

**INSTITUT DE LA PROMOTION SUPERIEURE DU TRAVAIL**

**ARCHITECTURE DES MACHINES  
&  
SYSTEMES INFORMATIQUES**

**LOGIQUE SEQUENTIELLE**

**A. M'ZOUGH**

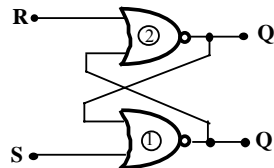
# LOGIQUE SEQUENTIELLE

## I. LES BASCULES

### I. 1. La bascule R-S

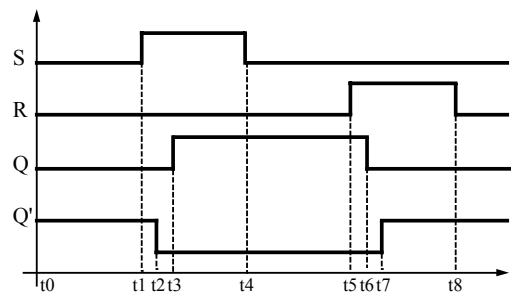
Une bascule est un élément de mémorisation unitaire. Elle permet de mémoriser l'état d'une variable booléenne pour un temps donné.

Exemple :



- Etat initial : au départ  $S = R = 0$ .
- Hypothèse : supposons que  $Q = 0$ .
- Vérifions du fait du rebouclage, que cette hypothèse est cohérente.  
 $Q = S = 0 \Rightarrow Q' = 1$  ;  $R = 0$  et  $Q' = 1 \Rightarrow Q = 0$  donc vrai

Etudions le comportement de cette bascule, lors de l'arrivée de signaux sur S et R selon le chronogramme suivant :



#### Etat initial

- En  $t_0$  :  $S = R = Q = 0$  et  $Q' = 1$ .

#### NOR 1

- En  $t_1$  : de l'état  $S = Q = 0$  on passe à l'état  $Q = 0$  et  $S = 1 \Rightarrow$

- En  $t_2$  : la sortie  $Q'$  passe de 1 à 0.

#### NOR 2

- En  $t_2$  : de l'état  $R = 0$  et  $Q' = 1$  on passe à l'état  $R = Q' = 0 \Rightarrow$

- En  $t_3$  : la sortie  $Q$  passe de 0 à 1.

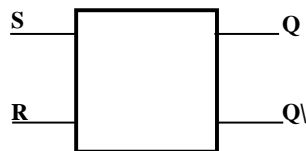
Ceci a pour effet de faire passer le NOR 1 de l'état  $S = 1$  et  $Q = 0$  à l'état  $S = Q = 1$  sans effet sur la sortie du NOR 1.

- En  $t_4$  : S repasse à 0, ce qui n'a pas d'effet sur l'état des sorties Q et Q'.

Le raisonnement est identique avec R. Ce type de bascule est connu sous le nom de R-S (Reset-Set). Les sorties Q et Q' sont inverses l'une de l'autre et traditionnellement appelées Q et Q'.

Toutes les combinaisons possibles des valeurs de R et S et Q peuvent se représenter sur une **table de transitions**, analogue à une table de vérité, quant à la présentation, mais différente quant au raisonnement nécessaire pour l'établir. Il faut en effet donner la valeur future de la sortie en fonction de sa valeur présente et de celle des variables R et S.

La représentation symbolique et la **table de transitions** sont les suivantes :



$S^n$	$R^n$	$Q^n$	$Q^{n+1}$	Remarques
0	0	0	0	Maintien de l'état précédent c'est l'opération (HOLD)
0	0	1	1	
0	1	0	0	Mise à 0 de la bascule c'est l'opération (RESET)
0	1	1	0	
1	0	0	1	Mise à 1 de la bascule c'est l'opération (SET)
1	0	1	1	
1	1	0	X	Combinaisons interdites
1	1	1	X	

L'état  $R = S = 1$  est interdit pour deux raisons :

- Les deux NORs ont leur sortie à 0 ce qui contraire à la définition : elles doivent être complémentées.
- Si R et S repassent simultanément à 0, les deux NORs auront 0 0 sur leurs entrées et les sorties vont repasser à 1 ce qui fait que le résultat final est imprévisible soit 1, soit 0, et ceci peut être obtenu après quelques oscillations !

La table de transitions décrit le comportement de la bascule, On peut tirer comme en logique combinatoire, une forme algébrique de la fonction séquentielle représentée.

L'équation de la bascule s'écrit :  $Q^{n+1} = S^n \cdot R^n \cdot Q^n + S^n \cdot R^n \cdot Q^n + S^n \cdot R^n \cdot Q^n$ .

Si l'on tient compte des deux combinaisons interdites :

$S^n R^n$	00	01	11	10
$Q^n$				
0	0	0	*	1
1	1	0	*	1

on obtient :

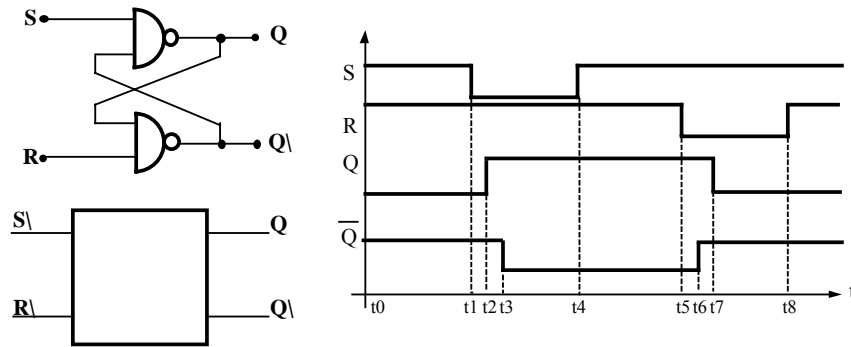
$$Q^{n+1} = S^n + R^n \cdot Q^n$$

On peut dire qu'une fonction séquentielle prend en considération **la valeur passée** de sa sortie, ou encore qu'elle se **souvient** de cette valeur passée, pour déterminer sa valeur logique **future**.

On rencontre parfois cette table, sous la forme réduite suivante :

$S^n$	$R^n$	$Q^{n+1}$
0	0	$Q^n$
0	1	0
1	0	1
1	1	X

On retrouve un fonctionnement identique avec des NANDs.



## II. NOTIONS DE TEMPS

### II. 1. Instant $n$ ( $t_n$ ) et instant $n+1$ ( $t_{n+1}$ )

L'instant  $n$  ( $t_n$ ) représente un temps arbitraire jusqu'auquel les entrées et les sorties ont une valeur stable donnée.

L'instant  $n+1$  ( $t_{n+1}$ ) représente l'instant où la valeur d'une sortie est stable suite à une transition sur l'une des entrées à l'instant  $n$  ( $t_n$ ).

### II. 2. Impulsion et niveau

En logique combinatoire on ne considère que les états stables, les signaux étant représentés par des niveaux. Les états transitoires associés aux changements de niveaux sont sans intérêt. En logique séquentielle le problème n'est pas le même. Supposons que  $S=R=Q=0$ , et que  $S$  passe à 1 et  $R$  reste à 0 jusqu'à **un instant arbitraire**. Si on était en logique combinatoire,  $S$  peut repasser à 0 n'importe quand. Ici  $S$  doit rester stable **"un certain temps"** afin que le changement d'état de la bascule s'effectue correctement. En effet, si  $S$  repasse à 0 avant  $t_3$ ,  $Q$  n'est pas encore à 1 et on passe par l'état  $S=Q=0$  qui a tendance à refaire passer  $Q$  à 1. Donc, il est raisonnable que  $S$  reste à 1 pour une **"courte"** période de temps, ensuite  $S$  peut repasser à 0 n'importe quand puisque la transition est sans effet. Mais il faut que ceci soit réalisé avant la prochaine transition sur  $R$ . Que signifie courte ? difficile à préciser, un temps correspondant à la traversée de quelques couches logiques ? Mais ce temps peut être beaucoup plus long et également considéré comme une impulsion. Est-ce alors pas un niveau ? la réponse est donnée par la façon dont les circuits répondent.

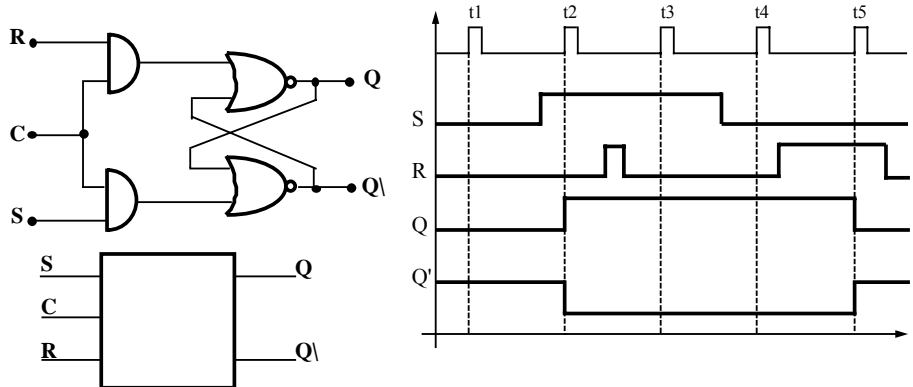
### II. 3. Transitions

Une impulsion quelque soit sa durée est constituée de deux transitions ; mais elle représente un seul événement, c'est à dire ne conduisant qu'à une seule réponse. La réponse peut se produire sur le front montant ou sur le front descendant de l'impulsion. Une impulsion peut être positive ou négative périodique ou apériodique.

## III. SYSTEME SYNCHRONES

Si un système logique comporte plusieurs bascules la conception est simplifiée si toutes les transitions s'effectuent au même moment. Ceci est réalisé en utilisant une source commune d'impulsions appelée horloge. La bascule doit donc avoir une entrée supplémentaire pour indiquer

**quand** la transition doit avoir lieu. Ceci est réalisé en conditionnant les entrées R et S par le signal d'horloge.

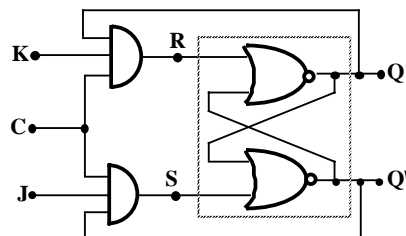


$S^n$	$R^n$	$C^n$	$Q^{n+1}$	Remarques
X	X	0	$Q^n$	Aucun effet si le signal d'horloge est à zéro
0	0	1	$Q^n$	Maintien de l'état précédent : HOLD
0	1	1	0	Mise à 0 de la bascule : RESET
1	0	1	1	Mise à 1 de la bascule : SET
1	1	1	X	Combinaison interdite

#### IV. AUTRES TYPES DE BASCULES

##### IV. 1. La bascule J-K

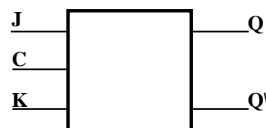
Il serait très utile si nous pouvons éliminer la combinaison interdite de la bascule R-S. Considérons le circuit logique suivant :



En rebouclant les sorties Q et  $Q'$  sur les entrées et en les conditionnant avec des entrées de contrôle Set et Reset externes (que nous appellerons maintenant respectivement J et K), nous garantissons que les entrées R et S internes ne seront jamais simultanément à 1 (Ceci suppose que Q et  $Q'$  ne sont jamais à 1 en même temps).

En plus pour éviter l'état interdit, nous introduisons une nouvelle possibilité : complémentarité de l'état courant. Par exemple, si  $Q = 1$  et  $Q' = 0$ , et  $J=K=1$ , alors les entrées internes sont dans l'état :  $R = 1$  et  $S = 0$ . Ce qui fait basculer Q à 0 et  $Q'$  à 1. Dans le cas contraire si  $Q = 0$  et  $Q' = 1$ , et  $J=K=1$ , alors les entrées internes sont dans l'état :  $R = 0$  et  $S = 1$ . Ce qui fait basculer Q à 1 et  $Q'$  à 0.

La bascule R-S est peu utilisée supplantée par la bascule J-K qui est **toujours synchrone** et dont la représentation symbolique et la table de transitions est la suivante :



$J^n$	$K^n$	$Q^n$	$C^n$	$Q^{n+1}$	Remarques
X	X	0	0	0	Aucun effet si le signal d'horloge est à zéro Maintien de l'état précédent
X	X	1	0	1	
0	0	0	1	0	Maintien de l'état précédent, c'est l'opération (HOLD)
0	0	1	1	1	
0	1	0	1	0	Mise à 0 de la bascule c'est l'opération (RESET)
0	1	1	1	0	
1	0	0	1	1	Mise à 1 de la bascule c'est l'opération (SET)
1	0	1	1	1	
1	1	0	1	1	Complémentation c'est l'opération (TOGGLE)
1	1	1	1	0	

La table réduite est la suivante :

$J^n$	$K^n$	$C^n$	$Q^{n+1}$
X	X	0	$Q^n$
0	0	1	$Q^n$
0	1	1	0
1	0	1	1
1	1	1	$Q^n \setminus$

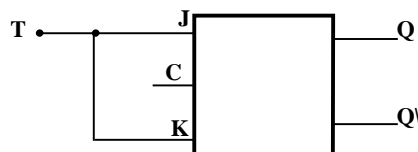
**Attention : J** analogue à **S** et **K** analogue à **R**.

Equation de la bascule J-K :

$$Q^{n+1} = K^n \setminus Q^n + J^n \cdot Q^n$$

#### IV. 2. La bascule T (Trigger ou Toggle)

Elle est peu utilisée on la rencontre surtout dans les compteurs. Elle a une seule entrée qui provoque la complémentation de l'état mémorisé à chaque fois que l'entrée est positionnée à 1. Habituellement, cette bascule n'existe pas dans les catalogues en tant que circuit intégré standard parce qu'elle est facile à construire à partir de la bascule J-K. Sa représentation symbolique est la suivante :



On a  $J = K = T$ , l'équation de la bascule T en remplaçant J et K dans l'équation de la bascule J-K par T est la suivante :  $Q^{n+1} = T^n \setminus Q^n + T^n \cdot Q^n = T^n \approx Q^n$ .

$$Q^{n+1} = T^n \approx Q^n$$

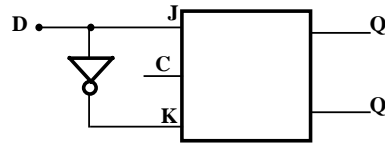
Si  $T^n = 0$  alors  $Q^{n+1} = Q^n$  pas de changement. Si  $T^n = 1$  alors  $Q^{n+1} = Q^n \setminus$ .

La bascule T peut être synchrone d'une horloge ou non (asynchrone).

Dans le cas asynchrone chaque impulsion sur T fait changer l'état de la bascule.

#### IV. 3. La bascule D (delay)

C'est une bascule très utilisée, elle peut être réalisée à partir de la bascule J-K. Sa représentation symbolique est la suivante :



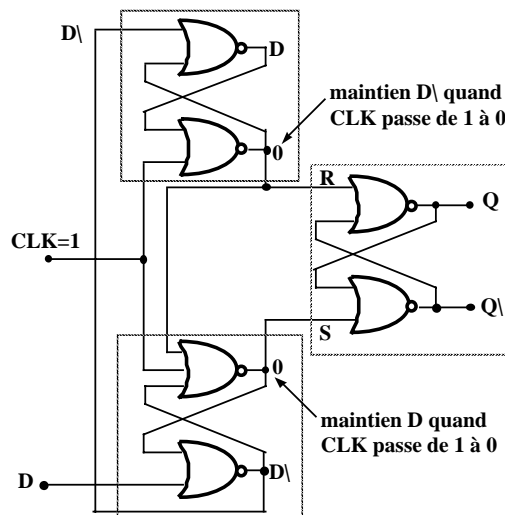
On a  $J = D$  et  $K = D'$ , l'équation de la bascule D en remplaçant J et K dans l'équation de la bascule J-K par D et  $D'$  est la suivante :  $Q^{n+1} = D' \cdot Q^n + D \cdot Q^n = D \cdot Q^n + D' \cdot Q^n = D^n$ .

$$Q^{n+1} = D^n$$

#### IV. 3. 1. la bascule D à déclenchement sur front (Edge Triggered)

Ce type de bascule est celui le plus utilisé dans les calculateurs, son fonctionnement interne est complexe mais son utilisation est celle qui présente les caractéristiques les plus simples, car l'entrée de donnée est mémorisée par l'un des deux fronts de l'horloge. La figure ci-dessous montre deux étages de trois bascules R-S communicantes entre elles (étage d'entrée deux bascules haute et basse, étage de sortie la bascule centrale). La bascule du bas de l'étage d'entrée échantillonne l'entrée D alors que la bascule du haut maintient  $D'$ . De plus, la sortie de la bascule du bas de l'étage d'entrée commande l'entrée S (Set) de la bascule centrale (étage de sortie), alors que la bascule de l'étage du haut commande l'entrée R (Reset) de la bascule centrale.

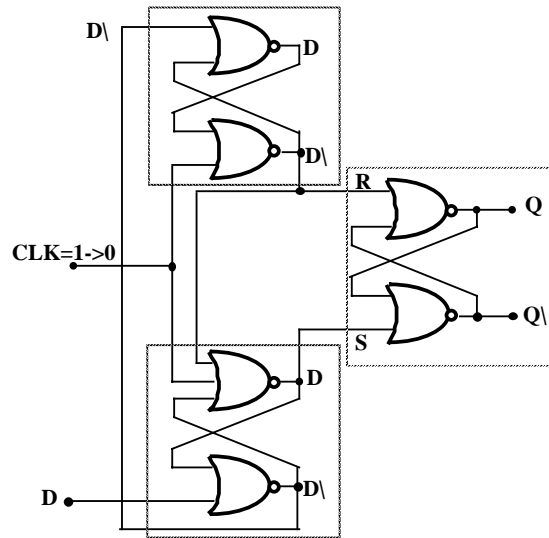
La figure ci-dessous montre l'état du circuit quand l'horloge est au niveau logique haut ( $CLK = 1$ ). L'horloge force les sorties des bascules de l'étage d'entrée à l'état 0, ainsi la bascule R-S centrale est dans l'état de maintien (HOLD). N'importe quel changement sur l'entrée D sera échantillonné par les deux bascules du haut et du bas, mais ces changements sont inhibés au niveau de l'étage final.



Bascule D à déclenchement sur front descendant quand l'horloge est à l'état haut.

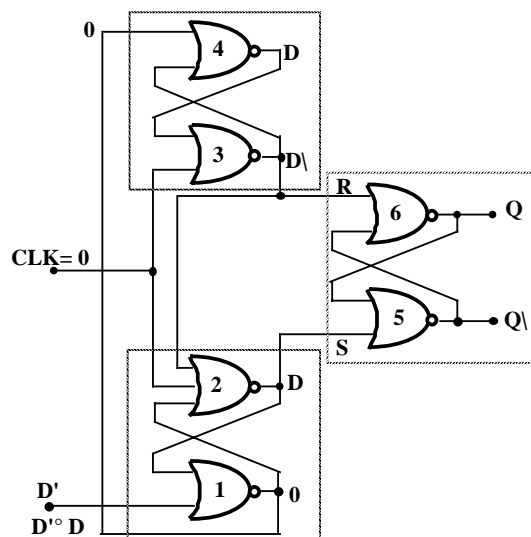
La figure ci-dessous montre ce qui se passe quand l'horloge passe de l'état haut à l'état bas ( $CLK = \text{transition } 1 \rightarrow 0$ ). La porte NOR de la bascule du bas dont la sortie est à zéro fonctionne maintenant comme un inverseur. La valeur de l'entrée D précédemment échantillonnée est présentée maintenant à l'entrée S, pendant que le complément de la valeur de D précédemment échantillonnée ( $D'$ ) est

présenté sur l'entrée R. Si  $D = 1$ , la bascule R-S de l'étage de sortie aura sur sa sortie  $Q = 1$ . Autrement, elle aura  $Q = 0$ .



Bascule D à déclenchement sur front descendant quand l'horloge passe de l'état haut à l'état bas

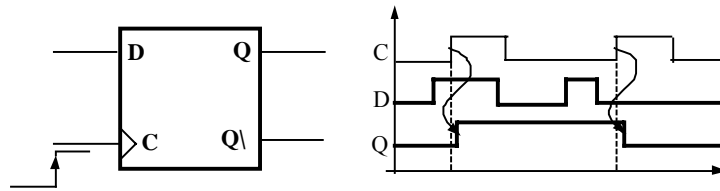
Que se passe-t-il maintenant quand l'horloge est à l'état bas ( $CLK = 0$ ), mais que  $D$  change de valeur ? La figure ci-dessous montre l'état de la bascule quand l'entrée de donnée change et devient  $D'$  (la valeur  $D'$  est différente de la valeur  $D$ ). Cette nouvelle valeur sur l'entrée force la sortie de la porte NOR1 à 0, ce qui garantit que chacune des portes 2, 4 et 5 maintienne sa précédente valeur. Par conséquent, la bascule de l'étage de sortie voit ses entrées R et S non affectées et donc les sorties sont dans l'état de maintien (HOLD). La variation de l'entrée de la valeur  $D$  en  $D'$  tant que l'horloge reste au niveau bas ou même si elle change de l'état bas vers l'état haut n'a pas de conséquence sur une bascule à déclenchement sur front descendant. Pour obtenir une bascule avec déclenchement sur front montant il suffit de prévoir un inverseur sur le chemin du signal CLK.



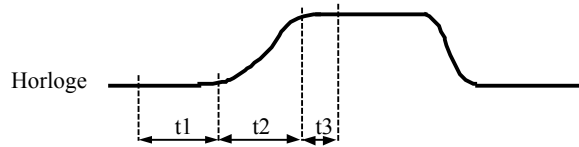
Bascule D à déclenchement sur front descendant quand l'horloge est à l'état bas

La représentation symbolique et le chronogramme de fonctionnement sont les suivants :





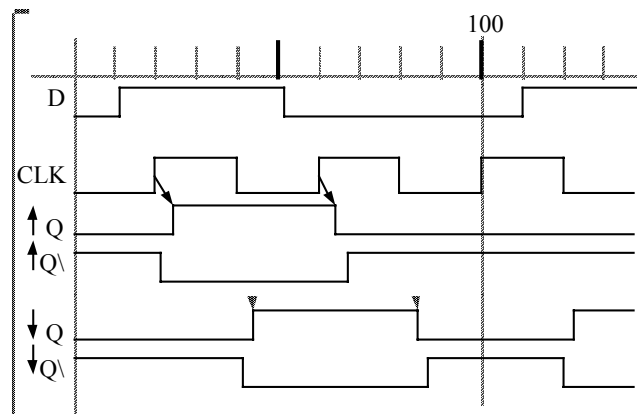
Pour que le fonctionnement soit correct, il faut que l'entrée de donnée soit stable un certain temps avant et après le front actif de l'horloge.



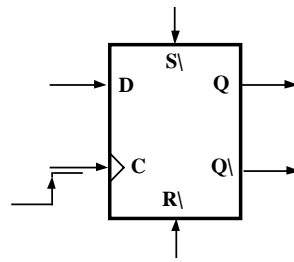
- $t_1$  : temps de prépositionnement (Setup Time).
- $t_2$  : temps de montée (Rise Time).
- $t_3$  : temps de maintien (Hold Time).

Un examen du circuit de la bascule à déclenchement sur front permet d'apprécier la nécessité des spécifications des temps de Setup et de Hold. Si l'entrée de donnée change d'état très proche de l'arrivée du front d'horloge approprié, sa valeur ne peut pas être correctement maintenue par les bascules de l'étage d'entrée. Pour de tels dispositifs, on voit aussi pourquoi les délais de propagation peuvent être assez longs.

La figure ci-dessous donne un chronogramme pour des bascules D fonctionnant sur front montant et sur front descendant. Les sorties changent après un petit délai de propagation à partir du front descendant ou montant. La dernière transition (temps de Setup 100) montre comment les dispositifs échantillonnent leur entrée. Dans ce cas, l'entrée change de 0 à 1 au milieu de l'état haut de l'horloge. Ce changement reste inaperçu par la bascule à déclenchement sur front montant (état 0 en sortie), mais reconnu par la bascule à déclenchement sur front descendant.



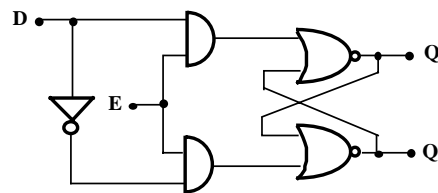
Ces caractéristiques sont fournies par le constructeur une valeur typique pour la technologie TTL est de 35 ns pour  $(t_1 + t_2 + t_3)$ . Certaines bascules D à déclenchement sur front sont disponibles avec des entrées de mise à 1 (PRESET) et de mise à 0 (RESET) utilisées pour l'état initial. La représentation symbolique et la table de transition de cette bascule sont les suivantes :



$R^n$	$S^n$	$D^n$	$C^n$	$Q^{n+1}$	Remarques
1	0	X	X	1	Mise à 1 (PRESET)
0	1	X	X	0	Mise à 0 (RESET)
1	1	X	0	$Q^n$	Effet de mémorisation de la valeur précédente
1	1	0	1	0	Ecriture de 0
1	1	1	1	1	Ecriture de 1
0	0	X	X	X	Combinaisons interdites (Sortie indéterminée)

#### IV. 3. 2. la bascule D à fonctionnement sur niveau (LATCH)

L'entrée E n'est pas une horloge mais une validation tant que E vaut 1, la bascule est "transparente" la sortie Q recopie (suit) l'état de l'entrée D. Lorsque E passe à 0 la sortie Q mémorise la dernière valeur présente sur l'entrée D. Ce type de bascule est dit "transparent" (LATCH).



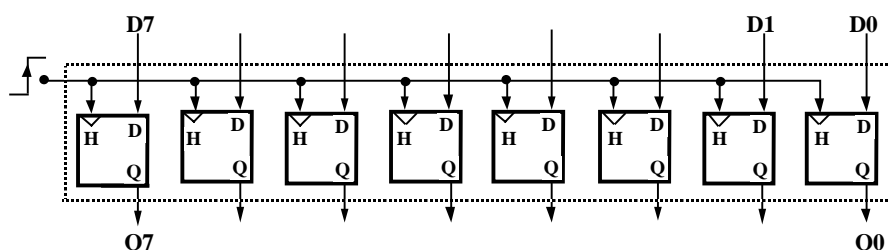
### V. CONCLUSION

- D à déclenchement sur front : la plus utilisée car l'instant de changement est parfaitement défini. Elle est toujours synchrone.
- D à fonctionnement sur niveau "transparent" : utilisée dans certains contextes où l'on doit laisser passer l'information à certains moments et la figer à d'autres moments.
- J-K : elle est toujours synchrone.
- T : utilisée dans des cas particuliers (Synchrone/Asynchrone).
- R-S : peu utilisée (Synchrone/Asynchrone).

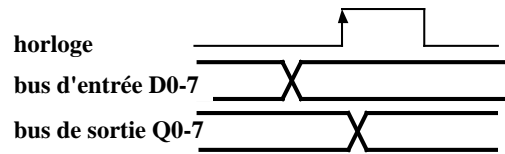
### VI. ASSOCIATION DE BASCULES : LES REGISTRES.

#### VI. 1. Le registre parallèle

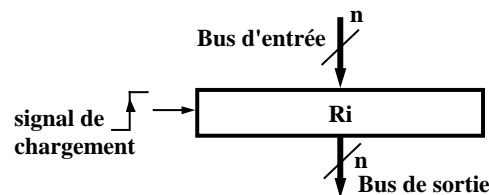
Considérons l'association parallèle suivante de bascules D :



On obtient ici un registre de 8 bits : les 8 bascules sont chargées en parallèle par la transition 01 du signal d'échantillonnage. les entrées sont censées véhiculer des informations de même nature : les lignes d'entrées constituent le **BUS** d'entrée. Les sorties Q constituent de même le bus de sortie. L'évolution de ce registre peut-être symbolisée par le chronogramme suivant :

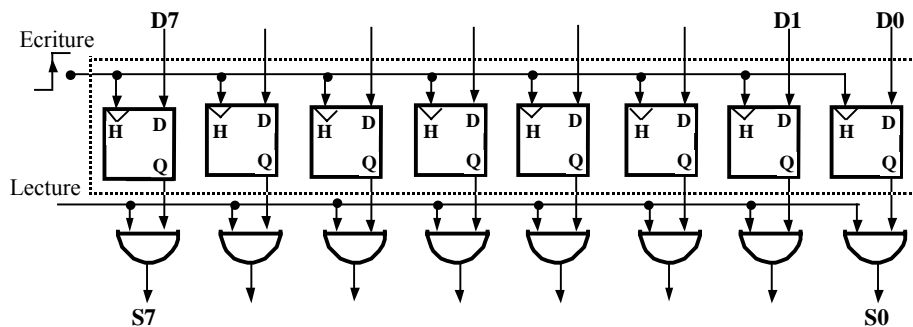


Une façon synthétique de représenter bus et registre est la suivante :



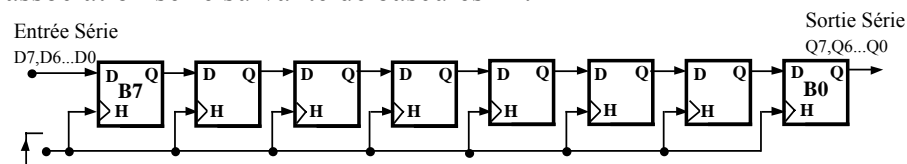
$n$ : nombre de bits du bus

### Registre à lecture et écriture parallèle



## VI. 2. Le registre série (registre à décalage)

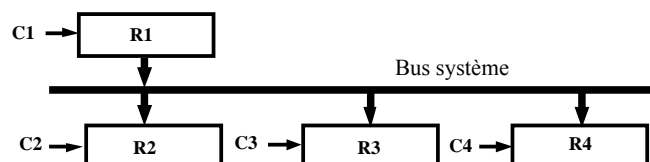
Considérons l'association série suivante de bascules D :



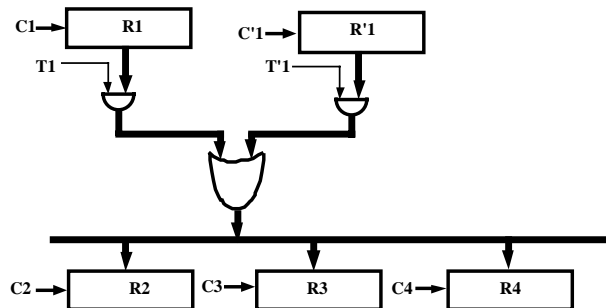
Ici 8 tops d'horloge sont nécessaire pour charger le registre et 8 tops sont aussi nécessaire pour lire les données par décalage. C'est un registre à décalage logique à droite.

## VII. ASSOCIATION DE REGISTRES

les registres peuvent être reliés par des bus :



pour transférer le contenu de R1 dans R3 il suffit d'activer C3. Le chargement en parallèle de plusieurs registres est possible, par exemple en activant C2 et C3 simultanément. Le problème est ici simple car il n'y a qu'une seule source d'information. Ajouter des registres en destination ne pose pas de problèmes majeurs. Par contre si l'on rajoute un registre en source d'information, de la logique supplémentaire est nécessaire :



Pour transférer le contenu de R1 dans R3 il faut :

- Activer le signal T1 de commande des portes ET.
- Laisser l'information se propager et se stabiliser sur le bus de communication.
- Activer le signal de chargement C3.
- Désactiver le signal T1 afin de libérer le bus.

T1 et T'1 ne doivent jamais être actifs simultanément sinon la fonction de transfert n'est pas correctement assurée.

### Remarque :

Le registre et ses portes de commande associées constituent un bloc modulaire. L'adjonction d'un registre supplémentaire revient à connecter au bus un bloc registre + portes de commande. Malheureusement il aurait fallu avoir prévu 8 entrées supplémentaires sur la porte OU : l'expansion (l'adjonction) ne semble donc pas simple.

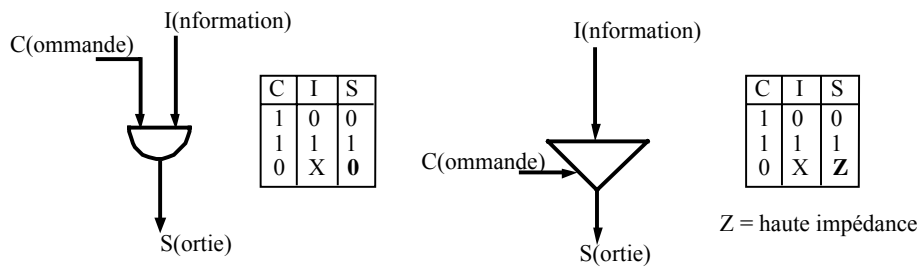
### \* Solution : logique 3 états

La technologie pallie à cet inconvénient en permettant de réaliser un **OU câblé (wired OR)**, à condition que la résistance (l'impédance) de sortie de la porte de commande soit très élevée lorsque cette dernière n'est pas commandée : dans ce cas elle peut être considérée comme retirée électriquement du bus.

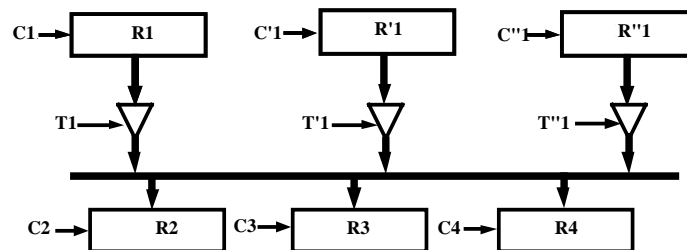
Cette porte spéciale ne se comporte plus tout à fait comme un ET logique, mais plutôt comme une barrière :

- ouverte elle présente en sortie l'état de son entrée : 0 ou 1,
- fermée elle présente en sortie un état haute impédance.

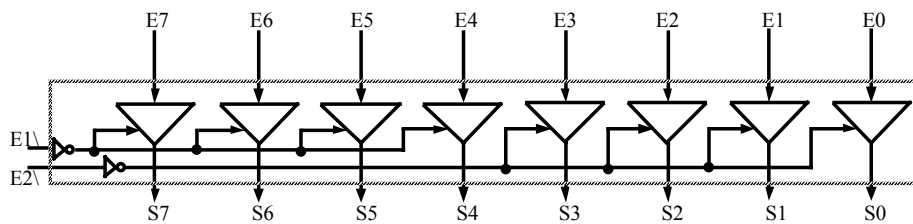
D'où son nom de barrière "trois états" (tri-state).



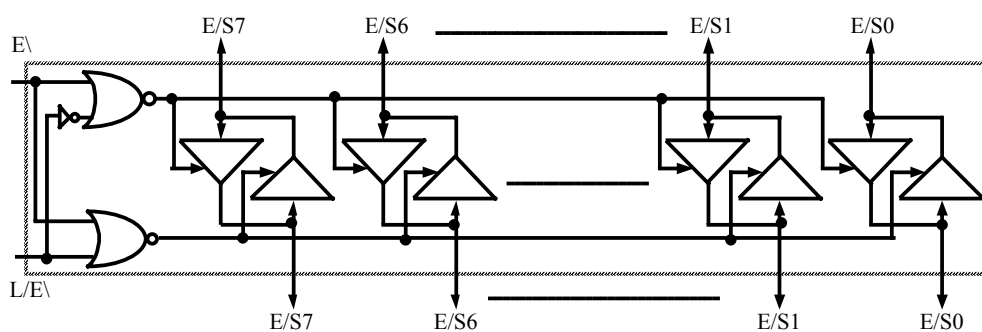
Ainsi il est possible de considérer un bus comme une voie de communication entre registres, ces derniers pouvant être en nombre quelconque. Un seul bus ne permet qu'un transfert à la fois. En rebouclant le bus sur les entrées des registres R1 R'1 R''1 et en connectant les sorties des registres R2 R3 R4 sur ce bus tous les transferts deviennent possibles, d'une seule source vers une ou plusieurs destinations.



Il existe des barrières 3 états unidirectionnelles et bidirectionnelles :



Barrière 3 états unidirectionnelles



Barrière 3 états bidirectionnelles

## VIII. LES COMPTEURS

### VIII. 1. Fonction des compteurs

Le terme général de compteur englobe tous les montages constitués par association de bascules synchrones et qui permettent de numérotter les impulsions qui leur sont appliquées, selon un code binaire lu sur les sorties Q des bascules.

On pourra ainsi repérer, par exemple, la 3ème et la 7ème impulsion d'un train de 10 impulsions successives ; ou encore identifier et extraire une impulsion toutes les N impulsions. Dans ce dernier cas on lui donne plutôt le nom de "diviseur par N".

### VIII. 2. Code de comptage

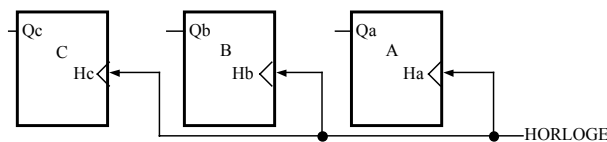
Après chaque impulsion les valeurs logiques des sorties Q des bascules, constituent une des combinaisons ou un des ETATS du compteur. La suite de ces états, **tous différents**, constitue le "code de comptage" qui permet la numérotation des impulsions.

En général, il se répète cycliquement toutes les N impulsions et comprend alors N états différents, apparaissant toujours dans le même ordre. On dit qu'il s'agit d'un compteur à N états (dans le cas particulier d'un compteur à 10 états, on lui donne souvent le nom de "décade". Le code comptage peut être absolument quelconque.

### VIII. 3. Compteurs synchrones ou asynchrones

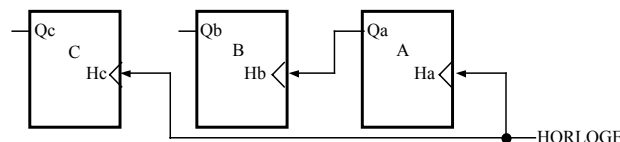
- Un compteur est dit synchrone si toutes ses bascules sont commandées simultanément par les impulsions à compter.

Exemple :



- Un compteur est dit asynchrone si ou plusieurs de ses bascules sont commandées par des transitions (0 --> 1 ou 1--> 0 selon le type synchronisation de la bascule) apparaissant sur les sorties d'autres bascules.

Exemple :



Dans la suite nous ne considérons que des compteurs synchrones.

### VIII. 4. Méthode de synthèse des compteurs synchrones

Sachant que les impulsions à compter sont appliquées sur l'entrée d'horloge de chaque bascule, la synthèse d'un compteur synchrone consiste à déterminer les valeurs logiques à donner aux entrées de chaque bascule, c'est à dire trouver les équations des entrées. Ces équations dépendent :

- de la nature et du nombre d'états du code comptage que l'on désire obtenir,
- du type de bascule choisi : RS, D ou JK. Le plus souvent toutes les bascules d'un compteur sont identiques.

Un compteur est constitué d'autant de bascules qu'il y a de bit dans chaque état du comptage.

#### Les différentes étapes de synthèse d'un compteur :

1. Mise en équation des entrées : à l'aide de la table de transition du type de bascule choisi, on dresse le tableau de toutes les valeurs logiques que doivent prendre les entrées de chaque bascule

pour donner tous les changements d'états imposés par le code de comptage. Sans oublier l'enchaînement du dernier avec le premier si le code doit se répéter cycliquement. Si l'on souhaite au contraire que le compteur se bloque dans le dernier état, il faut prévoir l'enchaînement de ce dernier avec lui même.

2. Simplification par la méthode de Karnaugh en donnant l'expression minimale de chaque équation d'entrée, à partir des états présents des sorties Q des bascules.
3. Enfin réalisation du circuit.

**Exemple 1 :** synthèse d'un compteur synchrone 3 bits avec des bascules JK.

Le mode de comptage est le binaire naturel. Le comptage est cyclique.

### Table de transitions :

Etats présents (EP)			Etats suivants (ES)			Entrées					
Qc	Qb	Qa	Qc	Qb	Qa	Jc	Kc	Jb	Kb	Ja	Ka
0	0	0	0	0	1	0	*	0	*	1	*
0	0	1	0	1	0	0	*	1	*	*	1
0	1	0	0	1	1	0	*	*	0	1	*
0	1	1	1	0	0	1	*	*	1	*	1
1	0	0	1	0	1	*	0	0	*	1	*
1	0	1	1	1	0	*	0	1	*	*	1
1	1	0	1	1	1	*	0	*	0	1	*
1	1	1	0	0	0	*	1	*	1	*	1

### Equations :

- En choisissant pour Ja et Ka la valeur indifférente (\*) à 1, on aura  $Ja = Ka = 1$

- Jb

QbQa	00	01	11	10
Qc				
0	0	1	*	*
1	0	1	*	*

- Kb

QbQa	00	01	11	10
Qc				
0	*	*	1	0
1	*	*	1	0

$$Jb = Kb = Qa$$

- Jc

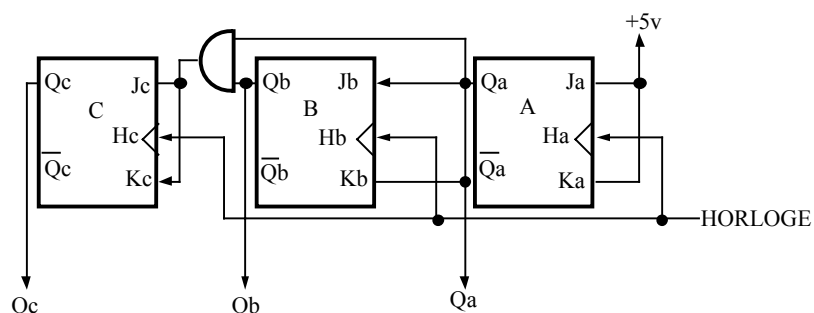
QbQa	00	01	11	10
Qc				
0	0	0	1	0
1	*	*	*	*

- Kc

QbQa	00	01	11	10
Qc				
0	*	*	*	*
1	0	0	1	0

$$Jc = Kc = Qb.Qa$$

### Schéma logique :



**Exemple 2 :** synthèse d'un compteur synchrone 3 bits avec des bascules D à déclenchement sur front. Le mode de comptage est le binaire naturel. Le comptage est cyclique.

**Table de transitions :**

Etats présents (EP)			Etats suivants (ES)			Entrées		
Qc	Qb	Qa	Qc	Qb	Qa	Dc	Db	Da
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

**Equations :**

- Da

$$Da = Qa \backslash$$

-Db

QbQa	00	01	11	10
Qc				
0	0	1	0	1
1	0	1	0	1

- Dc

QbQa	00	01	11	10
Qc				
0	0	0	1	0
1	1	1	0	1

$$Db = Qb \backslash . Qa + Qb . Qa \backslash = Qb \approx Qa$$

$$\begin{aligned} Dc &= Qc \backslash . Qb . Qa + Qc . Qb \backslash + Qc . Qa \backslash \\ &= Qc \backslash . Qb . Qa + Qc (Qb \backslash + Qa \backslash) \\ &= Qc \backslash . Qb . Qa + Qc . (Qb . Qa) \backslash \\ &= Qc \approx (Qb . Qa) \end{aligned}$$

**Schéma logique :**

