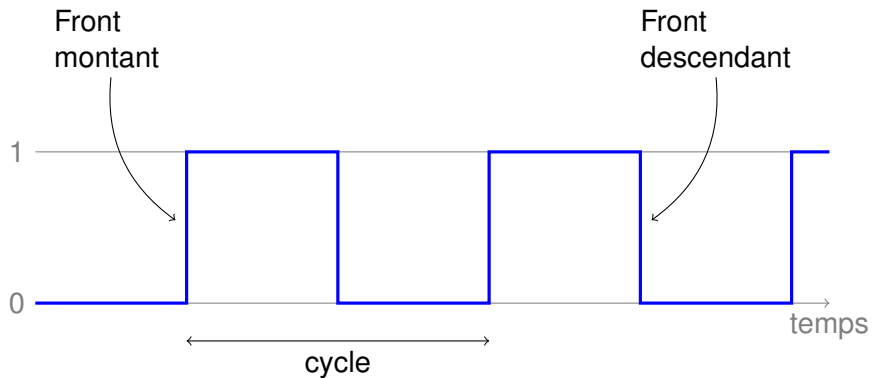


# Circuits séquentiels

Christophe Moulleron



# Horloge

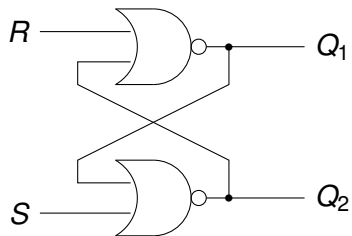


- 1 Verrous et Bascules
  - Verrous
  - Bascules
  
- 2 Circuits séquentiels synchrones
  - Banc de registres
  - Automates de Moore
  - Automates de Mealy

- 1 **Verrous et Bascules**
  - **Verrous**
  - Bascules
  
- 2 **Circuits séquentiels synchrones**
  - Banc de registres
  - Automates de Moore
  - Automates de Mealy

# Bizarrerie

Que fait le circuit suivant ?

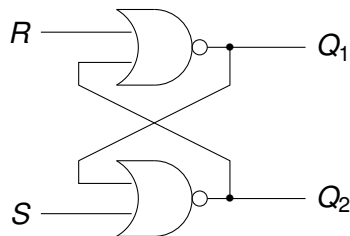


Entrées :

Sorties :

# Bizarrerie

Que fait le circuit suivant ?



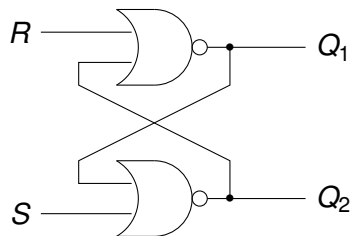
$R$	$S$	$Q_1$	$Q_2$
1	1		

Entrées :  $R$  et  $S$

Sorties :  $Q_1$  et  $Q_2$

# Bizarrerie

Que fait le circuit suivant ?



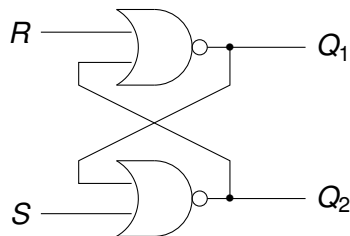
$R$	$S$	$Q_1$	$Q_2$
1	1	0	0
1	0		
0	1		
0	0		

Entrées :  $R$  et  $S$

Sorties :  $Q_1$  et  $Q_2$

# Bizarrerie

Que fait le circuit suivant ?



$R$	$S$	$Q_1$	$Q_2$
1	1	0	0
1	0	0	1
0	1	1	0
0	0	?	?

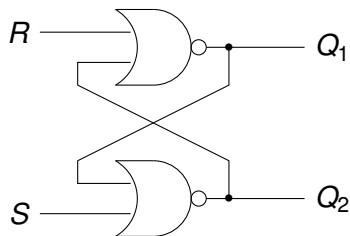
Entrées :  $R$  et  $S$  ... et  $Q_1$  et  $Q_2$

Sorties :  $Q_1$  et  $Q_2$



# Bizarrerie

Que fait le circuit suivant ?



$R$	$S$	$Q_1^{\text{old}}$	$Q_2^{\text{old}}$	$Q_1^{\text{new}}$	$Q_2^{\text{new}}$
1	1	-	-	0	0
1	0	-	-	0	1
0	1	-	-	1	0
0	0	1	1		
0	0	1	0		
0	0	0	1		
0	0	0	0		

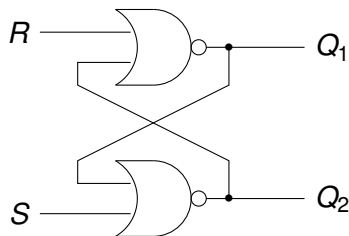
Entrées :  $R$  et  $S$

Rétrocontrôle :  $Q_1^{\text{old}}$  et  $Q_2^{\text{old}}$

Sorties :  $Q_1^{\text{new}}$  et  $Q_2^{\text{new}}$

# Bizarrerie

Que fait le circuit suivant ?



$R$	$S$	$Q_1^{\text{old}}$	$Q_2^{\text{old}}$	$Q_1^{\text{new}}$	$Q_2^{\text{new}}$
1	1	-	-	0	0
1	0	-	-	0	1
0	1	-	-	1	0
0	0	1	1	???	???
0	0	1	0	1	0
0	0	0	1	0	1
0	0	0	0	???	???

Entrées :  $R$  et  $S$

Rétrocontrôle :  $Q_1^{\text{old}}$  et  $Q_2^{\text{old}}$

Sorties :  $Q_1^{\text{new}}$  et  $Q_2^{\text{new}}$

Équilibre électrique :  $Q_1 = \overline{R + Q_2}$  et  $Q_2 = \overline{S + Q_1}$

## Bizarrerie (2)

Équilibre électrique :  $Q_1 = \overline{R + Q_2}$  et  $Q_2 = \overline{S + Q_1}$

- si  $R = 0$ , alors  $Q_1 = \overline{Q_2}$
- si  $S = 0$ , alors  $Q_2 = \overline{Q_1}$

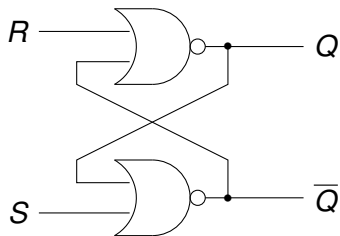
## Bizarrerie (2)

Équilibre électrique :  $Q_1 = \overline{R + Q_2}$  et  $Q_2 = \overline{S + Q_1}$

- si  $R = 0$ , alors  $Q_1 = \overline{Q_2}$
- si  $S = 0$ , alors  $Q_2 = \overline{Q_1}$

Si  $R = 0$  ou  $S = 0$  :

$$Q_1 = Q, Q_2 = \overline{Q}$$



$R$	$S$	$Q^{\text{old}}$	$Q^{\text{new}}$	$\overline{Q^{\text{new}}}$
1	0	1	0	1
1	0	0	0	1
0	1	1	1	0
0	1	0	1	0
0	0	1	1	0
0	0	0	0	1

# Bizarrie (3)

Bilan :

- cas  $R = S = 1$  à part
- valeur de  $Q^{\text{new}}$  fonction de  $R$ ,  $S$  et  $Q^{\text{old}}$
- changement de  $R$  ou  $S \Rightarrow$  mise à jour de  $Q$

valeur interdite

contrôle de  $Q$

# Verrou RS (3)

Bilan :

- cas  $R = S = 1$  à part
- valeur de  $Q^{\text{new}}$  fonction de  $R$ ,  $S$  et  $Q^{\text{old}}$
- changement de  $R$  ou  $S \Rightarrow$  mise à jour de  $Q$

valeur interdite

contrôle de  $Q$

Table de vérité compressée :

$R$	$S$	$Q^{\text{old}}$	$Q^{\text{new}}$	$\overline{Q^{\text{new}}}$
1	0	$x$	0	1
0	1	$x$	1	0
0	0	$x$	$x$	$\overline{x}$

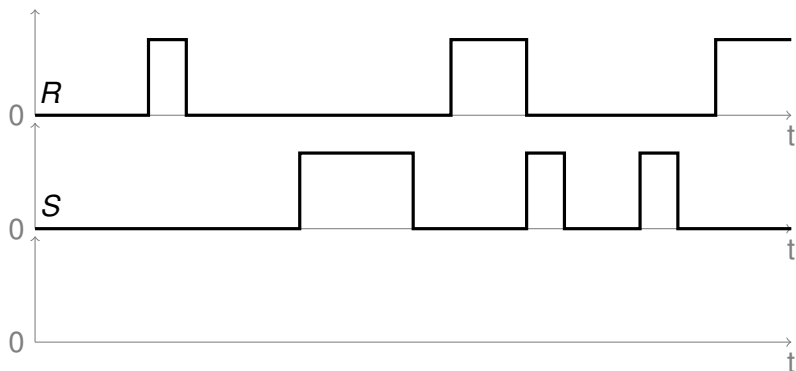
- $R$  pour reset

$\rightsquigarrow$  si  $R$  passe à 1,  $Q$  passe à 0

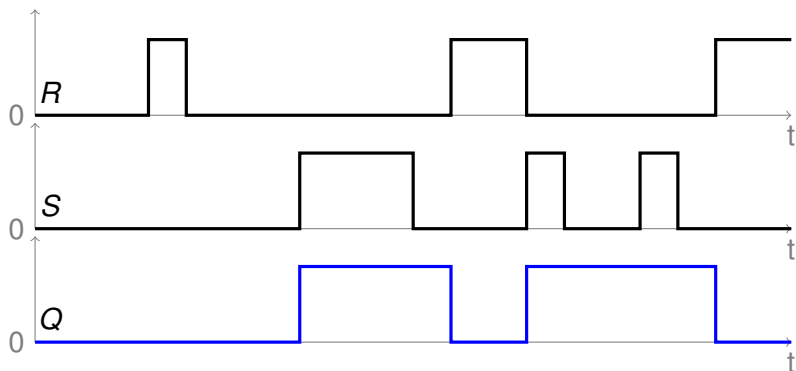
- $S$  pour set

$\rightsquigarrow$  si  $S$  passe à 1,  $Q$  passe à 1

# Chronogramme pour le verrou $RS$



# Chronogramme pour le verrou *RS*





Verrou =

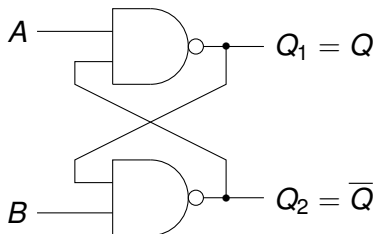
latch (in english)

- contrôle sur 1 bit
- circuit asynchrone
- **réaction immédiate** au changement des entrées  
à traversée de portes près

Nombreuses solutions :

- différents types de contrôle
- différences de portes logiques / agencement de transistors

## Autre exemple



Valeur interdite en entrée :

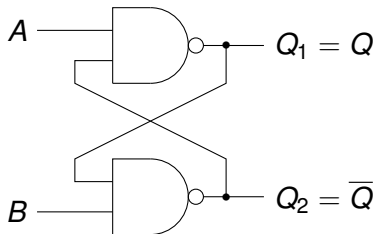
$$A = B = 0$$

Équilibre électrique :

$$\rightsquigarrow Q_1 = \overline{Q_2}$$

Table de vérité ?

## Autre exemple



Valeur interdite en entrée :

$$A = B = 0$$

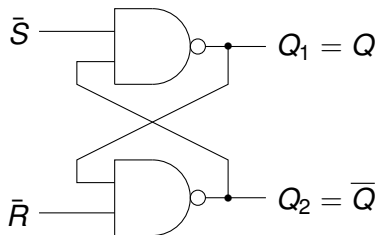
Équilibre électrique :

$$\rightsquigarrow Q_1 = \bar{Q}_2$$

Table de vérité :

A	B	$Q^{\text{old}}$	$Q^{\text{new}}$	$\bar{Q}^{\text{new}}$
0	1	x	1	0
1	0	x	0	1
1	1	x	x	$\bar{x}$

## Autre exemple : Verrou $\bar{R}\bar{S}$



Valeur interdite en entrée :

$$\bar{S} = \bar{R} = 0$$

Équilibre électrique :

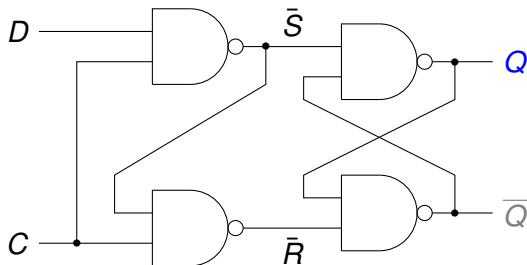
$$\rightsquigarrow Q_1 = \bar{Q}_2$$

Table de vérité :

$\bar{S}$	$\bar{R}$	$Q^{\text{old}}$	$Q^{\text{new}}$	$\bar{Q}^{\text{new}}$
0	1	x	1	0
1	0	x	0	1
1	1	x	x	$\bar{x}$

# Vers un registre de 1 bit

Verrou  $D$  :



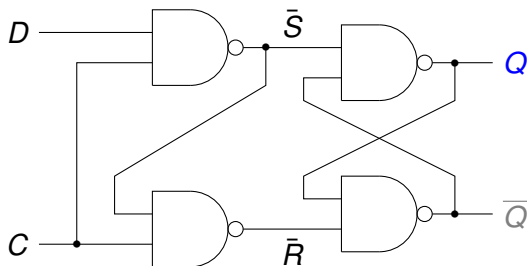
$C$	$D$	$\bar{S}$	$\bar{R}$	$Q^{\text{old}}$	$Q^{\text{new}}$
0	0			$x$	
0	1			$x$	
1	0			$x$	
1	1			$x$	

•  $C$  = horloge

clock

# Vers un registre de 1 bit

Verrou  $D$  :

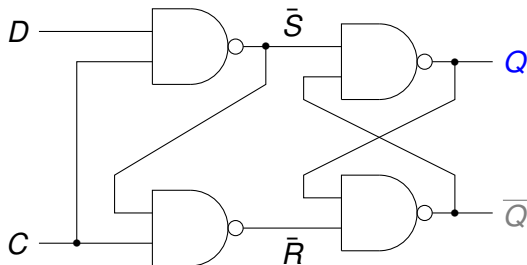


$C$	$D$	$\bar{S}$	$\bar{R}$	$Q^{\text{old}}$	$Q^{\text{new}}$
0	0	1	1	$x$	
0	1	1	1	$x$	
1	0	1	0	$x$	
1	1	0	1	$x$	

- $C$  = horloge clock
- $(\bar{S}, \bar{R})$  jamais en valeur interdite

# Vers un registre de 1 bit

Verrou  $D$  :

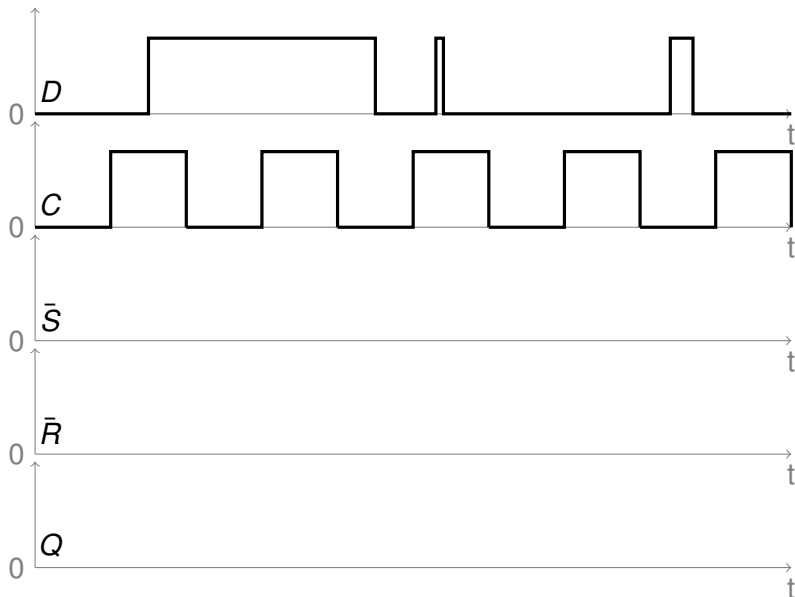


$C$	$D$	$\bar{S}$	$\bar{R}$	$Q^{\text{old}}$	$Q^{\text{new}}$
0	0	1	1	$x$	$x$
0	1	1	1	$x$	$x$
1	0	1	0	$x$	0
1	1	0	1	$x$	1

- $C$  = horloge clock
- $(\bar{S}, \bar{R})$  jamais en valeur interdite
- contrôle =  
 $Q \leftarrow D$  quand  $C = 1$

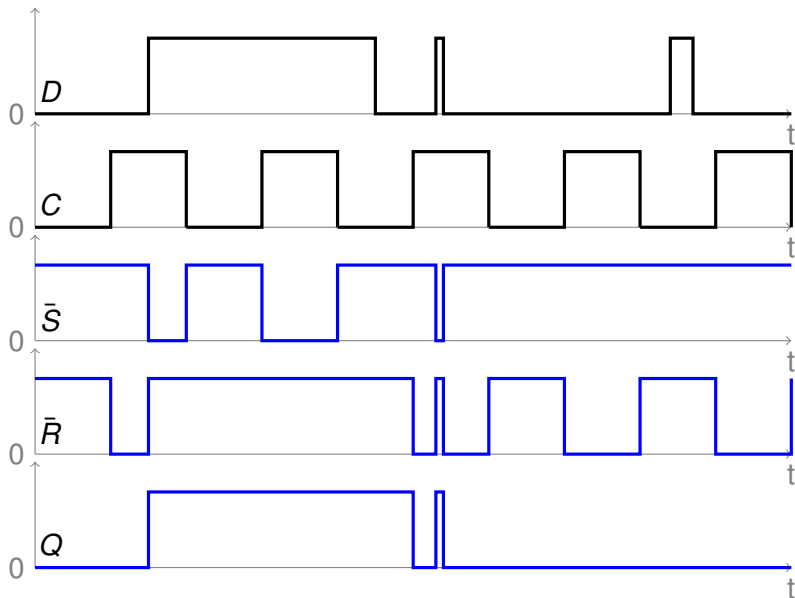
**Affectation de valeur** quand horloge à 1

# Chronogramme pour le verrou $D$





# Chronogramme pour le verrou $D$



- 1 **Verrous et Bascules**
  - Verrous
  - **Bascules**
  
- 2 **Circuits séquentiels synchrones**
  - Banc de registres
  - Automates de Moore
  - Automates de Mealy

Limites du verrou  $D$  :

- changement **instantané** tant que  $C = 1$
- valeur variable au cours du cycle

↔ pb. commun à tous les verrous

# Limite des verrous

Limites du verrou  $D$  :

- changement **instantané** tant que  $C = 1$
- valeur variable au cours du cycle

↔ pb. commun à tous les verrous

Besoin pour un **circuit synchrone** :

- valeur **stable sur un cycle**
- changement uniquement lors d'un front

↔ passage des verrous aux bascules

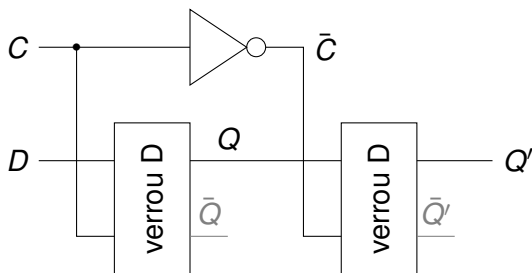
vraie sauvegarde

flip-flop (in english)

# Bascule D (1)

**Idée** : utiliser 2 verrous D en série

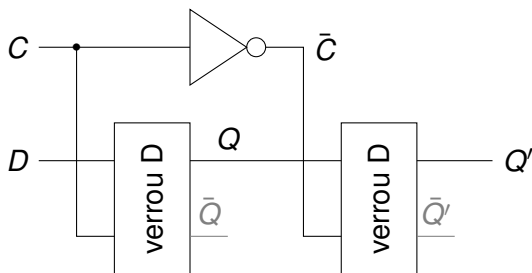
maître + esclave



# Bascule D (1)

**Idée** : utiliser 2 verrous D en série

maître + esclave



En régime permanent :

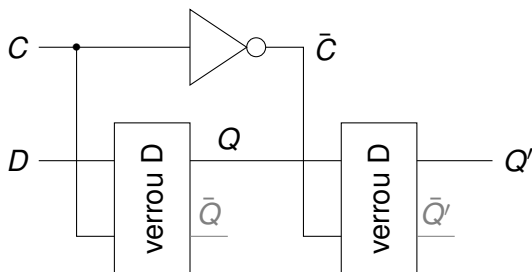
$C = 1$      $\bar{C} = 0$  donc  $Q'$  constant

$C = 0$      $Q$  constant donc  $Q'$  constant

# Bascule D (1)

**Idée** : utiliser 2 verrous D en série

maître + esclave



Front montant :

$C$  passe de 0 à 1

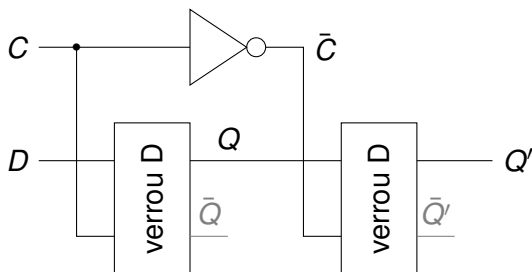
- $\bar{C}$  passe de 1 à 0
- $Q^{\text{old}} = Q^{\text{new}}$ , donc  $Q'$  constant
- $Q$  mis à jour

$Q \leftarrow D$

# Bascule D (1)

**Idée** : utiliser 2 verrous D en série

maître + esclave



Front descendant :

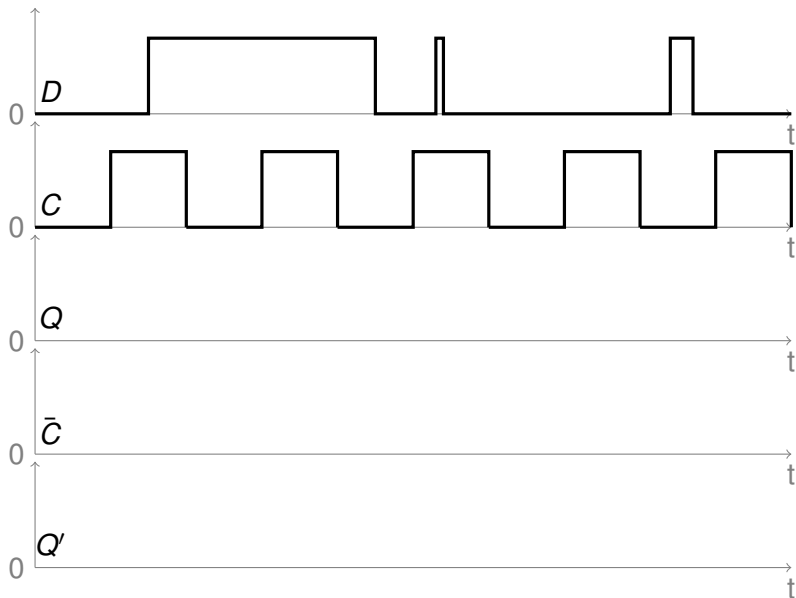
$C$  passe de 1 à 0

- $\bar{C}$  passe de 0 à 1
- $Q$  constant
- $Q'$  mis à jour

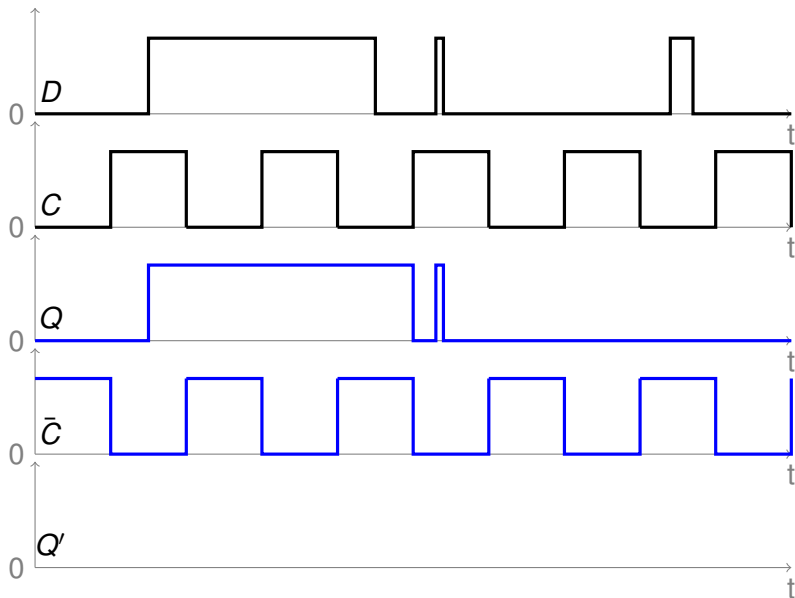
$Q' \leftarrow Q$



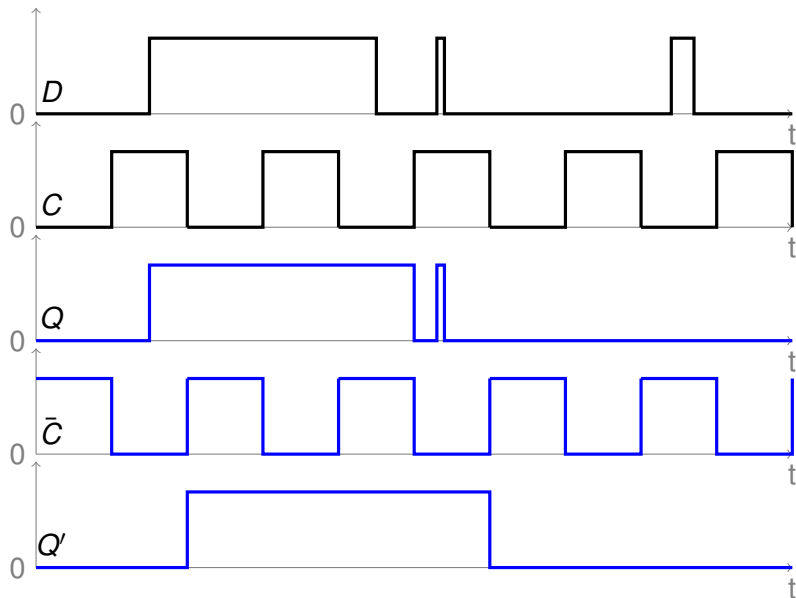
## Bascule D (2)



## Bascule D (2)



## Bascule D (2)



## Bascule D (3)

### Bilan :

- $Q$  variable uniquement quand  $C = 1$
- $Q'$  constant **sauf au front descendant**

↪ Bascule  $D$  synchronisée sur le front descendant

Beaucoup d'autres solutions possibles :

- bascule  $D$  synchronisée sur le front montant déplacer le NOT
- bascule  $D$  avec un design autre que maître/esclave
- bascule avec un contrôle différent bascule T

## 1 Verrous et Bascules

- Verrous
- Bascules

## 2 Circuits séquentiels synchrones

- Banc de registres
- Automates de Moore
- Automates de Mealy

# Circuits séquentiels synchrones

1 bascule  $D$  = registre de 1 bit

- résultat d'un calcul stocké dans  $Q$
- valeur restituée via  $Q'$  pendant tout le cycle suivant

$n$  bascules  $D$  en parallèle = registre de  $n$  bits

# Circuits séquentiels synchrones

1 bascule  $D$  = registre de 1 bit

- résultat d'un calcul stocké dans  $Q$
- valeur restituée via  $Q'$  pendant tout le cycle suivant

$n$  bascules  $D$  en parallèle = registre de  $n$  bits

Circuit séquentiel synchrone :

- calculs = circuit combinatoire
- résultats intermédiaires stockés dans un registre
- rythme dicté par l'**horloge**
- flot d'entrées/sorties

## 1 Verrous et Bascules

- Verrous
- Bascules

## 2 Circuits séquentiels synchrones

- Banc de registres
- Automates de Moore
- Automates de Mealy



# Banc de registres

Banc de registres =  $2^k$  registres de taille  $n$  bits en parallèle

Opérations : réalisables en parallèle

① lecture de la valeur d'un registre

② m à j de la valeur pour un registre

# Banc de registres

Banc de registres =  $2^k$  registres de taille  $n$  bits en parallèle

Opérations :

réalisables en parallèle

① lecture de la valeur d'un registre

$k$  entrées +  $n$  sorties

- ▶ choix d'un registre parmi  $k$
- ▶ multiplexeur  $2^k n \rightarrow n$

② m à j de la valeur pour un registre

# Banc de registres

Banc de registres =  $2^k$  registres de taille  $n$  bits en parallèle

Opérations : réalisables en parallèle

① lecture de la valeur d'un registre k entrées + n sorties

- ▶ choix d'un registre parmi  $k$
- ▶ multiplexeur  $2^k n \rightarrow n$

② mäj de la valeur pour un registre (k + 1) entrées + n entrées

- ▶ écriture dans 1 seul registre
- ▶ décodeur  $k \rightarrow 2^k$
- ▶ opération provoquée par l'horloge + bit de contrôle

écriture facultative

## Banc de registre (2)

Exemple d'un banc de 8 registres sur 8 bits :

↪ `cf register.lgf`

## Banc de registre (2)

Exemple d'un banc de 8 registres sur 8 bits :

↪ `cf register.lgf`

### Mémoire SRAM :

- même principe que le banc de registres
- mémoire volatile
- faible quantité de données
- très coûteuse
- très rapide
- surtout utilisée pour les caches

à plus grande échelle

## 1 Verrous et Bascules

- Verrous
- Bascules

## 2 Circuits séquentiels synchrones

- Banc de registres
- **Automates de Moore**
- Automates de Mealy

# Automate de Moore (1)

Automate =  $(Q, A, q_0, \delta, F)$

- ensemble d'états  $Q$
- alphabet d'entrée  $A$
  
- état initial  $q_0 \in Q$
- fonction de transition  $\delta : Q \times A \rightarrow Q$
- ensemble d'états terminaux  $F$

# Automate de Moore (1)

Automate de Moore =  $(Q, A, B, q_0, \delta, \lambda)$

- ensemble d'états  $Q$
- alphabet d'entrée  $A$
- alphabet de sortie  $B$
- état initial  $q_0 \in Q$
- fonction de transition  $\delta : Q \times A \rightarrow Q$
- fonction de sortie  $\lambda : Q \rightarrow B$

Automate fini + production d'une sortie en fonction de l'état courant

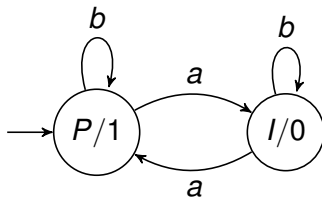


# Automate de Moore (2)

Exemple :

- $A = \{a, b\}$
- $B = \{0, 1\}$
- sortie = 1 ssi nombre pair de  $a$  lus

représentation graphique

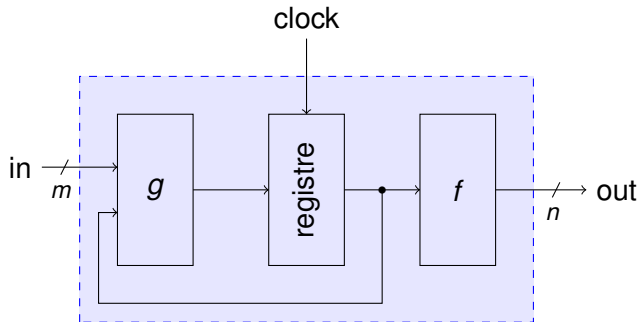


Entrée : a b b a b a ...

Sortie : 0 0 0 1 1 0 ...

# Automate de Moore → circuit (1)

Schéma général du circuit :



- $f / g =$  circuits combinatoires
- registre pour stocker l'état courant  $q_t$
- $out_t = f(q_t)$
- $q_{t+1} = g(q_t, in_t)$

$$\lambda : Q \rightarrow B$$

$$\delta : Q \times A \rightarrow Q$$

# Automate de Moore $\rightarrow$ circuit (2)

Conversion d'un automate de Moore en circuit :

- 1 Choisir un code pour chaque état  
 $\rightsquigarrow$  registre sur  $k$  bits où  $|Q| \leq 2^k$

$P : 0, I : 1$   
 $k = 1$

# Automate de Moore $\rightarrow$ circuit (2)

Conversion d'un automate de Moore en circuit :

- 1 Choisir un code pour chaque état

$\rightsquigarrow$  registre sur  $k$  bits où  $|Q| \leq 2^k$

$P : 0, I : 1$

$k = 1$

- 2 Choisir un code pour chaque symbole de  $A$  et  $B$

$a : 0, b : 1$

# Automate de Moore $\rightarrow$ circuit (2)

Conversion d'un automate de Moore en circuit :

1 Choisir un code pour chaque état

$\rightsquigarrow$  registre sur  $k$  bits où  $|Q| \leq 2^k$

$P : 0, I : 1$

$k = 1$

2 Choisir un code pour chaque symbole de  $A$  et  $B$

$a : 0, b : 1$

3 Faire le circuit combinatoire pour  $f$

- ▶ dresser la table de vérité
- ▶ méthode directe ou Karnaugh

$\text{out} = \overline{\text{state}}$

# Automate de Moore $\rightarrow$ circuit (2)

Conversion d'un automate de Moore en circuit :

1 Choisir un code pour chaque état

$\rightsquigarrow$  registre sur  $k$  bits où  $|Q| \leq 2^k$

$P : 0, I : 1$

$k = 1$

2 Choisir un code pour chaque symbole de  $A$  et  $B$

$a : 0, b : 1$

3 Faire le circuit combinatoire pour  $f$

- ▶ dresser la table de vérité
- ▶ méthode directe ou Karnaugh

$out = \overline{state}$

4 Faire le circuit combinatoire pour  $g$

$out' = \overline{in \oplus out}$

## 1 Verrous et Bascules

- Verrous
- Bascules

## 2 Circuits séquentiels synchrones

- Banc de registres
- Automates de Moore
- Automates de Mealy

# Automate de Mealy (1)

Automate de Mealy =  $(Q, A, B, q_0, \delta, \lambda)$

- ensemble d'états  $Q$
- alphabets d'entrée  $A$  et de sortie  $B$
- état initial  $q_0 \in Q$
- fonction de transition  $\delta : Q \times A \rightarrow Q$
- fonction de sortie  $\lambda : Q \times A \rightarrow B$



# Automate de Mealy (1)

Automate de Mealy =  $(Q, A, B, q_0, \delta, \lambda)$

- ensemble d'états  $Q$
- alphabets d'entrée  $A$  et de sortie  $B$
- état initial  $q_0 \in Q$
- fonction de transition  $\delta : Q \times A \rightarrow Q$
- fonction de sortie  $\lambda : Q \times A \rightarrow B$

En résumé :

**Moore** sortie dépendant uniquement de l'état courant

**Mealy** sortie dépendant de l'état courant et de l'entrée

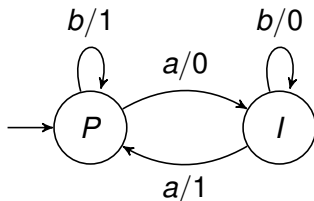
# Automate de Mealy (2)

Représentation graphique :

- $\delta$  et  $\lambda$  de même type
- sorties **sur les arêtes**

Exemple :

- $A = \{a, b\}$ ,  $B = \{0, 1\}$
- sortie = 1 ssi nombre pair de  $a$  lus



# Moore VS Mealy

Moore :

- ✓ valeur de sortie stable sur un cycle
- ✓ plus sûr

Mealy :

- ✗ réaction immédiate au changement de l'entrée

# Moore VS Mealy

Moore :

- ✓ valeur de sortie stable sur un cycle
- ✓ plus sûr
- ✗ résultat visible au cycle suivant

Mealy :

- ✗ réaction immédiate au changement de l'entrée
- ✓ plus réactif

# Moore VS Mealy

Moore :

- ✓ valeur de sortie stable sur un cycle
- ✓ plus sûr
- ✗ résultat visible au cycle suivant
- ✗ nécessite davantage d'états que Mealy

1 sortie / état

Mealy :

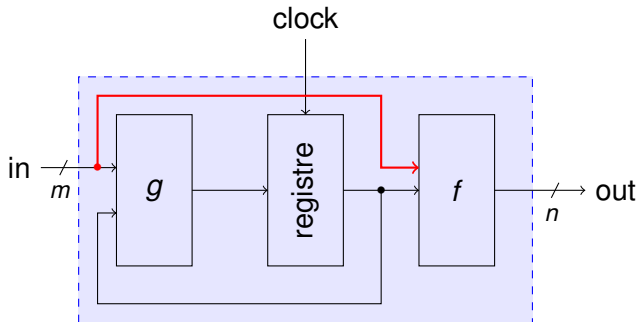
- ✗ réaction immédiate au changement de l'entrée
- ✓ plus réactif
- ✓ moins d'états que Moore

1 sortie / arête

# Automate de Mealy $\rightarrow$ circuit

Conversion similaire à celle pour Moore

Schéma général du circuit :



# Exemple : feux de circulation (1)

Situation :

- croisement de deux routes
- deux paires de feux de circulation
- voitures en attente sur chaque axe
- feu vert seulement si voiture en attente
- au moins un feu rouge

axes *NS* et *EW*

$F_{NS}$ ,  $F_{EW}$

$V_{NS}$ ,  $V_{EW}$

Cas où voitures en attente sur les 2 axes ?

# Exemple : feux de circulation (1)

Situation :

- croisement de deux routes
- deux paires de feux de circulation
- voitures en attente sur chaque axe
- feu vert seulement si voiture en attente
- au moins un feu rouge

axes *NS* et *EW*

$F_{NS}$ ,  $F_{EW}$

$V_{NS}$ ,  $V_{EW}$

Cas où voitures en attente sur les 2 axes ?

- si 2 feux rouges,  $F_{NS}$  passe au vert
- sinon, on change les feux

axe *NS* prioritaire

équité



## Exemple : feux de circulation (2)

Encodage :

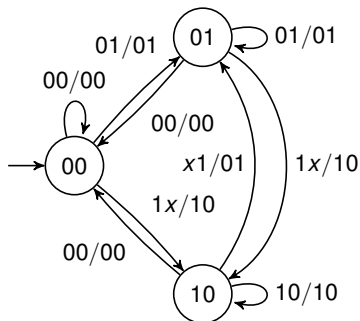
- vert = 1, rouge = 0
- voiture en attente = 1, pas de voiture = 0

$F_{NS}$  et  $F_{EW}$

$V_{NS}$  et  $V_{EW}$

Automate de Mealy :

- état = valeur des feux  $F_{NS}$  et  $F_{EW}$
- état initial = feux au rouge = 00
- lecture = attentes  $V_{NS}$  et  $V_{EW}$
- écriture = nouvelle valeur des feux



↪ cf feux.lgf