
Bascules et logique séquentielle

Daniel Etiemble
de@lri.fr

Logique séquentielle

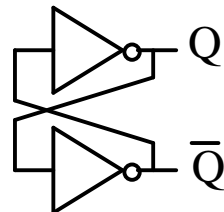
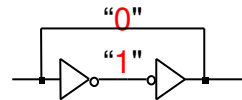
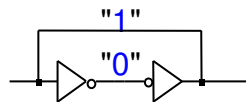
- Logique séquentielle
 - Le système a des « états »
 - Dans un système séquentiel
 - Éléments de mémorisation
 - Les sorties dépendent des états et des entrées
 - Le nouvel état est fonction des entrées et de l'état précédent
 - Systèmes synchrones
 - Une horloge indique le moment où les éléments de mémorisation acceptent les nouvelles valeurs et changent d'état
 - Systèmes asynchrones
 - Aucun indication sur le moment des changements d'état

Éléments de mémorisation : bascules et registres

- Mémorisation d'un bit
 - Bistable
 - Bascule RS
 - Bascules transparentes : latch
 - Bascules opaques : D
- Registres
 - Un registre est un ensemble de bascules la même commande d'horloge
 - Le registre a les mêmes propriétés que les bascules qui le composent
 - Transparence versus opacité

Le bistable

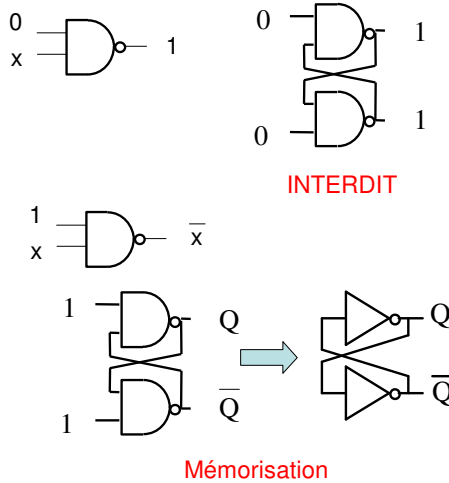
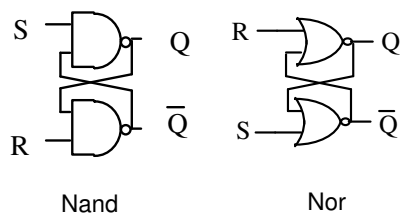
- Deux inverseurs en série
 - Deux états possibles : mémorise un bit
 - Ne peut changer d'état



Pour « écrire », il faut modifier le bistable
-Point mémoire SRAM
-Bascule RS
-Bascule Latch « CMOS »

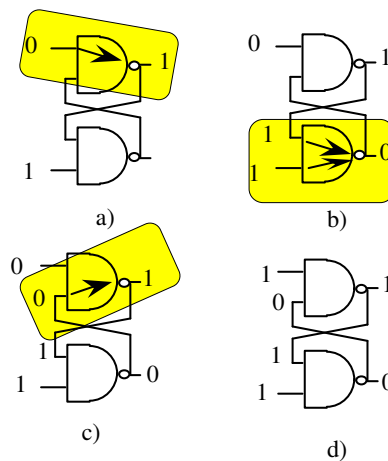
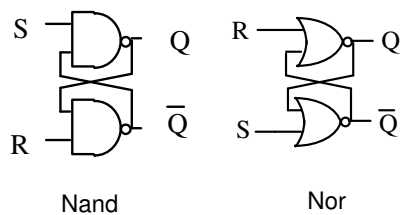
Bascule RS

- Deux types
 - NAND
 - NOR
- Fonctionnement RS NAND
 - État interdit
 - Mémorisation



Bascule RS

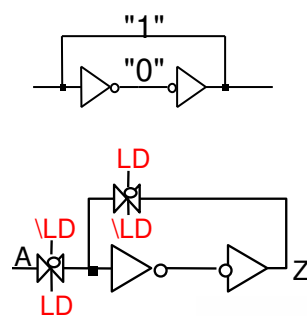
- Deux types
 - NAND
 - NOR
- Fonctionnement RS NAND
 - Ecriture d'une valeur



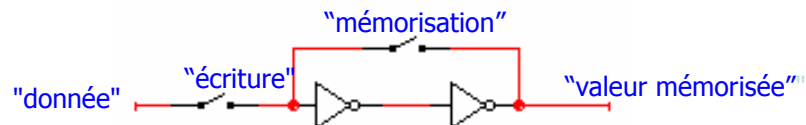
Résumé sur la bascule RS

- Fonctionnement
 - NAND
 - Si $R=S=1$, État mémoire
 - Si $R \neq S$, alors écriture $Q=R$
 - $R=S=0$ INTERDIT
 - NOR
 - Si $R=S=0$, État mémoire
 - Si $R \neq S$, alors écriture $Q=S$
 - $R=S=1$ INTERDIT
- Remarque
 - Les entrées R et S sont à la fois des entrées de type « état » et de type « temps »
 - On sépare « état » et « temps » avec des bascules latch
 - Entrée d'état : D
 - Entrée d'horloge : C

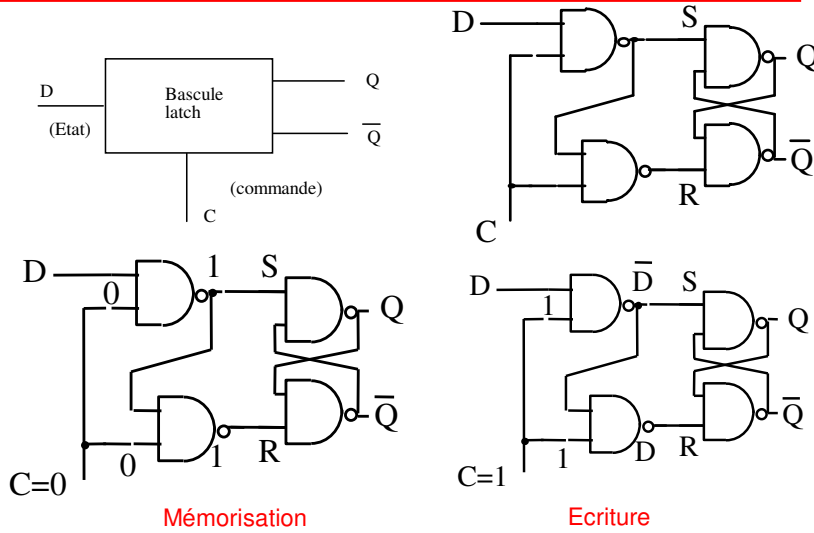
Le latch "CMOS"



- Utilise les interrupteurs (portes de transmission) pour
 - Réaliser la contre réaction (mémoire)
 - Couper la contre réaction pour charger une nouvelle valeur
- Fonctionnement
 - $LD = 1$ alors $Z = A$
 - $LD = 0$ alors $Z = Z$
 - Transparence lorsque $LD=1$



Bascule latch NAND ou NOR

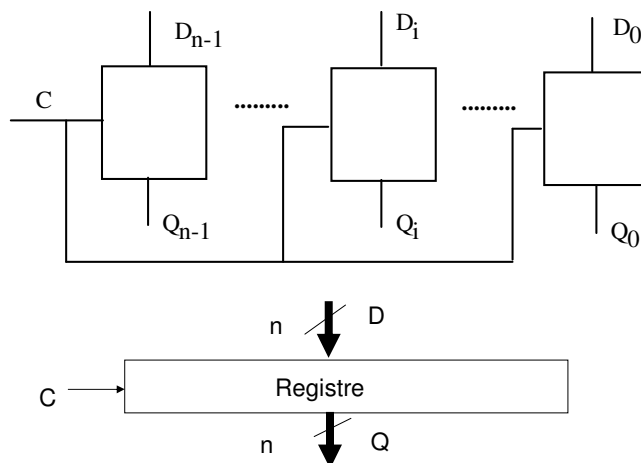


M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

9

Registres

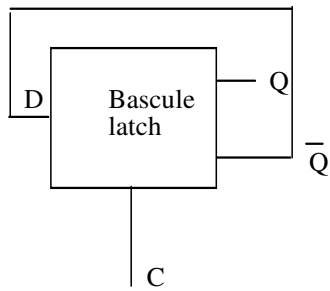
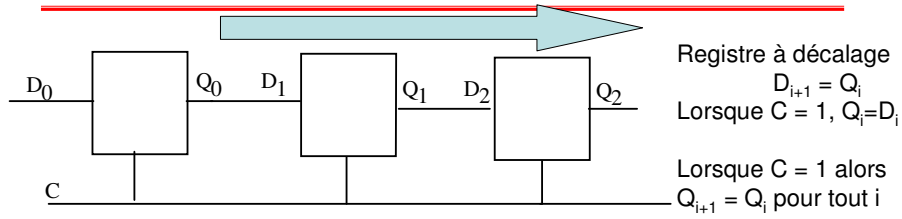


M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

10

Bascules, registres latch et transparence

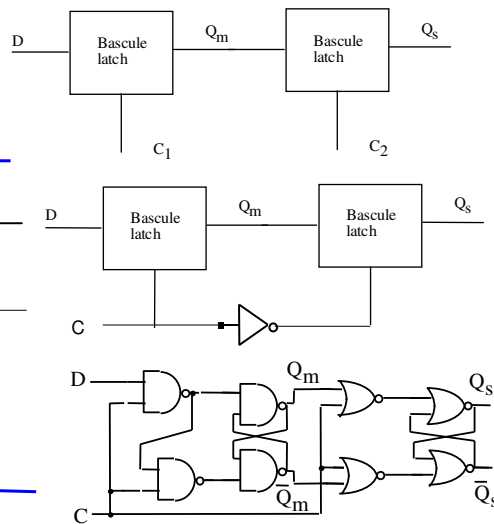
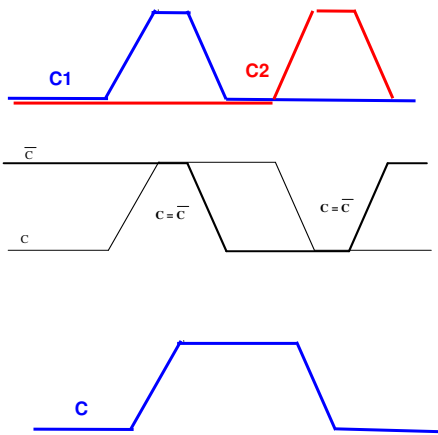


Liaison
 $D = \overline{Q}$
 Lorsque $C = 1$, $Q = D$
 Lorsque $C = 1$ alors $Q = \overline{Q}$

Nécessité de bascules NON transparentes

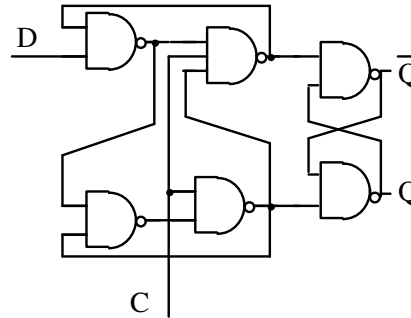
Bascule D maître esclave

- Bascules opaques
 - Maître esclave

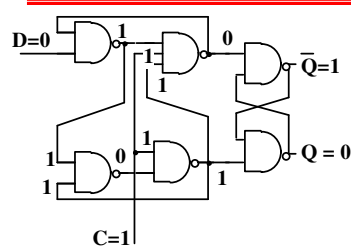


Bascules D à commande par flanc

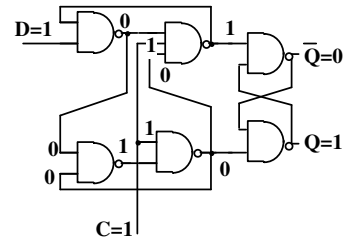
- Bascules opaques
 - Bascule D à commande par flanc
 - 6 portes NAND (sensible au front montant de l'horloge)
 - 6 portes NOR (sensible au front descendant de l'horloge)
 - Sur transition d'horloge, l'entrée D est recopié vers Q et mémorisée



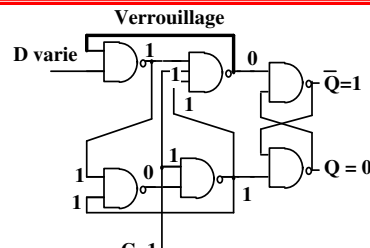
Bascule D à 6 portes NAND



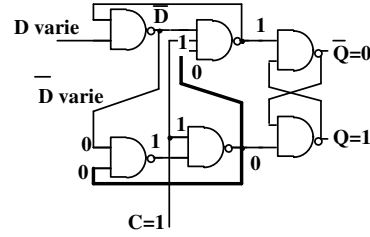
Écriture d'un 0



Écriture d'un 1

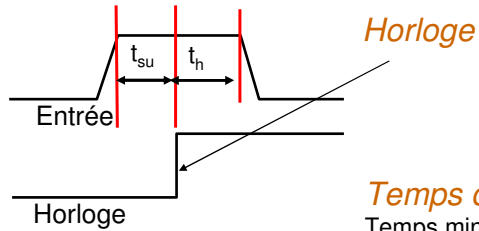


Verrouillage après écriture d'un 0



Verrouillage après écriture d'un 1

Temps d'établissement - temps de maintien



Temps d'établissement (T_{su})

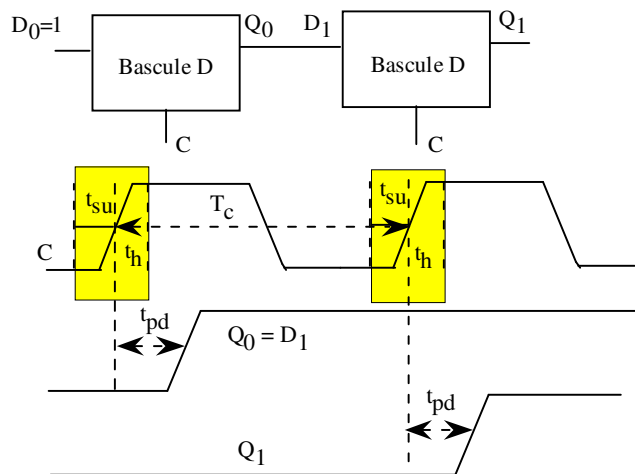
Temps minimum avant l'arrivée de l'horloge pendant lequel l'entrée doit être stable

Temps de maintien (T_h)

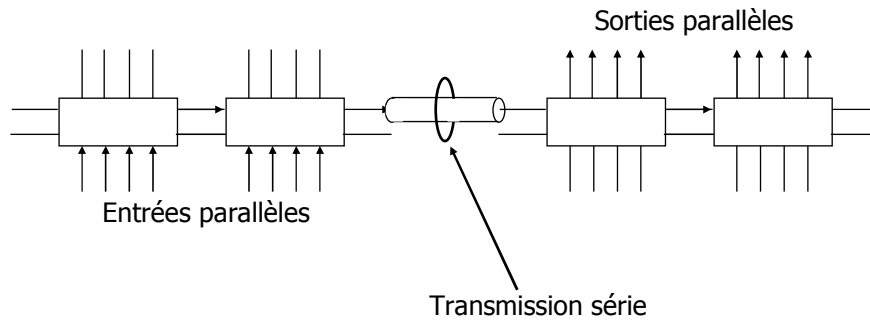
Temps minimum après l'arrivée de l'horloge pendant lequel l'entrée doit rester stable

Il y a une fenêtre temporelle de part et d'autre de la transition d'horloge pendant laquelle l'entrée doit rester stable

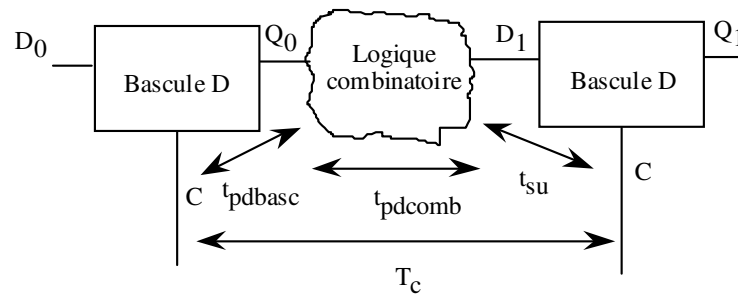
Registre à décalage avec bascules D



Application des registres à décalage

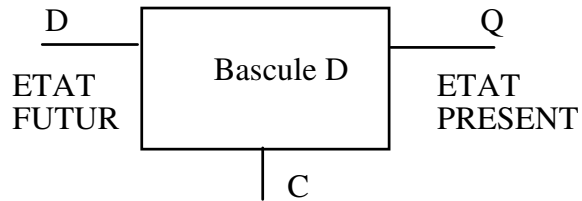


Principe fondamental de la logique synchrone



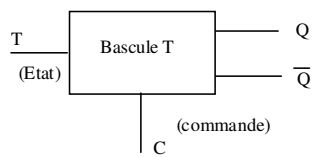
$$F_{\max} = \frac{1}{T_{c\min}} = \frac{1}{t_{pd(\text{bascule})} + t_{pd(\text{combinatoire})} + t_{su}}$$

Utilisation de la bascule D



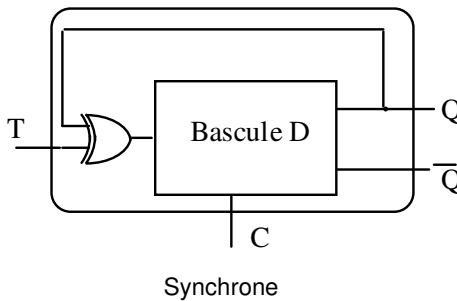
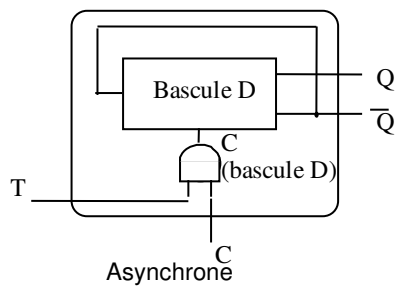
- Automate
 - État futur = fonction (État présent, Entrées)
 - État futur = entrées des bascules D (du registre D)
 - État présent = sorties des bascules D
 - Transition d'horloge = passage d'un état à l'état suivant

Autres bascules : bascule T

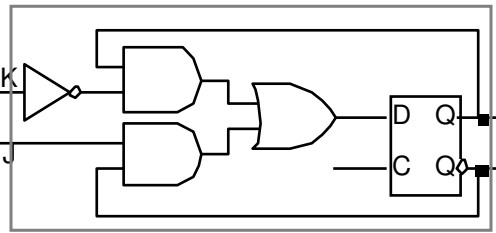
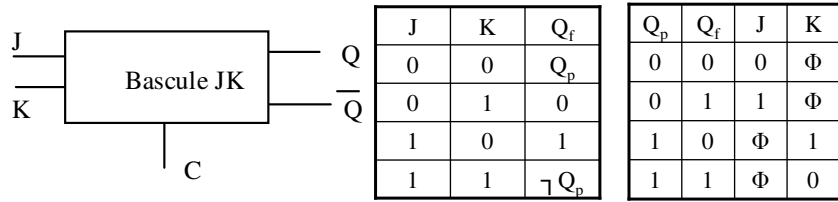


T	Q _p	Q _f
0	0	0
0	1	1
1	0	1
1	1	0

Q _p	Q _f	T
0	0	0
0	1	1
1	0	1
1	1	0



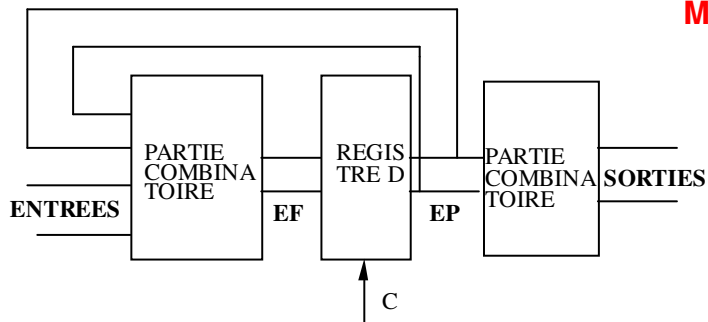
Autres bascules : bascule JK



J-K avec bascule D

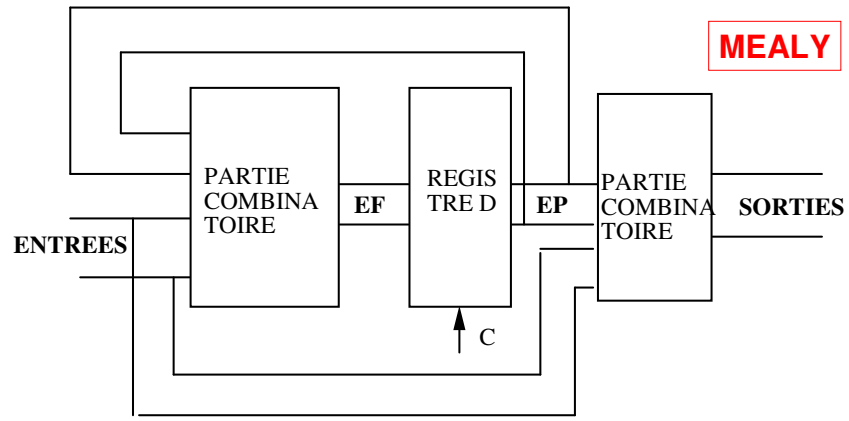
Automates synchrones

MOORE



$EF = f(EP, \text{Entrées})$
 $\text{Sorties} = g(EP)$

Automates synchrones



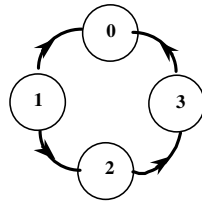
$EF = k(EP, \text{Entrées})$
 $\text{Sorties} = h(EP, \text{Entrées})$

Méthode de synthèse d'automates synchrones

- Méthode générale
 - Graphe de transition
 - Table de transition
 - Codage des états et table de transition codée
 - Implantation avec registre D et logique combinatoire
- Méthodes plus spécifiques
 - Certaines implémentations sont plus efficaces en utilisant des opérateurs particuliers
 - Registres à décalages
 - Compteurs
 - Etc.

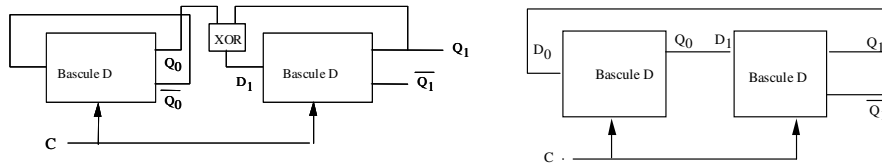
Les compteurs

- Un compteur est un automate sans entrées.



EP	EF
0	1
1	2
2	3
3	0

N	Naturel Q1	Naturel Q0	Gray Q1	Gray Q0	Pire Q1	Pire Q0
0	0	0	0	0	0	0
1	0	1	0	1	1	1
2	1	0	1	1	0	1
3	1	1	1	0	1	0



M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

25

Compteurs modulo 2^n

Compteur par 8

N	Q_2	Q_1	Q_0	D_2	D_1	D_0
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	1	1	0
6	1	1	0	1	1	1
7	1	1	1	0	0	0

$$D_0 = \overline{Q_0} = 1 \oplus Q_0$$

$$D_1 = Q_1 \oplus Q_0$$

$$D_2 = Q_2 \oplus (Q_1 Q_0)$$

Relation de récurrence

M2 ISIC
2008-2009

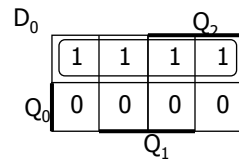
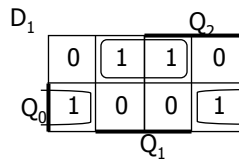
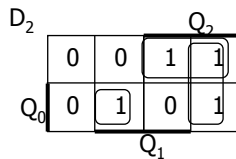
Fondements des systèmes numériques
D. Etiemble

26

Implémentation du compteur par 8

Q ₂	Q ₁	Q ₀	D ₃	D ₂	D ₀
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

$$\begin{aligned}
 D_0 &\leq Q_0' \\
 D_1 &\leq Q_0 Q_1' + Q_0' Q_1 \\
 &\leq Q_0 \text{ XOR } Q_1 \\
 D_2 &\leq Q_0 Q_1 Q_2' + Q_0' Q_2 + Q_1' Q_2 \\
 &\leq (Q_0 Q_1) Q_2' + (Q_0' + Q_1') Q_2 \\
 &\leq (Q_0 Q_1) Q_2' + (Q_0 Q_1)' Q_2 \\
 &\leq (Q_0 Q_1) \text{ XOR } Q_2
 \end{aligned}$$



Compteurs avec mise à zéro (méthode générale)

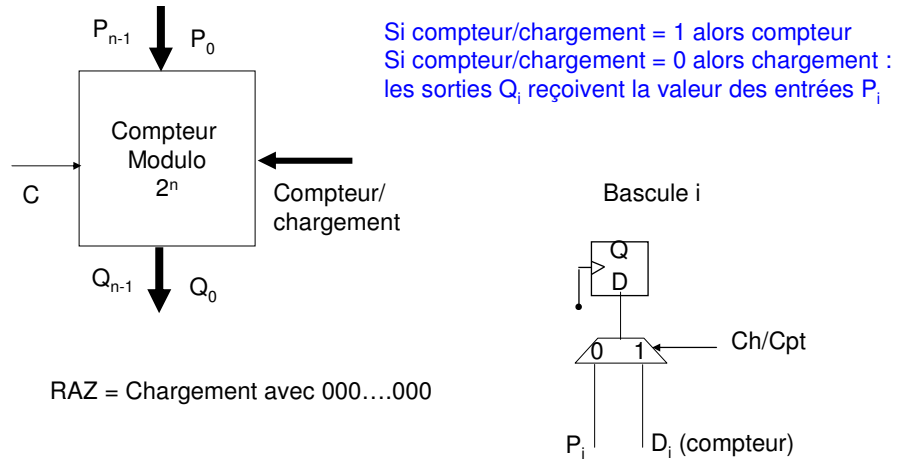
Exemple : compteur par 4 avec RAZ
 Si RAZ=1 alors EF = 0 pour tout EP
 Si RAZ=0 alors EF = (EP+1) mod 4

	RAZ=0	RAZ=1
EP	EF	EF
0	1	0
1	2	0
2	3	0
3	0	0

Entrée	EP	EF	D1	D0
RAZ	Q1	Q0	D1	D0
0	0	0	0	1
0	0	1	1	0
0	1	0	1	1
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	0
1	1	1	0	0

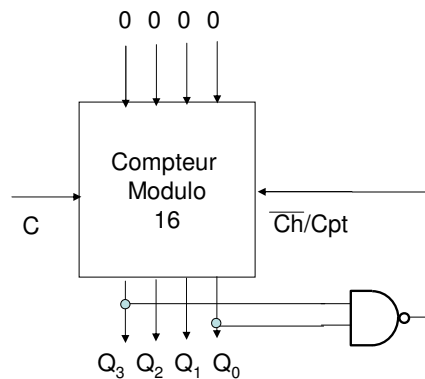
$$\begin{aligned}
 D_0 &= \overline{RAZ} \cdot Q_0 \\
 D_1 &= \overline{RAZ} \cdot (Q_1 \oplus Q_0)
 \end{aligned}$$

Compteur 2^n avec chargement

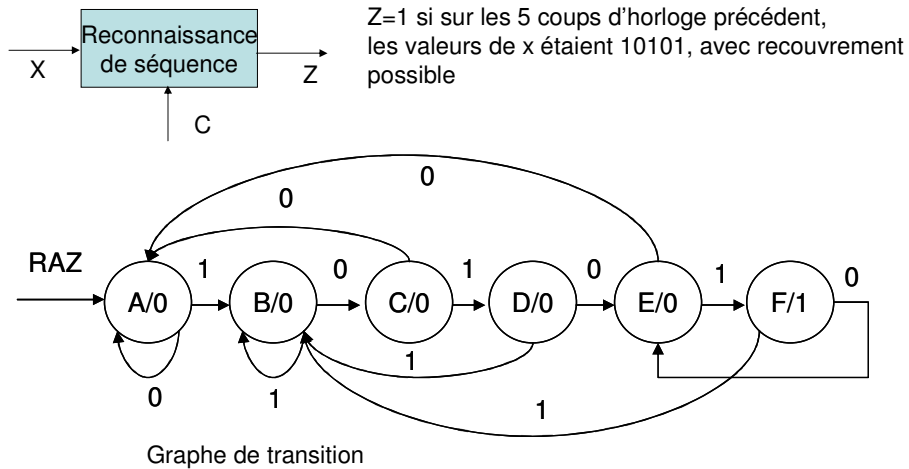


Compteur par 10 avec un compteur par 16

- Fonctionnement
 - De 0 à 9, compteur
 - Lorsque 9, chargement de 0



Synthèse d'automate



M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

31

Reconnaissance de séquence

ETAT PRÉSENT	ETAT FUTUR X=0	ETAT FUTUR X=1
A	A	B
B	C	B
C	A	D
D	E	B
E	A	F
F	E	B

$D0 = \sum m(4,5,7)$ avec $d = (2,6,10,14)$
 $D1 = \sum m(5,7,11)$ avec $d = (2,6,10,14)$

$D_0 = Q_2 \bar{X} + \bar{Q}_2 Q_0 X$ $D2 = X$
 $D_1 = Q_2 Q_0 \bar{X} + \bar{Q}_2 Q_1 X$ $Z = Q_2 \cdot Q_1 \cdot Q_0$

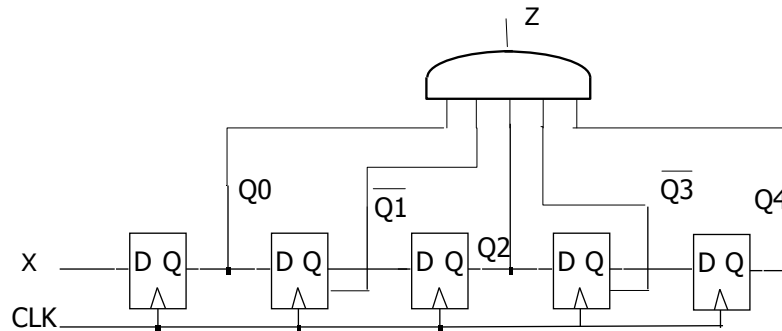
N	X	Q2	Q1	Q0	D2	D1	D0	Z
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0
3	0	0	1	1	0	0	0	0
4	0	1	0	0	0	0	1	0
5	0	1	0	1	0	1	1	0
7	0	1	1	1	0	1	1	1
8	1	0	0	0	1	0	0	0
9	1	0	1	1	1	0	1	0
11	1	0	1	1	1	1	1	0
12	1	1	0	0	1	0	0	0
13	1	1	0	1	1	0	0	0
15	1	1	1	1	1	0	0	1

M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

32

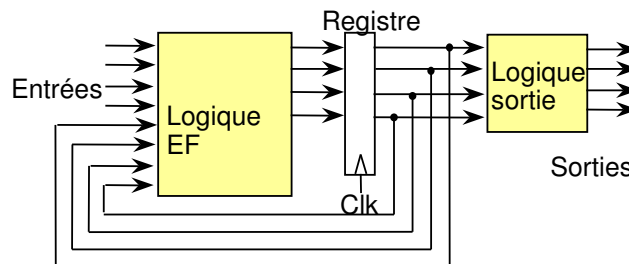
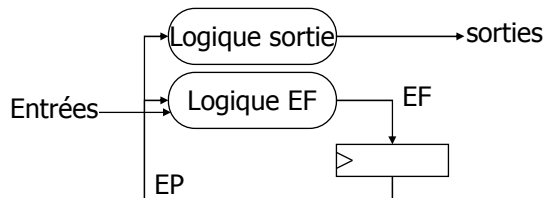
Reconnaissance de séquence (registre à décalage)



Moore versus Mealy

Moore

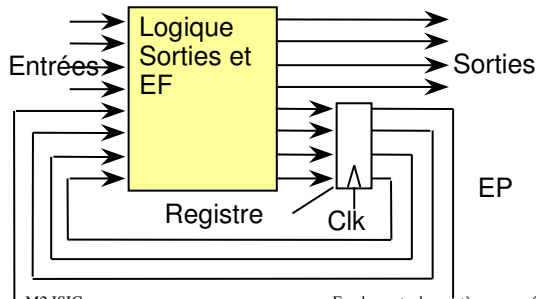
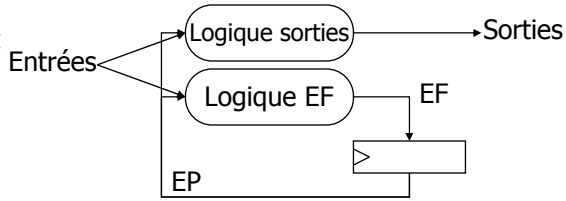
- Sorties fonction de l'EP
- Les sorties changent de manière **synchrone** avec le changement d'état



Moore versus Mealy

Mealy

- Les sorties dépendent de l'état et des entrées
- Un changement en entrée provoque un changement immédiat en sortie
- **Signaux asynchrones**

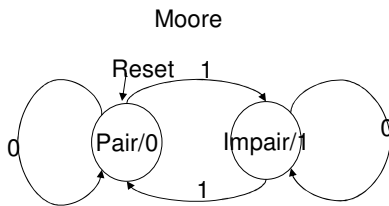


M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

35

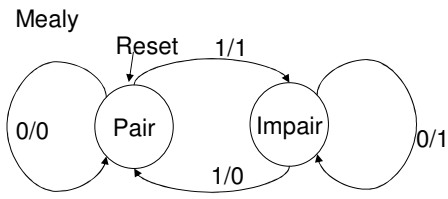
Moore/Mealy (détecteur parité impaire)



X	EP	EF	Z
0	P	P	0
0	I	I	1
1	P	I	0
1	I	P	1

$$D = X \oplus Q$$

$$Z = Q$$



X	EP	EF	Z
0	P	P	0
0	I	I	1
1	P	I	1
1	I	P	0

$$D = X \oplus Q$$

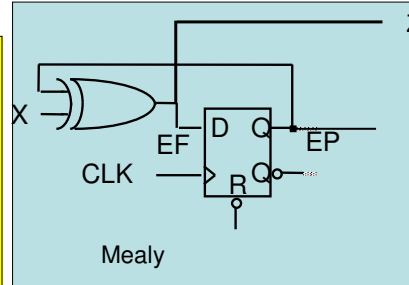
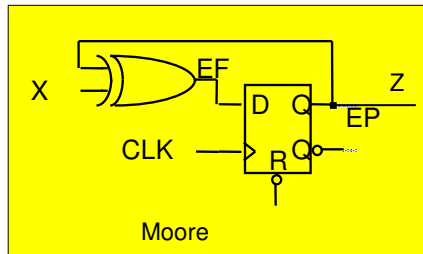
$$Z = D \oplus Q$$

M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

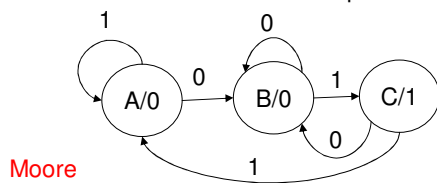
36

Moore/Mealy (détecteur parité impaire)



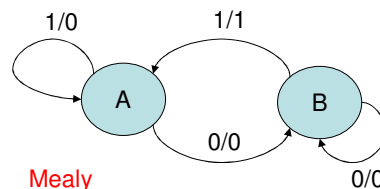
Moore/Mealy (détecteur de la séquence 01)

•Reconnaissance de la séquence 01



Moore

X	EP	EF	Z
0	A	B	0
0	B	B	0
0	C	B	1
1	A	A	0
1	B	C	0
1	C	A	1



Mealy

X	EP	EF	Z
0	A	B	0
0	B	B	0
1	A	A	0
1	B	A	1

Moore/Mealy (détecteur de la séquence 01)

X	Q1	Q0	D1	D0	Z
0	0	0	0	1	0
0	0	1	0	1	0
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	1	1	0
1	1	1	0	0	1

X	Q	F	Z
0	0	1	0
0	1	1	0
1	0	0	0
1	1	0	1

$$D_0 = \overline{X}$$

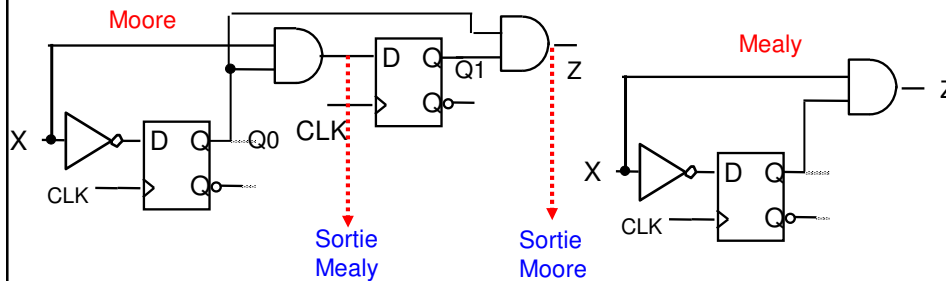
$$D_1 = X \cdot Q_0$$

$$Z = Q_1 \cdot Q_0$$

$$D = \overline{X}$$

$$Z = X \cdot Q$$

Moore/Mealy (détecteur de la séquence 01)



$$D_0 = \overline{X}$$

$$D_1 = X \cdot Q_0$$

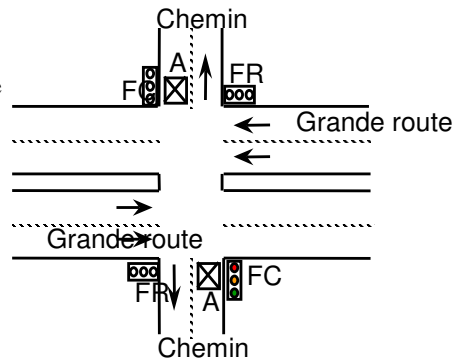
$$Z = Q_1 \cdot Q_0$$

$$D = \overline{X}$$

$$Z = X \cdot Q$$

Exemple : contrôleur de feux

- Croisement entre une grande route et un petit chemin
- Quand il n'y a pas d'automobile sur le chemin, le feu reste vert sur la grande route
- S'il y a une auto sur le chemin, au bout d'un certain temps, le feu passe à l'orange puis au rouge sur la grande route, ce qui le fait passer au vert sur le chemin. Le feu sur le chemin ne reste au vert qu'aussi longtemps qu'il y a une auto, mais jamais plus longtemps qu'un temps prédéfini.
- Même si des autos attendent sur le chemin, la grande route obtient le feu vert au moins pour un intervalle de temps.
- On suppose qu'il y a un "timer" qui génère un petit intervalle de temps (TC) et un grand intervalle (TL) en réponse à un signal de démarrage. TC est utilisé pour la durée du feu orange et TL pour la durée maximale du feu vert sur le chemin.



M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

41

Exemple du contrôleur de feux

Entrées

RAZ
A
TC
TL

Description

État initial
Automobile sur le chemin
L'intervalle de temps court est terminé
L'intervalle de temps long est terminé

Sorties

FRR, FRO, FRV
FCV, FCO, FCR
DI

Description

Feux route rouge, orange, vert
Feux chemin vert, orange, rouge
Démarrage d'un intervalle de temps

États

RV
RO
CV
CO

Description

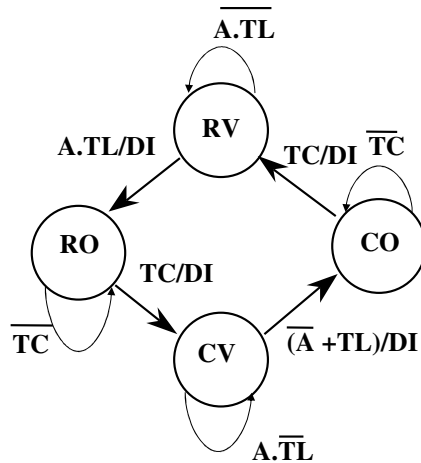
Route vert (chemin rouge)
Route orange (chemin rouge)
Chemin vert (route rouge)
Chemin orange (route rouge)

M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

42

Contrôleur feux (graphe des états)

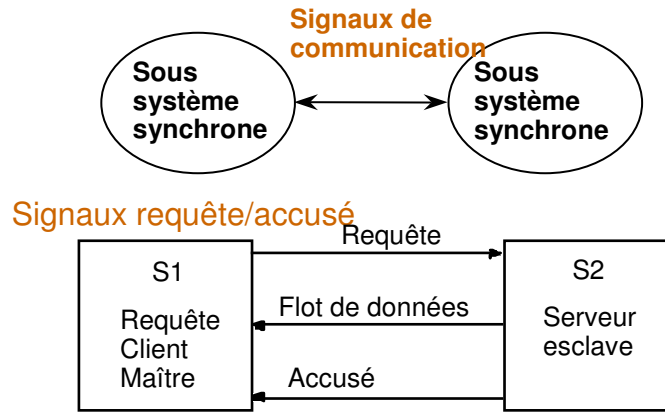


Contrôleur feux (Diagramme de transition)

État	Entrées	État futur	Feux route	Feux chemin	RAZ Compteur
RV	A=0	RV	V	R	Oui
	A.TL=0	RV	V	R	Non
	A.TL=1	RO	V	R	Oui
RO	TC=0	RO	O	R	Oui
	TC=1	CV	O	R	Non
CV	A=0	CO	R	V	Oui
	A.TL=0	CV	R	V	Non
	A.TL=1	CO	R	V	Oui
CO	TC=0	CO	R	V	Non
	TC=1	RV	R	V	Oui

Limites de l'approche synchrone

- Des circuits totalement synchrones ne sont pas possible pour de très gros systèmes à cause des dispersions d'horloge
- On partitionne le système en composants avec des horloges locales
- Ces composants communiquent via des protocoles indépendants des horloges.

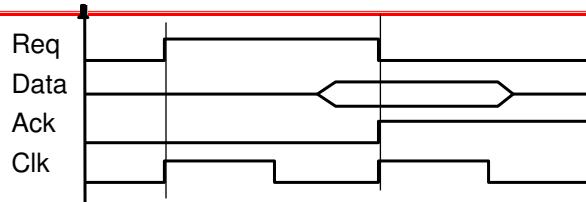


M2 ISIC
2008-2009

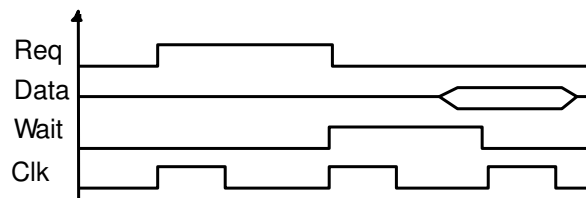
Fondements des systèmes numériques
D. Etiemble

45

Transferts synchrones



Requête de lecture du maître. L'esclave fournit les données et l'accusé en retour



Autre schéma synchrone : l'esclave émet un signal WAIT s'il ne peut répondre en 1 cycle d'horloge

M2 ISIC
2008-2009

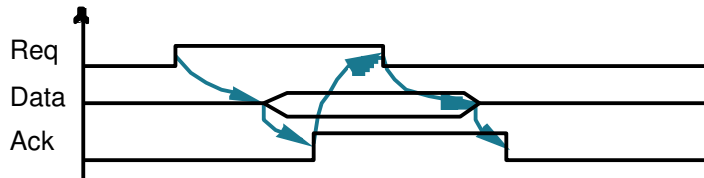
Fondements des systèmes numériques
D. Etiemble

46

Transfert asynchrone

L'information est transmise par le niveau des signaux plutôt que par les transitions. Pas de signal d'horloge

Protocole en 4 étapes



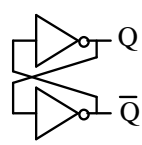
(1) Le maître émet une requête que l'esclave traite
(2) l'esclave émet un accusé quand il a fini

(3) Le maître accuse réception des données en retirant la requête
(4) l'esclave retire l'accusé

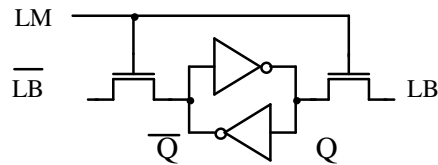
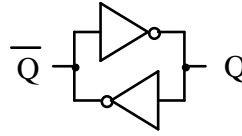
Les mémoires RAM

- Mémoires statiques (SRAM)
- Mémoires dynamiques (DRAM)

Point mémoire SRAM

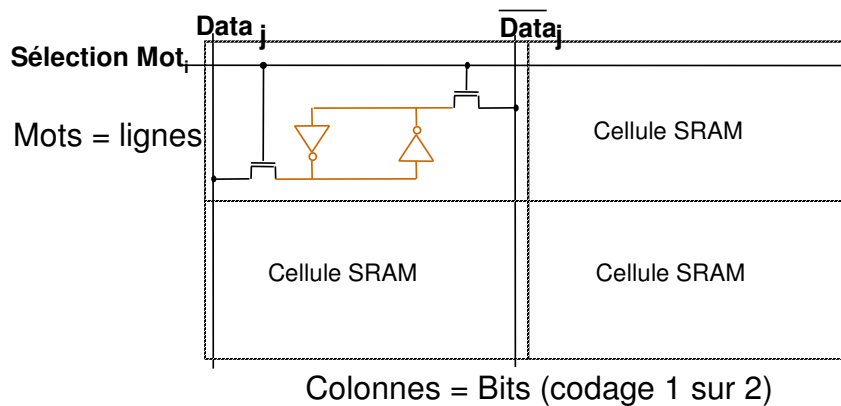


Bistable
↓

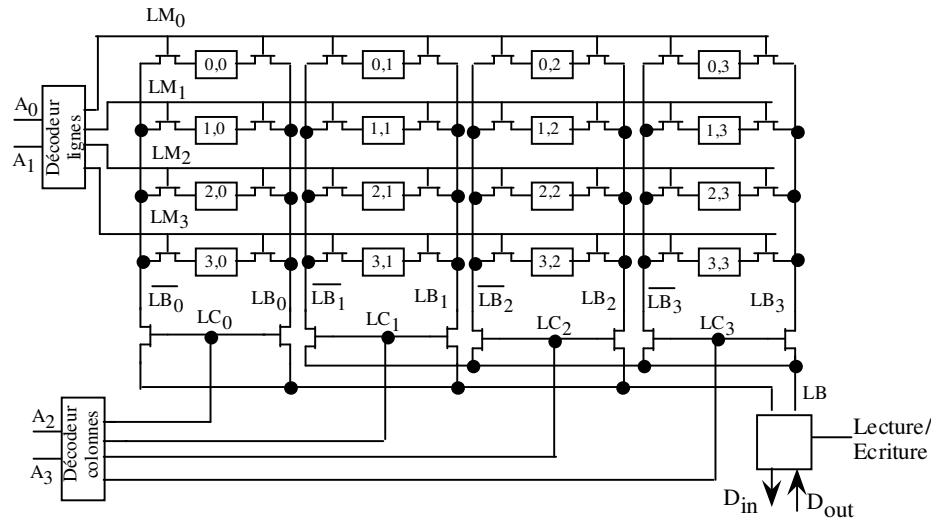


LM = 0 **Mémorisation**
 LM = 1 **Lecture**
 LM = 1 et LB/LB fixés **Écriture**

RAM statique



Principe SRAM 16 x 1 bit

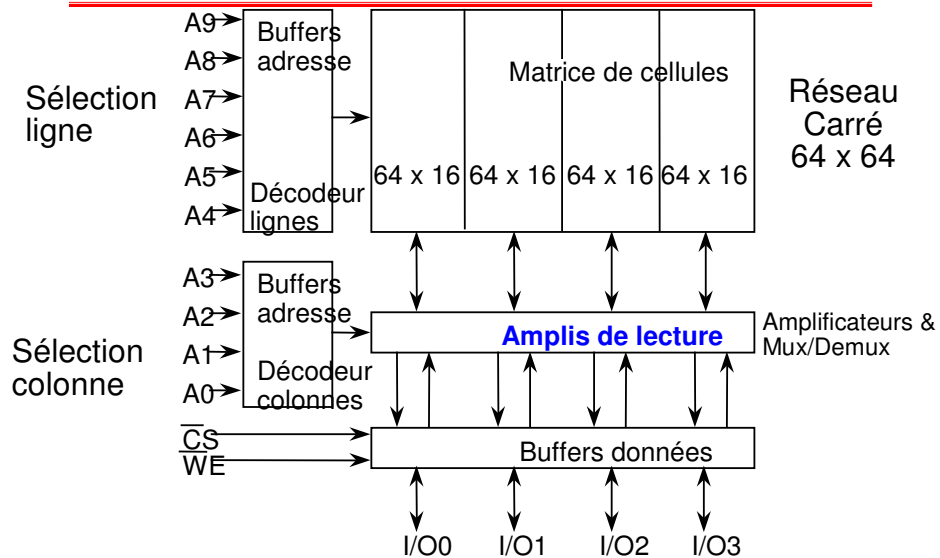


M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

51

Exemple SRAM 4 Ko (1K x4)

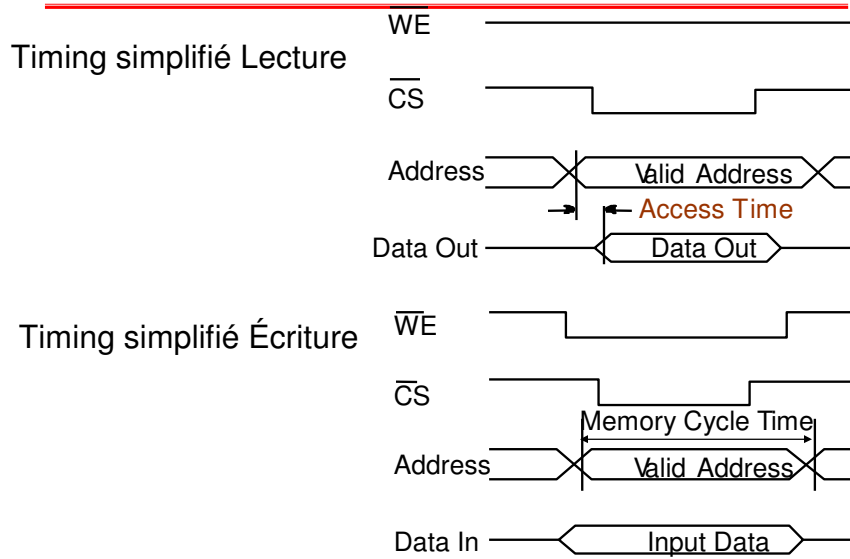


M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

52

Timing SRAM

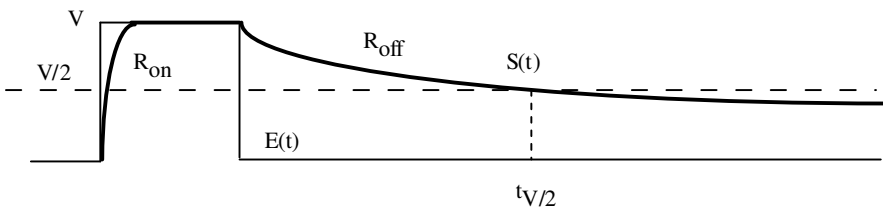
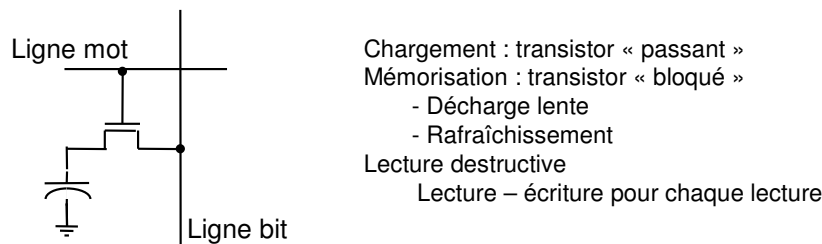


M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

53

Point mémoire DRAM



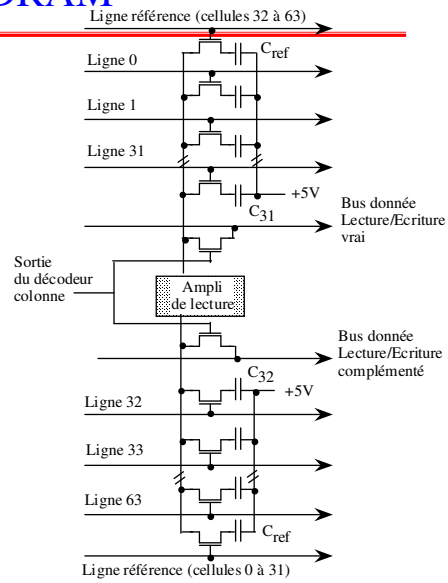
M2 ISIC
2008-2009

Fondements des systèmes numériques
D. Etiemble

54

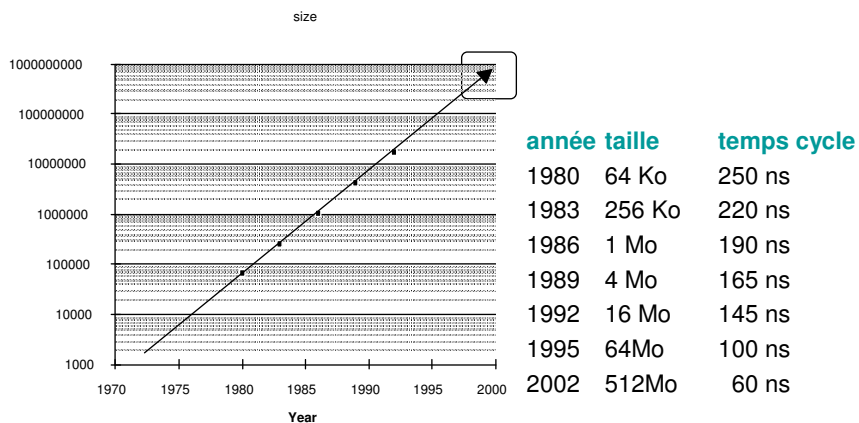
Organisation Mémoire DRAM

- Lecture différentielle
 - $C_{\text{mémoire}} \ll C_{\text{fil}}$

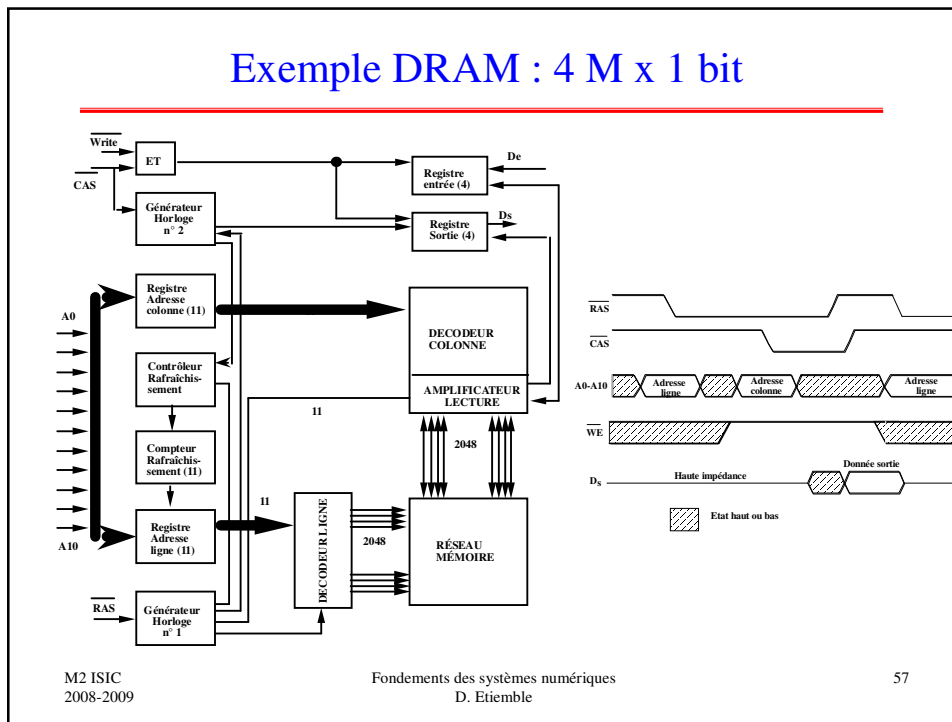


Capacité mémoires DRAM

- Par boîtier DRAM

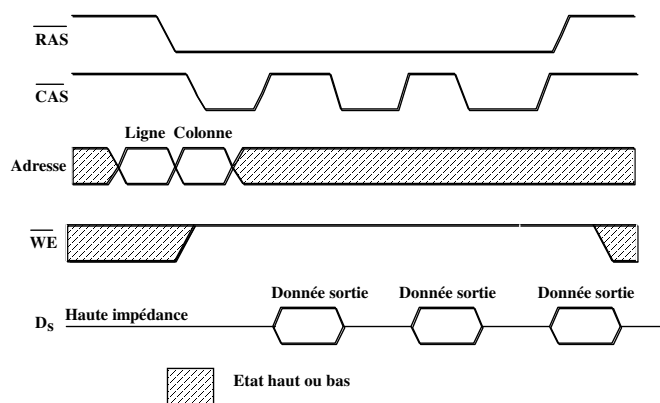


Exemple DRAM : 4 M x 1 bit

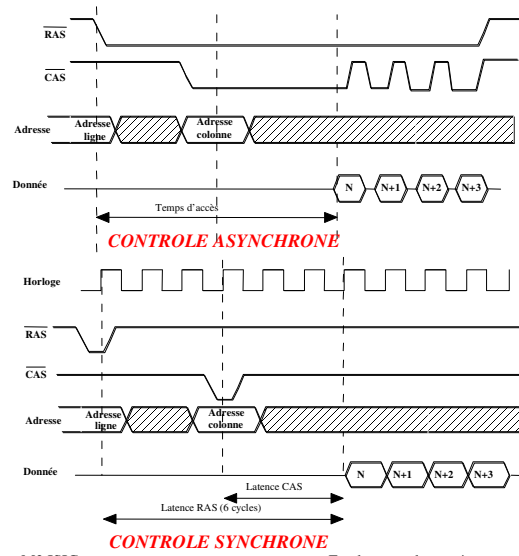


Amélioration DRAM : mode « nibble »

- Lecture de plusieurs bits successifs
 - RAS suivi de plusieurs CAS



DRAMs asynchrones et synchrones



SDRAM