

UNIVERSITÉ DE TECHNOLOGIE DE BELFORT-MONTBÉLIARD

Mémoires

Circuits numériques séquentiels

SY41

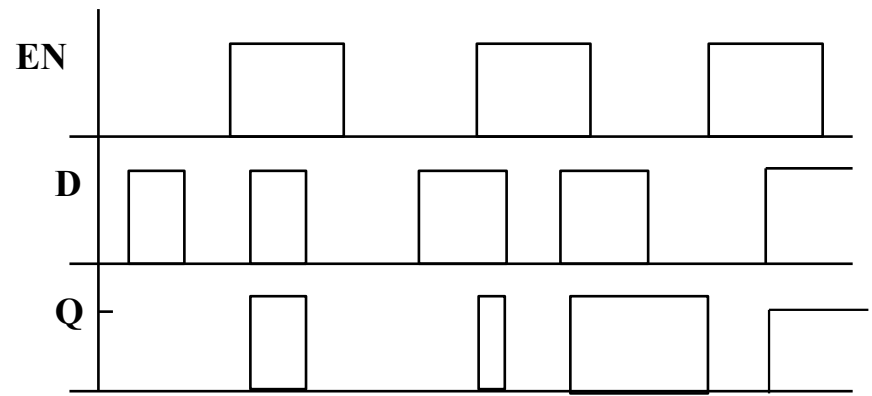
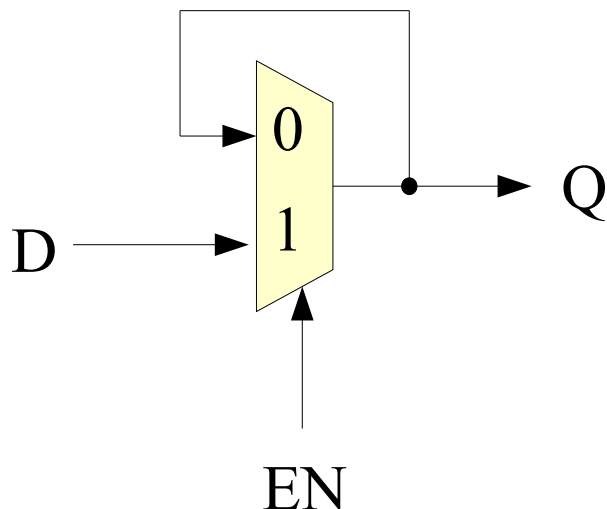
Nicolas Lacaille
Nicolas.lacaille@utbm.fr

Élément de mémorisation

- ◆ Maintien d'une valeur binaire dans un état stable
 - '0' ou '1'
- ◆ Bascule : élément de mémorisation pouvant maintenir un état
- ◆ Bascule : circuit logique
 - Entrée haute impédance
 - Sortie imposant un niveau logique haut ou bas correspondant à l'état stable
 - La sortie ne dépend pas uniquement des combinaisons d'entrée

Latch ou bascule D verrou

- ◆ Deux cas
 - $EN = 1$, le circuit est transparent $Q = D$
 - $EN = 0$, la dernière valeur présente sur Q est verrouillée : maintien par bouclage
- ◆ Utilisation de chronogrammes

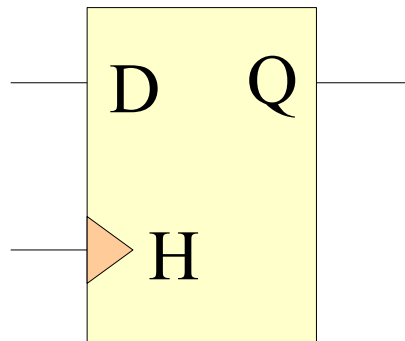


Bascule sur front

- ◆ Acquisition de valeur à un instant t , déterminé par un "front" de synchronisation (edge triggered).
 - Changement d'état niveau bas vers niveau haut : front montant
 - Changement d'état niveau haut vers niveau bas : front descendant
- ◆ Entre deux instants d'acquisition la valeur est mémorisée et son niveau logique est maintenu sur la sortie

Bascule D sur front (D flip flop)

◆ Représentation et table de vérité

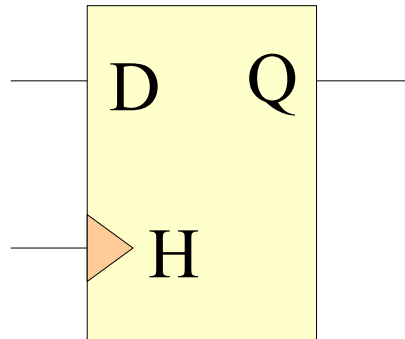


H	D	Q
▲ ↑	0	0
	1	1
Hors front	X	Q-

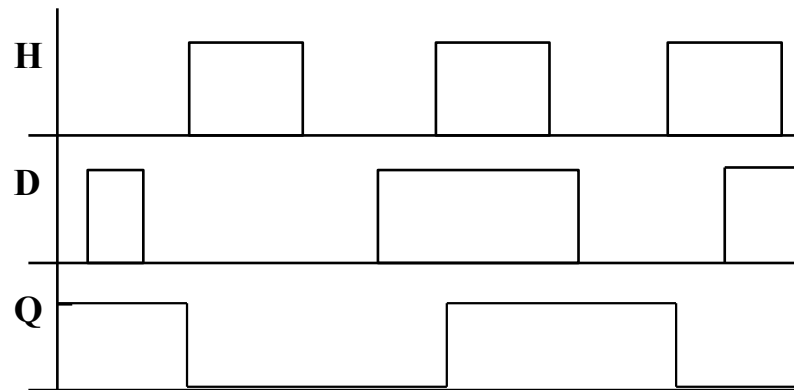
◆ Détail

- Acquisition du niveau logique (valeur) présent sur l'entrée D lors d'un front montant
- Le niveau logique acquis est maintenu sur la sortie Q (valeur mémorisée) jusqu'à la prochaine acquisition

Bascule D front (D flip flop)



H	D	Q
	0	0
	1	1
Hors front	X	Q-

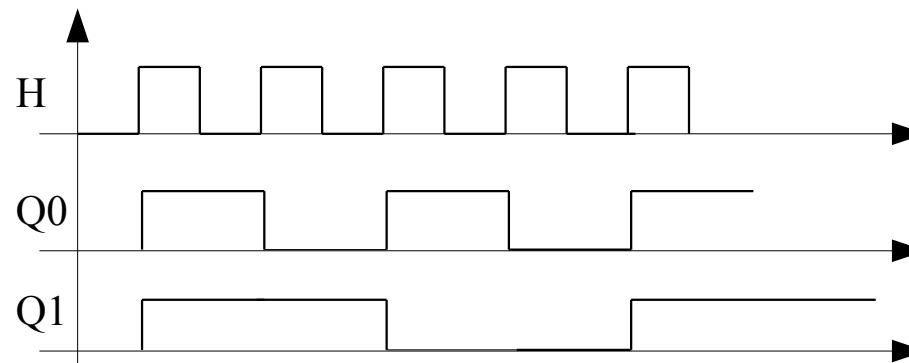
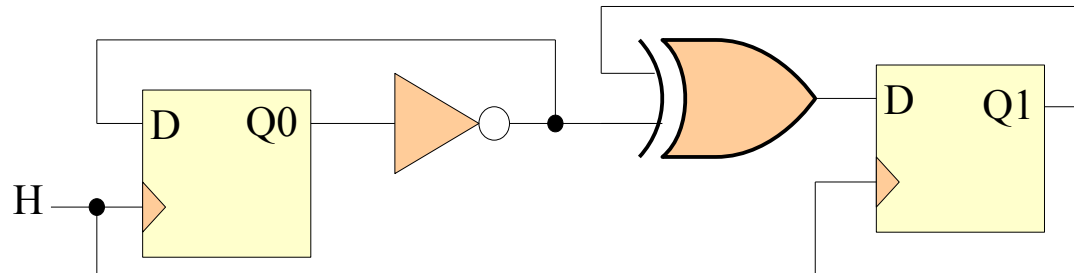


Bascules synchronisées, exemple

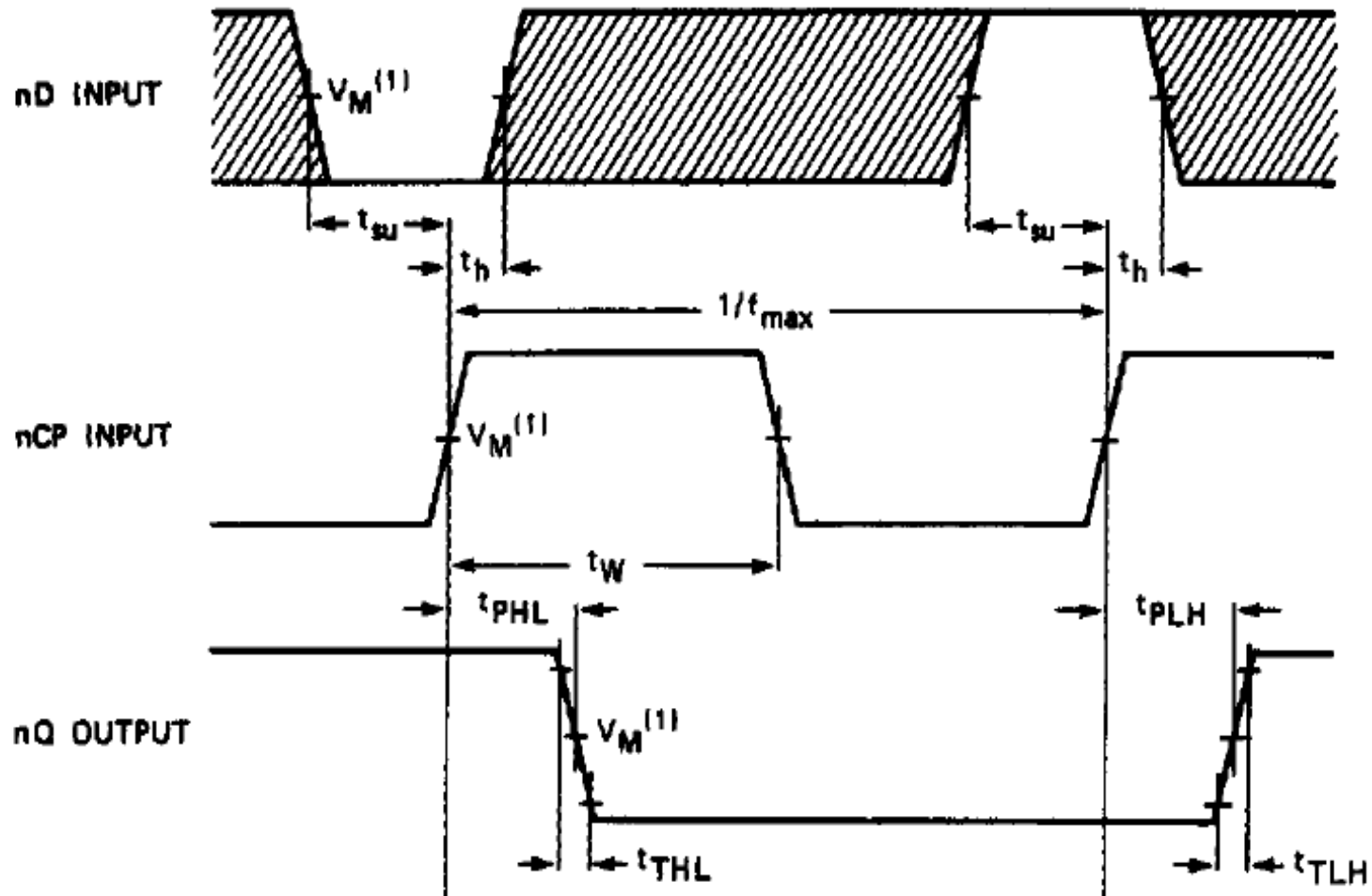
◆ Bascule :

- Acquisition valeur sur front d'horloge
- Mise à jour de la sortie **après** front d'horloge (délais de transition)

◆ Exemple



Caractéristiques techniques, ex 74HCT74

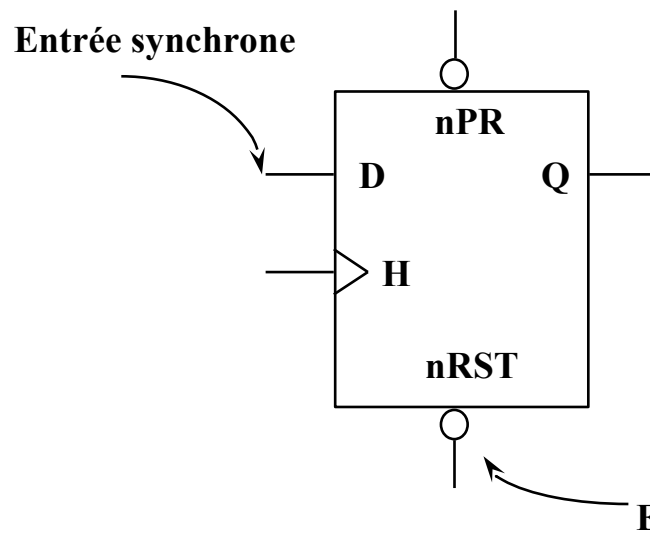


t_{su} : Set up time (10ns pour garantie et typ 5ns)
 t_h : hold time (3ns pour garantie et typ -3ns)

t_{PHL} : délais de propagation,
 typ = 20ns, max = 40ns

Entrées de forçage (entrées asynchrones)

- ◆ Entrées de forçage ou entrées asynchrones : permet de forcer l'état de la bascule indépendamment du signal de synchronisation (initialisation)
- ◆ Exemple



$nPR = 0$: Preset, forçage à 1 $Q = 1$

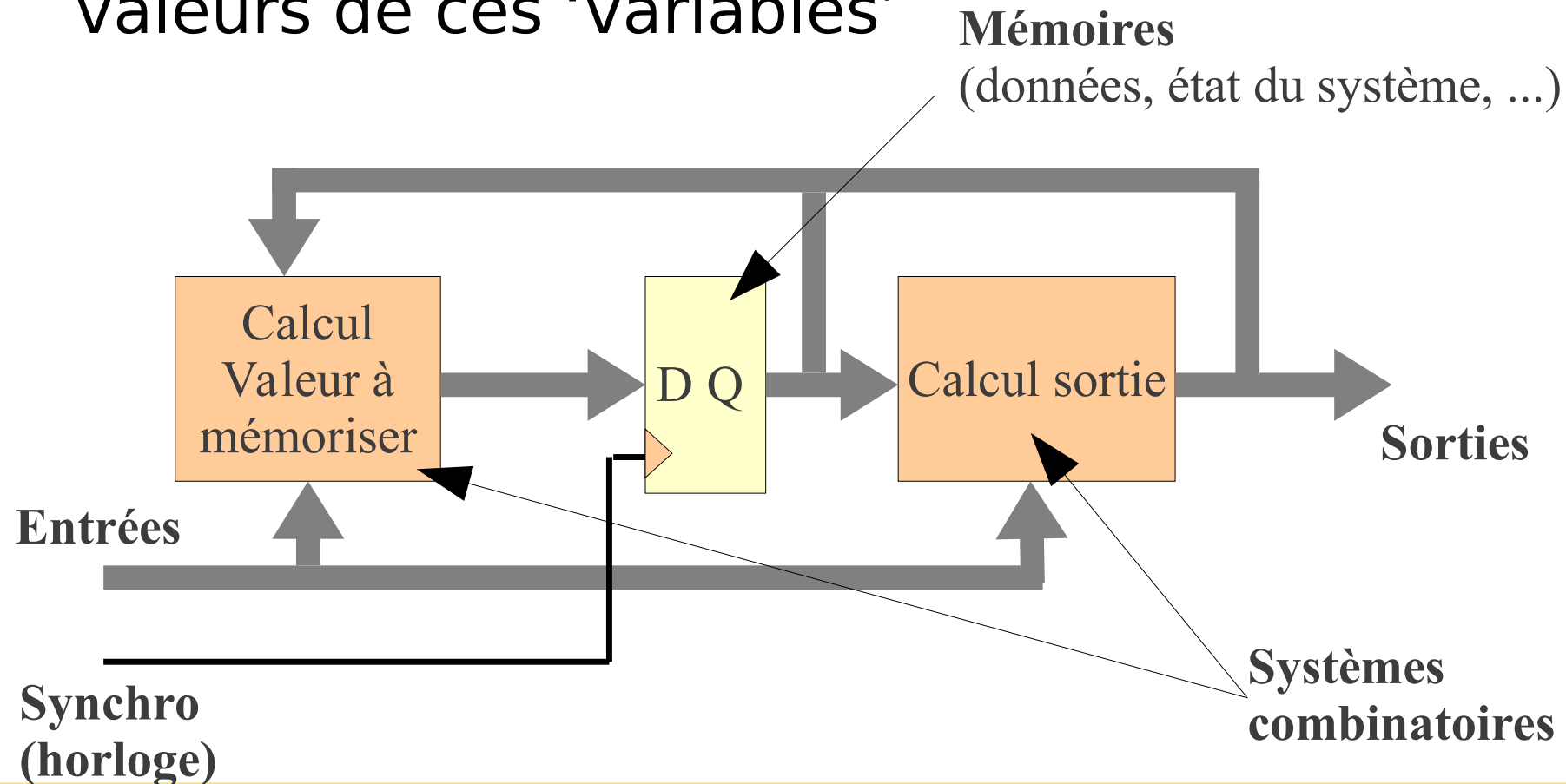
$nRST = 0$: Reset, forçage à 0 $Q = 0$

$nPR = nRST = 1$: fonctionnement normal

$nPR = nRST = 0$: indésirable (ambiguïté)

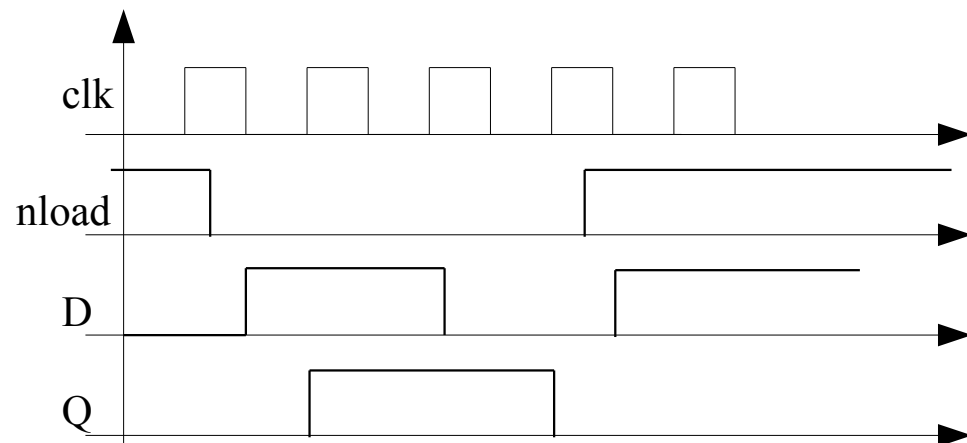
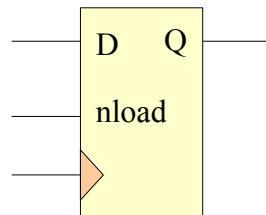
Système synchrone

- ◆ Le système évolue au rythme des mémorisations (évolution des 'variables' internes)
- ◆ Les circuits de calcul 'préparent' les prochaines valeurs de ces 'variables'

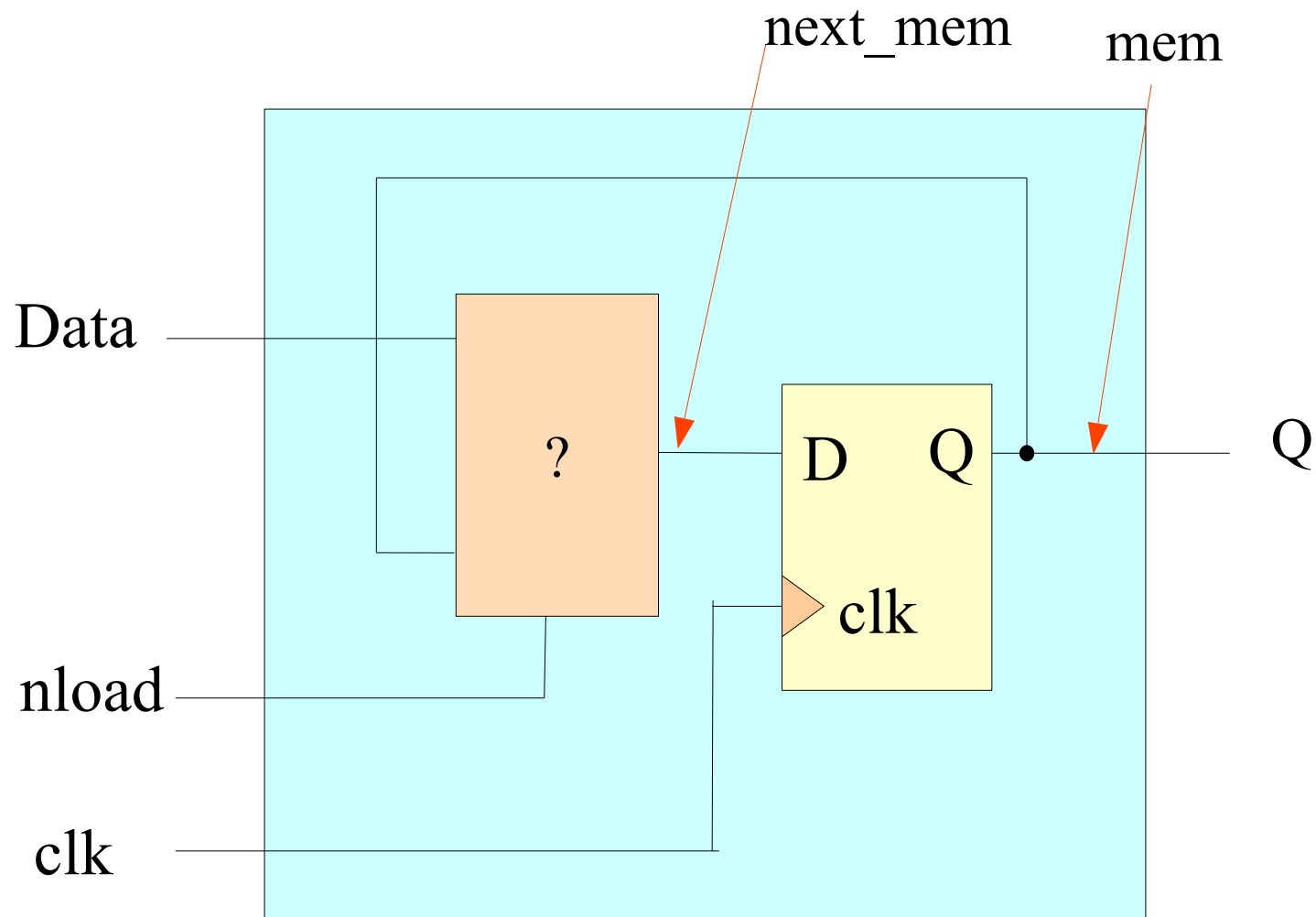


Exemple : mémoire avec commande de chargement

- ◆ Deux fonctionnements :
 - $nload = 0$, fonctionnement traditionnel (mémorisation à chaque front)
 - $nload = 1$, l'entrée de synchronisation n'a plus d'effet (mémoire)
- ◆ Exemple



Synthèse matérielle



Synthèse

If nload = '0' then

D <= Data ;

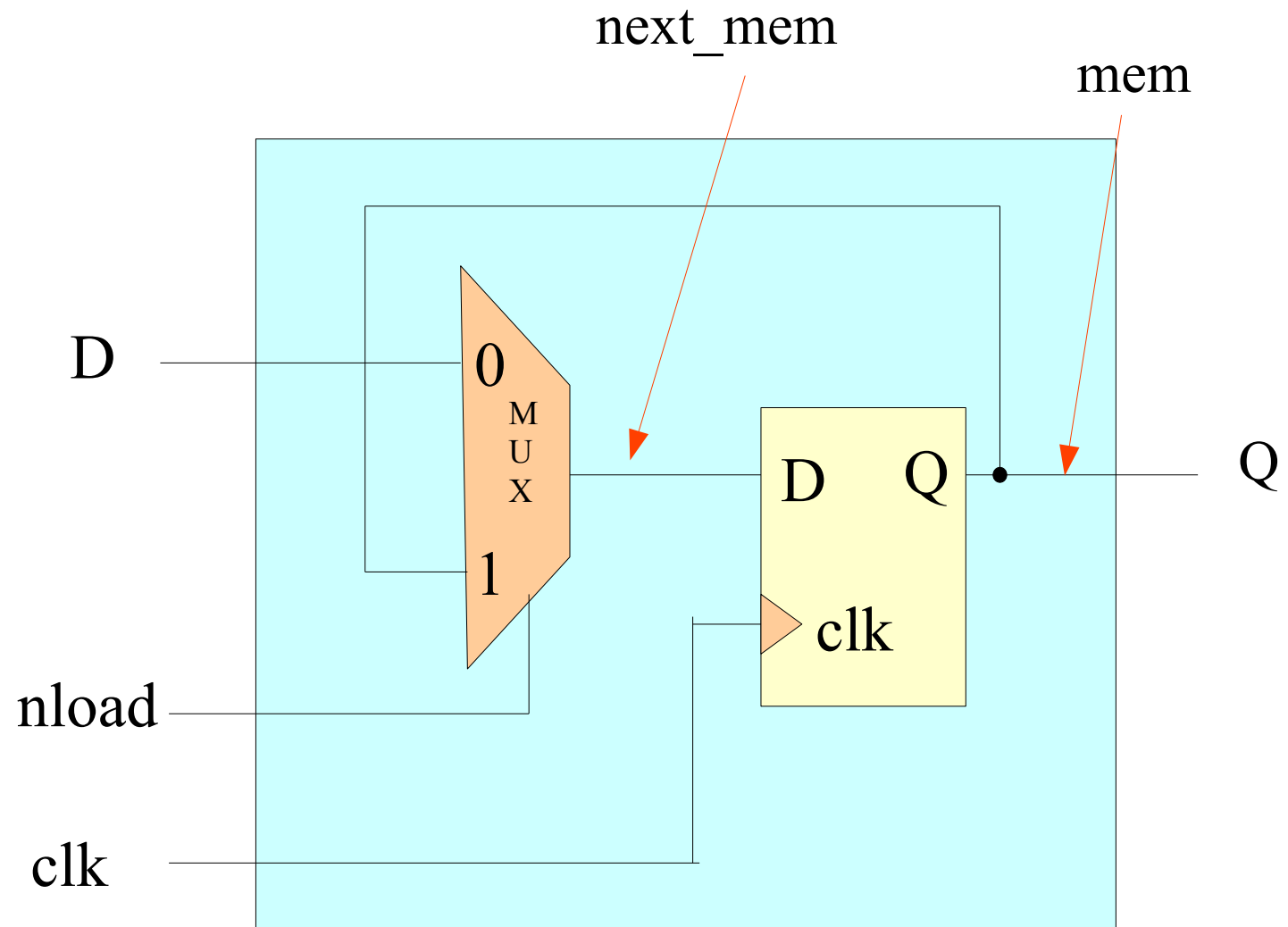
else

D <= Q ;

nload	D
0	Data
1	Q

$$D = \overline{nload} \text{ Data} + nload Q$$

Synthèse matérielle

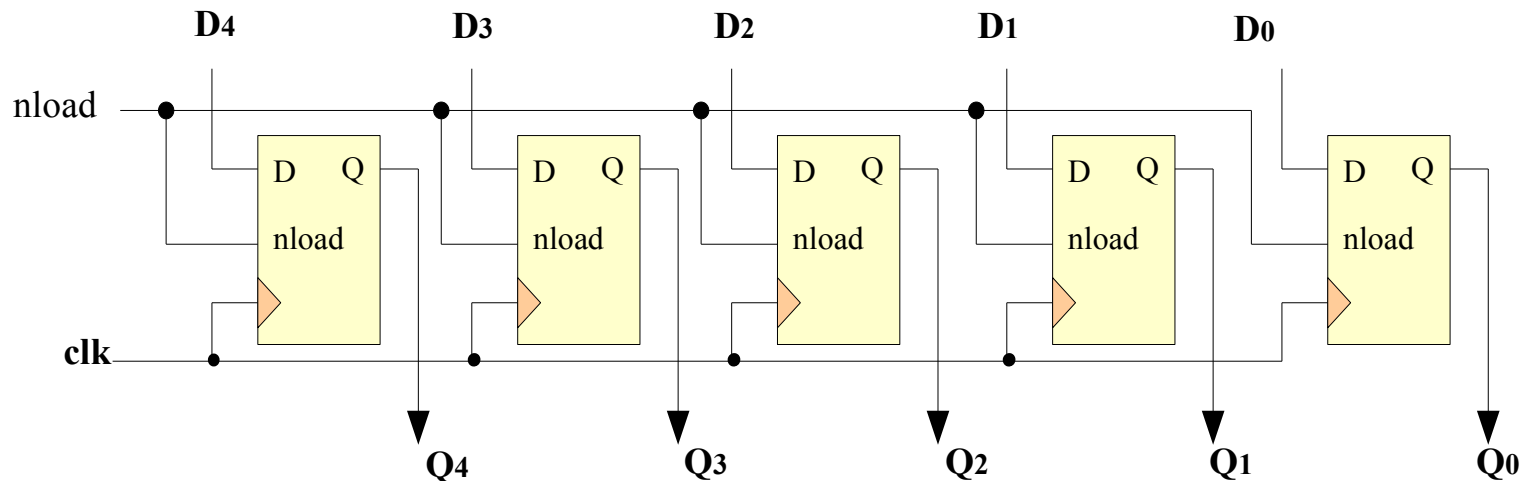


Registres

- ◆ Cellule mémoire élémentaire sur N bits
 - Mise en parallèle de N bascules
- ◆ Mode d'accès à la donnée
 - Chargement des données :
 - Parallèle (par mot)
 - Série (par décalage)
 - Lecture des données mémorisées
 - parallèle
 - Série
- ◆ Synchrone ou asynchrone

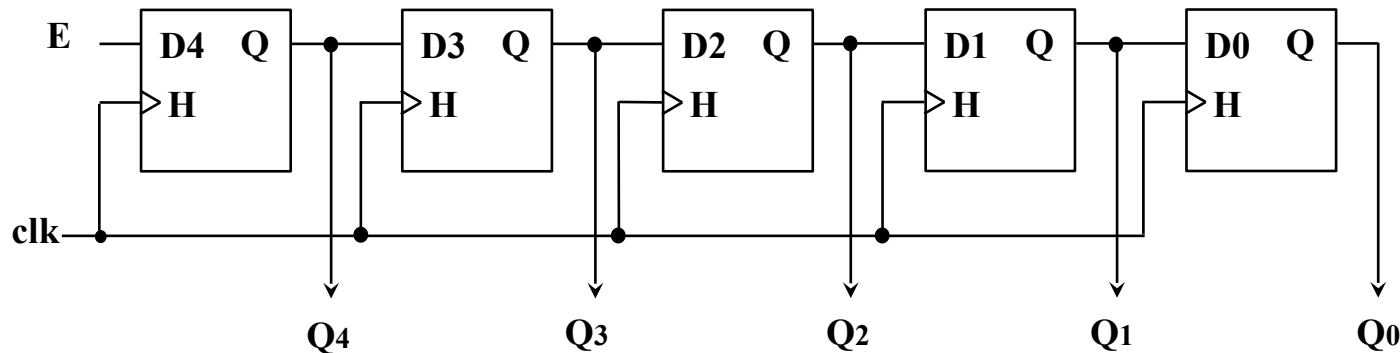
Exemple : chargement/lecture parallèle (synchrone)

- ◆ Registres 5 bits à commande de chargement



- ◆ Remarque : lecture donnée mémorisée = lecture des niveaux logiques présents en sortie

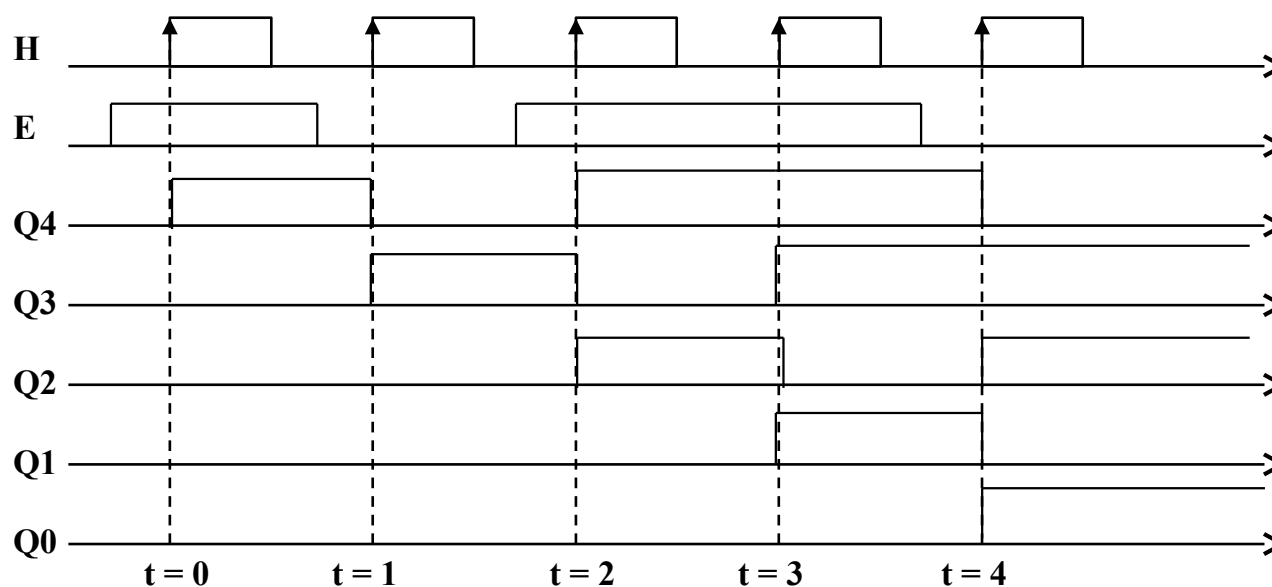
Registre à décalage (chargt. série lect. //)



Registre chargement série
lecture parallèle

Asynchrone

t = 4

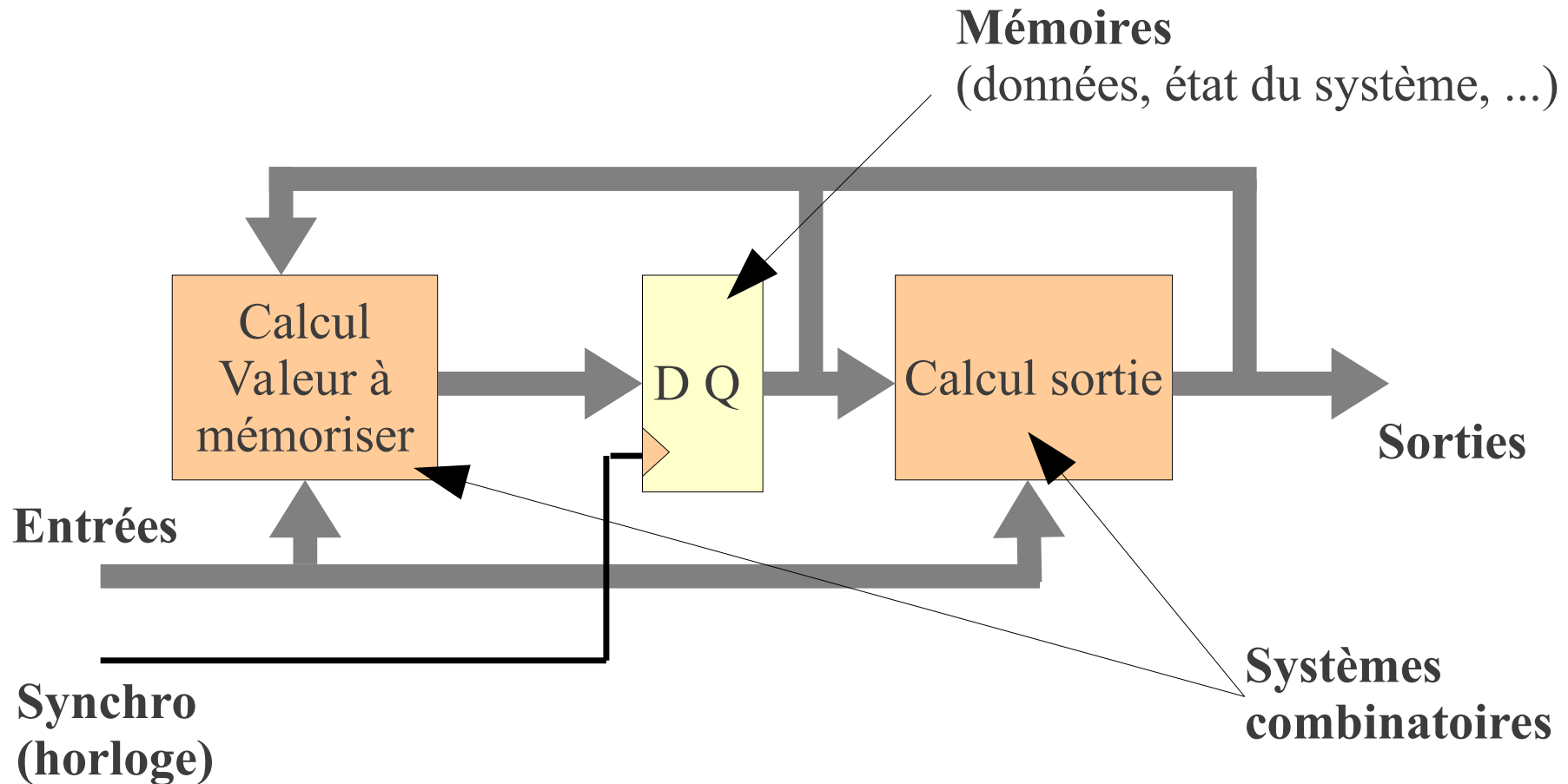


Chargement d'une donnée série
par la gauche (décalage droite soit
poids fort vers poids faible)

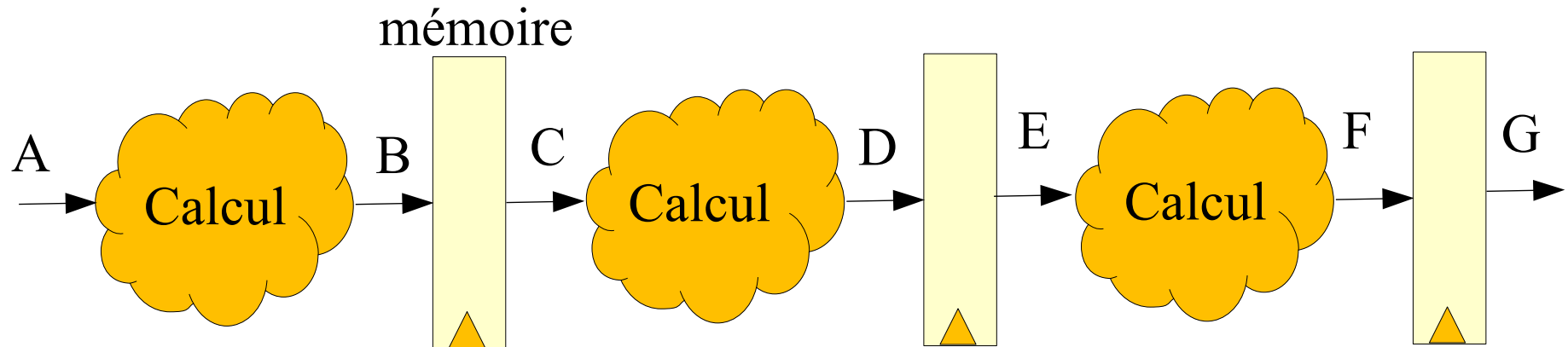
La donnée est $D = 01101$, elle est
transmise poids faible en premier

Au bout de 5 fronts d'horloge
 $Q=01101$

Systemes numériques utilisant des mémoires



Mélange calcul/mémorisation



- ◆ C, E, G : variable du système en mémoire
- ◆ B, D, F :
 - Données calculées à partir de A, C, E
 - Futures valeurs en mémoire

Exemple d'application d'un registre : compteur binaire

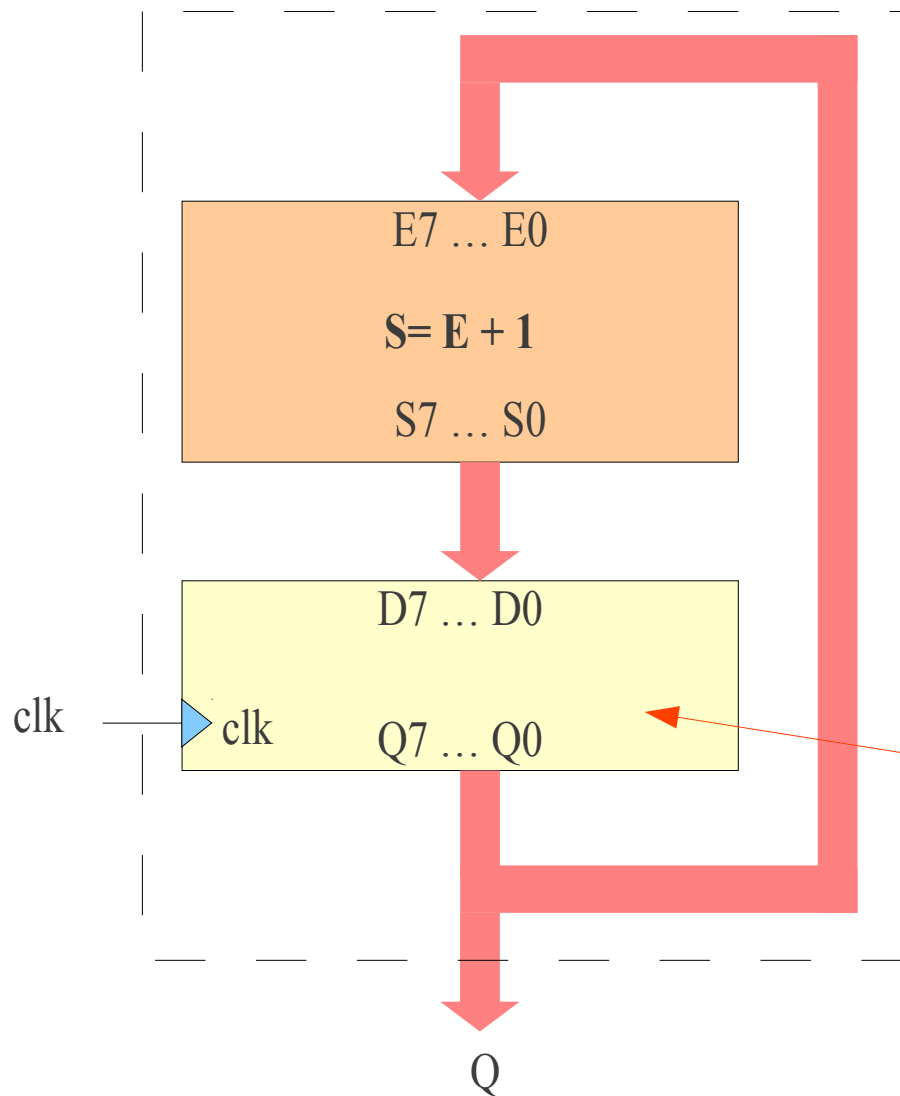


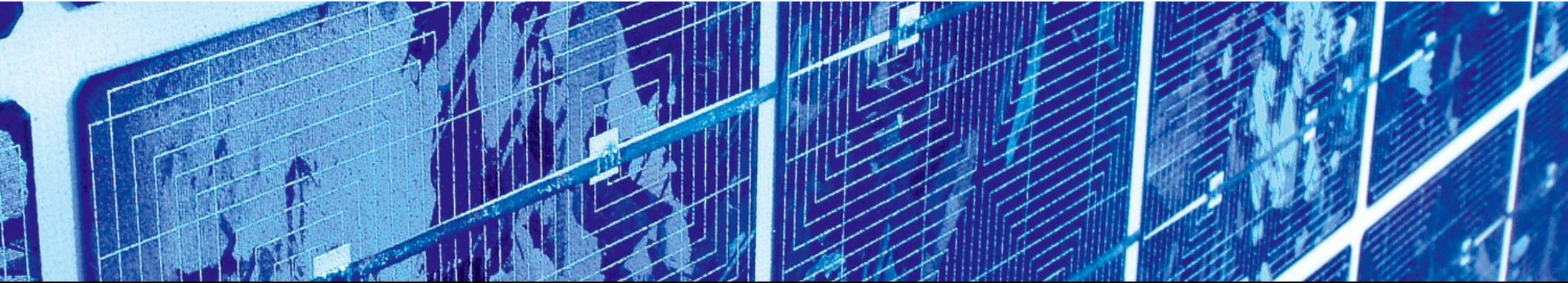
Table de vérité

E (=Q)	S (=D)
00000000	00000001
00000001	00000010
00000010	00000011
00000011	00000100
.	.
.	.
.	.

Registre 8 bits (variable mémoire)

Exemple d'un compteur par 6

	Etat présent			Etat Futur		
Etats	Q2	Q1	Q0	D2	D1	D0
0	0	0	0	0	0	1
1	0	0	1	0	1	0
2	0	1	0	0	1	1
3	0	1	1	1	0	0
4	1	0	0	1	0	1
5	1	0	1	0	0	0
6	1	1	0	x	x	x
7	1	1	1	x	x	x



UNIVERSITÉ DE TECHNOLOGIE DE BELFORT-MONTBÉLIARD

Circuits spécialisés mémoire

SY41

Nicolas Lacaille
Nicolas.lacaille@utbm.fr

Généralités sur les mémoires

- ◆ Dispositif capable d'emmagasiner et de restituer une information binaire
- ◆ Unité d'information : le bit.
- ◆ Le dispositif de stockage est appelé point mémoire ou cellule mémoire
- ◆ Une mémoire est organisée en mot mémoire (de N bits)
- ◆ Capacité d'une mémoire : quantité d'information qu'elle peut stocker (en bits ou mots)

Accès

◆ Accès aléatoire

- La cellule mémoire est accédée directement
- Sélection de la cellule par une adresse

◆ Accès séquentiel

- Les cellules mémoires sont accédées séquentiellement les unes après les autres
- Deux types de comportement pour les mémoires séquentielles
 - FIFO : First In First Out
 - LIFO : Last In First Out

Classification technologique

- ◆ Mémoires mortes
 - Mémoires qui conservent l'information stockée même sans alimentation
 - ROM, PROM, EEPROM, flash
- ◆ Mémoires vives
 - Mémoires qui perdent l'information stockée lorsque l'alimentation n'est plus présente
 - 2 types
 - SRAM : Ram Statique
 - DRAM : Ram Dynamique

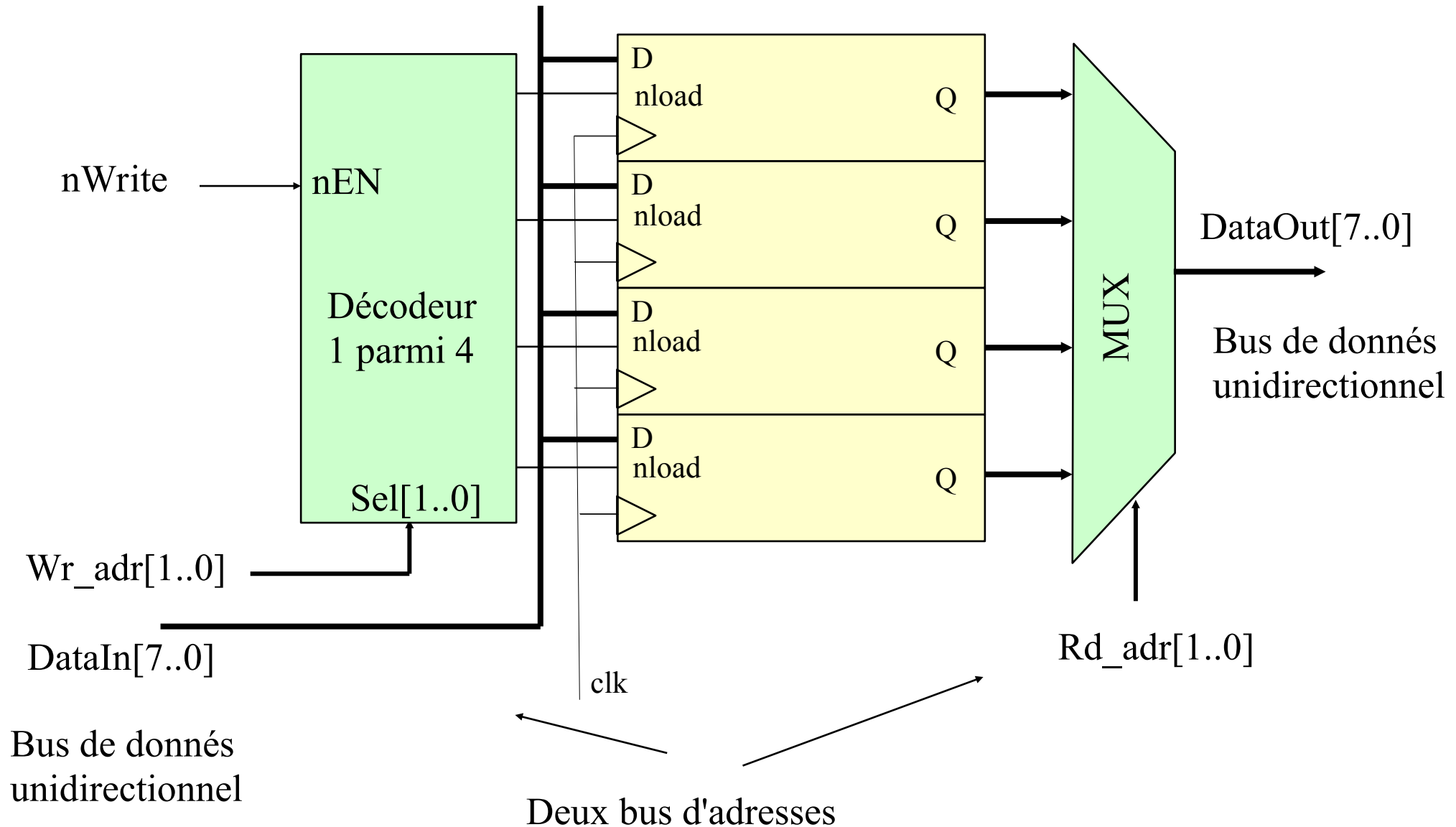
Accès mémoire

- ◆ L'accès aux dispositifs mémoires s'effectue via des lignes électriques
- ◆ Bus adresses : pour les mémoires à accès aléatoire
 - Entrées de sélection des mots mémoires
- ◆ Bus de données
 - Monodirectionnel pour les mémoire en lecture seule
 - Bidirectionnel pour les mémoire en lecture/écriture
- ◆ Une seule ligne d'accès adresses/données pour les mémoires à accès série

SRAM à accès aléatoires

- ◆ Static Random Access Memory
- ◆ Cellules mémoire = bascules (bistables)
- ◆ Organisation
 - Nombre de mots x taille des mots
- ◆ Bus d'adresse = $\log_2(\text{nombre de mot})$
- ◆ Bus de données = taille des mots

Exemple : SRAMDQ 4 x 8 Synchrones



Décodeur d'adresse (écriture)

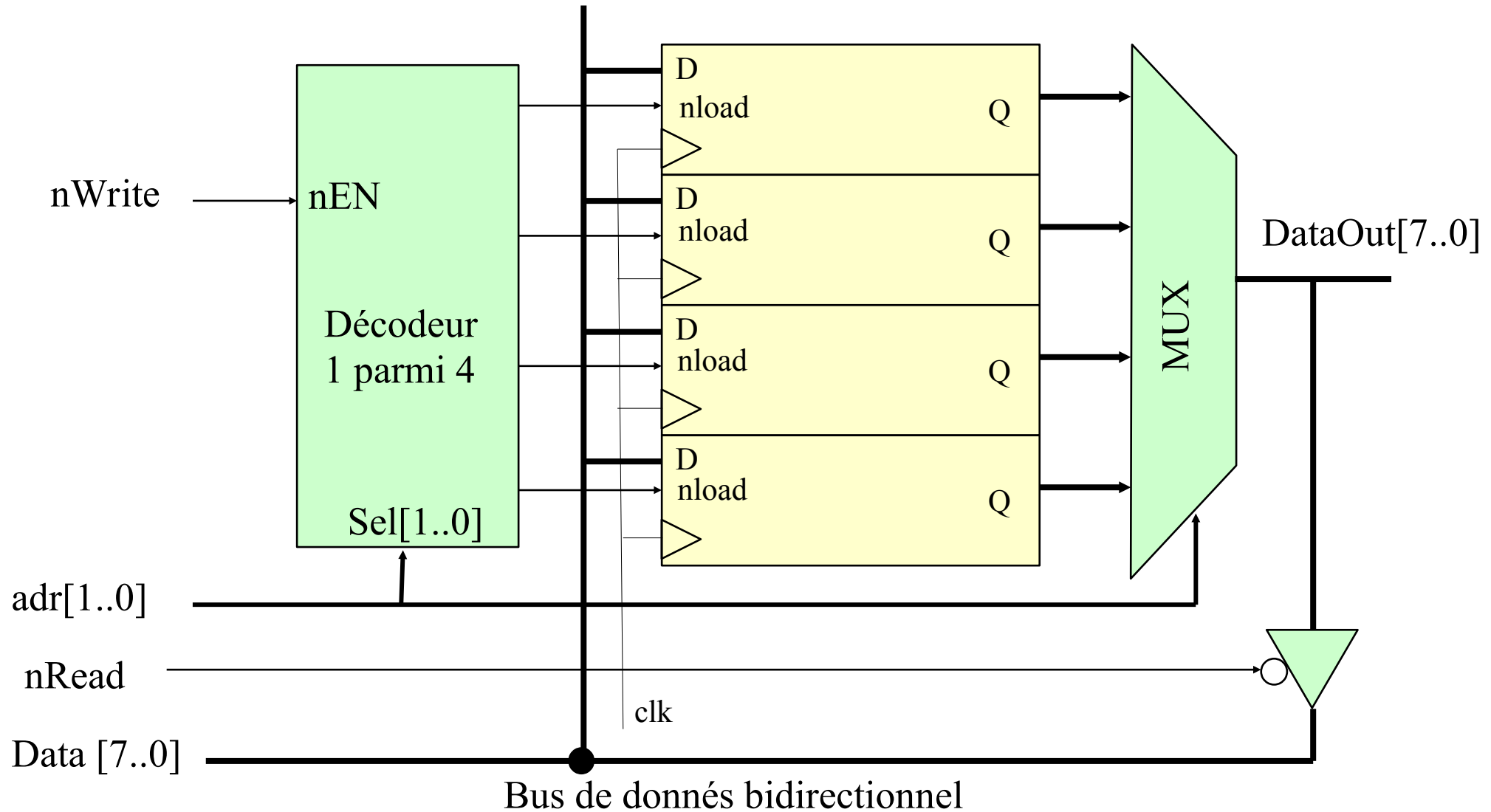
- ◆ Le décodeur place une sortie au niveau bas en fonction de la valeur sélectionnée par l'adresse en écriture
 - Permet de sélectionner un registre
 - Lorsque nWrite est à 0 : enregistrement de la donnée (front montant) dans le registre sélectionné
- ◆ La sélection de la donnée en lecture consiste à aiguiller une donnée sur le bus de sortie

nEn	Sel[1..0]	S[3..0]
1	X X	1 1 1 1
0	0 0	1 1 1 0
	0 1	1 1 0 1
	1 0	1 0 1 1
	1 1	0 1 1 1

SRAM avec un seul bus de données

- ◆ L'exemple précédent utilise
 - Deux bus d'adresses
 - Deux bus de données unidirectionnels
- ◆ Permet deux accès simultanés (en lecture et écriture) : mémoire double accès
- ◆ D'autres configurations sont possibles
 - Un seul bus d'adresse permettant la sélection d'une seule cellule
 - Un seul bus de données bidirectionnel

Exemple : SRAM DQ 4 x 8 synchrone



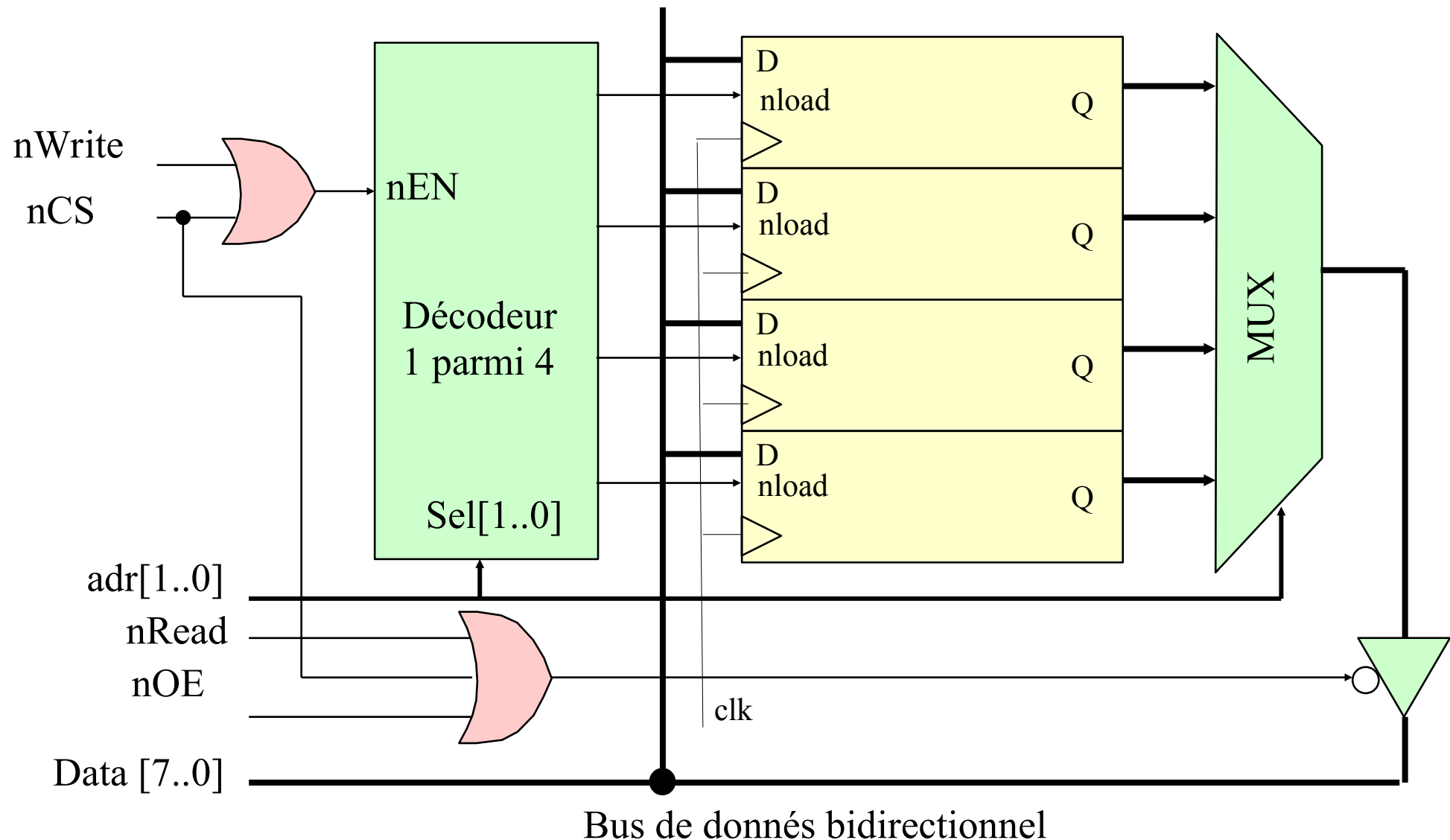
Lecture/écriture

- ◆ Mémoire accessible en lecture/écriture via un seul bus de données
- ◆ Les données correspondent à des potentiels électriques qui sont imposés par
 - La mémoire lors d'une lecture
 - Un dispositif externe lors d'une écriture
- ◆ Lors des écritures, les sorties des cellules mémoires doivent impérativement être isolées du bus

Lignes de contrôle

- ◆ Commande lecture écriture : nRead et nWrite
- ◆ Sélection du composant : Chip Select ou Chip Enable (nCS)
 - Activation/désactivation du circuit
 - Permet une déconnexion des bus du circuit
- ◆ Autorisation de sortie : Output Enable (nOE)
 - Commande des buffers 3, autorisant la mémoire à imposer des tensions sur les lignes
 - Commandes de protection supplémentaire pour éviter les conflits électriques

Exemple : SRAM 4 x 8 synchrone



Synchrone/asynchrone

- ◆ Deux types de synchronisation pour la mémorisation et la lecture
- ◆ Asynchrone : mémorisation par génération d'un front actif (CE ou nWrite)
- ◆ Synchrone : utilisation d'une horloge de synchronisation
 - Les commandes de lecture écriture sont échantillonnées sur front d'horloge
 - Les données sont mémorisées sur front d'horloge
 - Les données sont envoyées sur front