

Systemes Numériques

TE302

Chapitre 2: Les Registres

Logique séquentielle



Introduction

• Définition

- Un registre de longueur N est constitué de N bascules synchronisées sur la même horloge,
- Le nombre de bascules détermine la taille du mot binaire à stocker: ici N bits, on parle aussi de profondeur du registre,
- Les registres sont des zones de mémorisation interne des microprocesseurs.

• Utilisation

- Mémoriser une information binaire,
- Transférer un mot binaire
 - Bit à bit, sur une même sortie, c'est-à-dire les N Bits en série,
 - Les N bits en même temps, sur N sorties, c'est-à-dire en parallèle (//),
- Décaler les bits, dans un sens ou dans l'autre,
- Par un re-bouclage, faire circuler le mot binaire (rotation).

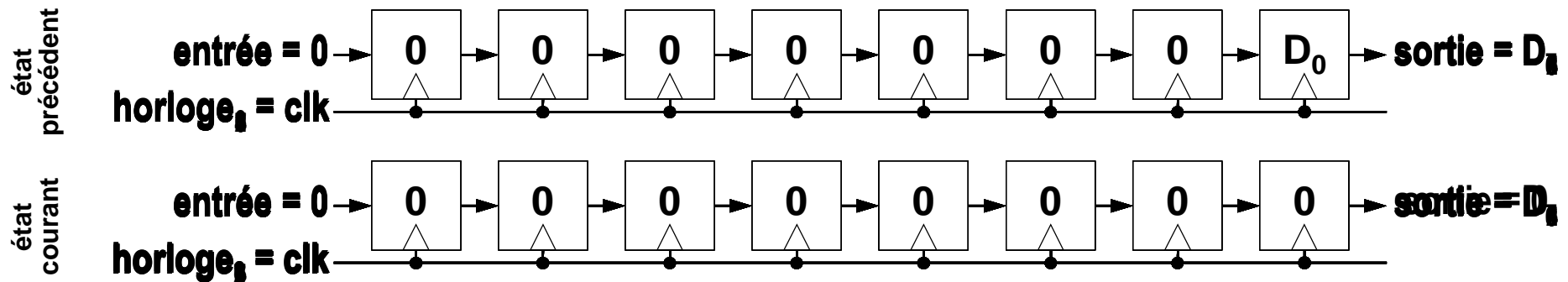
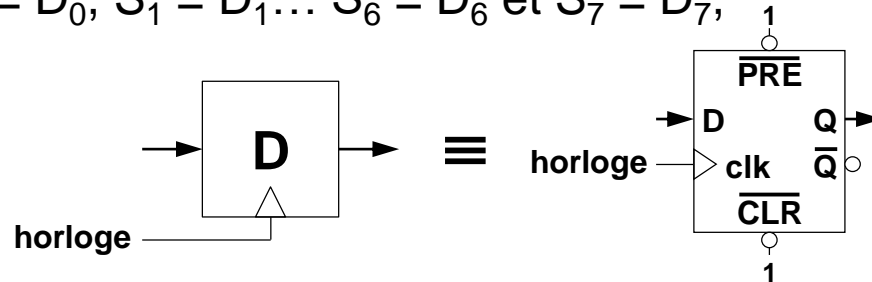
Décalage (1)

• Décalage à droite

- Dans ce cas:

- Tous les étages sont commandés par la même horloge et sur le même front: circuit synchrone sur front \uparrow ,
- L'entrée du registre se fait sur l'entrée de donnée e_0 de la bascule de gauche, cette entrée est, ici et arbitrairement, à 0,
- La sortie du registre se fait sur la sortie de donnée S_7 de la bascule de droite,
- Par construction: $e_i = S_{i-1}$, sauf pour e_0 ,
- Au départ (conditions initiales: horloge₀) les données sont présentes dans les bascules, ainsi: $S_0 = D_0$, $S_1 = D_1 \dots S_6 = D_6$ et $S_7 = D_7$,

• Attention:



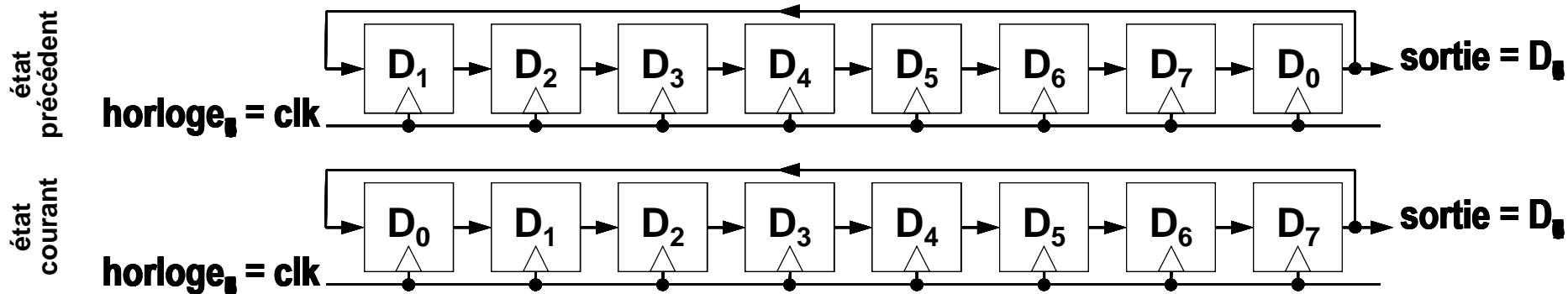
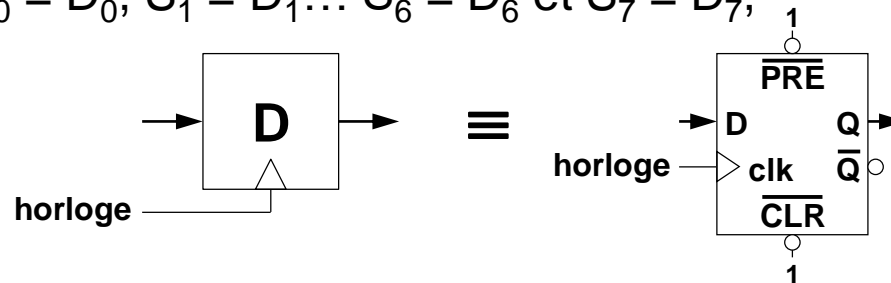
Décalage (2)

• Rotation à droite

- Dans ce cas:

- Circuit synchrone sur front \uparrow ,
- La sortie S_7 de la bascule de droite est bouclée sur l'entrée e_0 de la bascule de gauche,
- La sortie du registre se fait sur la sortie de donnée S_7 de la bascule de droite,
- Par construction: $e_i = S_{i-1}$, sauf pour $e_0 = S_7$,
- Au départ (conditions initiales: horloge₀) les données sont présentes dans les bascules, ainsi: $S_0 = D_0$, $S_1 = D_1 \dots S_6 = D_6$ et $S_7 = D_7$,

• Attention:



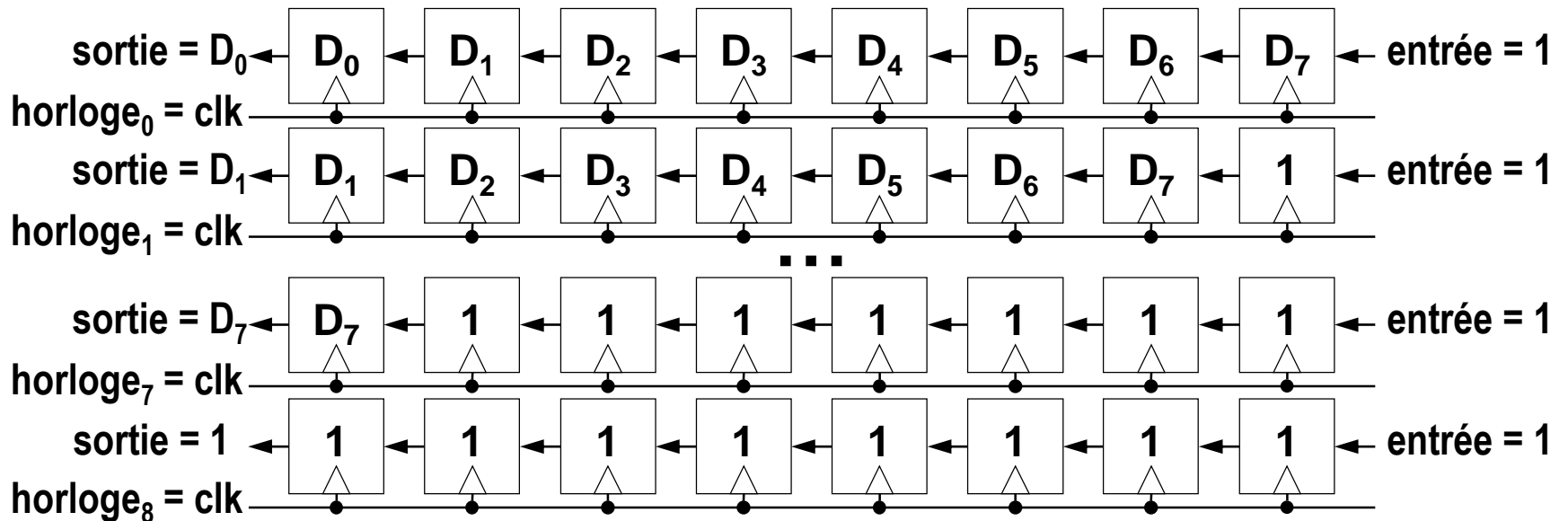
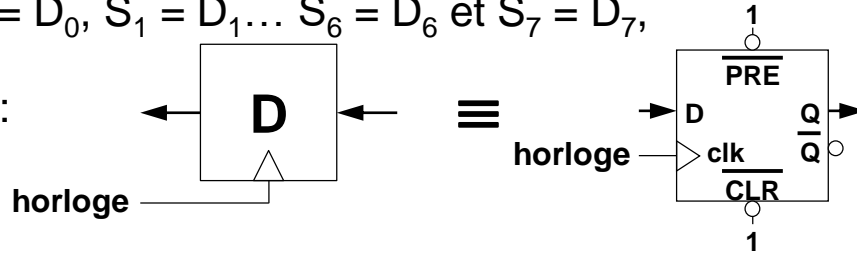
Décalage (3)

- Décalage à gauche

- Dans ce cas:

- Circuit synchrone sur front \uparrow ,
- L'entrée du registre se fait sur l'entrée de donnée e_7 de la bascule de droite, cette entrée est, ici et arbitrairement, à 1,
- La sortie du registre se fait sur la sortie de donnée S_0 de la bascule de gauche,
- Par construction: $e_i = S_{i+1}$, sauf pour e_7 ,
- Au départ (conditions initiales: horloge₀) les données sont présentes dans les bascules, ainsi: $S_0 = D_0$, $S_1 = D_1 \dots S_6 = D_6$ et $S_7 = D_7$,

- Attention:



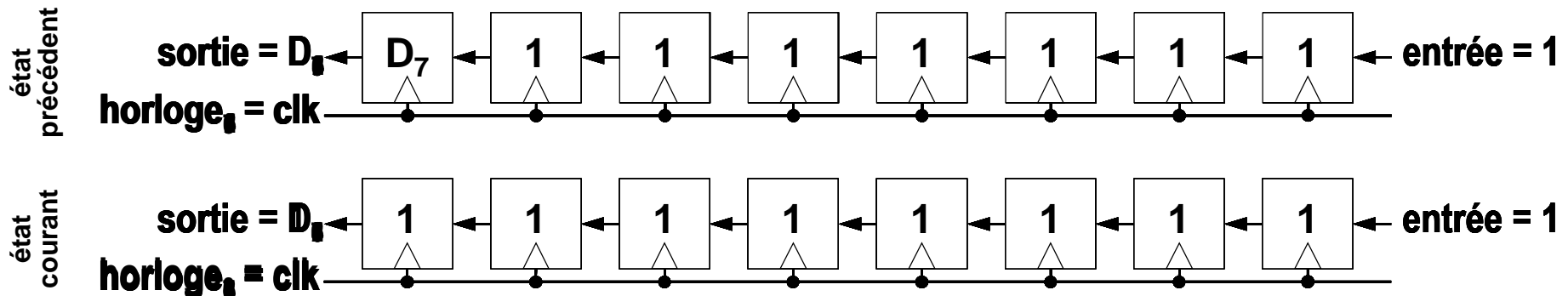
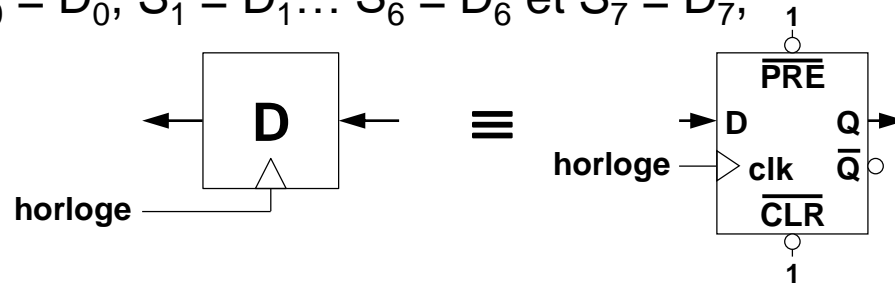
Décalage (3)

• Décalage à gauche

-Dans ce cas:

- Circuit synchrone sur front \uparrow ,
- L'entrée du registre se fait sur l'entrée de donnée e_7 de la bascule de droite, cette entrée est, ici et arbitrairement, à 1,
- La sortie du registre se fait sur la sortie de donnée S_0 de la bascule de gauche,
- Par construction: $e_i = S_{i+1}$, sauf pour e_7 ,
- Au départ (conditions initiales: horloge₀) les données sont présentes dans les bascules, ainsi: $S_0 = D_0$, $S_1 = D_1 \dots S_6 = D_6$ et $S_7 = D_7$,

• Attention:



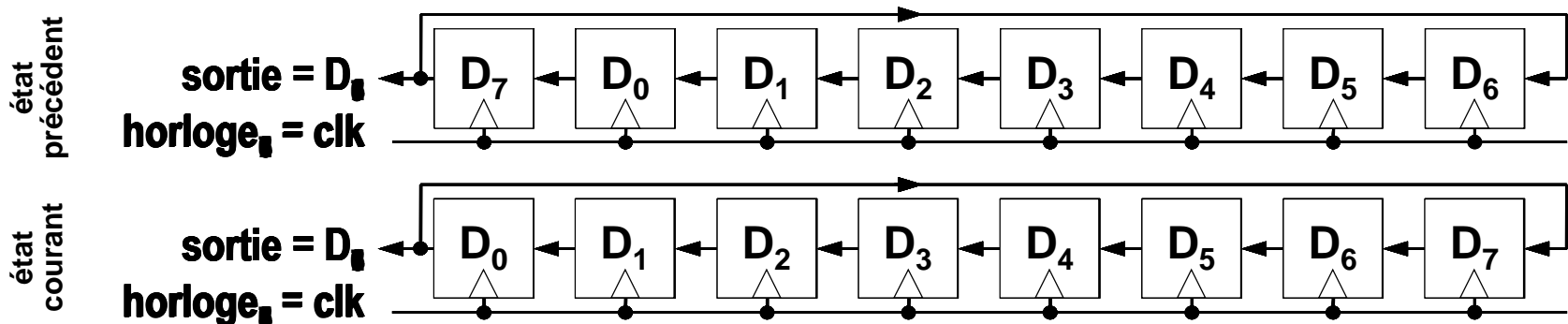
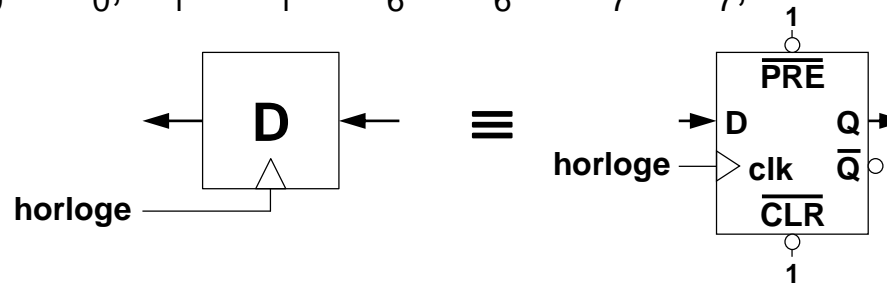
Décalage (4)

• Rotation à gauche

- Dans ce cas:

- Circuit synchrone sur front \uparrow ,
- La sortie S_0 de la bascule de gauche est bouclée sur l'entrée e_7 de la bascule de droite ,
- La sortie du registre se fait sur la sortie de donnée S_0 de la bascule de gauche,
- Par construction: $e_i = S_{i+1}$, sauf pour $e_7 = S_0$,
- Au départ (conditions initiales: horloge₀) les données sont présentes dans les bascules, ainsi: $S_0 = D_0$, $S_1 = D_1 \dots S_6 = D_6$ et $S_7 = D_7$,

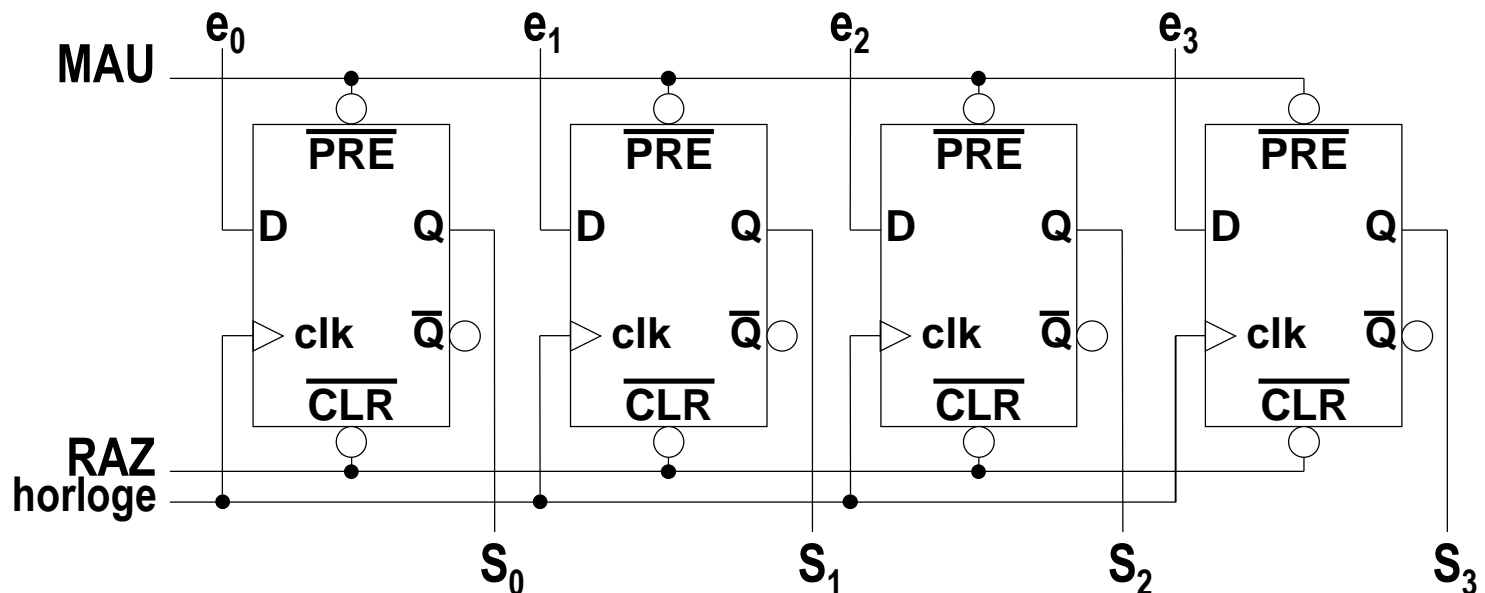
• Attention:



Registres parallèles et séries (1)

- **Entrées parallèles – sorties parallèles**

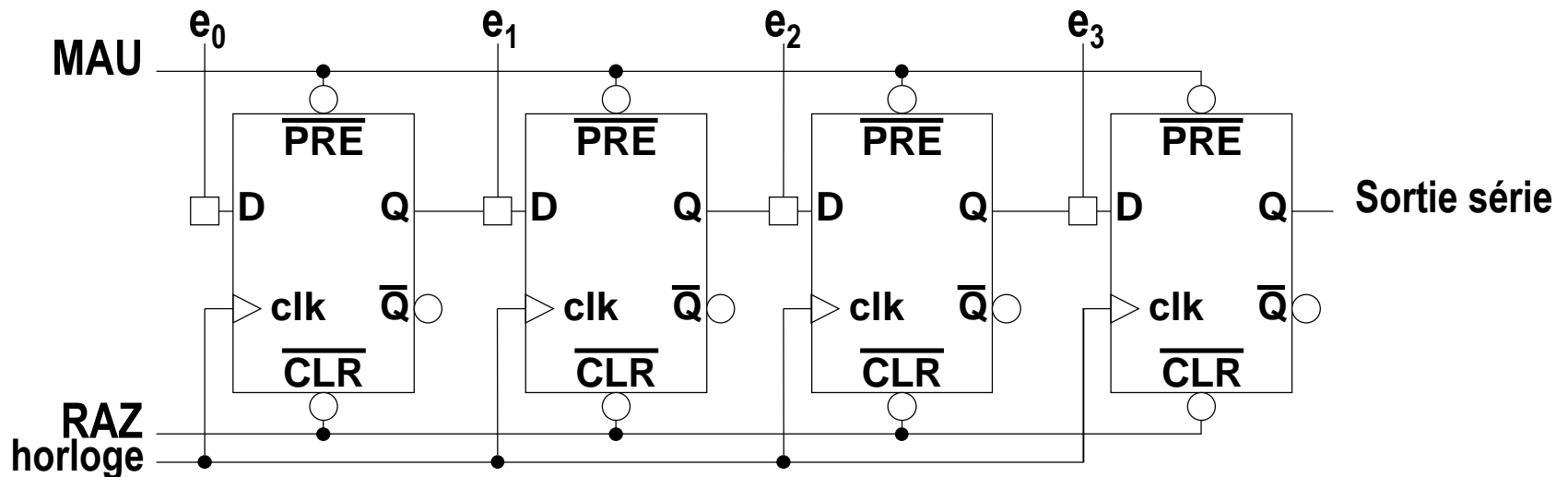
- Circuit synchrone sur front \uparrow ,
- Les entrées de données e_i respectant les conditions d'utilisation: t_{pp} , t_m (leçon 2, diapositive 10),
- Sont copiées en sorties S_i sur le même front d'horloge après le temps t_p (leçon 2, diapositive 10),
- MAU et RAZ sont des forçages prioritaires asynchrones appliqués sur les CLR_i et les PRE_i , respectivement,
- Exemple: bascules D sur fronts montants.



Registres parallèles et séries (2)

• Entrées parallèles (//) – sortie série (1)

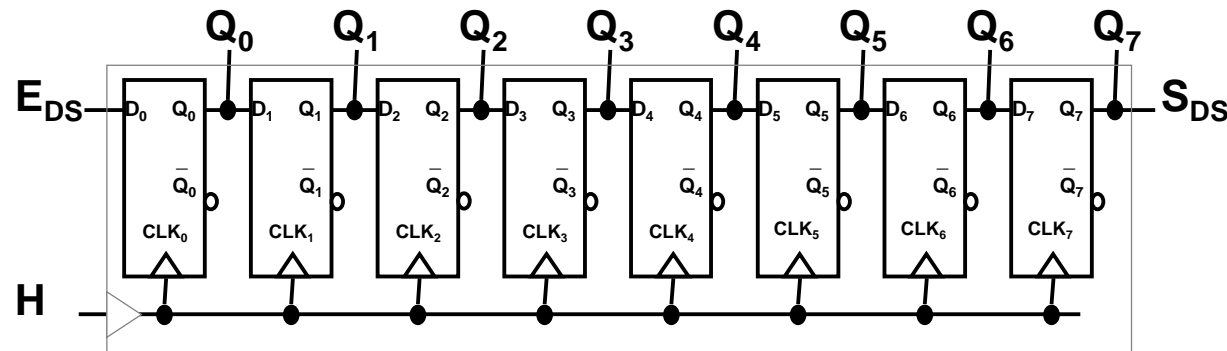
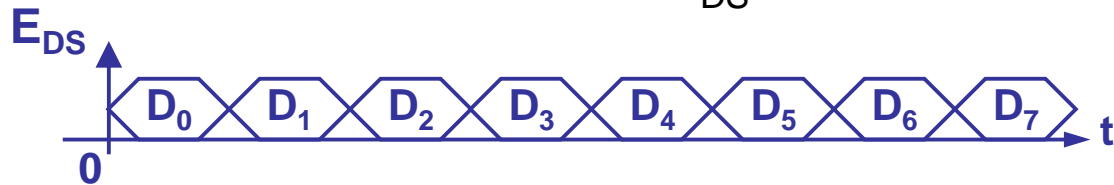
- Les entrées de données e_i respectant les conditions d'utilisation: t_{pp} , t_m (leçon 2, diapositive 10), sont copiées en sorties S_i sur le même front d'horloge après le temps t_p (Δt , leçon 2, diapositive 10),
- En mode chargement $D_i = e_i$: ☐,
- En mode conversion // série (décalage): $D_i = Q_{i-1}$: ☐,
- Sortie série du registre sur Q_3 ,
- RAZ et MAU sont des forçages prioritaires asynchrones appliqués sur les \overline{CLR}_i et les \overline{PRE}_i , respectivement,
- Exemple: bascules D sur fronts montants.



Registres parallèles et séries (3)

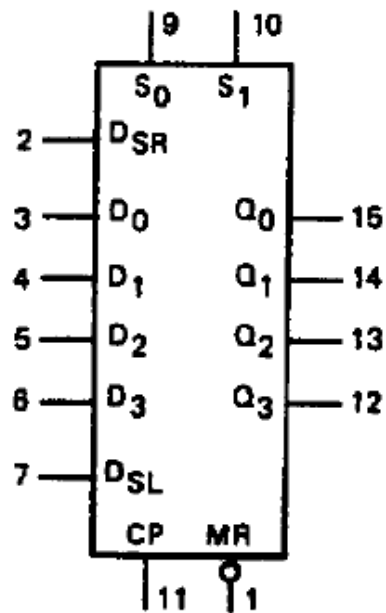
- Entrées série – sorties parallèles

- Registre à décalage de 8 bits (8 bascules),
- 8 bits se présentent sur l'entrée de données série E_{DS} ,
- Au bout de 8 impulsions d'horloge ces 8 bits sont disponibles sur les 8 sorties Q_0 à Q_7 ,
- Exemple, CI: $Q_i = 0$,
- Trame de données E_{DS} :



H	Q_0	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7
CI	0	0	0	0	0	0	0	0
1	D_0	0	0	0	0	0	0	0
2	D_1	D_0	0	0	0	0	0	0
3	D_2	D_1	D_0	0	0	0	0	0
4	D_3	D_2	D_1	D_0	0	0	0	0
5	D_4	D_3	D_2	D_1	D_0	0	0	0
6	D_5	D_4	D_3	D_2	D_1	D_0	0	0
7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	0
8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0

Registre universel intégré: 74 HC 194



Logic symbol

FUNCTION TABLE

OPERATING MODES	INPUTS							OUTPUTS			
	CP	$\overline{\text{MR}}$	S ₁	S ₀	D _{SR}	D _{SL}	D _n	Q ₀	Q ₁	Q ₂	Q ₃
reset (clear)	X	L	X	X	X	X	X	L	L	L	L
hold ("do nothing")	X	H	l	l	X	X	X	q ₀	q ₁	q ₂	q ₃
shift left	↑	H	h	l	X	l	X	q ₁	q ₂	q ₃	L
	↑	H	h	l	X	h	X	q ₁	q ₂	q ₃	H
shift right	↑	H	l	h	l	X	X	L	q ₀	q ₁	q ₂
	↑	H	l	h	h	X	X	H	q ₀	q ₁	q ₂
parallel load	↑	H	h	h	X	X	d _n	d ₀	d ₁	d ₂	d ₃

Notes

- H = HIGH voltage level
 h = HIGH voltage level one set-up time prior to the LOW-to-HIGH CP transition
 L = LOW voltage level
 l = LOW voltage level one set-up time prior to the LOW-to-HIGH CP transition
 q,d = lower case letters indicate the state of the referenced input (or output) one set-up time prior to the LOW-to-HIGH CP transition
 X = don't care
 ↑ = LOW-to-HIGH CP transition

CP: Clock Pulse

MR: Master Reset (actif sur 0 = 'L')

L'entrée série sur décalage à droite se fait sur D_{SR}, «Data Serial Right»,

L'entrée série sur décalage à gauche se fait sur D_{SL}, «Data Serial Left»,

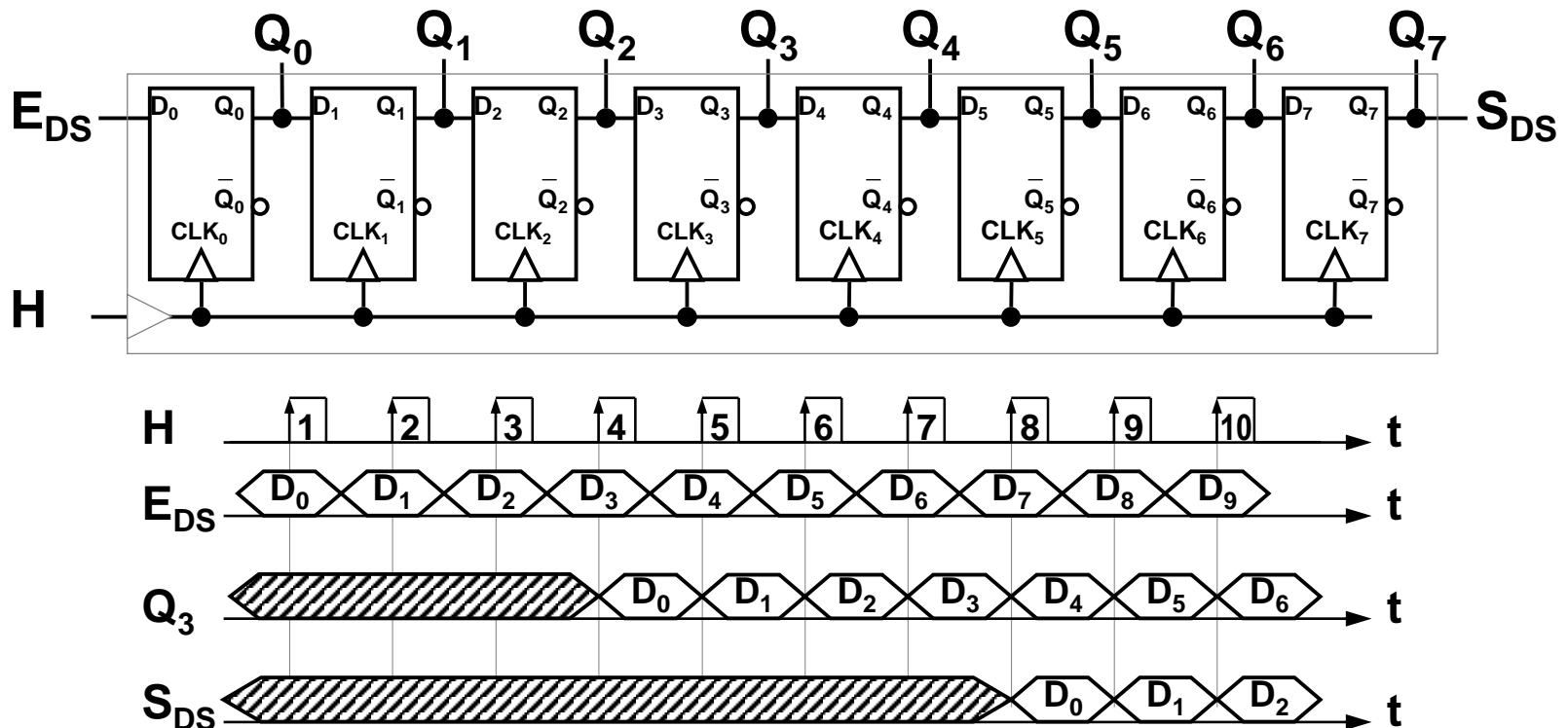
La sortie série sur décalage à droite se fait sur Q₃.

La sortie série sur décalage à gauche se fait sur Q₀.

Applications (1)

- Retard logique

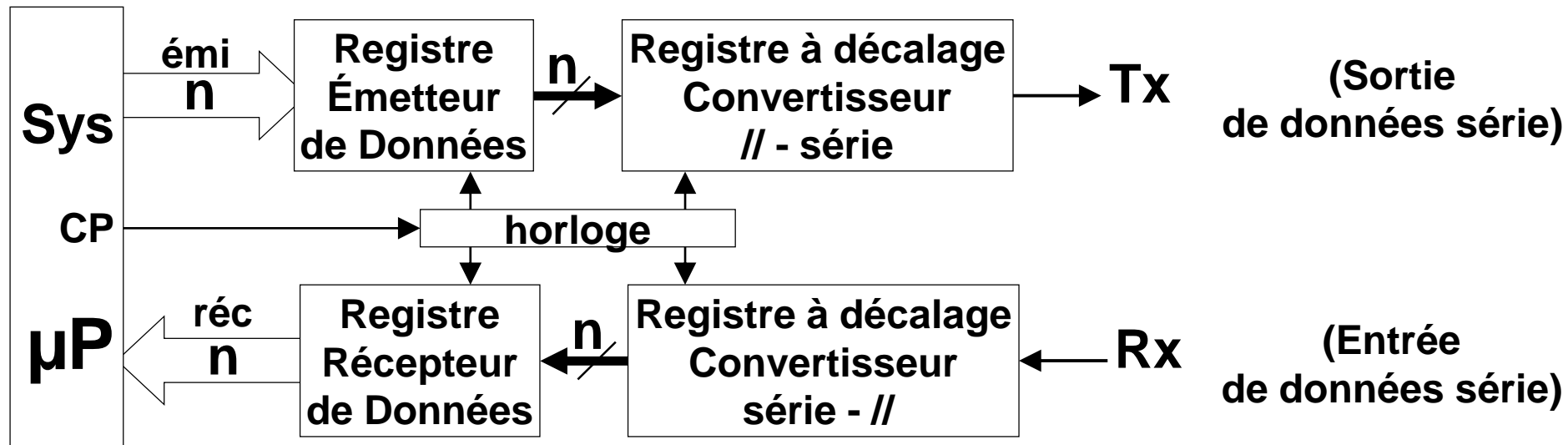
- Utilisation d'un registre à décalage (ici: de 8 bits, donc 8 bascules) à entrée et sortie de données série,
- D'une horloge (ici: $f_H = 1\text{MHz}$, $T_H = 1\mu\text{s}$),
- Les données entrantes sont retardées de $n \cdot T_H$, avec n le n° de la sortie. Exemple, ici: Q_3 et Q_7 .



Applications (2): UART (1)

Universal Asynchronous Receiver Transmitter

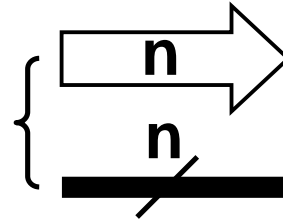
- **Émetteur récepteur asynchrone universel**
 - Ce composant est utilisé pour faire la liaison entre:
 - un système à μ processeur, travaillant sur n bits (n fils en // constituant un **bus de données**)
 - et le **port** ou **liaison série**, travaillant sur 1 bit (1 fil),
 - Le but est de limiter le nombre de fils de transmission,
 - L'UART est ainsi un convertisseur de données // en données série et réciproquement.



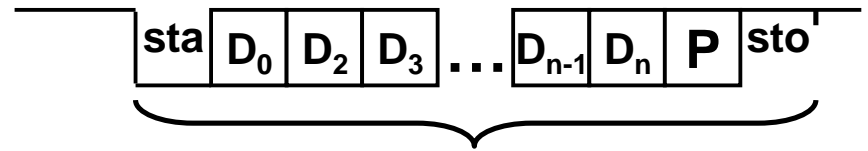
Application (3): UART (2)

- Définitions

- Bus n bits (//)



- Trame UART (**série**)



- Données binaires série: **D_0 à D_n**
 - Bit de parité: **P**
 - Bit de départ (start) = **0**: **sta**
 - Bit de fin (stop) = **1**: **sto**