

Logique séquentielle

Introduction

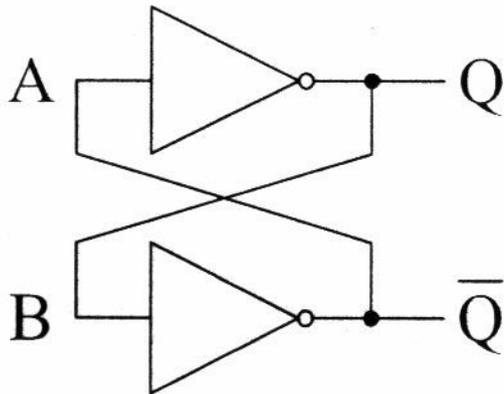
- **circuits de logique séquentielle** : circuits dans lesquels le temps intervient dans la définition des sorties
- pour un système dont l'état est noté Q , les entrées notées X et les sorties Y , on a de façon générale
 - $Q = f(Q, X)$
 - $Y = g(X, Q)$
- **logique séquentielle asynchrone** : les changements d'état des composants ne dépendent que du temps de réaction des composants et du temps de propagation des signaux
- **logique séquentielle synchrone** : les signaux périodiques d'une horloge servent à synchroniser tous les changements d'état
- un système séquentiel "garde la mémoire" du passé pour déterminer son état présent

Plan du cours

- les bascules
- les registres
- les compteurs

Les bascules

- bascule (flip-flop) : système permettant de mémoriser une information élémentaire
 - mémoire à 1 bit ayant 2 états : Q et \bar{Q}
 - utilise un mécanisme de verrou (latch)

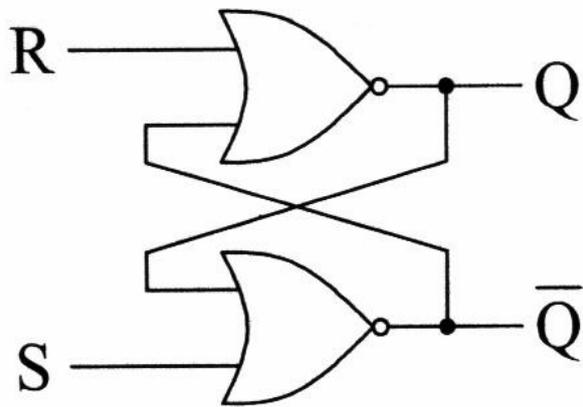


- on peut vérifier que les états sont cohérents
 - $(Q=1) \Rightarrow (B=1) \Rightarrow (\bar{Q}=0) \Rightarrow (A=0) \Rightarrow (Q=1)$
 - $(Q=0) \Rightarrow (B=0) \Rightarrow (\bar{Q}=1) \Rightarrow (A=1) \Rightarrow (Q=0)$
- 2 états seulement : "0" ($Q=0, \bar{Q}=1$) et "1" ($Q=1, \bar{Q}=0$): bistable

Les bascules RS

- Bascules R-S

- réalisées avec des portes NOR ou NAND
- portes NOR



R	S	Q	\bar{Q}
0	0	Q	\bar{Q}
0	1	1	0
1	0	0	1
1	1	0	0

sorties inchangées

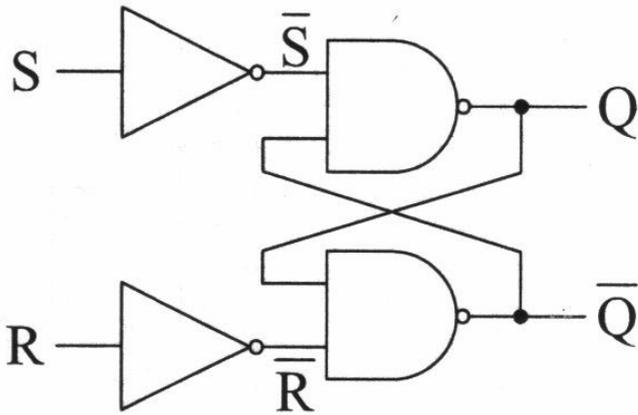
Set : remise à 1

Reset : remise à 0

à proscrire

Les bascules RS

➤ portes NAND



R	S	\bar{R}	\bar{S}	Q	\bar{Q}
0	0	1	1	Q	\bar{Q}
0	1	1	0	1	0
1	0	0	1	0	1
1	1	0	0	0	0

sorties inchangées

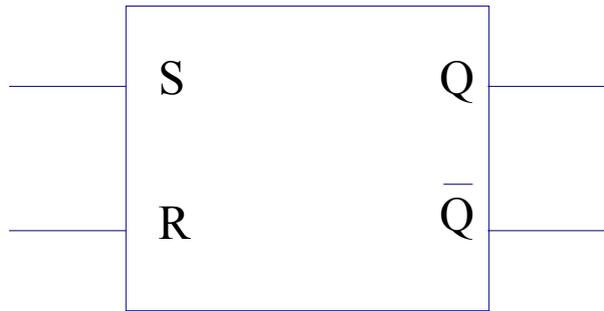
Set : remise à 1

Reset : remise à 0

à proscrire

Les bascules RS

- représentation générique



R	S	Q	\bar{Q}
0	0	Q	\bar{Q}
0	1	1	0
1	0	0	1
1	1	0	0

sorties inchangées

Set : remise à 1

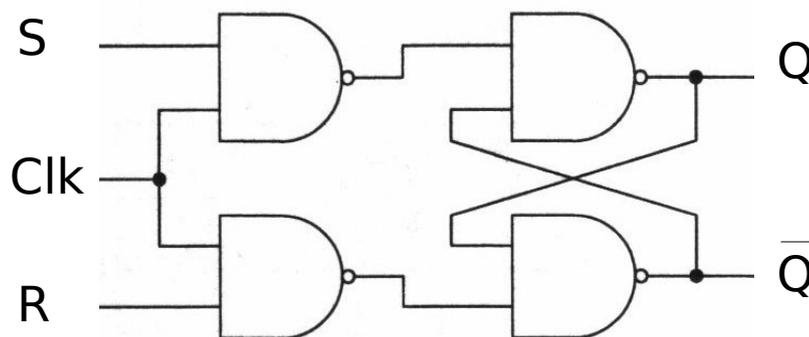
Reset : remise à 0

à proscrire

Les bascules RS

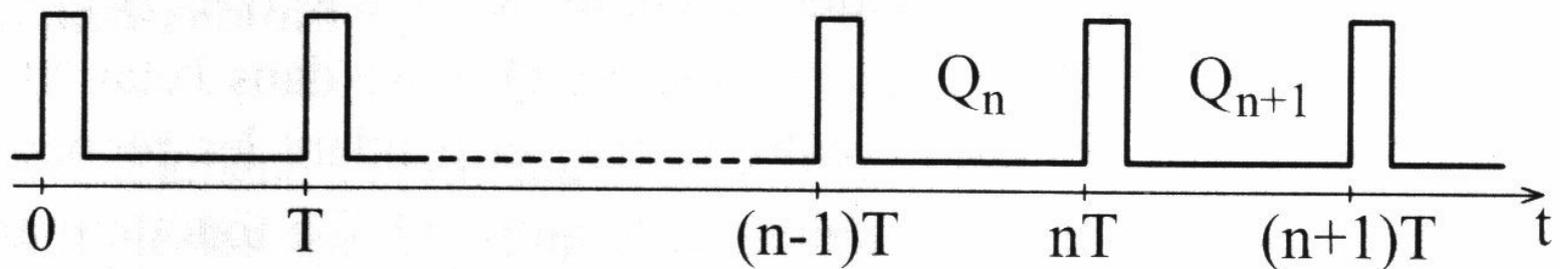
- Bascule RST :

- bascule RS dans laquelle les entrées R et S ne sont prises en comptes que si elles sont en coïncidence avec un signal de commande
 - ✓ bascule bloquée quand le signal de commande est à 0
- si le signal de commande est fourni par une horloge : bascule synchrone

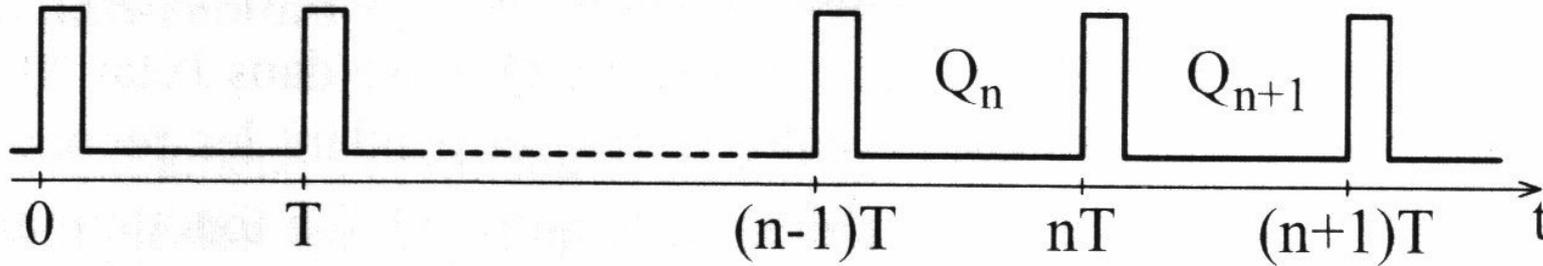


Les bascules RS

- signal de commande donné par une horloge de période T fournissant des impulsions
 - ✓ La largeur t_p des impulsions est supposée petite devant la période
 - ✓ Q_n : valeur de la sortie Q pendant le $n^{\text{ième}}$ intervalle précédant la $n^{\text{ième}}$ impulsion
 - ✓ Q_{n+1} : valeur de la sortie dans l'intervalle suivant la $n^{\text{ième}}$ impulsion



Les bascules RS



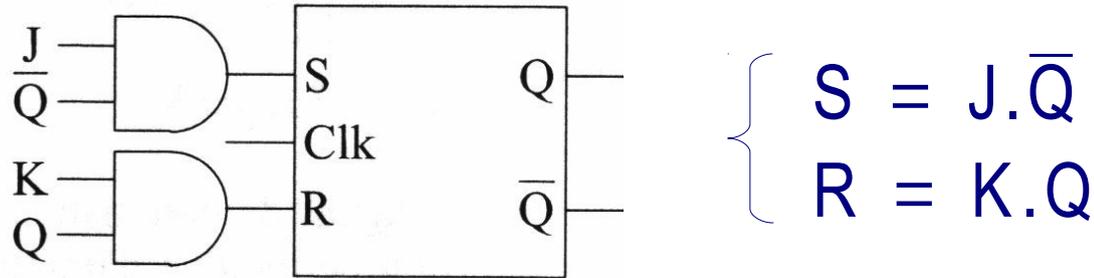
- à $t = nT + \varepsilon$, quand $\text{Clk} = 1$
 - ✓ si $R = S = 0$, la sortie de change pas
 - ✓ si $S = 1$ et $R = 0$, alors Q est forcée à 1
 - ✓ si $S = 0$ et $R = 1$, alors Q est forcée à 0
 - ✓ si $S = 1$ et $R = 1$, alors situation indéterminée

S_n	R_n	Q_{n+1}
0	0	Q_n
1	0	1
0	1	0
1	1	?

Les bascules JK

- bascule J-K

- permet de lever l'ambiguïté des bascules RST



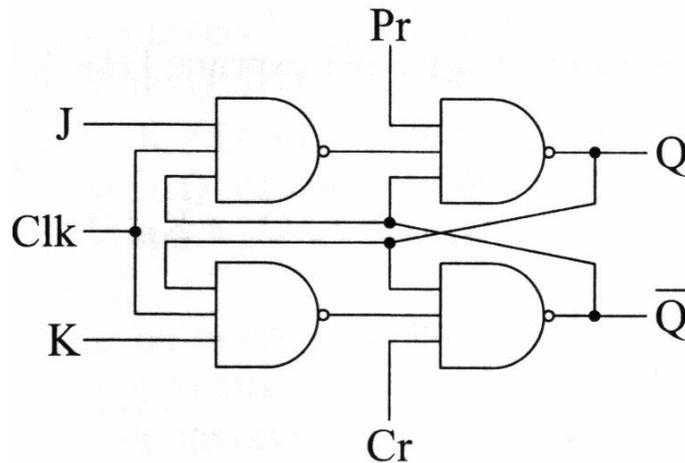
J_n	K_n	Q_n	\bar{Q}_n	S	R	Q_{n+1}
0	0	0	1	0	0	0
0	0	1	0	0	0	1
0	1	0	1	0	0	0
0	1	1	0	0	1	0
1	0	0	1	1	0	1
1	0	1	0	0	0	1
1	1	0	1	1	0	1
1	1	1	0	0	1	0

Les bascules JK

➤ table de vérité

J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$

➤ diagramme logique



Les bascules JK

- table de transition

Q_n	Q_{n+1}	J_n	K_n
0	0		
0	1		
1	0		
1	1		

Les bascules JK

➤ table de transition

Q_n	Q_{n+1}	J_n	K_n
0	0		
0	1		
1	0		
1	1		

- ✓ pour obtenir la transition $0 \rightarrow 0$
 - × $J = K = 0$ maintient l'état de la bascule à 0
 - × $J = 0, K = 1$ charge 0 dans la bascule (qui était déjà à 0)
 - ✓ on aura la transition $0 \rightarrow 0$ si $J=0$, quel que soit K

Les bascules JK

➤ table de transition

Q_n	Q_{n+1}	J_n	K_n
0	0	0	X
0	1		
1	0		
1	1		

- ✓ pour obtenir la transition $0 \rightarrow 0$
 - × $J = K = 0$ maintient l'état de la bascule à 0
 - × $J = 0, K = 1$ charge 0 dans la bascule (qui était déjà à 0)
 - ✓ on aura la transition $0 \rightarrow 0$ si $J=0$, quel que soit K

Les bascules JK

- table de transition

Q_n	Q_{n+1}	J_n	K_n
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Les bascules JK

- expression de Q_{n+1} en fonction de Q_n , J et K
 - ✓ à partir de la table de vérité de la bascule J-K

J_n	K_n	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

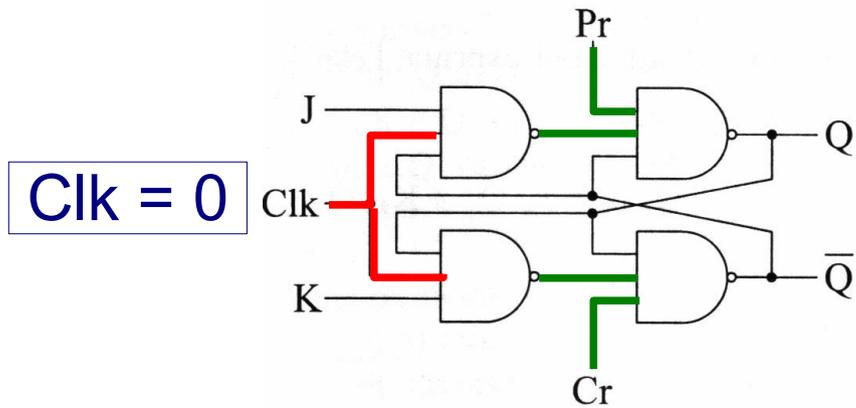
- ✓ on construit le tableau de Karnaugh

JK	00	01	11	10
Q				
0			1	1
1	1			1

$$Q_{n+1} = J_n \bar{Q}_n + \bar{K}_n Q_n$$

Les bascules JK

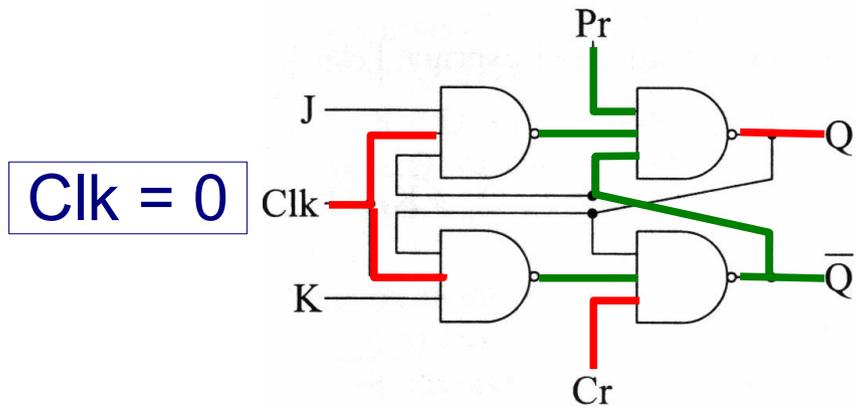
- rôle des entrées Pr et Cr



Pr	Cr	Q	\bar{Q}
1	1	Q	\bar{Q}

Les bascules JK

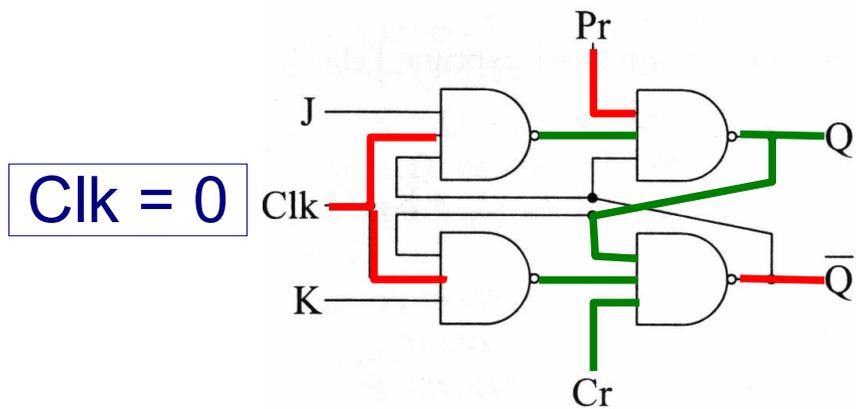
- rôle des entrées Pr et Cr



Pr	Cr	Q	\bar{Q}
1	1	Q	\bar{Q}
1	0	0	1

Les bascules JK

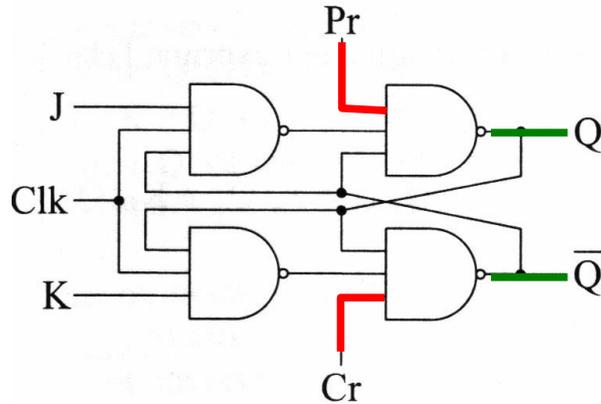
- rôle des entrées Pr et Cr



Pr	Cr	Q	\bar{Q}
1	1	Q	\bar{Q}
1	0	0	1
0	1	1	0

Les bascules JK

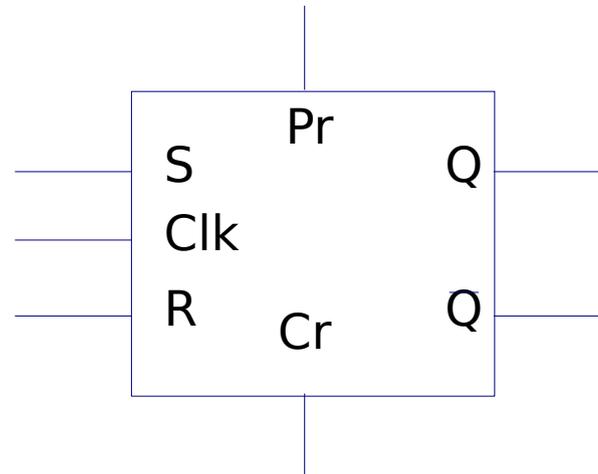
- rôle des entrées Pr et Cr



Pr	Cr	Q	\bar{Q}
1	1	Q	\bar{Q}
1	0	0	1
0	1	1	0
0	0	1	1

à proscrire

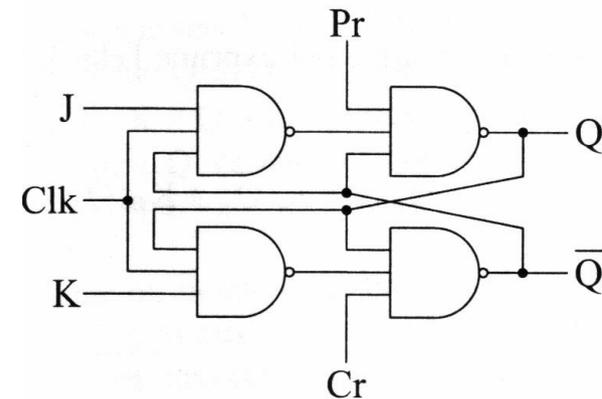
- ✓ entrées asynchrones (lorsque Clk = 0), pour assigner l'état initial de la bascule
- ✓ en fonctionnement normal, elles doivent être maintenues à 1



Les bascules JK

- bascules J-K Maître-Esclave

- on a construit les tables de vérité à partir de la logique combinatoire
- mais il y a maintenant des conditions d'asservissement entre les entrées et les sorties
 - ✓ supposons $J=K=1$ et $Q=0$



Les bascules JK

- bascules J-K Maître-Esclave

- on a construit les tables de vérité à partir de la logique combinatoire
- mais il y a maintenant des conditions d'asservissement entre les entrées et les sorties

- ✓ supposons $J=K=1$ et $Q=0$
- ✓ quand le signal d'horloge passe à 1, Q va passer à 1
- ✓ le changement se produit après un intervalle de temps Δt

J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$

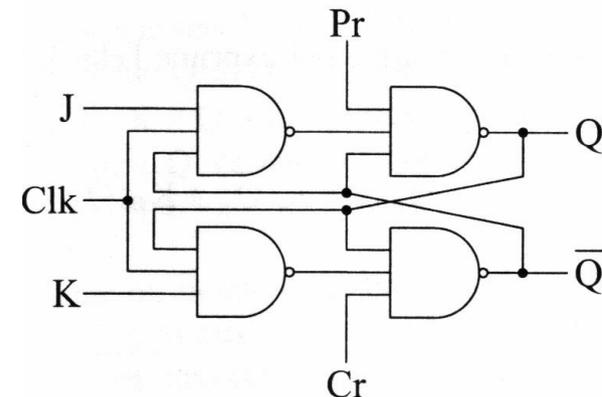
- ✓ on a alors $J=K=Q=1$ et le signal d'horloge est toujours 1
 Q doit revenir à 0 !

Les bascules JK

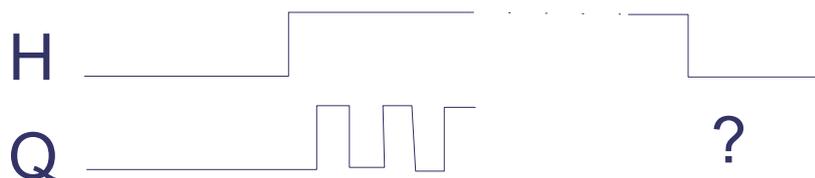
- bascules J-K Maître-Esclave

- on a construit les tables de vérité à partir de la logique combinatoire
- mais il y a maintenant des conditions d'asservissement entre les entrées et les sorties

- ✓ supposons $J=K=1$ et $Q=0$
- ✓ quand le signal d'horloge passe à 1, Q va passer à 1
- ✓ le changement se produit après un intervalle de temps Δt

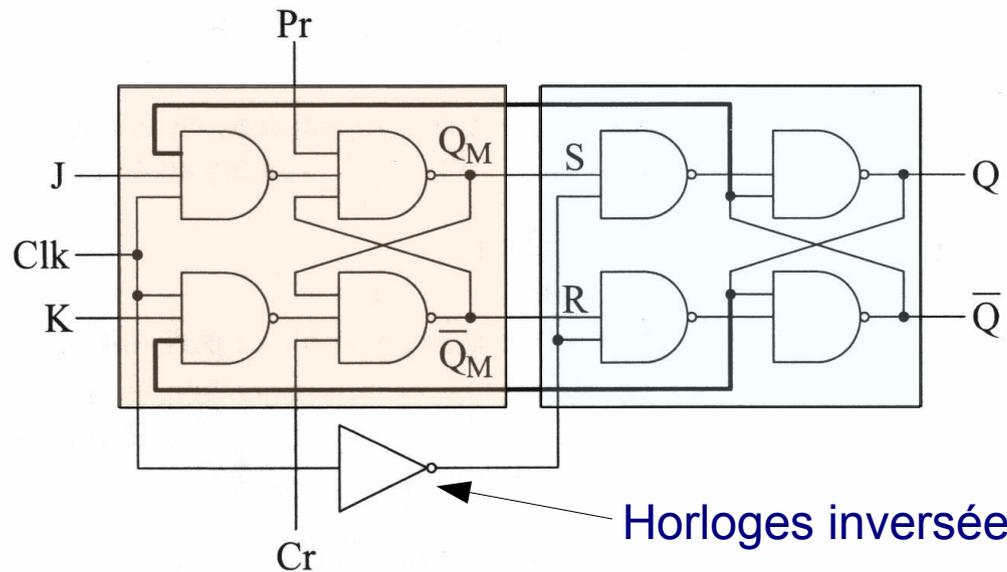


- ✓ on a alors $J=K=Q=1$ et le signal d'horloge est toujours 1
 Q doit revenir à 0 !
- ✓ d'où des oscillations pendant la durée du signal d'horloge



Les bascules JK

- solution au problème : au lieu d'un simple asservissement (bascule J-K) on monte en cascade deux bascules R-S en asservissant les entrées de la première (**Maître**) aux sorties de la seconde (**Esclave**)

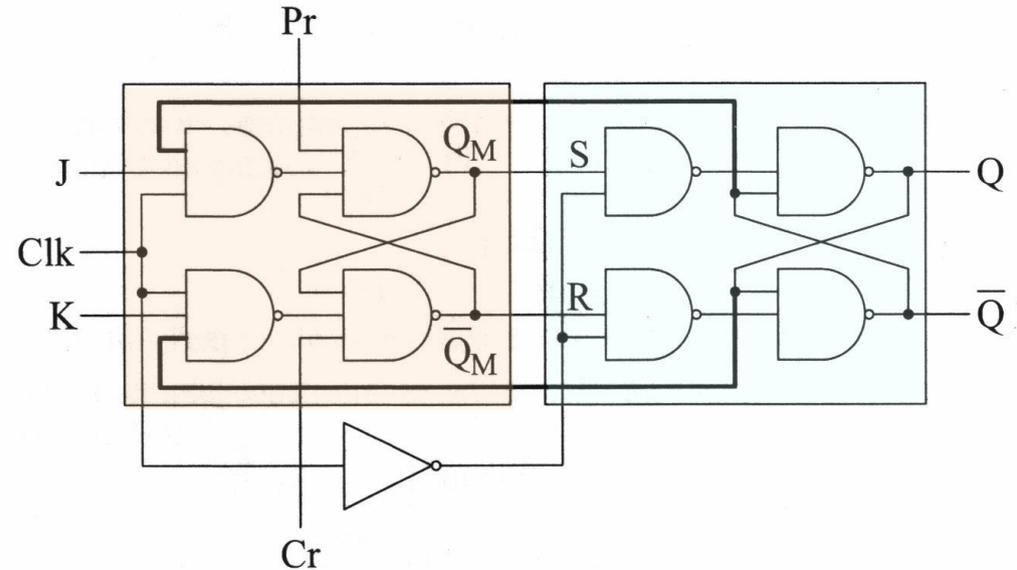


- ✓ pendant la $n^{\text{ième}}$ impulsion, le signal d'horloge est haut pour le maître, bas pour l'esclave. Par conséquent Q_n est invariant pendant la durée de t_p et le problème précédent est résolu

Les bascules JK

- ✓ l'état de la sortie de la bascule maître Q_M est donné par

J_n	K_n	Q_M
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$



- ✓ quand le signal d'horloge revient à 0 pour le maître qui devient bloqué, l'esclave est libéré et on a alors

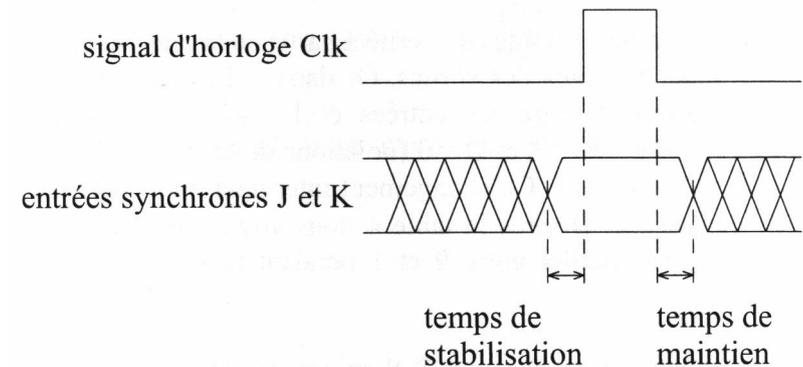
$$(Q_M = 1, \overline{Q_M} = 0) \Rightarrow (S = 1, R = 0) \Rightarrow (Q_{n+1} = 1, \overline{Q_{n+1}} = 0)$$

$$(Q_M = 0, \overline{Q_M} = 1) \Rightarrow (S = 0, R = 1) \Rightarrow (Q_{n+1} = 0, \overline{Q_{n+1}} = 1)$$

- ✓ on a transféré l'état de la bascule maître à la bascule esclave

Les bascules JK

- ✓ l'état de la machine est transféré à la machine esclave lors de la transition $1 \rightarrow 0$ (front descendant du signal d'horloge)
- ✓ les entrées J et K qui définissent le nouvel état des sorties sont prise en compte pendant que le signal d'horloge est à 1
- ✓ ces signaux J et K doivent donc être stables pendant que le signal Clk est haut
 - ✗ ils doivent être stabilisés un peu avant le front montant
 - ✗ et le rester un peu après le front descendant

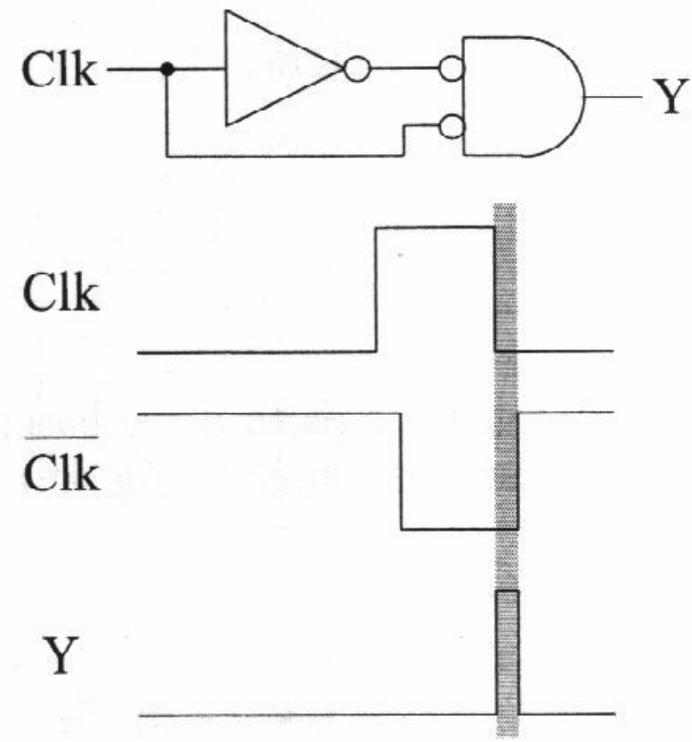
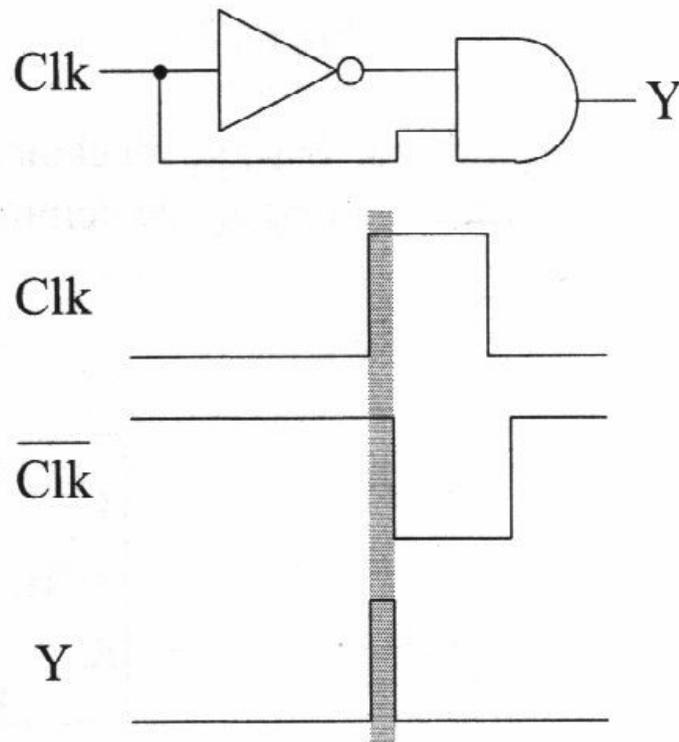


Les bascules JK

- déclenchement sur le front montant ou descendant d'une horloge
 - autre manière de résoudre le problème d'oscillation de la bascule J-K
 - circuit détectant le front d'une transition
 - ✓ produit en coïncidence avec le front montant ou descendant de Clk une impulsion de largeur juste suffisante pour permettre un basculement d'état
 - ✓ utilise le retard induit par le franchissement d'un circuit

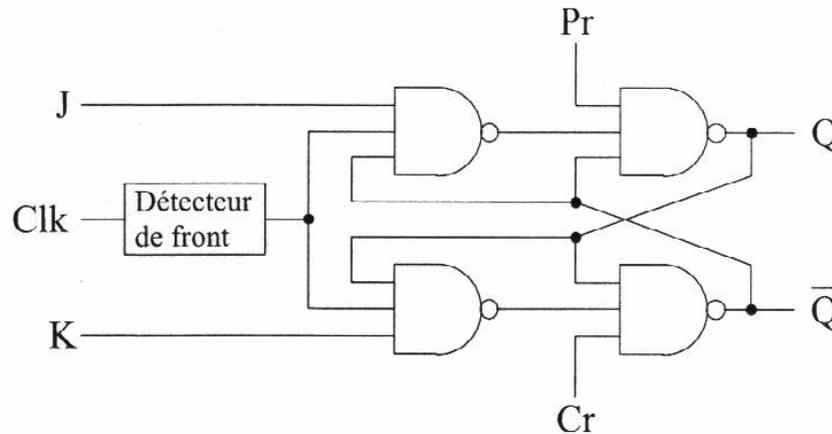
Les bascules JK

- ✓ impulsion fournie par les montages

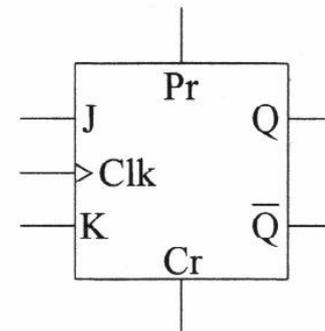


Les bascules JK

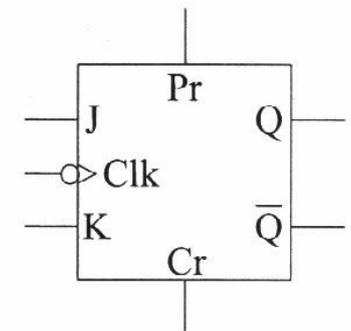
- en introduisant un détecteur de front entre l'horloge et la bascule J-K, on choisit le front sur lequel on déclenche



- les bascules déclenchées par un front montant ou descendant sont représentées par



bascule déclenchée par front montant

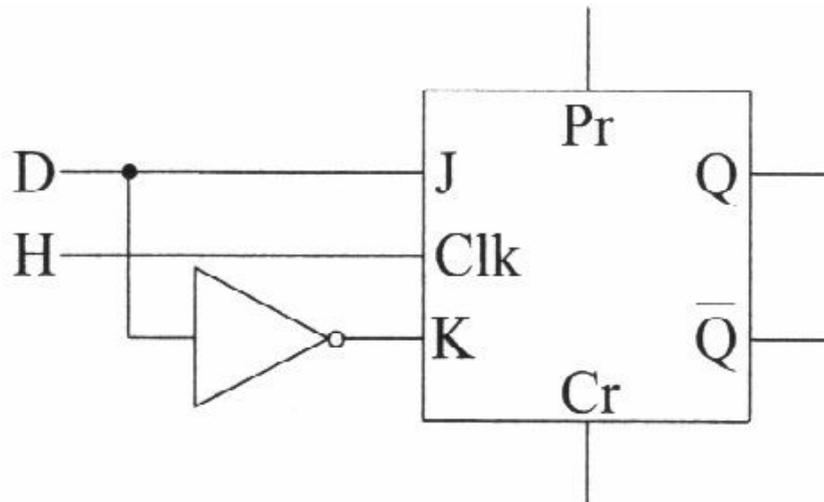


bascule déclenchée par front descendant

Les bascules D

- bascule D

- permettent de générer un "retard" (delay) ou de stocker de l'information (latch)
- en envoyant une donnée D sur l'entrée J et son inverse sur l'entrée K



Les bascules D

- à partir de la table de vérité de la bascule J-K

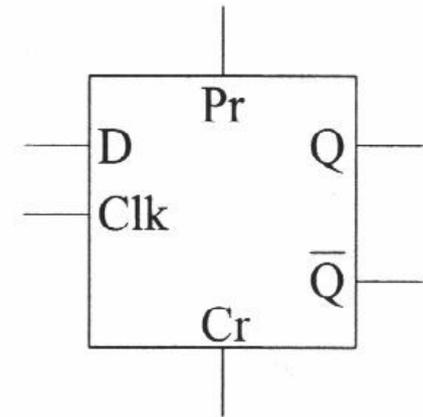
J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$

- on a

$$D_n = 1 \Rightarrow (J_n = 1, K_n = 0) \Rightarrow Q_{n+1} = 1$$

$$D_n = 0 \Rightarrow (J_n = 0, K_n = 1) \Rightarrow Q_{n+1} = 0$$

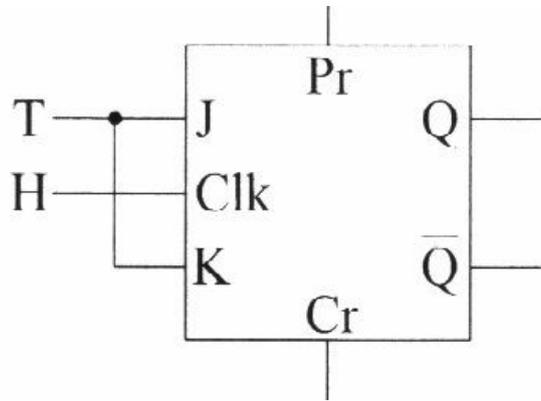
- soit : $Q_{n+1} = D_n$



Les bascules T

- bascule T

- obtenue en appliquant les mêmes valeurs aux deux entrées d'une bascule JK



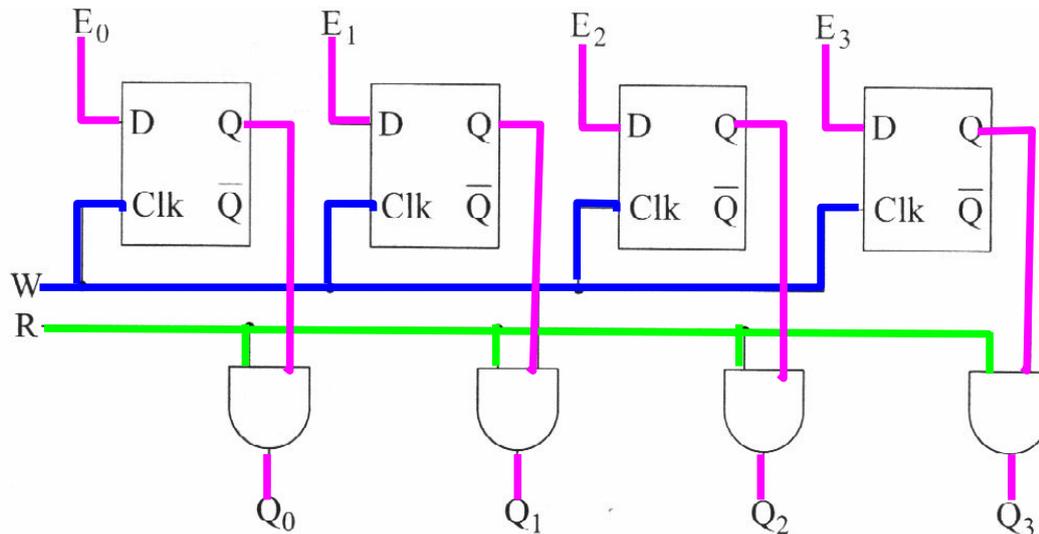
J_n	K_n	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$

- si $J = K = 0$ alors $Q_{n+1} = Q_n$
- si $J = K = 1$ alors $Q_{n+1} = \overline{Q_n}$

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

Les registres

- registre de mémorisation
 - association de n bascules D pour mémoriser n bits

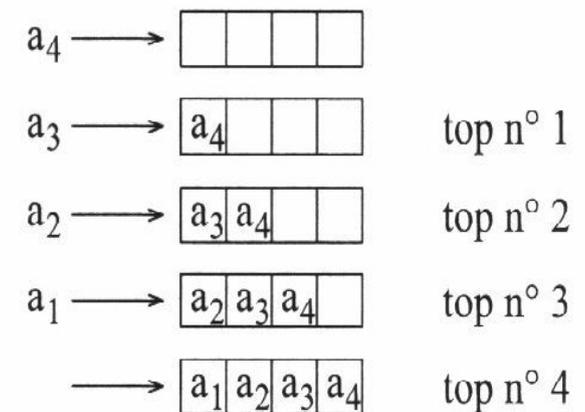


- les entrées présentes sur E_0, E_1, E_2, E_3 sont mémorisées en synchronisation avec le signal W
- elles peuvent être lues sur les sorties Q_0, Q_1, Q_2, Q_3 en coïncidence avec le signal de validation R
 - ✓ si les sorties se font sur un bus, portes à 3 états au lieu des ET

Les registres

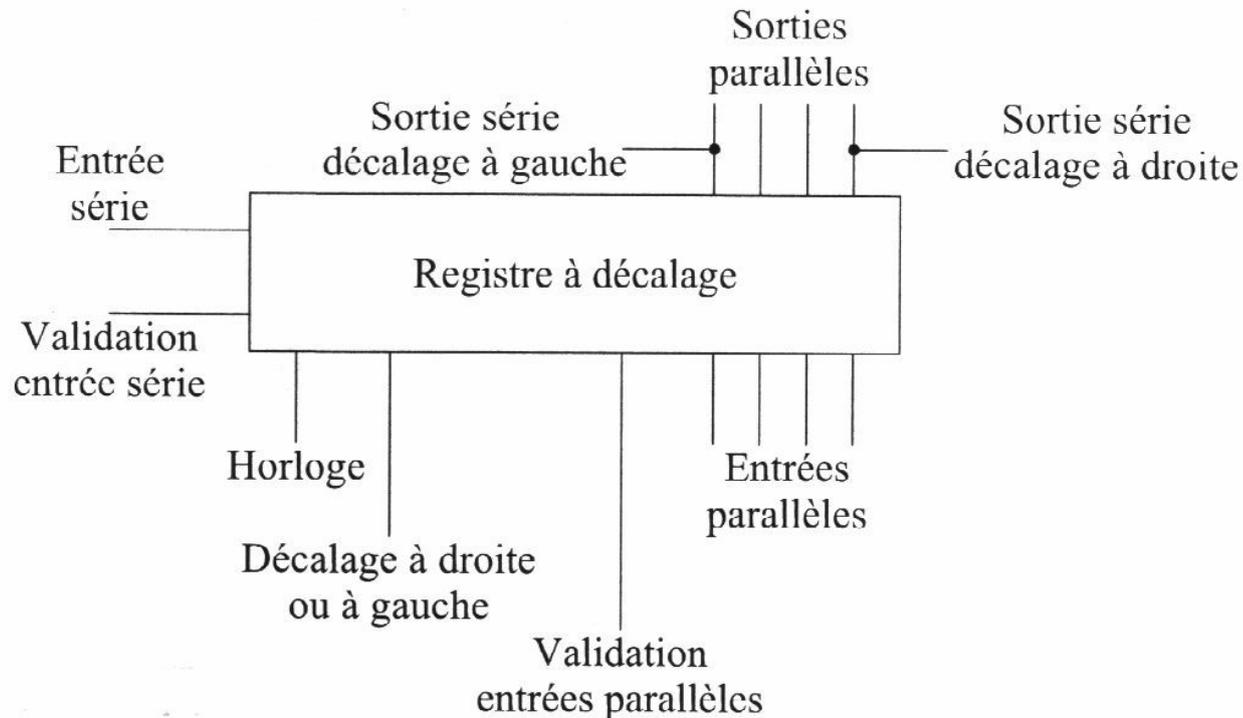
- registre à décalage

- bascules D interconnectées de façon à ce que l'état de la bascule de rang i soit transmis à la bascule de rang $i+1$ (ou $i-1$) quand un signal d'horloge est appliqué à l'ensemble des bascules
- 2 types d'entrées
 - ✓ parallèle, comme dans un registre de mémorisation
 - ✓ série
 - × l'information est présentée séquentiellement bit après bit à la 1ère bascule
 - × à chaque coup d'horloge, un nouveau bit est présenté et ceux déjà chargés sont décalés d'un rang



Les registres

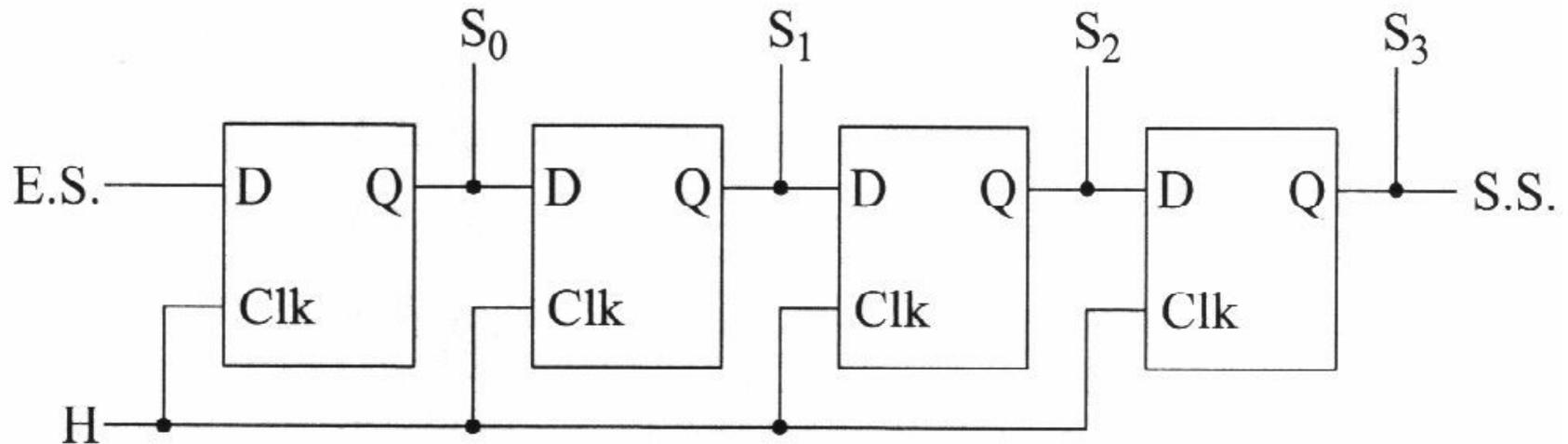
- l'information stockée dans un registre à décalage peut être lue de la même manière, en série ou en parallèle
- certains registres sont capables de décaler à droite ou à gauche (registres à décalage universels)



- généralement réalisés avec des bascules du type maître esclave D ou R-S

Les registres

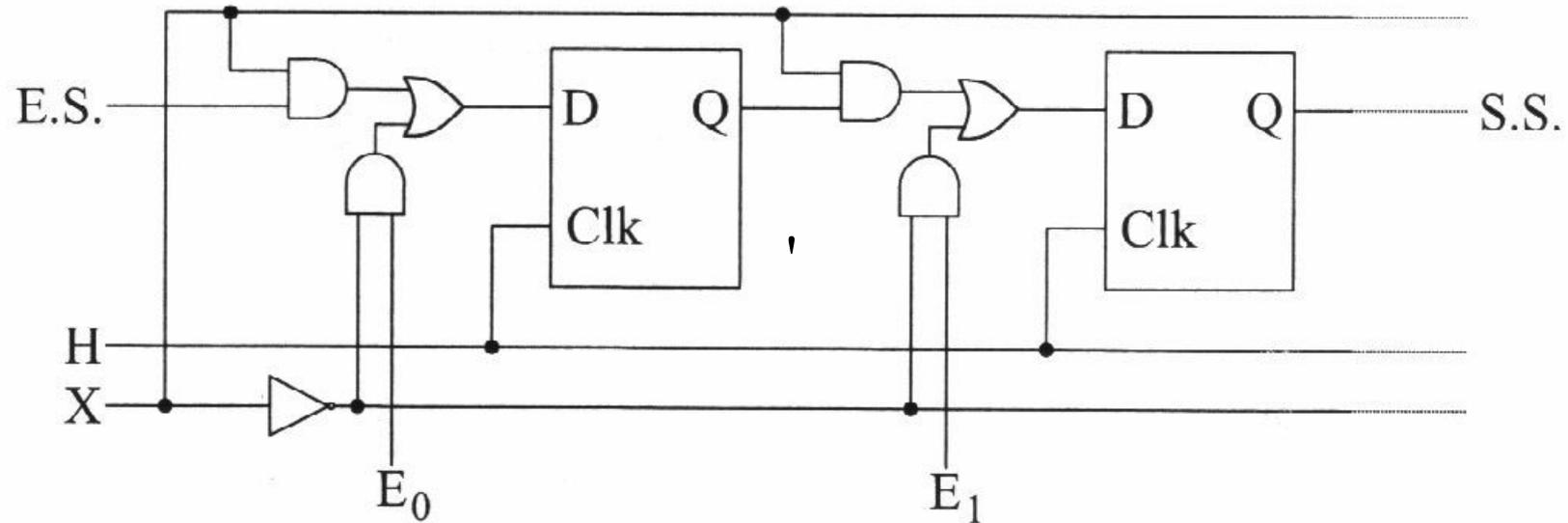
- entrée série - sortie parallèle



- pour transformer un codage temporel (succession des bits dans le temps en codage spatial (information stockée dans une mémoire statique
- on peut utiliser aussi la sortie série, avec éventuellement des fréquences d'horloge différentes en entrée et en sortie
 - ✓ le registre sert alors de mémoire tampon (buffer)

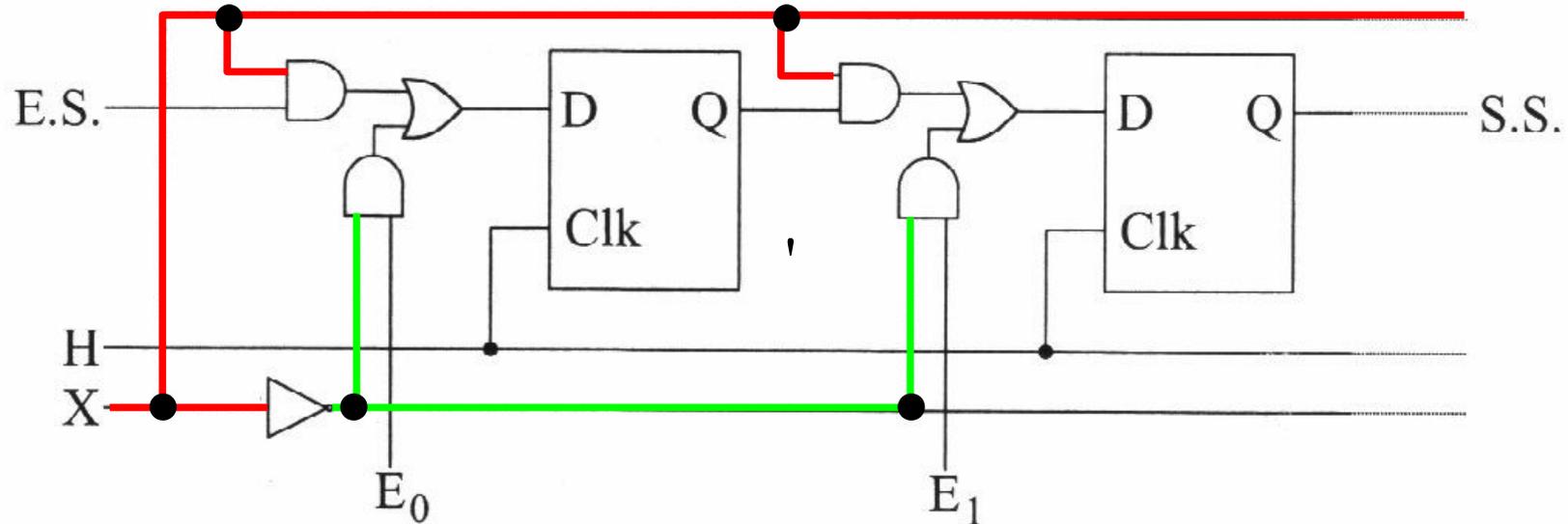
Les registres

- entrée parallèle - sortie série



Les registres

- entrée parallèle - sortie série

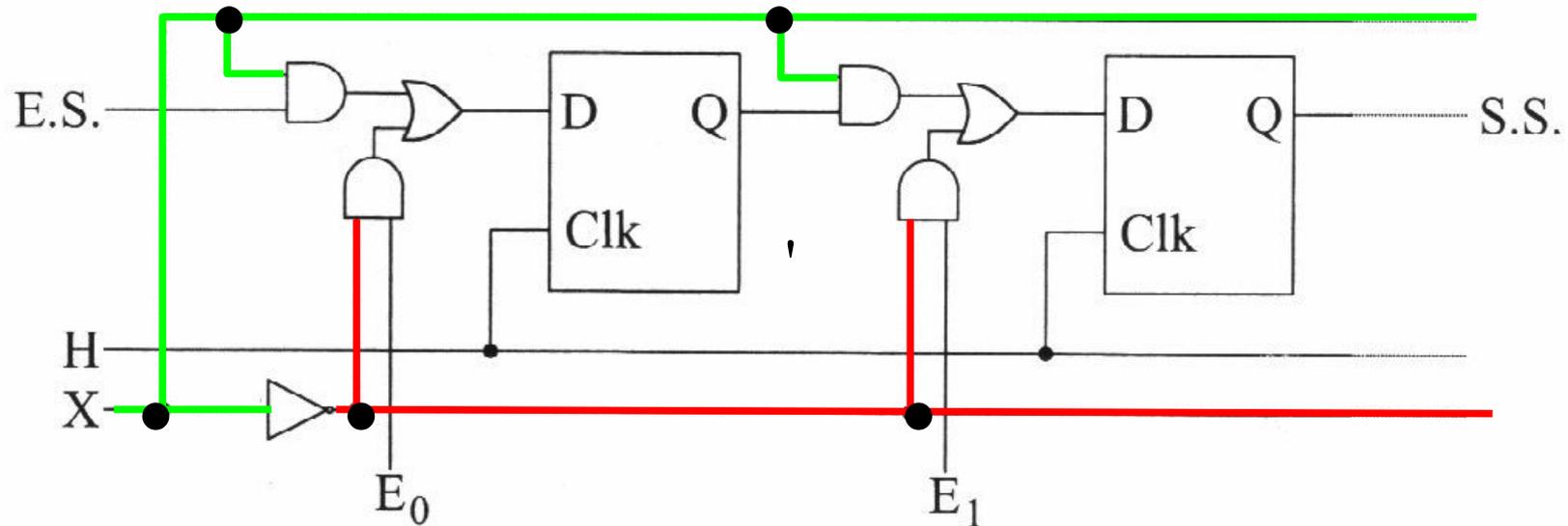


exemple d'une entrée parallèle+série et sortie série

- si $X=0$, l'entrée série est inhibée et l'entrée parallèle est validée

Les registres

- entrée parallèle - sortie série



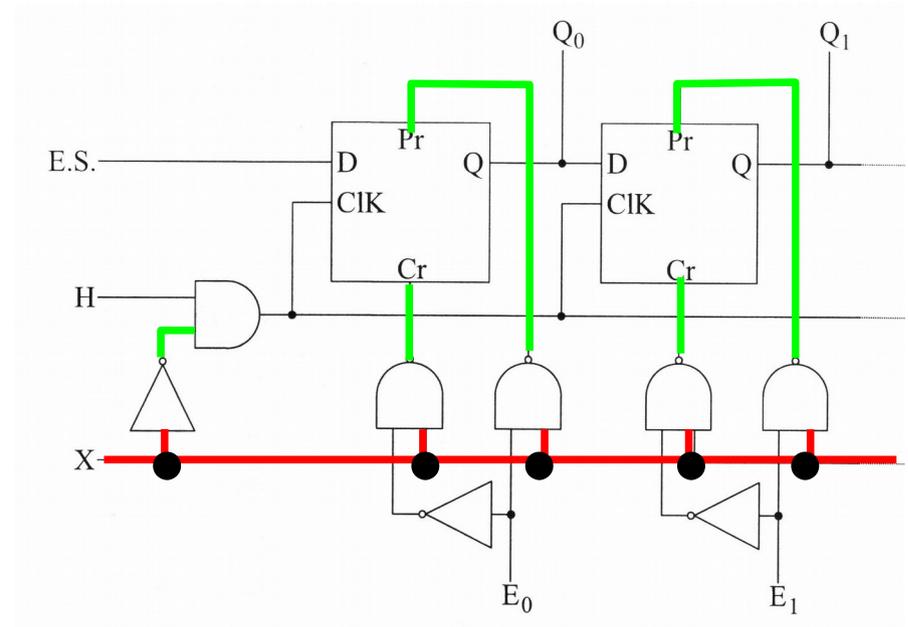
exemple d'une entrée parallèle+série et sortie série

- si $X=1$, l'entrée parallèle est inhibée et l'entrée série est validée
- permet de transformer un codage spatial en codage temporel

Les registres

- entrée parallèle - sortie parallèle

- la commande X permet de sélectionner le mode de chargement

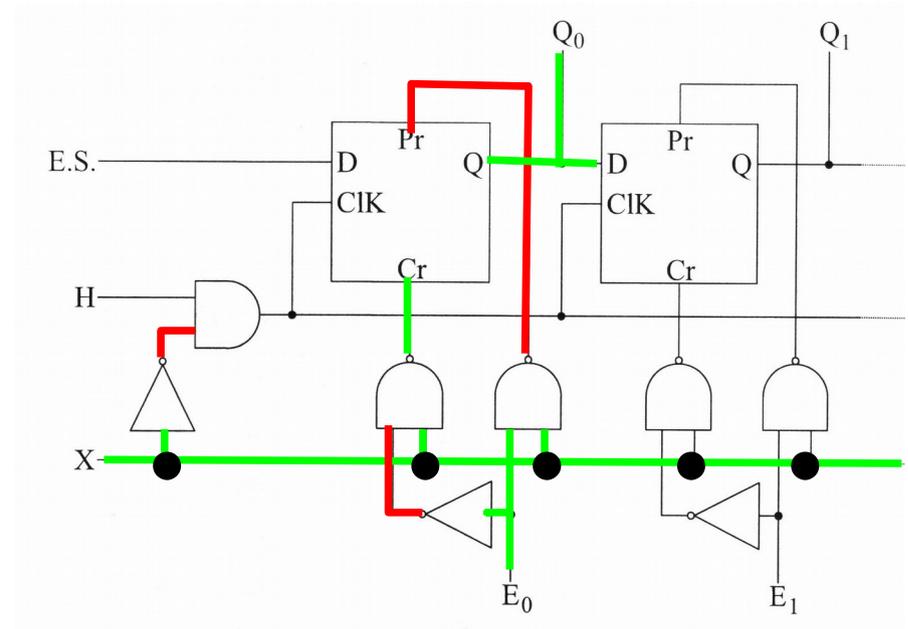


- ✓ $X=0 \Rightarrow Pr=Cr=1 \Rightarrow$ fonctionnement normal des bascules
 \Rightarrow inhibition des entrées parallèles

Les registres

- entrée parallèle - sortie parallèle

- la commande X permet de sélectionner le mode de chargement

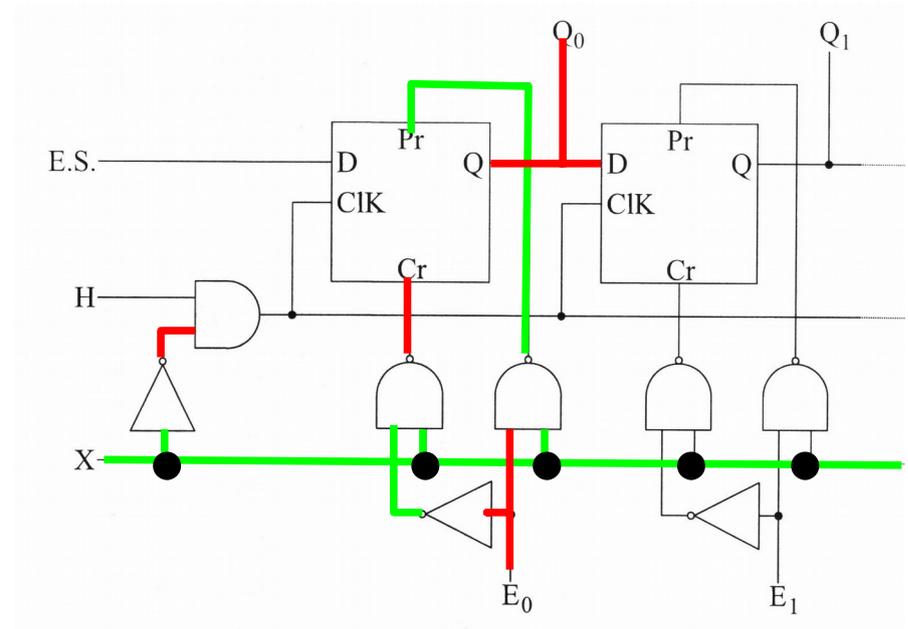


- ✓ $X=0 \Rightarrow Pr=Cr=1 \Rightarrow$ fonctionnement normal des bascules
 \Rightarrow inhibition des entrées parallèles
- ✓ $X=1$
 - × $E_i=1 \Rightarrow (Pr=0, Cr=1) \Rightarrow Q_i=1$

Les registres

- entrée parallèle - sortie parallèle

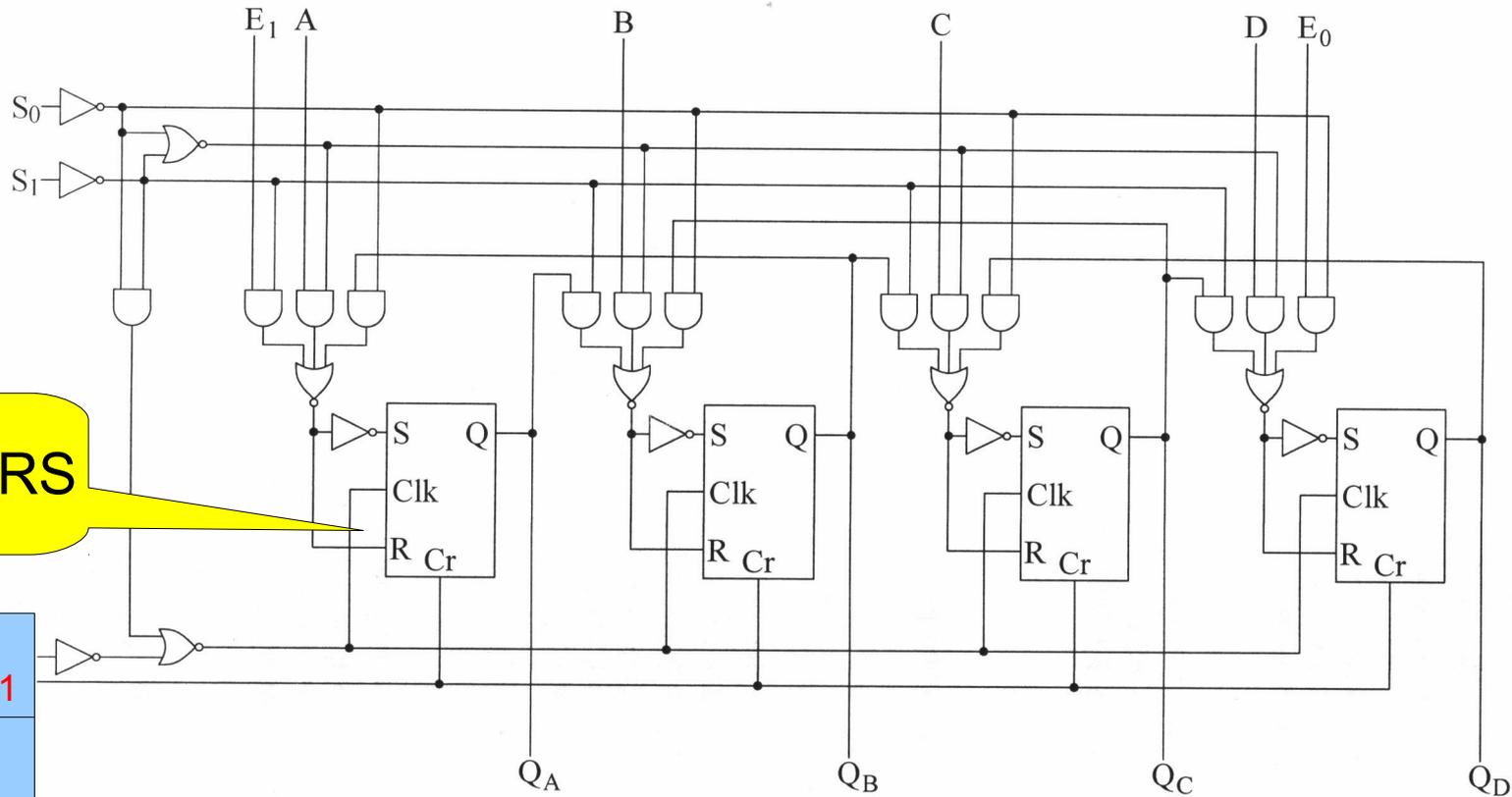
- la commande X permet de sélectionner le mode de chargement



- ✓ $X=0 \Rightarrow Pr=Cr=1 \Rightarrow$ fonctionnement normal des bascules
 \Rightarrow inhibition des entrées parallèles
- ✓ $X=1$
 - x $E_i=1 \Rightarrow (Pr=0, Cr=1) \Rightarrow Q_i=1$
 - x $E_i=0 \Rightarrow (Pr=1, Cr=0) \Rightarrow Q_i=0$ $\left. \vphantom{\begin{matrix} x E_i=1 \\ x E_i=0 \end{matrix}} \right\} Q_i = E_i$

Les registres

- registre à décalage à droite et à gauche



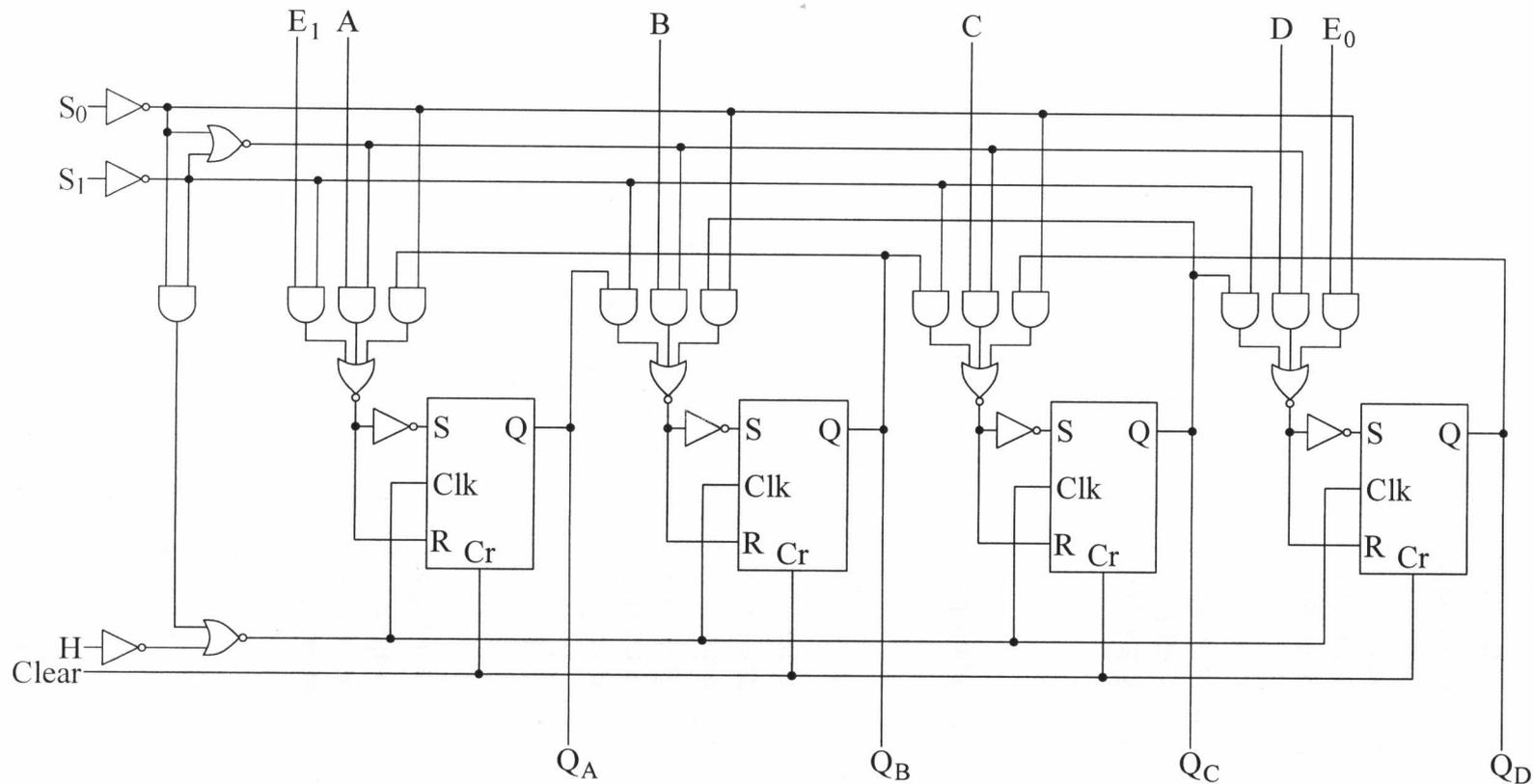
Basculas RS

R_n	S_n	Q_{n+1}
0	0	Q_n
0	1	1
1	0	0
1	1	interdit

- mode de fonctionnement commandé par les entrées S_0 et S_1

Les registres

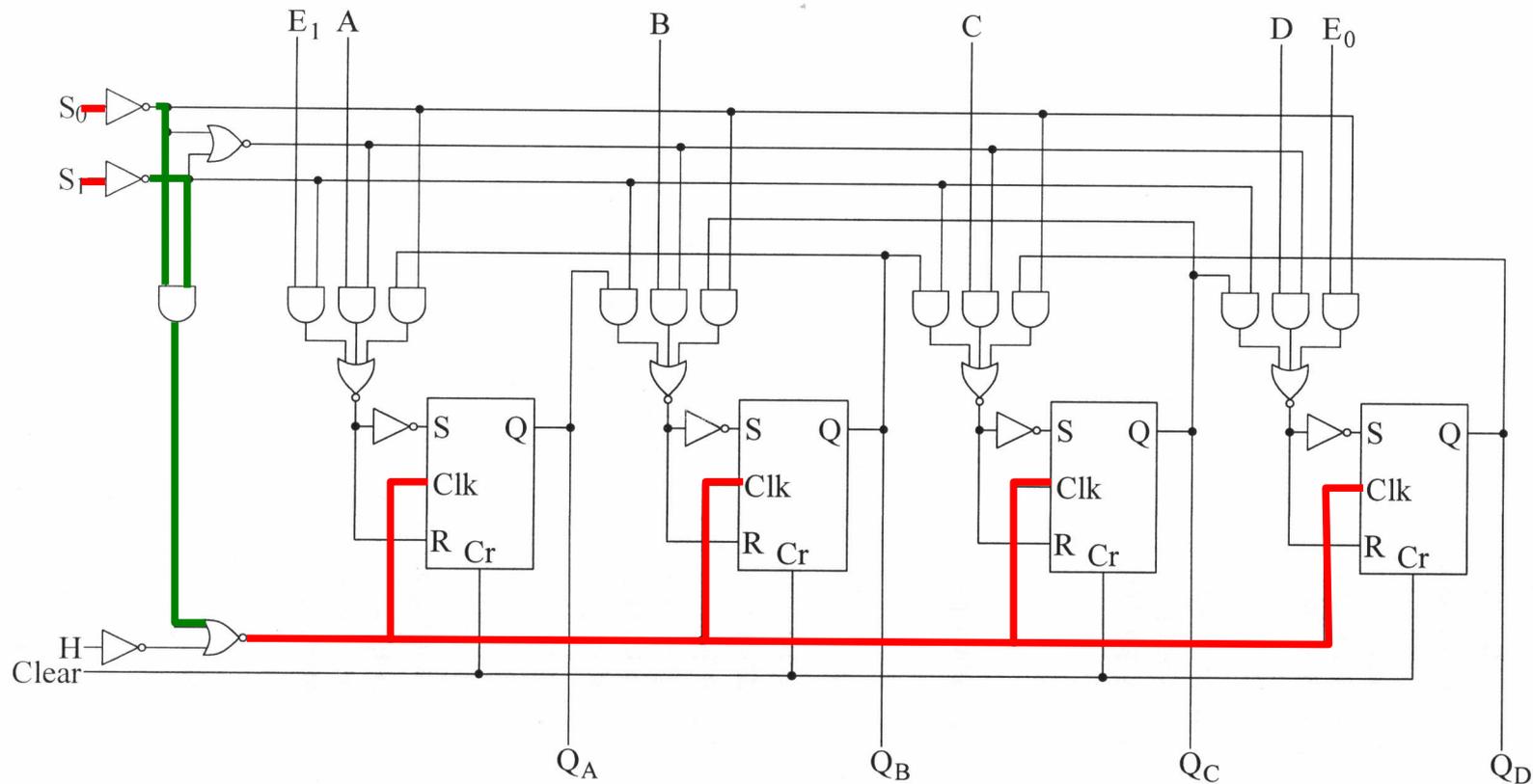
- registre à décalage à droite et à gauche



$$Clk = \overline{H + S_0 \cdot S_1} = H \cdot (S_0 + S_1)$$

Les registres

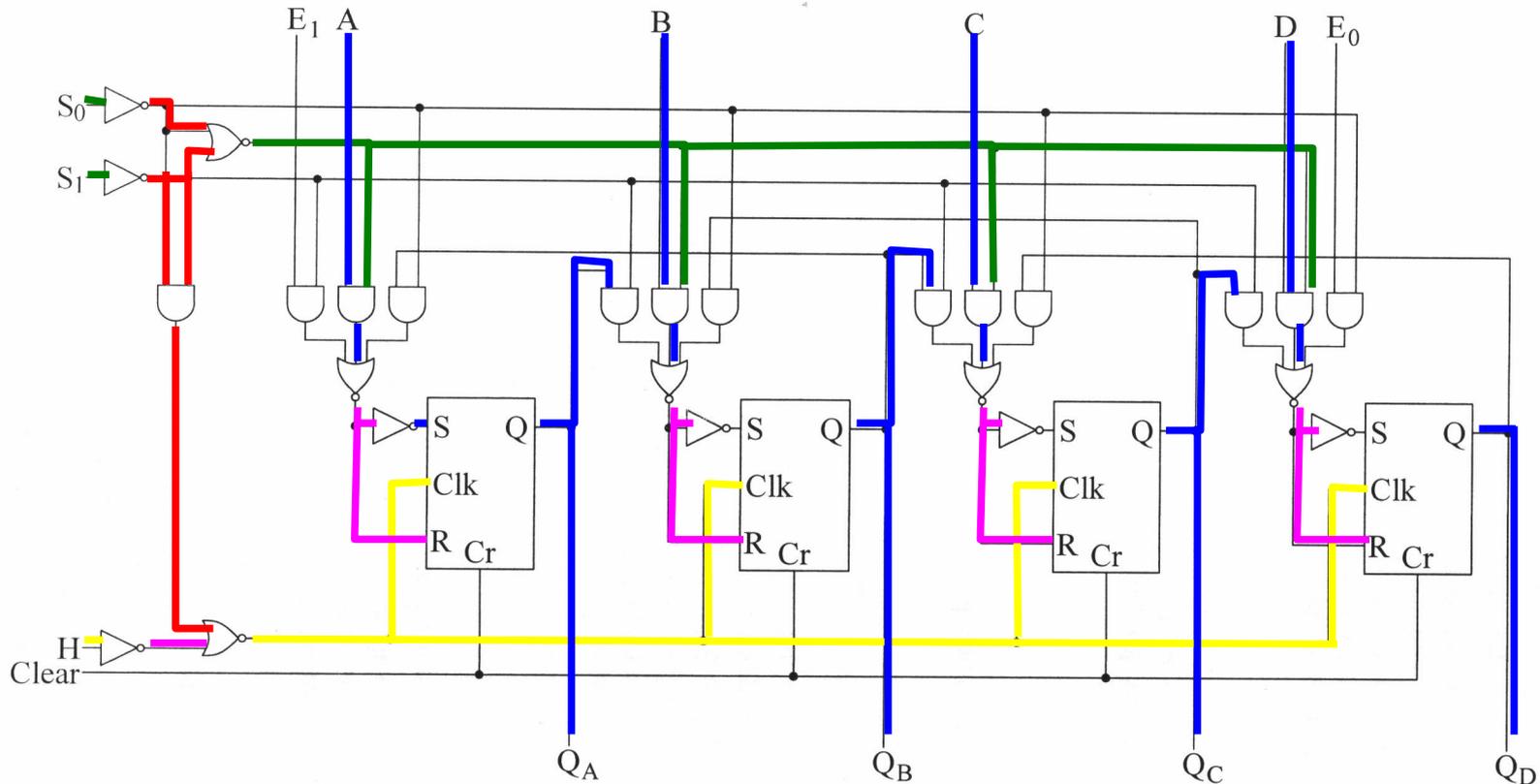
- registre à décalage à droite et à gauche



- $Clk = \overline{H + S_0 \cdot S_1} = H \cdot (S_0 + S_1)$
 - ✓ signal d'horloge inhibé si $S_0 = S_1 = 0$

Les registres

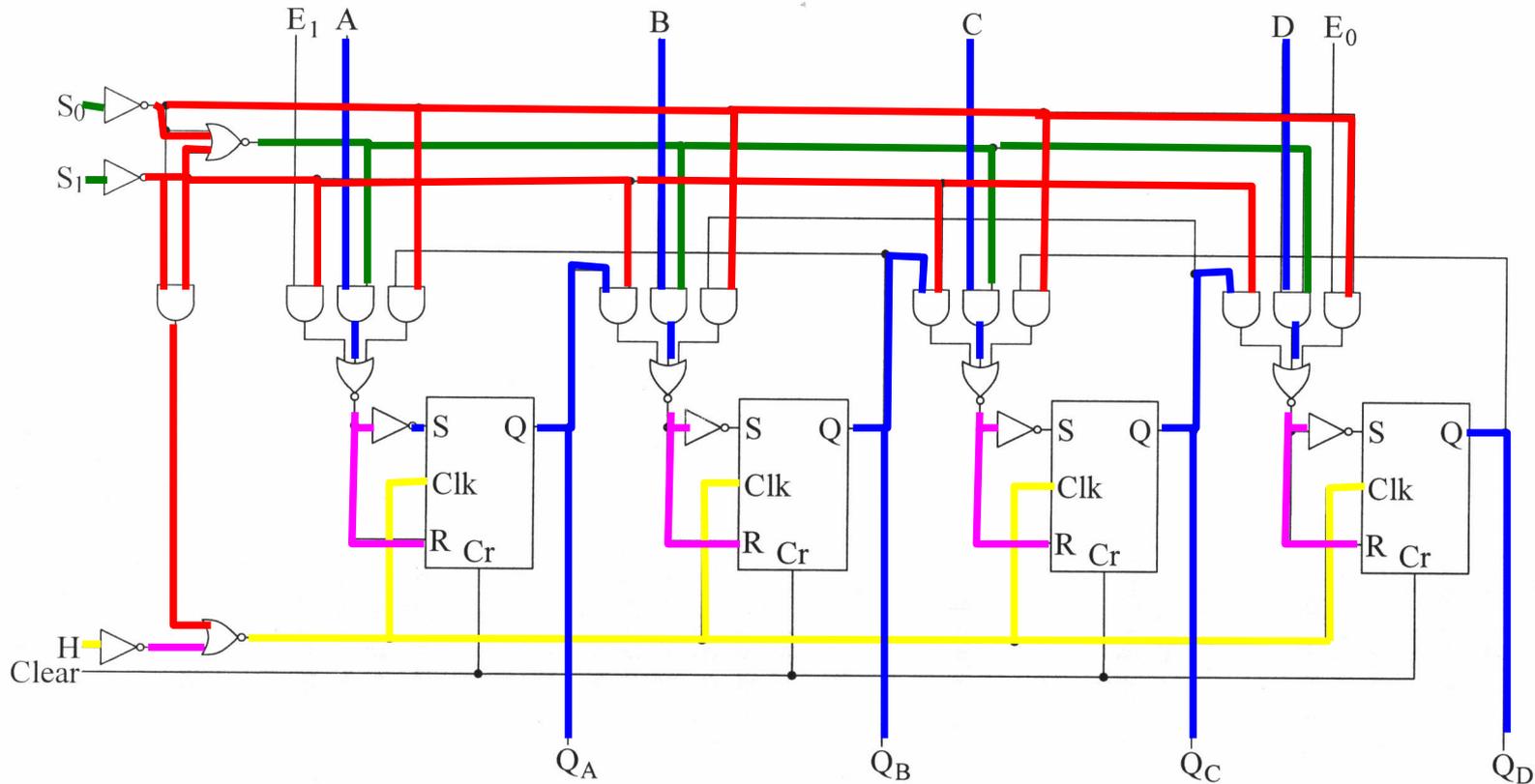
- registre à décalage à droite et à gauche



- sélection du chargement parallèle si
 - ✓ $\overline{S_0} + \overline{S_1} = S_0 \cdot S_1 \Rightarrow S_0 = S_1 = 1$
lignes d'entrées A, B, C, D validées

Les registres

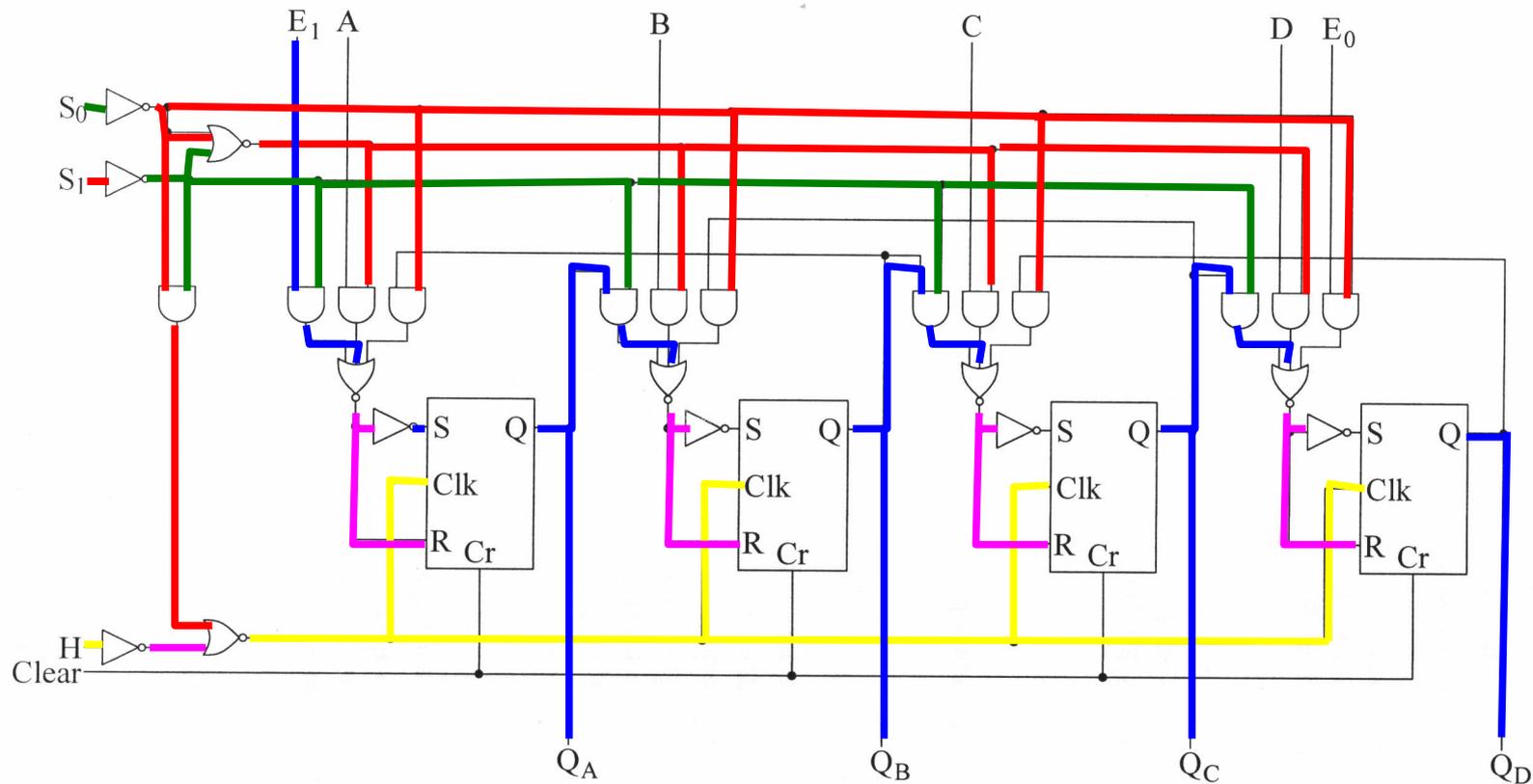
- registre à décalage à droite et à gauche



- sélection du chargement parallèle si
 - ✓ $\overline{S_0} + \overline{S_1} = S_0 \cdot S_1 \Rightarrow S_0 = S_1 = 1$
entrées E_0 et E_1 bloquées, liens Q-RS bloqués

Les registres

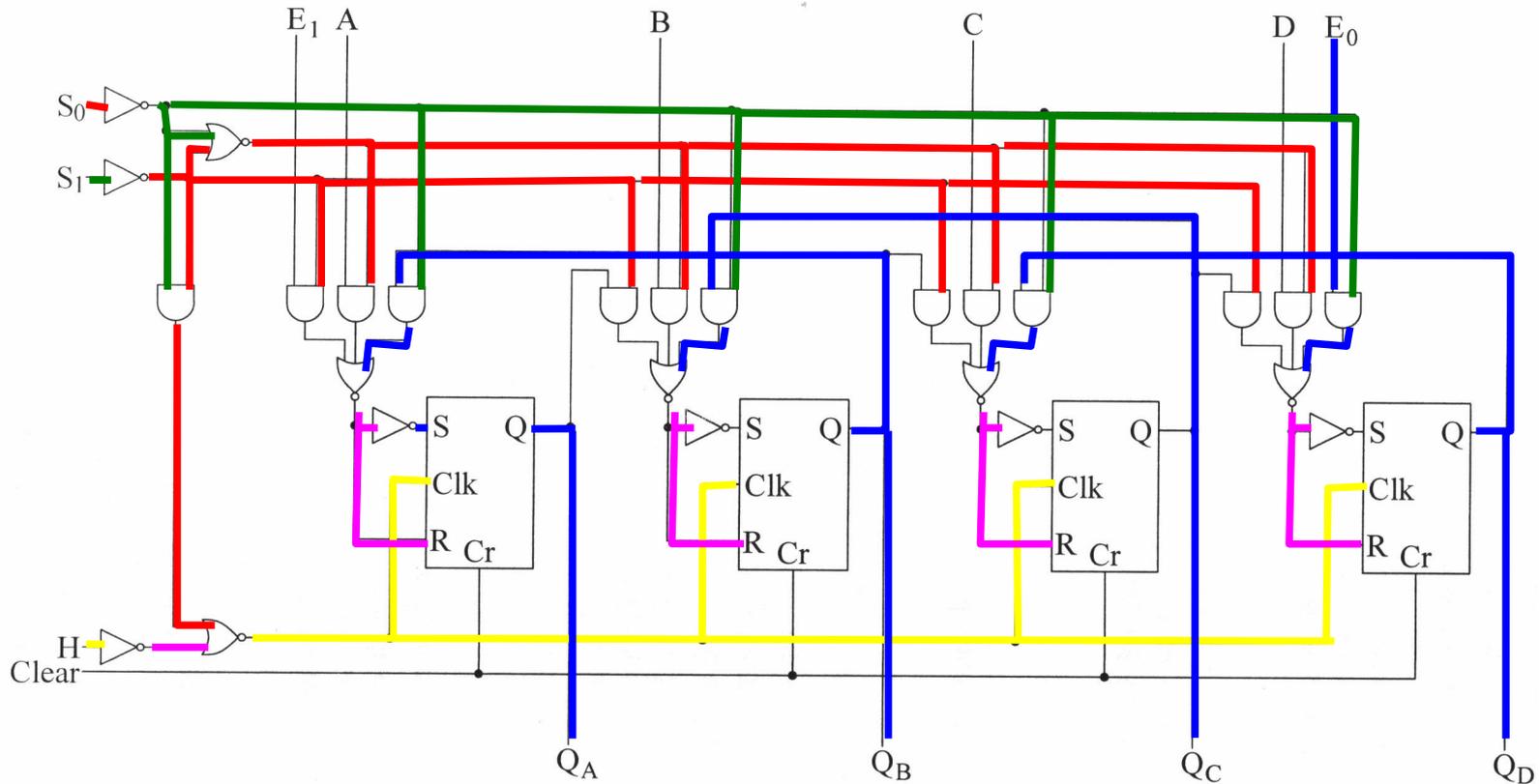
- registre à décalage à droite et à gauche



- sélection du décalage à droite (entrée E_1 , sortie Q_D) si
 - ✓ $S_0 = 1$ et $S_1 = 0$
entrée E_0 bloquée, lien Q – RS vers la droite

Les registres

- registre à décalage à droite et à gauche



- sélection du décalage à gauche si
 - ✓ $S_0 = 0$ et $S_1 = 1$
 - entrée E_1 bloquée, lien Q – RS vers la gauche

Les registres

- en résumé :

S_0	S_1	fonction
0	0	registre bloqué
0	1	décalage à gauche
1	0	décalage à droite
1	1	chargement parallèle

- les registres à décalage permettent d'effectuer des multiplications par 2 (décalage d'un rang vers la gauche) ou des divisions par 2 (décalage d'un rang vers la droite)

Les compteurs

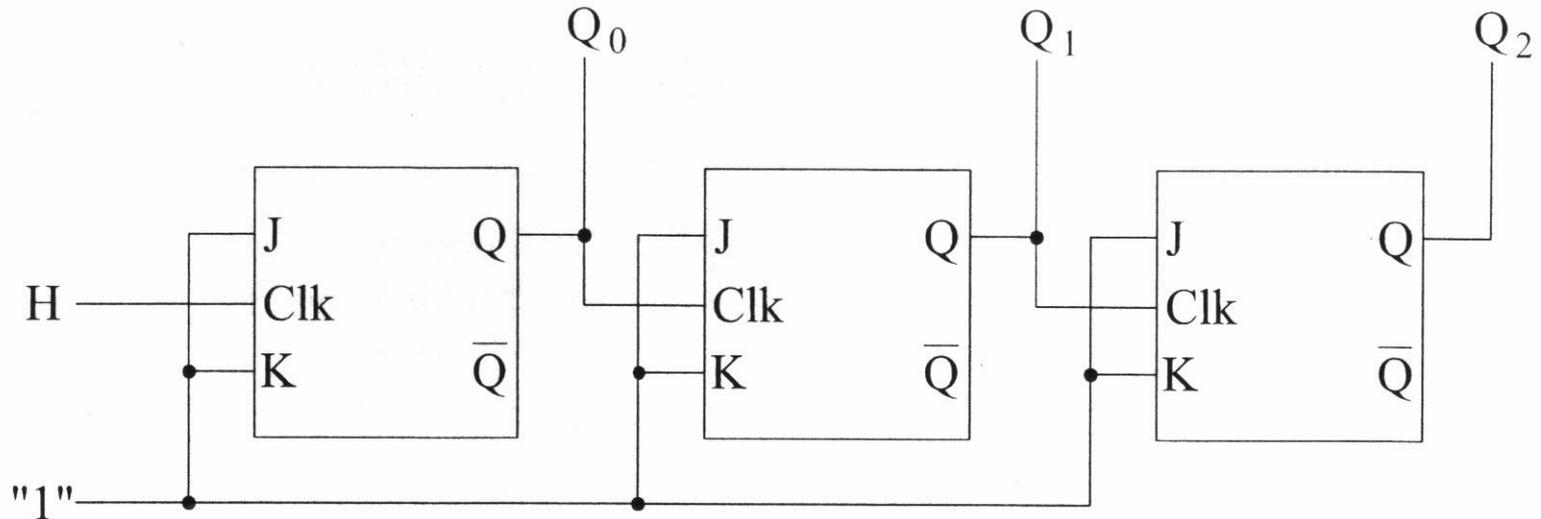
- **compteur** : ensemble de n bascules connectées par des portes logiques
- décrivent une **séquence** déterminée (c'est-à-dire occuper une suite d'états binaires) au rythme d'une horloge
- 2^n combinaisons possibles
- les états sont stables et accessibles entre deux impulsions de l'horloge
- N nombre total de combinaisons successives utilisées $N \leq 2^n$: **modulo** du compteur
- compteur asynchrones ou synchrones
 - réversibles ou compteurs-décompteurs

Les compteurs

- compteurs asynchrones
 - constitué de n bascules J-K fonctionnant en mode T
 - le signal d'horloge n'est reçu que par le 1^{er} étage (LSB)
 - le signal d'horloge des autres bascules est fourni par une sortie de l'étage précédent

Les compteurs

- exemple
 - ✓ compteur modulo 8 avec 3 bascules J-K maître-esclave



- ✓ on suppose que initialement toutes les bascules sont à 0
- ✓ Q₀ va changer d'état à chaque tick d'horloge
- ✓ à chaque fois que Q₀ va passer de 1 à 0, Q₁ va changer
- ✓ à chaque fois que Q₁ va passer de 1 à 0, Q₂ va changer

Les compteurs

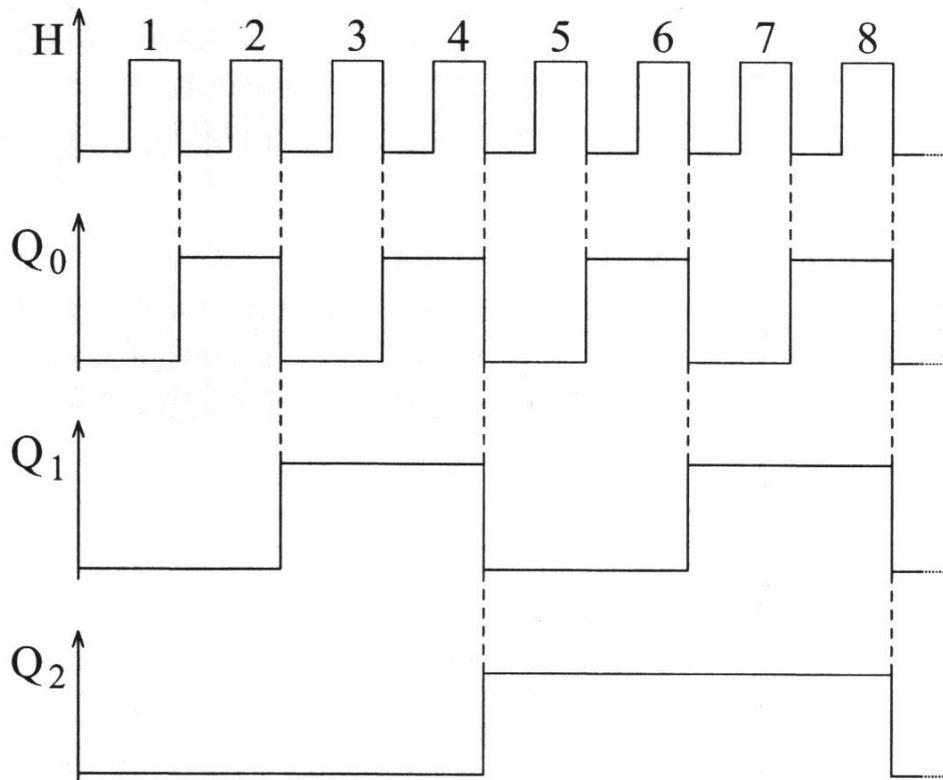
- ✓ liste des états successifs :

Impulsion	Q_2	Q_1	Q_0
état initial	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
5	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

- ✓ on a réalisé un compteur s'incrémentant de 1 à chaque top d'horloge
- ✓ les sorties Q_0 , Q_1 , Q_2 fournissent des horloges de période $T/2$, $T/4$ et $T/8$ (diviseurs de fréquence)

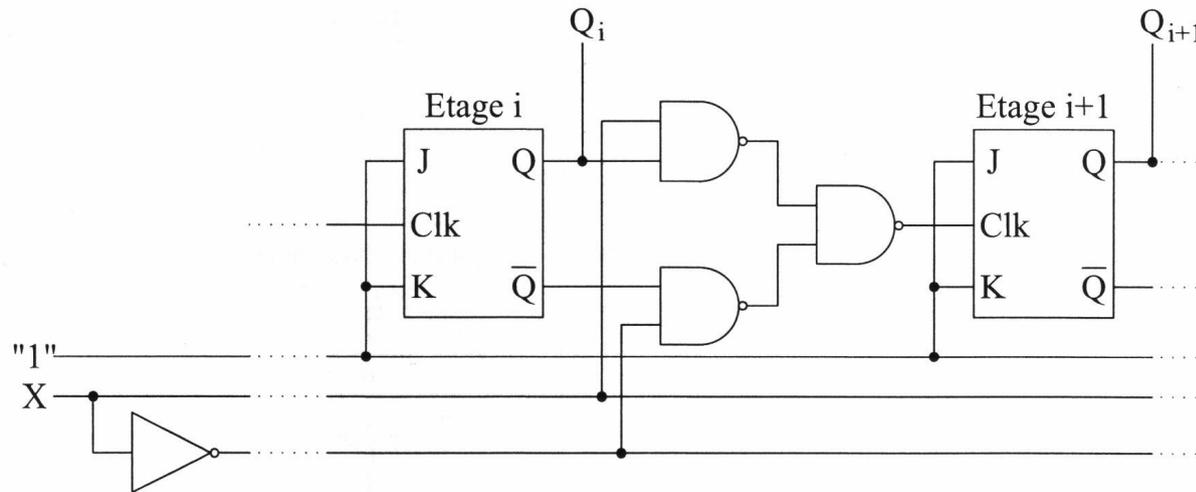
Les compteurs

- ✓ dans une bascule maître-esclave, l'entrée est validée sur le front montant de l'horloge (stockée dans la bascule maître) et l'état final de la bascule est validé sur le front descendant de l'horloge (transfert de l'état de la bascule maître vers la bascule esclave)



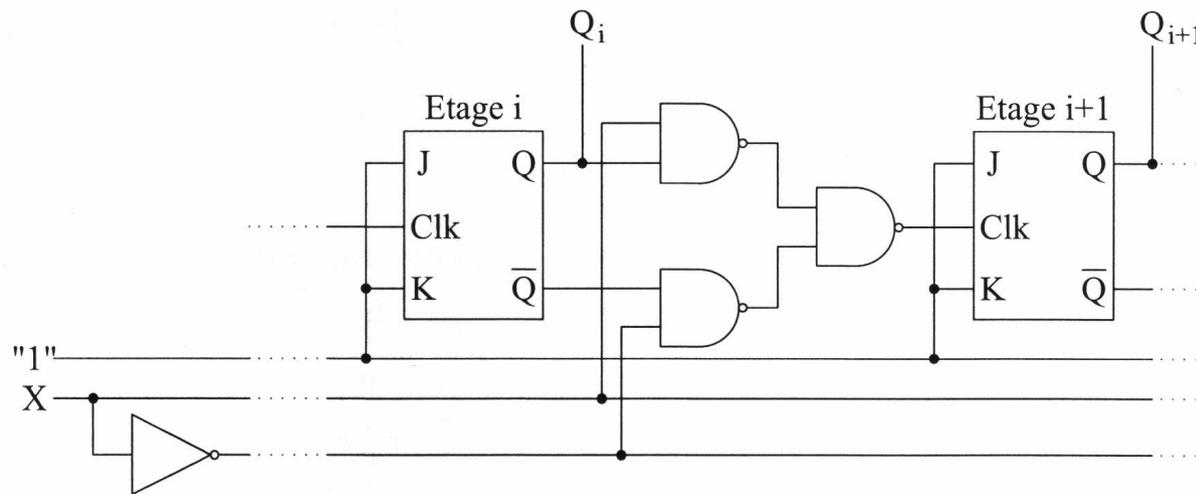
Les compteurs

- compteur-décompteur asynchrone
 - ✓ compteur : déclenche chaque bascule quand celle de rang inférieur passe de 1 à 0
 - ✓ décompteur : il faut déclencher quand la bascule de rang inférieur passe de 0 à 1 en utilisant la sortie Q



Les compteurs

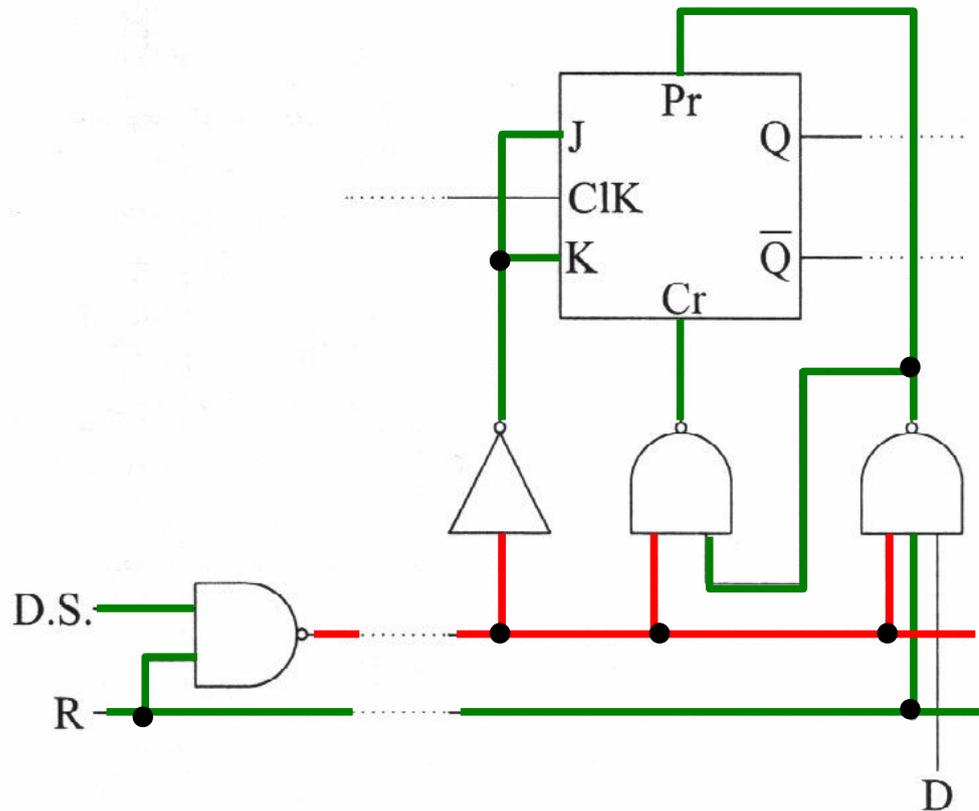
- compteur-décompteur asynchrone
 - ✓ compteur : déclenche chaque bascule quand celle de rang inférieur passe de 1 à 0
 - ✓ décompteur : il faut déclencher quand la bascule de rang inférieur passe de 0 à 1



- ✓ la ligne de commande X permet de sélectionner le mode :

Les compteurs

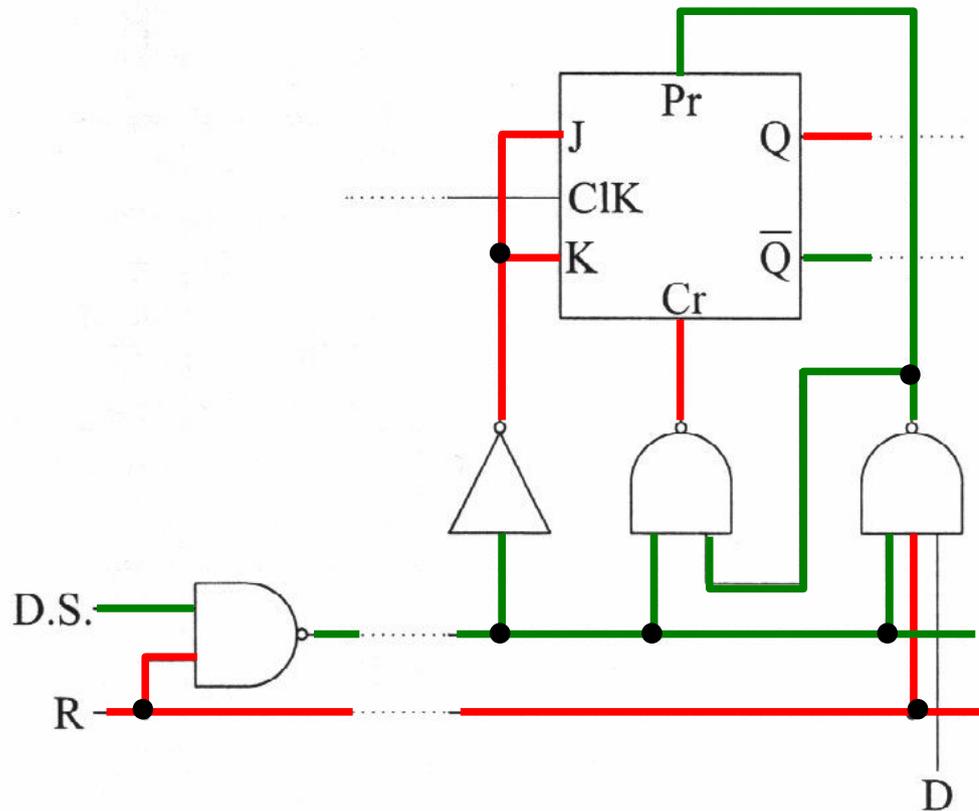
- remise à zéro et chargement d'un compteur



- ✓ en fonctionnement normal
DS = R = 1
⇒ J = K = Pr = Cr = 1 sur chaque bascule

Les compteurs

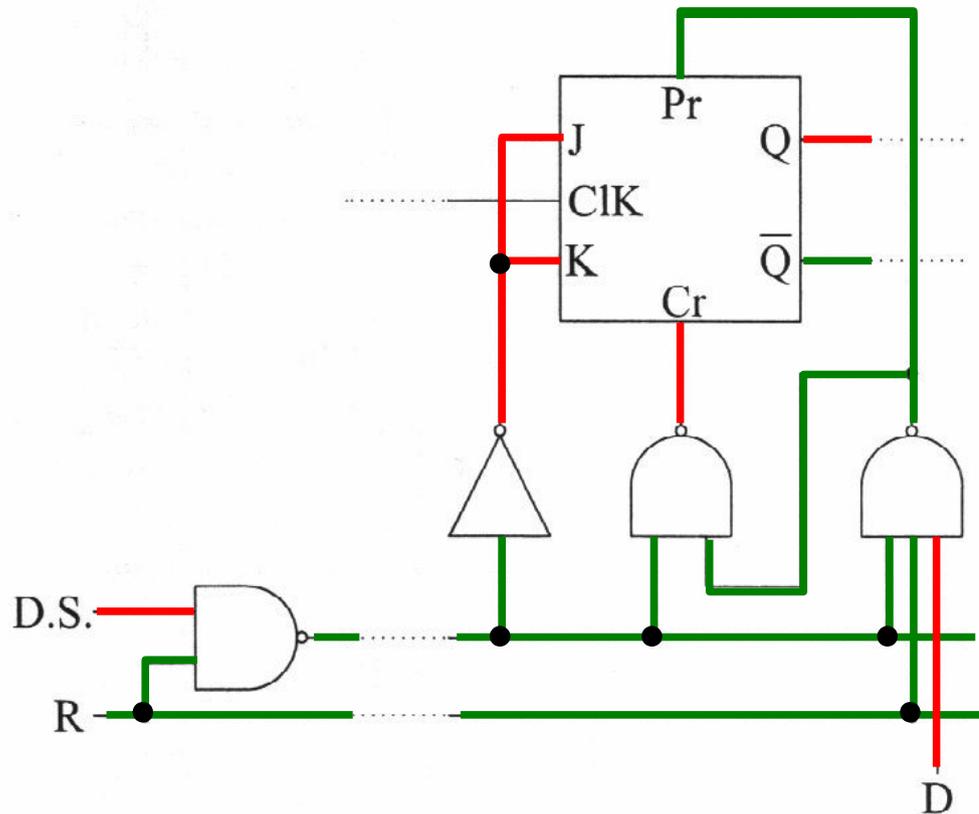
- remise à zéro et chargement d'un compteur



- ✓ RAZ :
 $R = 0 \Rightarrow J = K = 0$ (interdit tout basculement sur Clk)
 $\Rightarrow (Pr = 1, Cr = 0) \Rightarrow Q = 0$

Les compteurs

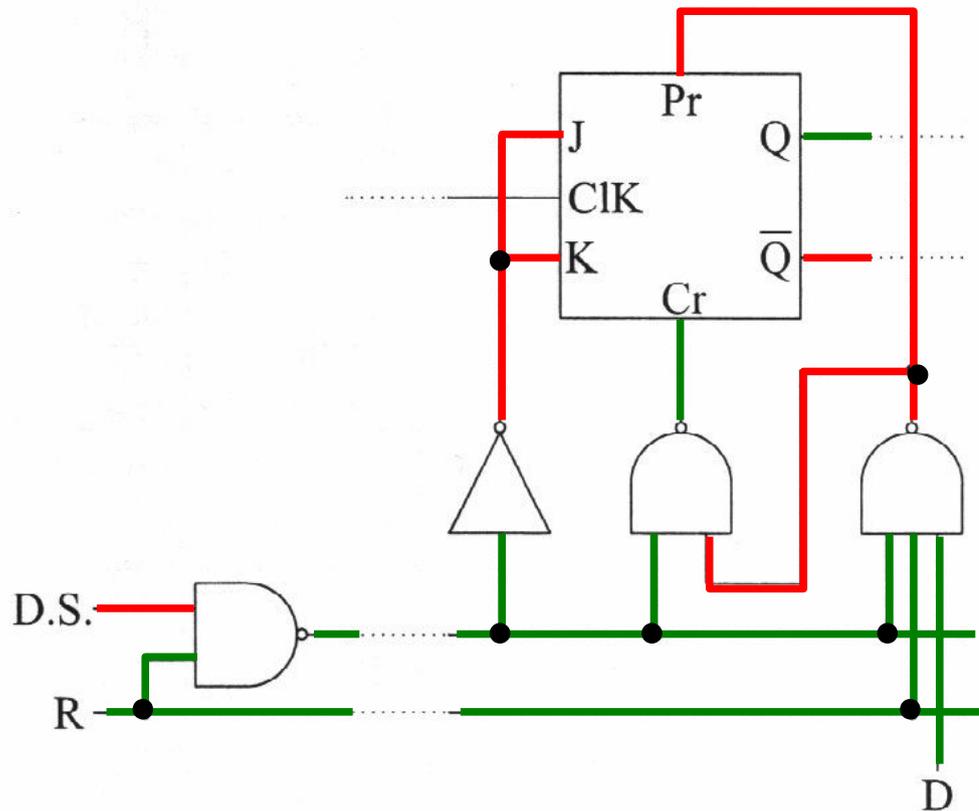
- remise à zéro et chargement d'un compteur



- ✓ Chargement : ($DS = 0, R = 1$)
⇒ $J = K = 0$ (interdit tout basculement sur Clk)
 $D = 0 \Rightarrow (Pr = 1, Cr = 0) \Rightarrow Q = 0$

Les compteurs

- remise à zéro et chargement d'un compteur



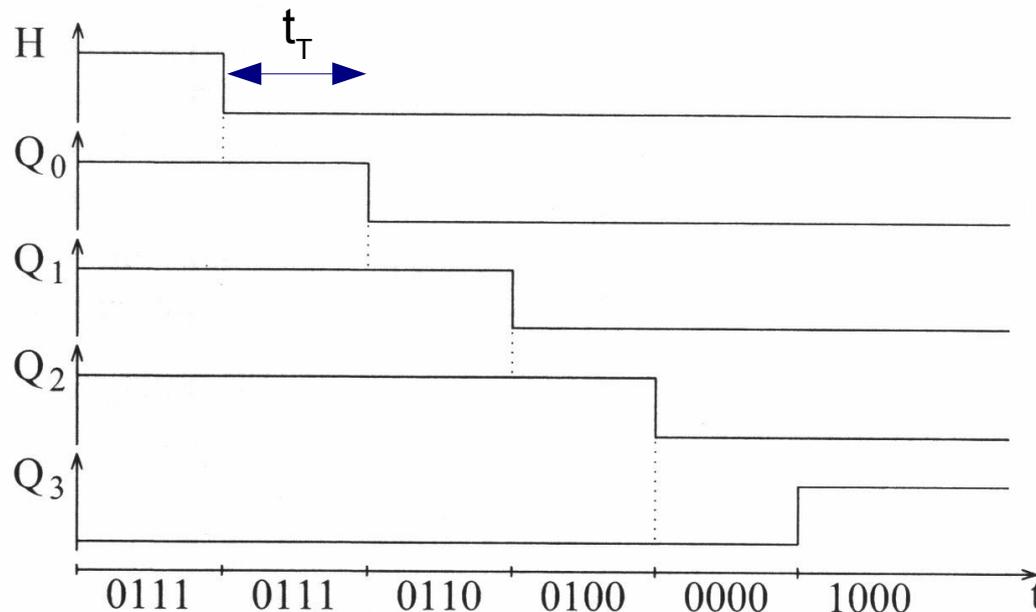
- ✓ Chargement : ($DS = 0, R = 1$)
⇒ $J = K = 0$ (interdit tout basculement sur Clk)
 $D = 1 \Rightarrow (Pr = 0, Cr = 1) \Rightarrow Q = 1$
dans les deux cas la bascule prend la valeur de D

Les compteurs

- compteur à cycle incomplet
 - ✓ pour compter jusqu'à un nombre N qui ne soit pas forcément une puissance de 2 (10 par exemple)
 - ✓ compteur de n bascules avec $2^n > N$ et asservissement du C_r pour remettre le compteur à 0 tous les N coups
 - ✓ exemple : compteur modulo 10

Les compteurs

- inconvénient des compteurs asynchrones
 - ✓ à cause du temps de réponse non nul des bascules, le signal d'horloge ne parvient pas simultanément à toutes les bascules
 - ✓ exemple : passage de 0111 à 1000 sur un compteur modulo 16, avec un temps de réponse t_T pour toutes les bascules



on passe par les états transitoires 0110, 0100, et 0000 qui sont faux

Les compteurs

- compteurs synchrones

- toutes les bascules reçoivent en parallèle le même signal d'horloge
- pour faire décrire au compteur une séquence déterminée, il faut définir les entrées J et K de chaque bascule à chaque top d'horloge, en utilisant la table de transition des bascules J-K

Q_n	Q_{n+1}	J_n	K_n
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

- on peut choisir $J = K$

Les compteurs

- exemple d'un compteur 3 bits selon le code binaire pur

#top	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

chaque ligne correspond à une même tranche de temps

Les compteurs

- exemple d'un compteur 3 bits selon le code binaire pur
 - ✓ table donnant la combinaison J-K qui permet de passer d'une ligne à la suivante

#top	Q_2	Q_1	Q_0	$J_2=K_2$	$J_1=K_1$	$J_0=K_0$
0	0	0	0	0	0	1
1	0	0	1			
2	0	1	0			
3	0	1	1			
4	1	0	0			
5	1	0	1			
6	1	1	0			
7	1	1	1			
8	0	0	0			

chaque ligne correspond à une même tranche de temps

Les compteurs

- exemple d'un compteur 3 bits selon le code binaire pur
 - ✓ table donnant la combinaison J-K qui permet de passer d'une ligne à la suivante

#top	Q_2	Q_1	Q_0	$J_2=K_2$	$J_1=K_1$	$J_0=K_0$
0	0	0	0	0	0	1
1	0	0	1	0	1	1
2	0	1	0			
3	0	1	1			
4	1	0	0			
5	1	0	1			
6	1	1	0			
7	1	1	1			
8	0	0	0			

chaque ligne correspond à une même tranche de temps

Les compteurs

- exemple d'un compteur 3 bits selon le code binaire pur
 - ✓ table donnant la combinaison J-K qui permet de passer d'une ligne à la suivante

#top	Q_2	Q_1	Q_0	$J_2=K_2$	$J_1=K_1$	$J_0=K_0$
0	0	0	0	0	0	1
1	0	0	1	0	1	1
2	0	1	0	0	0	1
3	0	1	1			
4	1	0	0			
5	1	0	1			
6	1	1	0			
7	1	1	1			
8	0	0	0			

chaque ligne correspond à une même tranche de temps

Les compteurs

- exemple d'un compteur 3 bits selon le code binaire pur
 - ✓ table donnant la combinaison J-K qui permet de passer d'une ligne à la suivante

#top	Q_2	Q_1	Q_0	$J_2=K_2$	$J_1=K_1$	$J_0=K_0$
0	0	0	0	0	0	1
1	0	0	1	0	1	1
2	0	1	0	0	0	1
3	0	1	1	1	1	1
4	1	0	0	0	0	1
5	1	0	1	0	1	1
6	1	1	0	0	0	1
7	1	1	1	1	1	1
8	0	0	0			

chaque ligne correspond à une même tranche de temps

Les compteurs

✓ on peut vérifier

$$J_0 = K_0 = 1$$

$$J_1 = K_1 = Q_0$$

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

✓ et plus généralement :

$$J_0 = K_0 = 1$$

$$J_i = Q_0 \cdot Q_1 \cdot \dots \cdot Q_{i-1} = J_{i-1} \cdot Q_{i-1}$$

➤ on peut procéder de même pour un décompteur

#top	Q_2	Q_1	Q_0	$J_2=K_2$	$J_1=K_1$	$J_0=K_0$
0	0	0	0	0	0	1
1	0	0	1	0	1	1
2	0	1	0	0	0	1
3	0	1	1	1	1	1
4	1	0	0	0	0	1
5	1	0	1	0	1	1
6	1	1	0	0	0	1
7	1	1	1	1	1	1
8	0	0	0			

$$J_0 = K_0 = 1$$

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

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

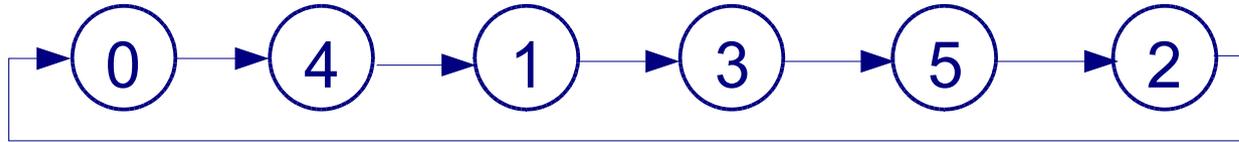
$$J_i = J_{i-1} \cdot Q_{i-1}$$

Générateurs de séquence

- but : afficher une suite prédéterminée de nombres
- principe :
 - écrire la suite des transitions à réaliser
 - déterminer les entrées des bascules qui vont réaliser ces transitions
 - exprimer ces entrées en fonction des sorties attendues

Générateurs de séquence

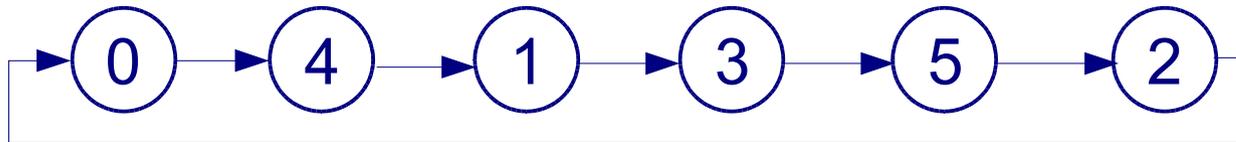
- exemple : réaliser la séquence



avec des bascules T

Générateurs de séquence

- exemple : réaliser la séquence



avec des bascules T

- table d'évolution

états	S_n			S_{n+1}					
	Q_2	Q_1	Q_0	Q'_2	Q'_1	Q'_0	T_2	T_1	T_0
0	0	0	0	1	0	0	1	0	0
4	1	0	0	0	0	1	1	0	1
1	0	0	1	0	1	1	0	1	0
3	0	1	1	1	0	1	1	1	0
5	1	0	1	0	1	0	1	1	1
2	0	1	0	0	0	0	0	1	0

Générateurs de séquence

- relations entre les entrées T des bascules et les sorties
 - avec des tables de Karnaugh

T_0

Q_2	Q_0Q_1	00	01	11	10
0		0	0	0	0
1		1	X	X	1

$$T_0 = Q_2$$

T_1

Q_2	Q_0Q_1	00	01	11	10
0		0	1	1	1
1		0	X	X	1

$$T_1 = Q_0 + Q_1$$

T_2

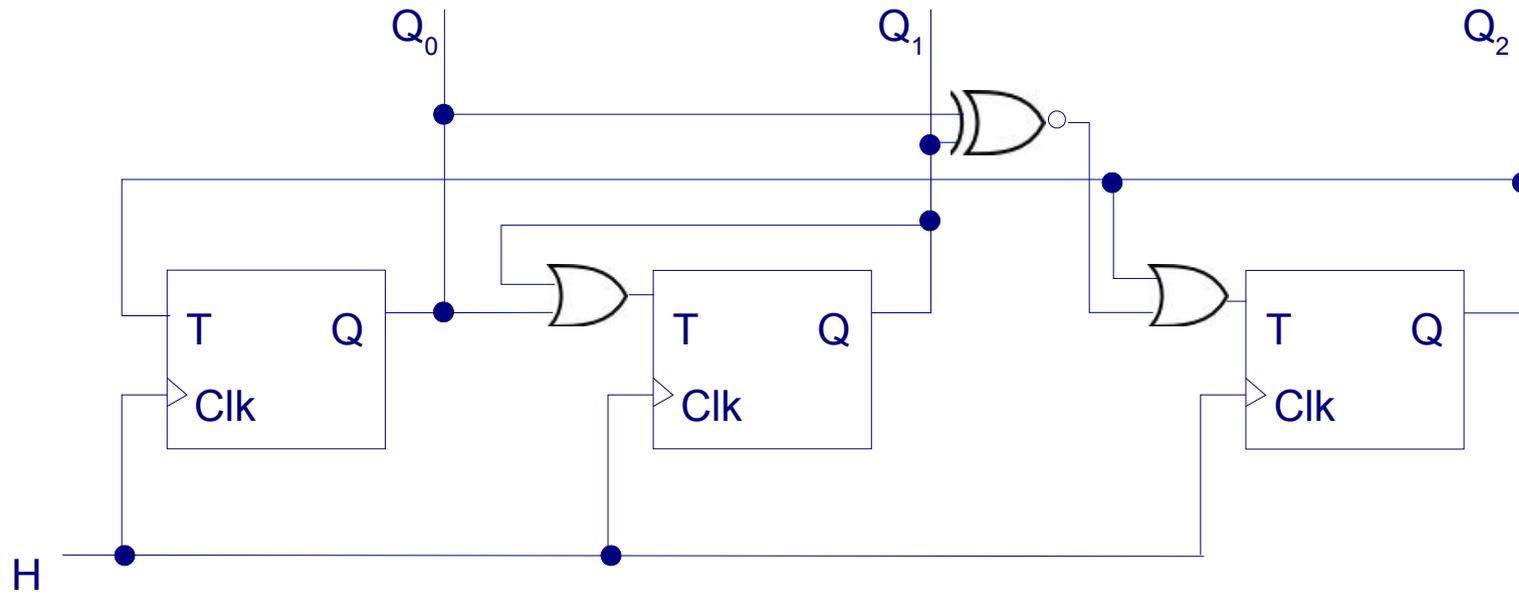
Q_2	Q_0Q_1	00	01	11	10
0		1	0	1	0
1		1	X	X	1

$$T_2 = \overline{Q_0} \overline{Q_1} + Q_0 Q_1 + Q_2$$

$$= Q_0 \oplus Q_1 + Q_2$$

Générateurs de séquence

- implémentation



à suivre...

