

Chapitre 2

Logique séquentielle

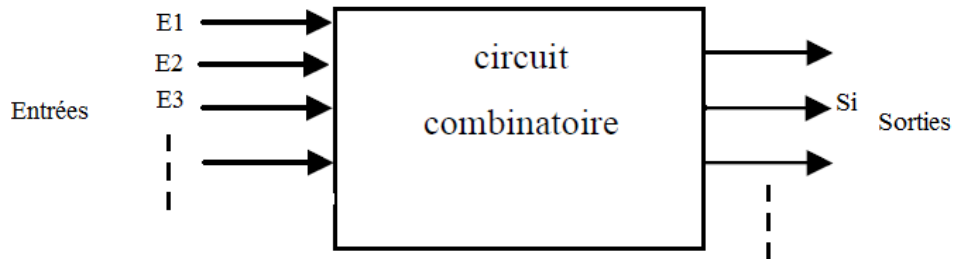
Partie I : Bascules RS, RST, D et registres

2.1. Introduction aux circuits séquentiels

Dans les circuits combinatoires une sortie S_i est directement liée aux entrées à chaque instant.

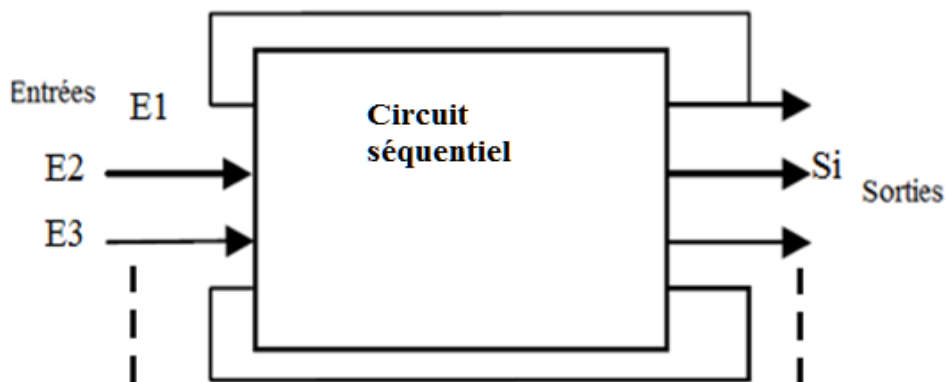
$$S_i = F(E_1, E_2, \dots)$$

à chaque combinaison d'entrée correspond un seul état de sortie.



Ainsi, la sortie ne dépend pas des états antérieurs des entrées. Par conséquent, le temps n'intervient pas dans le fonctionnement de ces circuits.

Dans les circuits séquentiels la sortie S_i dépend des entrées et des états antérieurs des sorties (leurs passé). Ainsi, le temps constitue un des paramètres essentiels.



$$S_i^t = F(E_1, E_2, \dots, S_i^{t-1})$$

Exemple :

Etat n°1 : $E1 = 0$ et $E2 = 0 \Rightarrow Si = 0$

Etat n°2 : $E1 = 1$ et $E2 = 0 \Rightarrow Si = 1$

Etat n°3 : $E1 = 0$ et $E2 = 0 \Rightarrow Si = 1$

Etat n°4 : $E1 = 0$ et $E2 = 1 \Rightarrow Si = 0$

Etat n°5 : $E1 = 0$ et $E2 = 0 \Rightarrow Si = 0$

Ainsi, un système logique séquentiel ne fournit pas la même sortie pour une même configuration d'entrées (état 1 et état 3). Sa fonctionnalité dépend donc de l'ordre des opérations (le déroulement des séquences).

Les circuits séquentiels fondamentaux:

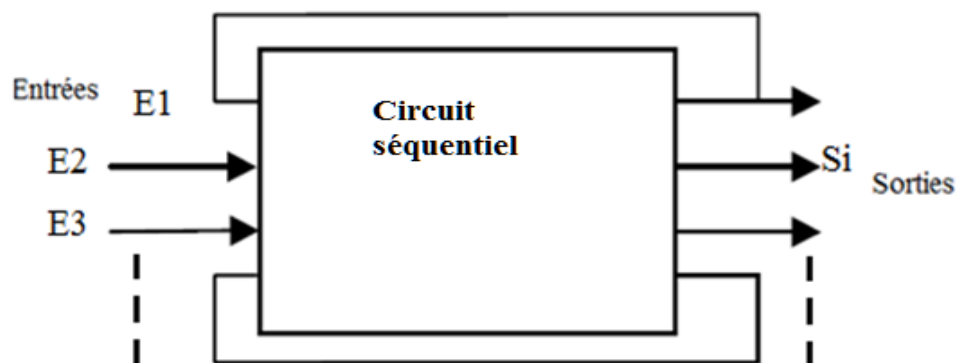
- Bascule,
- Registre,
- Compteur.

Les applications de base dans les systèmes informatiques sont :

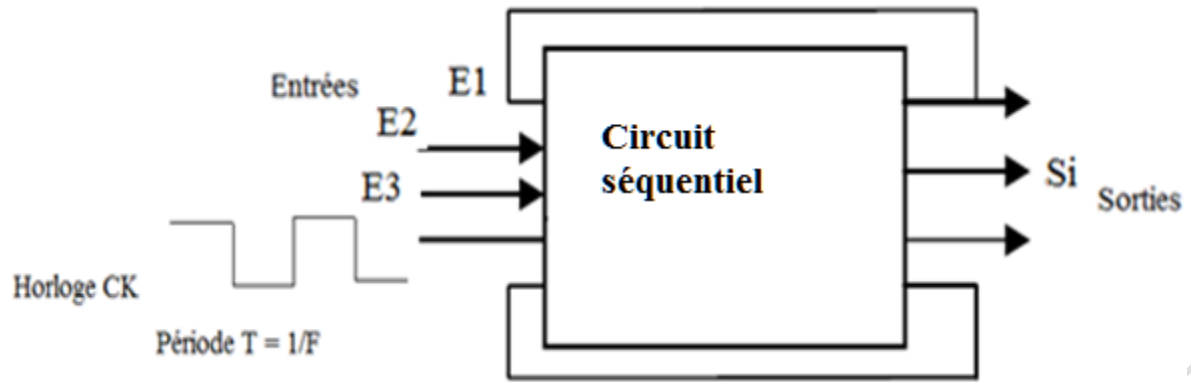
- Mémoire vive statique SRAM (Static Random Access Memory),
- Registres internes des microprocesseurs et microcontrôleurs

Différents types de circuit séquentiel

- Circuit séquentiel asynchrone: À tout moment Les sorties du montage peuvent changer dès qu'une ou plusieurs entrées changent après un temps de propagation (retard ou temps de réponse). Ils sont très difficiles à gérer.



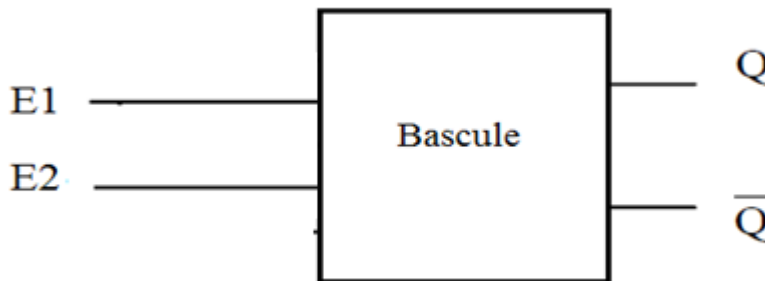
- Circuit séquentiel synchrone: Le changement des états de sorties est commandé, en plus des entrées, par un signal d'horloge CK. Il constitue une base de temps et représente un moyen de synchronisation des différentes opérations. Les changements d'état des sorties s'effectuent soit par **niveau logique** 1 ou 0 ou par une transition appelée « **front actif** » (montant ou descendant) du signal d'horloge. Ainsi, les entrées et les états antérieurs des sorties préparent les futurs changements d'états des sorties mais ne peuvent provoquer ces changements sans une synchronisation avec le signal d'horloge.



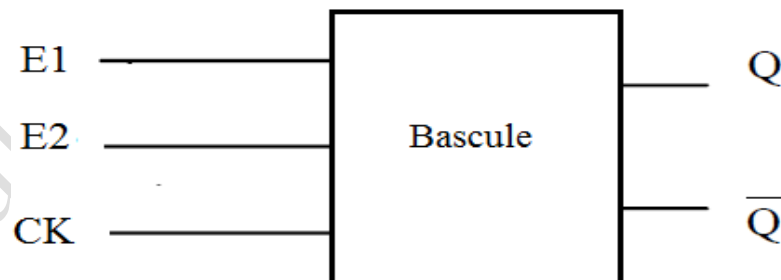
2.2. Bascules

Une bascule est un circuit logique séquentiel, appelé également éléments bistables, latch ou flip flop, qui possède deux entrées et deux sorties complémentaires Q et \bar{Q} .

Bascule asynchrone :

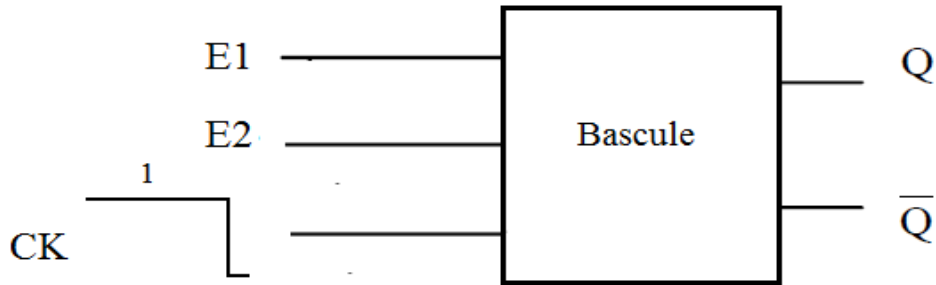


Bascule synchrone :

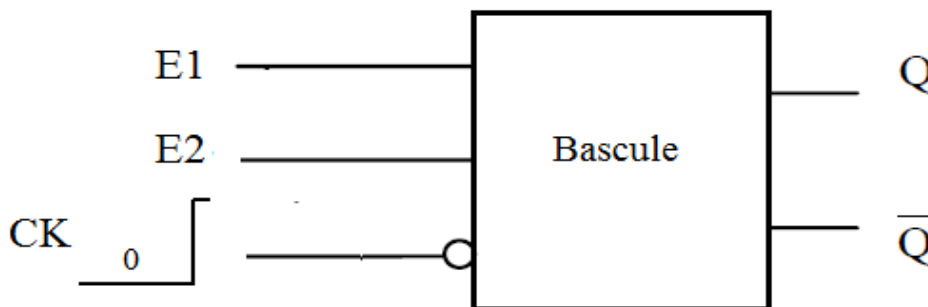


Différents types de bascules synchrones :

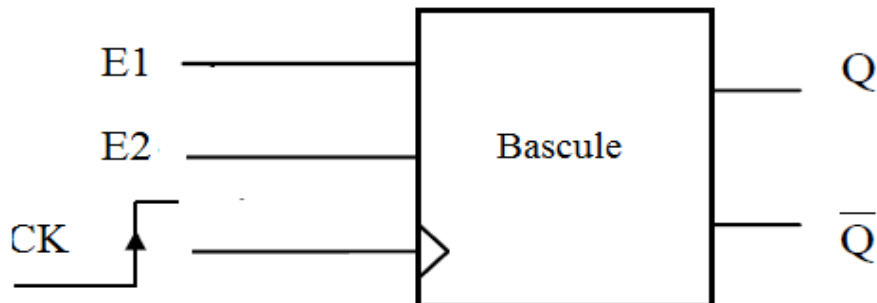
- a. Bascule commandée par niveau haut du signal d'horloge (1 logique)



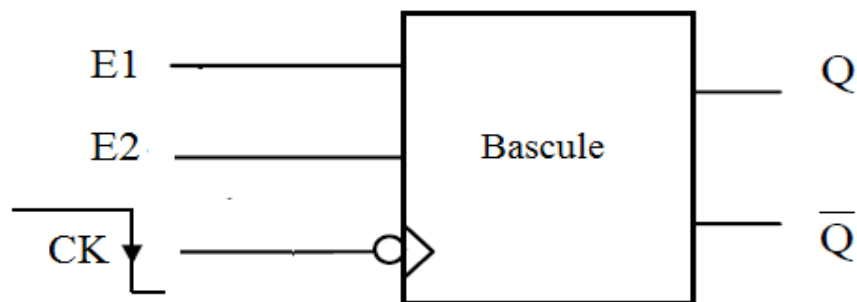
b. Bascule commandée par niveau bas du signal d'horloge (0 logique)



c. Bascule commandée par niveau front montant du signal d'horloge



d. Bascule commandée par niveau front descendant du signal d'horloge



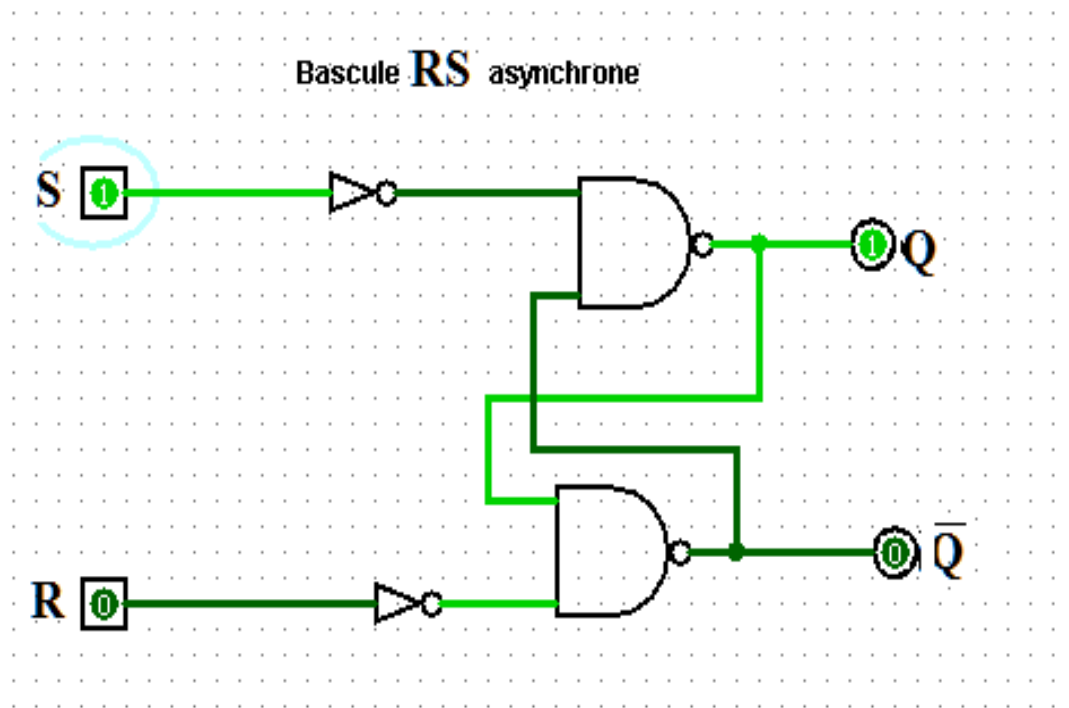
Il est à noter que la synthèse et la conception des circuits séquentiels sont effectuées par **la méthode de Huffman** (non traitée dans ce cours).

2.3. Bascule de type RS

a. Bascule RS asynchrone

La bascule RS est la base de toutes bascules standards (D, JK, ...). La sortie Q de la bascule est mise à 0 lorsque l'entrée RESET ($R = 1$) est activée et mise à 1 lorsque l'entrée SET ($S = 1$) est activée.

Circuit à base de portes NAND



Symbole

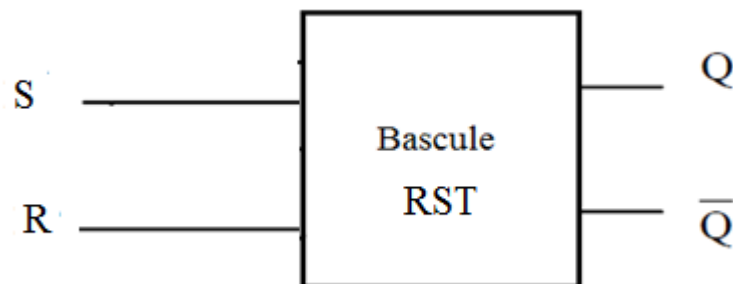


Table de vérité d'une bascule RS asynchrone

Sorties à t_n		Entrées		Sorties à t_{n+1}	
Q	\overline{Q}	S	R	Q	\overline{Q}
1	0	0	0	1	0
1	0	0	1	0	1 Reset
1	0	1	0	1	0
1	0	1	1	1 ind !!!	1 ind !!!
0	1	0	0	0	1
0	1	0	1	0	1
0	1	1	0	1	0 Set
0	1	1	1	1 ind !!!	1 ind !!!

Ind : indéterminé

On remarque que pour des états d'entrées identiques, les sorties sont différentes ($R=S=0 \Rightarrow Q=1$ ou $Q=0$ en fonction de l'état de sortie précédent).

Ainsi, les circuits séquentiels dépendent des états d'entrées et des états antérieurs des sorties.

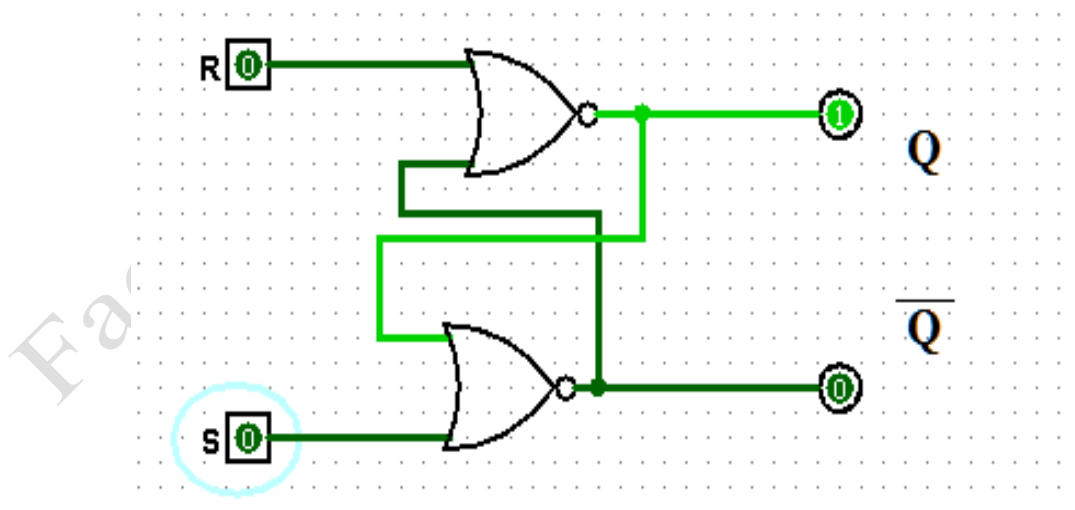
Table de vérité condensée

S	R	Q
0	0	S.C
0	1	Etat Reset
1	0	Etat Set
1	1	indéterminé

SC: Sans Changement, état de mémorisation

b. Bascule RS asynchrone à base des portes NOR

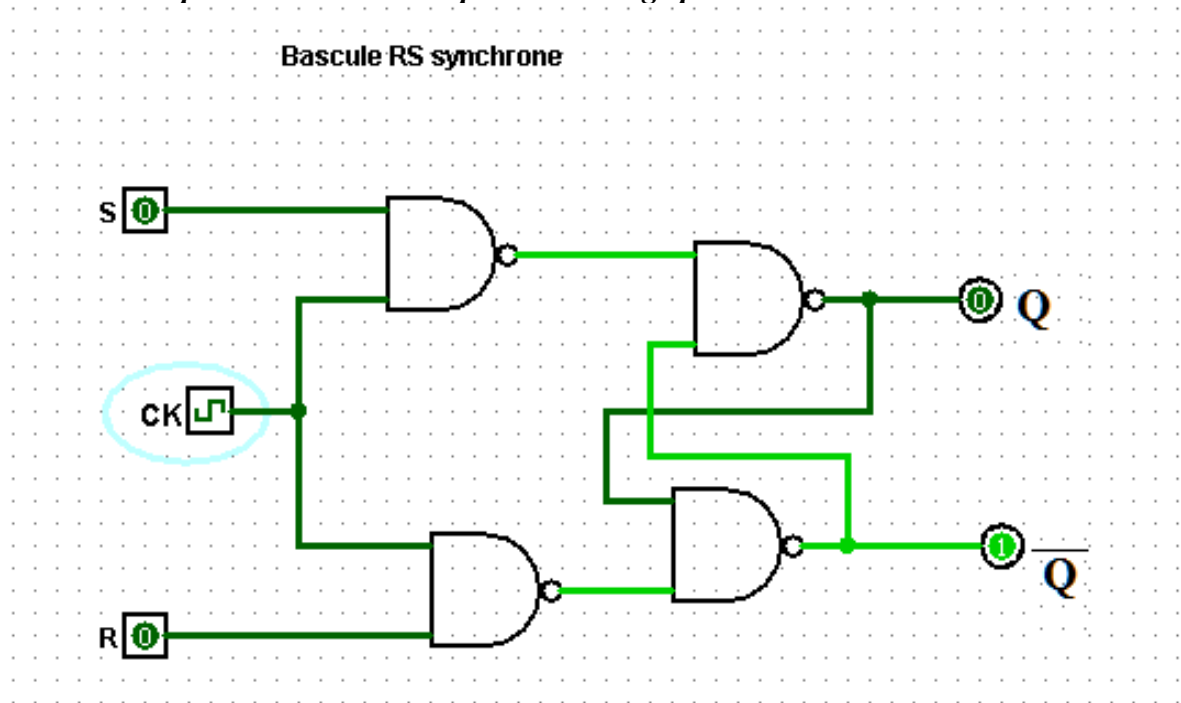
Même raisonnement que la bascule à base des portes NAND.



c. Bascule RS Synchronne

La bascule synchrone est pilotée par une horloge CK. Ce qui permet de changer d'état de la bascule à des moments précis. Elle est appelée **RST** ou **RSH**.

Circuit à base de portes NAND activé par une horloge par niveau haut



- Quand CK passe au niveau haut la bascule se trouvera sous le contrôle des entrées R et S.
- Lorsque l'entrée d'horloge CK est au niveau 0, les portes C et D sont à l'état 1 indépendamment des entrées R et S. La bascule se maintient ainsi dans un état pris antérieurement.

Table de vérité d'une bascule RS synchrone

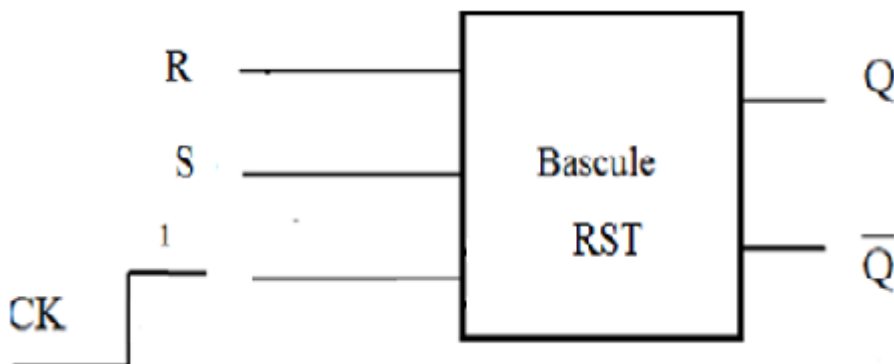
S	R	CK	Q	\bar{Q}
0	0	0	S.C	S.C
0	0	1	S.C	S.C
0	1	0	S.C	S.C
0	1	1	0	1 Etat Reset
1	0	0	S.C	S.C
1	0	1	1	0 Etat Set
1	1	0	S.C	S.C
1	1	1	Ind	ind

SC : sans changement, état de mémorisation

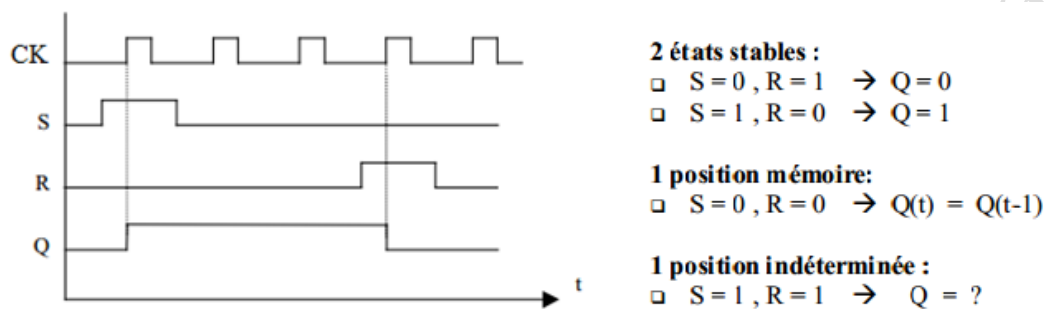
Ind: indéterminé.

L'état indéterminé constitue un inconvénient de la bascule. Depuis cet état, quand R ou S change d'état (de 1 à 0), nous serons dans l'impossibilité de déduire dans quels états vont basculer Q et \bar{Q} .

Symbole

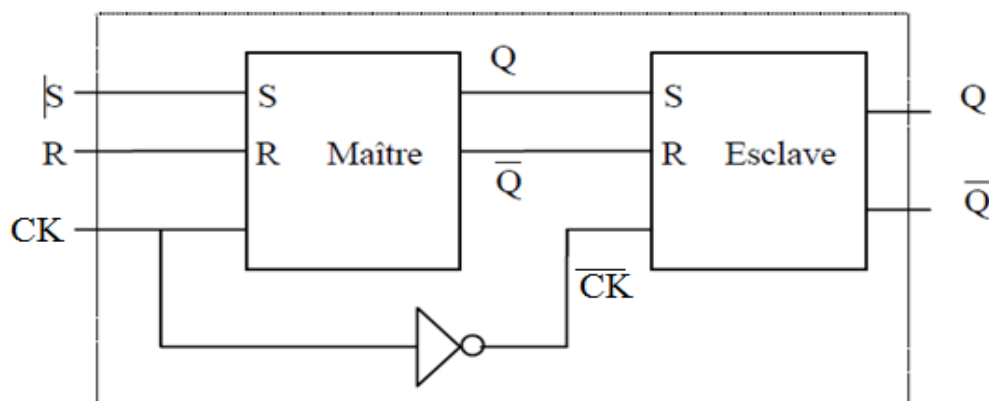


Chronogramme : La bascule RST est sensible au front montant du signal d'horloge



d. Bascules RST maître-esclave

Le circuit RST maître-esclave est composé deux bascules RST, elles montées en cascade et commandées par deux horloges en opposition de phase réalisent la même fonction qu'une seule bascule.



Les bascules maître-esclaves permettent de diminuer la sensibilité aux bruits et aux perturbations externes, en minimisant la durée d'activation de la bascule.

- sur le niveau bas de l'horloge, le maître est dans l'état « sans changement » (fermé), et l'esclave fonctionne normalement RS (ouvert) ;
- sur le niveau haut de l'horloge, le maître fonctionne normalement (ouvert), et l'esclave est dans l'état « sans changement » (fermé).

La période pendant laquelle la bascule est sensible aux bruits et aux perturbations externes se résume donc à la durée de commutation de l'horloge du niveau haut au niveau bas (front descendant).

e. Bascule D (Data)

La bascule de type D est une bascule RST pour laquelle on n'a conservé que les deux combinaisons $RS = (0,1)$ et $RS = (1,0)$. Ce qui permet éliminer l'état indéterminé et d'utiliser des entrées complémentaires. La bascule possède donc une seule entrée, nommée D '(Data).

$$D = S \implies R = \bar{S}$$

Circuit de la bascule D

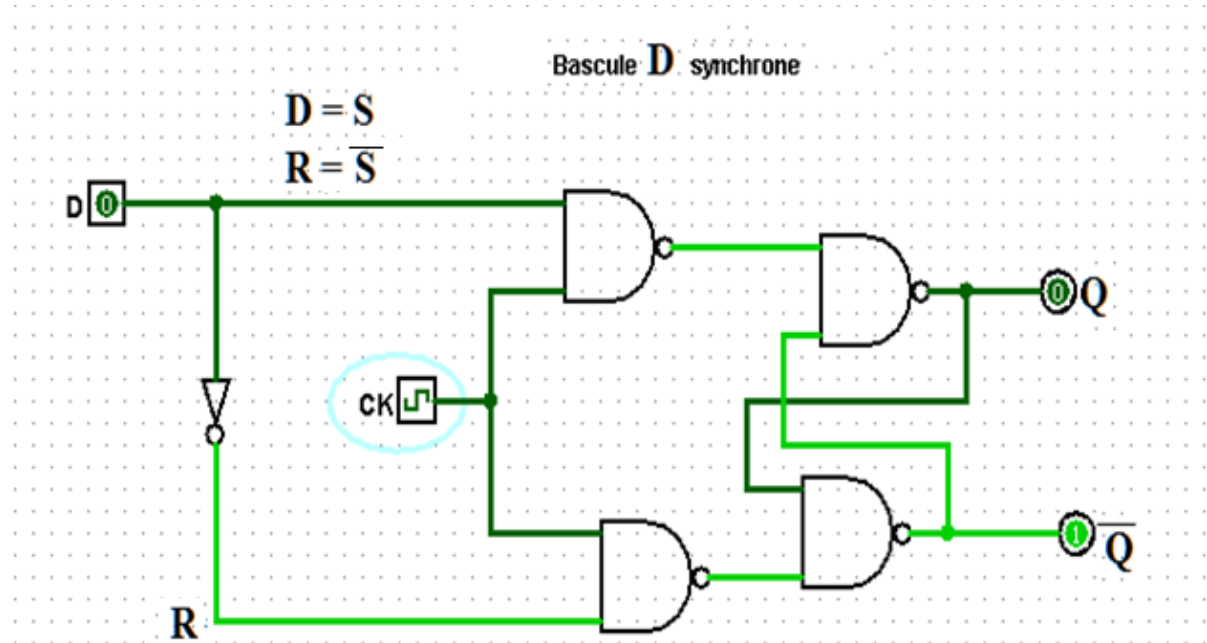
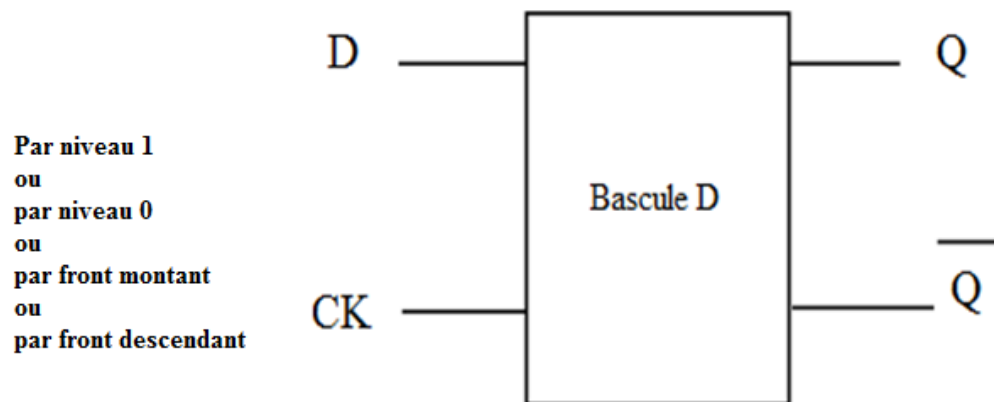


Table de vérité de la bascule D

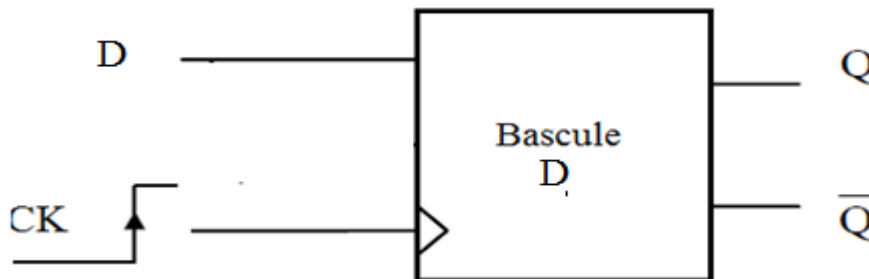
D	CK	Q
0	0	S.C
0	1	0 état reset
1	0	S.C
1	1	1 état set

Symbole

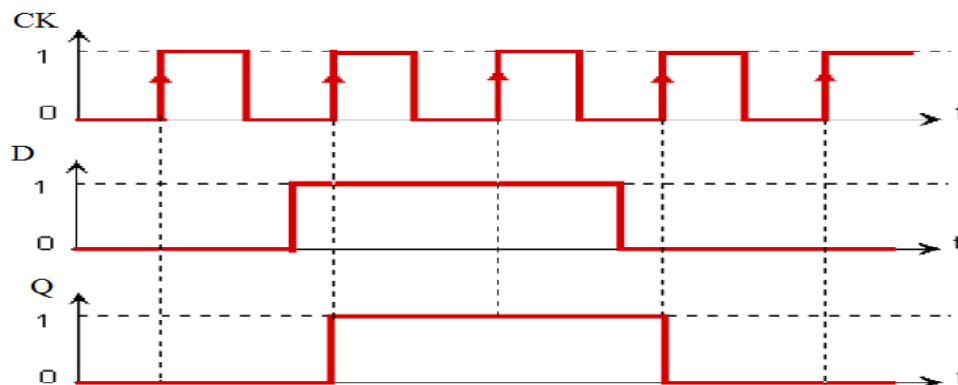


La bascule D est appelée à verrouillage : détection par niveau de CK (appelée latch), dans ce cas l'entrée horloge CK est appelée entrée de validation. Si la bascule D est activée par détection de front de CK (elle est appelée flip flop).

Symbol



Chronogramme de la bascule D à front montant



Ainsi, quand CK passe de 0 \Rightarrow 1 : **C'est la phase d'écriture dans la bascule.**

Si $D=0 \Rightarrow Q=0$

Si $D=1 \Rightarrow Q=1$

Quand CK passe de 1 \Rightarrow 0, la bascule mémorise l'état logique enregistré. C'est la phase de mémorisation dans la bascule. Quelque soit l'état de D, la bascule mémorise l'état logique précédent de la sortie Q. Ceci montre que la bascule D matérialise le stockage d'un bit (0 ou 1). C'est pour cette raison qu'on appelle la bascule D "**un point mémoire**". Pour mémoriser une information

codée sur n bits, on utilise n bascules D. Il est à noter qu'on peut cascader deux bascules D en maître esclave pour réduire la sensibilité aux bruits et aux perturbations.

2.4. Domaine d'application des bascules D

Dans les microprocesseurs, les microcontrôleurs, les interfaces et les mémoires vives statiques, on trouve des entités appelées **Registres dont la conception est fondée sur les bascules**.

Un registre est un circuit constitué de n bascules synchronisées permettant de stocker temporairement d'une information binaire codée sur n bits en vue de son transfert dans un autre circuit (pour traitement, affichage, mémorisation, etc.)

Le schéma d'un tel système comporte autant de bascules (de type D) que de bits binaires à mémoriser. Toutes les bascules sont commandées par le même signal d'horloge.

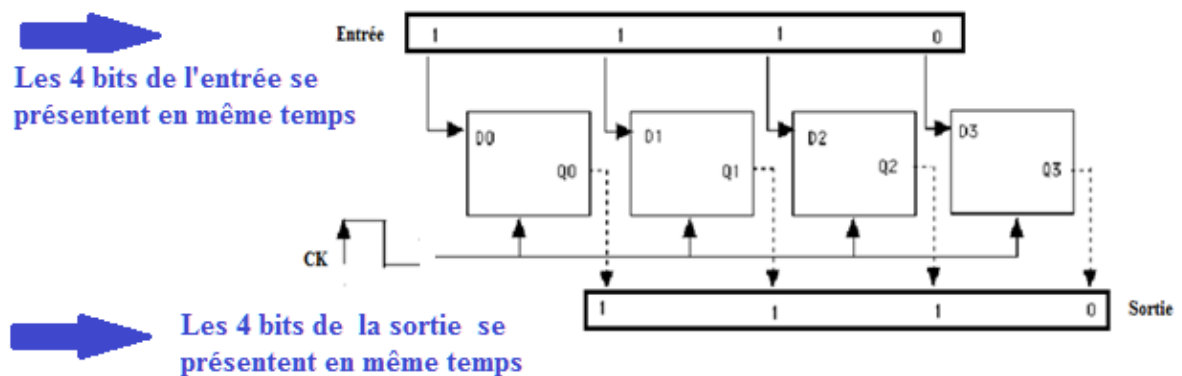
bn-1	bn-2	b1	b0
-------------	-------------	-------	-------	-------	-------	-------	-------	-------	-----------	-----------

Chaque bit est matérialisé par une bascule,

En pratique, il existe plusieurs façons de réaliser un registre. Le mode d'utilisation d'un registre est très varié, il peut être utilisé comme un simple moyen de stockage, ou dans les calculs arithmétiques, ou bien dans les conversions parallèle-série ou série-parallèle des informations (registres à décalage), ...etc.

- **Registre à entrées parallèles/ sorties parallèles : *Registre de stockage***

type parallèle /parallèle



Exemple : Registre de stockage de 4 bits (4 Bascules D à front montant)

A $t = 0$, $D0 = 1$, $D1 = 1$, $D2 = 1$, $D3 = 0 \Rightarrow$ à $t = T$, $Q0 = 1$, $Q1 = 1$, $Q2 = 1$, $Q3 = 0$

Fonctionnement.

Une information se présente aux niveaux des entrées Do...D3, à ce moment le signal d'horloge CK est positionné à 1 \Rightarrow mémorisation de l'information binaire Do...D3.

Quand CK passe à 0 \Rightarrow même si l'information disparaît au niveau des entrées, il restera mémorisé au niveau des bascules D.

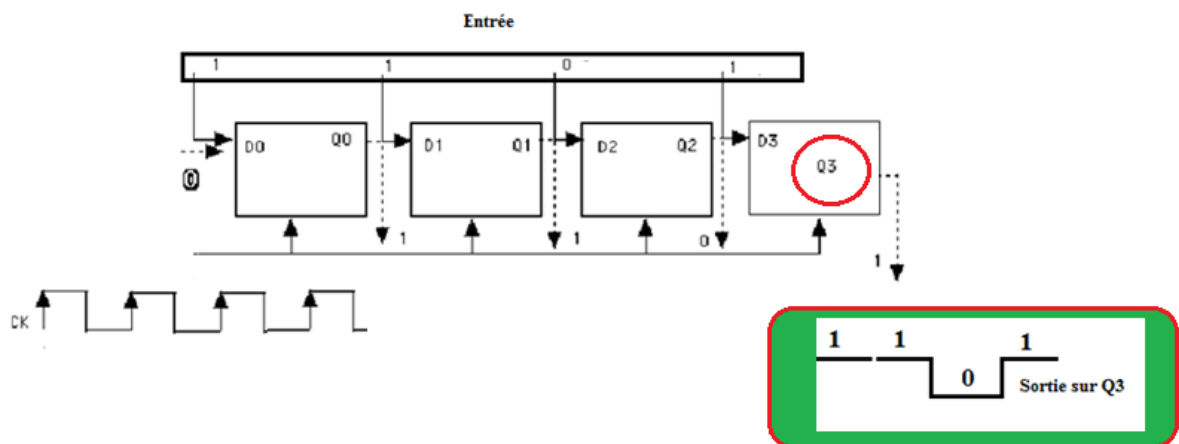
Registre à décalage

Comme son nom l'indique, un registre à décalage consiste à décaler bit par bit une information binaire soit vers la gauche, soit vers la droite. Ce type de registre est représenté par des registres à entrée parallèle/ sortie série ou des registres à entrée série/sortie parallèles

- **Registre à entrées parallèle /sortie série : *Registre à décalage***

Registre de transfert de données, utilisé comme convertisseur parallèle-série, il est nécessaire à l'émission lors d'une transmission série

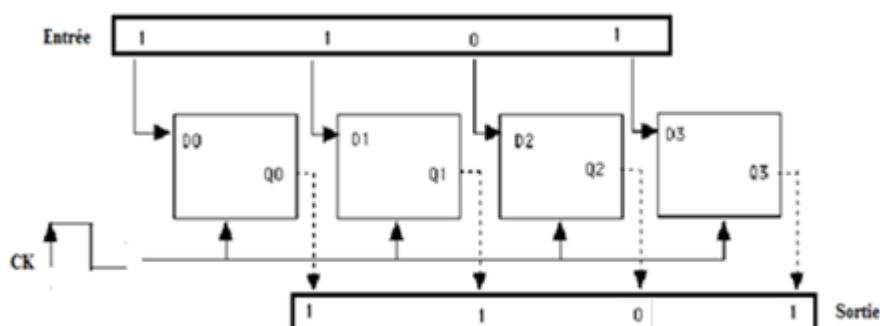
Exemple : Registre à décalage à droite de 4 bits (4 Bascules D à front montant)



Première phase : Ecriture de l'information en parallèle:

A $t = 0 \rightarrow D0 = 1, D1 = 1, D2 = 0, D3 = 1 \Rightarrow$ à $t = T \rightarrow Q0 = 1, Q1 = 1, Q2 = 0, Q3 = 1$.

L'information restera mémorisée. Ainsi, un registre à décalage fonctionne également comme un registre de stockage (entrée parallèle sortie parallèle).



Deuxième Phase : le décalage

Explication :

à $t = T$,

Avant l'activation de CK

D0 = 0 (forcée)

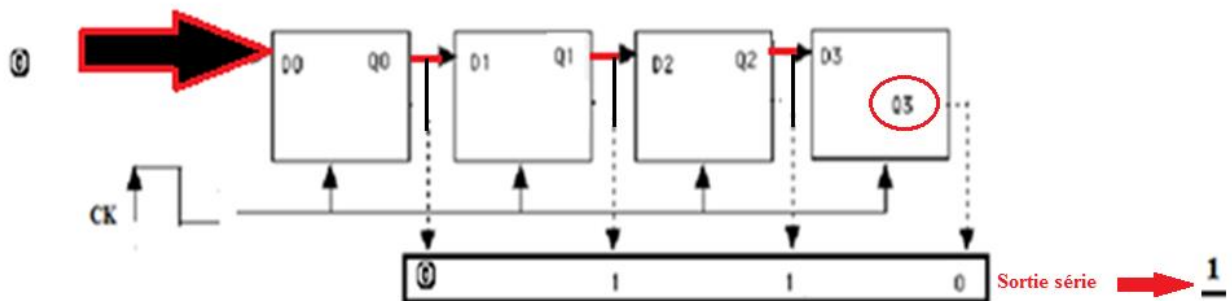
D1 = Q0 = 1

D2 = Q1 = 1

et D3 = Q2 = 0

Après l'activation de CK

$\Rightarrow Q0=0, Q1=1, Q2=1 \text{ et } Q3=0 \Rightarrow$ Sortie du premier bit : "1"



à $t = 2T$,

Avant l'activation de CK

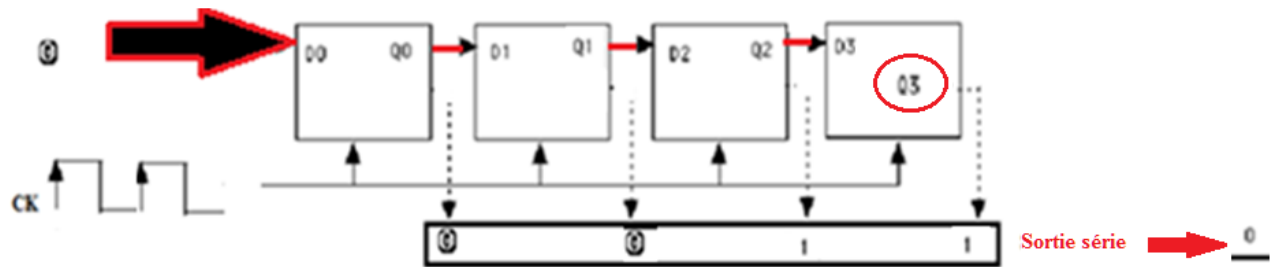
D0 = 0

D1 = Q0 = 0

D2 = Q1 = 1

et D3 = Q2 = 1

Après l'activation de CK



$\Rightarrow Q0 = 0, Q1 = 0, Q2 = 1 \text{ et } Q3 = 1 \Rightarrow \text{Sortie du deuxième bit: "0"}$

à $t = 3T$,

Avant l'activation de CK

$D0 = 0$

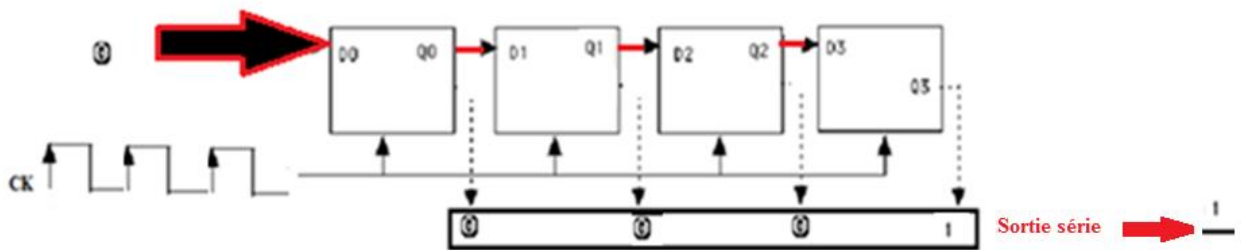
$D1 = Q0 = 0$

$D2 = Q1 = 0$

et $D3 = Q2 = 1$

Après l'activation de CK

$\Rightarrow Q0 = 0, Q1 = 0, Q2 = 0 \text{ et } Q3 = 1 \Rightarrow \text{Sortie du troisième bit: "1"}$



à $t = 4T$,

Avant l'activation de CK

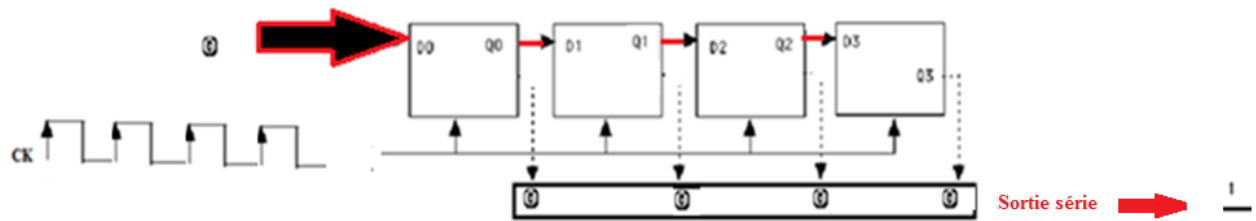
$D0 = 0$

$D1 = Q0 = 0$

$D2 = Q1 = 0$

et $D3 = Q2 = 0$

Après l'activation de CK

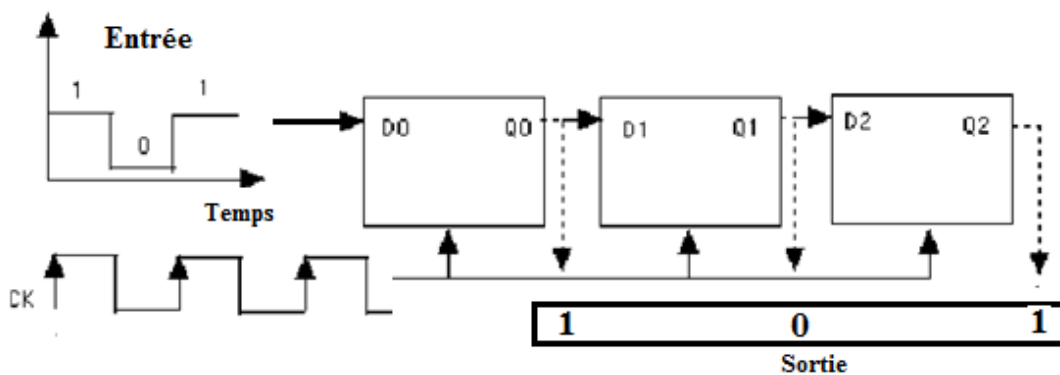


$\Rightarrow Q0=0, Q1=0, Q2=0$ et $Q3=0 \Rightarrow$ Sortie du quatrième bit : "1".

- **Registre à entrée série/ sorties parallèles : *Registre à décalage***

Registre de transfert de données, utilisé comme convertisseur parallèle-série, il est nécessaire à la réception de données lors d'une transmission série.

Exemple : Registre à décalage à droite de 3 bits (3 Bascules D)



à $t = 0$, $Q0 = 0, Q1 = 0, Q2 = 0$

à $t = T$

$D0 = 1, \Rightarrow Q0 = 1$ et $Q1 = 0, Q2 = 0$

$D1 = Q0 = 0$

$D2 = Q1 = 0$

à $t = 2T$

$D0 = 0,$

$D1 = Q0 = 1 \Rightarrow Q0 = 0$ et $Q1 = 1, Q2 = 0$

$D2 = Q1 = 0$

à $t = 3T$

$D0 = 1,$

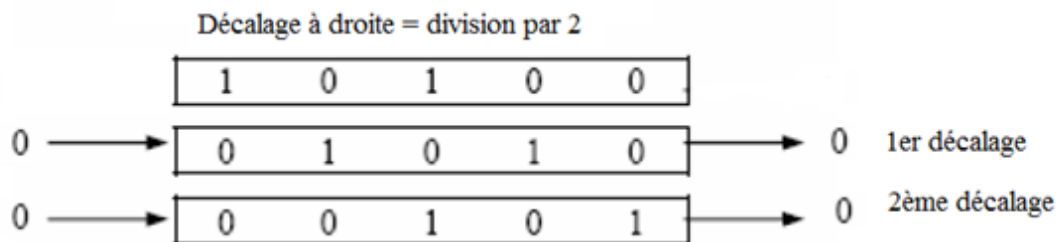
$D1 = Q0 = 0$

$D2 = Q1 = 1 \Rightarrow Q0 = 1$ et $Q1 = 0, Q2 = 1$

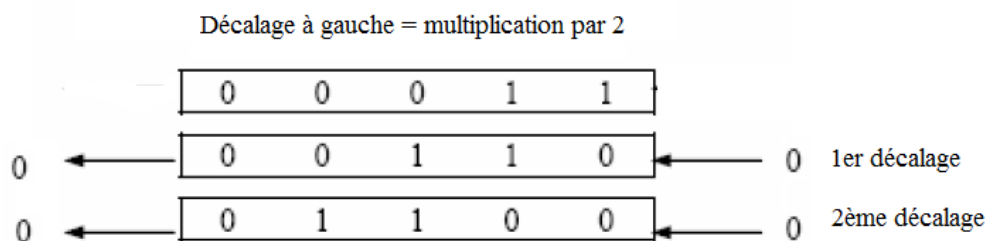
Exemple d'application des registres à décalage Application : Multiplication / Division en binaire pur:

La fonction décalage par insertion de 0 peut être utilisée pour réaliser les multiplications ou des divisions par 2.

Exemple : Registre à décalage à droite de 5 bits : $20/4=5$



Exemple : Registre à décalage à gauche de 5 bits : $3*4=12$



En général, x décalages à droite = à une division par 2^x , et x décalages à gauche = à une multiplication par 2^x .

- **Registres universels**

Un registre universel peut être un registre à « entrées parallèles sorties parallèles » (registre de stockage) ou un registre à décalage à droite ou à gauche (registre à décalage).

- **Registres et mémoires**

Les registres sont utilisés pour concevoir des mémoires vives statiques SRAM (mémoire cache des microprocesseurs, mémoire de données dans les microcontrôleurs)

- **Registres et microprocesseurs**

Les microprocesseurs utilisent les registres comme moyen de stockage d'informations (données, instructions) qui vont être traitées soit par l'UAL (Unité Arithmétique et Logique) ou décodées par

la logique de commande et de contrôle, plusieurs types de registres existent à l'intérieur d'un microprocesseur:

Accumulateurs, registres opérationnel ou à usage général, registre d'état, registres d'adresses, registres d'indexation, registres d'instructions, pointeur de pile...etc.

- **Registres et interfaces d'Entrées/Sorties**

Les registres typiques sont:

- registre de données
- registre d'état pour le fonctionnement de l'interface
- registre de contrôle pour la programmation de l'interface
- registre de direction pour la définition du sens de transfert de données.