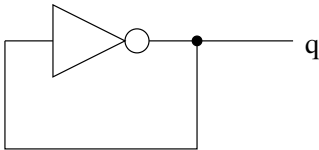


# Architecture des ordinateurs

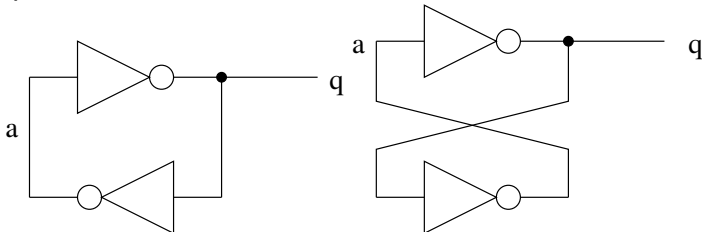
Circuits pour mémoriser

# Boucle d'inverseurs

- ▶ Circuit instable : la sortie  $q$  s'inverse sans arrêt



- ▶ Avec un deuxième inverseur : le signal se stabilise  
 $q = \neg a$



- ▶ Le signal est "mémorisé" par le circuit, mais il ne peut pas être modifié.

## La bascule RS (Reset, Set) (1/2)

- ▶ On utilise des portes NON-OU :  
le signal  $a$  est inversé si  $b = 0$  sinon le signal est remis à 0.

$a$	$b$	$\overline{a + b}$
0	0	1
0	1	0
1	0	0
1	1	0

- ▶ On va utiliser les portes NON-OU à la place des portes NON pour pouvoir ajouter des signaux SET et RESET pour contrôler la mémorisation.

## La bascule RS (Reset, Set) (2/2)

- But : mémoriser le signal S et l'avoir en sortie ou remettre à 0 quand R est à 1

départ :  $S = 0$  et  $R = 0 \rightarrow q = 0$

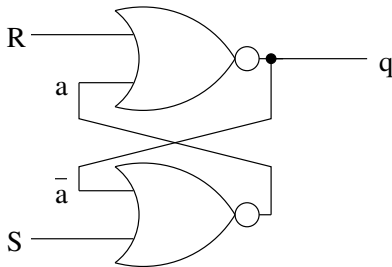
set :  $S = 0$  et  $R = 1 \rightarrow q = 0$

$S = 0$  et  $R = 0 \rightarrow q = 0$

reset :  $S = 1$  et  $R = 0 \rightarrow q = 1$

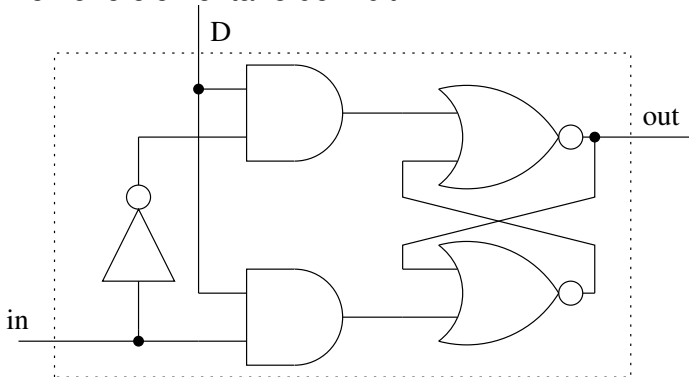
$S = 0$  et  $R = 0 \rightarrow q = 1$

**état instable :  $S = 1$  et  $R = 1 !!!$**



# Bascule D

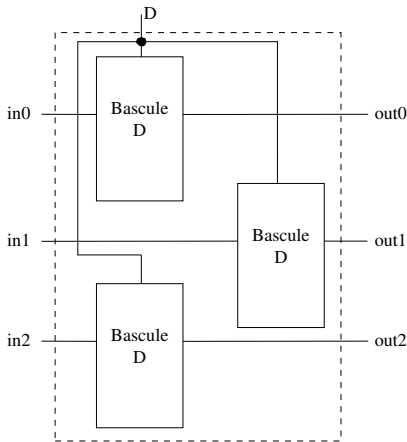
- ▶ Mémoire élémentaire de 1 bit



- ▶ L'état interdit est évité
- ▶ Souvent le signal D est contrôlé par une *horloge*

# Mot mémoire

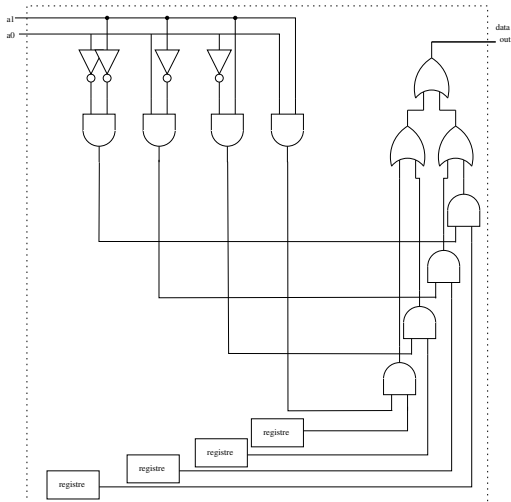
- Construction d'un mot de 3 bits à partir de 3 bascules D



- 8, 16, 32 ou 64 bits : multiplier, sur le même modèle, le nombre de bascules D et de signaux en entrée et en sortie.

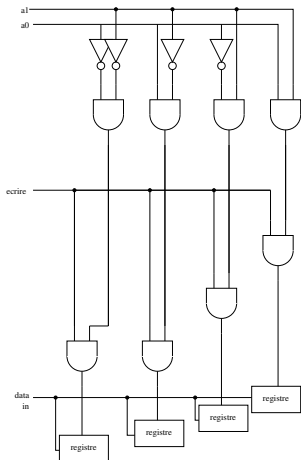
# Lecture dans une mémoire de $4 \times 1$ bit

- ▶ a1a0 : adresse mémoire que l'on souhaite lire
- ▶ registres : mots mémoire de 1 bit mémorisés
- ▶ data out : sortie



# Ecriture dans une mémoire de $4 \times 1$ bit

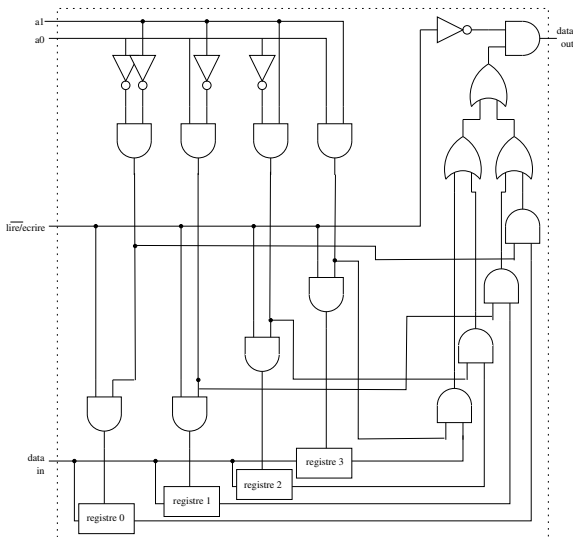
- ▶ a1a0 : adresse mémoire que l'on souhaite écrire
- ▶ data in : la donnée à mémoriser
- ▶ écrire : un signal qui active la mémorisation (D)





# Lecture/Écriture dans une mémoire de $4 \times 1$ bit

- $\overline{\text{lire/écrire}}$  : signal qui active la mémorisation ou la sortie



# Mémoire $2^k \times n$ bits

- ▶ on peut augmenter la taille et le nombre des registres
- ▶ mémoire  $2^k \times n$  bits :  
     $2^k$  signaux d'entrée codés sur  $k$  bits,  
    chaque signal contrôle  $n$  bascules D formant 1 registre,  
     $n$  signaux *data in* et  $n$  sorties
- ▶ vue simplifiée d'une mémoire : modèle proche des registres qu'on trouve dans l'unité de calcul ou l'unité de contrôle

# Les différents types de mémoire

- ▶ **mémoire RAM** (*Random Access Memory*) : accès direct à une information (par opposition aux bandes magnétiques, HDD, CDROM et DVD : accès en séquence)
- ▶ **mémoire statique** : à base de bascules D, mémoire chère, rapide, de faible capacité (mémoire cache de niveau 2)
- ▶ **Mémoire ROM** (*Read Only Memory*) et variantes
  - ▶ **PROM**, (*Programmable ROM*) : fusibles qu'on fait fondre
  - ▶ **EPROM** (*Erasable PROM*) : effaçable par une exposition aux ultraviolets
  - ▶ **EEPROM** (*Electrically Erasable PROM*) : effaçable par un courant électrique (utilisé pour stocker le BIOS)
  - ▶ **Flash** : sorte d'EEPROM, on y efface les données par bloc et non par octet, s'altère après quelques milliers ou dizaines de milliers d'écritures.
- ▶ **NVRAM** (*Non Volatile RAM*) ou Mémoire CMOS (sur les PC) : mémoires statiques ordinaires alimentées par des batteries