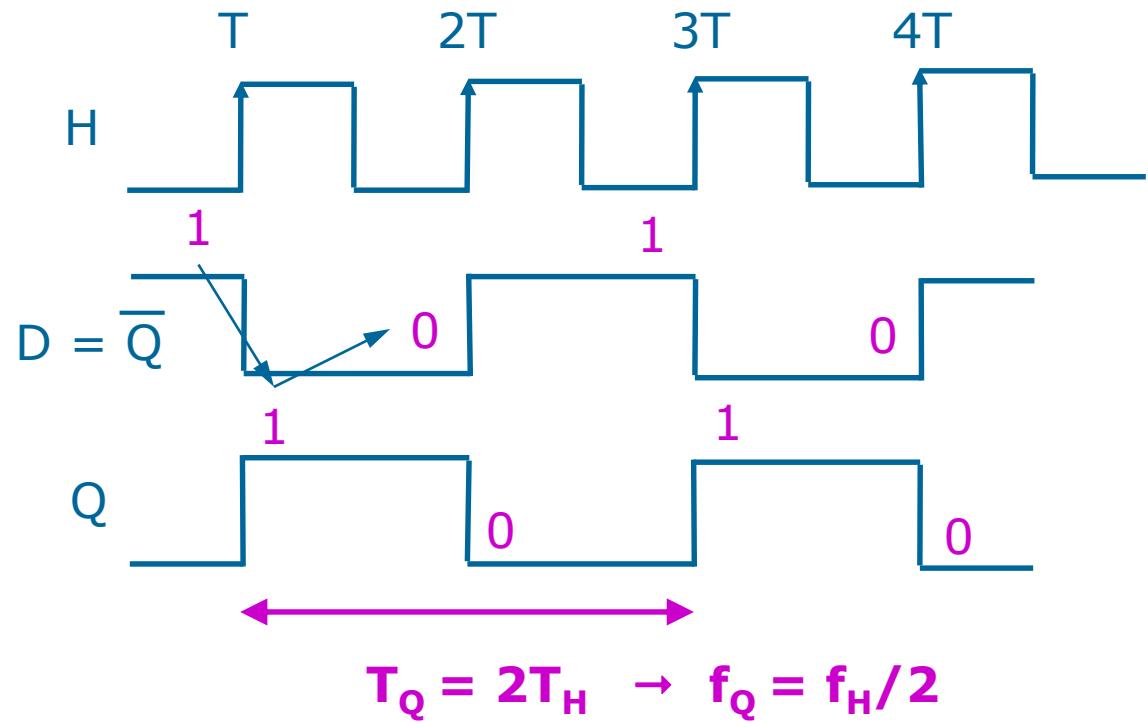
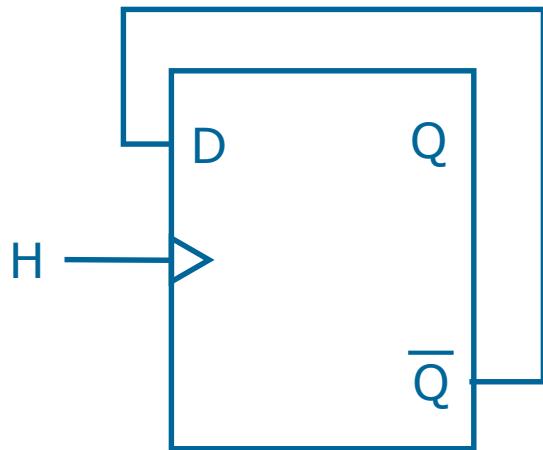
- **Un circuit séquentiel qui génère une suite sur  $N$  sorties et qui possède donc  $N$  états ( $E_0, E_1, \dots, E_{n-1}$ ).**
- **À chaque front (ou niveau) d'horloge, il passe de l'état  $E_i$  à l'état  $E_{i+1}$ .**
- **Il revient toujours à l'état initiale  $E_0$ .**
- **Il possède un cycle ( une séquence d'états).**
- **Il est constitué de  $n$  bascules.**
- **Le nombre d'états d'un compteur est inférieurs ou égale à  $2^n$ .**
- **Il existe deux types de compteur : les compteurs synchrones et asynchrones.**
  - Les bascules qui constituent un compteur synchrone possèdent la même horloge.
  - Les bascules qui constituent un compteur asynchrone possèdent des horloges différentes.

# Types de compteur

- **Les compteurs modulo  $2^n$  (cycle complet) :**
  - ◆  $n=2 : 0, 1, 2, 3, 0 \rightarrow \text{modulo } 2^2=4$
  - ◆  $n=3 : 0, 1, 2, 3, 4, 5, 6, 7, 0 \rightarrow \text{modulo } 2^3=8$
  - ◆  $n=4 : 0, 1, 2, 3, 4, \dots, 14, 15, 0 \rightarrow \text{modulo } 2^4=16$
- **Les compteurs modulo N (cycle incomplet) :**
  - ◆  $N=5 : 0, 1, 2, 3, 4, 0 \rightarrow \text{modulo } 5$
  - ◆  $N=10 : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0 \rightarrow \text{modulo } 10$
- **Les compteurs à cycle quelconque :**
  - ◆  $0, 2, 5, 6, 7, 8, 10, 0$

# Diviseur de fréquence – Bascule D

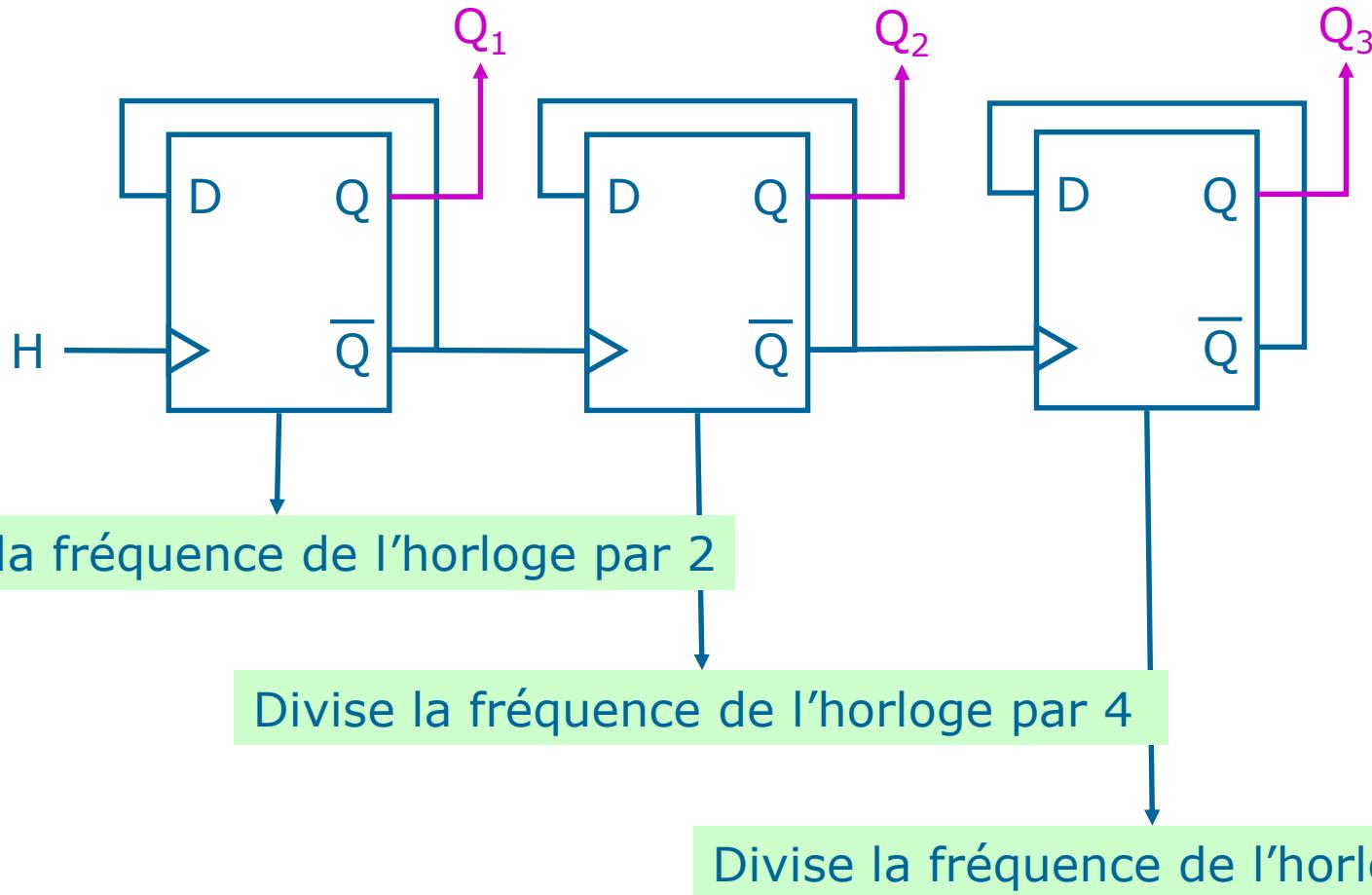
- **Diviseur par 2**



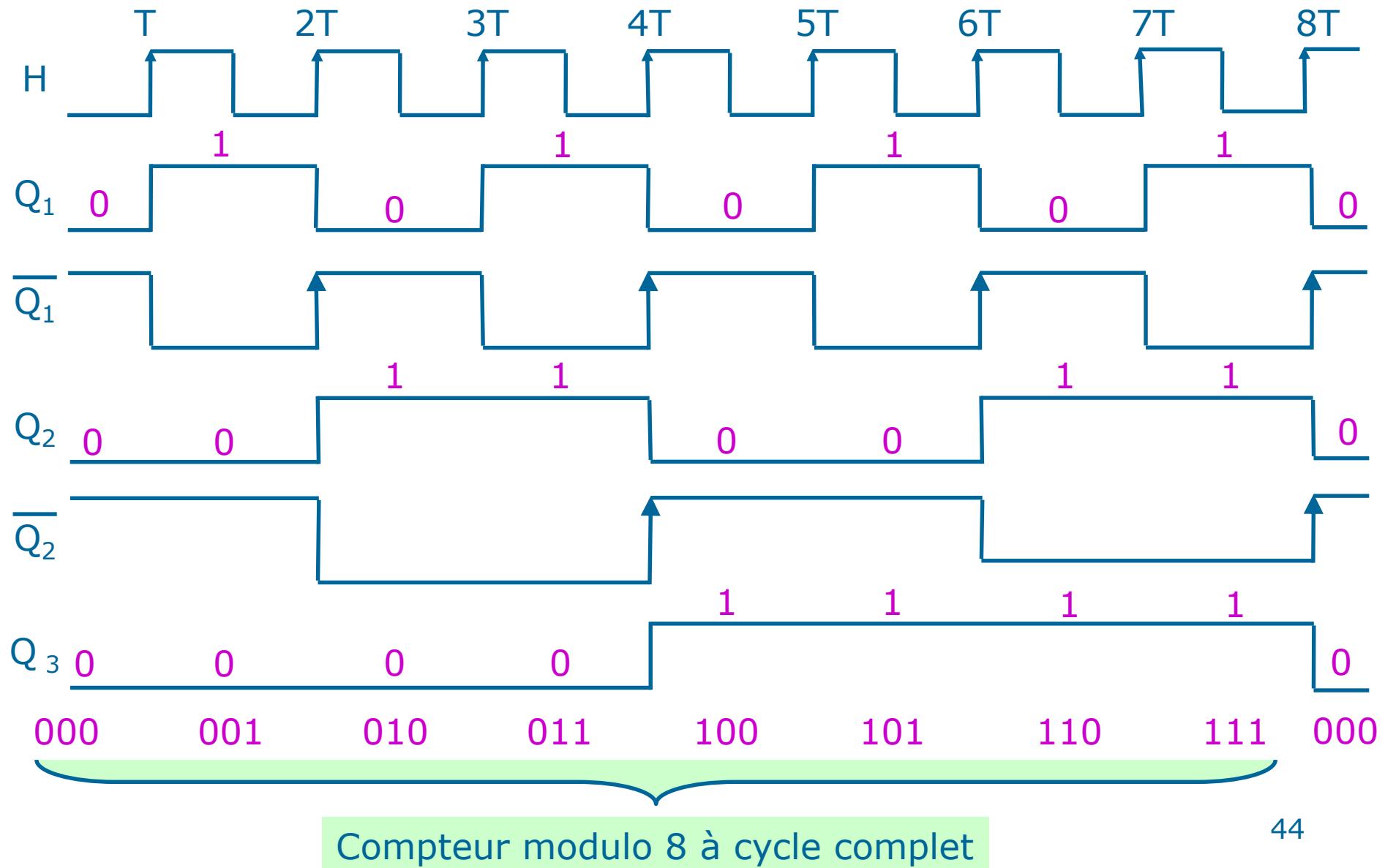
- **Le circuit du diviseur de fréquence par 2 est un simple compteur sur 1 bit. → Compteur modulo 2**

# Compteur asynchrone – Bascule D

- Cascade des diviseurs en fréquences



# Compteur asynchrone – Bascule D

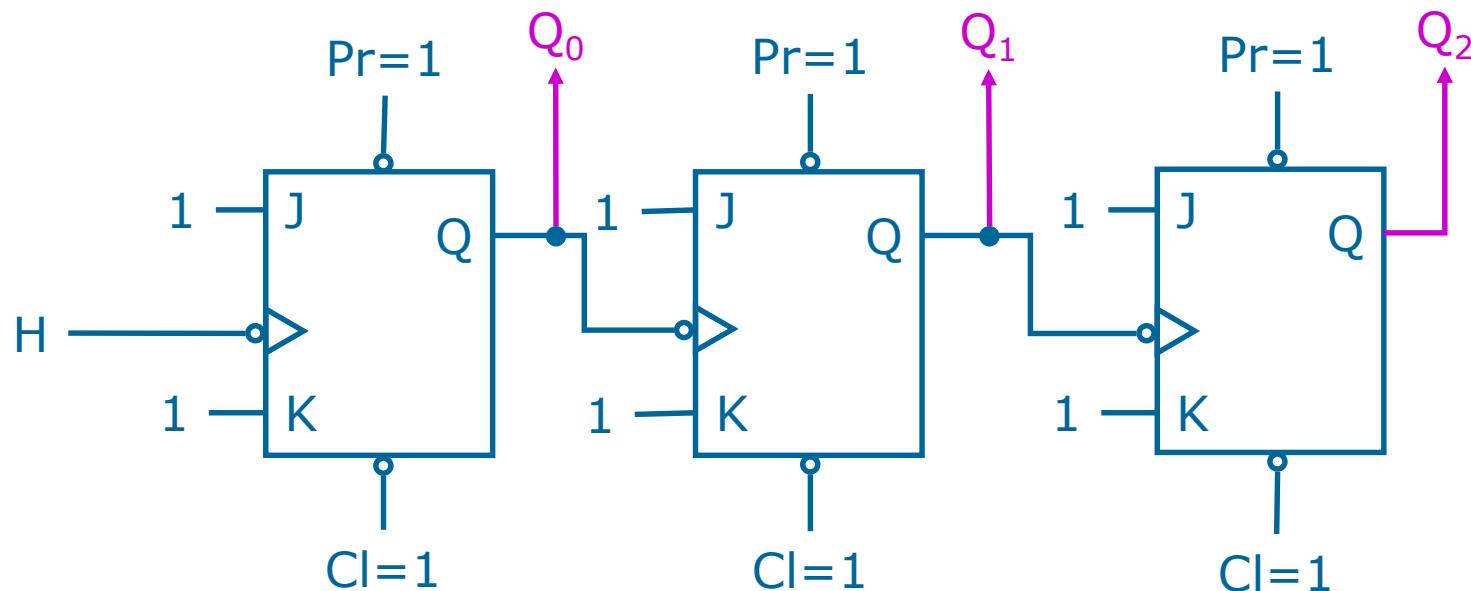


# Compteur asynchrone – Bascule JK

<b>Q2</b>	<b>Q1</b>	<b>Q0</b>		<b>Q2<sup>+</sup></b>	<b>Q1<sup>+</sup></b>	<b>Q0<sup>+</sup></b>
0	0	0		0	0	1
0	0	1		0	1	0
0	1	0		0	1	1
0	1	1		1	0	0
1	0	0		1	0	1
1	0	1		1	1	0
1	1	0		1	1	1
1	1	1		0	0	0

- **Analyse de la table de transition d'un compteur modulo 8**
  - Q<sub>0</sub> passe à chaque fois de 0 à 1 et de 1 à 0.
    - ▶ Q<sub>0</sub> utilise l'horloge du circuit.
    - ▶ basculement J<sub>0</sub>=K<sub>0</sub>=1
  - Q<sub>1</sub> passe de 0 à 1 et de 1 à 0 à chaque fois que Q<sub>0</sub> passe de 1 à 0.
    - ▶ Q<sub>0</sub> est utilisée comme horloge pour Q<sub>1</sub>.
    - ▶ basculement J<sub>1</sub>=K<sub>1</sub>=1
  - Q<sub>2</sub> passe de 1 à 0 et de 1 à 0 à chaque fois que Q<sub>1</sub> passe de 1 à 0.
    - ▶ Q<sub>1</sub> est utilisée comme horloge pour Q<sub>2</sub>.
    - ▶ basculement J<sub>2</sub>=K<sub>2</sub>=1

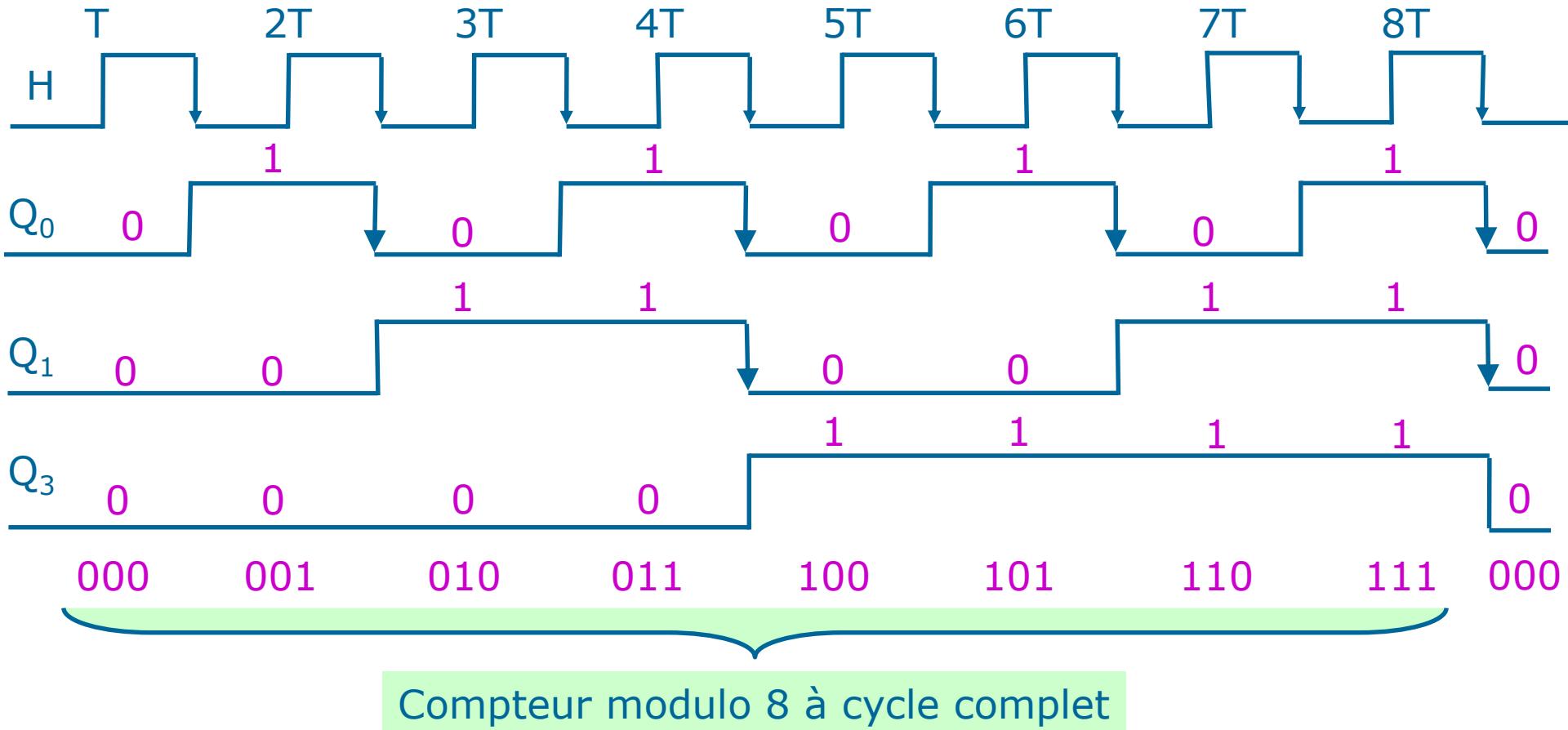
# Compteur asynchrone – Bascule JK



- **3 bascules créent un compteur modulo  $2^3=8$ .**
- **Les 3 bascules sont en modes de basculement  $J=K=1$ .**
- **La sortie de chaque bascule sert comme horloge pour la bascule suivante.**

# Compteur asynchrone – Bascule JK

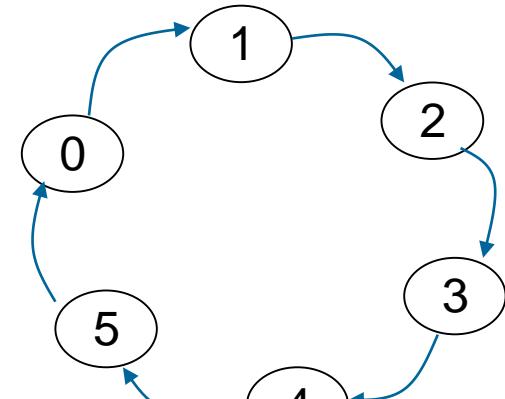
- Bascules JK sensibles au front descendant de l'horloge**



# Compteur asynchrone modulo N < 2<sup>n</sup>

- Compteur asynchrone modulo 6

$Q_2$	$Q_1$	$Q_0$		$Q_2^+$	$Q_1^+$	$Q_0^+$		Pri	Cli
0	0	0		0	0	1		1	1
0	0	1		0	1	0		1	1
0	1	0		0	1	1		1	1
0	1	1		1	0	0		1	1
1	0	0		1	0	1		1	1
1	0	1		1	1	0		1	1
1	1	0		0	0	0		1	0



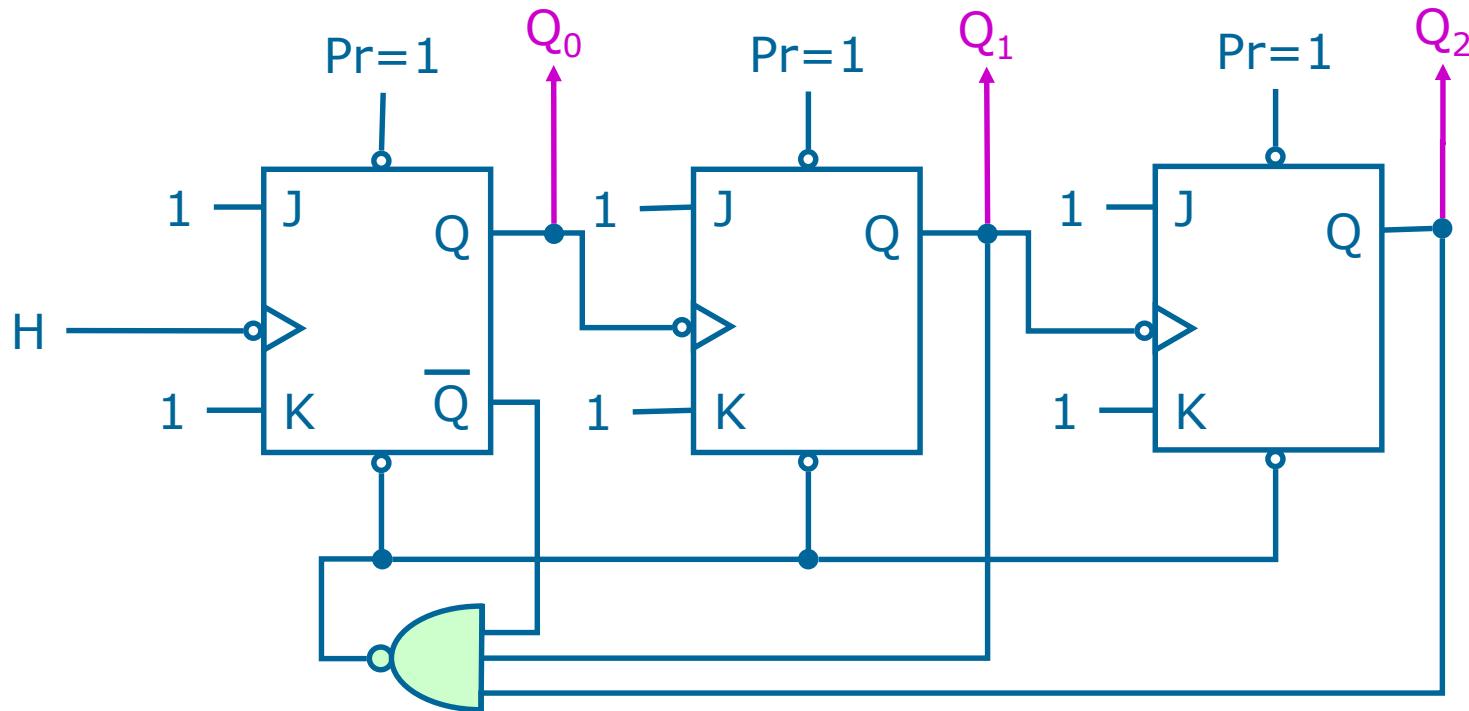
$$\overline{Cli} = Q_2 \cdot Q_1 \cdot \overline{Q_0}$$

$$Cli = \overline{Q_2 \cdot Q_1 \cdot \overline{Q_0}}$$

État inaccessible

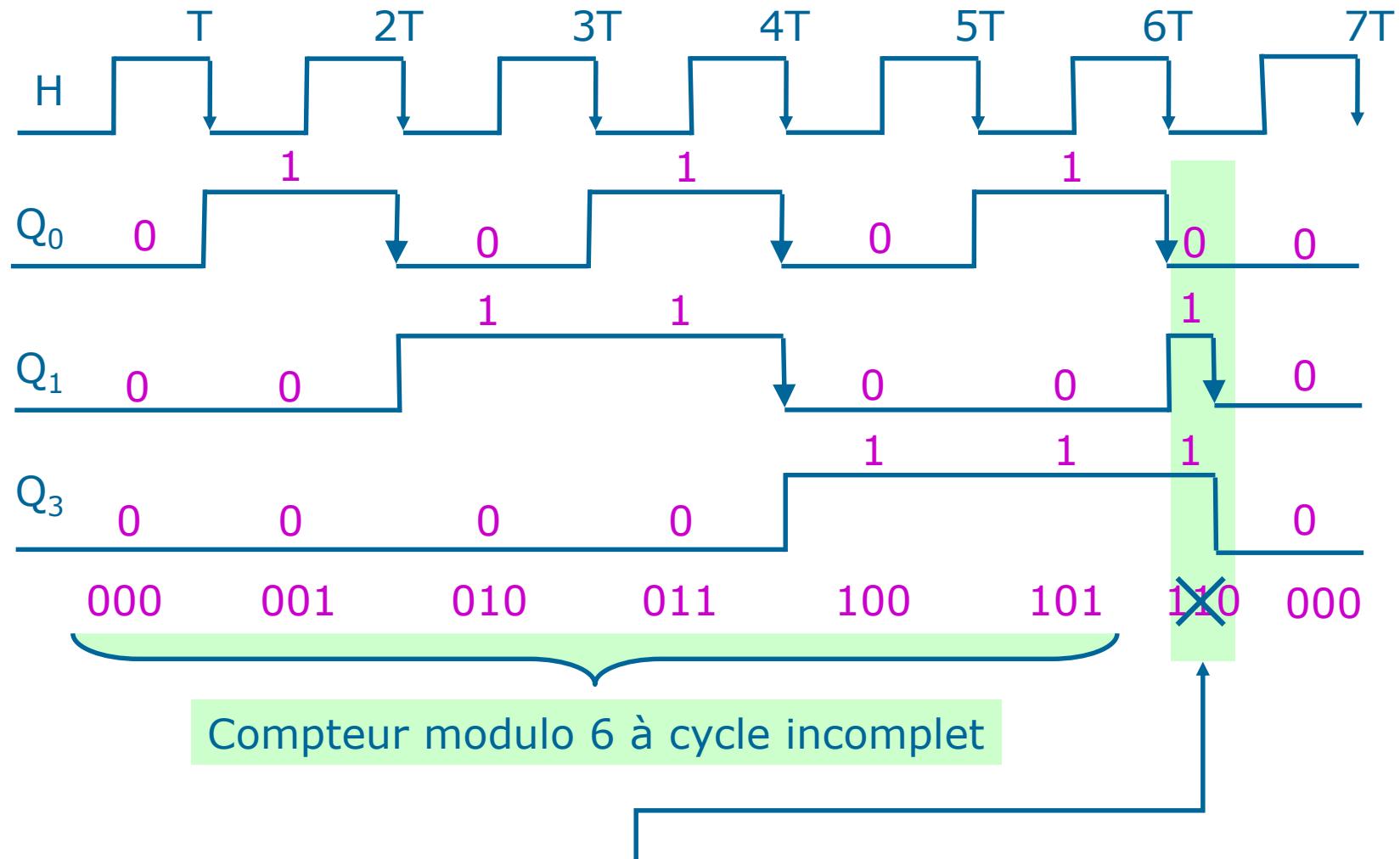
Détection de l'état 110 et remise à zéro asynchrone : Cli = 0 et Pri = 1

# Compteur asynchrone modulo 6



- **3 bascules créent un compteur modulo  $5 < 2^3$ .**
- **Les 3 bascules sont en modes de basculement  $J=K=1$ .**
- **L'entrée asynchrone « Clear » est programmée pour éviter l'état de sortie **110** et de forcer les sorties à 0 pour que le cycle du compteur recommence.**

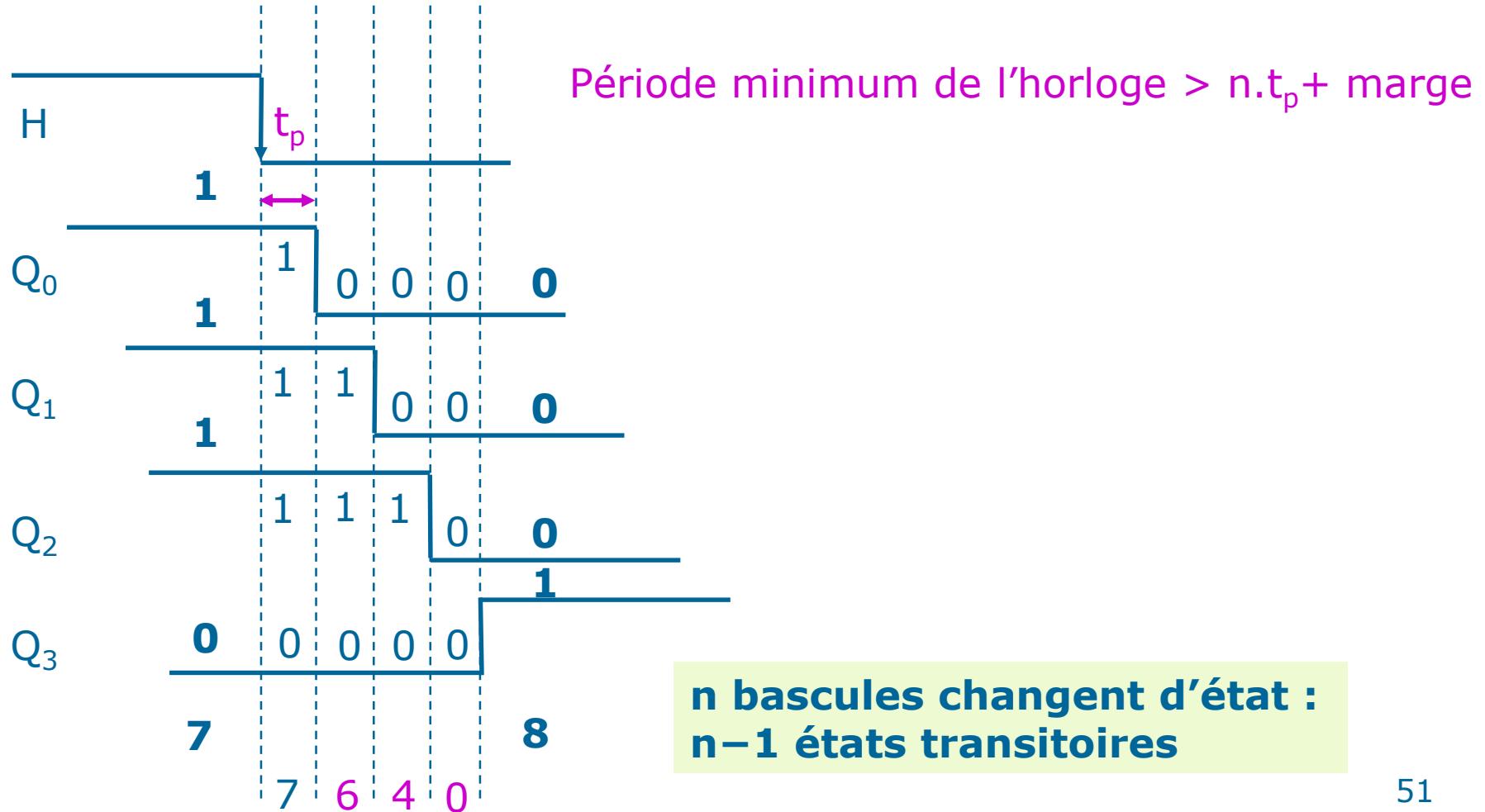
# Compteur asynchrone modulo 6



Détection de l'état **110** et mise à 0 asynchrone des bascules.

# Inconvénient du compteur asynchrone

- Accumulation des temps de propagation
- Exemple : passage de 7 à 8 dans un compteur modulo 16



# Compteur synchrone

- **Compteur à N états**
  - ◆  $2^{n-1} < N < 2^n \rightarrow n$  est le nombre de bascules
- **Les bascules partagent la même horloge.**
- **Il faut construire une table de transition, extraire les expressions des entrées de chaque bascule et ajouter des portes logiques pour préparer les entrées.**

# Compteur synchrone – Bascule JK

- **Compteur synchrone modulo 8**
  - Toutes les bascules possèdent la même horloge.
  - A partir de la table de transition on détermine les  $J_i$  et  $K_i$

$Q_2$	$Q_1$	$Q_0$	$Q_2^+$	$Q_1^+$	$Q_0^+$	$J_2$	$K_2$	$J_1$	$K_1$	$J_0$	$K_0$
0	0	0	0	0	1	0	x	0	x	1	x
0	0	1	0	1	0	0	x	1	x	x	1
0	1	0	0	1	1	0	x	x	0	1	x
0	1	1	1	0	0	1	x	x	1	x	1
1	0	0	1	0	1	x	0	0	x	1	x
1	0	1	0	0	0	x	0	1	x	x	1
1	1	0	1	1	1	x	0	x	0	1	x
1	1	1	0	0	0	x	1	x	1	x	1

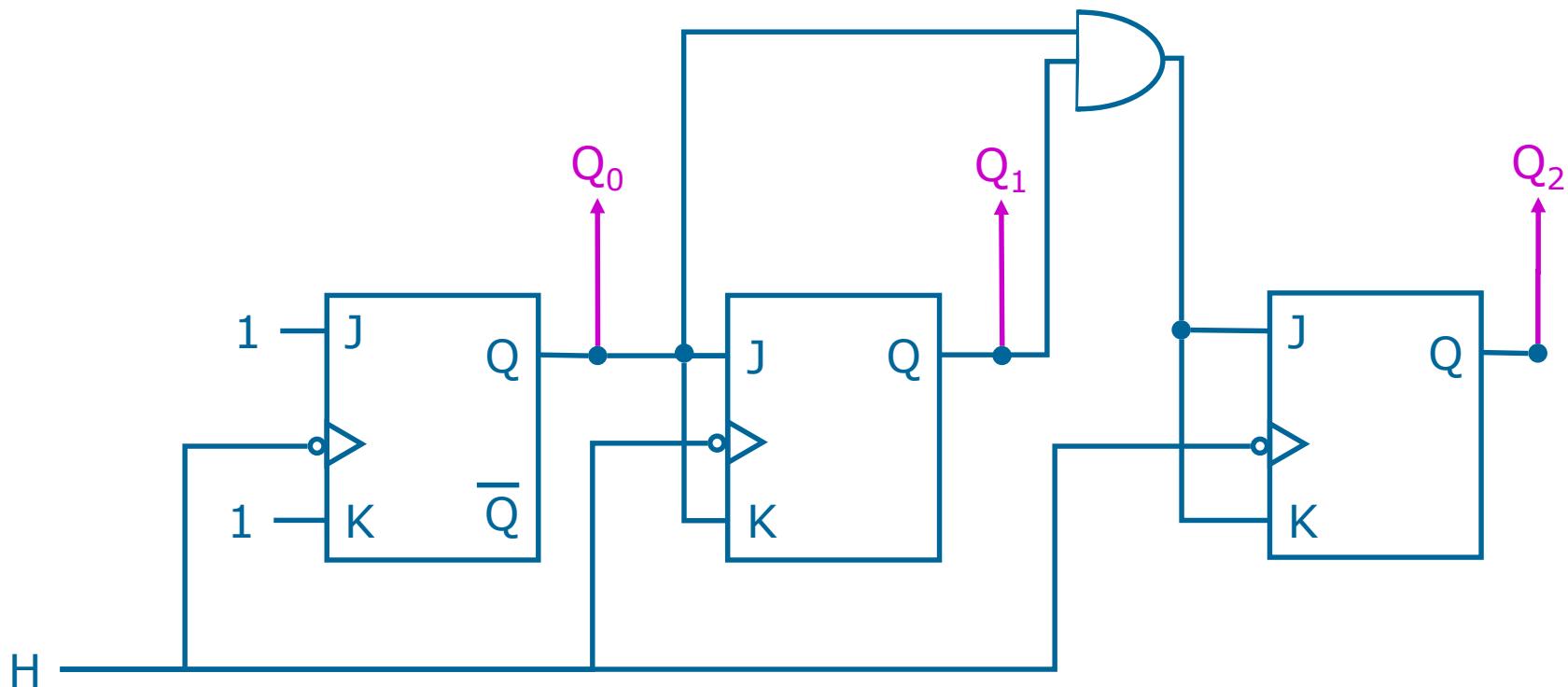
# Compteur synchrone – Bascule JK

$$J_2 = K_2 = Q_0 \cdot Q_1$$

$$J_1 = K_1 = Q_0$$

$$J_0 = K_0 = 1$$

- **Schéma**



# Compteur synchrone – Bascule D

- Compteur modulo 8

Etats précédents      Etats suivants

$Q_2$	$Q_1$	$Q_0$		$Q_2^+$	$Q_1^+$	$Q_0^+$		$D_2$	$D_1$	$D_0$
0	0	0		0	0	1		0	0	1
0	0	1		0	1	0		0	1	0
0	1	0		0	1	1		0	1	1
0	1	1		1	0	0		1	0	0
1	0	0		1	0	1		1	0	1
1	0	1		1	1	0		1	1	0
1	1	0		1	1	1		1	1	1
1	1	1		0	0	0		0	0	0

$$D_0 = \overline{Q_0}$$

$$D_1 = Q_1 \oplus Q_0$$

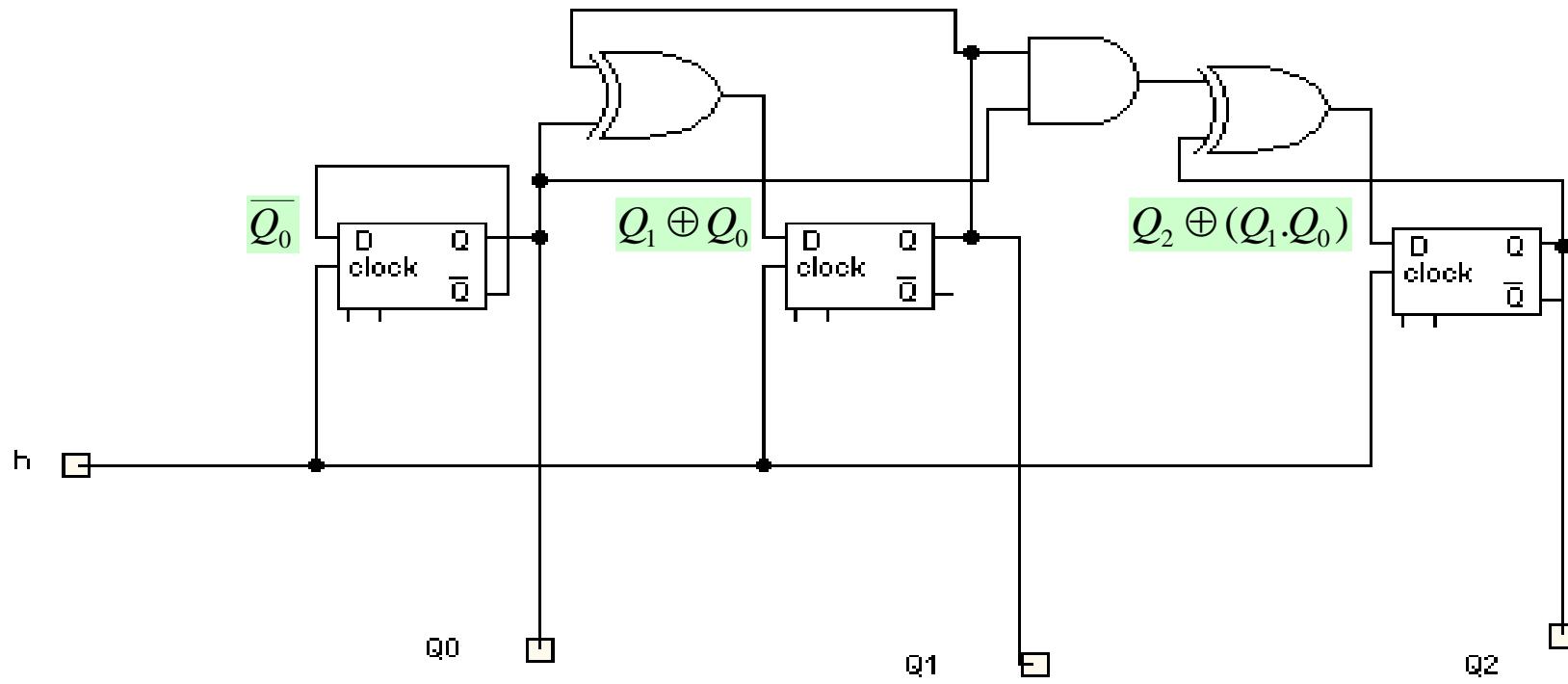
$$D_2 = Q_2 \oplus (Q_1 \cdot Q_0)$$

Tableau de Karnaugh de  $D_2$

$Q_1 Q_0$	00	01	11	10
$Q_2$	0	0	1	0
0	0	1	1	1
1	1	1	0	1

# Compteur synchrone – Bascule D

- Schéma



# Compteur synchrone modulo $2^n$ – Bascule D

- **Règle générale pour les compteurs synchrones à cycle complet avec les bascules D :**

$$D_0 = \overline{Q_0}$$

$$D_i = Q_i \oplus (Q_{i-1} \cdot Q_{i-2} \dots \cdot Q_1 \cdot Q_0) \quad i = 1, \dots, n$$

# Compteur synchrone – Bascule JK

- Compteur synchrone modulo 6

$Q_2$	$Q_1$	$Q_0$	$Q_2^+$	$Q_1^+$	$Q_0^+$	$J_2$	$K_2$	$J_1$	$K_1$	$J_0$	$K_0$
0	0	0	0	0	1	0	x	0	x	1	x
0	0	1	0	1	0	0	x	1	x	x	1
0	1	0	0	1	1	0	x	x	0	1	x
0	1	1	1	0	0	1	x	x	1	x	1
1	0	0	1	0	1	x	0	0	x	1	x
1	0	1	0	0	0	x	1	0	x	x	1
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x



→ Détection de l'état **101** et mise à zéro synchrone

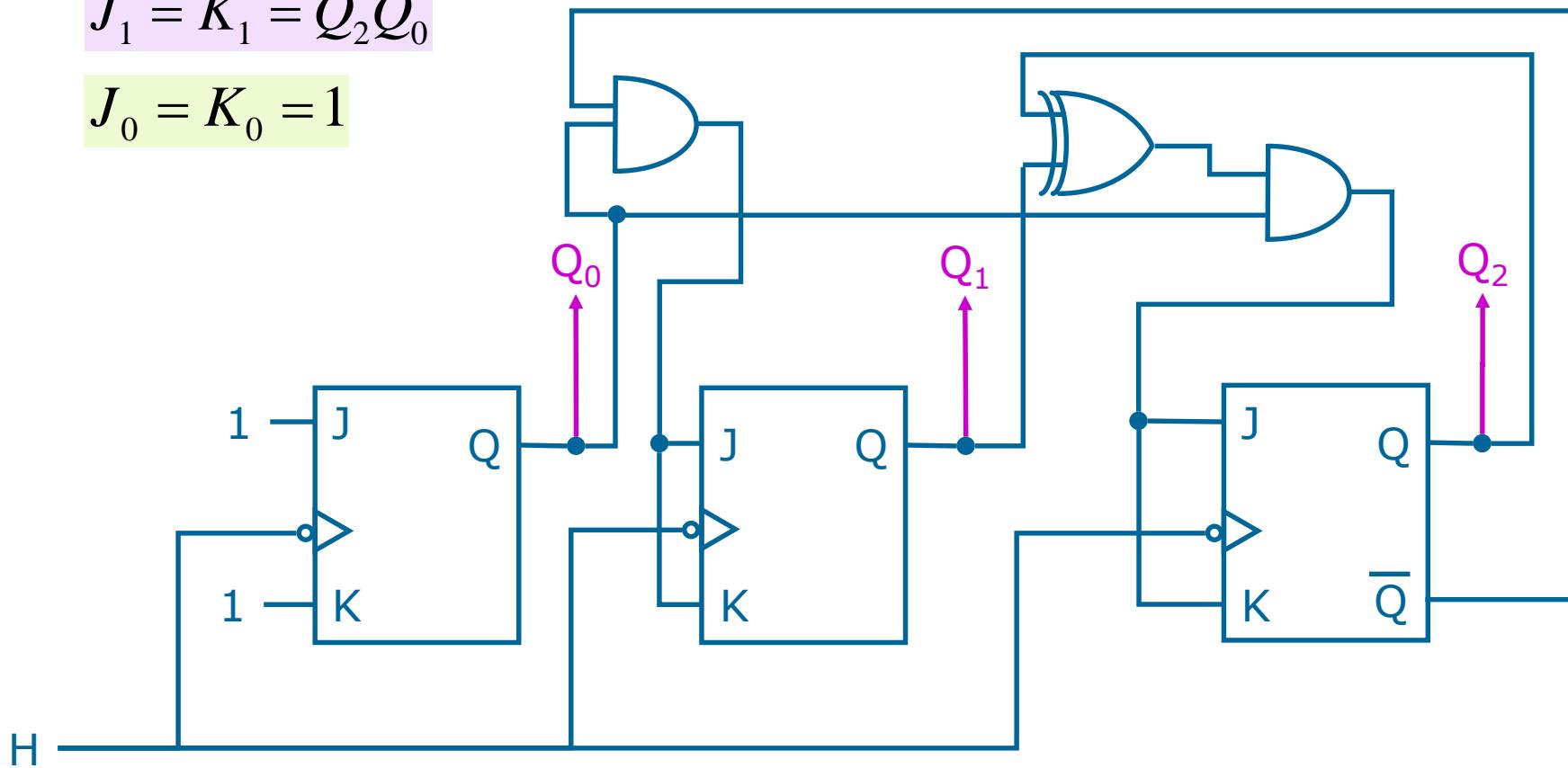
# Compteur synchrone modulo 6

- Schéma

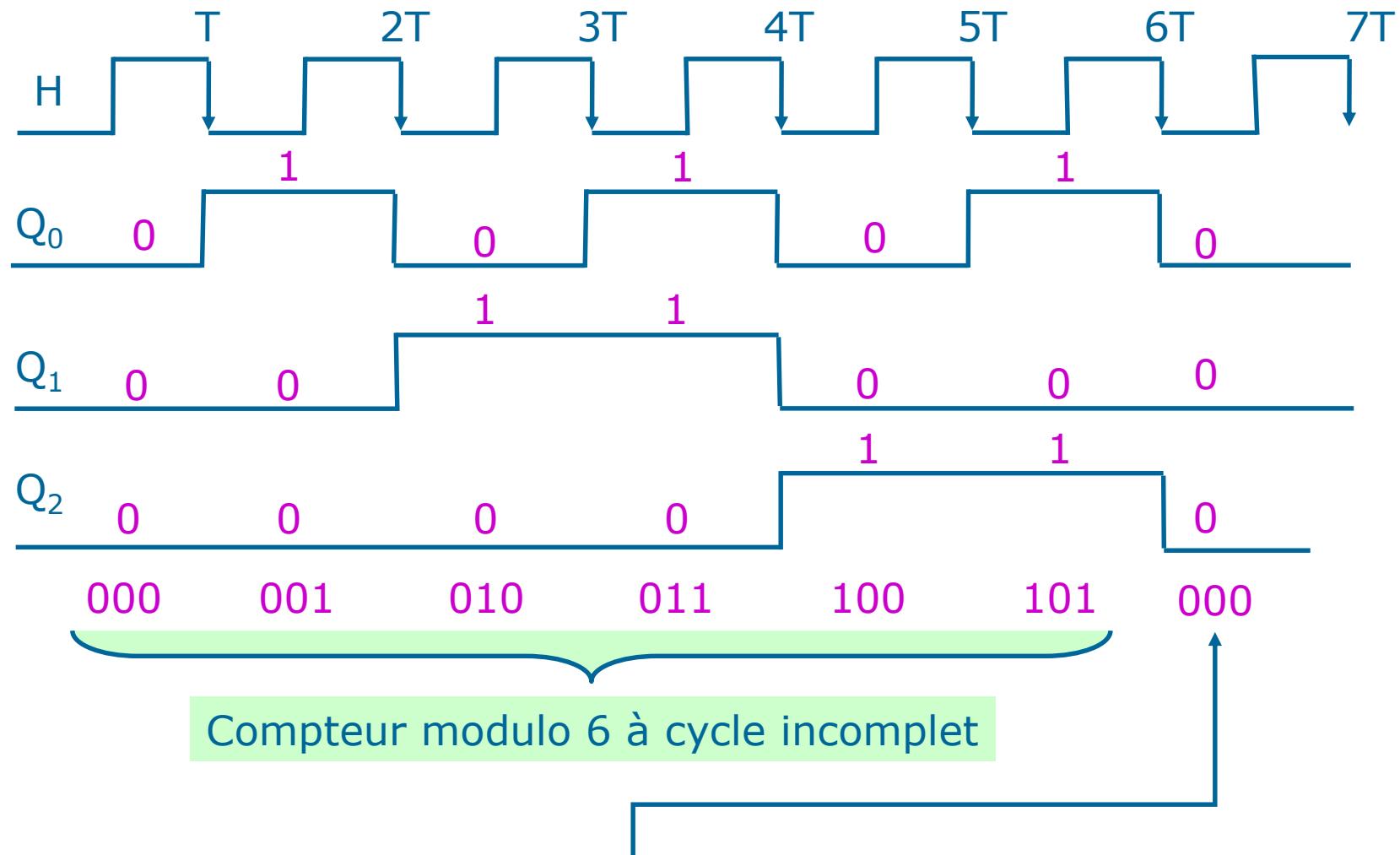
$$J_2 = K_2 = \overline{Q}_2 \cdot Q_1 \cdot Q_0 + Q_2 \cdot \overline{Q}_1 \cdot Q_0 = Q_0 \cdot (Q_2 \oplus Q_1)$$

$$J_1 = K_1 = \overline{Q}_2 Q_0$$

$$J_0 = K_0 = 1$$



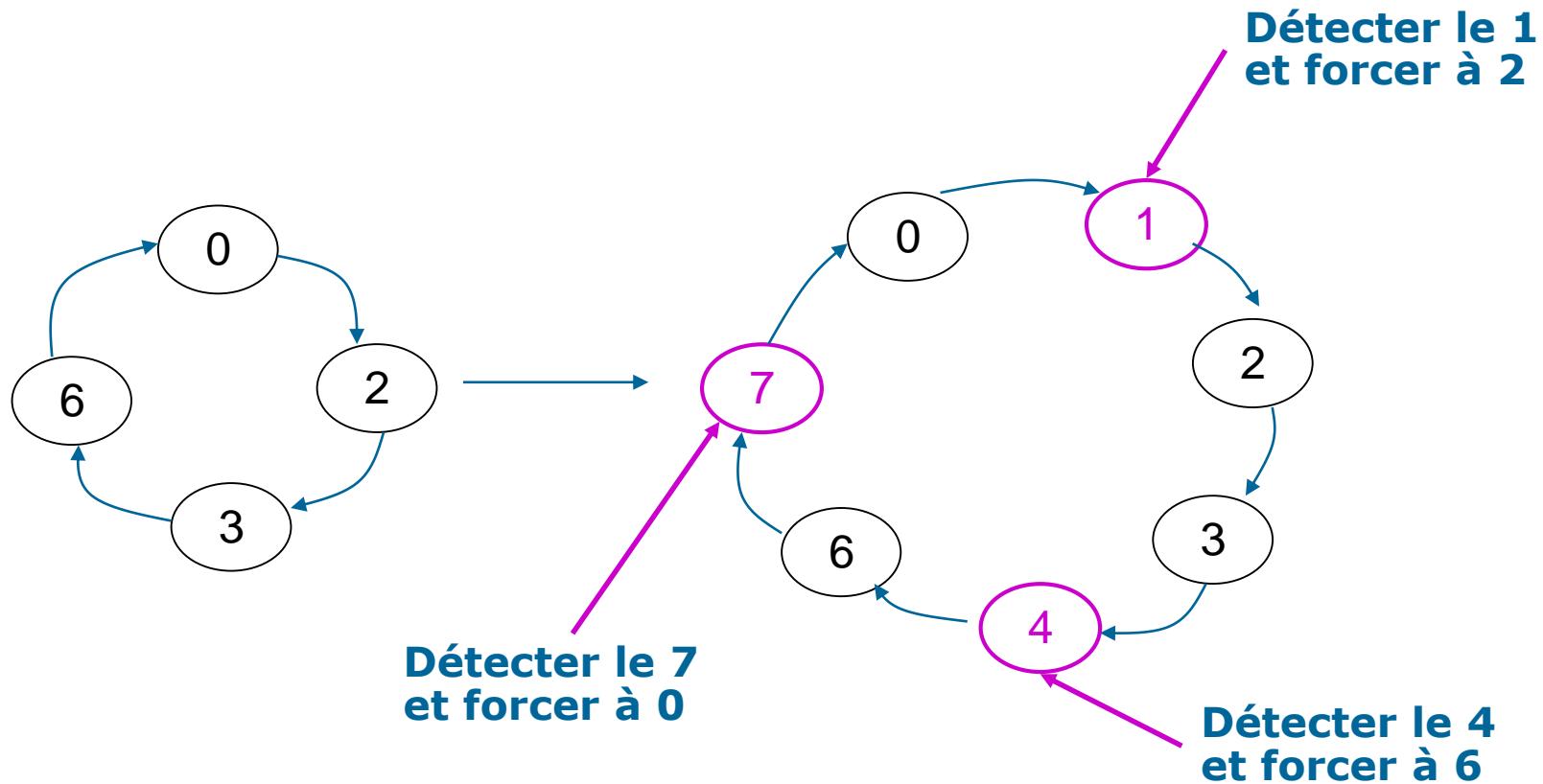
# Compteur synchrone modulo 6



Détection de l'état **101** et mise à **0** synchrone des bascules.

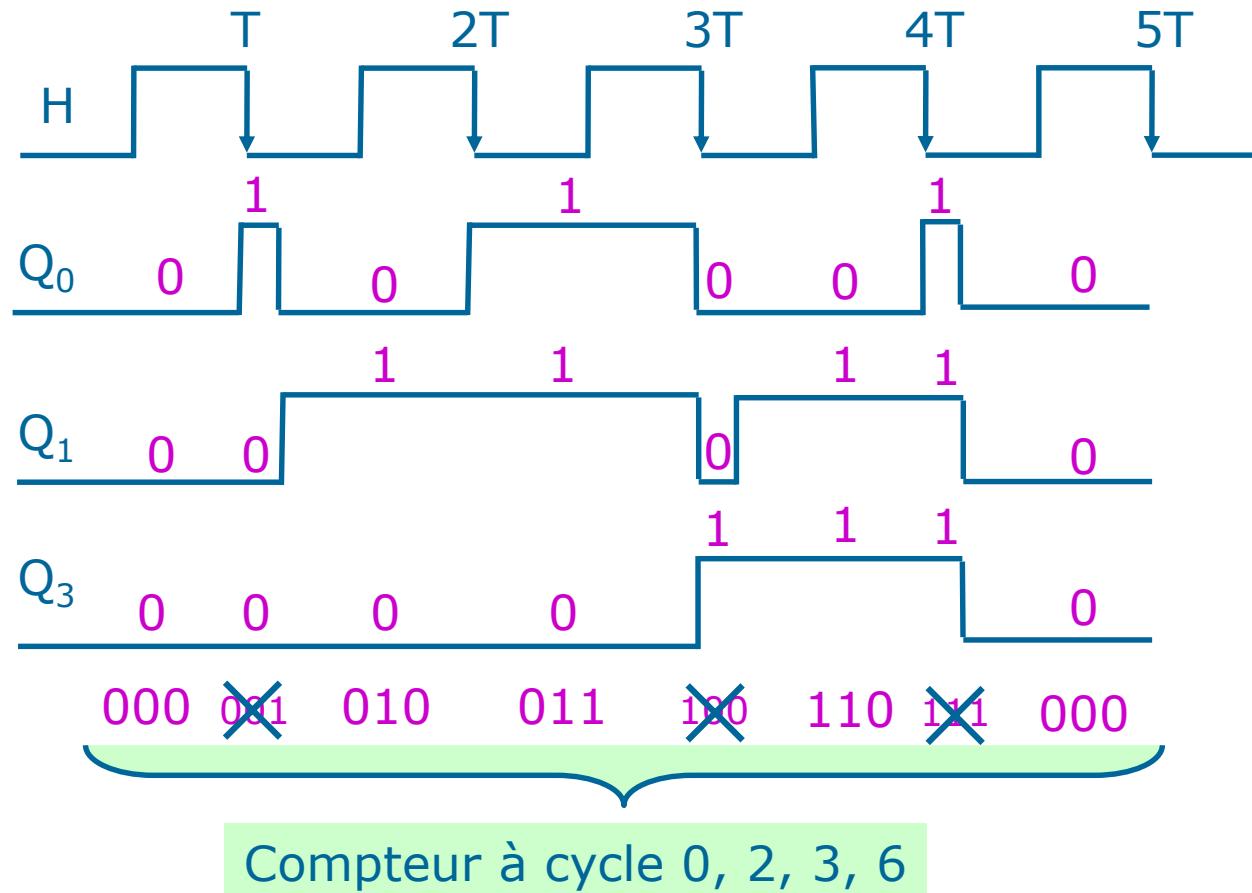
# Compteur asynchrone à cycle quelconque

- Soit un compteur asynchrone ayant le cycle suivant :



- Pour forcer le compteur d'un état à un autre, il faut agir sur les entrées asynchrones Cli et Pri des bascules.

# Compteur asynchrone à cycle quelconque



# Compteur asynchrone à cycle quelconque

$Q_2$	$Q_1$	$Q_0$	$Q_2^+$	$Q_1^+$	$Q_0^+$	$Pr_2$	$Cl_2$	$Pr_1$	$Cl_1$	$Pr_0$	$Cl_0$
0	0	0	0	0	1	1	1	1	1	1	1
0	0	1	0	1	0	1	1	0	1	1	0
0	1	0	0	1	1	1	1	1	1	1	1
0	1	1	1	0	0	1	1	1	1	1	1
1	0	0	1	1	0	1	1	0	1	1	1
1	1	0	1	1	1	1	1	1	1	1	1
1	1	1	0	0	0	1	0	1	0	1	0

$$Pr_2 = 1$$

$$\overline{Cl_2} = Q_2 \cdot Q_1 \cdot Q_0$$

$$\overline{Pr_1} = \overline{Q_2} \cdot \overline{Q_1} \cdot Q_0 + Q_2 \cdot \overline{Q_1} \cdot \overline{Q_0}$$

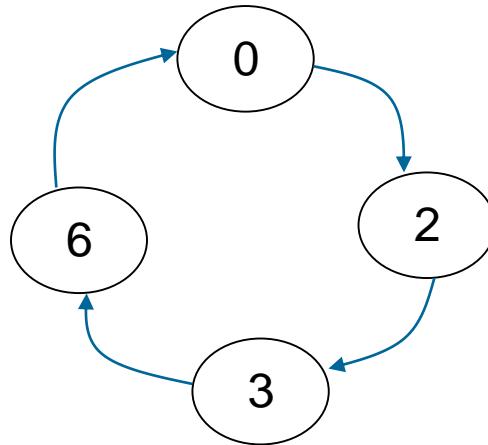
$$\overline{Cl_1} = Q_2 \cdot Q_1 \cdot Q_0$$

$$Pr_0 = 1$$

$$\overline{Cl_0} = \overline{Q_2} \cdot \overline{Q_1} \cdot Q_0 + Q_2 \cdot Q_1 \cdot \overline{Q_0}$$

# Compteur synchrone à cycle quelconque

- Soit un compteur synchrone ayant le cycle suivant :



- Pour forcer le compteur d'un état à un autre, il faut agir sur les entrées synchrones  $J_i$  et  $K_i$
- Pour les états qui n'appartiennent pas au cycle du compteur, il faut les considérer comme étant des états indéterminés ( $J_i=X$  et  $K_i=X$ )

# Compteur synchrone à cycle quelconque

$Q_2$	$Q_1$	$Q_0$	$Q_2^+$	$Q_1^+$	$Q_0^+$	$J_2$	$K_2$	$J_1$	$K_1$	$J_0$	$K_0$
0	0	0	0	1	0	0	X	1	X	0	X
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	1	0	1	X	X	0	X	1
1	1	0	0	0	0	X	1	X	1	0	X
0	0	1	X	X	X	X	X	X	X	X	X
1	0	0	X	X	X	X	X	X	X	X	X
1	0	1	X	X	X	X	X	X	X	X	X
1	1	1	X	X	X	X	X	X	X	X	X

$$J_2 = K_2 = Q_0 \cdot Q_1 \cdot \overline{Q_2} + \overline{Q_0} \cdot Q_1 \cdot Q_2 = Q_1 \cdot (Q_2 \oplus Q_0)$$

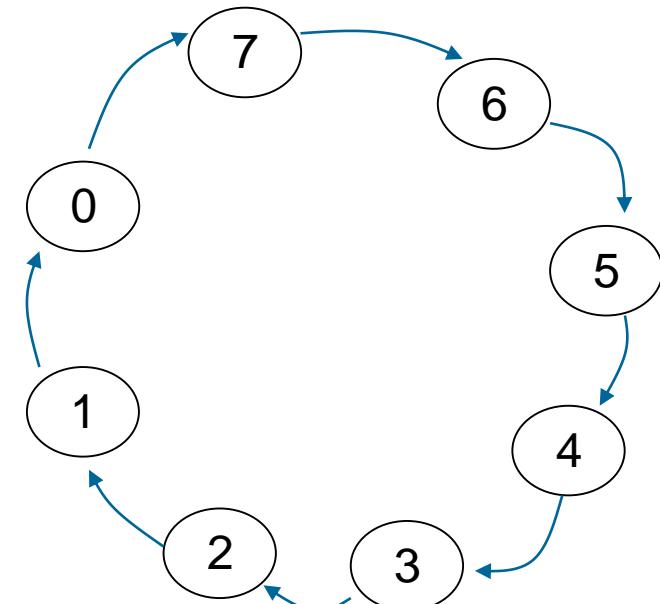
$$J_1 = K_1 = \overline{Q_0} \cdot \overline{Q_1} \cdot \overline{Q_2} + \overline{Q_0} \cdot Q_1 \cdot Q_2 = \overline{Q_0} \cdot (Q_2 \oplus Q_1)$$

$$J_0 = K_0 = \overline{Q_2} \cdot Q_1 \cdot \overline{Q_0} + \overline{Q_2} \cdot Q_1 \cdot Q_0 = \overline{Q_2} \cdot Q_1$$

# Décompteur

- L'étude des décompteurs se fait exactement de la même manière que l'étude des compteurs.
- Exemple d'une table de transition :

$Q_2$	$Q_1$	$Q_0$		$Q_2^+$	$Q_1^+$	$Q_0^+$
1	1	1		1	1	0
1	1	0		1	0	1
1	0	1		1	0	0
1	0	0		0	1	1
0	1	1		0	1	0
0	1	0		0	0	1
0	0	1		0	0	0
0	0	0		1	1	1



# Décompteur synchrone – Bascule JK

$Q_2$	$Q_1$	$Q_0$	$Q_2^+$	$Q_1^+$	$Q_0^+$	$J_2$	$K_2$	$J_1$	$K_1$	$J_0$	$K_0$
1	1	1	1	1	0	x	0	x	0	x	1
1	1	0	1	0	1	x	0	x	1	1	x
1	0	1	1	0	0	x	0	0	x	x	1
1	0	0	0	1	1	x	1	1	x	1	x
0	1	1	0	1	0	0	x	x	0	x	1
0	1	0	0	0	1	0	x	x	1	1	x
0	0	1	0	0	0	0	x	0	x	x	1
0	0	0	1	1	1	1	x	1	x	1	x

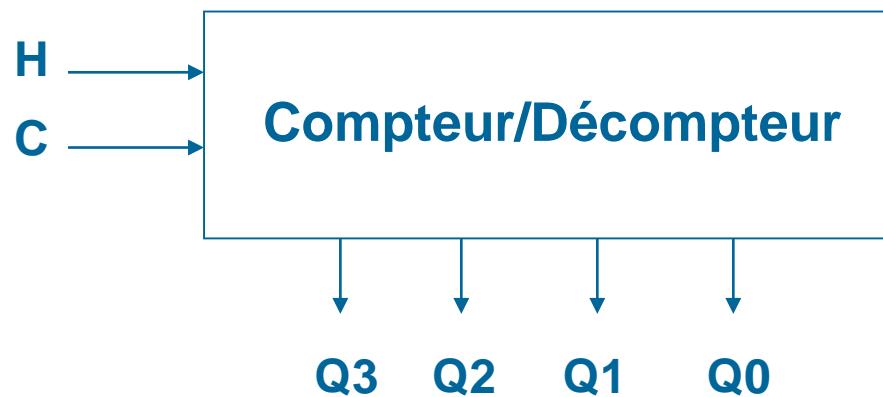
$$J_2 = K_2 = \overline{Q_0} \cdot \overline{Q_1}$$

$$J_1 = K_1 = \overline{Q_0}$$

$$J_0 = K_0 = 1$$

# Compteurs/Décompteurs

- **Le même circuit peut offrir l'opération du comptage et du décomptage**
- **Rajouter une entrée C :**
  - $C=0$  alors comptage
  - $C=1$  alors décomptage



# Exemple d'un compteur/décompteur

- **Résultat du compteur synchrone modulo 8 (page 53) :**

$$J_2 = K_2 = Q_0 \cdot Q_1$$

$$J_1 = K_1 = Q_0$$

$$J_0 = K_0 = 1$$

- **Résultat du décompteur synchrone modulo 8 (page 66) :**

$$J_2 = K_2 = \overline{Q_0} \cdot \overline{Q_1}$$

$$J_1 = K_1 = \overline{Q_0}$$

$$J_0 = K_0 = 1$$

- **Avec l'entrée de commande, on regroupe les deux séries d'expression :**

- C=0 → compteur
- C=1 → décompteur

$$J_2 = K_2 = \overline{C} \cdot Q_0 \cdot Q_1 + C \cdot \overline{Q_0} \cdot \overline{Q_1}$$

$$J_1 = K_1 = \overline{C} \cdot Q_0 + C \cdot \overline{Q_0}$$

$$J_0 = K_0 = 1$$

# Résumé de la synthèse des compteurs

## Compteur synchron

Cycle comp.  
 $2^n$

### Bascule D

- Table de transition
- Préparer les expressions des  $D_i$

Cycle incomp.  
 $N < 2^n$

- Table de transition
- Déetecter l'état avant l'état interdit
- Remise à 0 synchrone dans la table de transition
- Préparer les expressions des  $D_i$

### Bascule JK

- Table de transition
- Préparer les expressions des  $J_i$  et  $K_i$

## Compteur asynchrone

Cycle comp.  
 $2^n$

- n bascules montées en diviseur de fréquence ( $\bar{Q}_i \rightarrow D_i$ )
- $|Q_i \rightarrow H_{i+1}$

Cycle incomp.  
 $N < 2^n$

- n bascules montées en diviseur de fréquence ( $\bar{Q}_i \rightarrow D_i$ )
- $|Q_i \rightarrow H_{i+1}$
- Déetecter l'état interdit
- Remise à 0 asynchrone en utilisant les entrées asynchrones

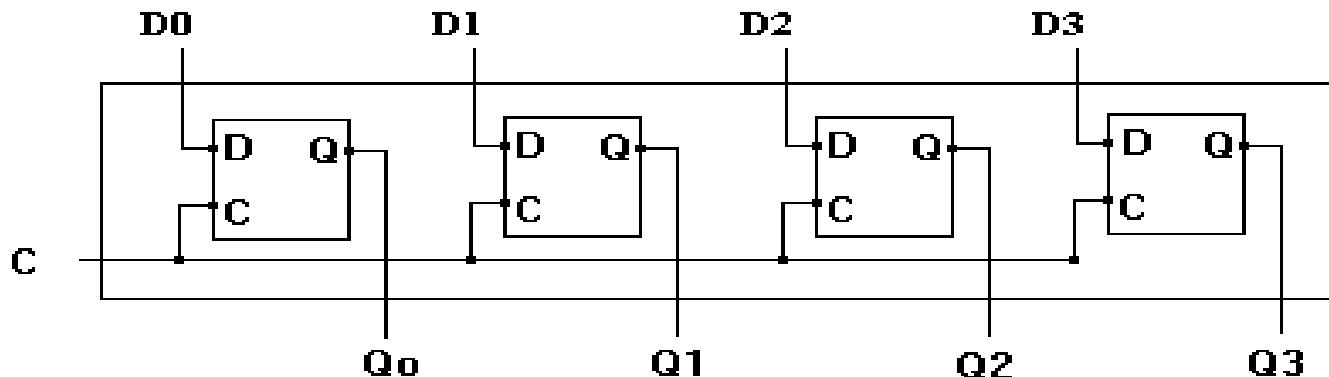
- n bascules ( $J_i = K_i = 1$ )
- $Q_i \rightarrow H_{i+1}$

- n bascules ( $J_i = K_i = 1$ )
- $Q_i \rightarrow H_{i+1}$
- Déetecter l'état interdit
- Remise à 0 asynchrone en utilisant les entrées asynchrones

# Registres

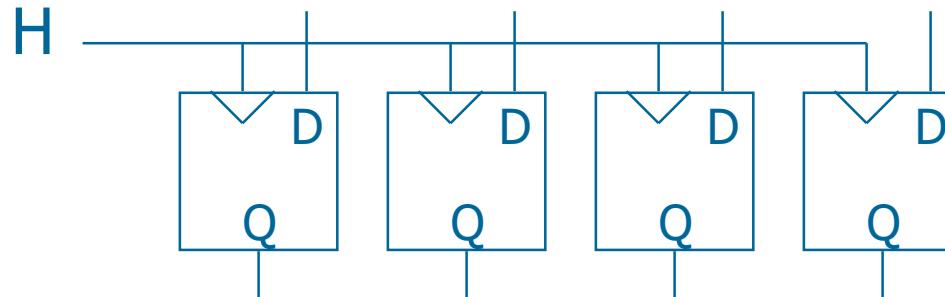
# Définition

- Une bascule permet de mémoriser un seul bit.
- Un registre est un ensemble ordonné de **n** bascules.
- Un registre permet de mémoriser (sauvegarder) une information sur **n** bits.
- Exemple :
  - ◆ Registre à 4 bits

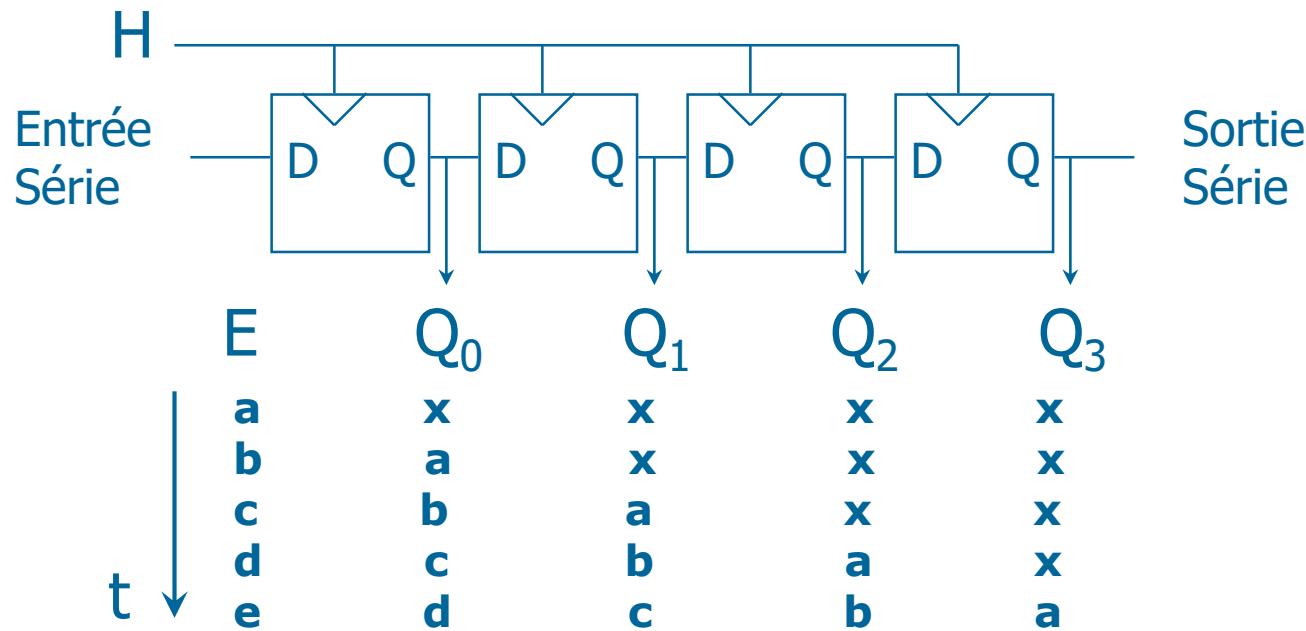


# Définition

- **Registre : ensemble de bascules D avec la même horloge sensible au front d'horloge**



- **Registre à décalage : registre avec l'entrée D de chaque bascule connectée à la sortie de la bascule précédente**



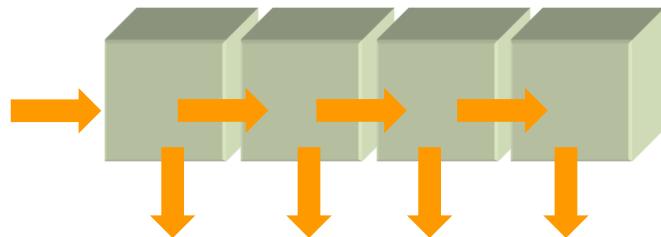
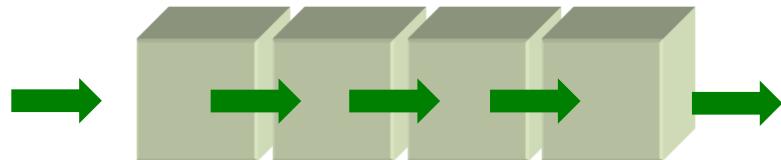
# Types de registre

- **Il existe plusieurs types de registre :**
  - ◆ Registre à entrée série et sortie série. (SISO)
  - ◆ Registre à entrée série et sortie parallèle. (SIPO)
  - ◆ Registre à entrée parallèle et sortie série. (PISO)
  - ◆ Registre à entrées parallèles et sorties parallèles. (PIPO)
  - ◆ Registre à décalage circulaire.

# Types de registre

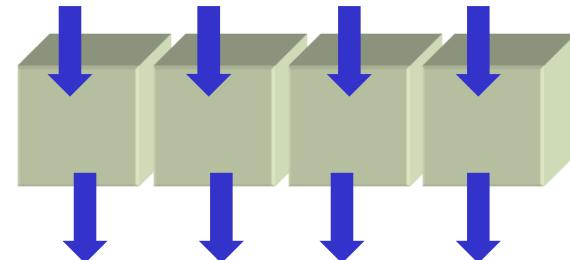
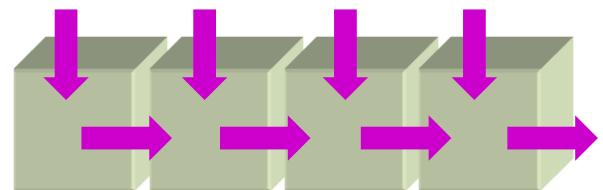
- **Chargement**

Sortie	Entrée	Série	Parallèle
Série		SISO	PISO
Parallèle		SIPO	PIPO



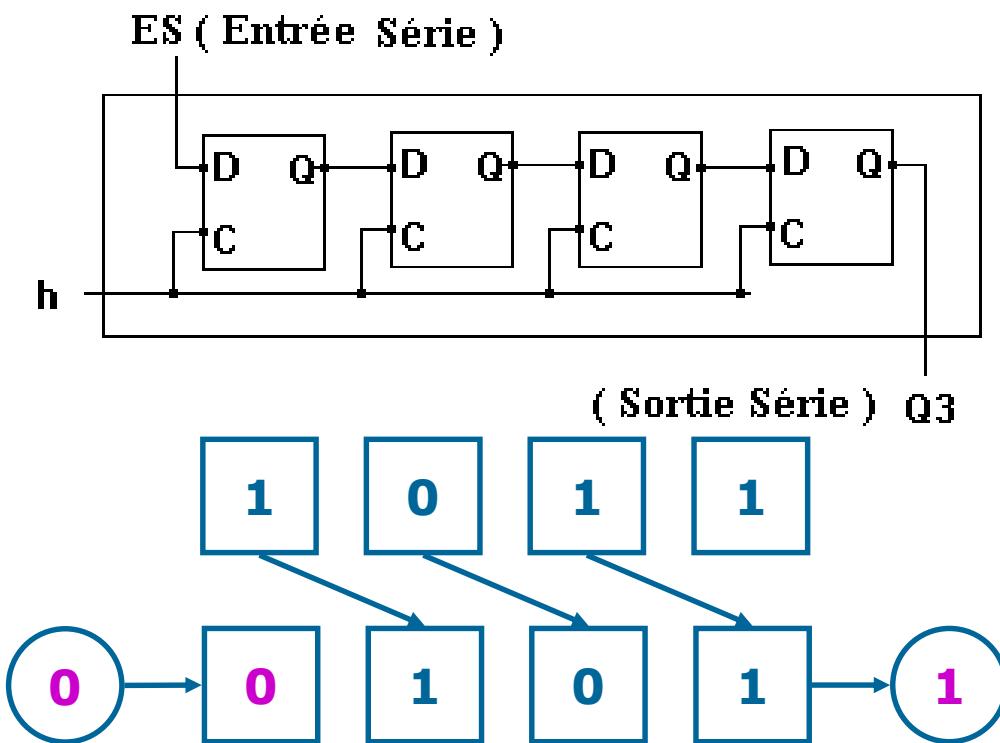
- **Déplacement des données**

- Décalage G/D
- Rotation G/D



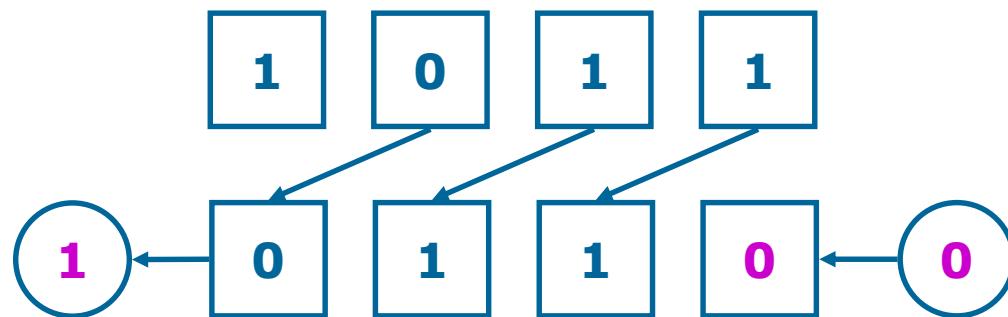
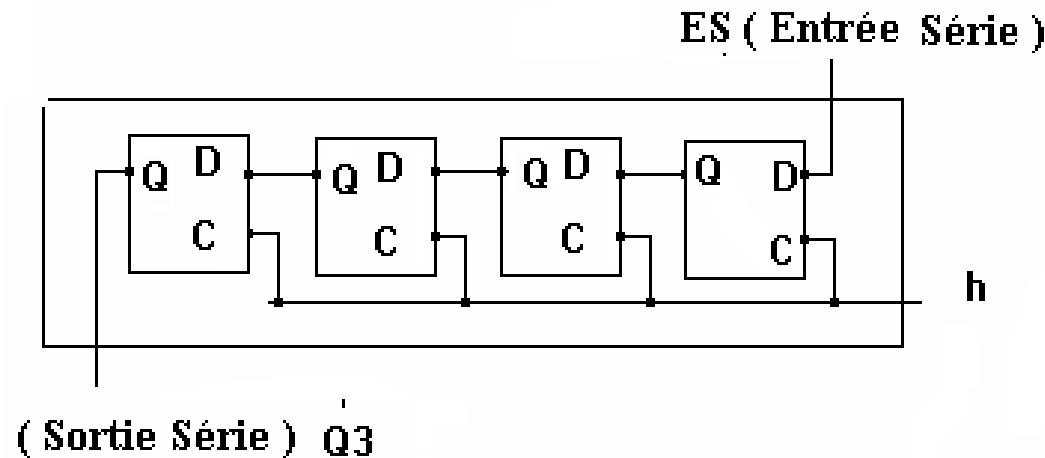
# Registre à entrée série et sortie série (1)

- L'information est introduite bit par bit, en série.
- L'ensemble du registre est décalé d'une position ( $B_i \rightarrow B_{i+1}$ ) et la bascule  $B_0$  reçoit un nouveau bit de l'entrée ES.
- Un tel registre est appelé registre à entrée série à gauche et à sortie série à droite.

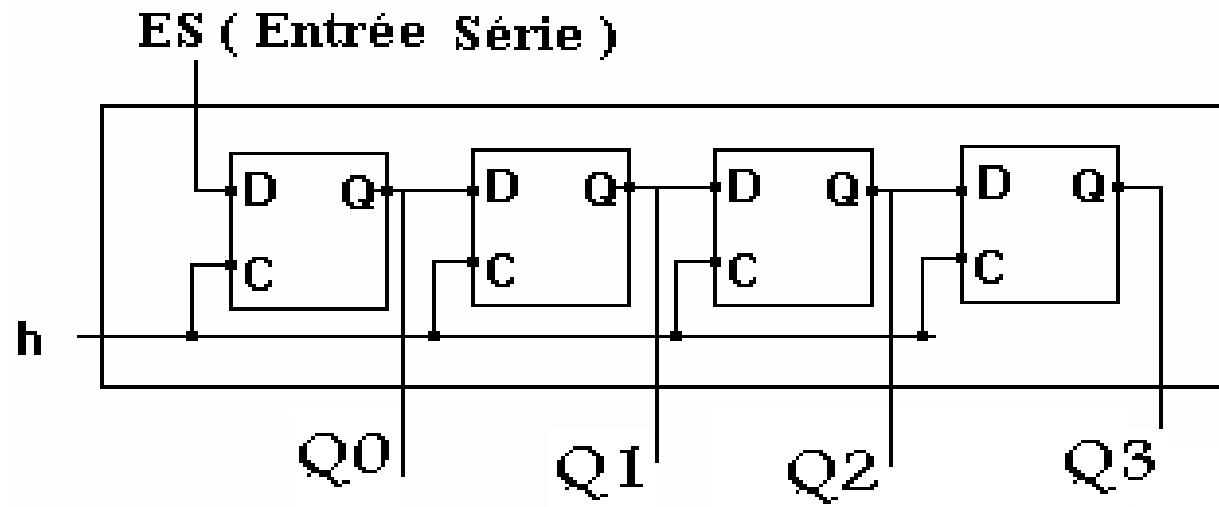


# Registre à entrée série et sortie série (2)

- Un tel registre est appelé **registre à entrée série à droite et à sortie série à gauche**.

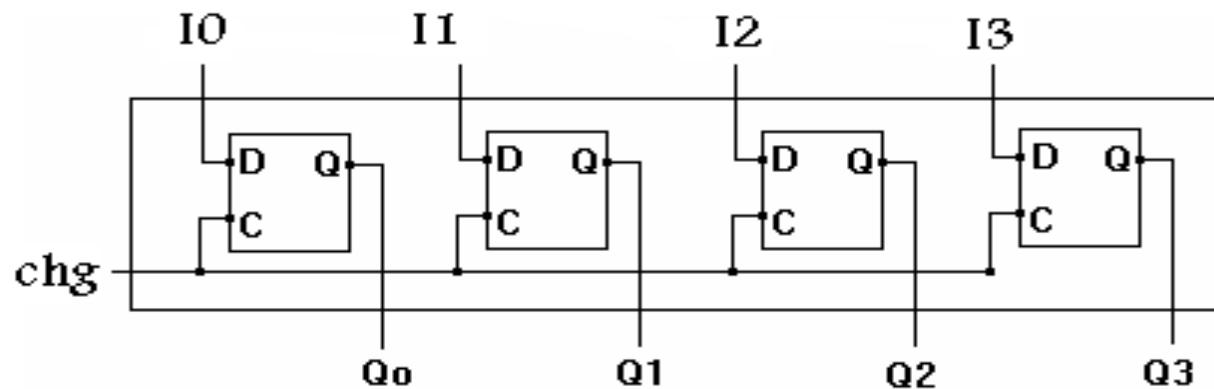


# Registre à entrée série et sortie parallèle



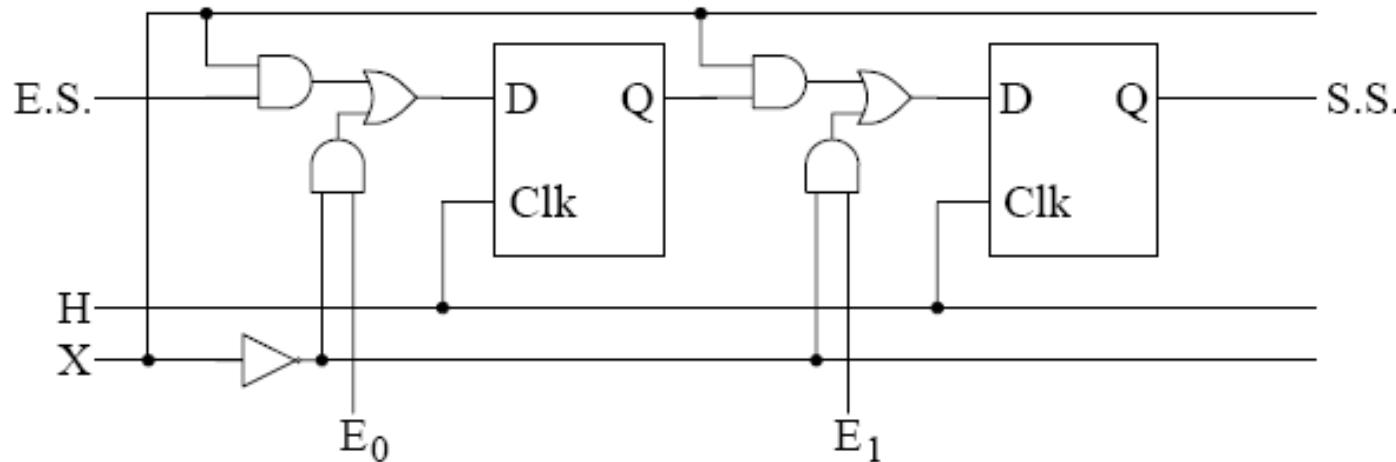
# Registre à entrées parallèles et sorties parallèles

- Il peut charger une information sur **n bits en même temps.**
- Les **n bascules** changement d'états en même temps.
- Chaque bascule  $B_i$  prend la valeur de l'information  $I_i$ .
- Il possède une entrée de chargement (ici l'horloge des bascules D sensibles au niveau) :
  - ◆ chg=0 état mémoire
  - ◆ chg=1 chargement



# Registre à entrées séries ou parallèles et sortie série

- **E.S. : Entrée série**
- **E0 et E1 : Entrées parallèles**
- **X : Signal de contrôle**



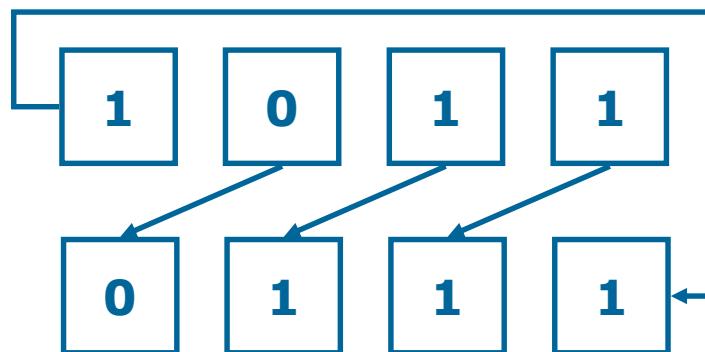
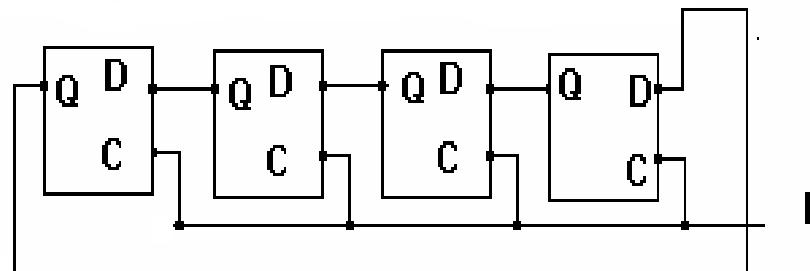
$$D_0 = E.S \cdot X + E_0 \cdot \bar{X}$$

$$D_1 = Q_0 \cdot X + E_0 \cdot \bar{X}$$

- **Selon la valeur de X, le registre utilise comme source son entrée série ou ses entrées parallèles.**

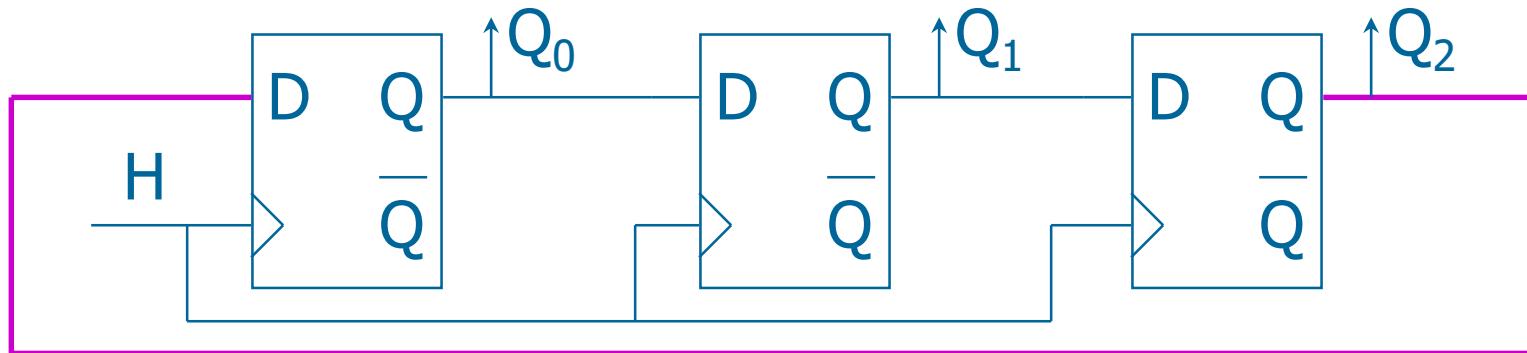
# Registre à décalage circulaire

- C'est un registre qui effectue un décalage en répercutant la sortie de la dernière bascule vers l'entrée de la première bascule.
- Le décalage peut être un décalage droite (circulaire droite) ou gauche (circulaire gauche).



# Compteur en anneau

- **Registre à décalage circulaire**



- n bascules : cycle de comptage de longueur n
- Avantage : Simple et rapide
- Les cycles :

→ 000  
□

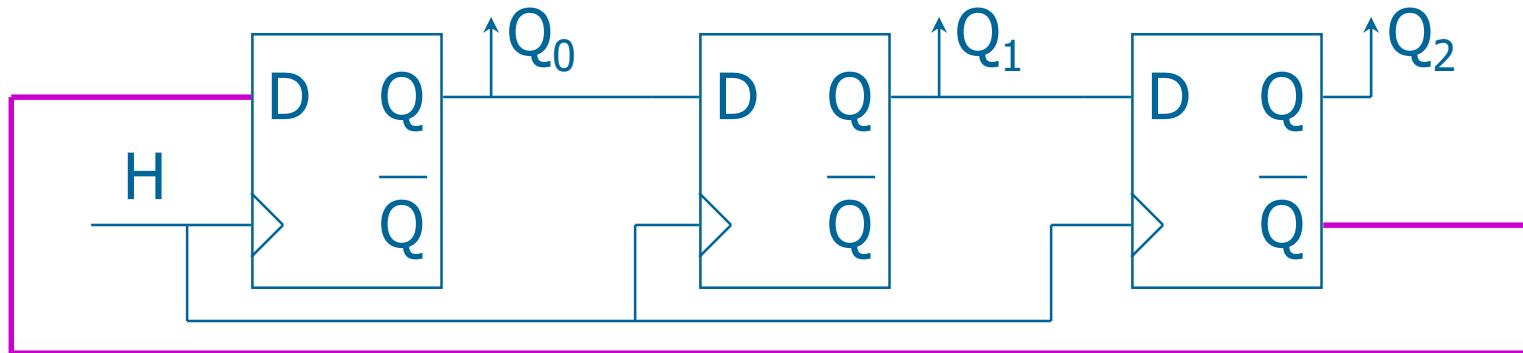
→ 100  
010  
001  
□

→ 110  
011  
101  
□

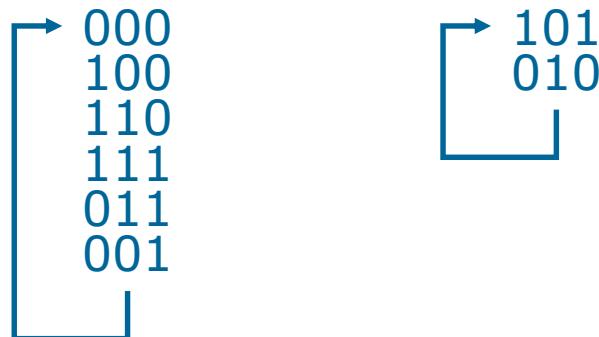
→ 111  
□

# Compteur Johnson

- **Registre à décalage circulaire par inversion**



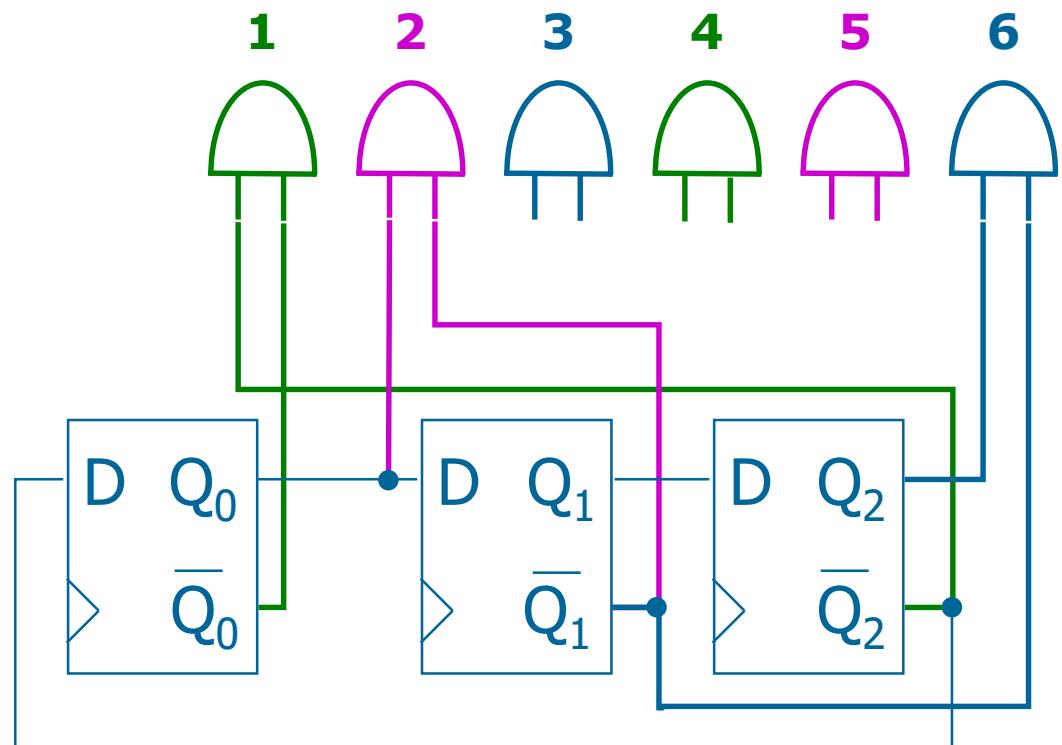
- ◆  $n$  bascules : Cycles de comptage de longueur  $2^n$
- ◆ Avantage : Simple et rapide
- ◆ Les cycles :



# Compteur Johnson avec les portes logiques

- Le but est d'avoir  $2n$  sorties qui s'allume chacune à leur tour
- Les  $n$  sorties des bascules sont combinées avec les portes ET

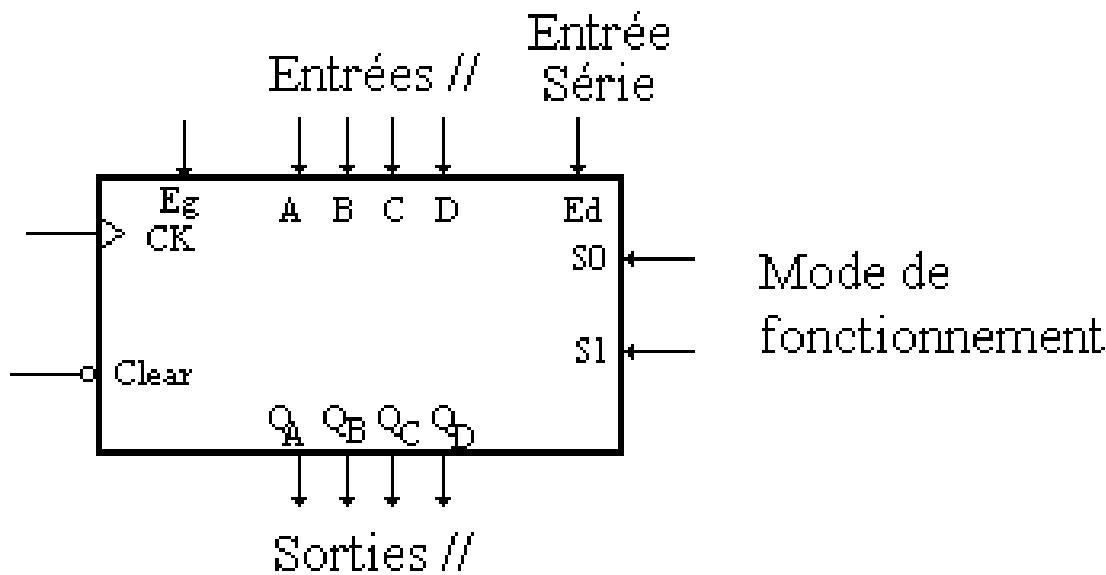
$Q_0$	$Q_1$	$Q_2$	Porte ET
0	0	0	$\overline{Q_0} \overline{Q_2}$
1	0	0	$Q_0 \overline{Q_1}$
1	1	0	$\overline{Q_1} \overline{Q_2}$
1	1	1	$Q_0 Q_1$
0	1	1	$\overline{Q_0} Q_1$
0	0	1	$\overline{Q_1} Q_2$



- L'inconvénient est de tomber dans le cycle secondaire :  
 $(101 \rightarrow 010)$

# Registres programmables

- Il existe des registres qui permettent :
  - le décalage à droite (circulaire droite),
  - le décalage à gauche (circulaire gauche),
  - le chargement parallèle.



# Registres programmables

- **Table de vérité**

<b>h</b>	<b>S<sub>0</sub></b>	<b>S<sub>1</sub></b>	<b>Q<sub>A</sub></b>	<b>Q<sub>B</sub></b>	<b>Q<sub>C</sub></b>	<b>Q<sub>D</sub></b>	
<b>X</b>	<b>0</b>	<b>0</b>	<b>Q<sub>A</sub></b>	<b>Q<sub>B</sub></b>	<b>Q<sub>C</sub></b>	<b>Q<sub>D</sub></b>	<b>Mémoire</b>
<b>↑</b>	<b>0</b>	<b>1</b>	<b>Eg</b>	<b>Q<sub>A</sub></b>	<b>Q<sub>B</sub></b>	<b>Q<sub>C</sub></b>	<b>Décalage à droite</b>
<b>↑</b>	<b>1</b>	<b>0</b>	<b>Q<sub>B</sub></b>	<b>Q<sub>C</sub></b>	<b>Q<sub>D</sub></b>	<b>Ed</b>	<b>Décalage à gauche</b>
<b>↑</b>	<b>1</b>	<b>1</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>Chargement Synchrone</b>

# Exercice

- Réalisation d'un registre avec le tableau de fonctionnement suivant en utilisant les bascules D :

RAZ	Chg	Dg	Dd	H	$Q_3+$	$Q_2+$	$Q_1+$	$Q_0+$
1	X	X	X	X	0	0	0	0
0	1	X	X	X	$E_3$	$E_2$	$E_1$	$E_0$
0	0	1	X	↑	$Q_2$	$Q_1$	$Q_0$	$S_L$
0	0	0	1	↑	$S_R$	$Q_3$	$Q_2$	$Q_1$
0	0	0	0	X	$Q_3$	$Q_2$	$Q_1$	$Q_0$

# Solution – Entrées asynchrones

Raz	Chg	Dg	Dd	H	Q <sub>3</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	Pr <sub>i</sub>	Cl <sub>i</sub>
1	X	X	X	X	0	0	0	0	1	0
0	1	X	X	X	E <sub>3</sub>	E <sub>2</sub>	E <sub>1</sub>	E <sub>0</sub>	—E <sub>i</sub>	E <sub>i</sub>

$$Pr_i = \overline{Raz}.\overline{Chg} + Raz + \overline{Raz}.Chg.\overline{E}_i = Raz + \overline{Chg} + \overline{E}_i$$

$$Cl_i = \overline{Raz}.\overline{Chg} + \overline{Raz}.Chg.E_i = \overline{Raz}.\overline{Chg} + \overline{Pr}_i = \overline{\overline{Raz}.\overline{Chg}}.Pr_i$$



Pour que les entrées asynchrones Pr et Cl soient désactivées pendant le fonctionnement synchrone.

# Solution – Entrées synchrones

Raz	Chg	Dg	Dd	H	$Q_3+$	$Q_2+$	$Q_1+$	$Q_0+$	$D_3$	$D_2$	$D_1$	$D_0$
0	0	1	X	↑	$Q_2$	$Q_1$	$Q_0$	$S_L$	$Q_2$	$Q_1$	$Q_0$	$S_L$
0	0	0	1	↑	$S_R$	$Q_3$	$Q_2$	$Q_1$	$S_R$	$Q_3$	$Q_2$	$Q_1$
0	0	0	0	X	$Q_3$	$Q_2$	$Q_1$	$Q_0$	$Q_3$	$Q_2$	$Q_1$	$Q_0$

$$D_3 = D_g \cdot Q_2 + \overline{D_g} \cdot D_d \cdot S_R + \overline{D_g} \cdot \overline{D_d} \cdot Q_3$$

$$D_2 = D_g \cdot Q_1 + \overline{D_g} \cdot D_d \cdot Q_3 + \overline{D_g} \cdot \overline{D_d} \cdot Q_2$$

$$D_1 = D_g \cdot Q_0 + \overline{D_g} \cdot D_d \cdot Q_2 + \overline{D_g} \cdot \overline{D_d} \cdot Q_1$$

$$D_0 = D_g \cdot S_L + \overline{D_g} \cdot D_d \cdot Q_1 + \overline{D_g} \cdot \overline{D_d} \cdot Q_0$$