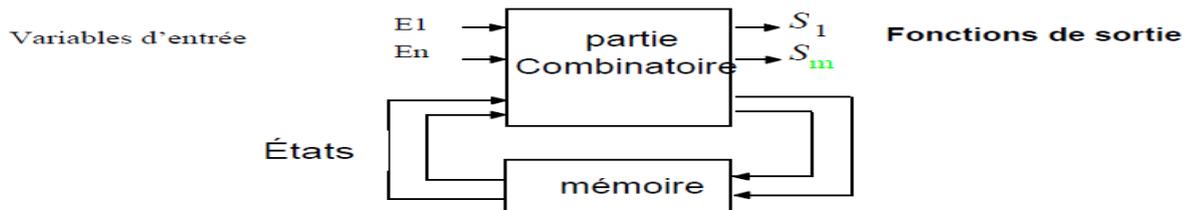


Chapitre 2 : Circuits Séquentiels

1. Définition

Un circuit est dit séquentiel si le comportement des sorties dépend des valeurs assignées aux variables d'entrée et selon son histoire.

De tels circuits contiennent une mémoire à côté d'une partie combinatoire. Cette mémoire a pour rôle de conserver l'histoire du circuit



RQ :

- Dans les circuits combinatoires, les signaux de sortie ne dépendent que des signaux d'entrée présents au même instant.
- Dans les circuits séquentiels, il y a de la rétroaction : les signaux de sortie ne dépendent pas uniquement des entrées, mais aussi de leur séquence. Le circuit se rappelle des entrées et des états antérieurs : il a une mémoire du passé.
- L'étude des circuits combinatoires repose sur l'algèbre de Boole. Celle des circuits séquentiels repose sur la théorie des automates finis.

- Les systèmes séquentiels sont classés en 2 catégories :

Circuits séquentiels asynchrones : les sorties changent d'états dès qu'il y a des changements des états des entrées.

Circuits séquentiels synchrones : les sorties changent d'états après avoir eu une autorisation d'un signal de synchronisation appelé souvent signal « Horloge » noté H ou CLK.

2. Circuits séquentiels de base : bascules

2.1. Définition

Une bascule est un circuit séquentiel élémentaire, capable de mémoriser une variable d'état et donc qui peut prendre 2 états (état 0 et état 1). Il existe différents types de bascules, qui peuvent être classés par :

- leurs entrées de valeur,
- l'interprétation de leur(s) entrée(s) d'activation,

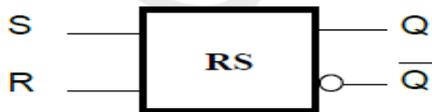
- les entrées d'initialisation dont elles disposent.

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

2.2. Les bascules asynchrones

2.2.1. Bascule RS

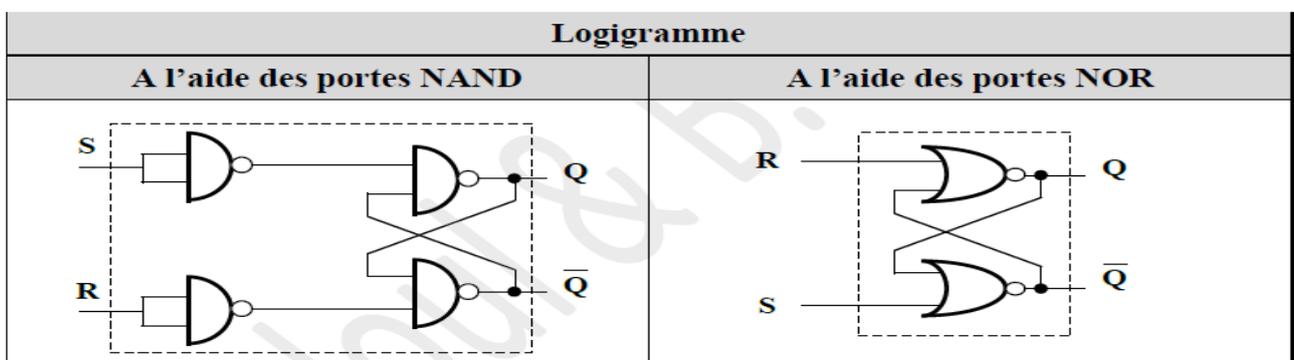
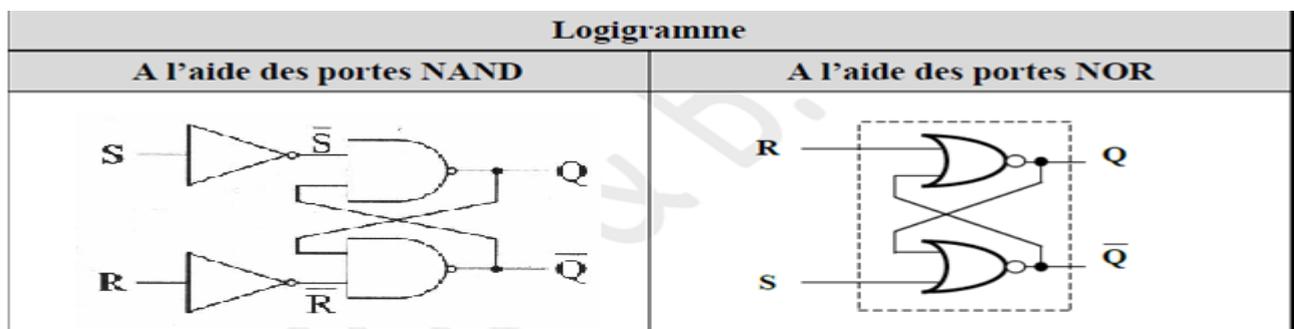
Symbole



Une impulsion sur **S** (set) → Mise à 1 de **Q** (marche)

Une impulsion sur **R** (Reset) → Mise à 0 de **Q** (Arrêt)

Table de vérité						Equation des sorties
Entrées		Sorties			Mode de fonctionnement	
R	S	Q_n	Q_{n+1}	Q_{n+1}		
0	0	0	0	1	Etat précédent	
0	0	1	1	0	Etat précédent	
0	1	0	1	0	Enclenchement	
0	1	1	1	0	Maintien à 1	
1	0	0	0	1	Maintien a 0	
1	0	1	0	1	Déclenchement	
1	1	0	-	-	Interdit	
1	1	1	-	-	Interdit	



NB : L'état $R=S=1$ est un état interdit puisqu'il nous donne le deux sorties complémentaires Q et Q' au même état ce qui n'est pas logique.

2.2.2. Bascule D

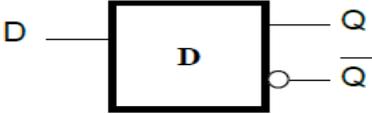
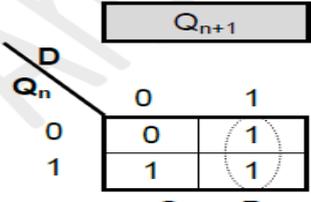
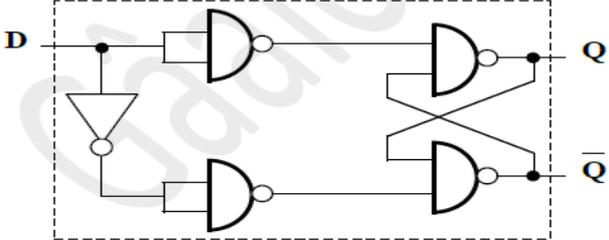
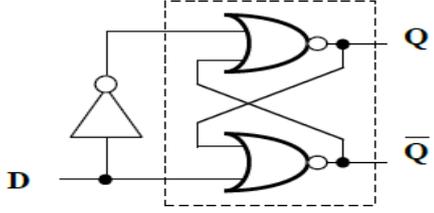
Symbole	Explication
	<p>Un appui sur D → Mise à 1 de Q Un relâchement de D → Mise à 0 de Q</p>

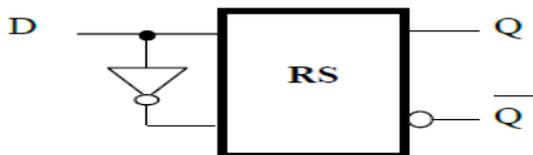
Table de vérité				Equation des sorties
Entrées		Sorties		Mode de fonctionnement
D	Q _n	Q _{n+1}	Q _{n+1}	
0	0	0	1	Maintien à 0 : μ ₀ Déclenchement : δ Enclenchement : ε Maintien à 1 : μ ₁
0	1	0	1	
1	0	1	0	
1	1	1	0	

Equation des sorties	
	$Q_{n+1} = D$

Logigramme	
A l'aide des portes NAND	A l'aide des portes NOR
	

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

Ainsi on obtient une bascule **D** en rajoutant un inverseur entre **S** et **R**.



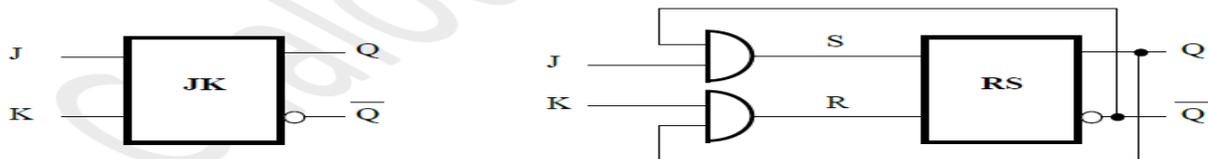
2.2.3. Bascule JK

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

Table de vérité					Equation des sorties
Entrées			Sorties		Mode de fonctionnement
J	K	Q_n	Q_{n+1}	Q_{n+1}	
0	0	0	0	1	Etat précédent
0	0	1	1	0	Etat précédent
0	1	0	0	1	Maintien à 0 : μ_0
0	1	1	0	1	Déclenchement : δ
1	0	0	1	0	Enclenchement : ε
1	0	1	1	0	Maintien à 0 : μ_1
1	1	0	1	0	Enclenchement : ε
1	1	1	0	1	Déclenchement : δ

Equation des sorties				
Q_{n+1}				
JK				
Q_n				
	00	01	11	10
0	0	0	1	1
1	1	0	0	1

$Q_{n+1} = J\bar{Q}_n + KQ_n$



2.2.4. Bascule T

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

Table de vérité					Equation des sorties
Entrées		Sorties		Mode de fonctionnement	
T	Q_n	Q_{n+1}	Q_{n+1}		
0	0	0	1	Maintien à 0 : μ_0	
0	1	1	0	Maintien à 1 : μ_1	
1	0	1	0	Enclenchement : ε	
1	1	0	1	Déclenchement : δ	

Equation des sorties				
Q_{n+1}				
T				
Q_n				
	0	1		
0	0	1		
1	1	0		

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

Remarque : En remplaçant J et K par T dans l'équation de la bascule JK on aura

$$Q_{n+1} = TQ_n + \bar{T}\bar{Q}_n = T \oplus \bar{Q}_n.$$



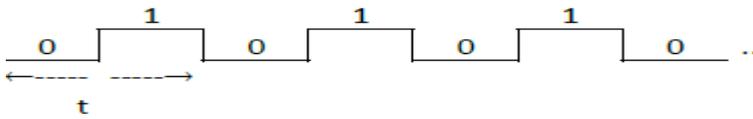
2.3. Les bascules synchrones :

On distingue deux catégories principales de bascules :

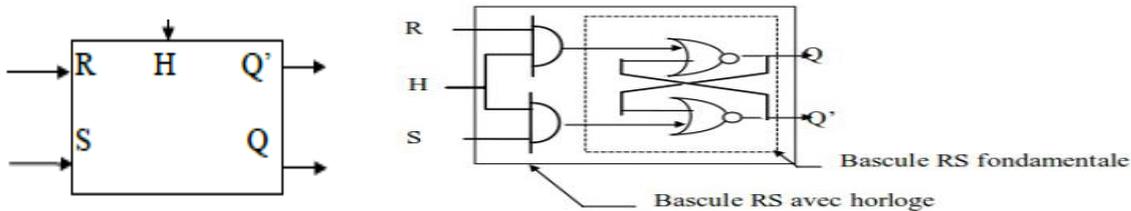
- les *basculés asynchrones* que l'on nomme *verrous* (ou *latch* en anglais)
- et les *basculés synchrones* (dépendant d'un signal d'horloge) que l'on nomme simplement *basculés* (ou *flip-flop* en anglais).

Une bascule est synchrone quand ses sorties ne changent d'état que si un signal supplémentaire est appliquée sur une entrée, dite entrée **d'horloge** (notée **H** ou **CLK**).

Un signal d'horloge est un signal basculant d'un état à son inverse périodiquement. La caractéristique fondamentale de ce signal est sa fréquence ou sa période. Schématiquement le signal d'horloge est représenté comme suit :



Par exemple, on représente la bascule RSH comme suit:



il suffit de faire un ET logique entre ses entrées et le signal d'horloge, pour imposer à ce que la prise en considération de ses entrées soit rythmée par le signal d'horloge.

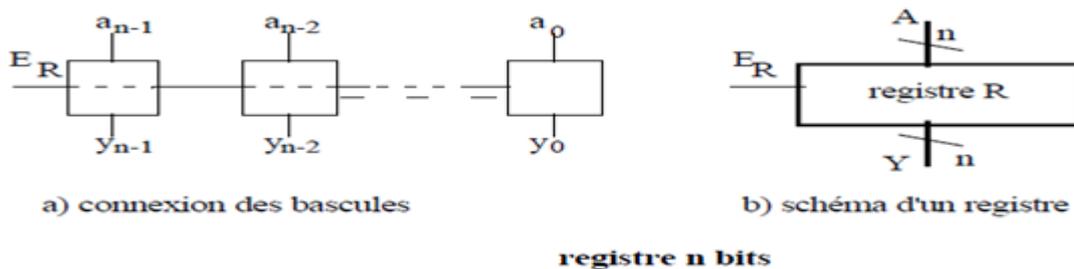
On distingue plusieurs type de composants séquentiels construits à partir de bascules RS, D, T, JK, etc. Parmi eux on peut citer: les registres, les compteurs et les mémoires

3. Les Registres

Les registres sont composés d'un ensemble de bascules mémorisant n bits d'information. On les utilisent donc pour stocker un nombre limité de bits (8, 16, 32 et 64 bits sont souvent utilisés).

3.1. Registre N Bits

Pour mémoriser un ensemble de n bits $a_{n-1}, a_{n-2}, \dots, a_0$, ou mot de n bits, on utilise n bascules qui sont activées en même temps, donc par le même signal



Le type de bascules utilisé est choisi d'après l'utilisation du registre. On utilise souvent des bascules à deux entrées d'activation (**horloge** et **autorisation**) : l'horloge du système est connectée à l'entrée d'horloge des bascules, le signal E_R est connecté à l'entrée d'autorisation.

La classification des modes de fonctionnement est la suivante :

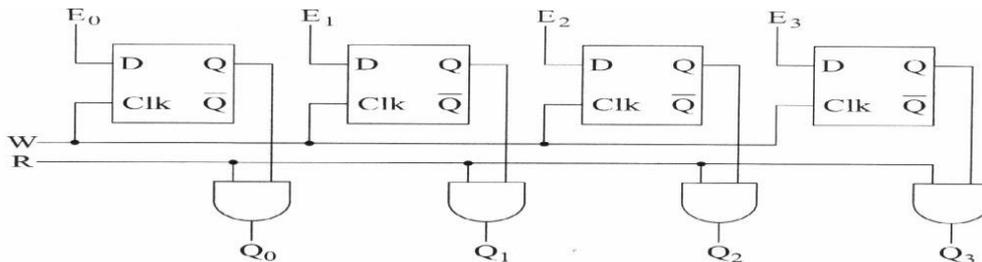
- Des registres à entrées parallèles et sorties parallèles : **PIPO** (*Parallel IN-Parallel OUT*).
- Des registres à entrées parallèles et sorties séries : **PISO** (*Parallel IN-Serial OUT*).

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

3.2. Registre De Mémoire (Registre parallèle)

➤ association de n bascules D pour mémoriser n bits

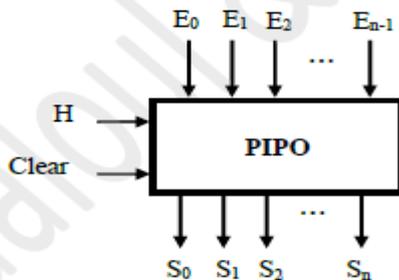
Exemple : **Registre de mémorisation 4 bits**



Les 4 bascules sont chargées en parallèle et lues en parallèle en synchronisme avec le signal d'écriture W. Ce type de registre est appelé aussi registre **PIPO**.

- les entrées présentés sur E0, E1, E2, E3 sont mémorisées en synchronisation avec le signal W
- elles peuvent être lues sur les sorties Q0, Q1, Q2, Q3 en coïncidence avec le signal de validation R

Schéma fonctionnel d'un registre PIPO.



3.3. Registre A Décalage

Ce type de registre est principalement utilisé comme mémoire d'information dynamique ; 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

le registre de décalage a 2 types d'entrées(2 fonctionnalités)

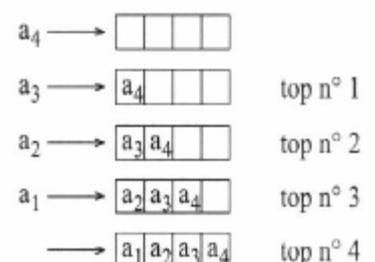
✓ chargement parallèle, comme dans un registre de mémorisation (decal = 0)

✓ décalage = chargement série (decal = 1) (SISO)

✗ l'information est présentée séquentiellement

bit après bit à la 1ère bascule

✗ à chaque coup d'horloge, un nouveau bit est

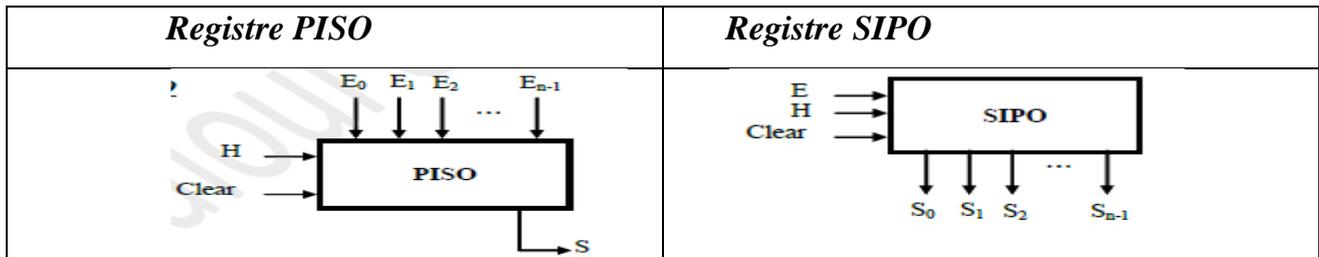


présenté et ceux déjà chargés sont décalés d'un rang

En entrée de chaque bascule, un multiplexeur commandé par le signal "decal" permet de sélectionner soit la valeur du bit de gauche, soit la valeur d'entrée. Dans tous les cas, il faut activer le signal E_R pour charger le registre.

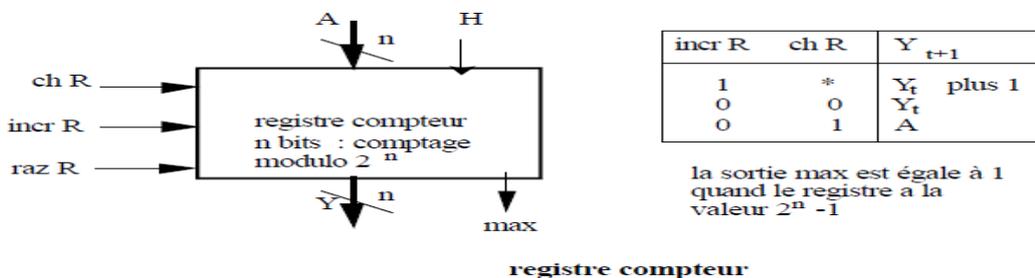
3.4. Registre Mixte

On peut trouver des registres mixtes, donc on peut écrire en parallèle et lire en série (**PISO**), ou vice versa (**SIPO**), ou qui offrent les deux possibilités « au choix ».



3.5. Les Registres Compteurs (les Compteurs)

Si on veut pouvoir incrémenter et / ou décrémenter l'information contenue dans un registre facilement, il est possible de le connecter à un circuit d'incrément et/ou de décrémentation pour obtenir un registre compteur/ décompteur. La figure suivante donne le schéma d'un registre compteur à entrée parallèle, sortie parallèle.



- Ce compteur modulo 2^n est activé à chaque période de l'horloge H.
- modulo 2^n (Le nombre total N des combinaisons successives est appelé le modulo du compteur)
- Il peut être forcé à 0 par la commande raz R (asynchrone), ou à la valeur d'entrée A par la commande de chargement ch R (prise en compte à l'activation par H).
- La commande d'incrément incr R est prioritaire sur la commande de chargement.
- La sortie max est à 1 quand le compteur atteint sa valeur maximale $2^n - 1$.