

Chapitre3

Les registres

Objectifs :

- ✓ **Connaitre les différents types de registres.**
- ✓ **Comprendre la méthode de synthèse d'un registre binaire.**
- ✓ **Apprendre à analyser le fonctionnement d'un circuit de registre.**

1. Introduction

Un registre est un ensemble de mémoires élémentaires (bascules), synchronisées par le même signal d'horloge.

Les registres sont largement utilisés dans les systèmes de traitement numérique (les ordinateurs par exemple) pour réaliser des opérations : de mémorisation provisoire (mémoire-tampon), de décalage, de rotation, ...


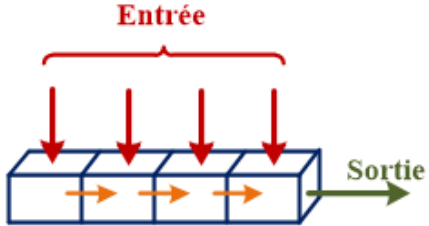
2. Caractéristiques d'un registre

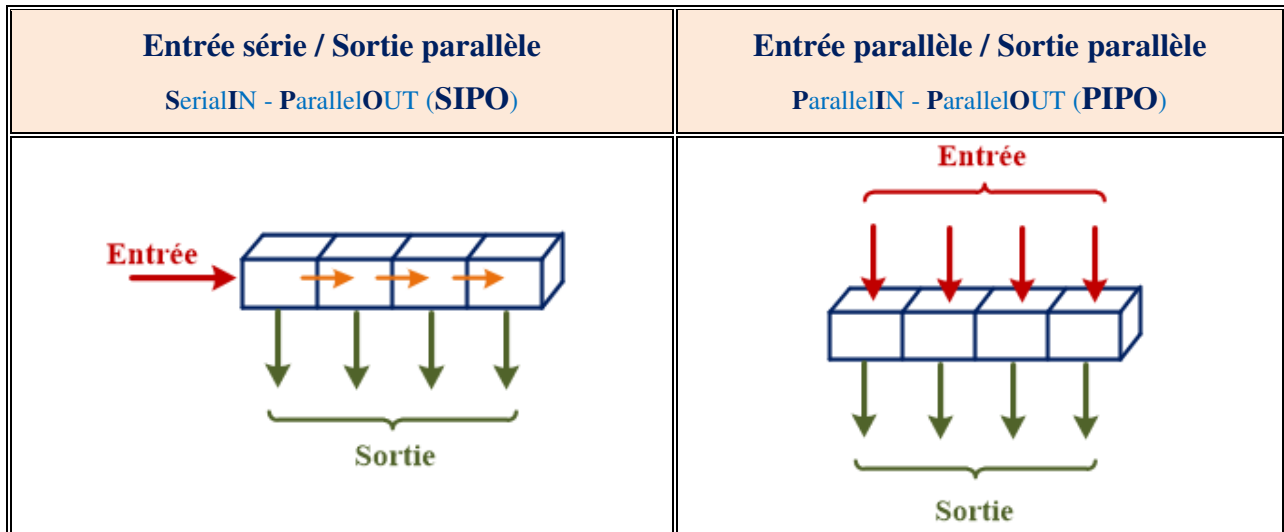
Tout registre est caractérisé par :

- **La capacité:** nombre de bits du mot binaire qu'il peut mémoriser.
- **Le mode d'écriture ou de chargement:** dépend du nombre d'entrées :
 - écriture série : génération bit par bit, avec transmission par un seul fil conducteur.
 - écriture parallèle : génération globale du mot de **n** bits, avec transmission par un bus de **n** bits (**n** fils conducteurs).
- **Le mode de lecture:**
 - lecture série : exploitation bit par bit du mot (une seule sortie).
 - lecture parallèle : exploitation globale du mot (**n** sorties).

3. Différents types de registre

Selon le mode d'accès en écriture (entrée) et en lecture (sortie), série ou parallèle, Il existe quatre types de registre :

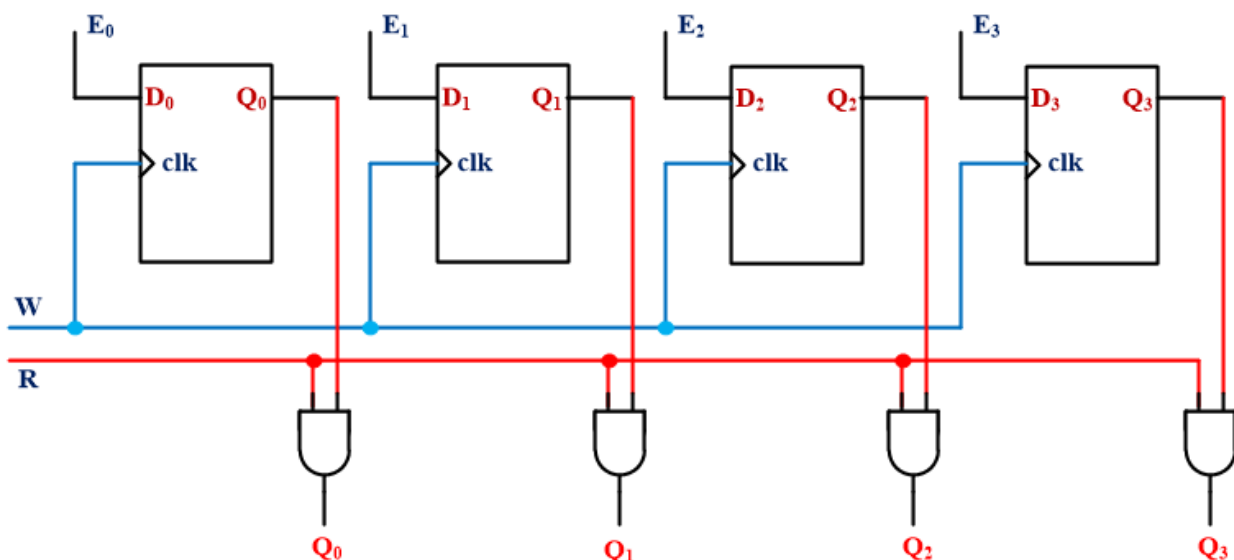
Entrée série / Sortie série SerialIN - SerialOUT(SISO)	Entrée parallèle / Sortie série ParallelIN - Serial OUT (PISO)
	



Ces quatre types peuvent être classés en deux catégories : les *registres de mémorisation* (tampon) et les *registres à décalage*.

3.1 Registre de mémorisation

Ce registre permet la mémorisation de n bits. Il est donc constitué de n bascules, mémorisant chacune un bit. L'information sur n bits est chargée (écrite) au moyen d'un signal de commande (W) qui est le signal d'horloge des bascules puis elle est conservée et devient disponible en lecture. La figure suivante donne un exemple de registre de mémorisation 4 bits réalisé à base de bascules D.



Le registre mémorise les états des entrées E_0 , E_1 , E_2 et E_3 en synchronisme avec le signal d'écriture W . Ces états sont conservés jusqu'au prochain signal de commande W . Dans cet exemple les données mémorisées peuvent être lues sur les sorties Q_0 , Q_1 , Q_2 et Q_3 au moyen du signal de validation R . On remarque que ce registre est du type **PIPO**.

3.2 Registres à décalage

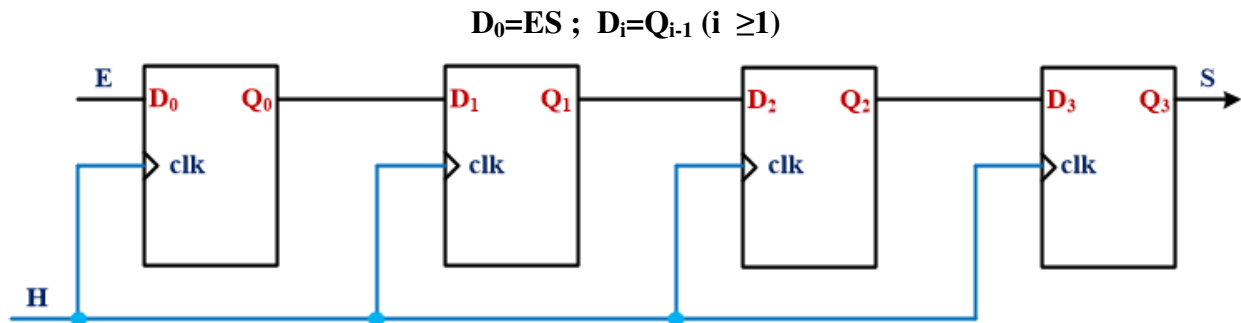
Dans un registre à décalage les bascules sont interconnectées de façon à ce que l'état logique de la sortie de la bascule de rang (i) puisse être transmis à la bascule de rang (i+1) lorsqu'un signal d'horloge est appliqué à l'ensemble des bascules. L'information peut être chargée en série (les n bits sont chargés l'un après l'autre) ou en parallèle (les n bits sont chargés simultanément).

3.2.1 Registre à décalage entrée série -sortie série

Les bits d'information sont présentés séquentiellement bit après bit à l'entrée de la première bascule et se propagent à travers le registre à chaque impulsion du signal d'horloge, pour sortir par la dernière bascule.

- **Décalage à droite**

Ci-dessous le circuit d'un registre à décalage à droite entrée série-sortie série 4 bits à base de bascules D.



Exemple : Chargement du mot binaire 1001 :

Mot **1001** ⇒

1^{ère} impulsion

2^{ème} impulsion

3^{ème} impulsion

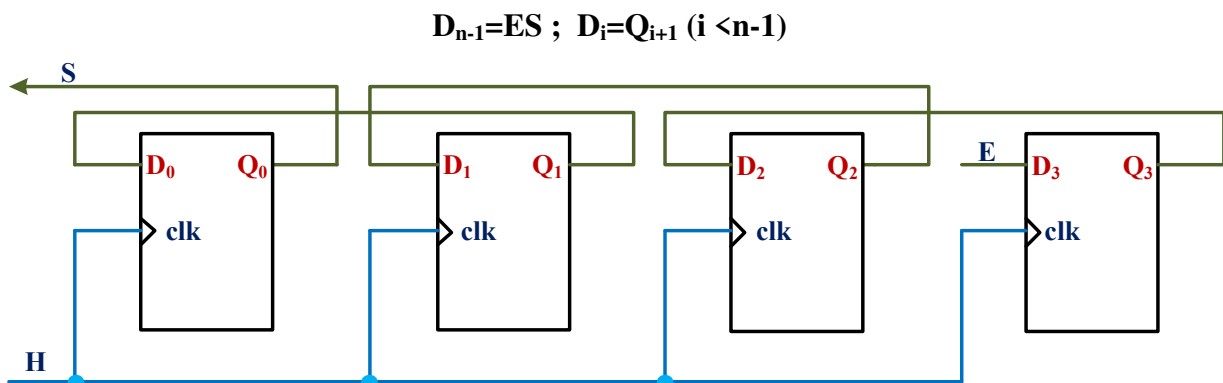
4^{ème} impulsion

Registre

1			
0	1		
0	0	1	
1	0	0	1

- **Décalage à gauche**

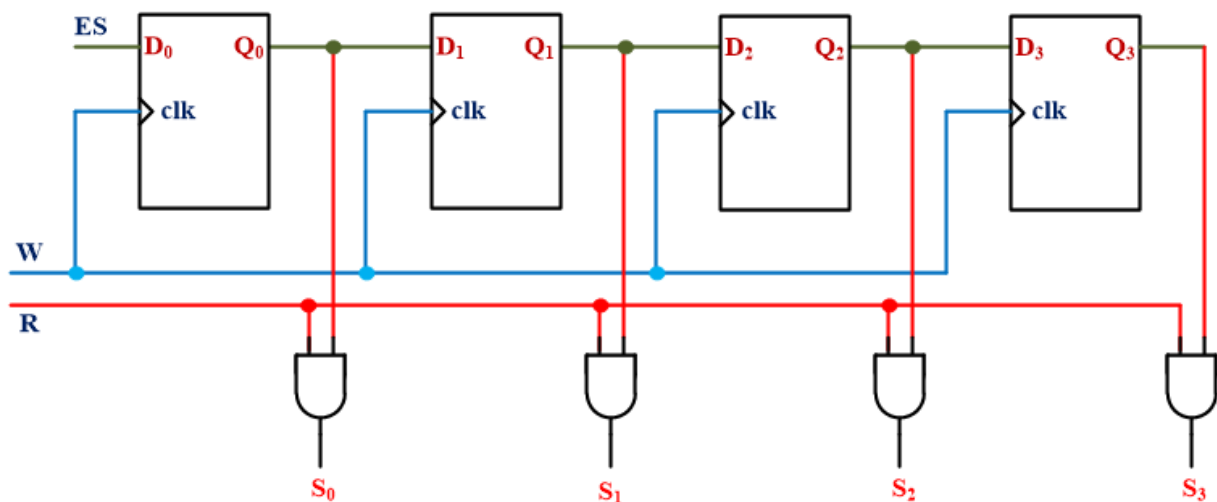
Dans ce cas l'entrée de la bascule D de rang i doit être connectée à la sortie de la bascule de rang i+1.



3.2.2 Registre à décalage entrée série - sortie parallèle

Ce type de registre permet de transformer un codage temporel (succession des bits dans le temps) en un codage spatial (information stockée en mémoire statique).

La figure suivante donne un exemple de registre de 4 bits à entrée série et sortie parallèle réalisé avec des bascules D. Le signal R (Read) n'est pas obligatoire, il permet juste de commander la lecture des sorties en mêmes temps, de façon à éviter la lecture au moment du chargement.



$$\begin{cases} D_0 = ES \\ D_i = Q_{i-1} \end{cases}$$

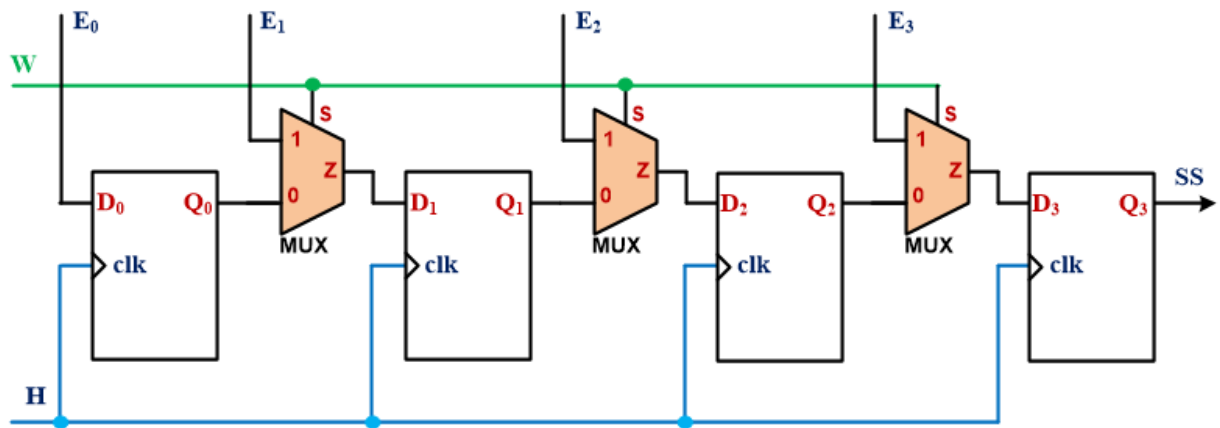
3.2.3 Registre à décalage entrée parallèle - sortie série

Le chargement parallèle des données peut s'effectuer de deux manières : synchrone ou asynchrone.

- **Chargement synchrone**

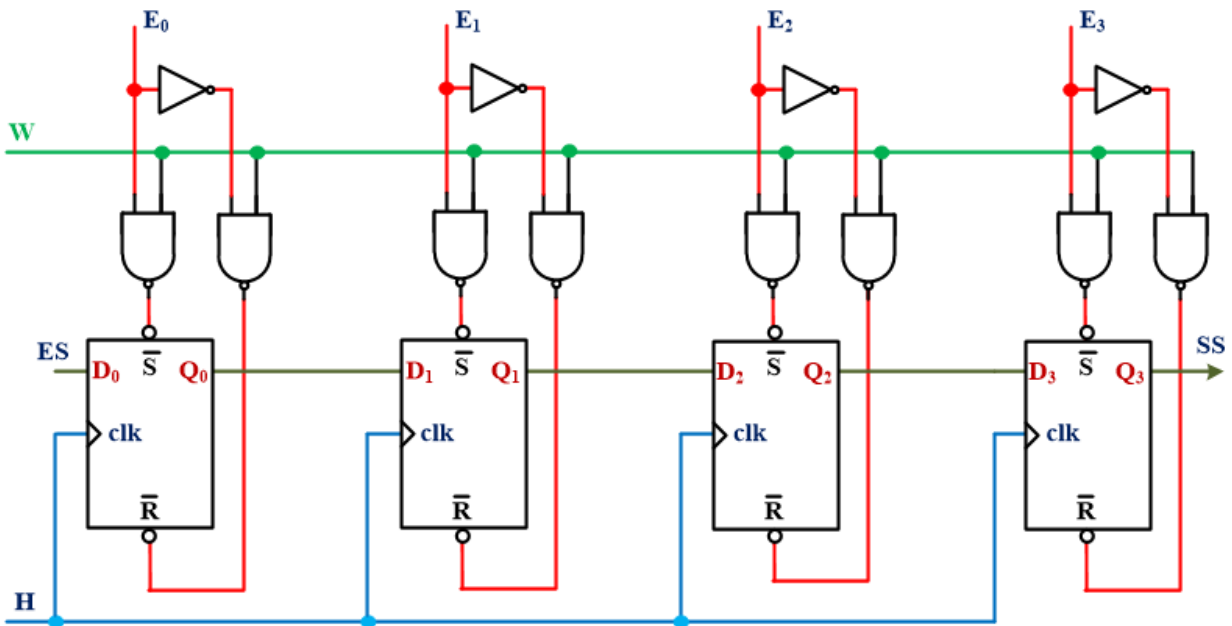
Dans ce cas il faut appliquer les données aux entrées synchrones D_i . En fonction de l'ordre de chargement (écriture) W , chaque bascule recopie l'entrée E_i ou bien la sortie de la bascule $i-1$, à chaque front d'horloge. Il faut donc utiliser un multiplexeur 2 vers 1 à l'entrée D_i de chacune des bascules 1 à $n-1$ comme le montre le circuit ci-dessous qui représente un registre 4 bits.

$$\begin{cases} D_0 = E_0 \\ D_i = Z_i = Q_{i-1}\bar{W} + E_iW \end{cases}$$



• **Chargement asynchrone**

On utilise ici les entrées asynchrones \bar{R} et \bar{S} pour forcer chaque bascule à 0 ou à 1 indépendamment du signal d'horloge. Les entrées synchrones sont utilisées pour la propagation des données à travers le registre comme l'indique la figure suivante.

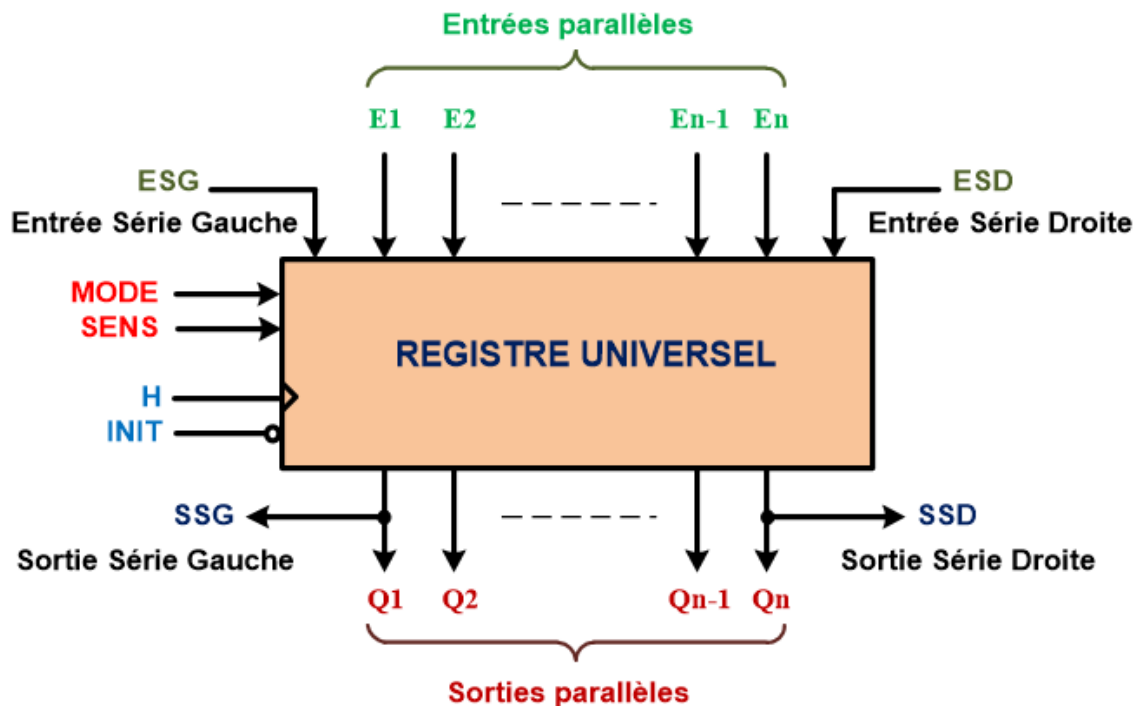


- Si $W=0$ on a : $\bar{S}_i = \bar{R}_i = 1$: fonctionnement normal des bascules (décalage à droite des données à travers le registre).
- Si $W=1$ on a :

$$\begin{cases} E_i = 0 \Rightarrow \bar{S}_i = 1 \text{ et } \bar{R}_i = 0 \Rightarrow Q_i = 0 \\ E_i = 1 \Rightarrow \bar{S}_i = 0 \text{ et } \bar{R}_i = 1 \Rightarrow Q_i = 1 \end{cases} \Rightarrow Q_i = E_i$$

3.3 Registre universel

C'est un registre qui effectue un chargement des données série ou parallèle et un décalage à gauche ou droite avec une lecture série ou parallèle. Il dispose d'entrées de mode de fonctionnement qui définissent le type de chargement et le sens de décalage. La figure ci-dessous représente la configuration d'un tel registre.



- L'entrée **MODE** permet de choisir le mode de chargement série ou parallèle.
- L'entrée **SENS** permet de choisir le sens de décalage à gauche ou à droite.
- L'entrée **INIT** permet d'initialiser le registre.

Ce type de registre existe sous forme de circuit intégré qui assure toutes les fonctions indiquées sur la figure ci-dessus (le circuit **74194** par exemple).

4. Applications des registres

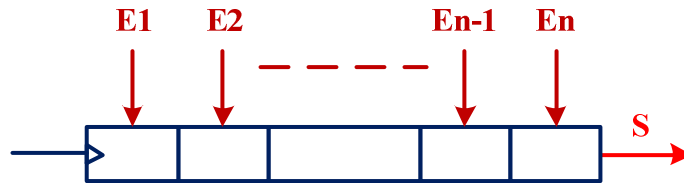
Dans ce qui suit nous citons quelques applications des registres.

4.1 Mémorisation temporaire d'une information

Les registres sont utilisés dans les microprocesseurs pour des mémorisations temporaires. En effet chaque registre mémorise temporairement un mot de **n** bits en attendant son traitement.

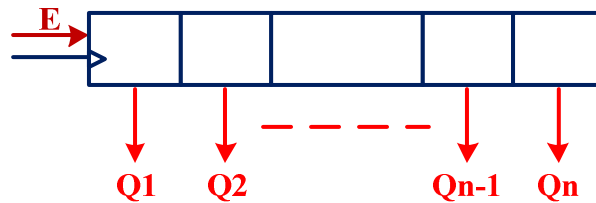
4.2 Conversion parallèle-série de mots binaires

Le mot binaire sur n bits est chargé en parallèle dans le registre puis des opérations de décalage successives permettent de le convertir en série.



4.3 Conversion série-parallèle d'un train binaire

Un train binaire est lit en série et décalé puis récupéré sous forme binaire sur les sorties Q1 à Qn.



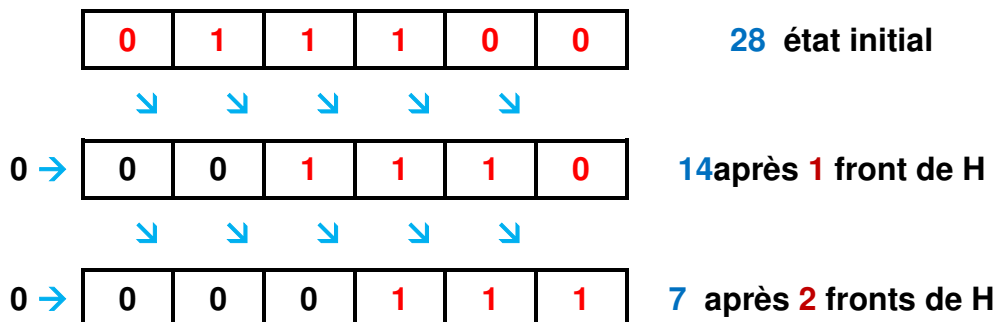
4.4 Ligne à retard numérique

Dans ce cas le registre permet de retarder un train binaire de n périodes du signal d'horloge.



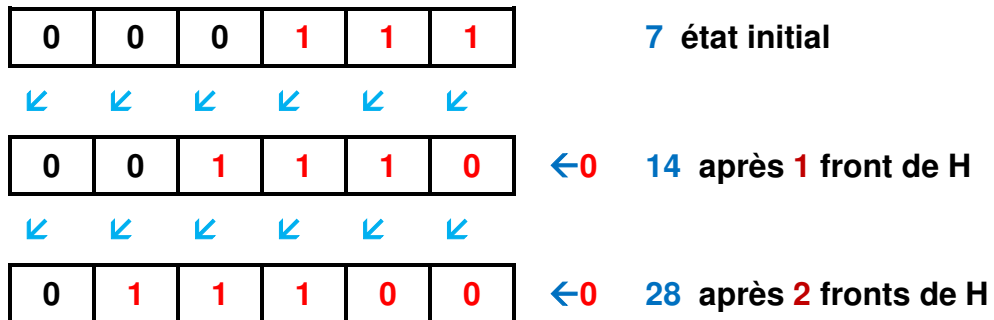
4.5 Division et multiplication par 2ⁿ

- *Décalage à droite de n bits : division par 2ⁿ*



$$\frac{28}{2^2} = 7 \quad : \text{division par } 4 \text{ soit } 2^2 \text{ après } 2 \text{ impulsions d'horloge.}$$

- **Décalage à gauche de n bits : multiplication par 2^n**



$7 \times 2^2 = 28$: multiplication par 4 soit 2^2 après 2 impulsions d'horloge.

5. Exercice d'application

Réaliser, à base de bascules **D**, un registre à décalage à droite et à gauche 4 bits.

