

Architecture de Ordinateurs

Module 1

Notions d'électronique numérique

(électronique pour informaticiens)

Dr. Yannick HERVE

V 1.0

Les notions plus détaillées seront présentées
dans le cours de Morgan Madec

Définitions

- Informatique : traiter de l'information numérique avec une machine dédiée
- L'information de base est codée sous forme de bit (architecture classique) ou « Binary digiT »
- La discipline qui traite du matériel est l'électronique numérique

En français l'adjectif *digital* se rapporte à *doigt*
(ex : empreinte digitale)

In english, the word *digital* refers to *digit*

Les pionniers

- Gottfried Wilhelm Leibnitz (1646-1716)
Le binaire comme outil efficace de calcul (1703)
- Georges Boole (1815-1864)
Algèbre binaire (1847 et 1854)
- Claude Shannon (1916-2001)
Théorie de l'information (1940)
- John Von Neumann (1903-1957)
... informatique ...
- Alan Turing (1912-1954)
Décidabilité, fonction calculable

Bit logique

- Bit $\in [0,1]$
- Notion **symbolique** du tiers exclu
(l'un ou l'autre, pas l'un et l'autre ni autre chose)
- Représentation (symbolique) de
 - Vrai/faux
 - Présent/absent
 - Ouvert/fermé
 - ...

Bit physique

Bit logique \Leftrightarrow valeur physique

- Tension

- La plus élevée = 1 \rightarrow logique positive
- La moins élevée = 1 \rightarrow logique négative

- Courant

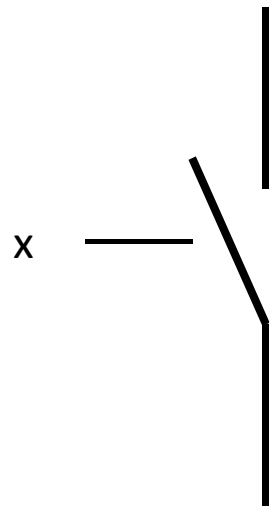
- 4/20 mA (bruit, longue distance)

- Fréquence

- Phase

- ...

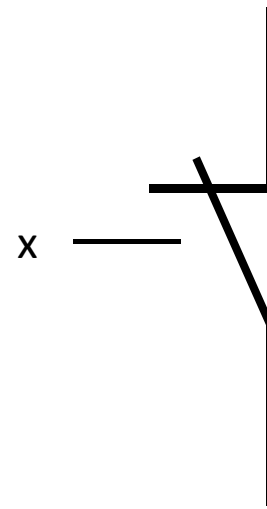
Logique des interrupteurs



Piloté à la
fermeture

$x = +V \rightarrow$ fermé

$x = 0 \rightarrow$ ouvert

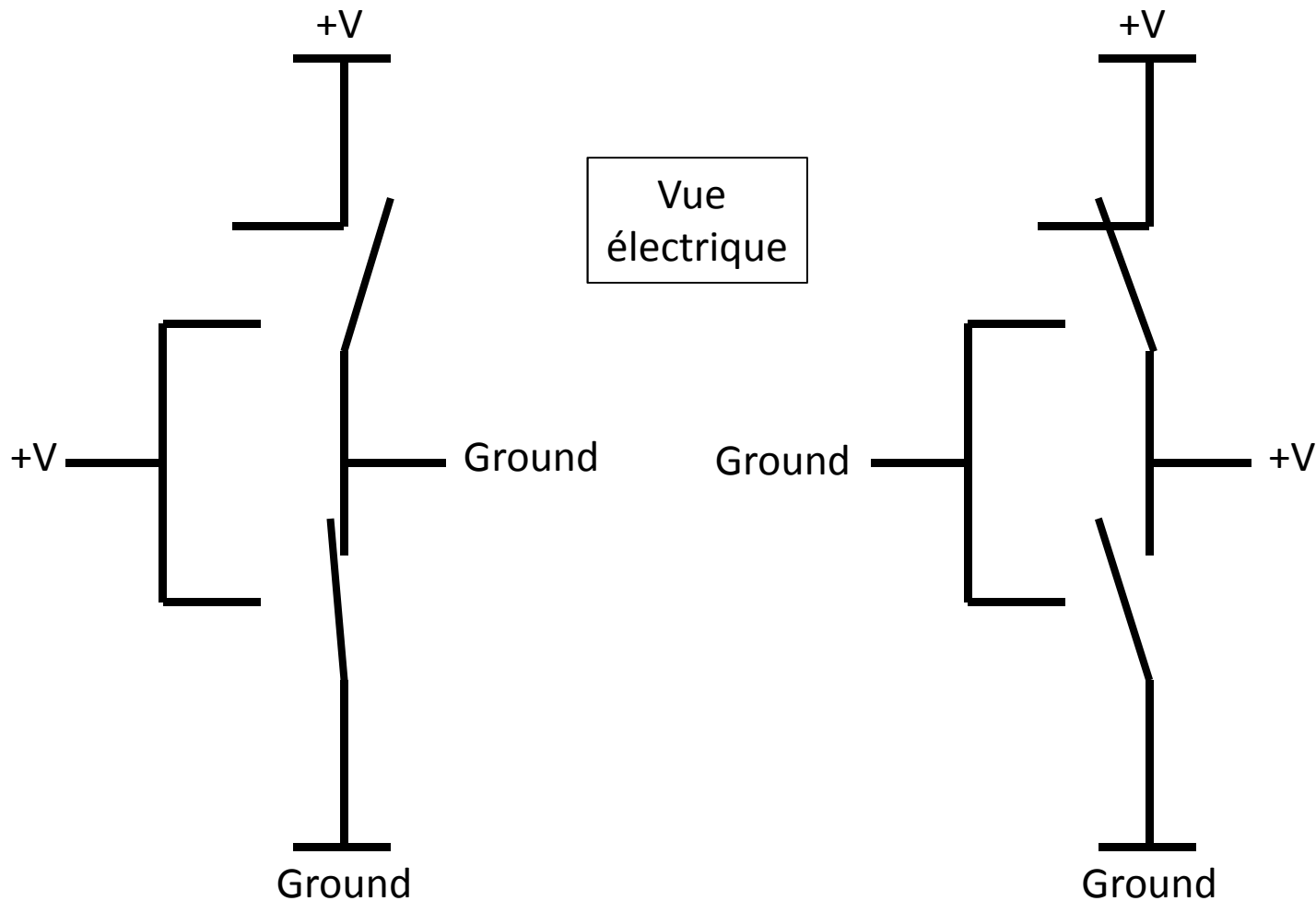


Piloté à la
l'ouverture

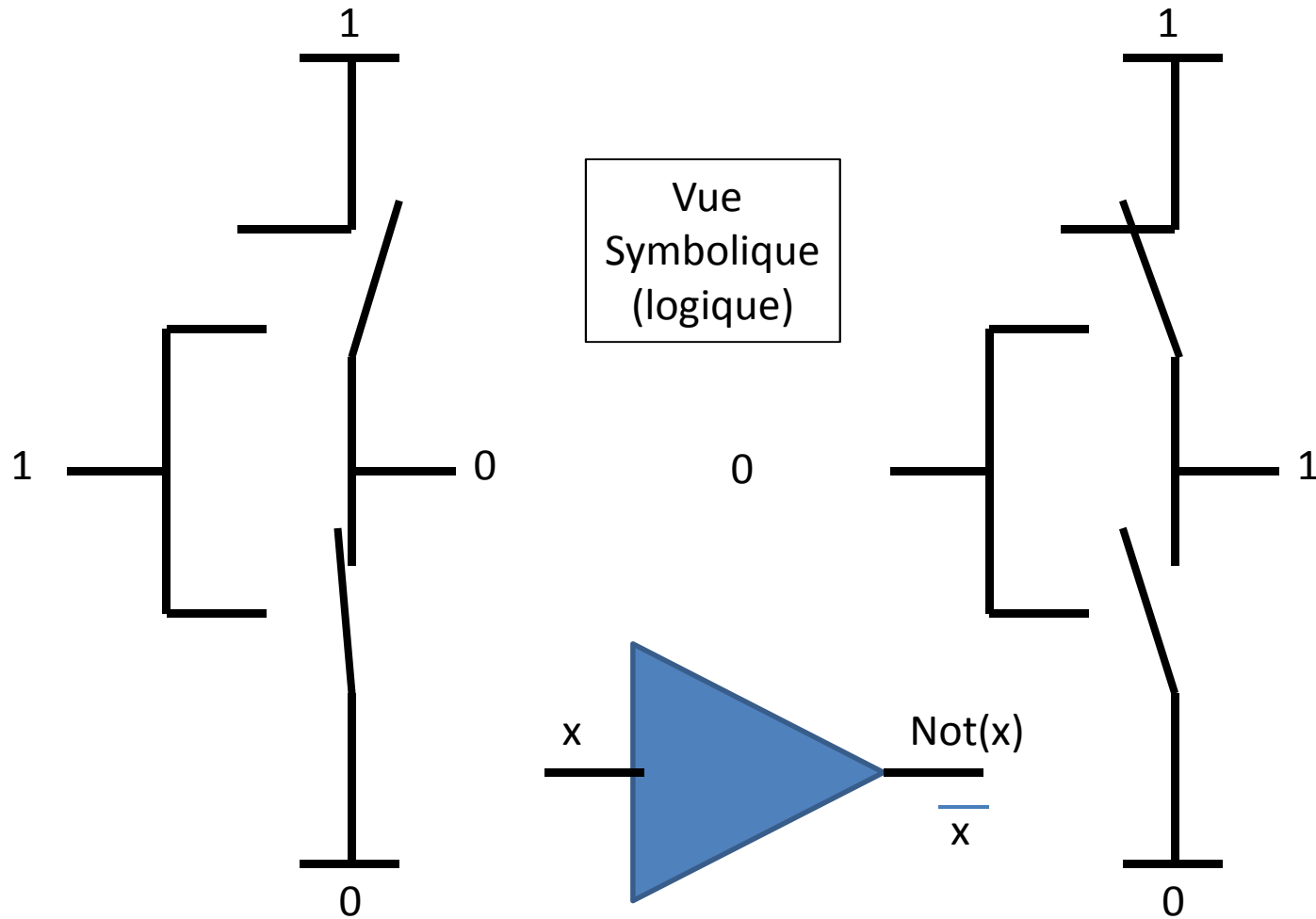
$x = +V \rightarrow$ ouvert

$x = 0 \rightarrow$ fermé

Logique des interrupteurs : inverseur



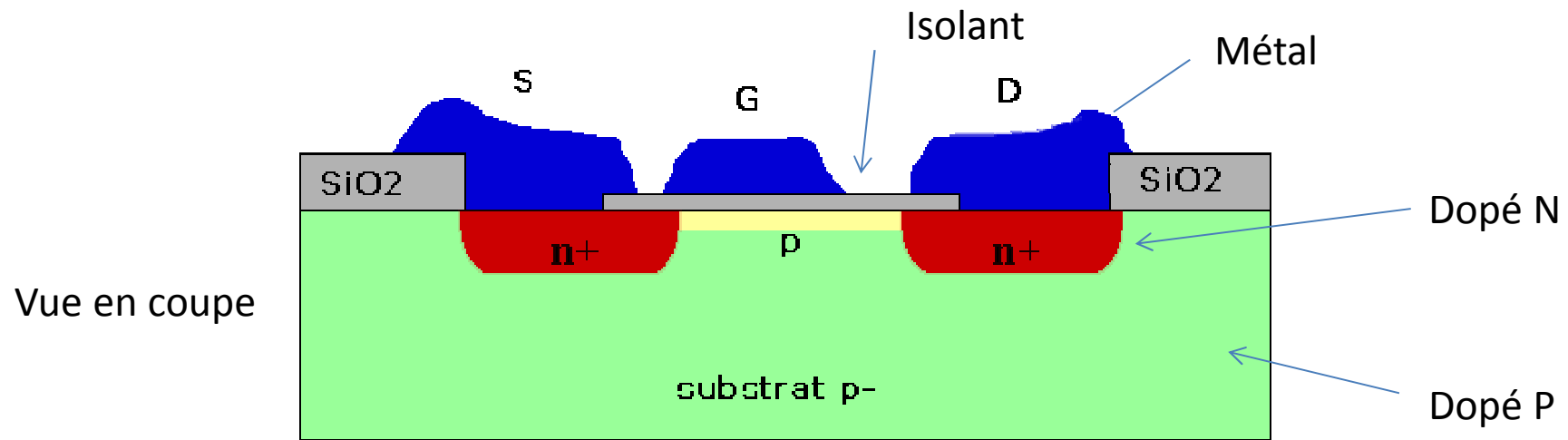
Logique des interrupteurs : inverseur



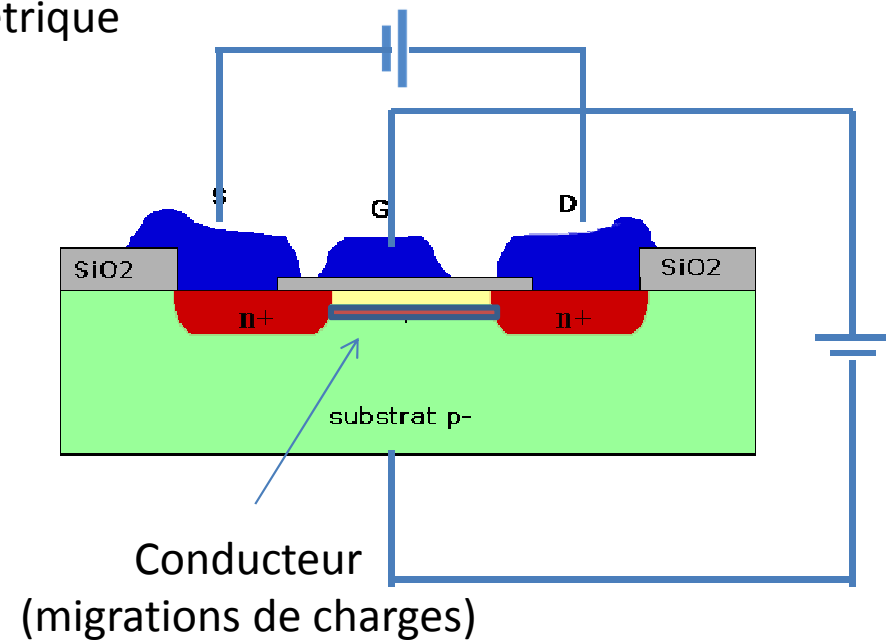
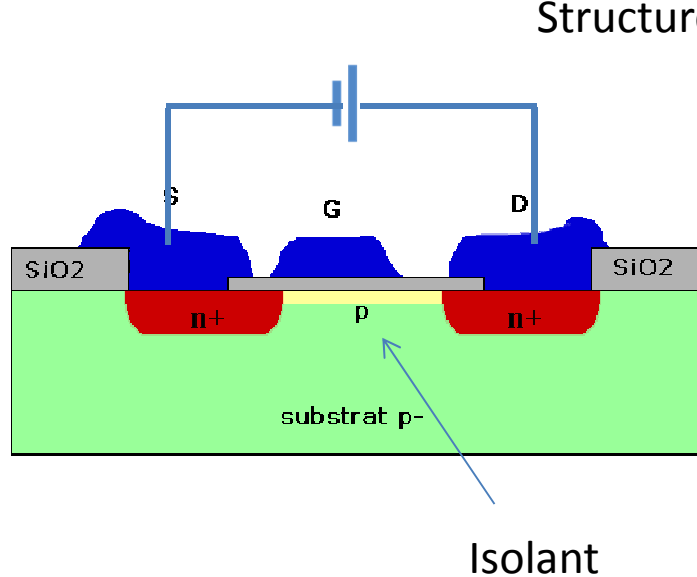
Réalisation des interrupteurs

- Transistor MOS : Metal Oxyde Semi-Conducteur
 - Transistor = Transfer Resistor (résistance de transfert)
 - Ou résistance pilotée par une tension
- Théorie en 1928 par Lilienfield (FET), brevet 1930
- Fabrication 1959 : MOS-FET (lent)
 - Facile à fabriquer
- A l'époque Bipolaire NPN et PNP
 - Compliqués à fabriquer

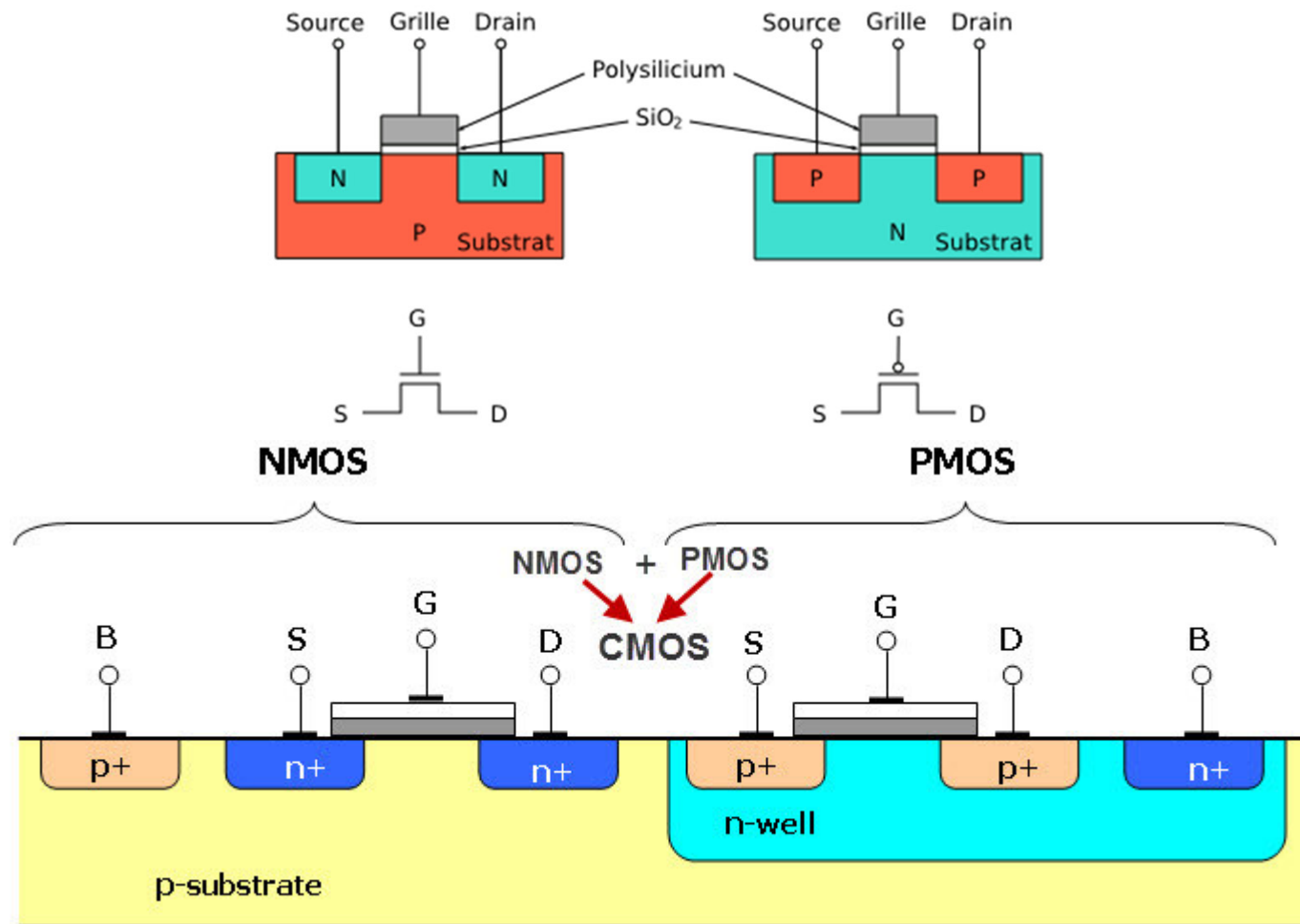
Technologie et fonctionnement



Structure symétrique



NMOS + PMOS = CMOS



CMOS (Complementary MOS : cohabitation PMOS/NMOS, 1962)

Schéma logique

(vue de l'électronicien)

Dessin du masque

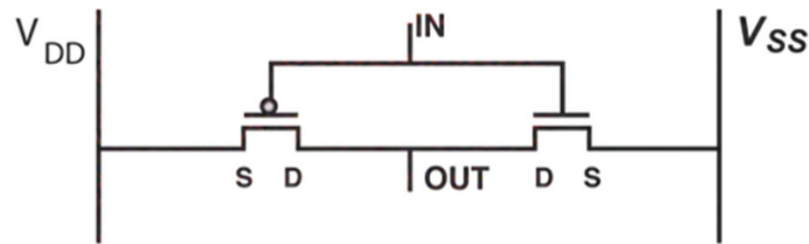
(vue du concepteur)

Techno : vue en coupe

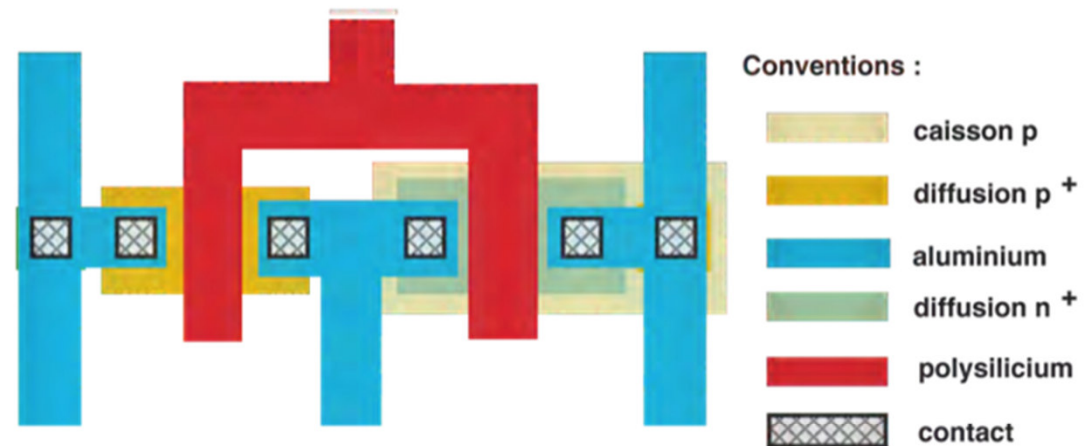
(vue du technologue)

Fabrication d'un inverseur CMOS

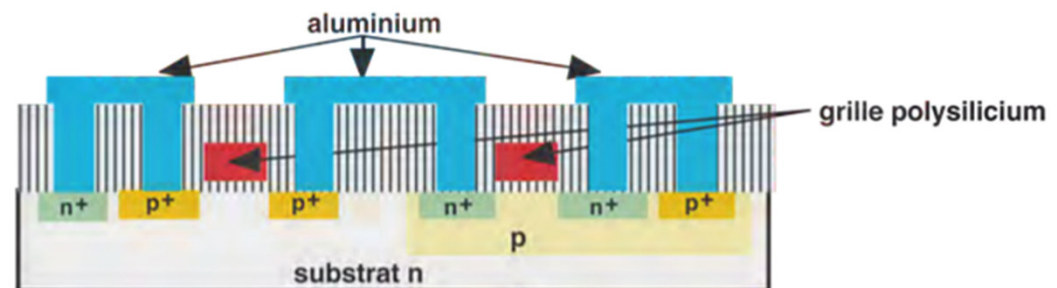
SCHÉMA SYMBOLIQUE



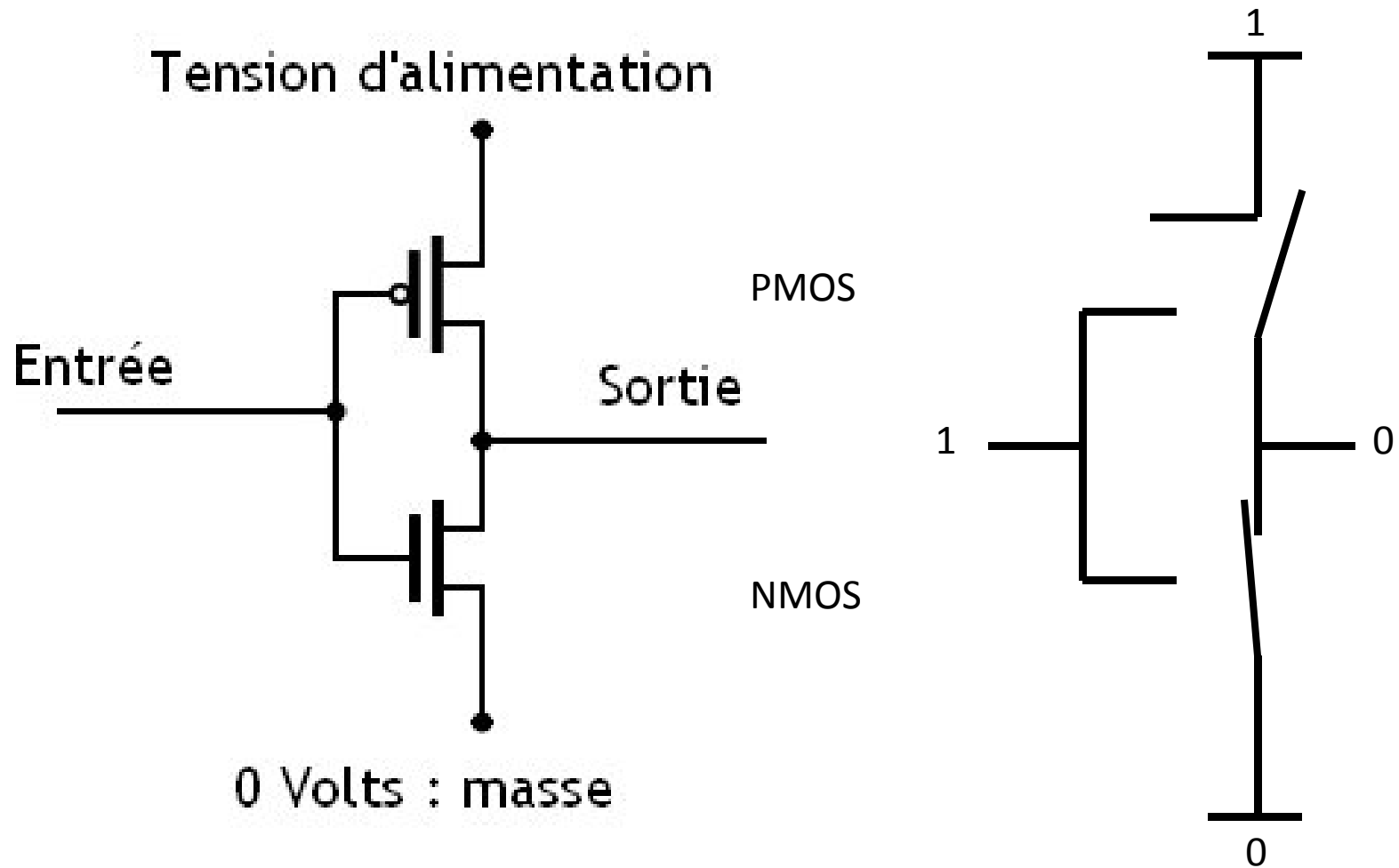
MASQUES UTILISÉS POUR LA FABRICATION

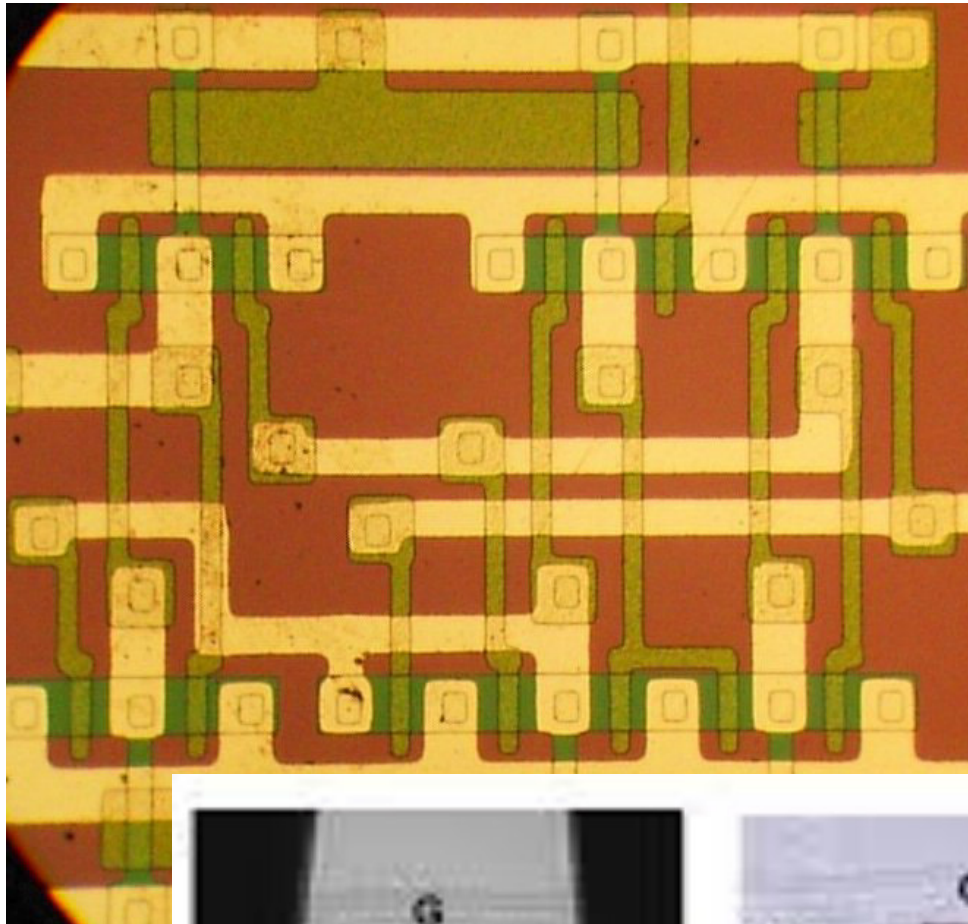


APRÈS FABRICATION (VUE EN COUPE)



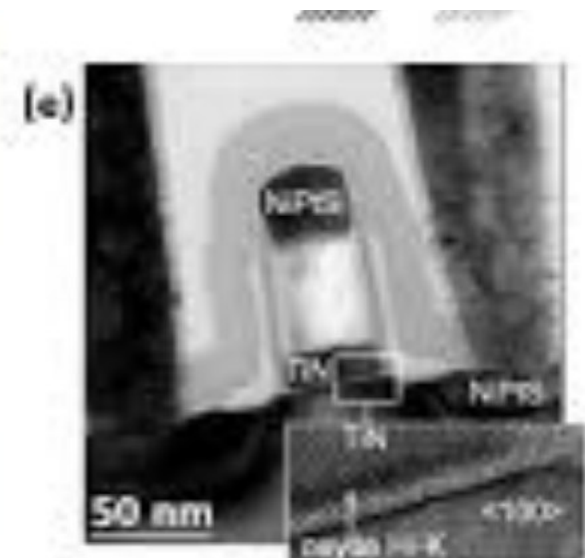
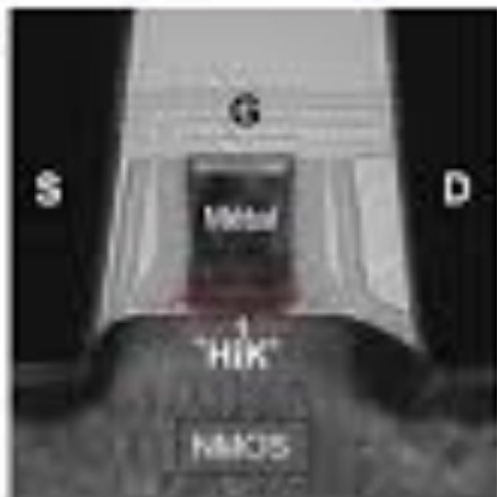
Utilisation en interrupteur



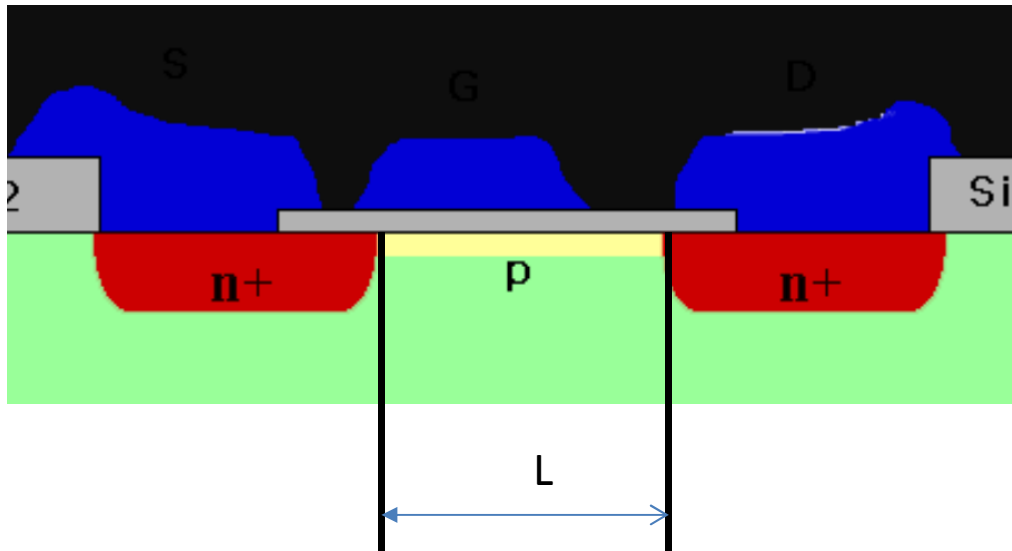


Dans la réalité

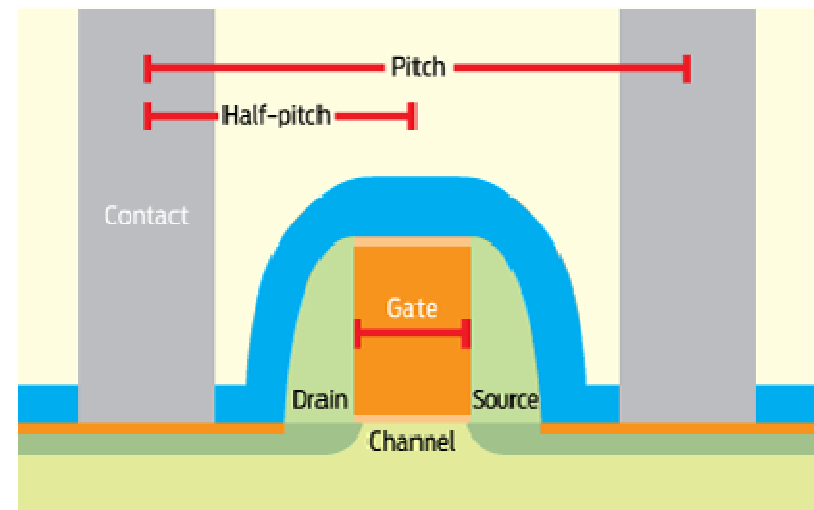
Dans un CI numérique il n'y a que des MOS



Notion de taille de gravure



Ancien : longueur de canal
Puis : plus petit élément gravable



Moderne : demi-pitch
Revendiqué 10/11 nm :
Demi-pitch = 18 nm
Largeur gate (fin/aileron) = 6 nm

Plus de relation entre la « taille revendiquée » et la taille réelle du transistor

Actuellement : 14, 10, 7 nm ... 4 nm

Algèbre binaire : base

- Variable binaire $\in [0,1]$
- Fonction binaire $\in [0,1]$, combine des variables

Algèbre de Boole

- 2 loi de composition interne : ET, OU
- 1 loi unaire : NON
- Les propriétés d'une algèbre

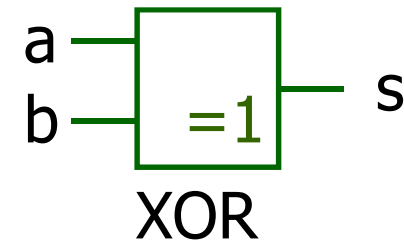
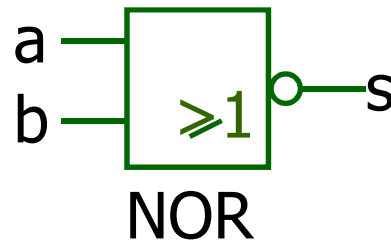
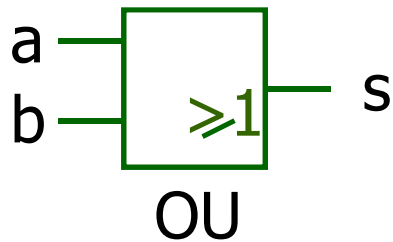
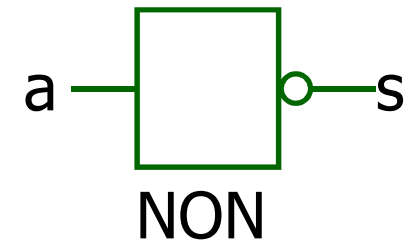
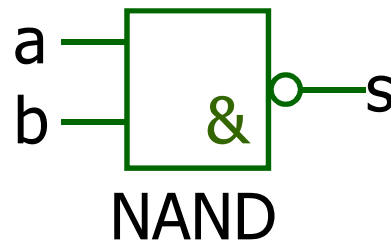
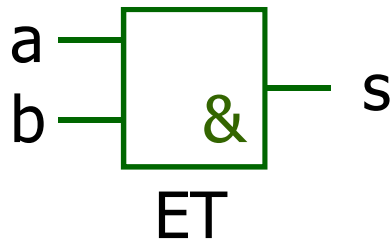
Algèbre binaire : opérateurs

- **NON** (NOT) : si $a = 1$ alors $\text{not}(a)=0$
s'écrit aussi \bar{a} , se lit « a barre »
- **ET** (AND) : $x = a \text{ ET } b = a.b$: $x = 1$ si a ET b valent 1
- **OU** (OR): $c = a \text{ OU } b = a + b$: $x = 1$ si a OU B vaut 1

TOUTE L'ELECTRONIQUE NUMERIQUE PEUT ETRE
CONSTRuite A PARTIR DE CES OPERATEURS
(donc toute l'informatique)

Symboles graphiques

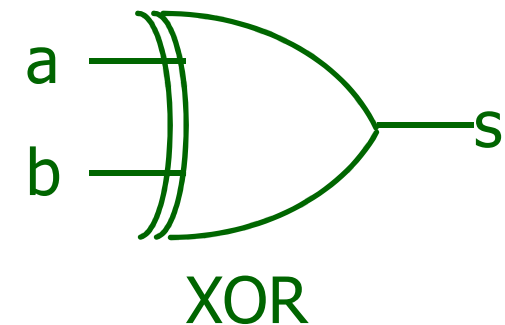
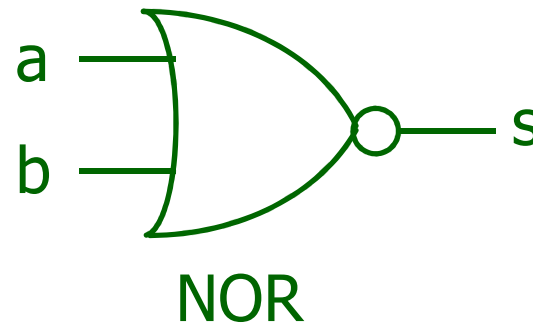
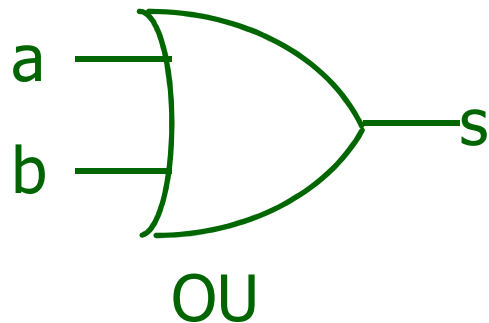
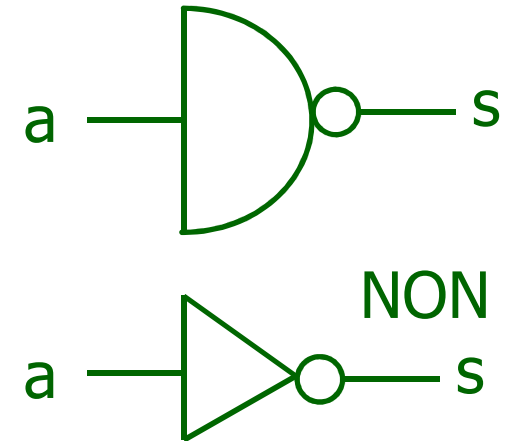
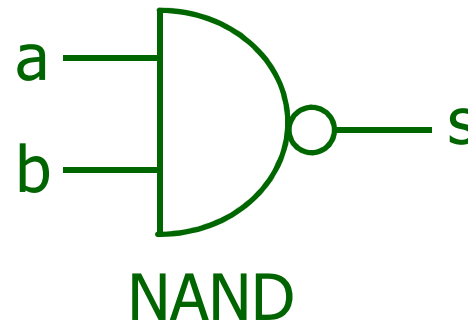
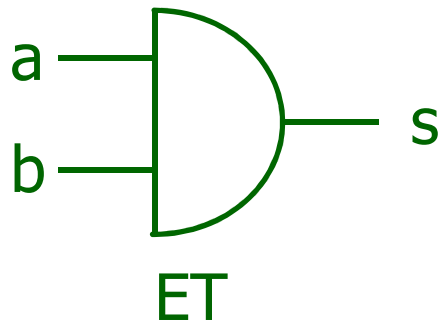
Electronique numérique : théorie mathématique et technologie associée



Norme ANSI/IEEE Std: 91-1984 : Standard Graphic Symbols for Logic Functions

Symboles graphiques

Norme IEEE



Nb : La norme américaine est la norme utilisée dans les « datasheets » pour dessiner les diagrammes (schémas) logiques.

Théorèmes utiles (pour nous)

- Involution

$$\text{not}(\text{not}(x)) = x$$

- Théorème de De Morgan

$$\text{not}(a \text{ **op** } b) = \text{not}(a) \text{ **not(op)** } \text{not}(b)$$

$$\text{not}(a + b) = \text{not}(a) . \text{not}(b)$$

$$\text{not}(a . b) = \text{not}(a) + \text{not}(b)$$

Autres opérateurs utiles

- **NAND** (NO-AND) : $x = a \uparrow b = \text{not}(a.b)$
- **NOR** (NO-OR) : $x = a \downarrow b = \text{not}(a+b)$
- Intérêt : opérateurs complets

**TOUTE L'ELECTRONIQUE NUMERIQUE PEUT ETRE
CONSTRuite A PARTIR D'UN SEUL OPERATEUR
(donc toute l'informatique)**

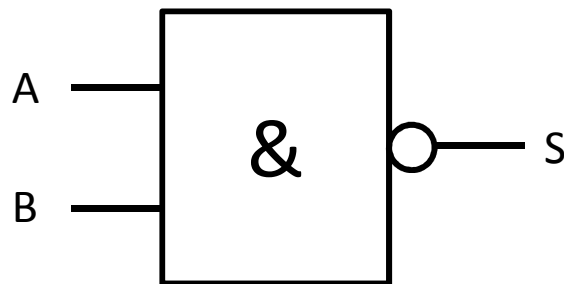
[De plus : propriétés technologiques intéressantes]

Preuve de complétude

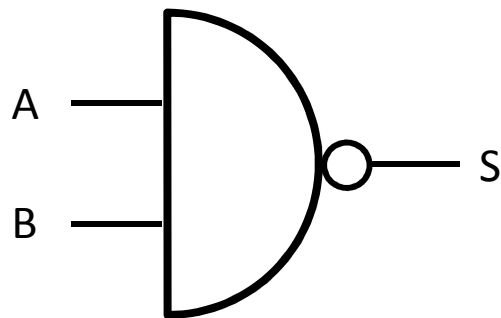
- (ET, OU, NON) = groupe complet
 - $a \uparrow a = \text{not}(a.a) = \text{not}(a)$
 - $(a \uparrow a) \uparrow (b \uparrow b) = \text{not}(\text{not}(a).\text{not}(b)) = a + b$
 - $(a \uparrow b) \uparrow (a \uparrow b) = \text{not}(\text{not}(a.b)) = a.b$
- NAND permet d'écrire le groupe complet, il est donc complet

Opérateurs : symbole et schéma

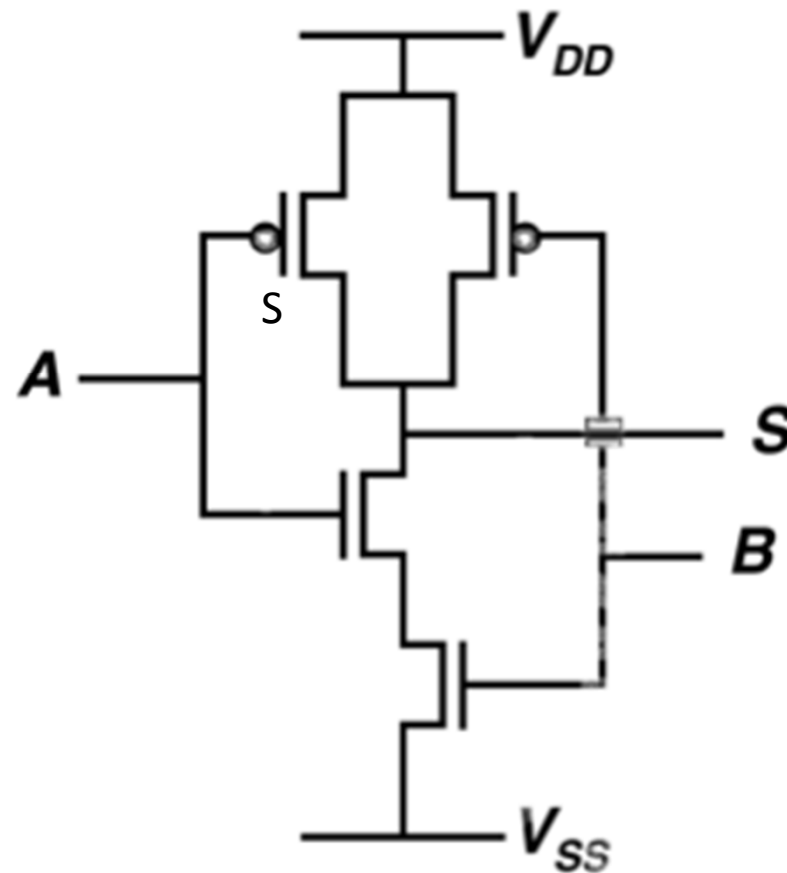
Porte NAND CMOS



Norme ANSI



Norme IEEE



Autres opérateurs importants

- OU Exclusif (XOR) : $x = a \oplus b = a.\text{not}(b) + \text{not}(a).b$
x vaut 1 si a ou b vaut 1 (mais pas les deux)
ou x vaut 1 si a et b sont différents
opérateur de disjonction ou de différence
- XNOR (not XOR) : $x = a \overline{\oplus} b = \text{not}(a).\text{not}(b) + a.b$
x vaut 1 si a et b sont identiques
opérateur de conjonction ou d'égalité
- XOR à plusieurs entrées : vaut 1 si nombre variable à 1 est impair
- XNOR à plusieurs entrées : vaut 1 si nombre variables à 1 est pair

Logique combinatoire

Combinatoire = relation univoque E/S
pas de boucle (cycle)
pas de mémoire interne

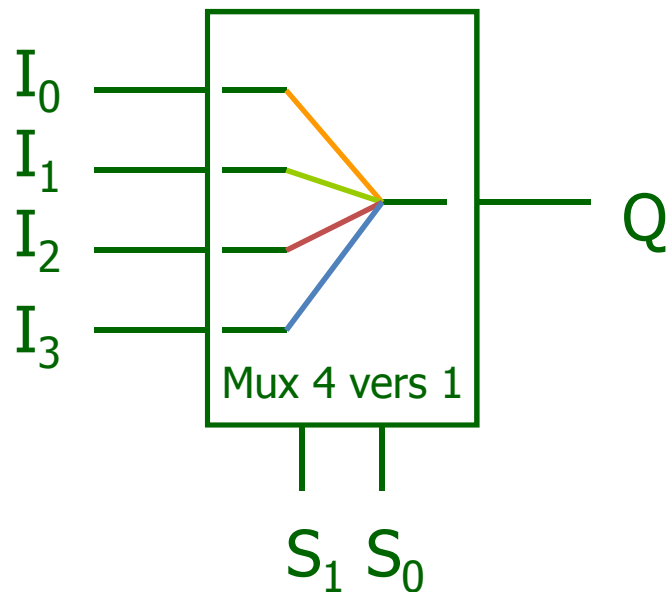
Fonctions combinatoires

- Multiplexeur
- Démultiplexeur
- Comparateur
- (Additionneur)
- Unité arithmétique et logique (ALU)

Construites à partir des opérateurs de base

Multiplexeur : principe

Sélection d'une voie parmi 2^N par N bits de commande



Si $(S_1 S_0)_2 = 0$ alors $Q = I_0$

$$Q = \overline{S_0} \cdot \overline{S_1} \cdot I_0$$

Si $(S_1 S_0)_2 = 1$ alors $Q = I_1$

$$Q = S_0 \cdot \overline{S_1} \cdot I_1$$

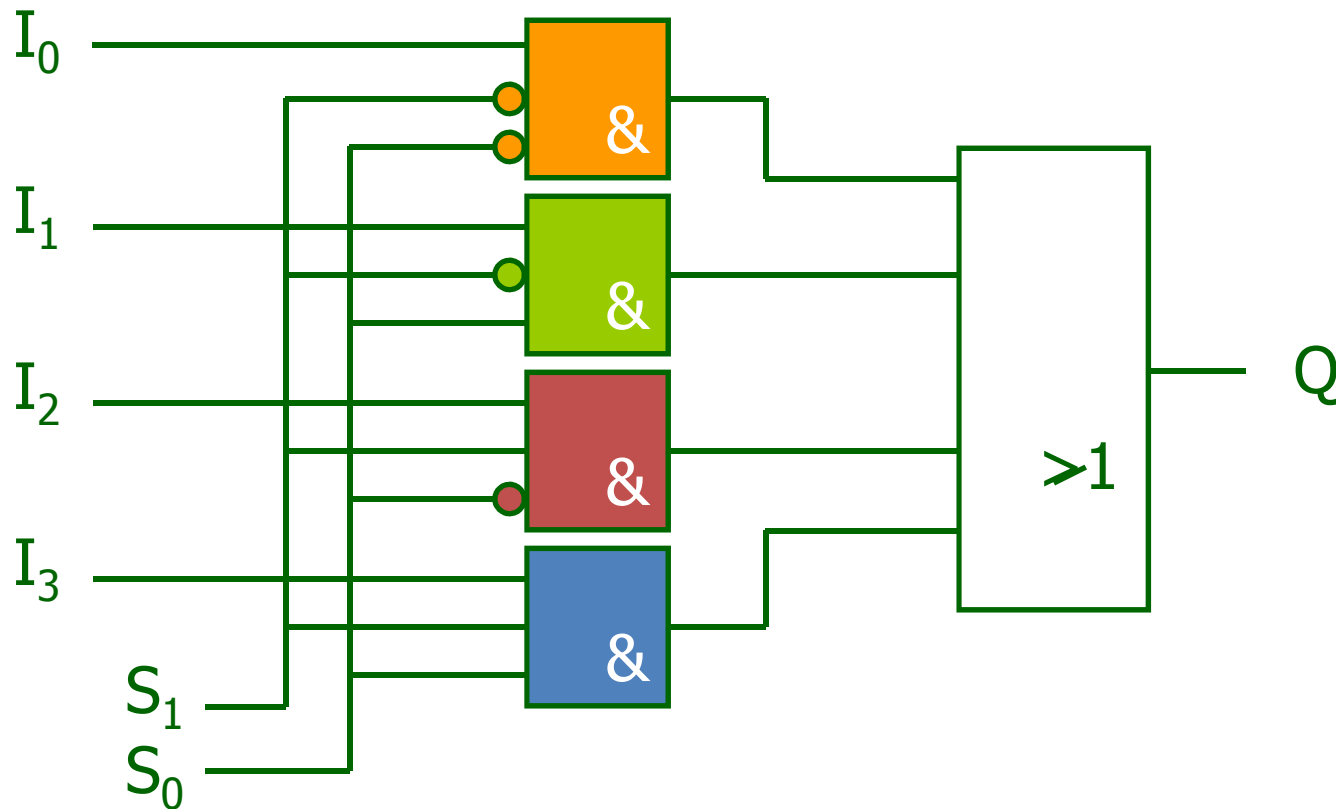
...

...

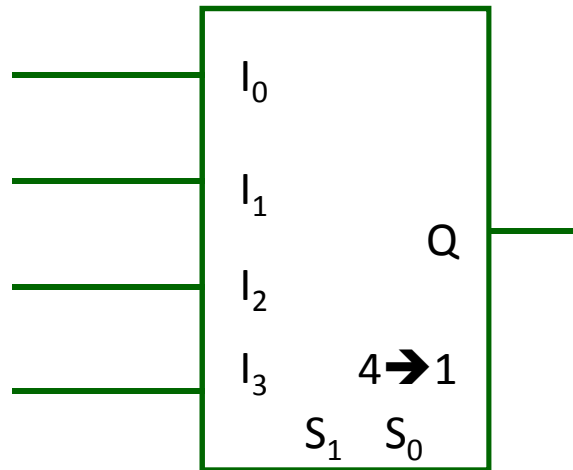
$$Q = \overline{S_1} \cdot \overline{S_0} \cdot I_0 + \overline{S_1} \cdot S_0 \cdot I_1 + S_1 \cdot \overline{S_0} \cdot I_2 + S_1 \cdot S_0 \cdot I_3$$

Multiplexeur : schéma

$$Q = \overline{S_1} \cdot \overline{S_0} \cdot I_0 + \overline{S_1} \cdot S_0 \cdot I_1 + S_1 \cdot \overline{S_0} \cdot I_2 + S_1 \cdot S_0 \cdot I_3$$



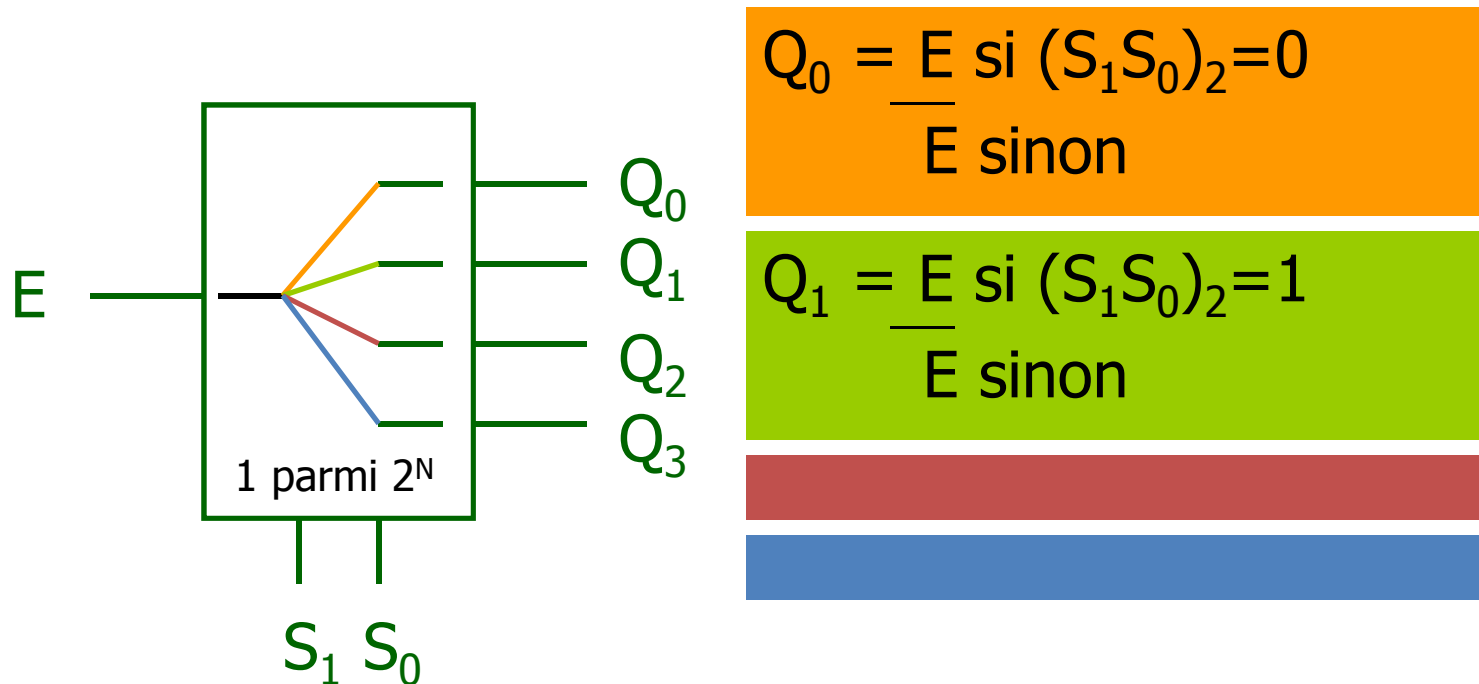
Multiplexeur : symbole



On n'utilise que l'abstraction du symbole sans redescendre au niveau portes
Les logiciels de conception feront la « synthèse logique »

Démultiplexeur 1 vers 2^N

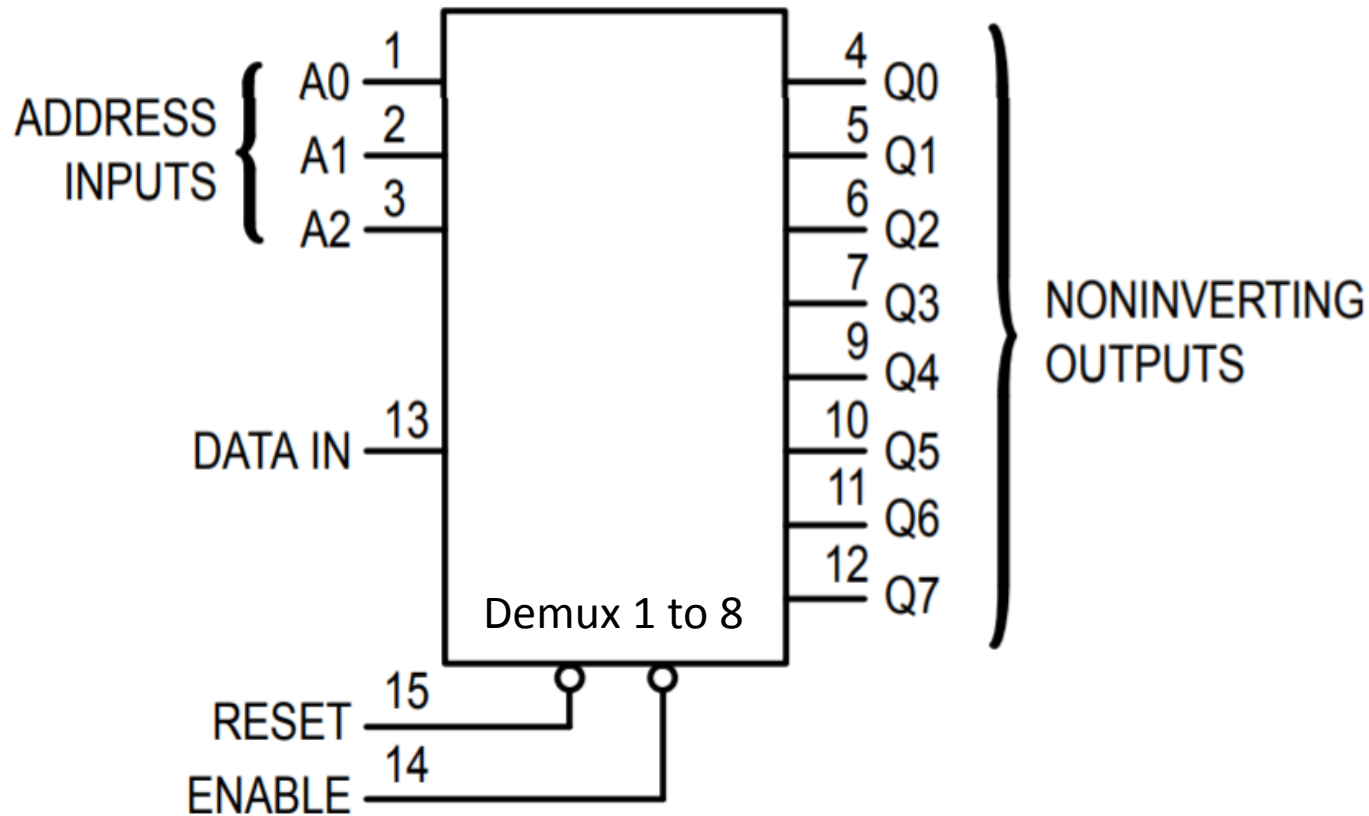
- Fonction duale du multiplexeur



Remarque : E peut ne pas être «disponible»

Sortie sélectionnée = 1 les autres 0
ou Sortie sélectionnée = 0 les autres 1

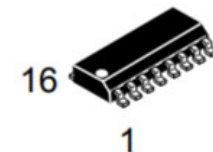
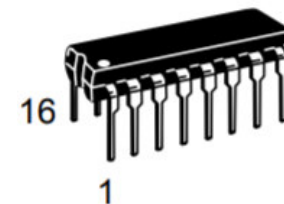
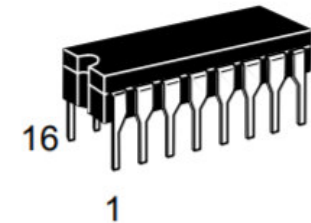
Demux : symbole



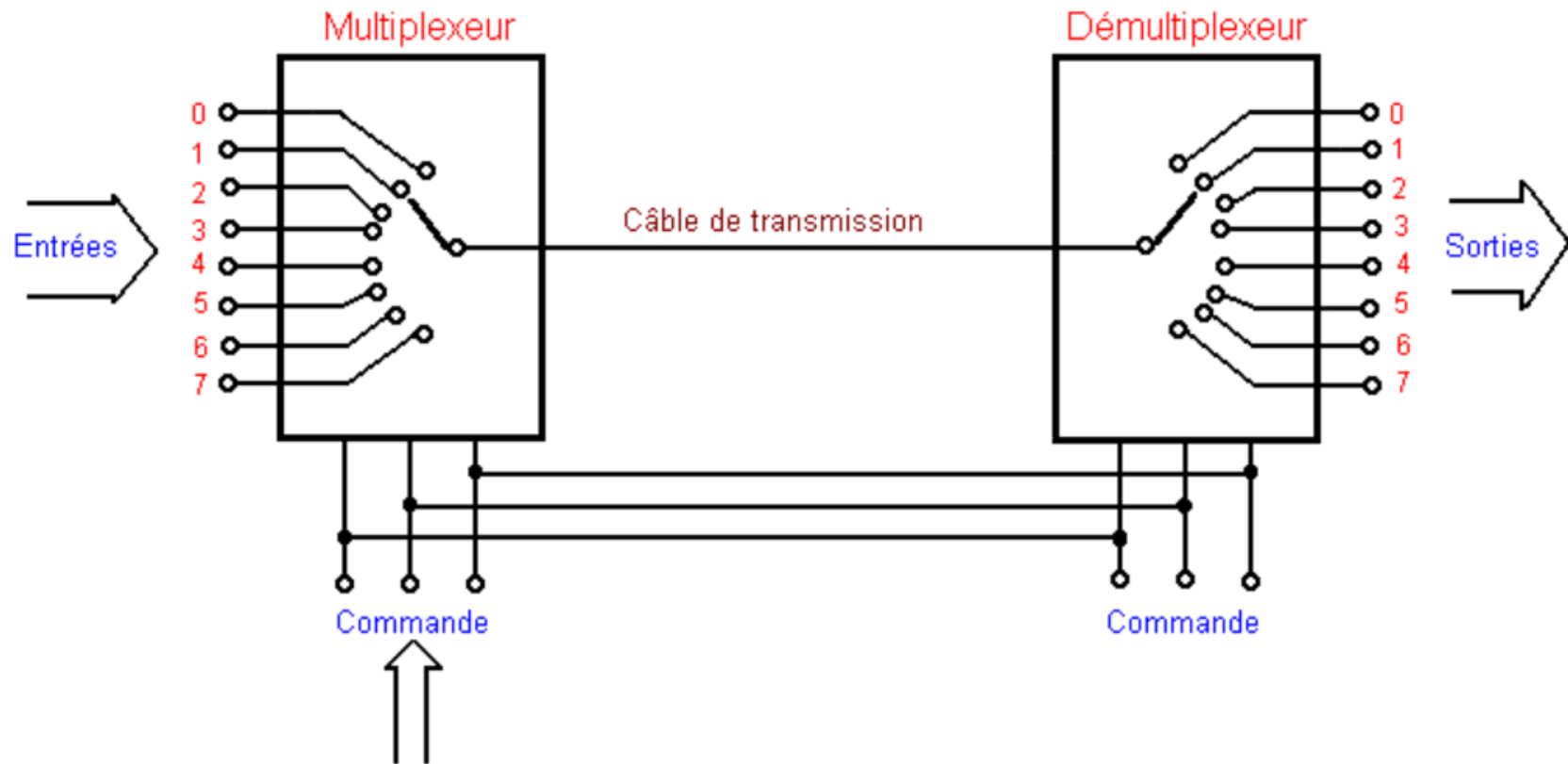
Motorola MC74HC259

PIN ASSIGNMENT

A0	1	16	VCC
A1	2	15	RESET
A2	3	14	ENABLE
Q0	4	13	DATA IN
Q1	5	12	Q7
Q2	6	11	Q6
Q3	7	10	Q5
GND	8	9	Q4



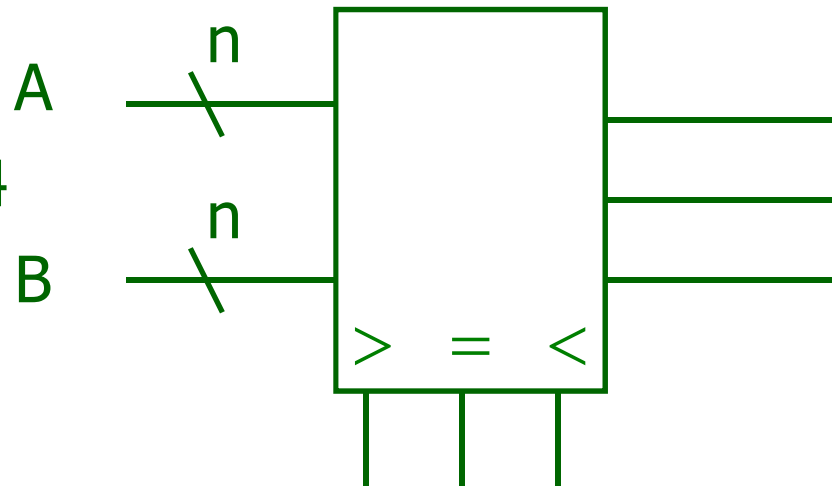
Exemple d'utilisation



Comparateur

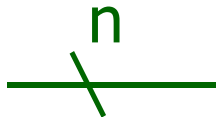
classiqu. $n=4$

A et B interprétés
comme nombres



Entrées de cascadage

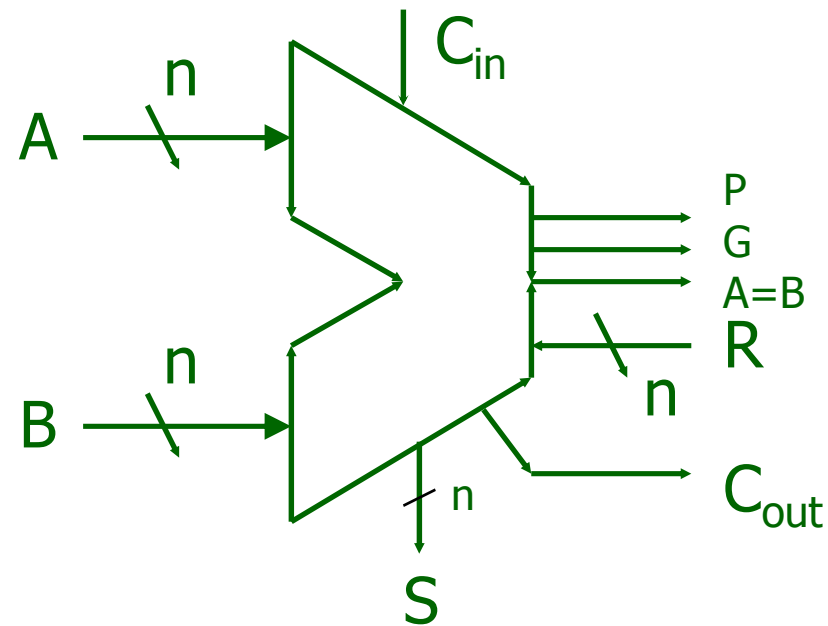
$S = 1$ si $A > B$
 $E = 1$ si $A = B$
 $I = 1$ si $A < B$



Notion de bus : regrouper n fils avec un symbole

ALU (UAL) : Unité Arithmétique et Logique

Classique. $n=4$



P/G pour la propagation
Rapide de retenue en cas
de cascade (8,12,16 ...bits)

n bits : choix de la
fonction (max 2^n cas)

Exemple :

$$R = A + \overline{B}$$

$$R = A + B$$

$$R = A + B + 1$$

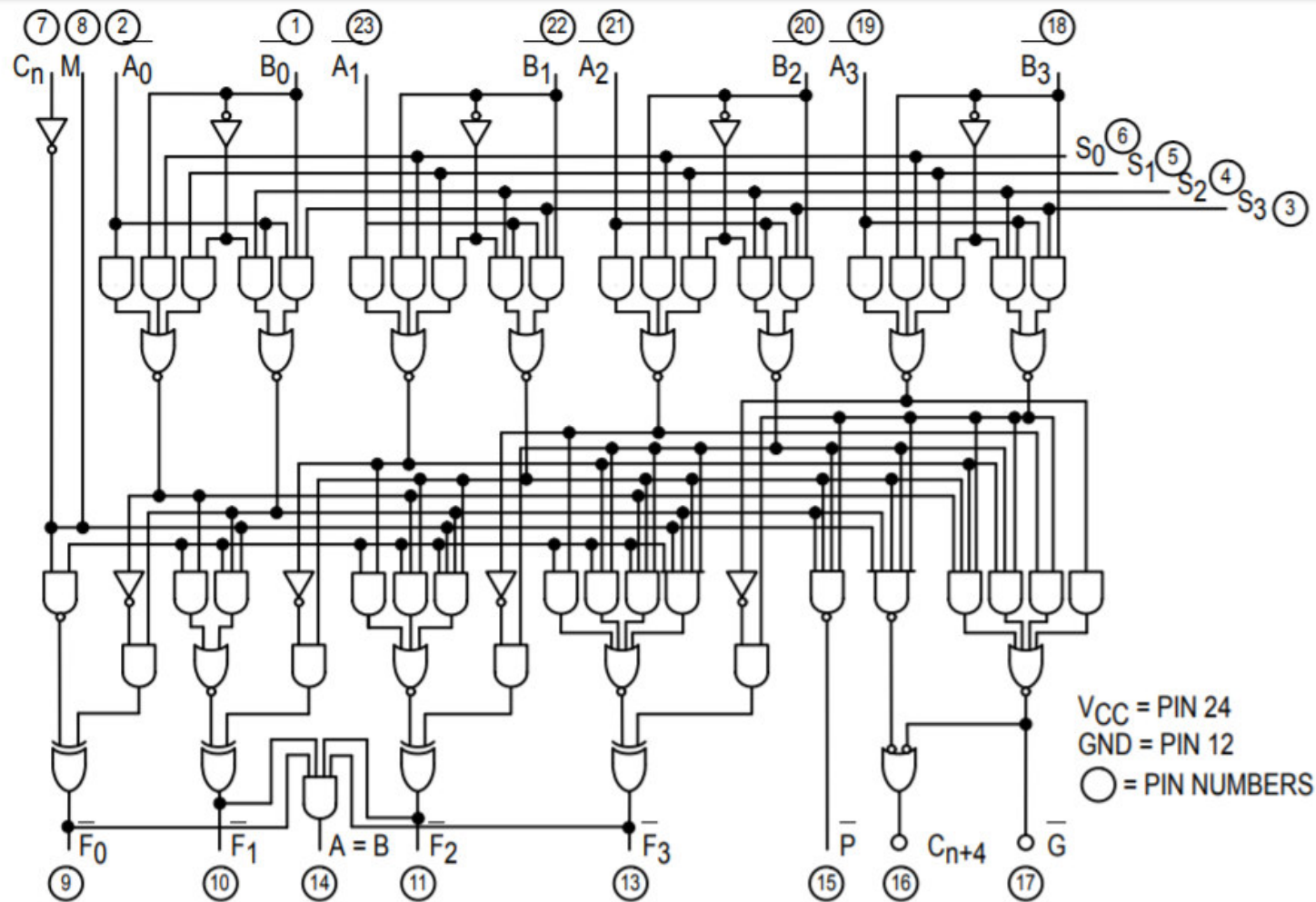
...

$$R = A \text{ ou } B$$

$$R = A \text{ nand } B$$

...

Example ALU : 74181 (1)



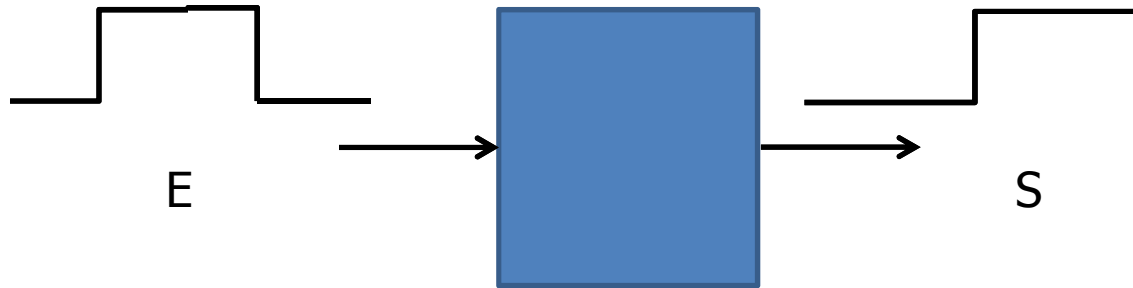
FUNCTION TABLE

MODE SELECT INPUTS				ACTIVE LOW INPUTS & OUTPUTS		ACTIVE HIGH INPUTS & OUTPUTS	
S ₃	S ₂	S ₁	S ₀	LOGIC (M = H)	ARITHMETIC** (M = L) (C _n = L)	LOGIC (M = H)	ARITHMETIC** (M = L) (C _n = H)
L	L	L	L	\overline{A}	A minus 1	\overline{A}	A
L	L	L	H	\overline{AB}	\overline{AB} minus 1	$\overline{A + B}$	A + \overline{B}
L	L	H	L	A + B	AB minus 1	AB	A + B
L	L	H	H	Logical 1	minus 1	Logical 0	minus 1
L	H	L	L	$\overline{A + B}$	A plus (A + \overline{B})	\overline{AB}	A plus AB
L	H	L	H	\overline{B}	AB plus (A + B)	B	(A + B) plus AB
L	H	H	L	$A \oplus \overline{B}$	A minus B minus 1	$A \oplus B$	A minus B minus 1
L	H	H	H	$\overline{A + B}$	A + B	\overline{AB}	AB minus 1
H	L	L	L	AB	A plus (A + B)	$\overline{A + B}$	A plus AB
H	L	L	H	$A \oplus B$	A plus B	$A \oplus B$	A plus B
H	L	H	L	B	AB plus (A + B)	B	(A + B) plus AB
H	L	H	H	A + B	A + B	AB	AB minus 1
H	H	L	L	Logical 0	A plus A*	Logical 1	A plus A*
H	H	L	H	AB	\overline{AB} plus A	A + B	(A + \overline{B}) plus A
H	H	H	L	AB	AB plus A	A + B	(A + B) Plus A
H	H	H	H	A	A	A	A minus 1

Logique séquentielle

Séquentielle = relation E/S non univoque
boucle (cycle)
mémoire interne

Non univoque ?



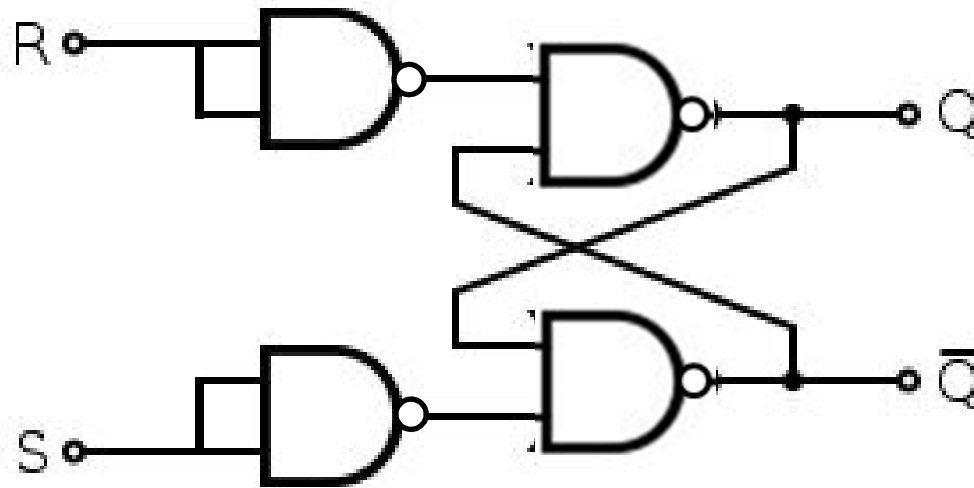
Même configuration d'entrées

➔ (possible) sorties différentes

Fonctions séquentielles

- Mémoires RS, RST, D-Latch
- Bascule RS-MS, D (D-Edge), JK, T
- Registre, registre à décalage
- Compteur
- Circuit mémoire
 - Technologies associées [voir cours spécifique]
- Séquenceur

Mémoire RS



Circuit à 1 boucle

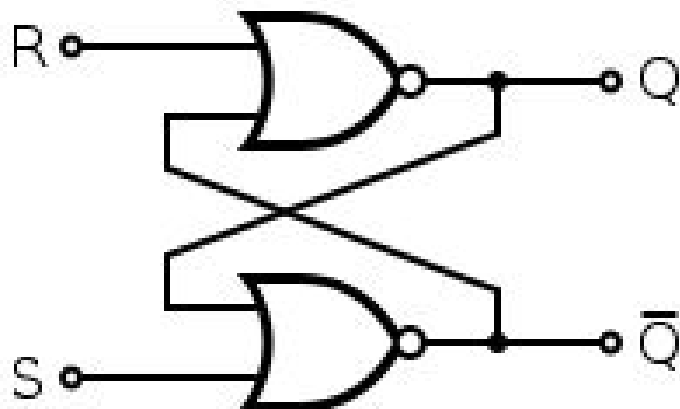
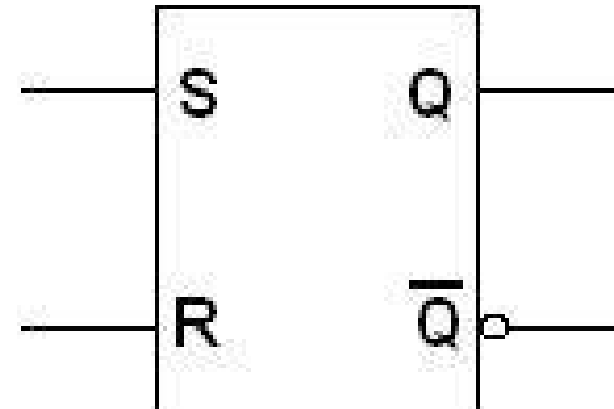


Table de vérité :

S	R	Q	\bar{Q}	remarque
0	0	q	\bar{q}	mémorisation
0	1	0	1	mise à 0
1	0	1	0	mise à 1
1	1	0	0	cas particulier

Pas
univoque



Mémoire RS



Table de vérité :

Pas
univoque

Cas particulier $R=S=1$:

- Différencie les deux architectures
deux sorties à 0 ou deux sorties à 1
- Les deux sorties ne sont pas complémentaires

Contrairement à une littérature bas de gamme :

- Cet état n'est pas interdit
- Cet état n'est pas dangereux

On évite de l'utiliser pour se simplifier la vie

S

R

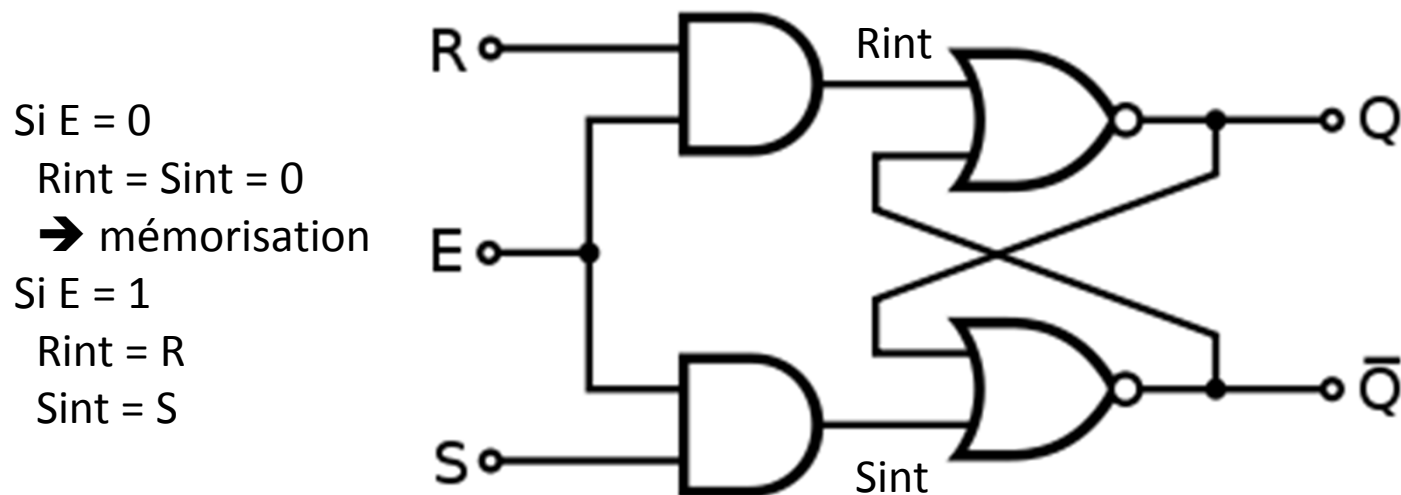
S

tion

ulier

Mémoire RST

On veut isoler le système des entrées R/S sauf pendant un temps court sous notre contrôle



L'entrée E est nommée Enable (« Horloge » abus)
Le système fonctionne sur niveau de E

Notions générales

- Le signal nommé Horloge (Clock) permet de synchroniser toutes les parties du système en servant de métronome (sur un ou deux fronts).
- C'est un signal carré, classiquement périodique, pas forcément de rapport cyclique 50%
 - A haute fréquence (GHz) c'est plus à une sinusoïde.
- Dans un système bien conçu
 - Tous les traitements sont synchrones (front horloge)
 - Aucun calcul n'est fait sur l'horloge

Mémoire D-Latch

D = Data

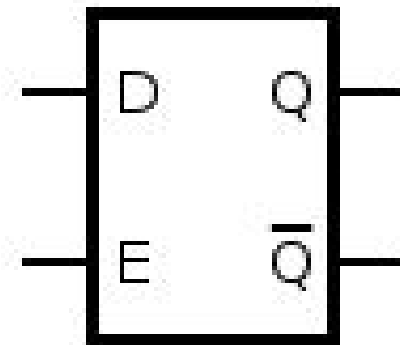
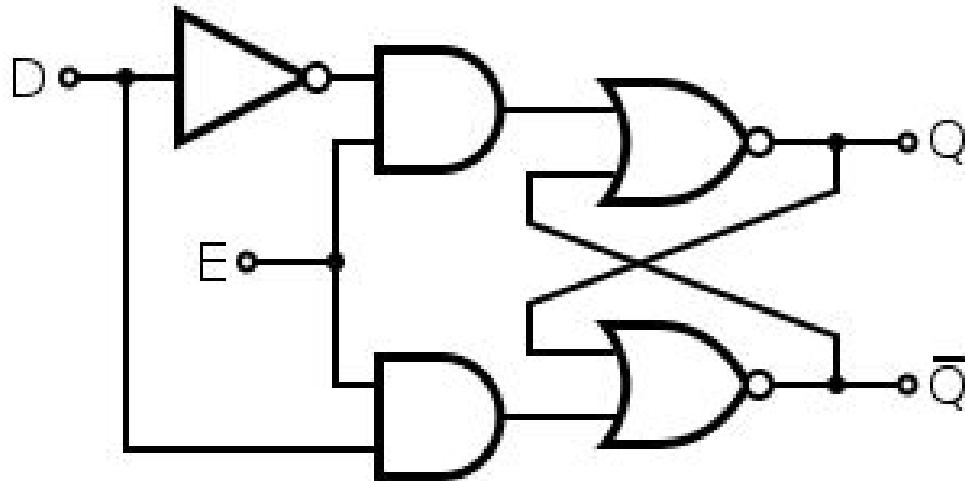


Table de vérité :

E	D	Q	\bar{Q}	remarque
0	x	q	\bar{q}	mémorisation
1	0	0	1	mise à 0
1	1	1	0	mise à 1

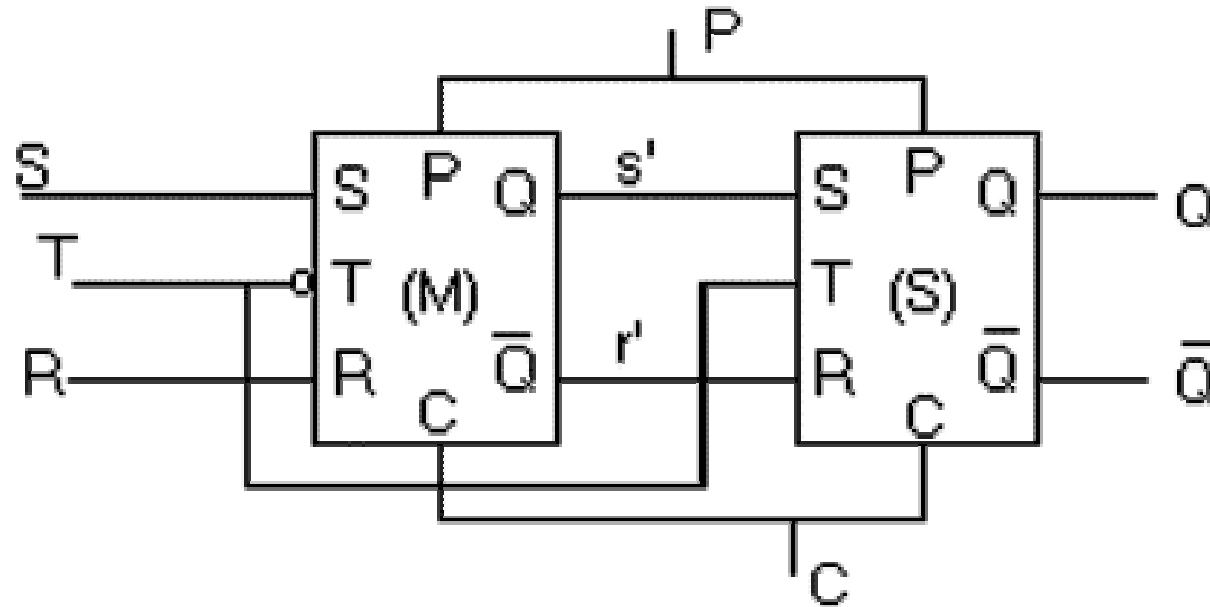
On a toujours $R = \text{Not}(S)$

Vocabulaire et précisions

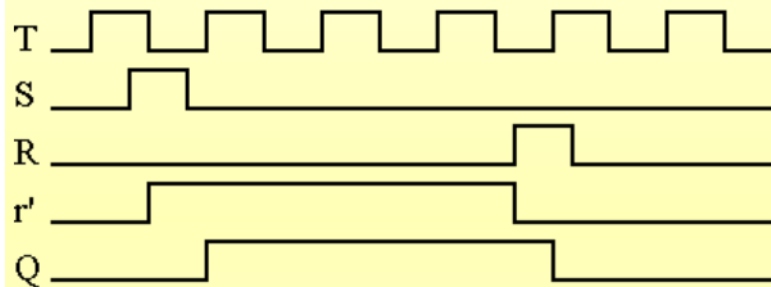
- Latch = Verrou = mémoire
asynchrone ou sur niveau d'horloge
ne peut pas compter
- Flip-flop = bascule
sur front d'horloge
peut compter

On ne mélange pas les deux définitions

Bascule RS-MS

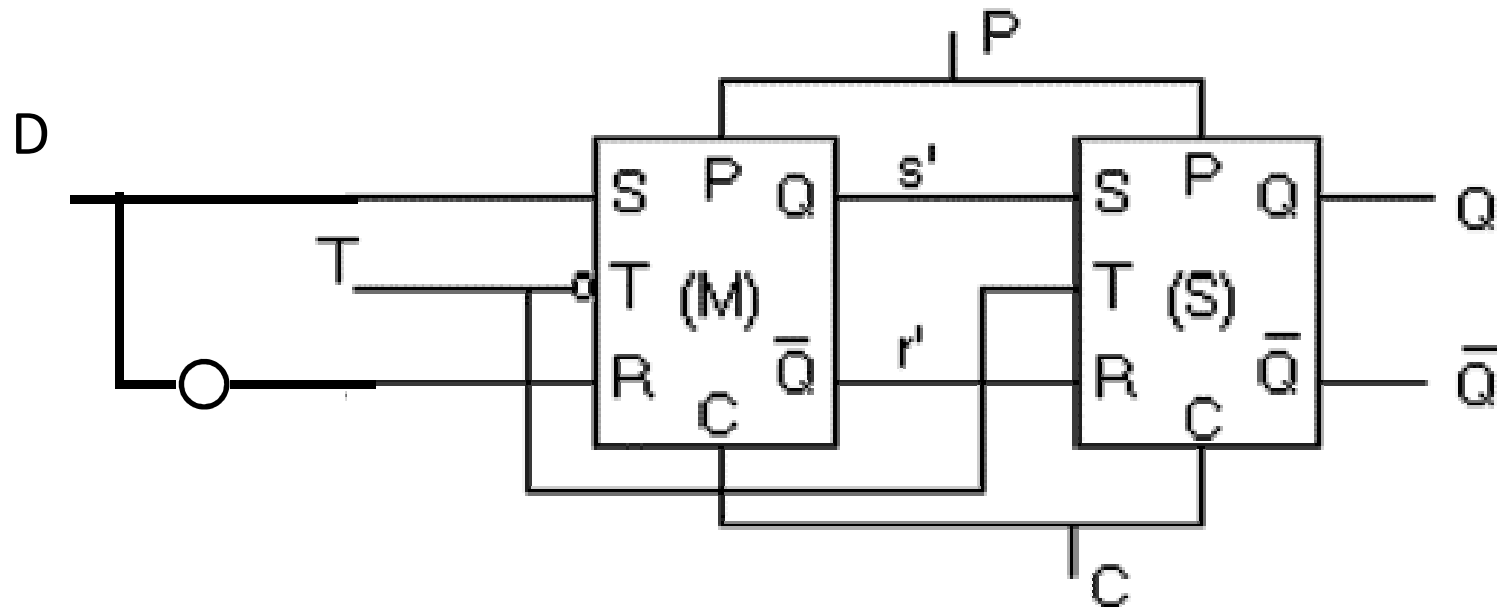


	maître (Master)	esclave (Slave)
Si T=0	information S/R transmise en s'r'	non transmis en Q (ancien Q)
Si T=1	R S en attente (ancien r's')	ancien r's' transmis en Q



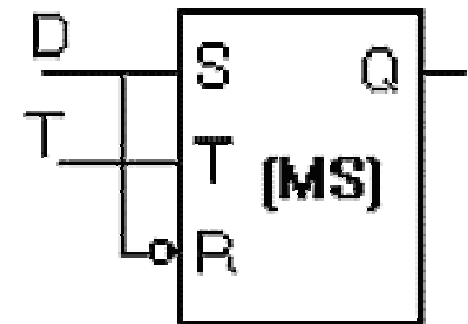
On remarque donc que l'information est transmise au prochain front montant de l'horloge T.

Bascule D-MS



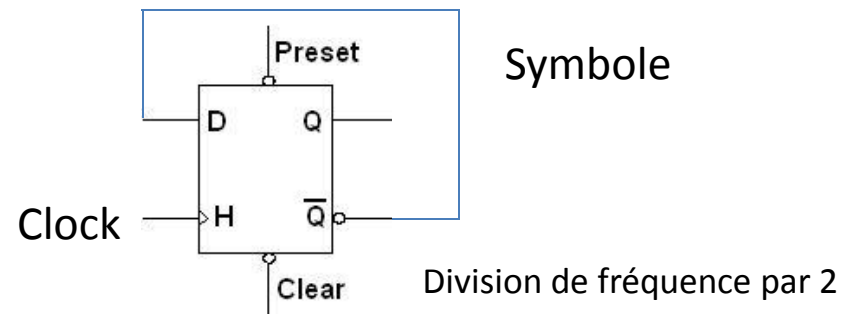
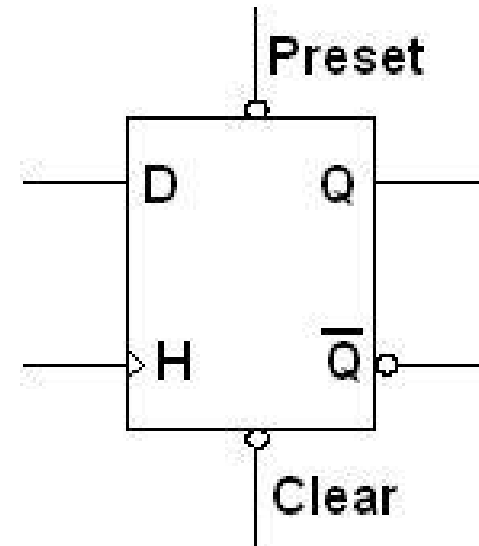
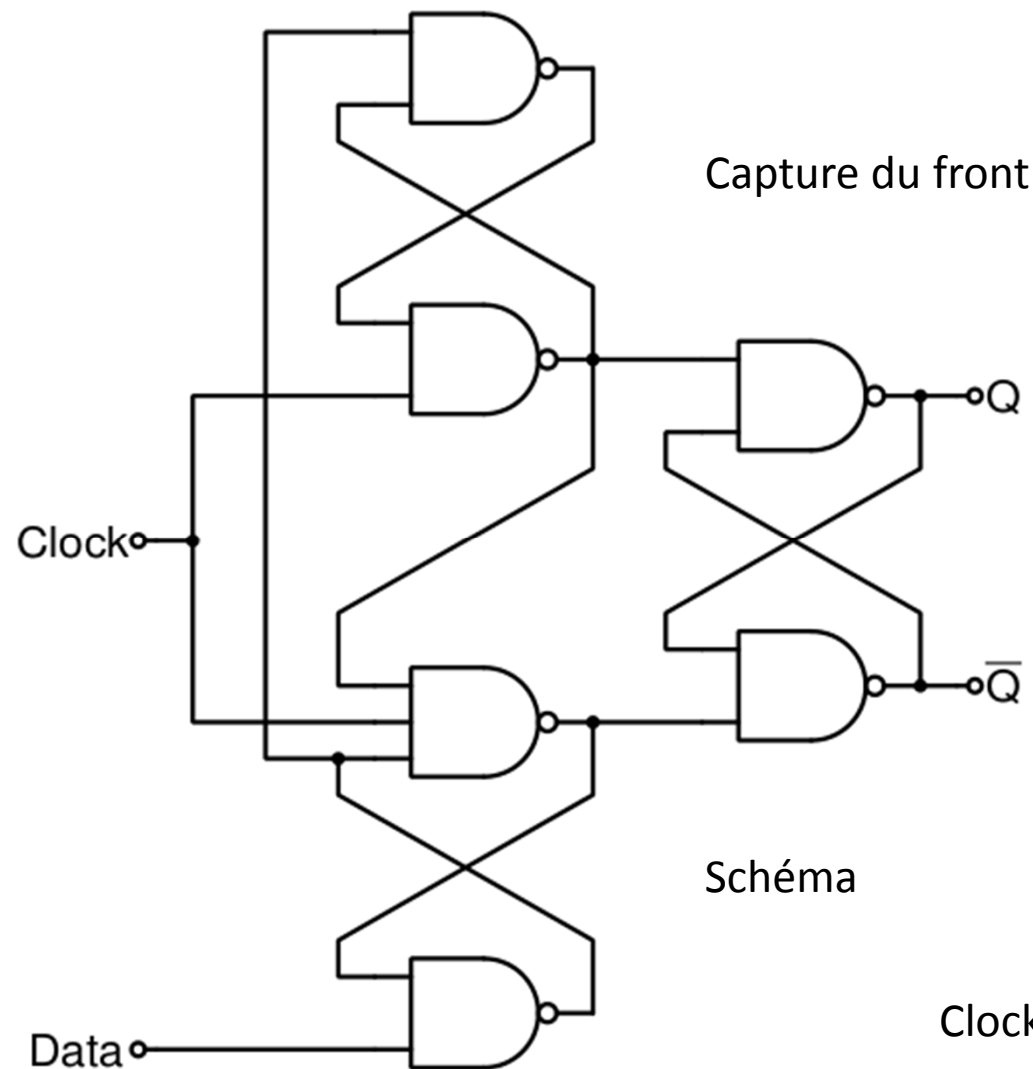
Fonctionnement sur front montant de T

Au front montant : $D \rightarrow Q$



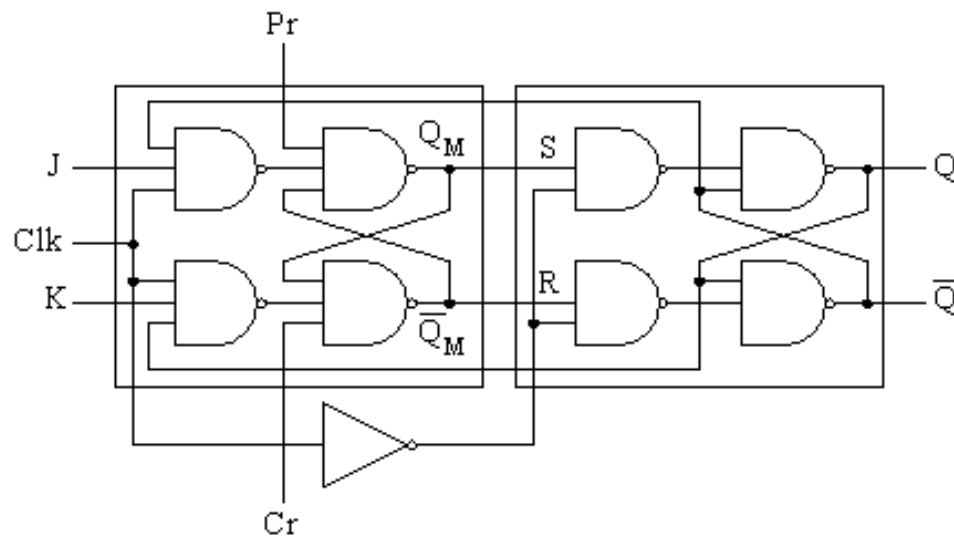
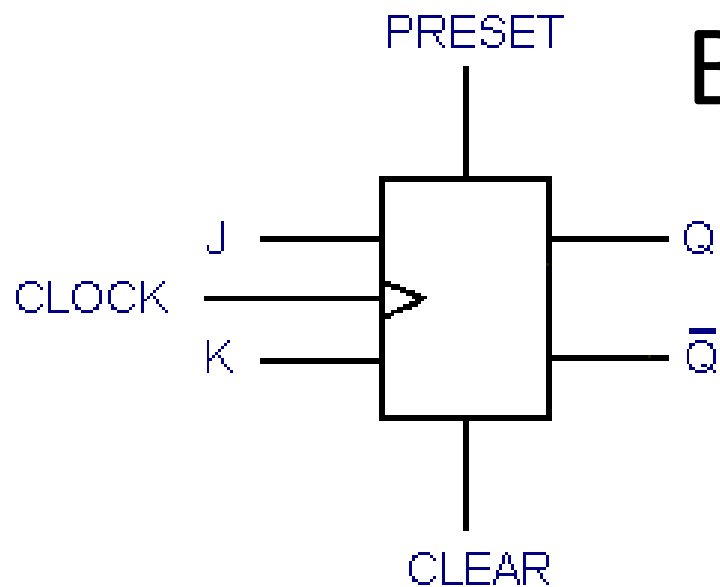
Bascule D-Edge

Au front montant : $D \rightarrow Q$



Bascule JK

Q = Queen / J = Jack / K = King

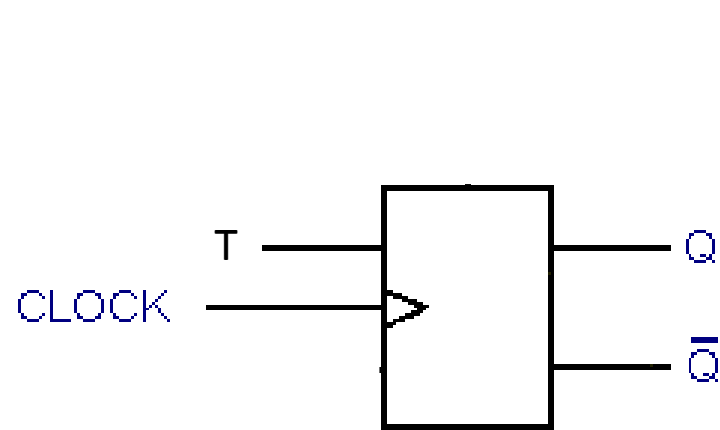


J	K		Q
0	0		Q
0	1		0
1	0		1
1	1		\bar{Q}

RST-ME avec rebouclages

Bascule T

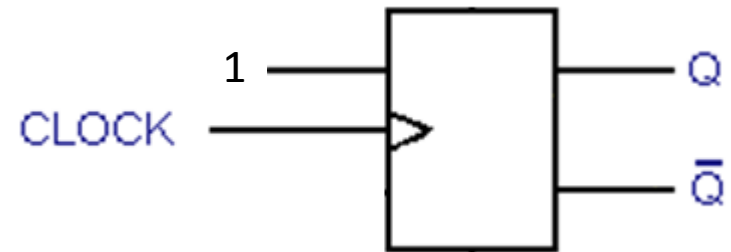
T = Toggle



T

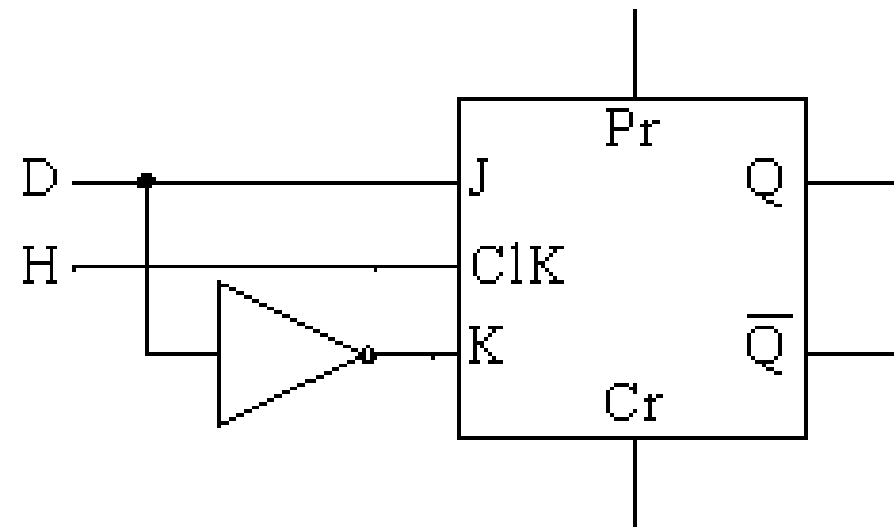
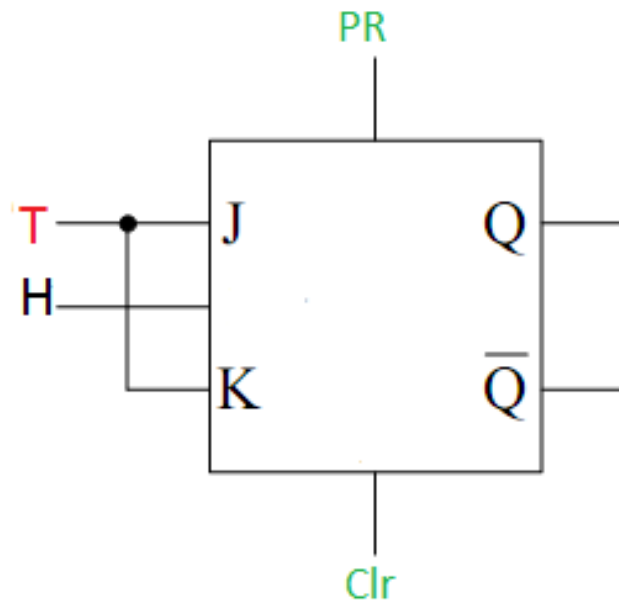
J	K		Q
0	0		Q
1	1		\bar{Q}

Particularité : pas fabriquée



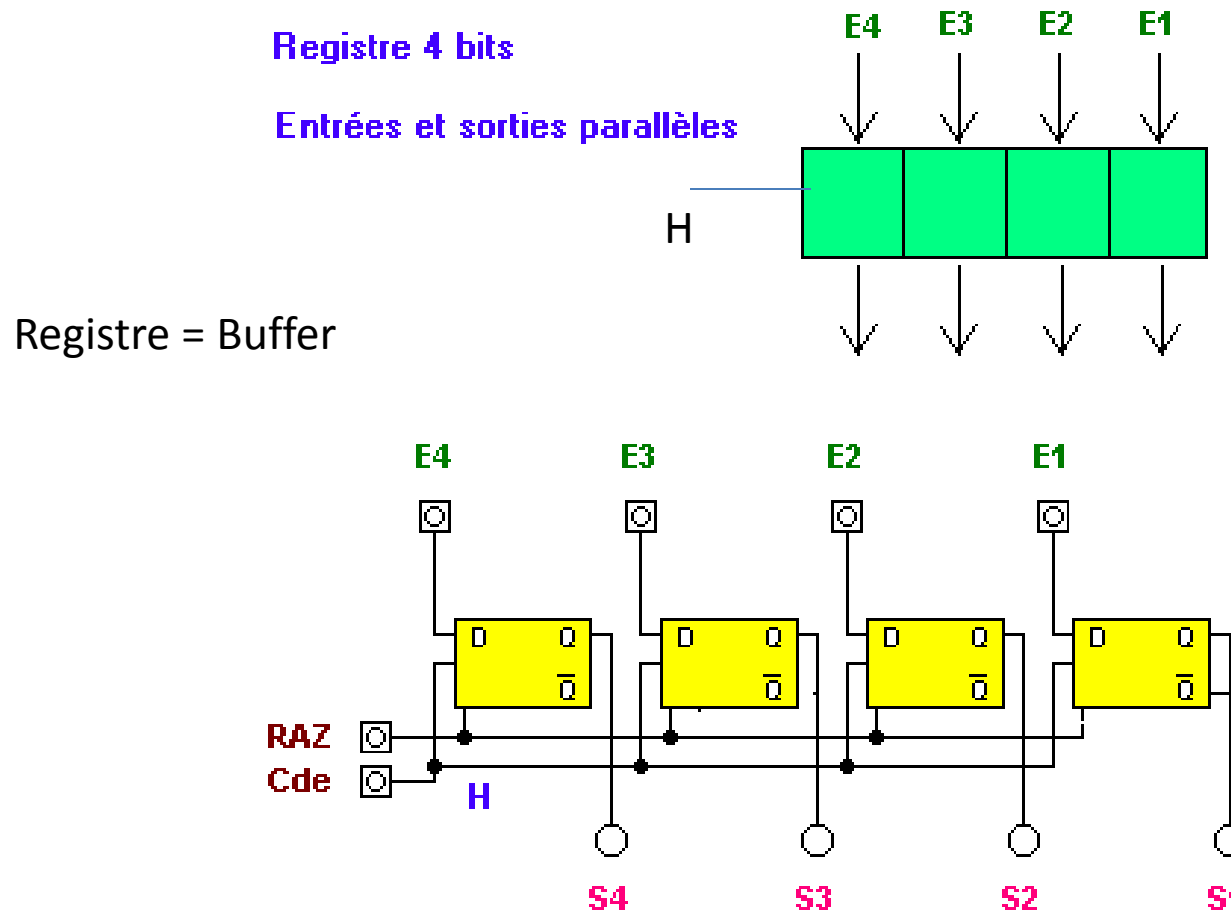
Division de fréquence par 2

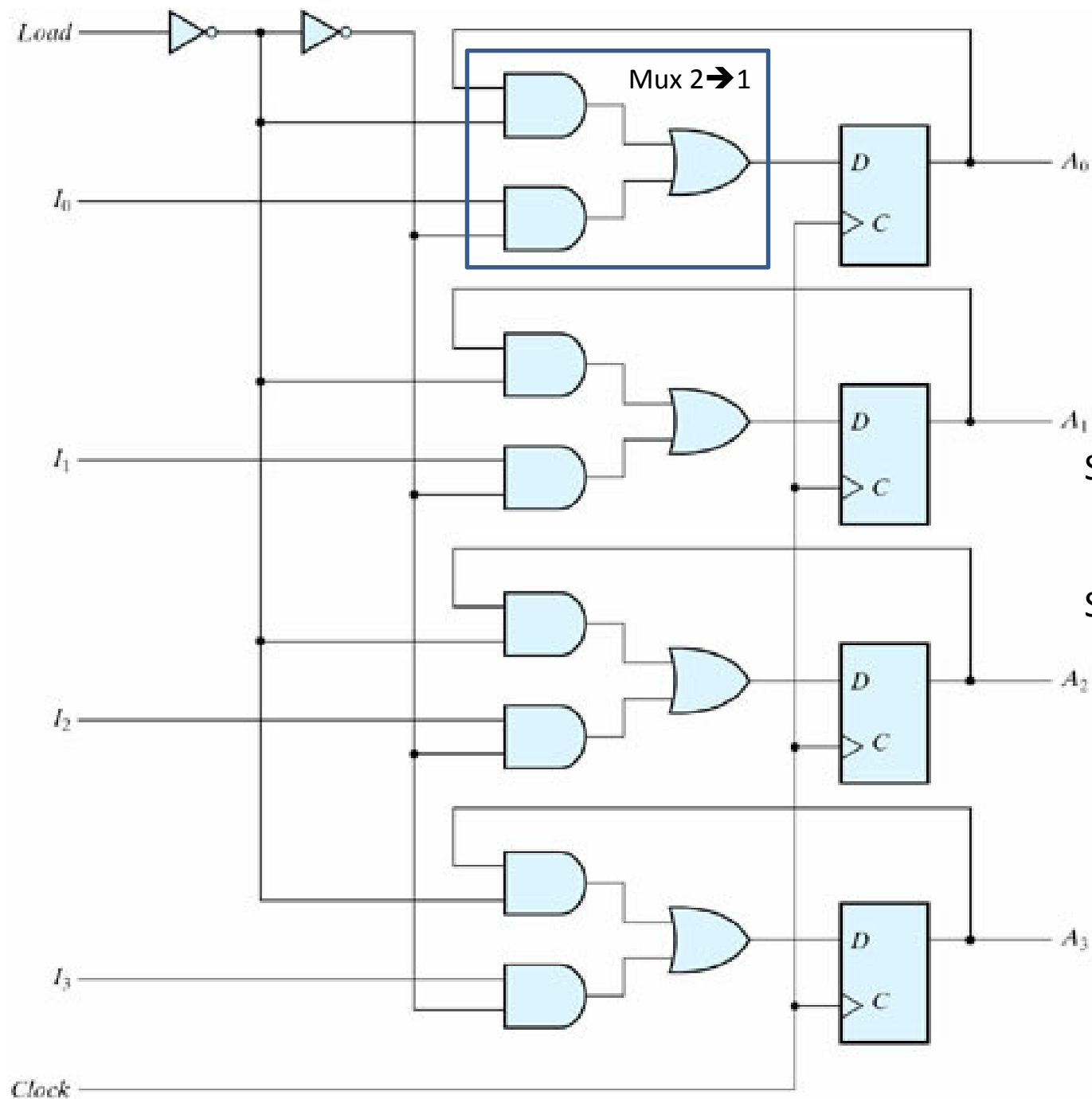
Equivalence des bascules



Registre, Registre à décalage

- Registre = association de bascules (souvent D)

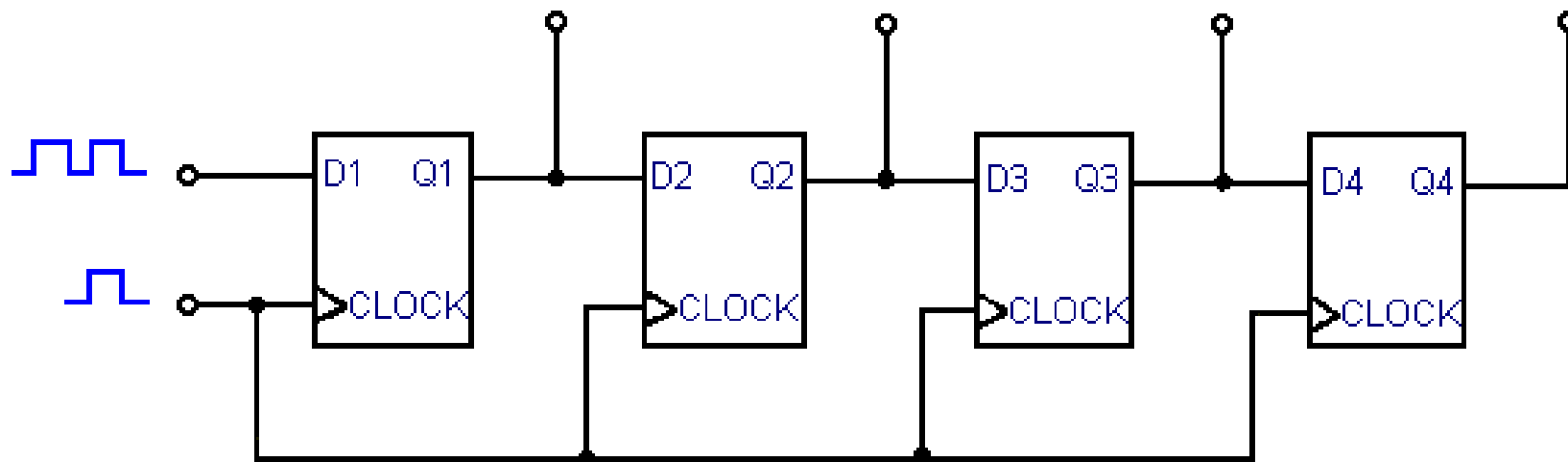




Si Load = 1
 $D = Q \rightarrow$ gelé

Si Load = 0
 $D = A \rightarrow Q = D$ sur $\uparrow C$

Registre, Registre à décalage



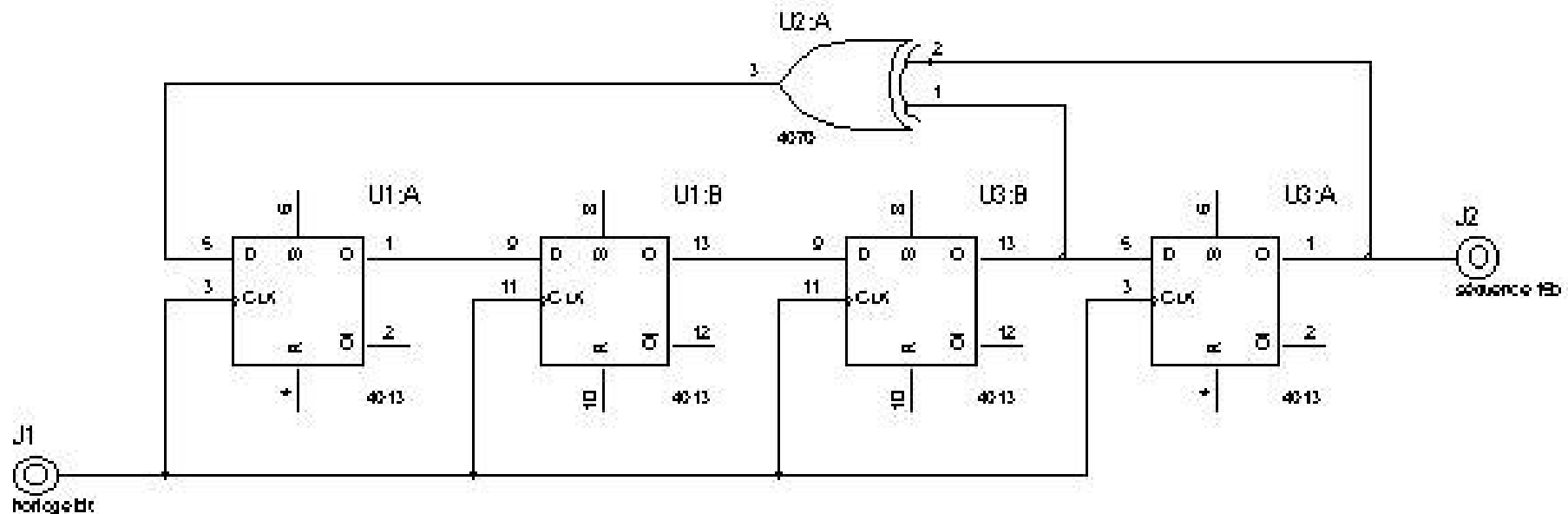
Registre à décalage avec plusieurs sorties.

$$(11001100)_2 * 2 = 11001100\mathbf{0}$$

Multiplication par 2 en binaire = décalage à gauche

Registre, Registre à décalage

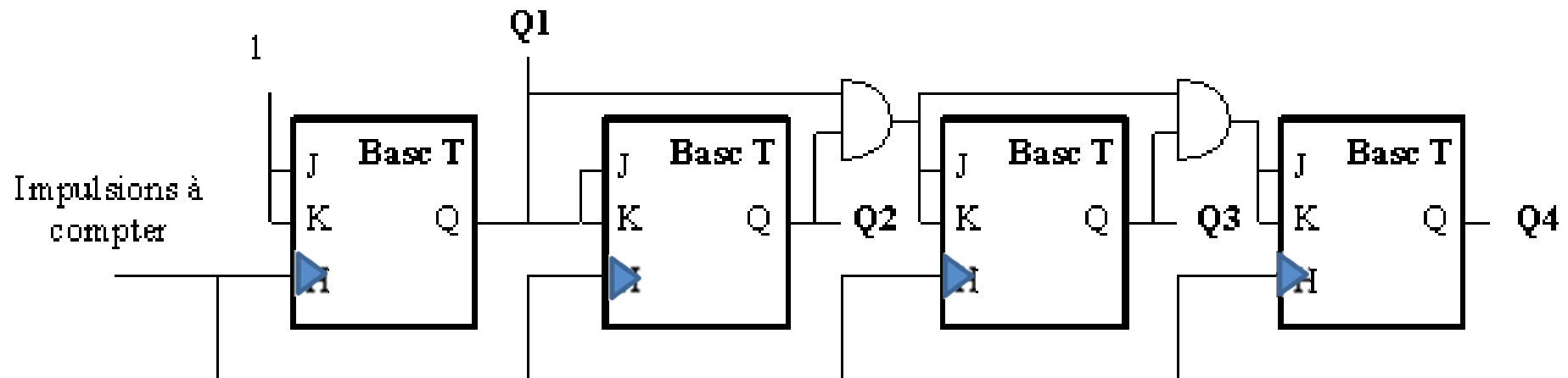
Générateur de nombres (pseudo-)aléatoires



Voir tables de Polynômes générateurs

Compteur

Parcours d'un code au rythme d'une horloge
(le plus classique : comptage binaire)



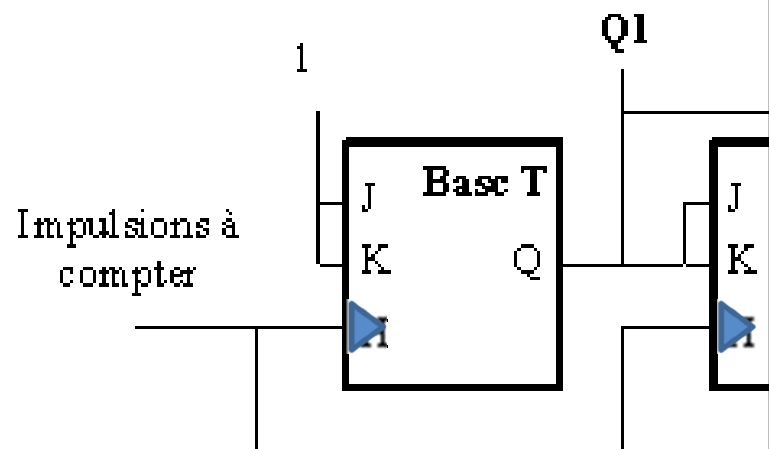
Algorithme : 0111 → 1000

Pour chaque sortie : On inverse quand tous les bits à droite sont à 1

1111 → 0000 Auto-cyclique

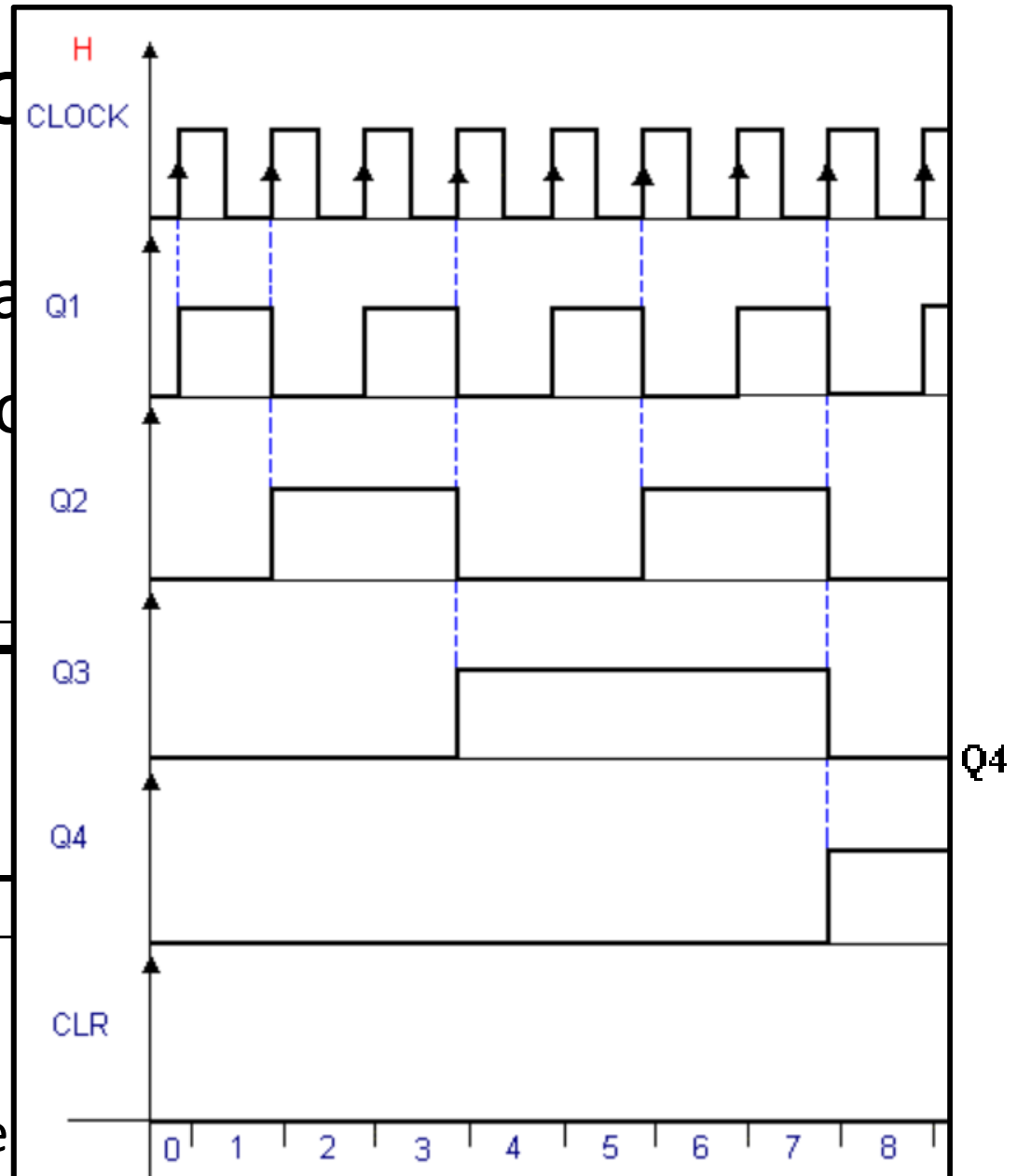
Co

Parcours d'un code a
(le plus classique : co



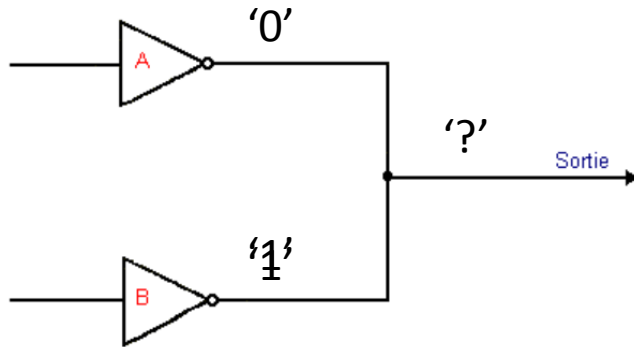
0111 → 1000

Pour chaque sortie : On inve

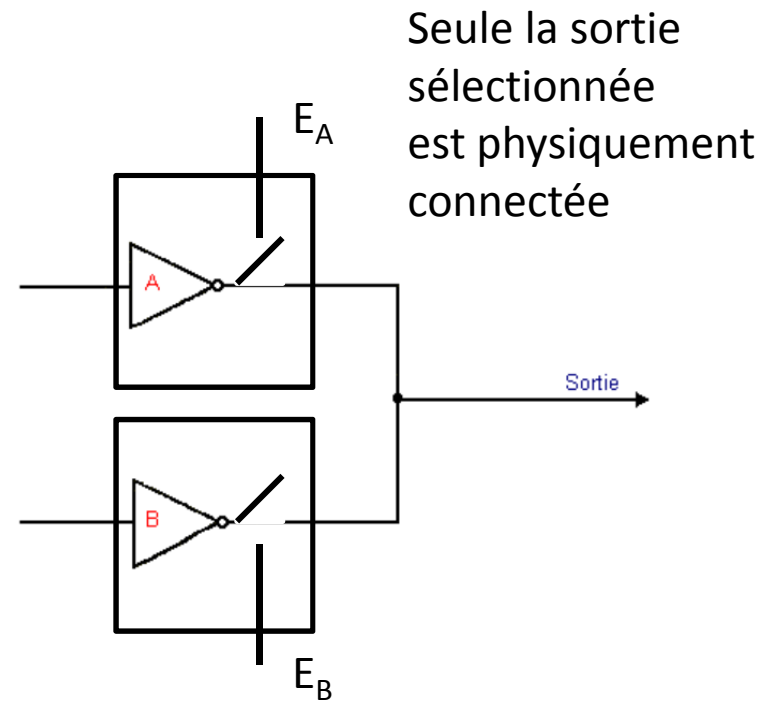
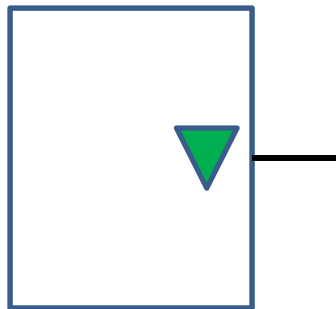


Sortie « haute impédance »

Tri State = Haute Impédance

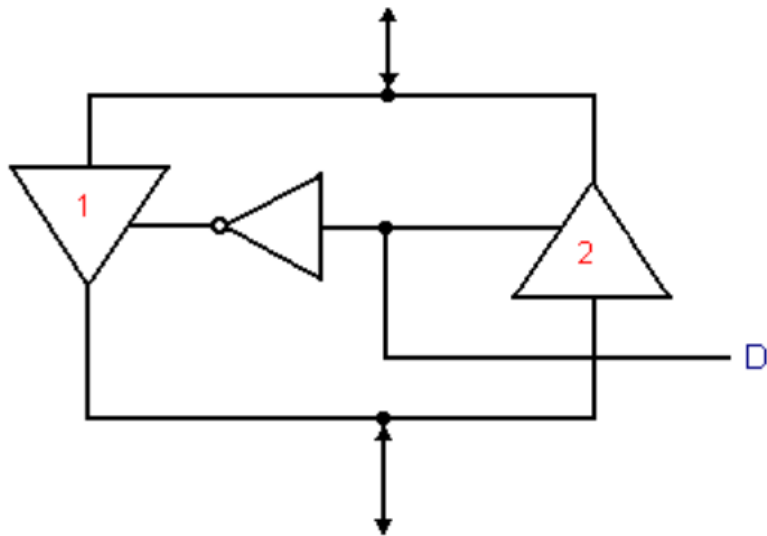


Court circuit
éventuellement
destructeur

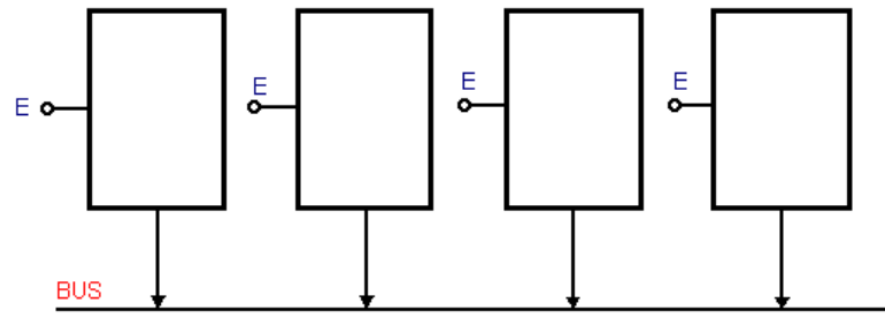


OE : Output Enable
E : Enable

Utilisation



Broche bidirectionnelle



Connexion partagée sur un bus
Une seule sortie active à la fois
(sinon : conflit de bus)
Idem multiplexeur

Circuits mémoires

(introduction des principes)

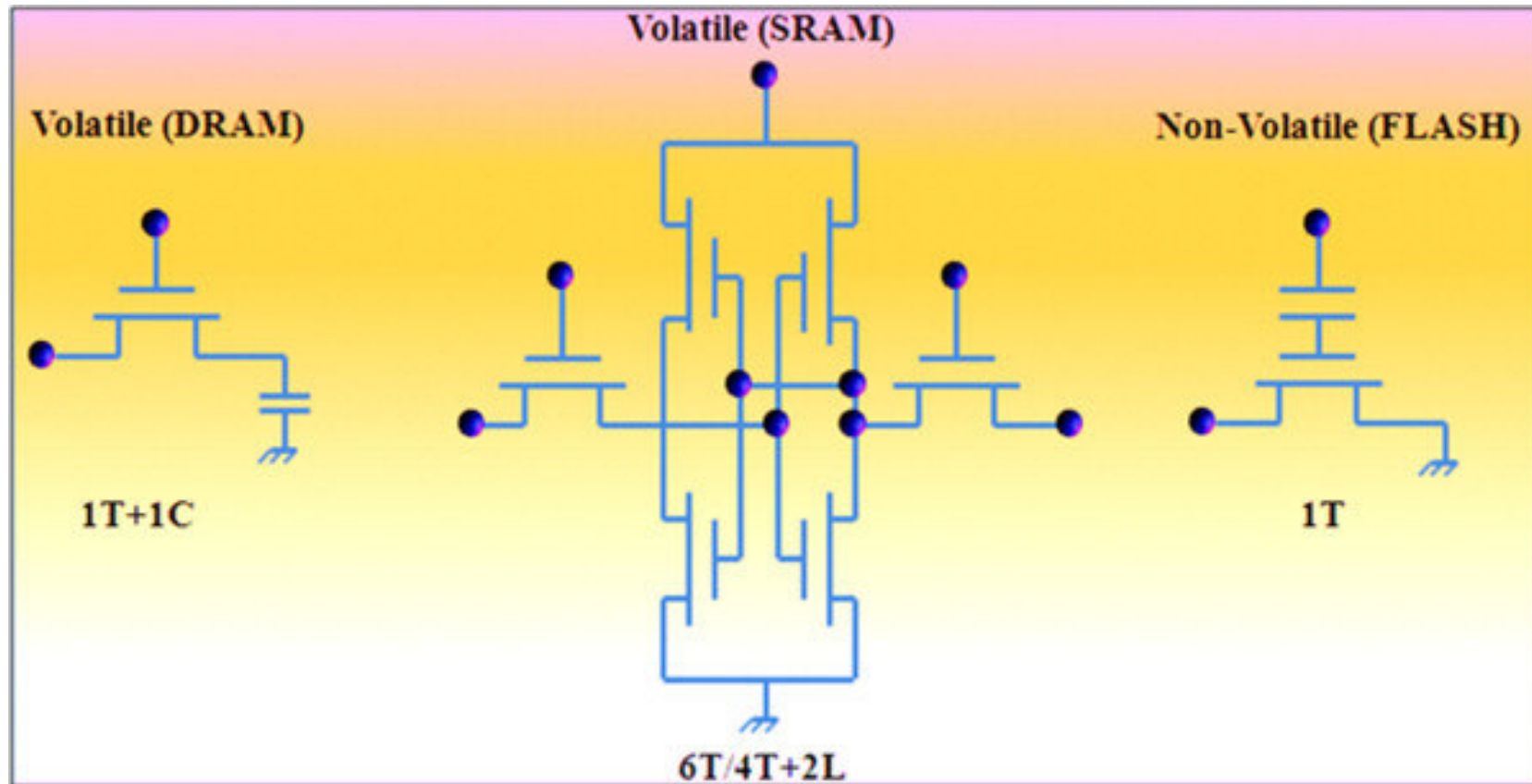
Circuits séquentiels

Circuit mémoire : principes

Mémorisation de l'information binaire

- Cellule de base (1 bit) \Leftrightarrow technologie
- Organisation en mots et table
- Mode d'accès
- Vieilles technologies non abordées
ROM, PROM, EEPROM, UV-EPROM ...
- Vocabulaire
 - RAM : Random Access Memory
 - ROM : Read Only Memory

Cellules



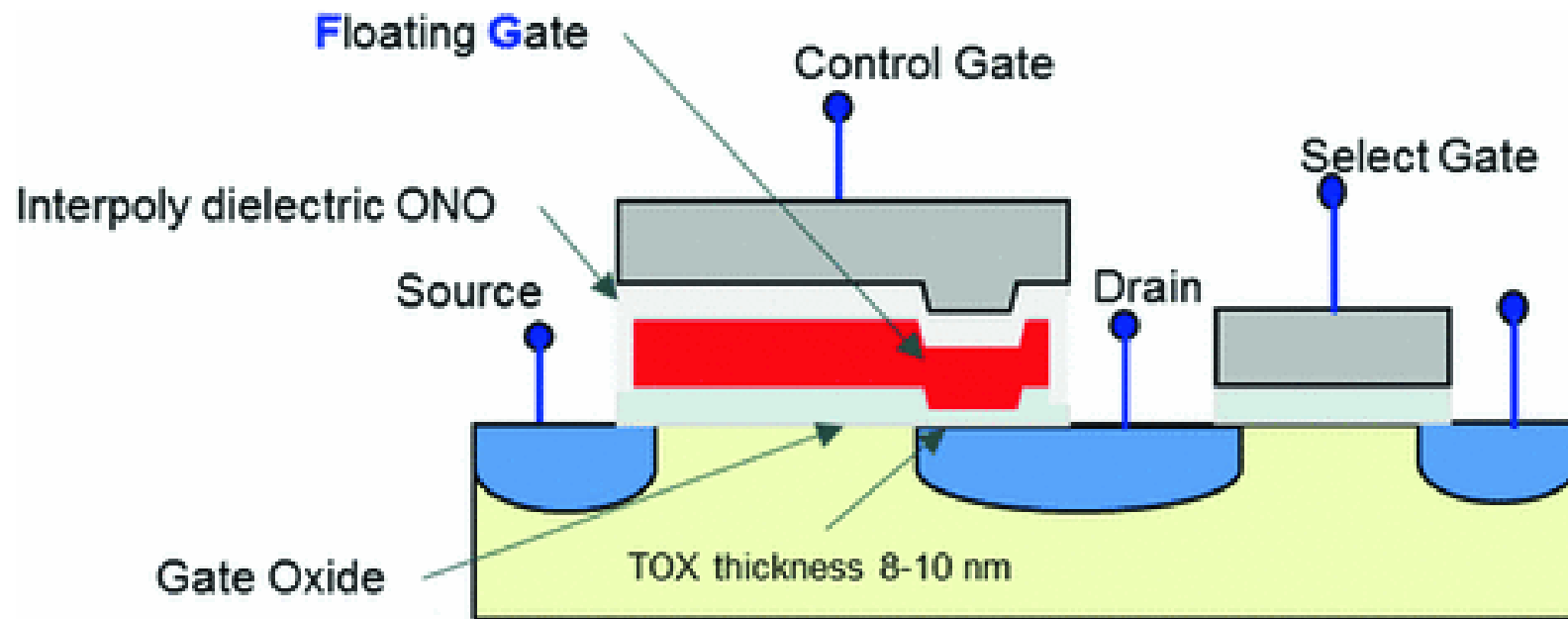
DRAM : Dynamic RAM – stockage de charges dans un condensateur – rafraîchissement

SRAM : Static RAM – mémoire RS

Flash : Injection de charges dans la grille flottante – gestion par blocs

développée au départ pour les militaires : effacement global possible

Cellule de mémoire Flash



Usages

- SRAM : mémoire interne, caches
 - + : rapide
 - : consommation, volatile
- DRAM : mémoire centrale
 - + : intégration, consommation
 - : gestion, rapidité, volatile même avec alim
- FLASH : clés USB, SSD, SmartCARD, configuration
 - + : Non volatile
 - : nombre de cycle « limité » (↗), écriture lente

Deux architectures d'accès : NOR et NAND

	Temps d'accès	Capacité	volatile	amovible	Prix	Utilisation
SRAM	< 10 ns	< 1 Go	oui	non	100 € / Go	Mémoire très rapides (registres, caches)
DRAM	50/100 ns	1-4 Go	oui	non	10-20 € / Go	Mémoire centrale
ROM/PROM	100 / 200 ns		non	non		Mémoires fixes (boot, mémoires de commande)
FLASH	100 / 200 ns	1-16 Go	non	oui et non	< 5 € / Go	Mémoire de masse, archives, objets courants
DISQUE magnétique	Env 10 ms	100-1000 Go	non	non	< 0,5 € / Go	Mémoire de masse, archives
CD	Env 100 ms	700 Mo	non	oui	< 1€ / Go	Archives
DVD	Env 100 ms	4,7-100 Go	non	oui	0,1 à 0,4 € / Go	Archives
Bandes DV	Qq. minutes		non	oui		Archives

Attention : les technologies évoluant vite juste le relatif est intéressant

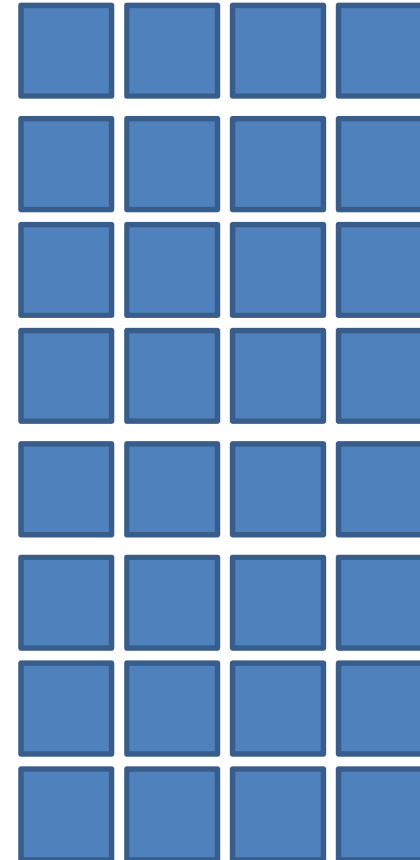
Organisation



Cellule = 1 bit

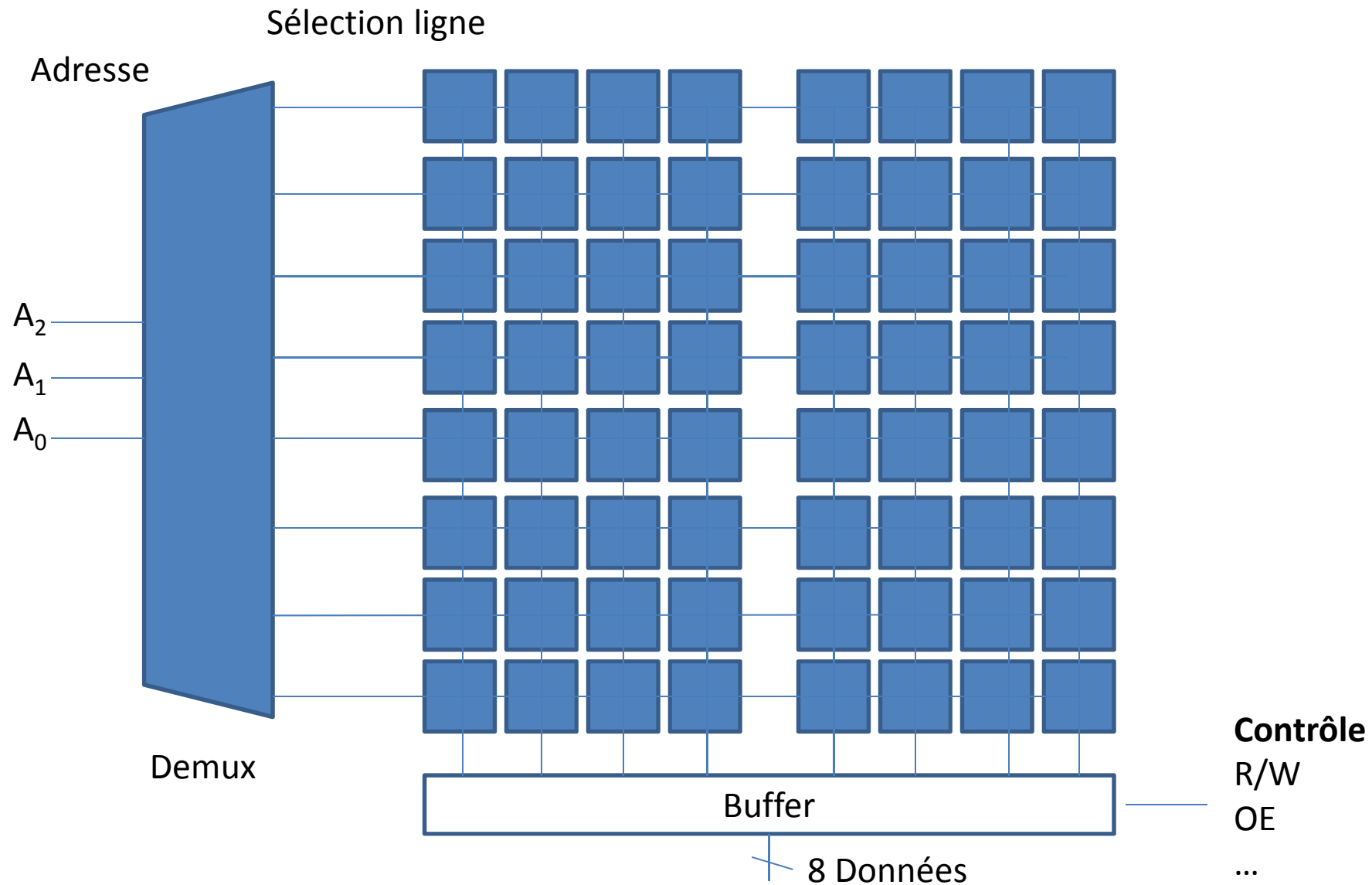


Mot = 4 bit

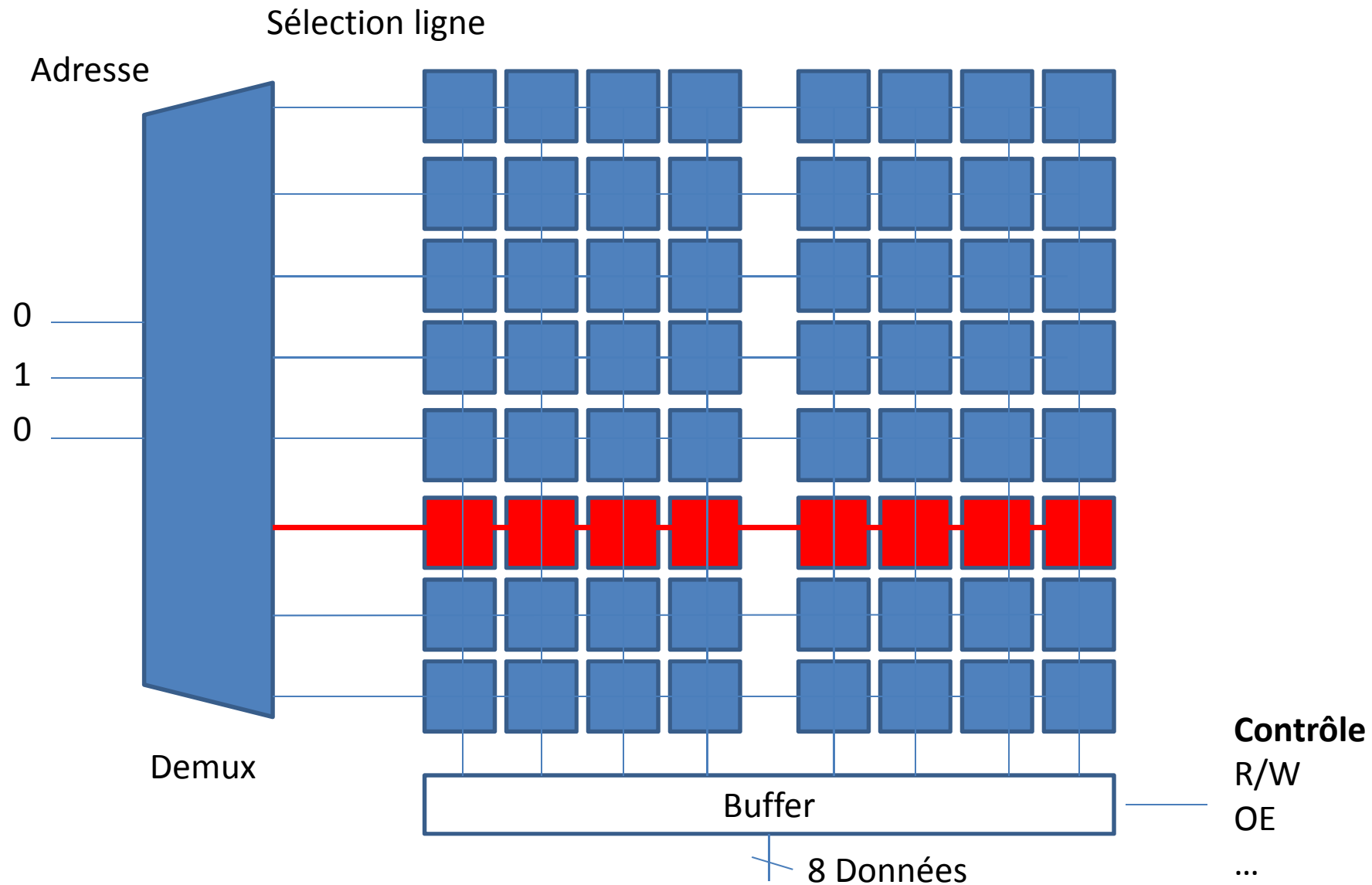


Plan = 8 mots

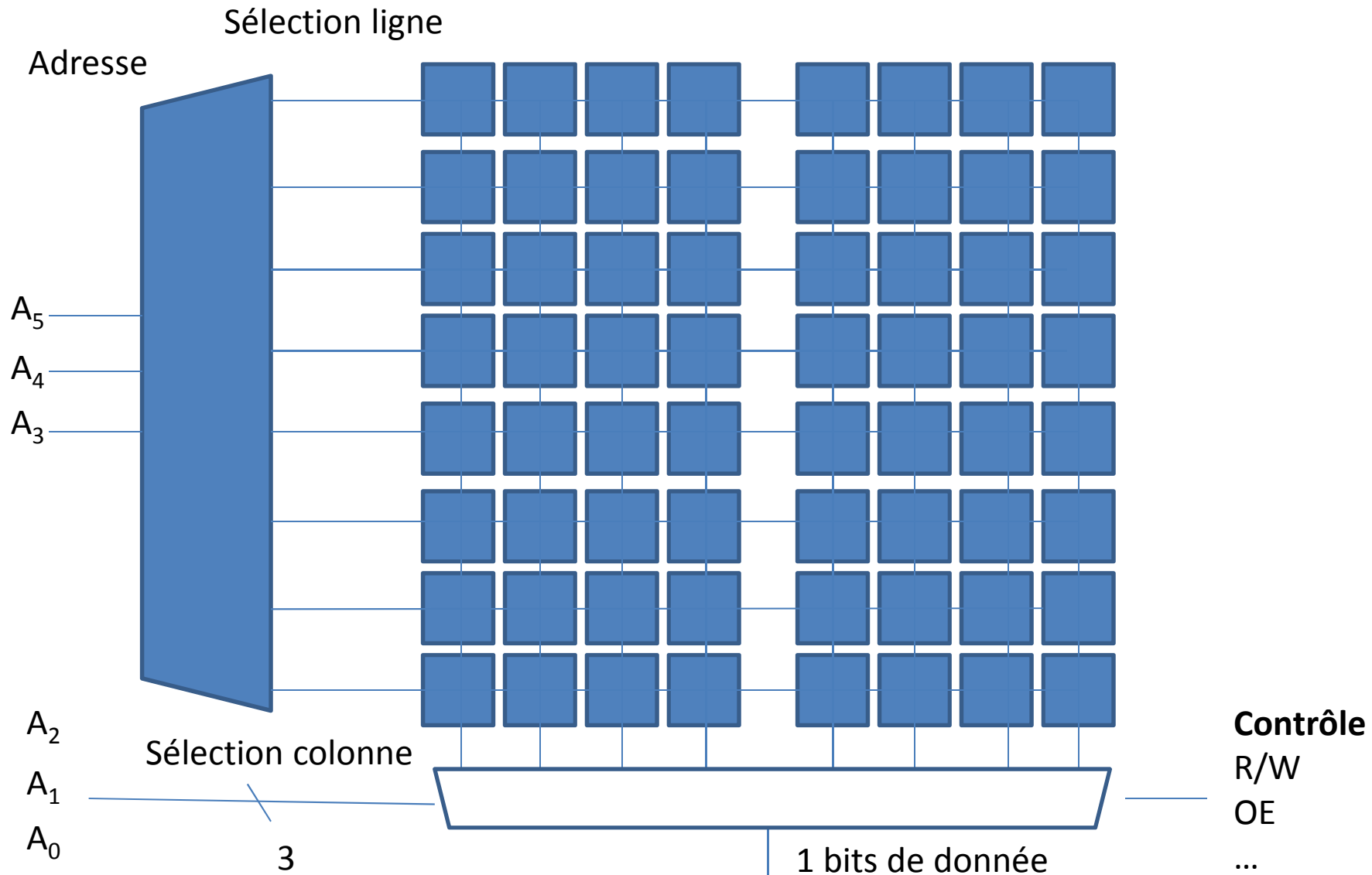
Accès par mot : Mémoire 8 octets



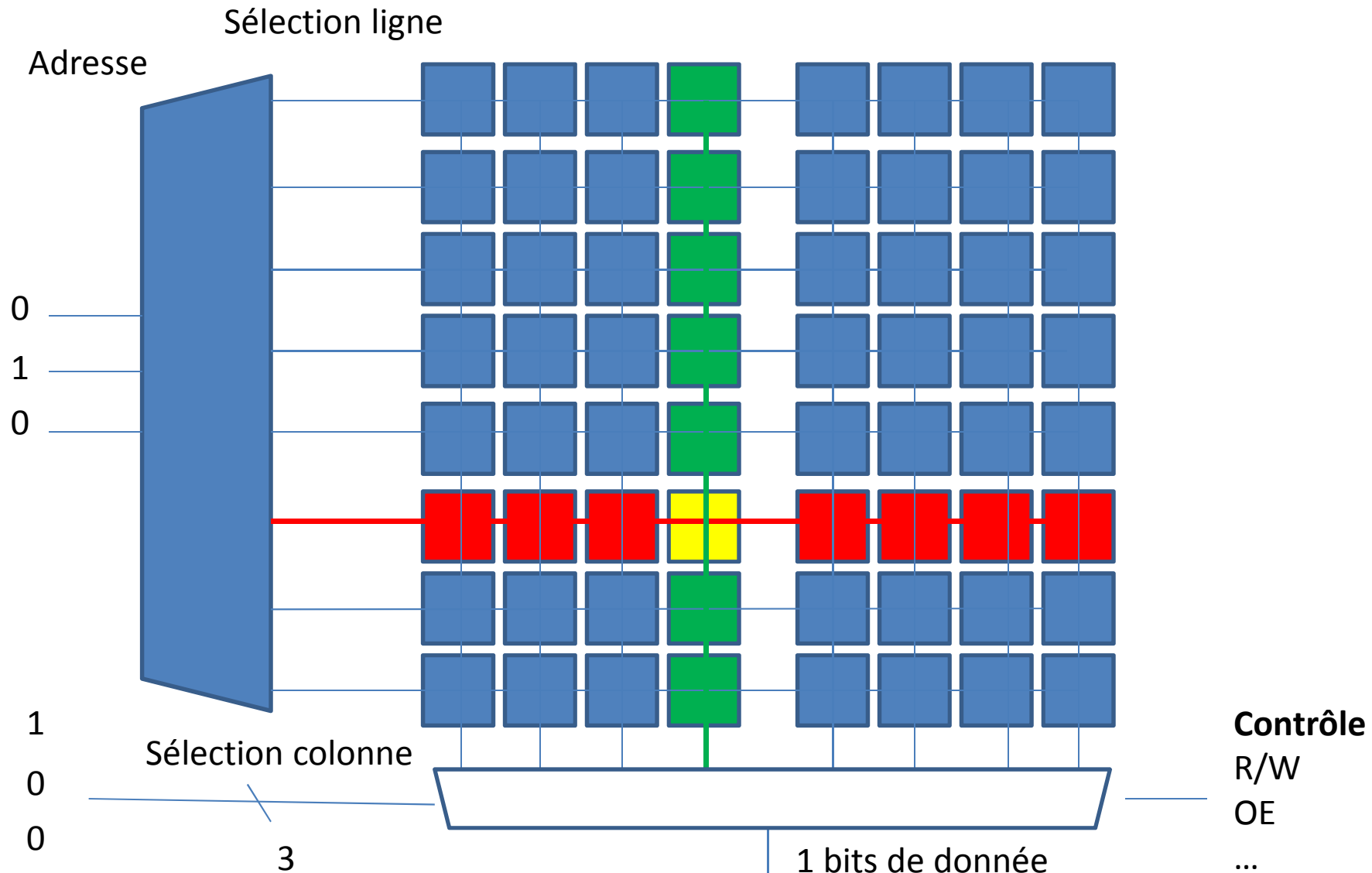
Accès par mot : Mémoire 8 octets



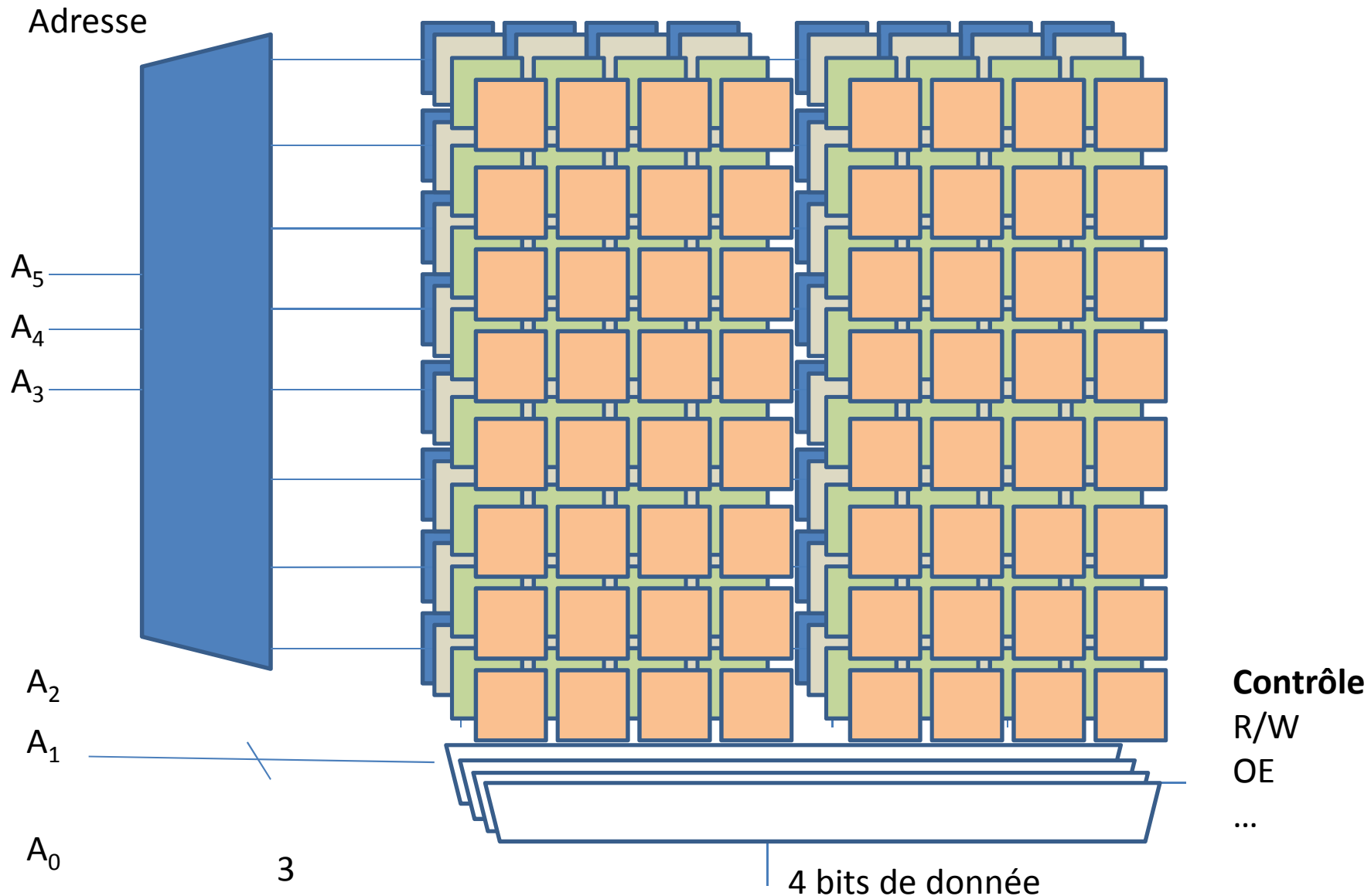
Accès par bit : Mémoire 64 bits



Accès par bit : Mémoire 64 bits

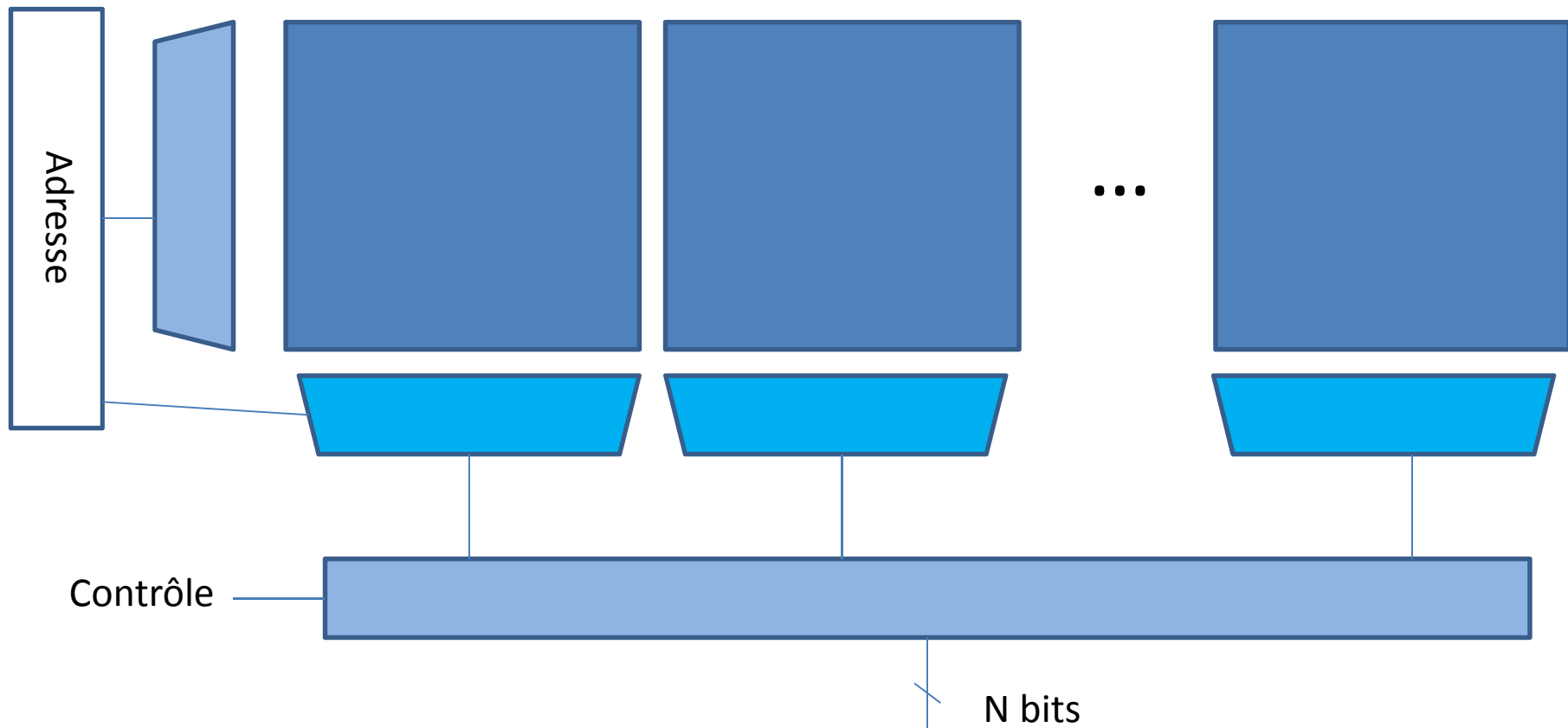


Plans de bits : 64 mots de n bits



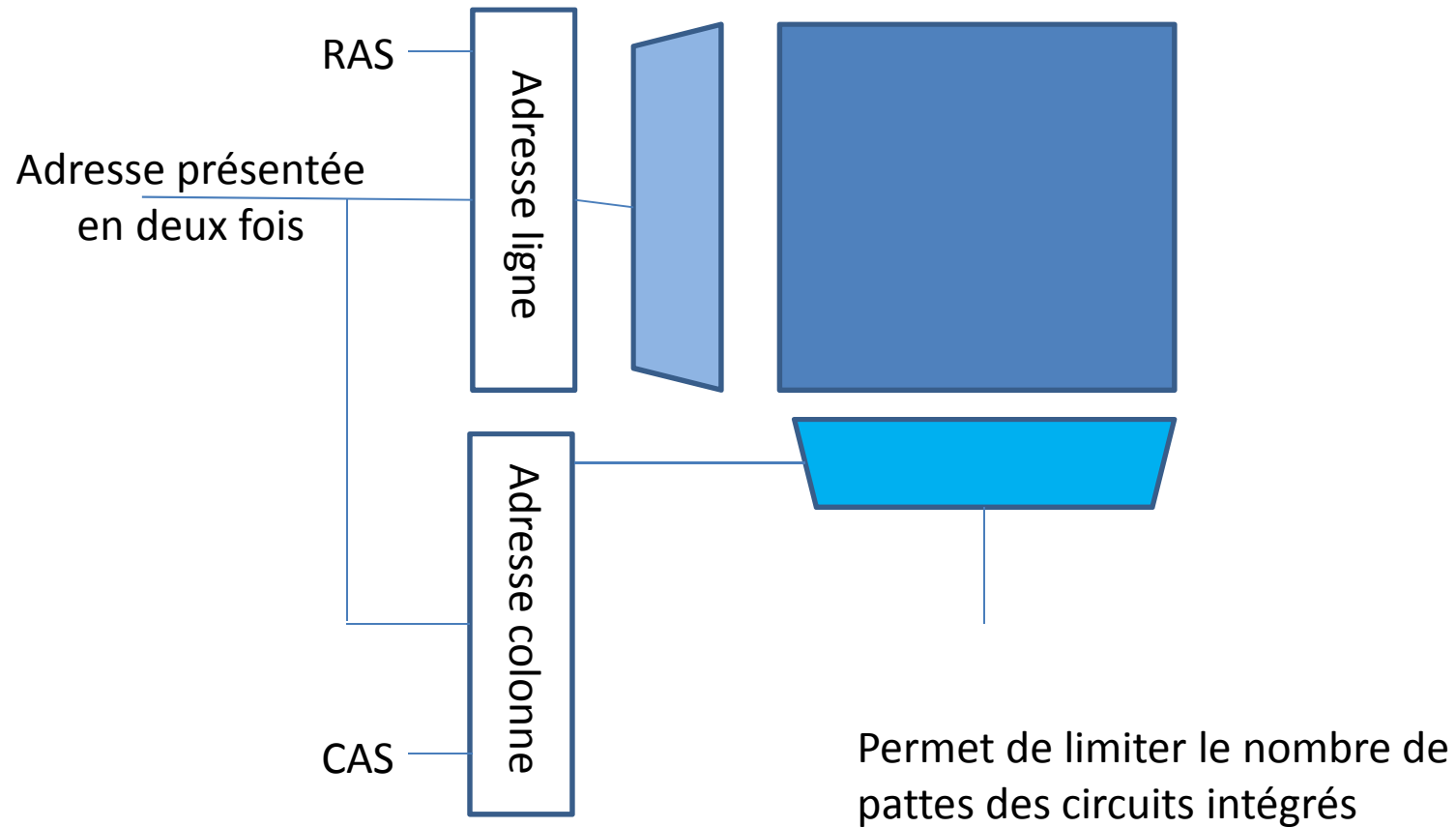
Architecture technologique

N plans à plat

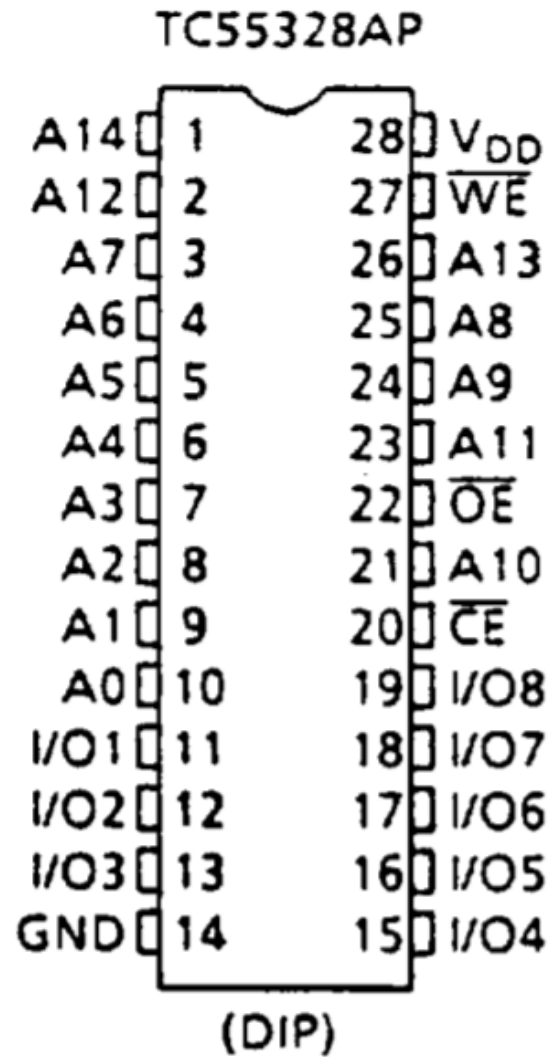


Evolution technologique : Implantation 3D (en couches)

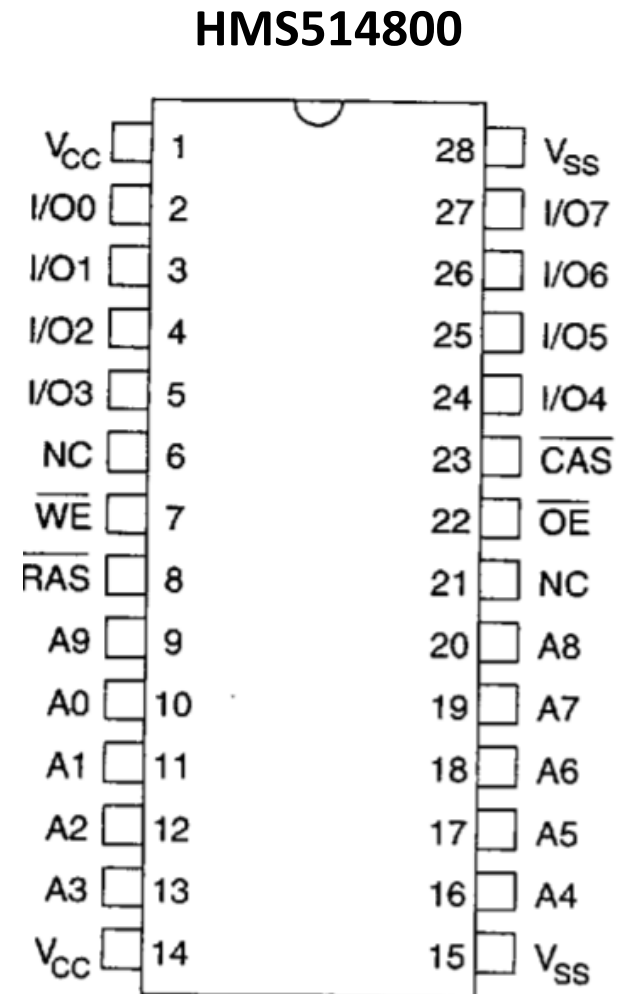
DRAM : Adressage multiplexé



Mémoire : vue externe (utilisateur)

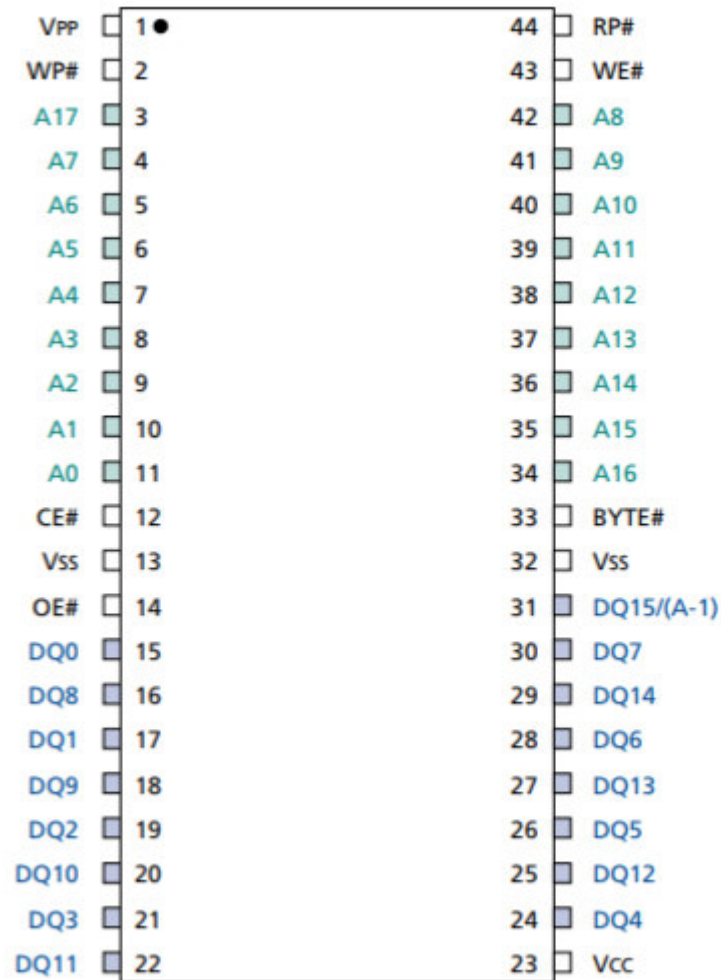


SRAM

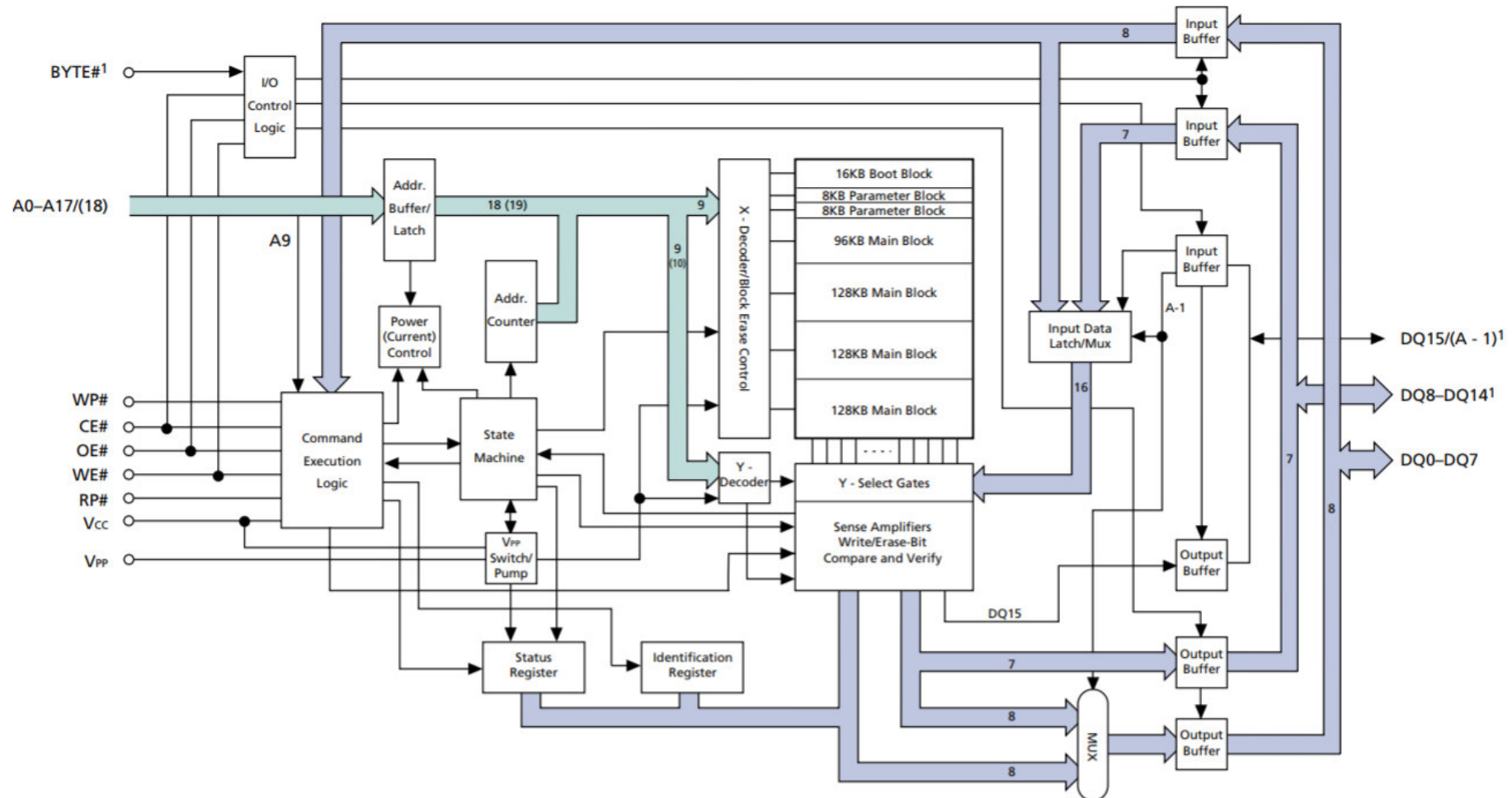


DRAM

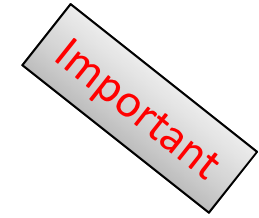
Flash 4Mb



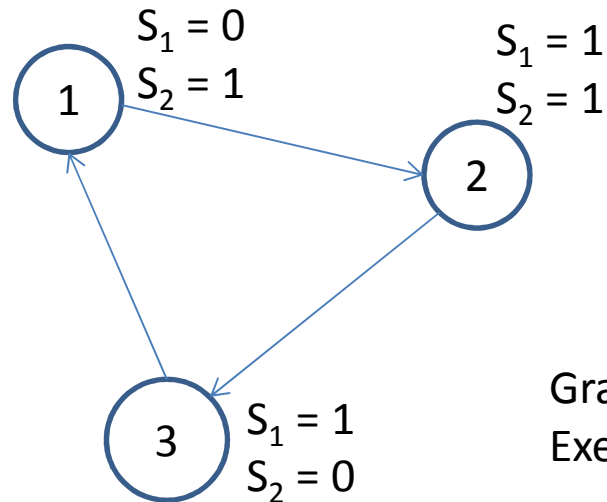
Flash : Structure interne



Notion de séquenceur



- Séquenceur : machine parcourant un code au rythme d'une horloge de façon conditionnelle ou inconditionnelle
- Notion de graphe : états liés par des transitions



Notation symbolique (parmi d'autres)

Sous entendu : graphe synchrone

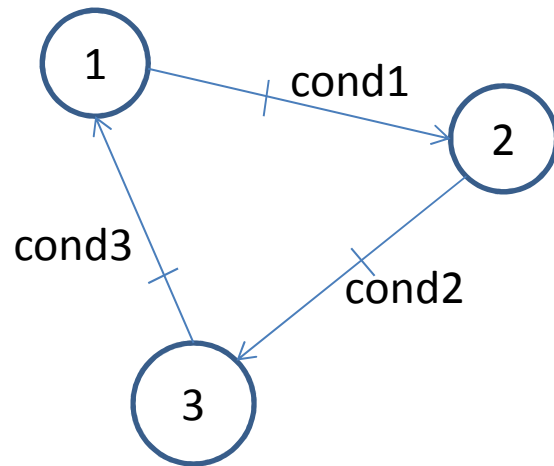
Transition : franchie sur un front d'horloge

Etat : impose la valeur des sorties

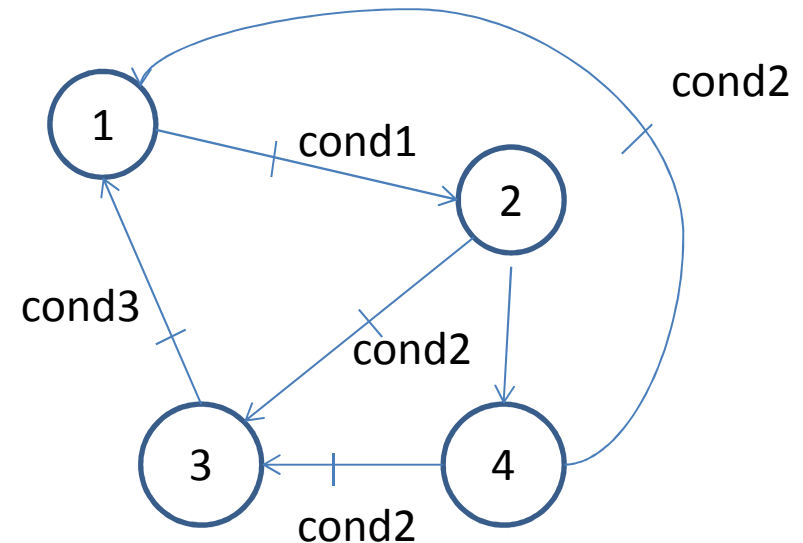
Graphe inconditionnel

Exemple : compteur

Graphe conditionnel



La transition n'est franchie que si la condition booléenne est vraie (sur un front de l'horloge)



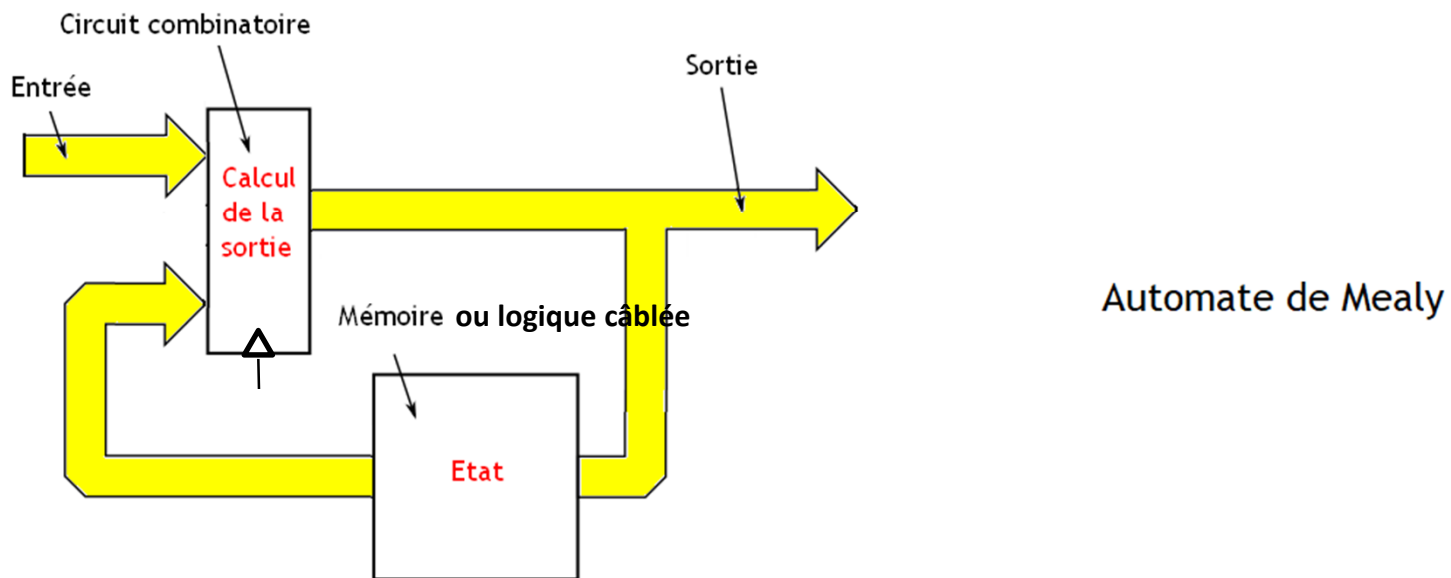
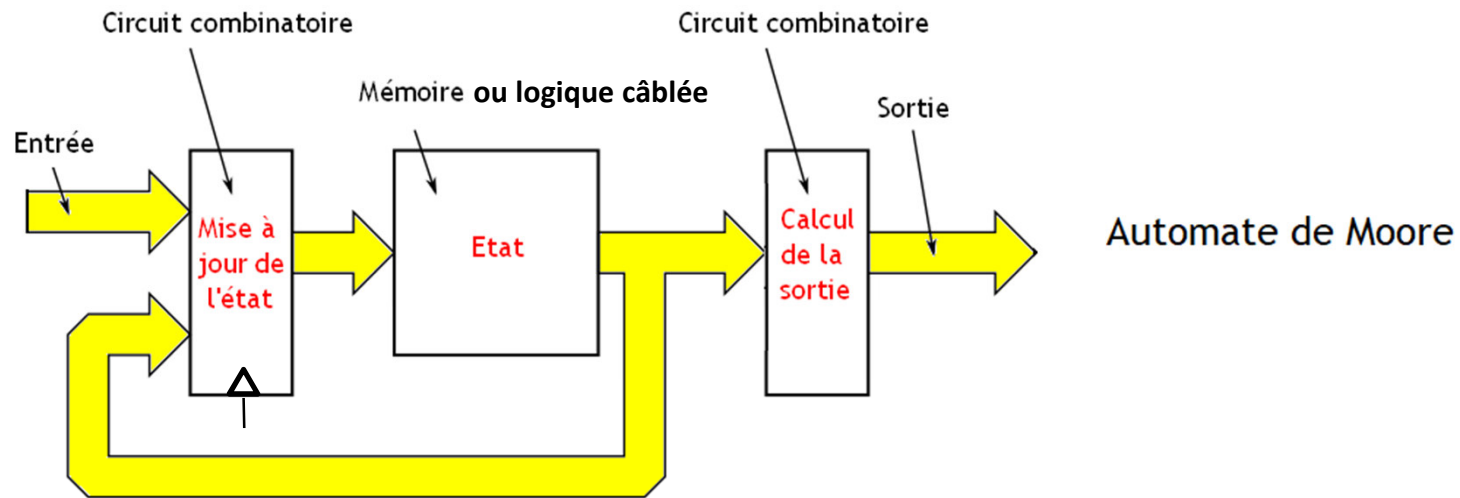
Les conditions permettent des fonctionnements complexes avec des choix (branchements)

Vocabulaire

Graphe = graphe de fluence = automate (d'états fini) = machine d'états

Théories associées : J. Von Neumann, réseau de Petri, Machine de Turing

Réalisation de séquenceurs



Synthèse des séquenceurs

- A partir d'une représentation symbolique (graphe synchrone)
 - Méthode de calcul « à la main » ➔ réalisation
 - Synthèse logique ➔ réalisation automatique quasi-optimale sur un critère (vitesse, consommation, surface ...)
- Possible : plusieurs milliers d'états
- On laisse ça aux spécialistes

Consommation des CI CMOS

Puissance consommée : $P \approx k.V^2.F + V \cdot I_{\text{leak}}$

- $P = f(V^2, F, I_{\text{leak}}, n)$ [n : dépend du nombre de transistors « switchés »]
- Proportionnelle à la tension au carré
- Proportionnelle à la fréquence
 - Consommation dynamique / charge capa parasites

Problème : intégration

- Dissipation d'énergie $\rightarrow \nearrow$ température
- Géométrie $\searrow \rightarrow$ Augmentation des fuites

Conclusion du module

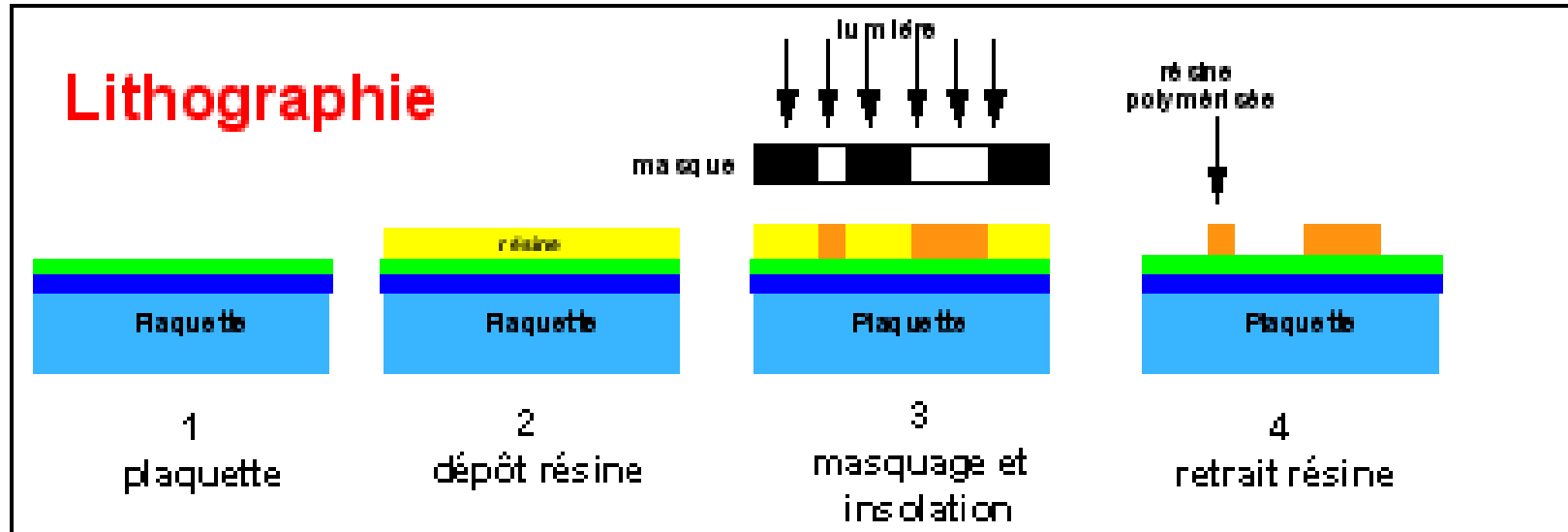
Nous sommes maintenant armés pour aborder
l'architecture des ordinateurs

(après avoir compris la forme
de nombres manipulés)

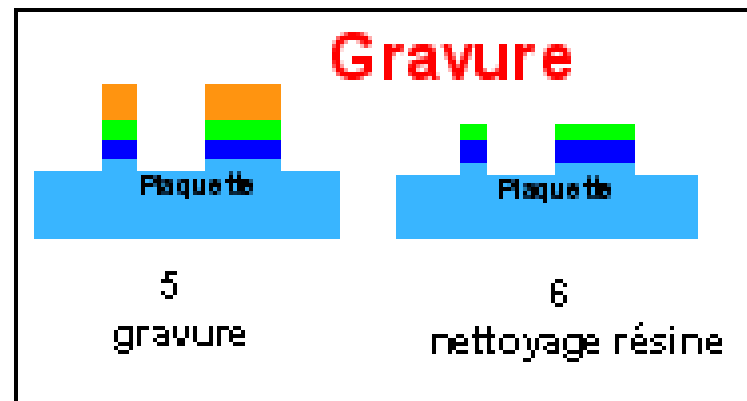
Culture

Front-end : principes de base lithographie et gravure

Lithographie



Gravure



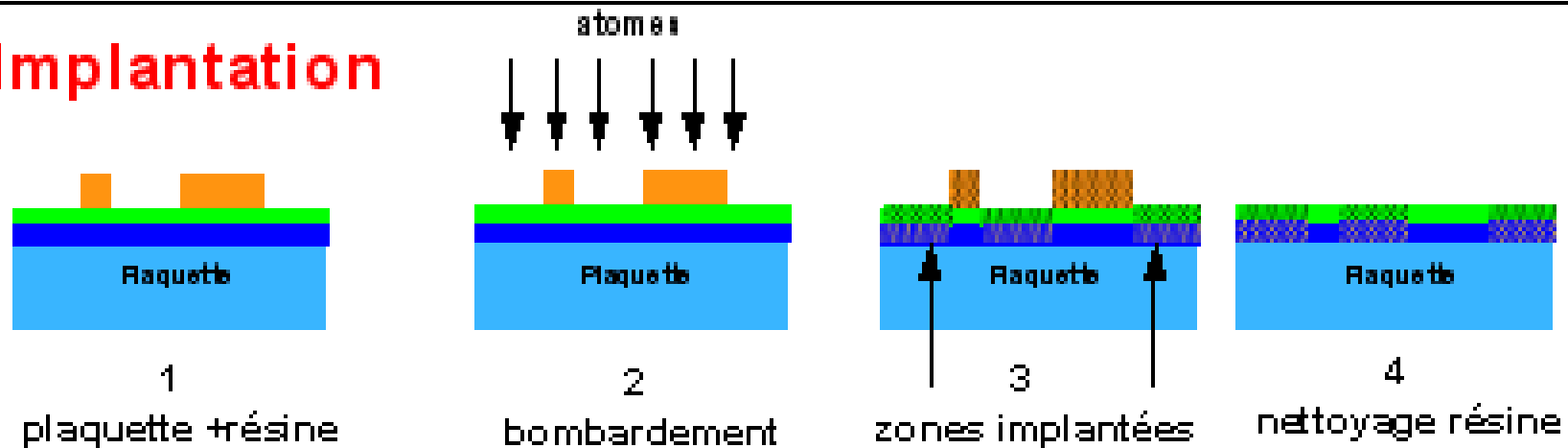
Au lieu de graver des couches, on peut aussi insérer des ions dans la surface de la plaquette par implantation ou diffusion, et faire croître des couches (épitaxie).

En répétant ces opérations on réalise les différentes couches d'un circuit intégré ou discrete

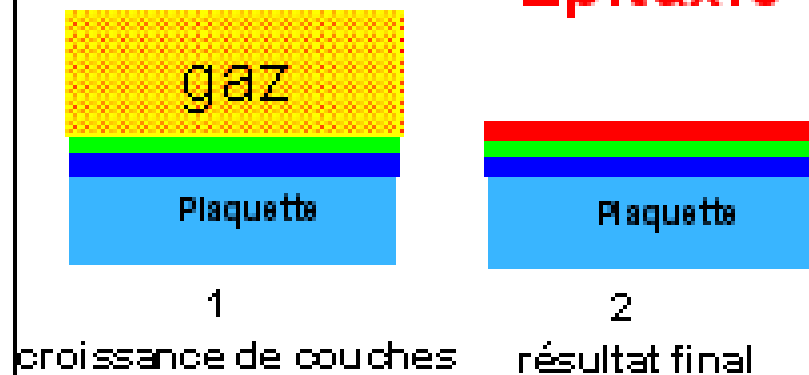


Autres procédés élémentaires

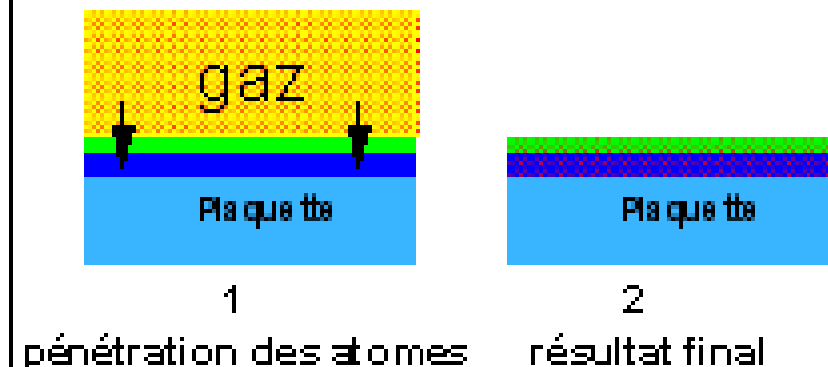
Implantation



Epitaxie

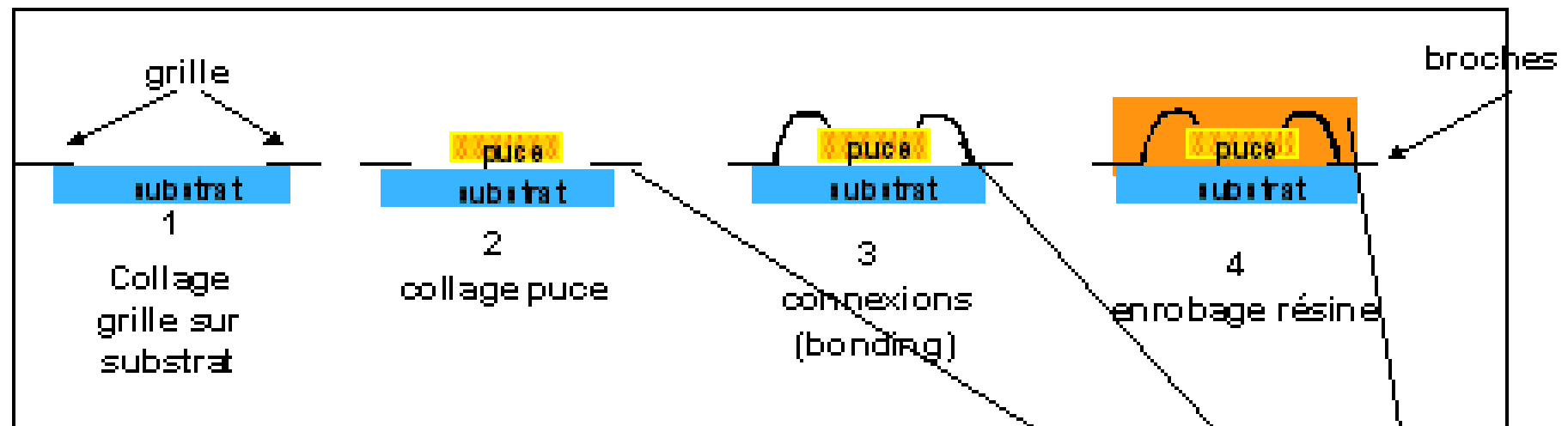


Diffusion



Back-end

Principe de base de l'assemblage



Il existe de nombreuses variantes d'assemblage pour les circuits:

- DIL : Dual in line : broches sur 2 cotés d'un rectangle
- QFP : Quad Flat Package : carré, avec broches sur le pourtour
- PGA : Pin Grid Array : Broches traversantes sous le circuit
- BGA : Ball Grid Array : Billes sous le circuit
- etc

