



## Cours exposé

# ELECTRONIQUE NUMERIQUE

email : nasser\_baghdad @ yahoo.fr

# ELECTRONIQUE NUMERIQUE

## Sommaire

Chapitre I : Technologies des circuits logiques : TTL et CMOS

Chapitre II : Les bases de numération

Chapitre III : Les portes logiques

Chapitre IV : Les fonctions binaires

Chapitre V : Les circuits combinatoires

Chapitre VI : Les circuits séquentiels

# ELECTRONIQUE NUMERIQUE

## Chapitre. VI

### Les circuits séquentiels

# Chapitre VI : Les circuits séquentiels

I. Les bascules

II. Les circuits de comptage : compteurs/décompteurs

III. Les registres

IV. Les mémoires

# Chapitre VI : Les circuits séquentiels

## III. Les registres

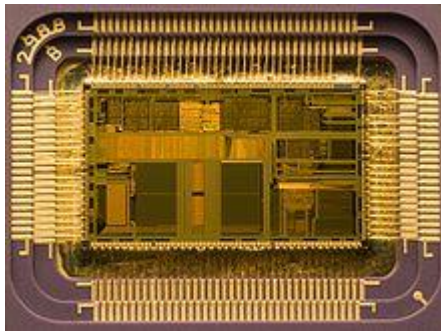
# Chapitre VI : Les circuits séquentiels

- 1°) Notion de processeur
- 2°) Notion de registre
- 3°) Registre à écriture (ou chargement) parallèle / lecture (ou déchargement) parallèle
- 4°) Registre à écriture (ou chargement) parallèle / lecture (ou déchargement) série
- 5°) Registre à écriture (ou chargement) série / lecture ((ou déchargement) parallèle
- 6°) Registre à écriture (ou chargement) série / lecture (ou déchargement) série
- 7°) Registre à décalage réversible
- 8°) Registre à décalage avec rétroaction linéaire
- 9°) Registres universels
- 10°) Exemples d' application des registres à décalage
- 11°) Registres intégrés 74LS194

# Chapitre VI : Les circuits séquentiels

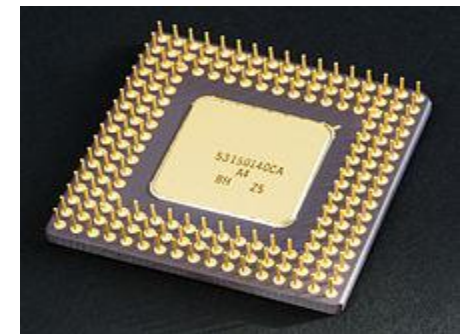
## 1°) Notion de processeur

- ▶ Le processeur (ou CPU) est le composant de l'ordinateur qui exécute les instructions machine des programmes informatiques.
- ▶ Un processeur construit en un seul circuit intégré est un microprocesseur.



Intel - 1968

29.000 - 1978



Intel 80486DX2 microprocesseur  
en boîtier céramique PGA

La puce d'un microprocesseur Intel 80486DX2 dans son boîtier (taille réelle : 12 × 6,75 mm).

### 1<sup>er</sup> microprocesseur :

- ▶ L'invention du microprocesseur date de 1971, lorsque la société Intel réalisa le 4004.

### Caractéristiques :

- ▶ deux paramètres principaux permettent de caractériser un microprocesseur :
  - son architecture;
  - sa fréquence.

# Chapitre VI : Les circuits séquentiels

## Caractéristiques :

- Production : de 15 novembre 1971 à 1981
- Fabricant : Intel
- Fréquence : 740 kHz
- Finesse de gravure : 10  $\mu\text{m}$
- Architecture : 4 bits
- Boîtier : DIP à 16 broches
- 2 300 transistors



Version céramique C4004



# Chapitre VI : Les circuits séquentiels

## Intel va commercialiser un processeur Core i9 équipé de 18 cœurs et de 36 threads

### Technologie :

AMD- 1969

L'arrivée de la gamme Ryzen d'AMD sur le marché avait fait naître quelques doutes sur la hiérarchie actuelle des fondeurs de processeurs.

16 cœurs et 32 threads



Avec ses X-Series, dont de monstrueux i9, Intel semble dire qu'il est toujours **numéro 1**.

le i9-7920X embarque 12 cœurs et 24 threads  
le i9-7940X embarque 14 cœurs et 28 threads  
le i9-7960X embarque 14 cœurs et 28 threads  
le i9-7980XE embarque 18 cœurs et 36 threads.



# Chapitre VI : Les circuits séquentiels

## Intel : i9-7920X

	Base Clock speed	Intel Turbo Boost Technology 2.0	Intel Turbo Boost Max Technology 3.0
Intel : i9-7920X 12 cœurs et 24 threads	3,3 GHz	4,3 GHz	4,5 GHz

### Processeurs grand public :

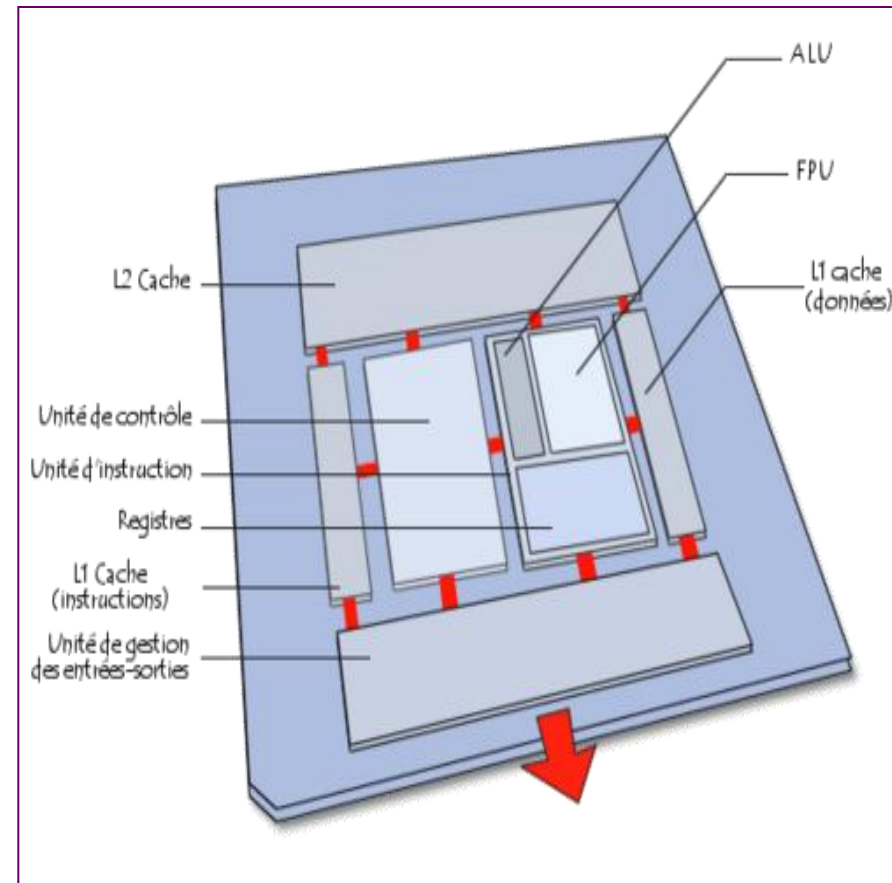
- 1971 : 4004 : 2 300
- 1978 : 8086 : 29 000
- 1982 : 80286 : 275 000
- 1989 : 80486 : 1,16 million
- 1993 : Pentium : 3,1 millions
- 1995 : Pentium Pro : 5,5 millions
- 1997 : Pentium II : 27 millions
- 2001 : Pentium 4 : 42 millions
- 2004 : Pentium Extreme Edition : 169 millions
- 2006 : Core 2 Duo : 291 millions
- 2006 : Core 2 Quad : 582 millions
- 2008 : Core i7 Bloomfield : 730 millions
- 2010 : Core i7 Gulftown : 1,17 milliard

# Chapitre VI : Les circuits séquentiels

## Composition d'un processeur

► Les parties essentielles d'un processeur sont :

- **l'Unité arithmétique et logique (UAL)** qui prend en charge les opérations logiques et arithmétiques ;
- **l'unité de commande (contrôle) (UC)** ou séquenceur, qui permet de synchroniser les différents éléments du processeur.
- **les registres (R)**, qui sont des mémoires de petite taille (quelques octets), suffisamment rapides pour que l'UAL puisse manipuler leur contenu à chaque cycle de l'horloge.
- **l'horloge (H)** qui synchronise toutes les actions de l'unité centrale.
- **l'unité d'entrée-sortie (E/S)**, qui prend en charge la communication avec la mémoire de l'ordinateur ou la transmission des ordres permettant au processeur d'accéder aux périphériques de l'ordinateur.

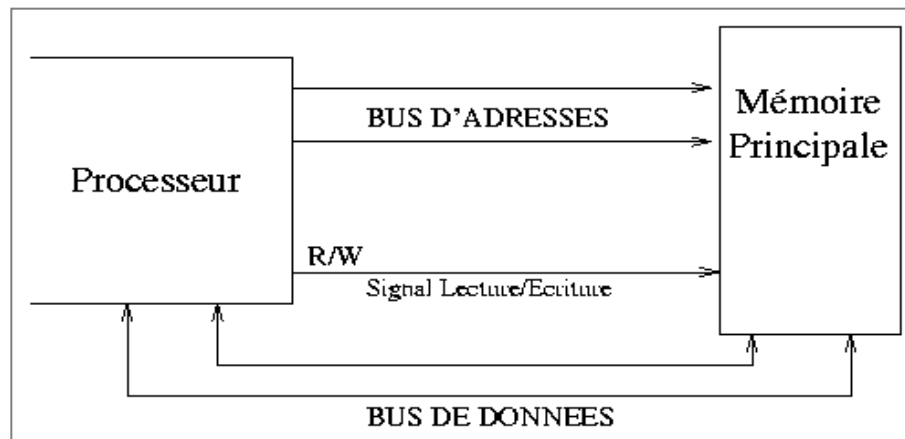


# Chapitre VI : Les circuits séquentiels

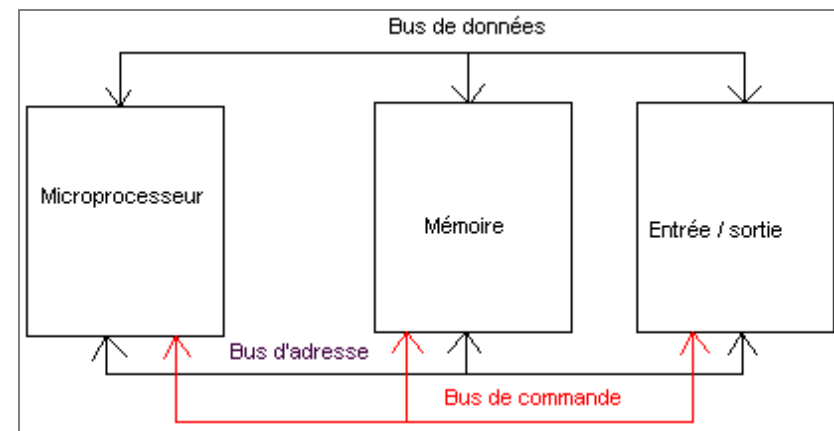
## Taches du processeur :

► Il y a quatre étapes que presque toutes les architectures utilisent :

- **fetch** - recherche de l'instruction ;
- **decode** - décodage de l'instruction (opération et opérandes) ;
- **execute** - exécution de l'opération ;
- **writeback** - écriture du résultat.



La mémoire est divisée en emplacements de taille fixe  
(par exemple 8 bits)  
utilisés pour stocker instructions et données.



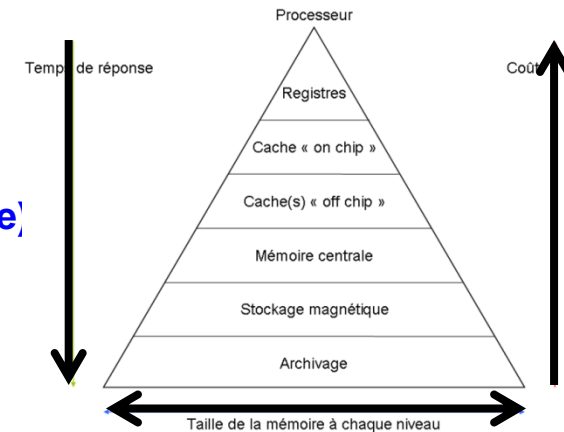
- bus d'adresses
- bus de données
- bus de commande

# Chapitre VI : Les circuits séquentiels

## 2°) Notion de registre

➤ Un registre est un emplacement de mémoire interne à un processeur chargé de stocker des données temporaires qui peuvent être :

- une adresse mémoire
- des opérandes  
(des données en entrée d'une opération arithmétique ou logique)
- le résultat d'un calcul effectué par l'UAL.



- Les registres sont utilisés pour assurer le fonctionnement entre l'unité de traitement (partie opérative) et l'unité de commande (partie contrôle).
- Les registres se situent au sommet de la hiérarchie mémoire : il s'agit de la mémoire la plus rapide d'un ordinateur, mais dont le coût de fabrication est le plus élevé car la place dans un microprocesseur est limitée.
- Leur capacité dépasse donc rarement quelques dizaines d'octets.

# Chapitre VI : Les circuits séquentiels

## Les registres

- ▶ Les registres sont de deux types : ceux accessibles par le programme (spécialisés) et ceux qui ne le sont pas (généraux) :
  - **les registres de données** stockant temporairement des informations ;
  - **les registres d'adresses** qui sont des pointeurs stockant des adresses importantes, en particulier le **registre d'index** qui va permettre l'adressage dit indexé
  - **compteur ordinal** : ou compteur d'instruction, encore appelé compteur de programme, qui suit pas à pas l'exécution d'un programme ;
  - **accumulateur** : ce registre est utilisé pour stocker les données en cours de traitement par l'UAL ;
  - **registre d'instructions** : il contient l'instruction en cours de traitement ;
  - **registre d'état** : comporte différents bits positionnés à 0 ou 1 et indiquant, selon leur état, si le résultat d'une opération est nul, ou négatif, avec ou sans retenue, etc...
  - **pointeur de pile** : est un registre qui pointe vers une zone particulière de la mémoire appelée pile. Son rôle est important lors des interruptions
- **registres généraux** : ces registres sont disponibles pour les calculs ;

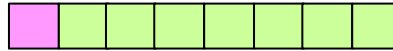
# Chapitre VI : Les circuits séquentiels

## Registre d'une mémoire

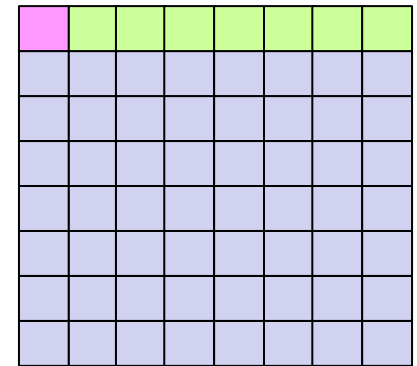
- ▶ Une bascule permet de mémoriser une information binaire élémentaire (1 bit).
- ▶ Un registre est un ensemble de  $n$  bascules qui sert à mémoriser un mot binaire de  $n$  bits (Il faut autant de bascules qu'il y a d'éléments binaires à mémoriser).
- ▶ Les registres sont caractérisés par le nombre d'élément mémoire et par les méthodes d'introductions et de restitution de l'information.



bascule



registre



mémoire

- ▶ L'interconnexion entre les bascules permet certaine manipulation de l'information stockée.

# Chapitre VI : Les circuits séquentiels

- ▶ Il existe dans un ordinateur plusieurs variétés de registres, les registres parallèles, les registres à décalage (décalage à droite ou décalage à gauche) et les registres séries.
- ▶ Les bascules de type D sont les plus utilisées pour construire des registres de différents types en fonction de la disposition des entrées et des sorties des bascules :
  - les registres à entrée série/sortie série;
  - les registres à entrée série/sortie parallèle;
  - les registres à entrée parallèle/sortie parallèle;
  - les registres à entrée parallèle/sortie série.



# Chapitre VI : Les circuits séquentiels

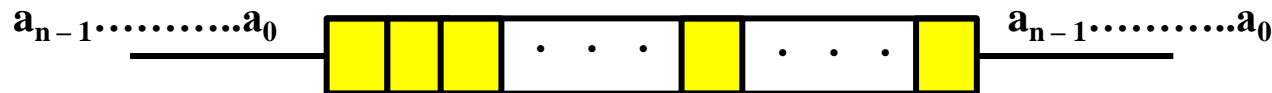
## ■ Notations des différents type de registres :

► Le registre peut **recevoir** une information :

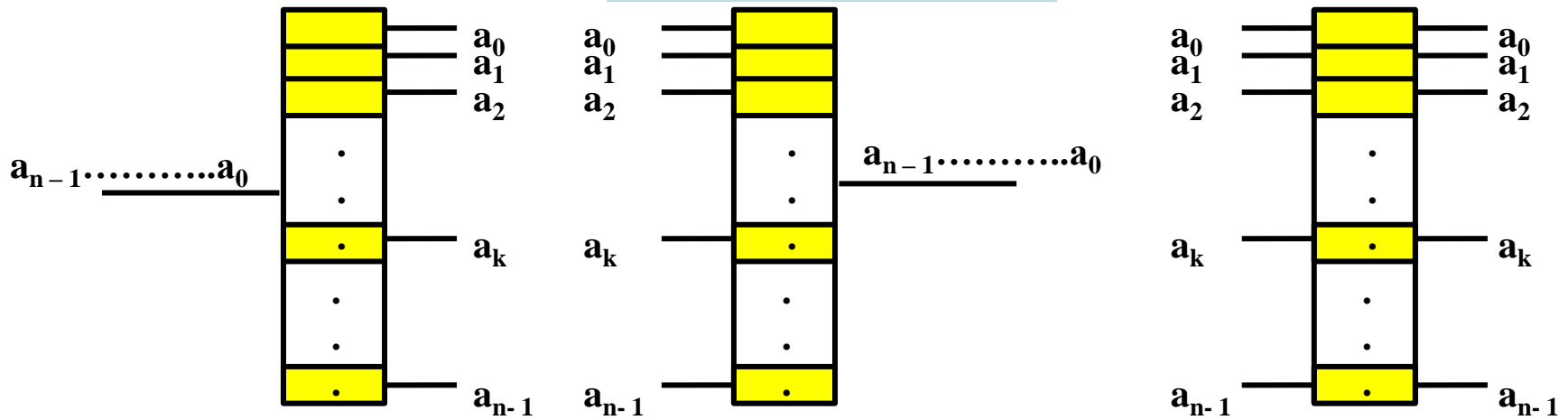
- soit par chargement parallèle (c) et (d) (l'information doit être présentée sur les entrées parallèles).
- soit par chargement série (a) et (b) (décalage à droite ou à gauche)

► Le registre peut **retransmettre** l'information :

- par déchargement parallèle (b) et (d) (lecture des sorties parallèles)
- par déchargement série (a) et (c) (décalage à droite ou à gauche)



(a) : registre série/série



(b) : registre série/parallèle

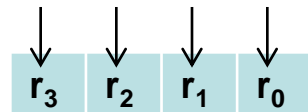
(c) : registre parallèle/série

(d) : registre parallèle/parallèle

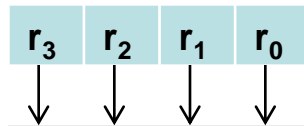
# Chapitre VI : Les circuits séquentiels

## Modes de fonctionnement d'un registre.

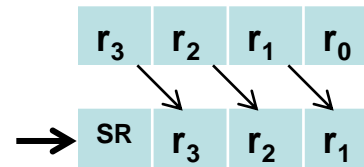
► Les registres peuvent réaliser différentes manipulations :



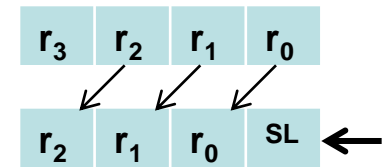
(a) écriture parallèle



(b) Lecture  
(sans écriture)



(c) écriture série  
(avec décalage à droite)

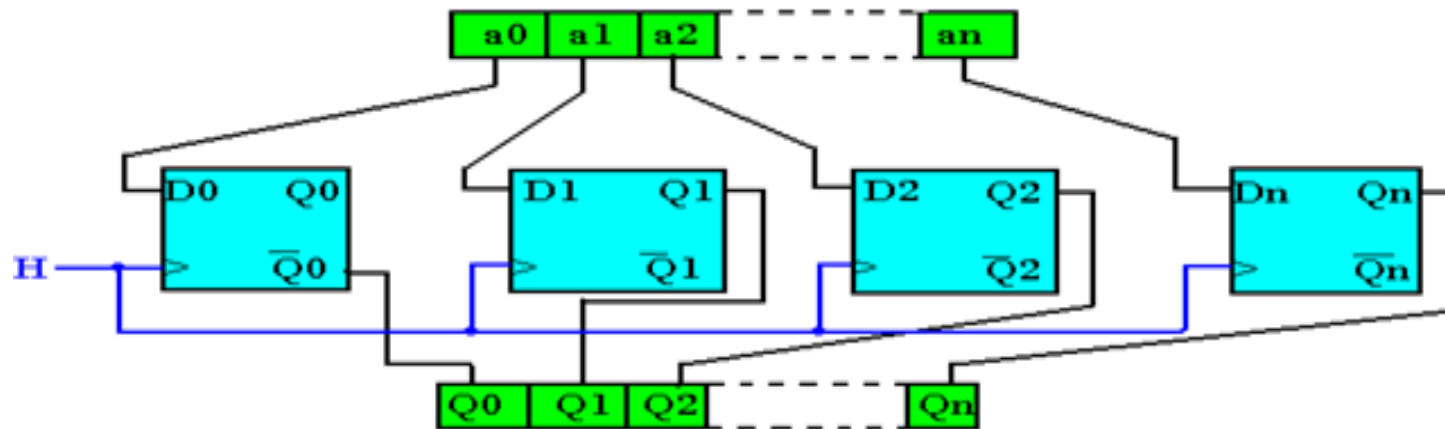


(d) écriture série  
(avec décalage à gauche)

# Chapitre VI : Les circuits séquentiels

## 3°) Registre à écriture parallèle / lecture parallèle

- Tous les bits sont transmis et mémorisés en même temps dans ce type de registre.



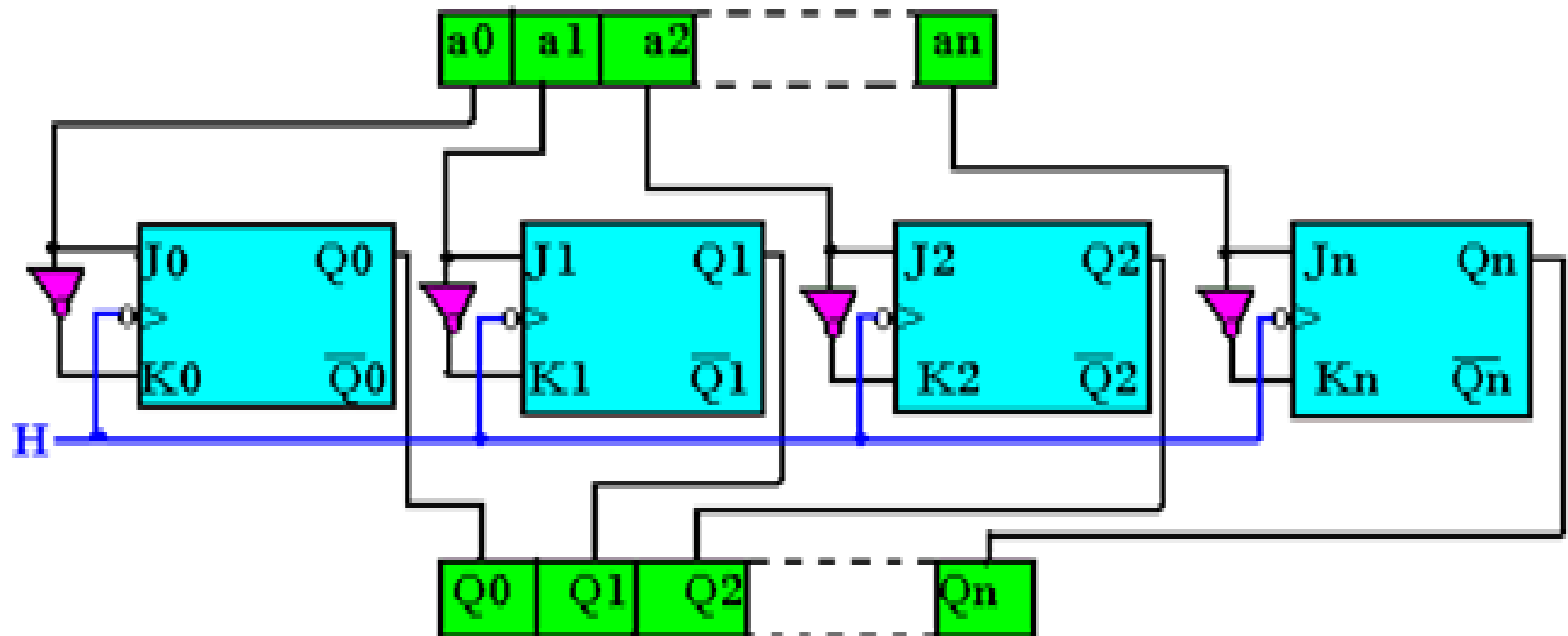
Entrée parallèle et sortie parallèle :  
EPSP (PIPO)

Exemple :

Avant le Front Montant	{	D0	D1	D2	D3
		1	0	0	1
Après le Front Montant	{	Q0	Q1	Q2	Q3
		0	0	0	0
	{	Q0	Q1	Q2	Q3
		1	0	0	1

# Chapitre VI : Les circuits séquentiels

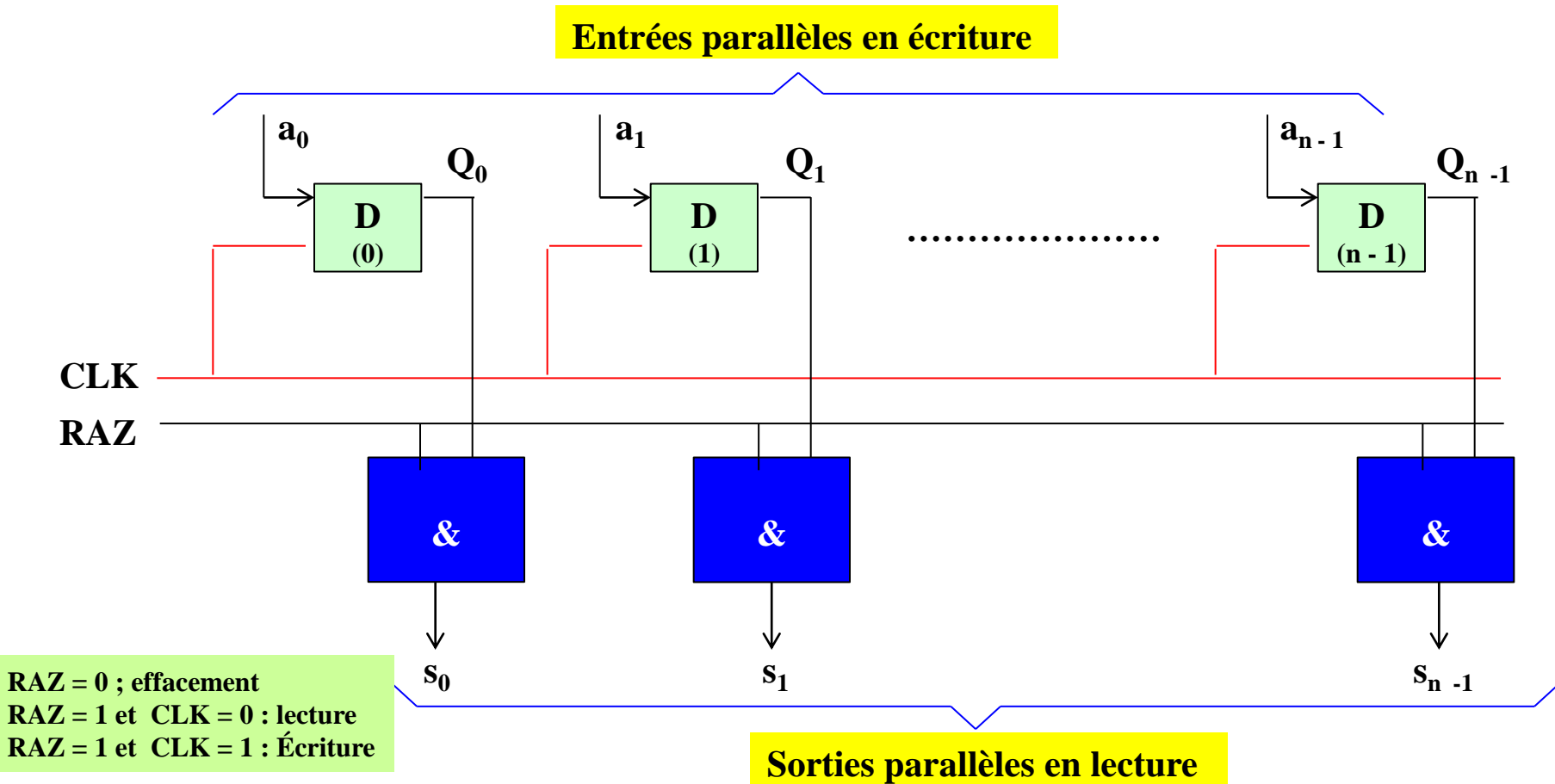
## Exemple n° 1 : avec les bascules JK



# Chapitre VI : Les circuits séquentiels

## Exemple n°2 : avec des bascules D

► Voici un exemple de registre à  $n$  entrées parallèles ( $a_0, a_1, \dots, a_{n-1}$ ) et à  $n$  sorties parallèles ( $s_0, s_1, \dots, s_{n-1}$ ) construit avec des bascules de type D :



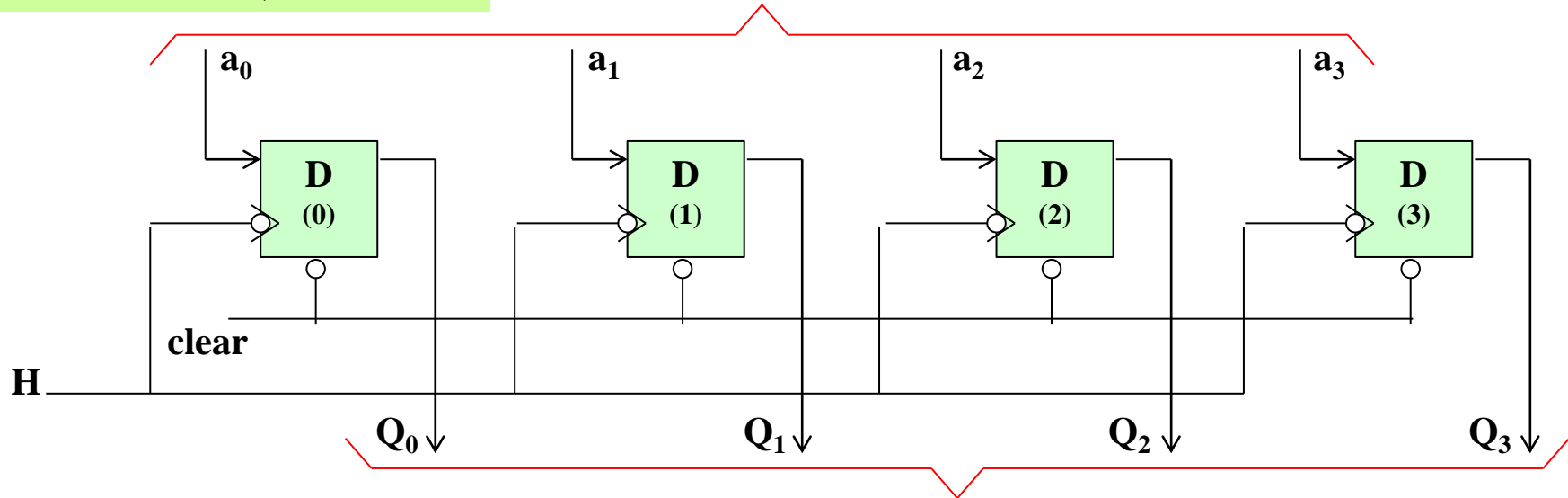
# Chapitre VI : Les circuits séquentiels

## Exemple n°3 : mémorisation temporaire

### ■ Registre de mémorisation temporaire sans entrée de chargement (Load)

Clear = 0 ; effacement  
Clear = 1 et H = 0, 1,  $\uparrow$  : lecture  
Clear = 1 et H =  $\downarrow$  : Écriture

Entrées parallèles en écriture



Sorties parallèles en lecture

### Remarque :

On peut utiliser les entrées asynchrones Preset et Clear pour commander la lecture et l'écriture.

# Chapitre VI : Les circuits séquentiels

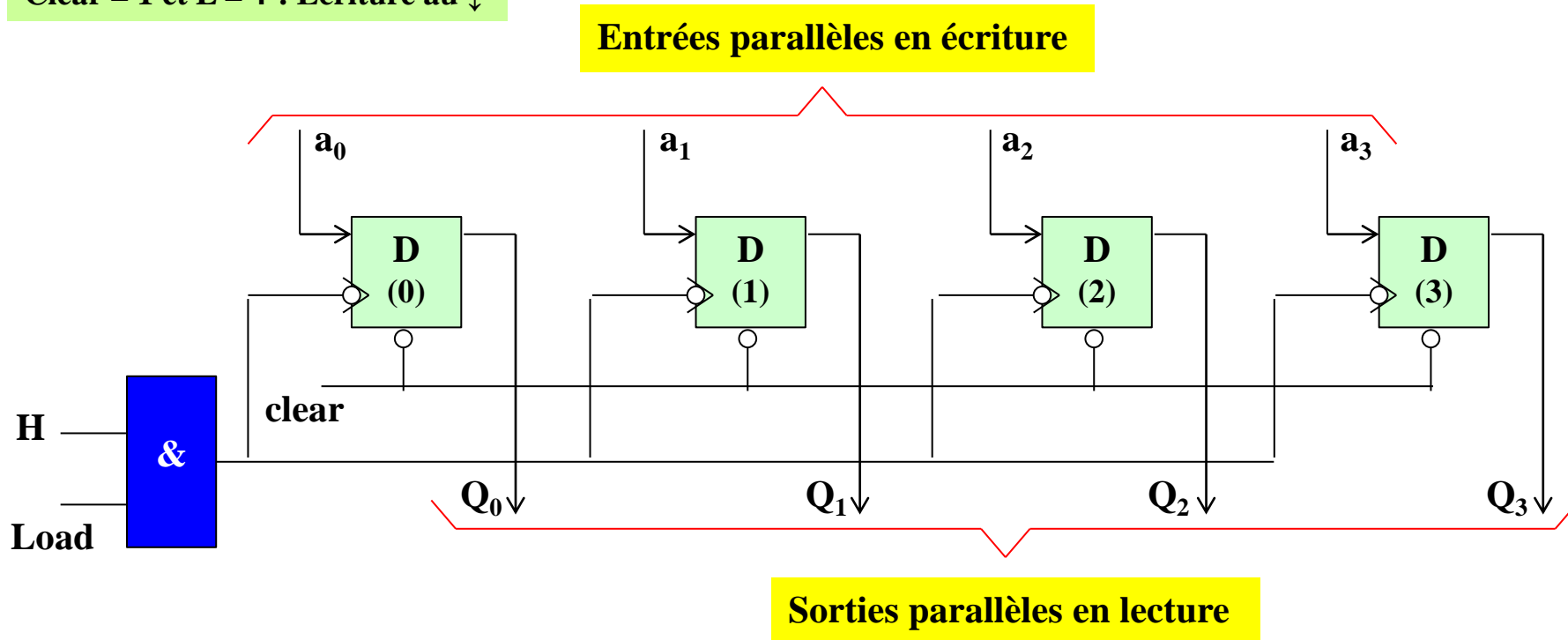
## Exemple n° 3 : Mémorisation

### ■ Registre de mémorisation avec entrée de chargement (Load)

Clear = 0 ; effacement

Clear = 1 et L = 0 : lecture

Clear = 1 et L = 1 : Écriture au ↓



# Chapitre VI : Les circuits séquentiels

## Exemple d'application :

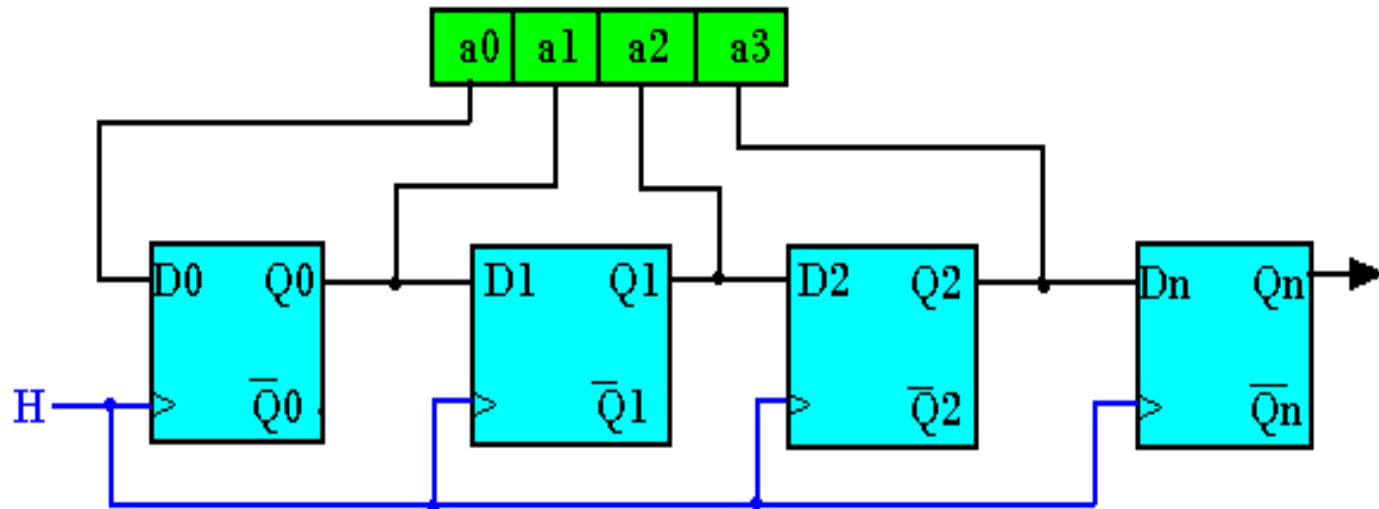
- ▶ PIPO : En décalant tous les bits d'un nombre binaire vers la droite ou vers la gauche, on divise ou on multiplie le nombre par 2.
- ▶ Un registre PIPO peut donc être utilisé pour effectuer des calculs (**multiplication ou division par une puissance de 2**).
- ▶ Il suffit d'opérer le nombre adéquat de décalages vers la gauche ou la droite entre le moment où l'on introduit les bits dans le registre et le moment où on les récupère.



# Chapitre VI : Les circuits séquentiels

## 4°) Registre à écriture parallèle / lecture série

► Ce type de registre dans lequel les données sont placées en parallèle seront lues en série.



Entrée parallèle et sortie série :  
EPSS (PISO)

# Chapitre VI : Les circuits séquentiels

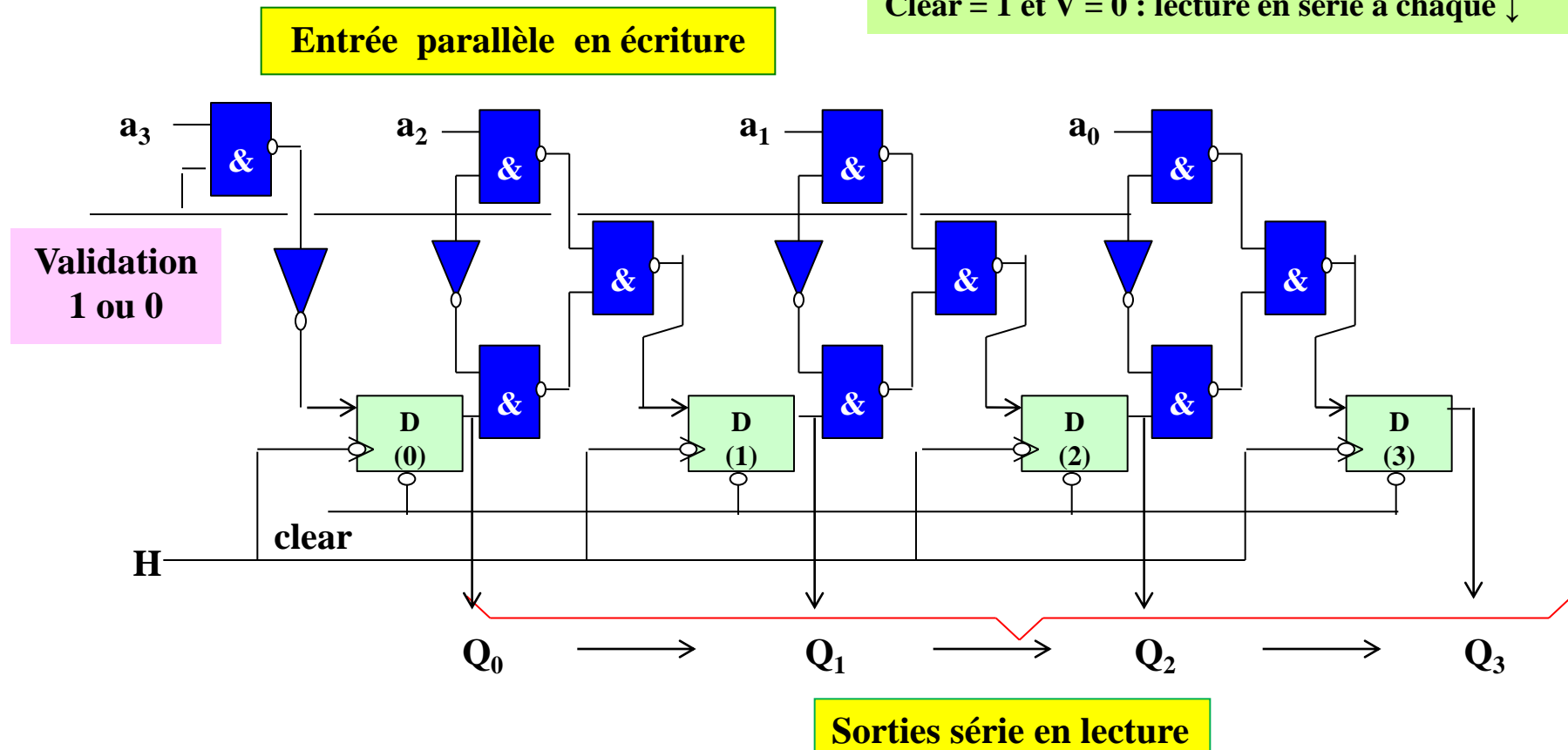
## Exemple n° 1 : Mémorisation temporaire

### ► Circuit de conversion parallèle - série à l'aide des bascules DH

Clear = 0 ; effacement

Clear = 1 et V = 1 : Écriture en parallèle au ↓

Clear = 1 et V = 0 : lecture en série à chaque ↓



# Chapitre VI : Les circuits séquentiels

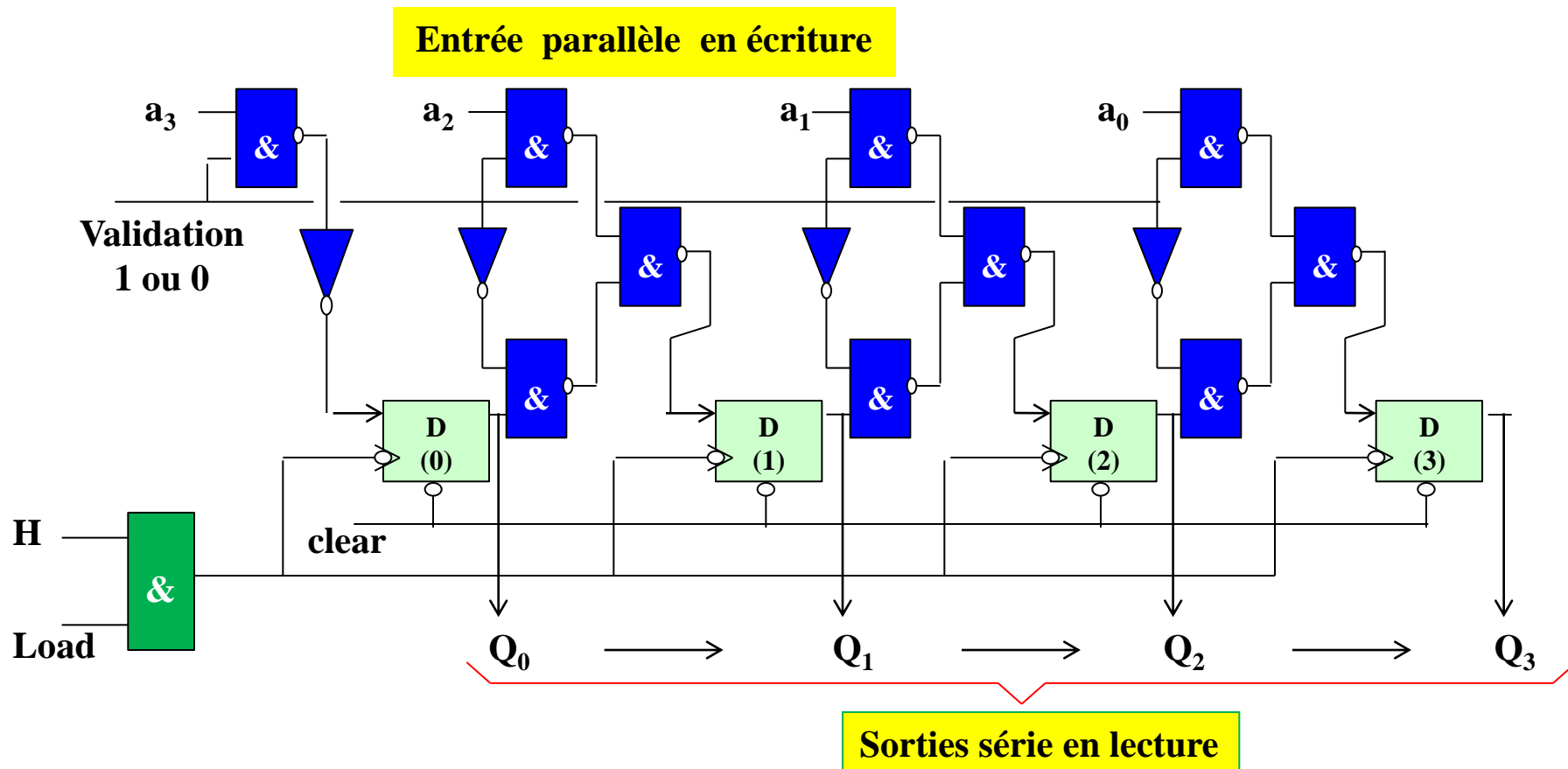
## Exemple n° 2 : Mémorisation

Clear = 0 : effacement

Clear = 1 et L = 0 : lecture en parallèle

Clear = 1, L = 1 et V = 1 : Écriture en parallèle au ↓

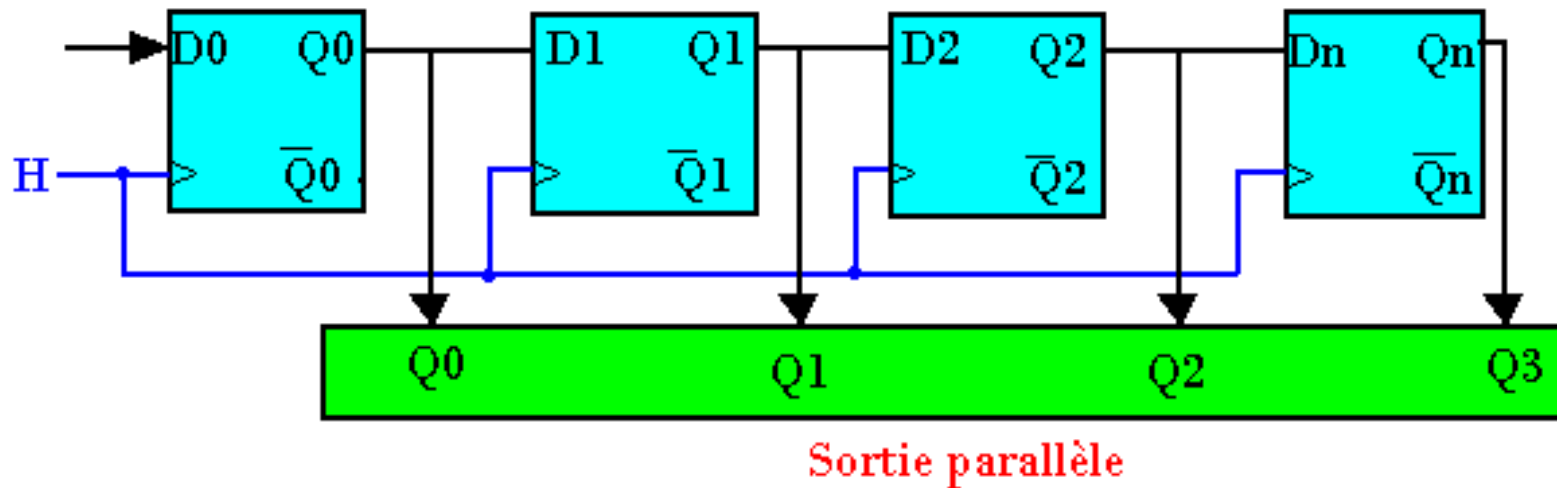
Clear = 1, L = 1 et V = 0 : lecture en série à chaque ↓



# Chapitre VI : Les circuits séquentiels

## 5°) Registre à écriture série / lecture parallèle

- C'est un type de registre dans lequel les données arrivent en série et ressortent en parallèle.



Entrée série et sortie parallèle :  
ESSP (SIPO)

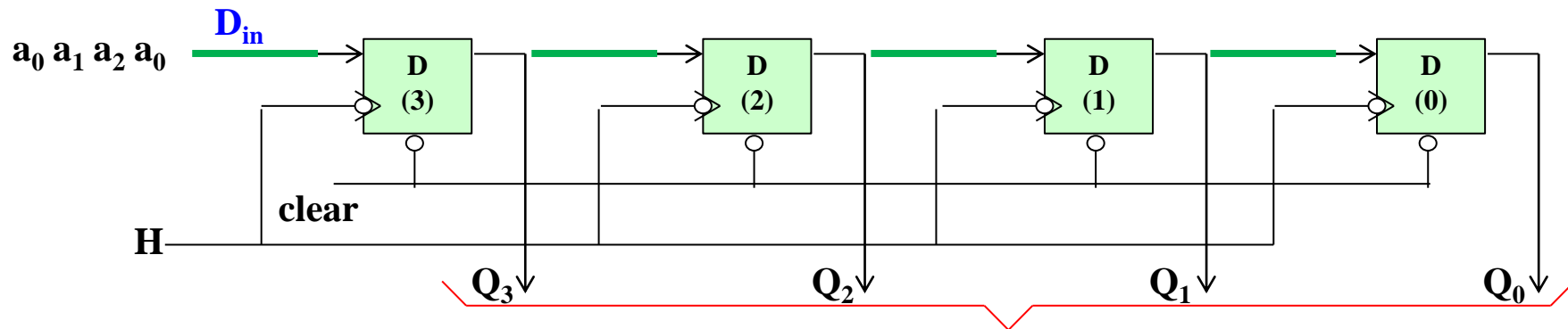
- La transmission parallèle des informations d'un registre à un autre est la plus facile.
- La transmission série utilise peu d'éléments donc peu coûteux.

# Chapitre VI : Les circuits séquentiels

## Exemple n°1 : Mémorisation temporaire

### ► Circuit de conversion série – parallèle à l'aide des bascules D

#### Entrée série en écriture



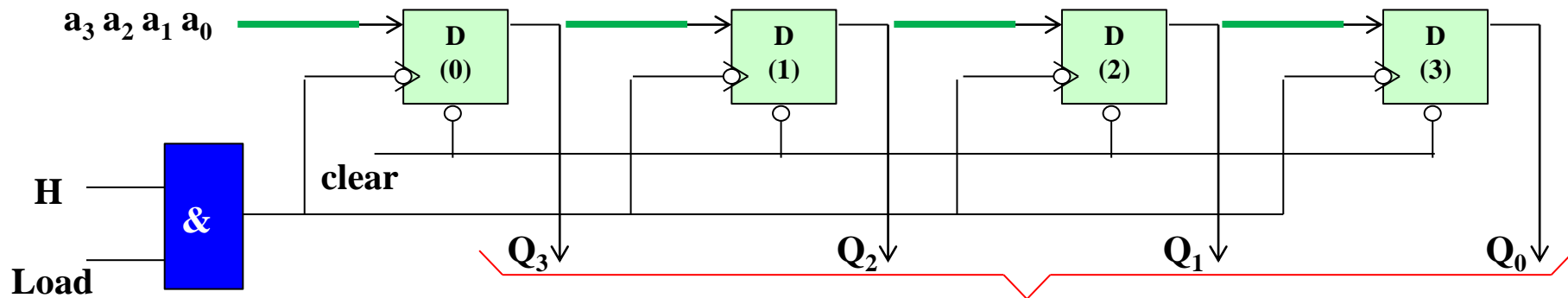
#### Sorties parallèles en lecture

- Pour charger initialement ce registre avec le nombre  $a_3 a_2 a_1 a_0$  ( $a_0$  ou est le bit du poids le plus faible), on applique successivement  $a_0$ ,  $a_1$ ,  $a_2$  puis  $a_3$  sur l'entrée  $D_{in}$  au rythme du signal  $H$ .
- On parle alors de chargement série. Après 4 impulsions de l'horloge, les 4 bits sont stockés dans le registre et ils apparaissent simultanément sur les lignes de sortie.

# Chapitre VI : Les circuits séquentiels

## Exemple n° 2 : Mémorisation

### Entrée série en écriture



### Sorties parallèles en lecture

- L'entrée 'Load' permet de charger ou de ne pas charger le registre.
- Tant que Load est à 1, le registre charge les valeurs d'entrées bit par bit à chaque front actif de H
- Si Load = 0, le registre mémorise les dernières données stockées.

# Chapitre VI : Les circuits séquentiels

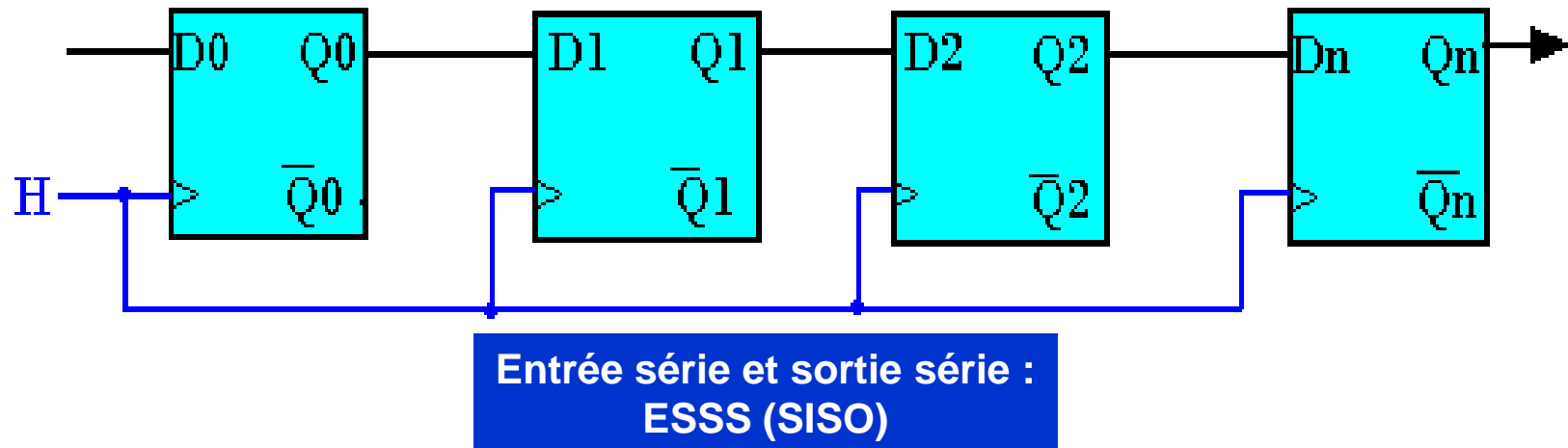
## Exemple d'application : PISO et SIPO

- ▶ **PISO et SIPO** : Ces deux types de registres sont utilisés dans les **liaisons série** ; ils forment la base des **UART** (émetteur-récepteur asynchrone universel). et des **modems** (modulateur – démodulateur).
- ▶ Imaginons que l'on veuille transmettre une information entre deux ordinateurs distants de quelques mètres ou dizaines de mètres.
- ▶ Transmettre l'information sous forme "parallèle" nécessiterait au moins 9 fils (8 pour les 8 bits, un pour la masse), sans compter les fils supplémentaires pour le dialogue entre les ordinateurs.
- ▶ Il est plus simple d'employer un registre **PISO** pour envoyer les bits constituant chaque octet que l'on désire transmettre en une suite de 8 bits apparaissant l'un après l'autre sur une seule ligne.
- ▶ Au bout de la ligne, un registre **SIPO** reçoit les bits qui arrivent à la queue-leu-leu et reconstitue des octets qui sont transmis à l'ordinateur de destination.

# Chapitre VI : Les circuits séquentiels

## 6°) Registre à écriture série / lecture série

► C'est un type de registre dans lequel les données arrivent en série et sont transmises en série (sur une seule sortie)



► Il décale de 1 bit tous les bits d'entrée soit vers "la droite" (vers les bits de poids faibles), soit vers "la gauche" (vers les bits de poids forts).

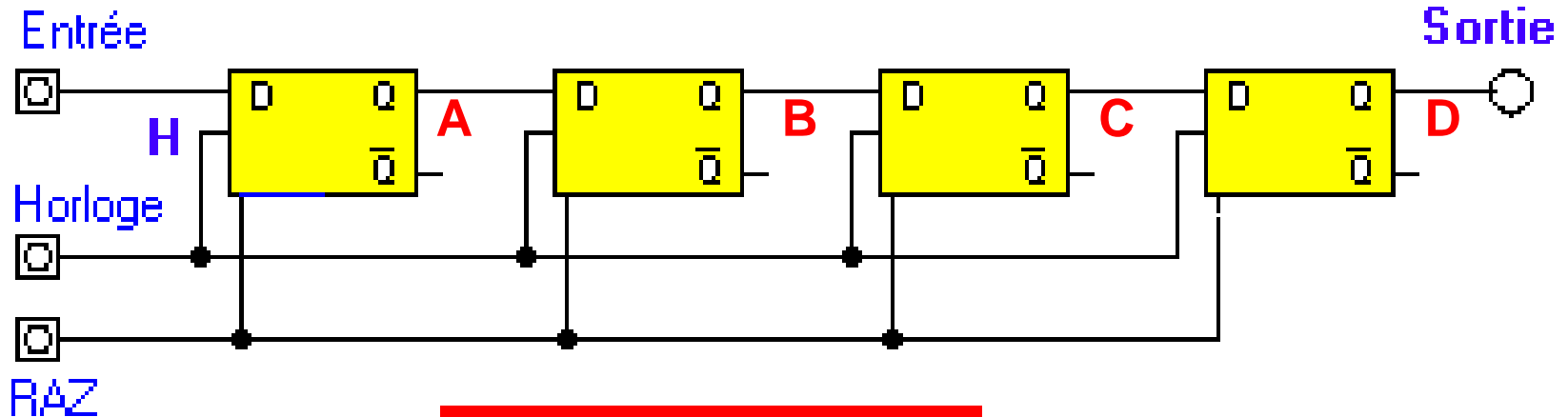
► Un registre à décalage dans un ordinateur correspond :

- soit à une **multiplication par 2** dans le cas du décalage à gauche;
- soit à une **division par 2** dans le cas du décalage à droite.



# Chapitre VI : Les circuits séquentiels

Exemple avec les bascules D :



Registre à décalage 4 bits

Entrée et sortie série

# Chapitre VI : Les circuits séquentiels

## Décalage

	D	C	B	A	
fig 1	0	0	0	0	← 1 0 1 1
fig 2	0	0	0	1	← 0 1 1
fig 3	0	0	1	0	← 1 1
fig.4	0	1	0	1	← 1
fig5	1	0	1	1	
fig 6	0	1	1	0	

Après 4 impulsions de l'horloge, les 4 bits sont stockés

# Chapitre VI : Les circuits séquentiels

## Exemple d'application :

- ▶ SISO : L'information que l'on veut introduire dans le registre est présentée à l'entrée de la première bascule.
- ▶ Lors d'une impulsion d'horloge, le bit d'information est introduit dans le registre, et tous les autres bits sont décalés.
- ▶ Le bit qui était mémorisé dans la dernière bascule est perdu s'il n'est pas stocké ou réinséré dans la structure d'une manière quelconque.
- ▶ Les registres SISO sont utilisés pour réaliser des **lignes à retard numériques**.
- ▶ Le délai entre l'entrée de l'information dans le registre et sa sortie dépend du nombre de bascules et de la fréquence d'horloge.

# Chapitre VI : Les circuits séquentiels

## 7°) Registres à décalage réversible (Shift registers)

► Le registre à décalage réversible est un registre à décalage où le décalage s'effectue vers la droite ou vers la gauche en fonction du niveau logique appliqué à l'entrée "Sens de décalage".

**Exemple :**

Décalage à droite et à gauche d'un mot de 4 bits

Mot de 4 bits :

Décalage à droite : (Right)

DR

x

1 1 0 1

Décalage à gauche : (Left)

1

0

1

x

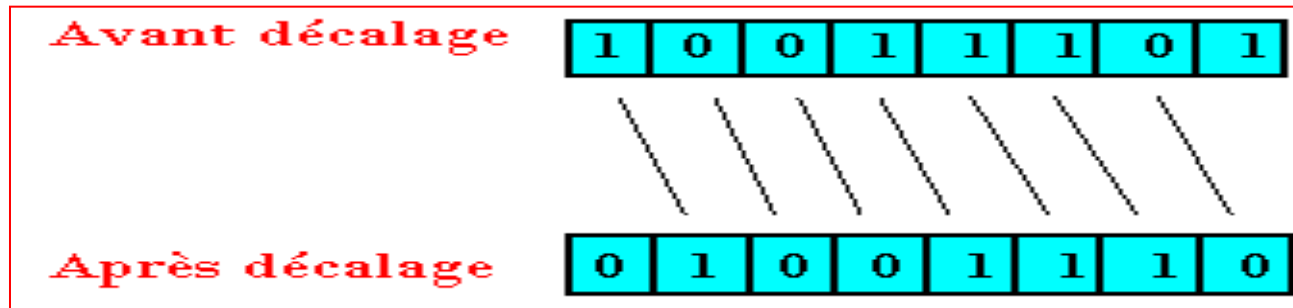
DL

► Un registre à décalage dans un ordinateur correspond soit à une **multiplication par 2** dans le cas du décalage à gauche, soit à une **division par 2** dans le cas du décalage à droite.

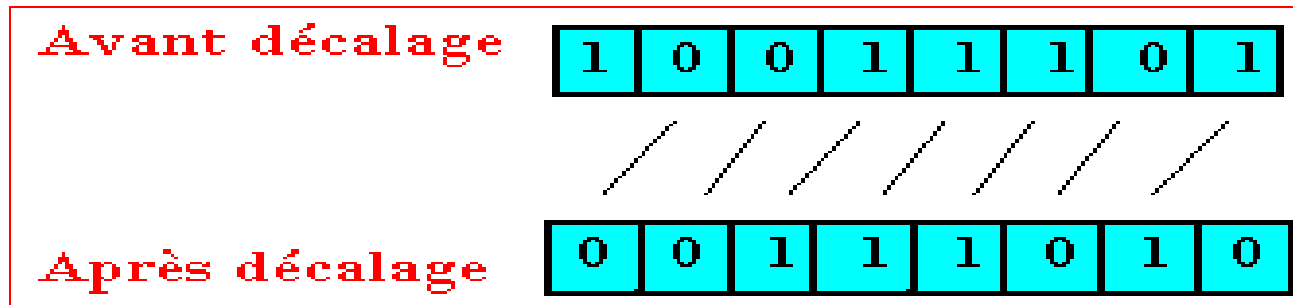
# Chapitre VI : Les circuits séquentiels

## Registre à décalage

- Le décalage à droite consiste à faire avancer l'information vers la droite

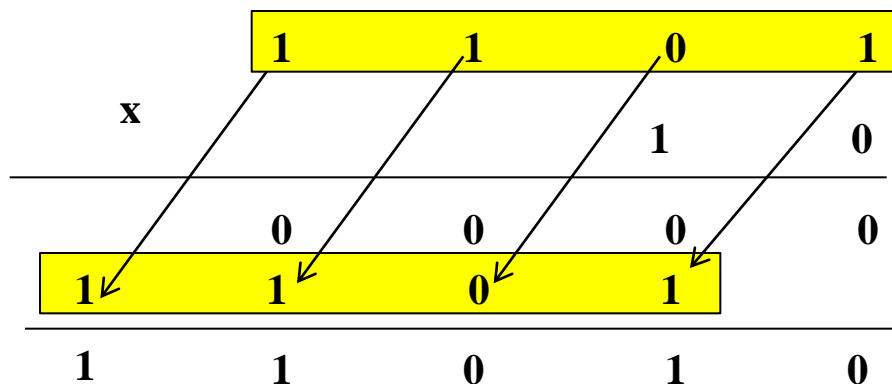


- Le décalage à gauche consiste à faire avancer l'information vers la gauche.

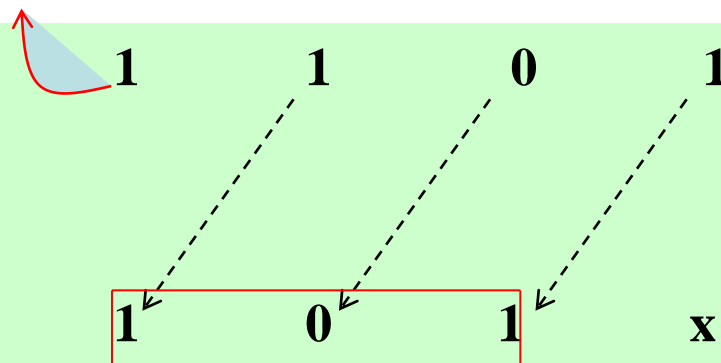


# Chapitre VI : Les circuits séquentiels

multiplication par 2 :



Mot de 4 bits :



Décalage à gauche : (Left)

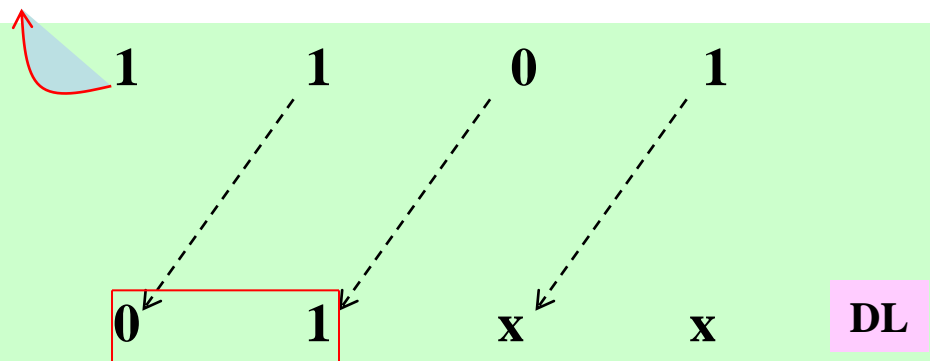
# Chapitre VI : Les circuits séquentiels

multiplication par 4 :

			1	1	0	1
				1	0	0
		0	0	0	0	0
	0	0	0	0	0	
1	1	0	1			
1	1	0	1	0	0	

Mot de 4 bits :

Décalage à gauche : (Left)



# Chapitre VI : Les circuits séquentiels

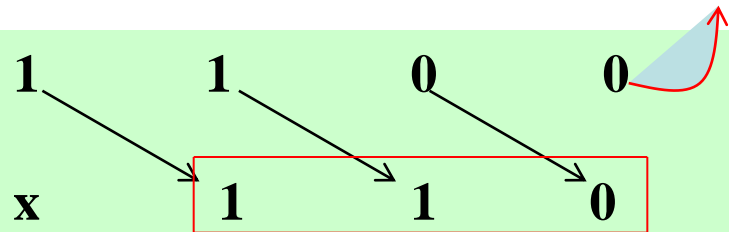
## Division par 2 :

1	1	0	0	1	0
1	0			1	1
	1	0			
	1	0			
	0	0	0		

## Mot de 4 bits :

## Décalage à droite : (Right)

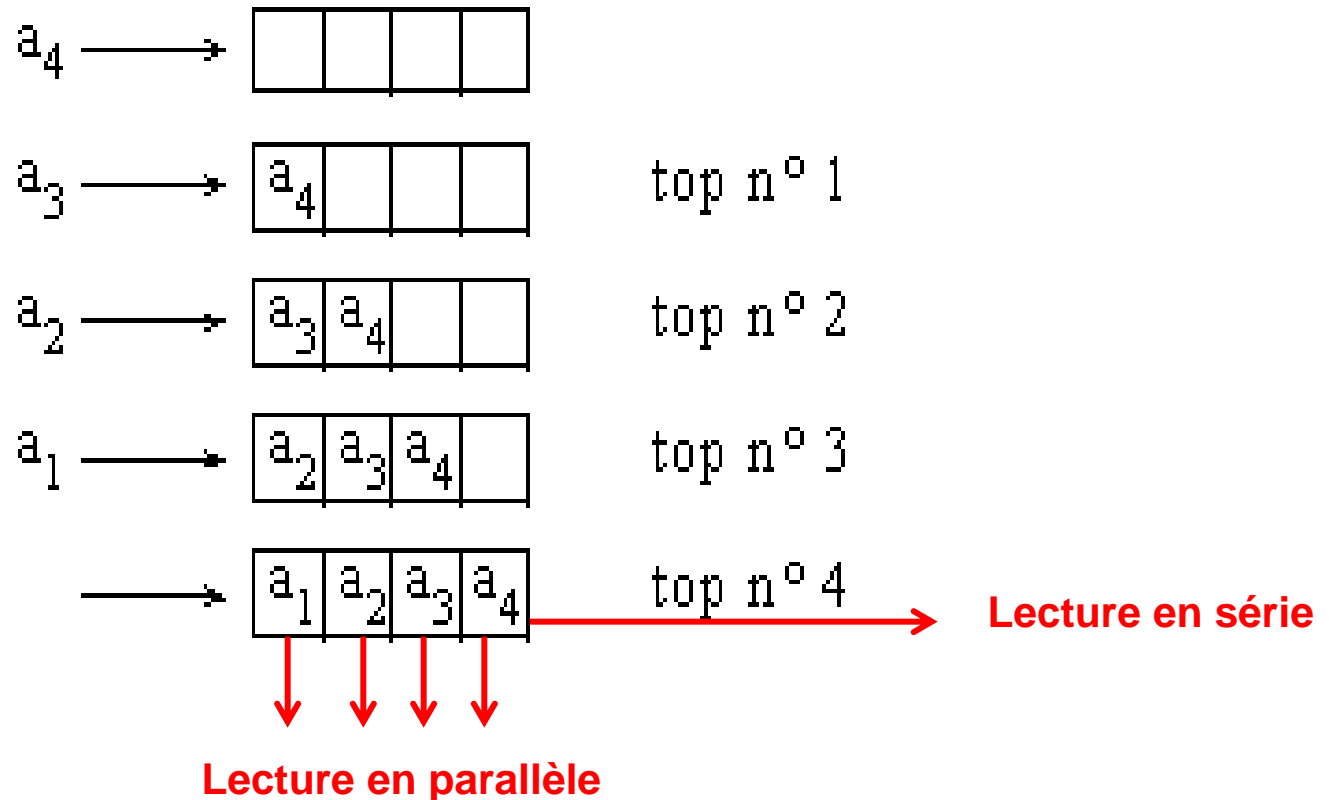
DR





# Chapitre VI : Les circuits séquentiels

Exemple de chargement d'un registre 4 bits en quatre coups d'horloge.

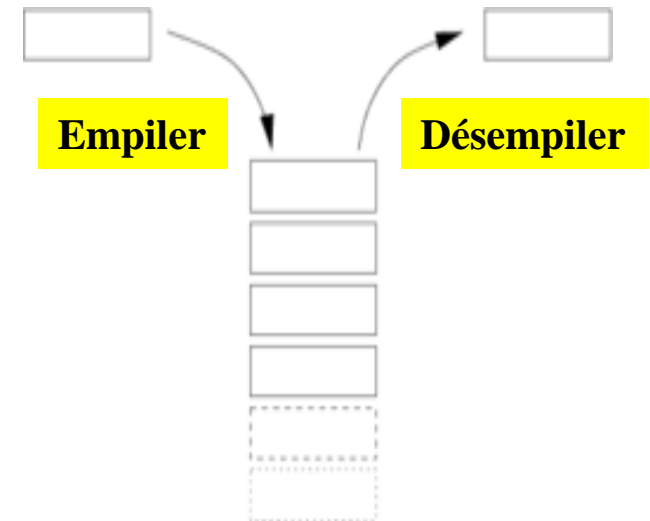


- De même l'information peut être lue en série ou en parallèle.
- D'autre part, certains registres peuvent être capables de décaler à gauche et à droite.

# Chapitre VI : Les circuits séquentiels

## Exemple d'application :

- ▶ Registres SISO réversibles : Ils permettent par exemple de réaliser ce que l'on appelle des piles LIFO (Last In, First Out) :
- ▶ On charge les bits dans le registre; puis on inverse le sens du décalage.
- ▶ Les bits apparaissent à la sortie de la première bascule dans l'ordre inverse de leur entrée.
- ▶ En informatique, une pile (stack) est une structure de données (structure logique destinée à contenir des données) fondée sur le principe :  
  
« dernier arrivé, premier sorti »  
(ou **LIFO** pour Last In, First Out),
- ▶ Ce qui veut dire, qu'en général, le dernier élément ajouté à la pile sera le premier à être utilisé.



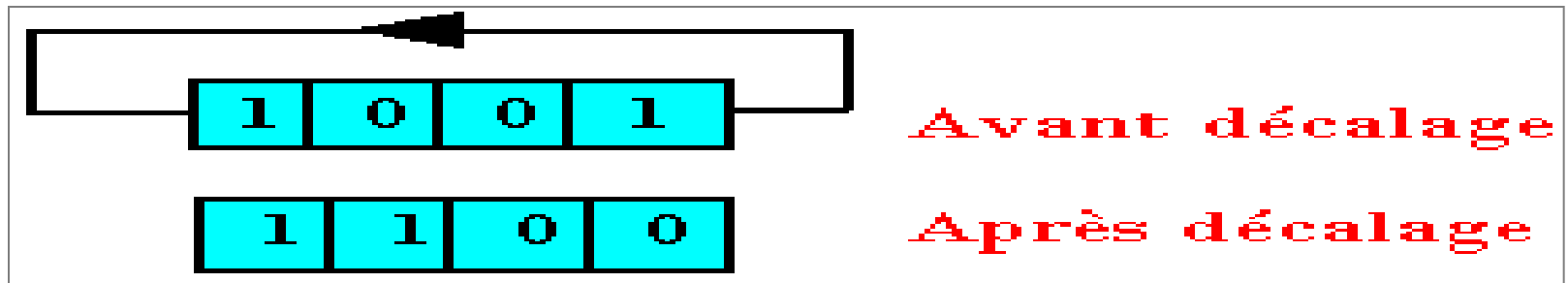
Une pile est gérée en *last in, first out*.

# Chapitre VI : Les circuits séquentiels

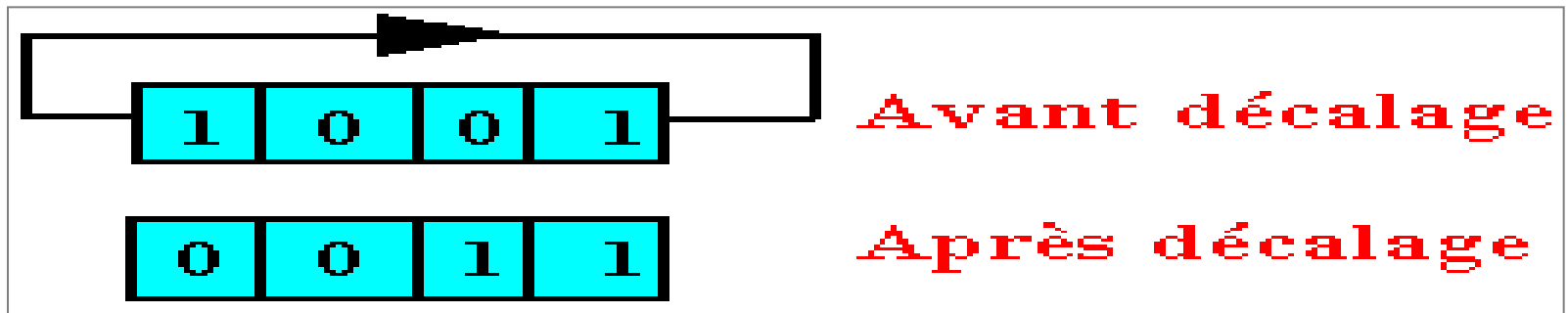
## 8°) Registre à décalage avec rétroaction linéaire

► Ce sont des registres à décalage en anneau ou registres à décalage circulaire

■ Décalage circulaire à droite:

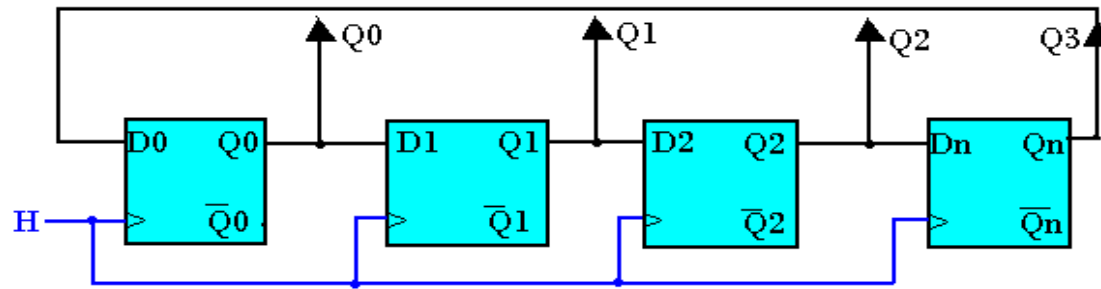


■ Décalage circulaire à gauche

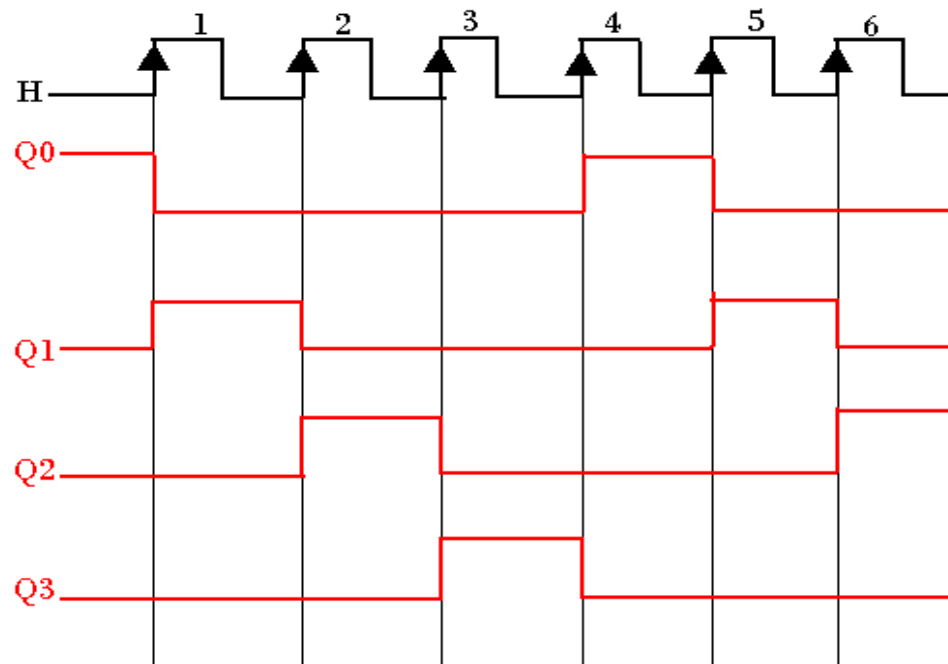


# Chapitre VI : Les circuits séquentiels

## Exemple d'un registre à décalage circulaire 4 bits à bascule D



Chronogramme:



H	Q <sub>0</sub>	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>
0	1	0	0	0
1 <sup>e</sup>	0	1	0	0
2 <sup>e</sup>	0	0	1	0
3 <sup>e</sup>	0	0	0	1
4 <sup>e</sup>	1	0	0	0
5 <sup>e</sup>	0	1	0	0
6 <sup>e</sup>	0	0	1	0

# Chapitre VI : Les circuits séquentiels

## Exemple Registre à décalage à rétroaction linéaire : LFSR

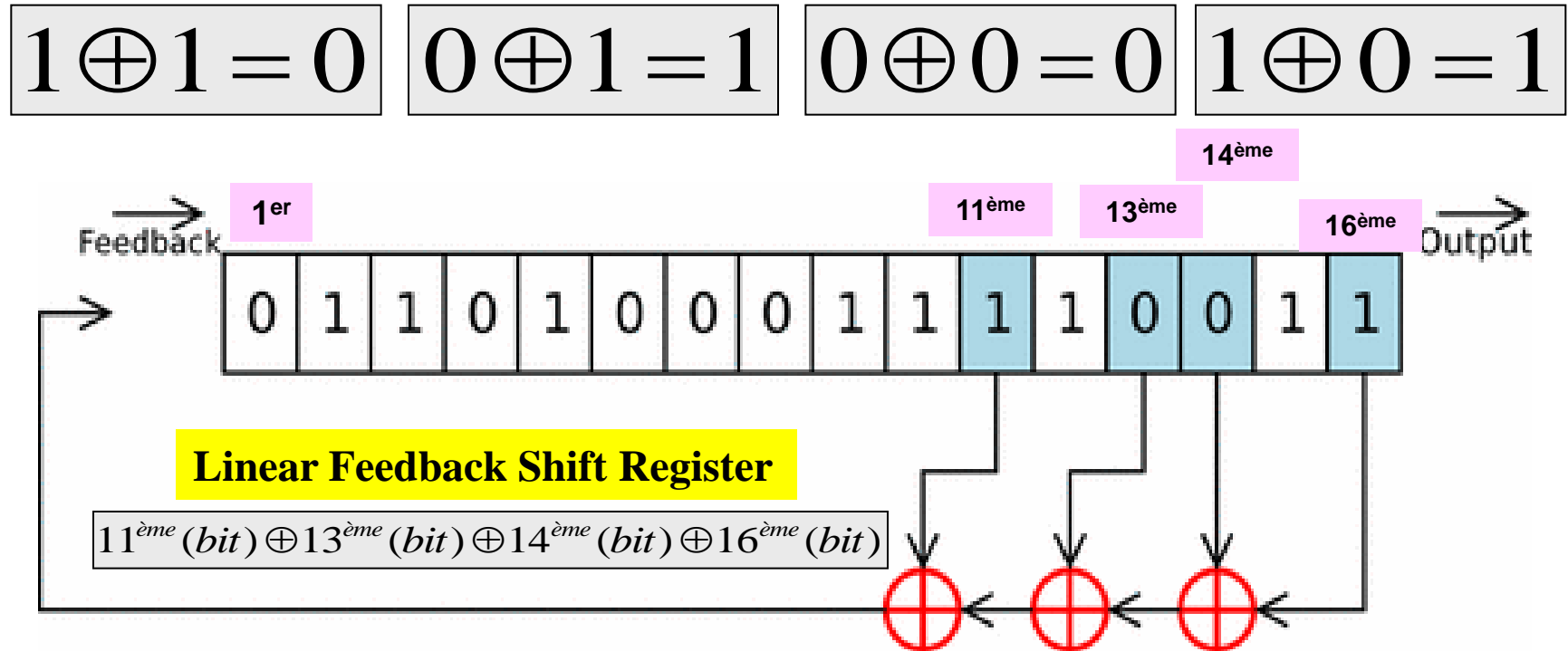


Schéma de fonctionnement d'un LFSR :

le bit de rétroaction est obtenue par **XOR** sur une sélection des bits du registre.

c'est un registre à décalage avec rétroaction linéaire, ce qui signifie que le bit entrant est le résultat d'un **OU exclusif** (ou XOR) entre plusieurs bits du registre

# Chapitre VI : Les circuits séquentiels

## Exemple d'application :

- ▶ Les LFSR sont utilisés en **cryptographie** pour engendrer des suites de nombres pseudo-aléatoires.
- ▶ La fonction de rétroaction est alors choisie de façon à obtenir une période la plus grande possible.
- ▶ L'étendue des applications est très large :
  - chiffrement des communications;
  - contrôle d'erreurs sur la transmission de données;
  - auto-test des composants électroniques...

# Chapitre VI : Les circuits séquentiels

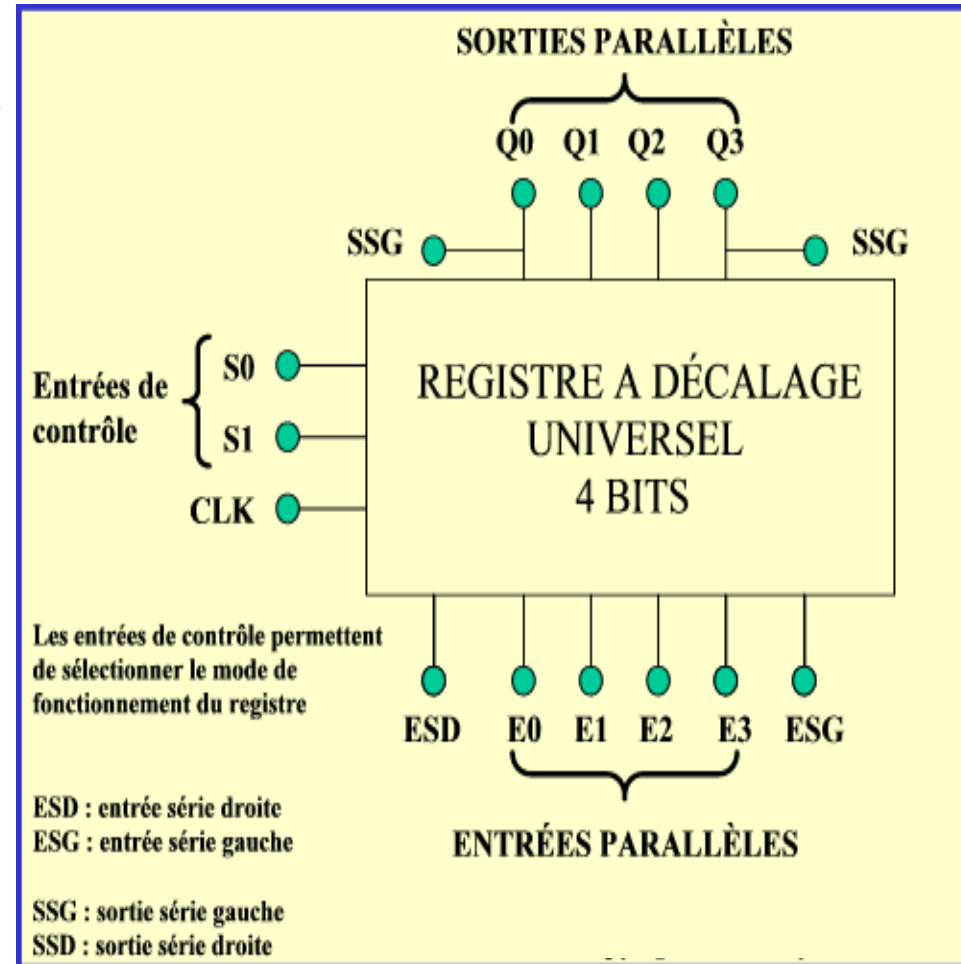
## 9°) Registres universels intégrés

### Organisation d'un registre à décalage universel de 4 bits

► Un registre à décalage peut posséder plusieurs modes de fonctionnement :

- chargement parallèle
- maintien des sorties (mémorisation)
- décalage à gauche
- décalage à droite

► Un registre peut posséder aussi une entrée de remise à zéro.



# Chapitre VI : Les circuits séquentiels

## Fonctionnement en mode chargement parallèle

- L'information de n bits est introduite d'un seul coup dans le registre.
- Cliquer sur le bouton "Chargement parallèle", pour observer le fonctionnement.
- Cliquer sur le bouton "État initial", pour revenir à la figure de départ.

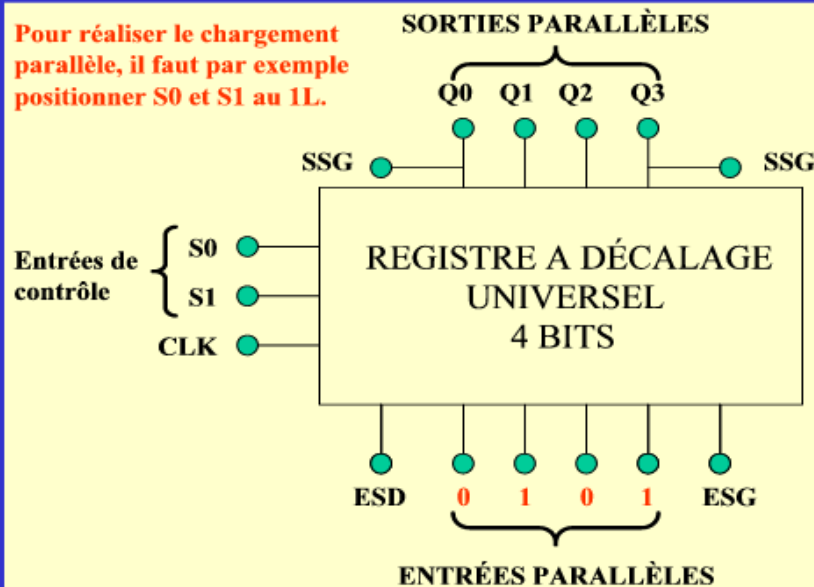
### Chargement parallèle

### État initial

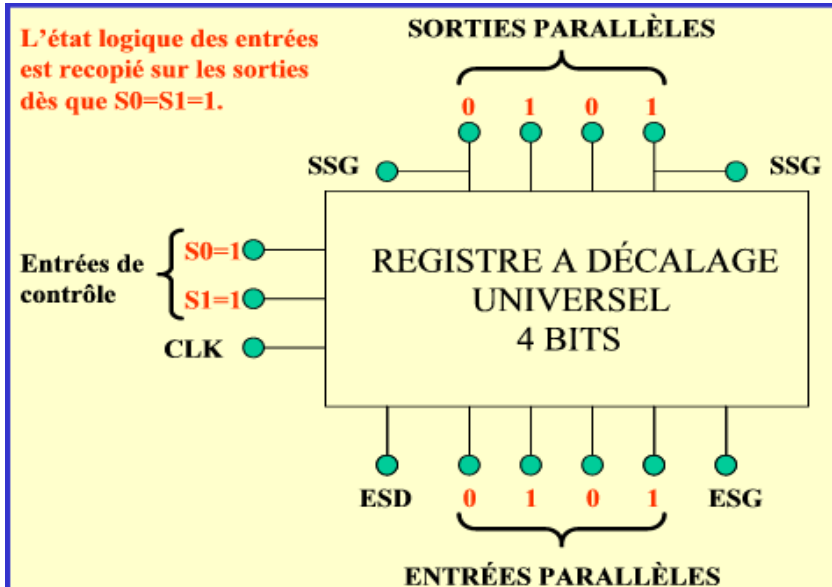
### Chargement parallèle

### État initial

Pour réaliser le chargement parallèle, il faut par exemple positionner S0 et S1 au 1L.



L'état logique des entrées est recopié sur les sorties dès que S0=S1=1.

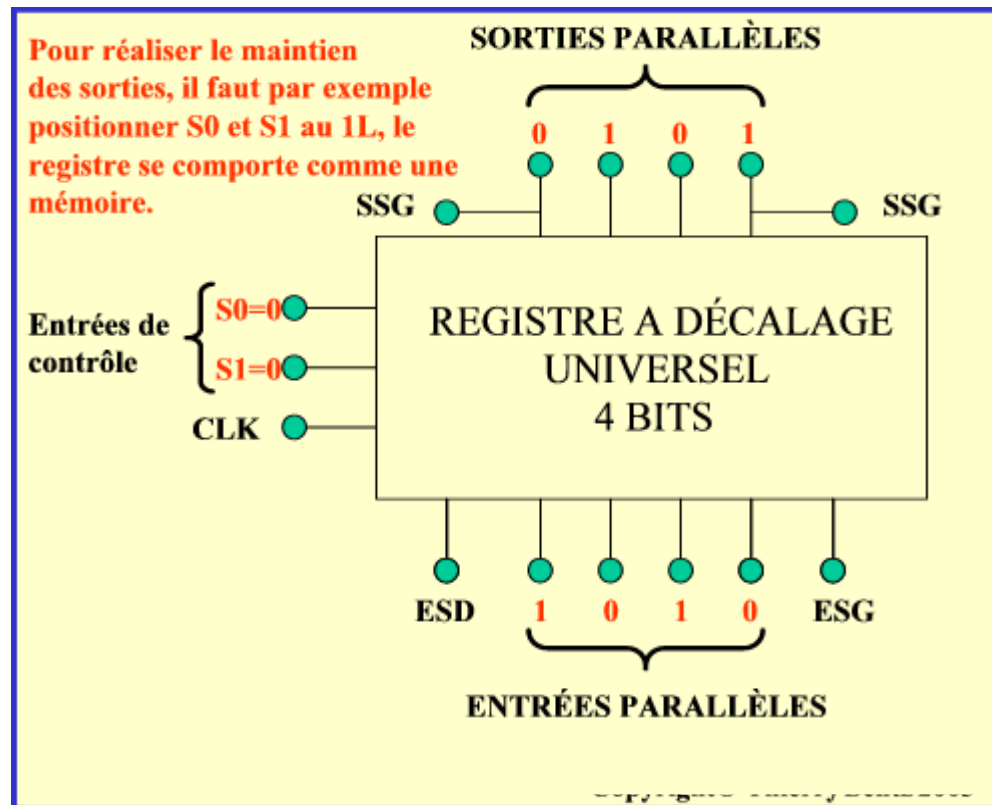




# Chapitre VI : Les circuits séquentiels

## Fonctionnement en mode maintien des sorties

- Le registre peut être utilisé en mémoire, l'évolution des entrées n'affecte plus les sorties.
- L'information mémorisée ici est celle du chargement parallèle précédent.



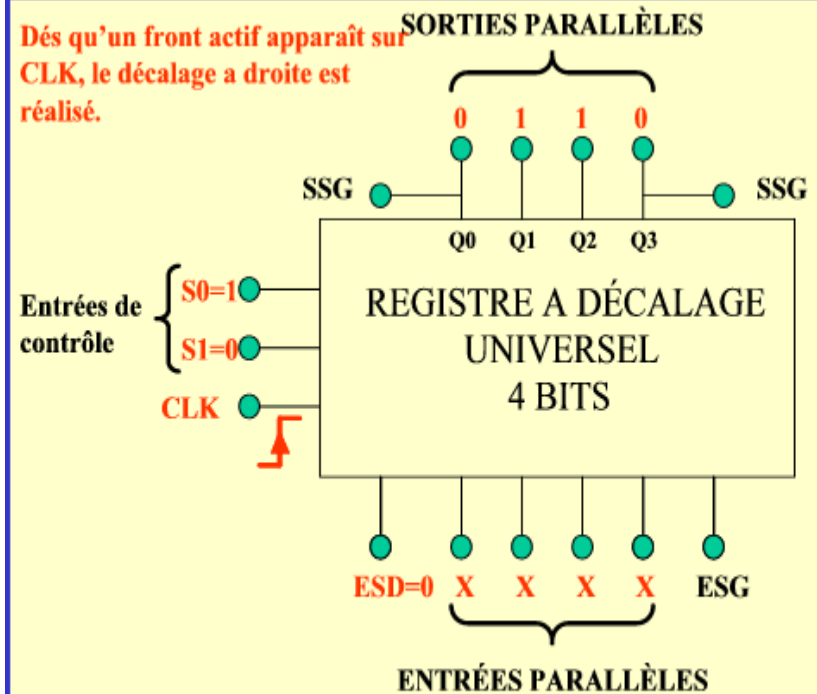
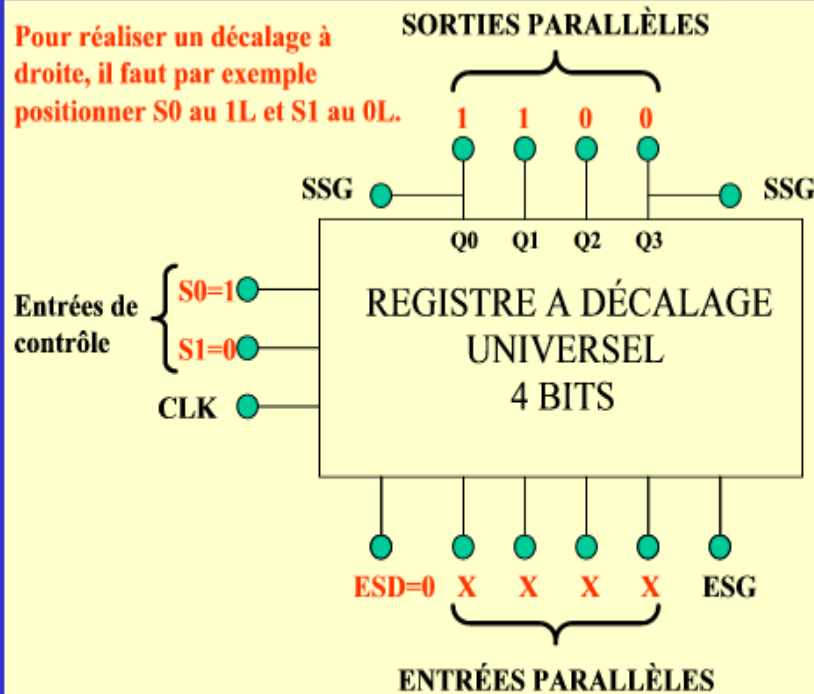
# Chapitre VI : Les circuits séquentiels

## Décalage à droite

## État initial

## Décalage à droite

## État initial



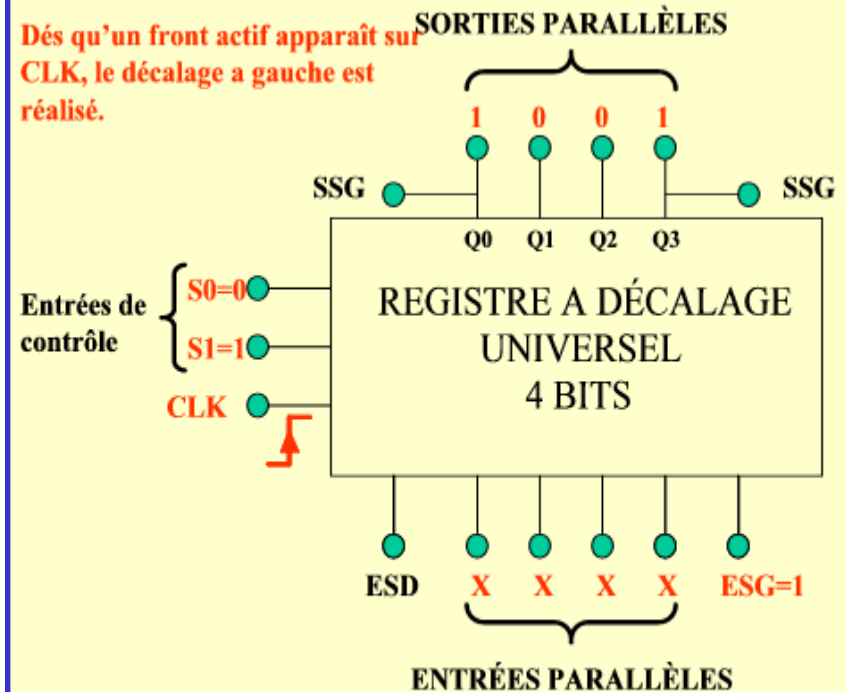
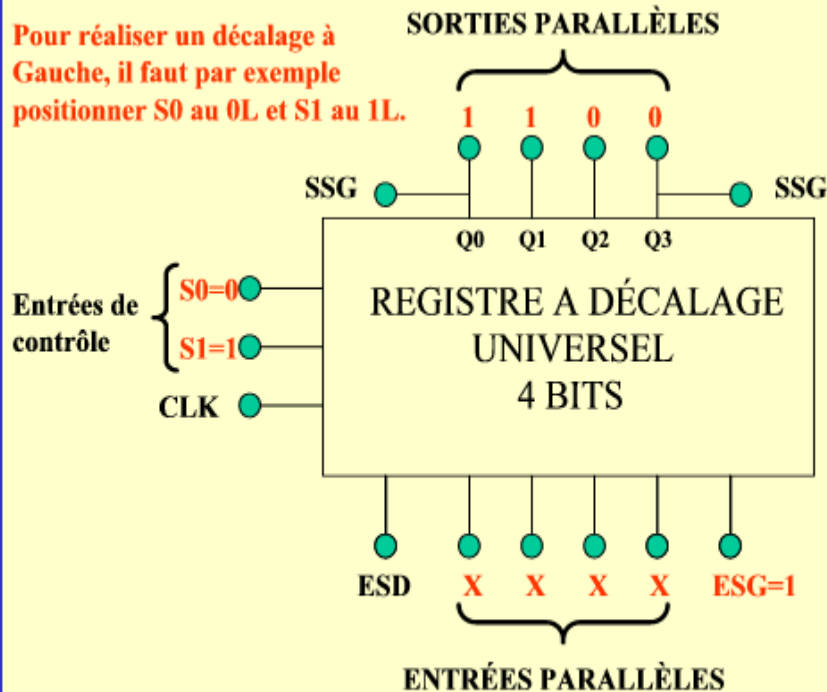
# Chapitre VI : Les circuits séquentiels

Décalage à gauche

État initial

Décalage à gauche

État initial



# Chapitre VI : Les circuits séquentiels

## 10°) Exemples d'application des registres à décalage

### Transmission série

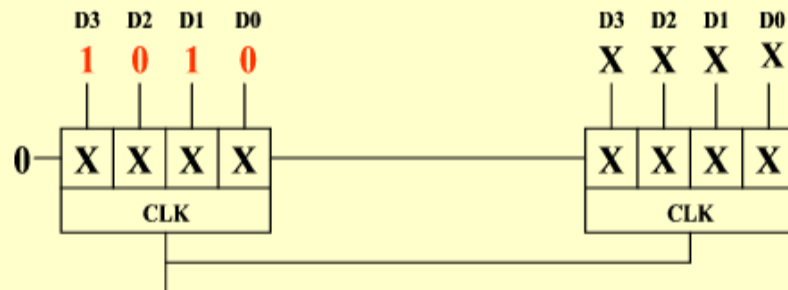
Autoriser la transmission

État initial

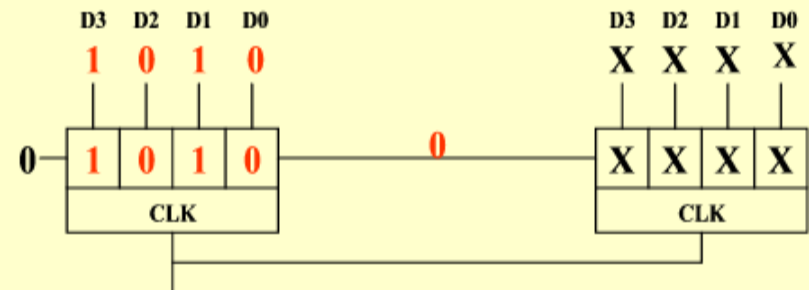
Autoriser la transmission

État initial

Application : liaison série



Application : liaison série



# Chapitre VI : Les circuits séquentiels

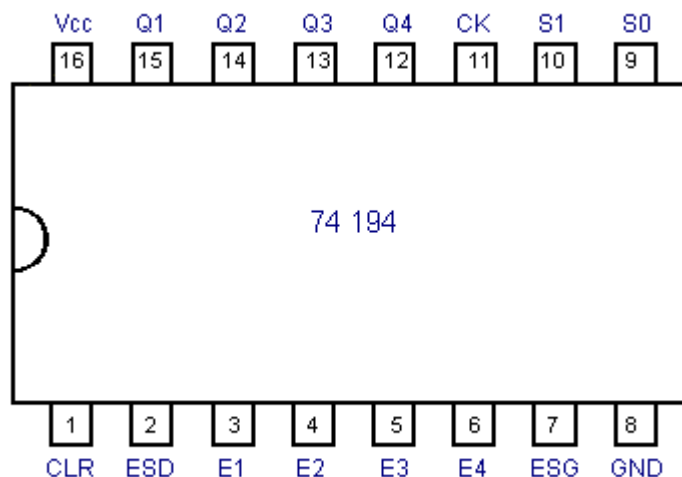
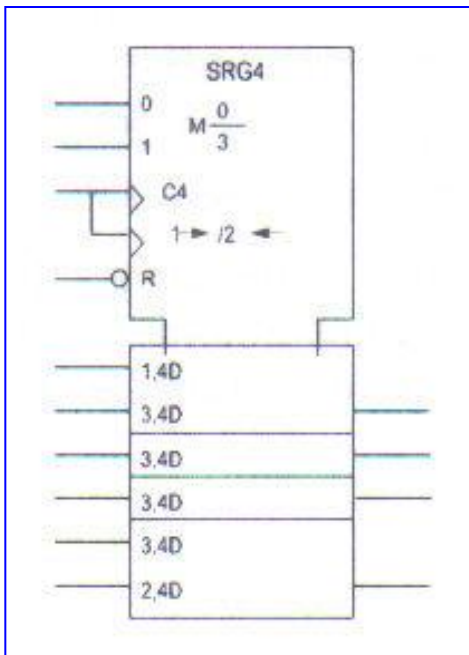
## 11°) Registres intégrés 74LS194

- Moyennant une interconnexion entre les bascules, un registre à décalage consiste à décaler bit par bit un mot binaire initialement stockée dans le registre. - Le décalage peut s'effectuer soit vers la droite (décalage vers la droite), soit vers la gauche (décalage vers la gauche).
- On utilise des bascules synchrones afin que le décalage se fasse en même temps pour chacune d'elles avec le même signal d'horloge.
- La méthode la plus simple pour réaliser un registre à décalage est d'utiliser des bascules D pour lesquelles, on trouve en sortie le signal fourni à l'entrée et ceci à chaque coup d'horloge.

# Chapitre VI : Les circuits séquentiels

## Registre à décalage universel (SRG4) : 74LS194

- Les applications du registre à décalage universel sont nombreuses et variées.
- L'entrée/sortie série peut être combinée à volonté avec l'entrée/sortie parallèle.
- Pour ce faire, le registre universel à décalage doit disposer de deux lignes d'adresses qui déterminent le mode de fonctionnement selon le tableau suivant :



Brochage du circuit intégré 74 194

Mode	S0	S1	Fonction
0	0	0	pas de fonction
1	1	0	entrée série, décalage à droite
2	0	1	entrée série, décalage à gauche
3	1	1	entrée parallèle

Symbole du composant.

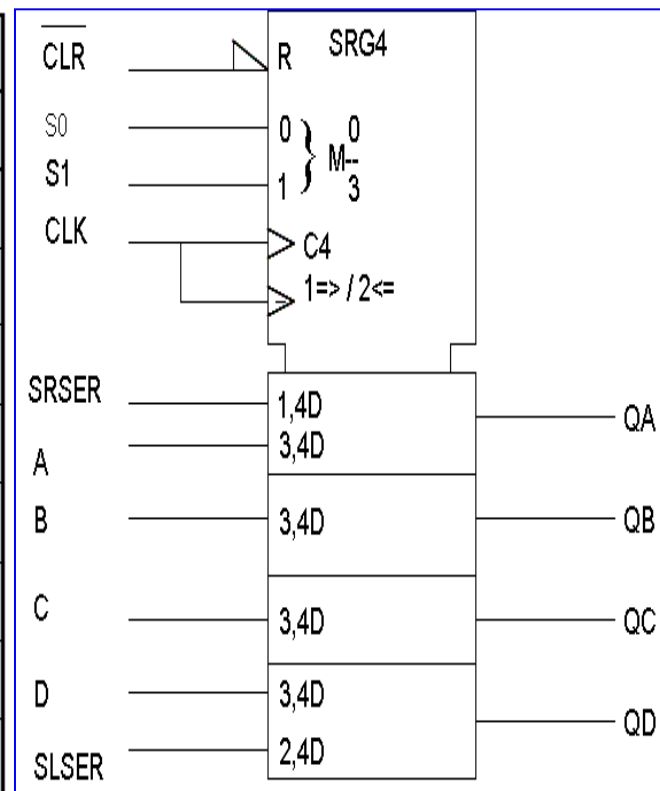
Table de vérité du 74 194

# Chapitre VI : Les circuits séquentiels

## Registre à décalage universel (SRG4) : 74LS194

ENTREES										SORTIES			
CLR	S1	S0	ESG	ESD	E1	E2	E3	E4	CK	Q1	Q2	Q3	Q4
0	X	X	X	X	X	X	X	X	X	0	0	0	0
1	X	X	X	X	X	X	X	X	0	PAS DE CHANGEMENT			
1	1	1	X	X	E1	E2	E3	E4	↑	E1	E2	E3	E4
1	0	1	X	1	X	X	X	X	↑	1	Q1n	Q2n	Q3n
1	0	1	X	0	X	X	X	X	↑	0	Q1n	Q2n	Q3n
1	1	0	1	X	X	X	X	X	↑	Q2n	Q3n	Q4n	1
1	1	0	0	X	X	X	X	X	↑	Q2n	Q3n	Q4n	0
1	0	0	X	X	X	X	X	X	X	PAS DE CHANGEMENT			

Table de vérité du circuit intégré 74 194



Symbole du composant.

# Chapitre VI : Les circuits séquentiels

## Fonctionnement :

- Toutes les bascules (flipflops) peuvent être réarmés par l'entrée de ré-enclenchement statique R.
- C4/1 → 2 ← : entrée d'horloge de décalage (gauche/droite) et entrée parallèle des données selon le mode choisi (0 .. 3).
- 1,4 D : entrée série pour décalage à droite.
- 2,4 D : entrée série pour décalage à gauche.
- 3,4 D : entrées pour décalage parallèle.

## Remarque :

Le premier chiffre de chaque entrée définit le mode de fonctionnement dont elle dépend.