

Electronique Numérique

Famille des circuits Logique

Les circuits intégrés (logique) sont classés suivant leur caractéristiques et performances électriques et leur technologies de fabrication

- La tension d'alimentation
- Les niveaux logiques en entrée et en sorties
- Les courant en entrée et en sorties
- La température de fonctionnement
- Les performances dynamiques

- Technologie Bipolaire
- Technologie CMOS
- BiCMOS

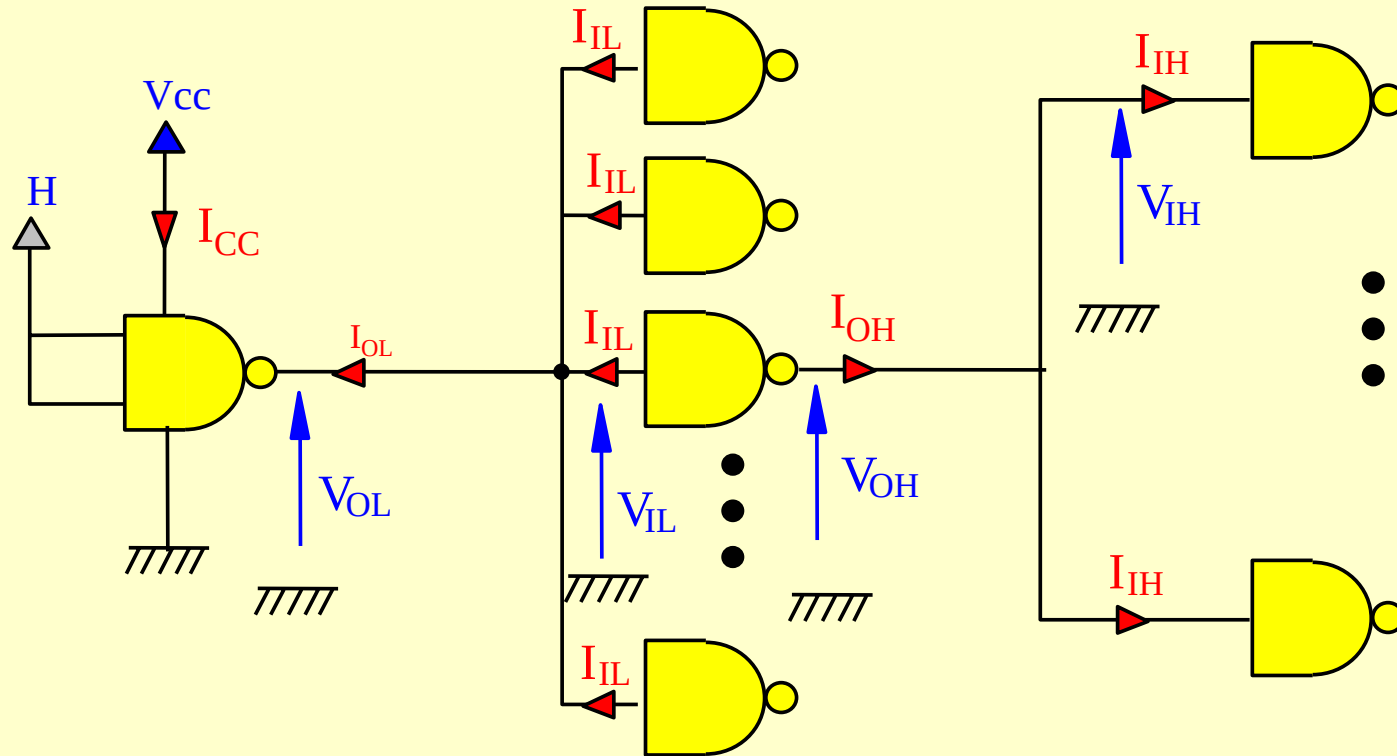
Nomenclature Commerciale

La nomenclature commerciale doit permettre (facilement) de déterminer la famille logique du circuit. Les constructeurs de circuits logiques ont finalement commencé à utiliser une nomenclature unifiée

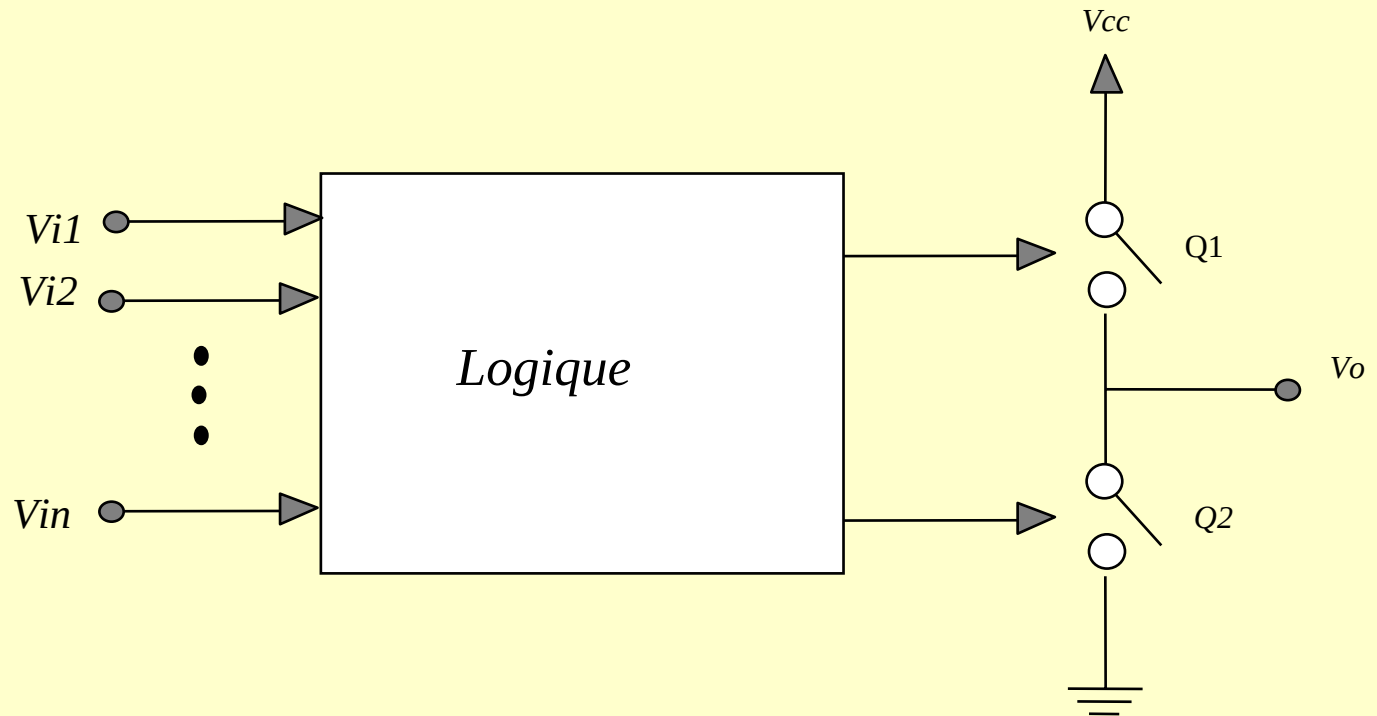
1	2	3	7	8
SN	74	LS	76	N

- 1** – Circuit normal ou ayant une spécification particulière.
SN=normal
- 2** – Série (température), 74=civile[0,70°], 54=militaire[-55,125°]
- 3** – Famille (Ici : Low power Schottky)
- 4,5,6** – On peut s'en passer pour le moment
- 7** – La référence du circuit : 76 = 2 bascules JK
- 8** – Boîtier : N = dual in line plastique

Conventions pour tension et courants



Modèle simplifié d'une porte Logique



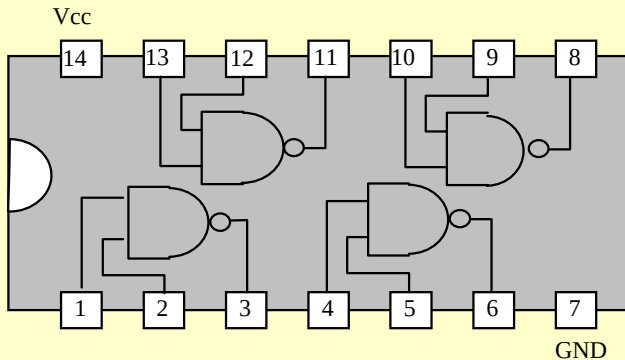
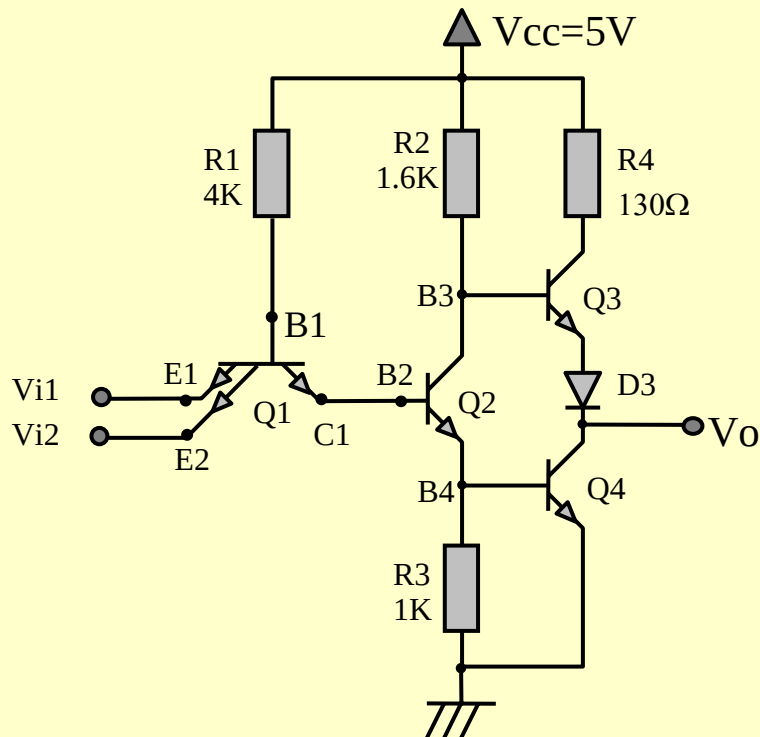
- $Q1$ fermé : sortie au niveau haut
- $Q2$ fermé : sortie au niveau bas
- $Q1$ et $Q2$ fermés : interdit par la logique (court circuit)
- $Q1$ et $Q2$ ouverts : sortie en haute impédance

Famille TTL : Transistor Transistor Logique

- Utilise des transistors bipolaires.
- S'alimente par 5V
- constituée de plusieurs variantes :

- **74** : TTL Standard, Obsolète , (10ns, 10 mW)
- **74 H** : High Speed, Obsolète , (6ns, 22 mW)
- **74 L** : Low Power, Obsolète , (33ns, 1 mW)
- **74 S** : Schottky, remplacée par AS , (3ns, 19 mW)
- **74 LS** : Low Power Schottky, fortement utilisée pendant longtemps, elle est maintenant en perte de vitesse (10ns, 2 mW)
- **74 AS** : Advanced Schottky , (1.5ns, 8 mW)
- **74 ALS** : Advanced Low Powe Schottky , (4ns, 1 mW)
- **74 F** : Fast , rare (3.7ns, 5.5 mW)

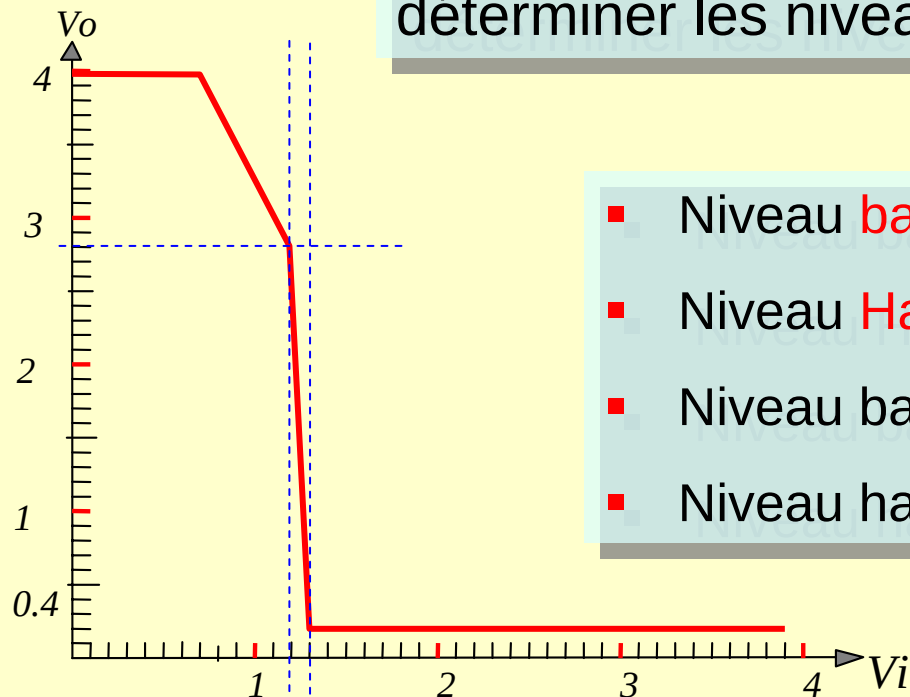
Porte TTL élémentaire 7400



- Le transistor Q1 ne fonctionne pas en transistor mais seulement comme trois diodes reliées par leurs anodes
- $V_{i1} = V_{i2} = 0 \rightarrow V_{B1} = 0.7V$
 - $\rightarrow$ Q2 et Q4 bloqués
 - $\rightarrow$ Q3 conduit et la sortie est de l'ordre de 4 Volts
- $V_{i1} = V_{i2} = 5V$, si les jonctions B1-Ei conduisent $\rightarrow V_{B1} = 5.7V \rightarrow$ c'est trop pour les 3 jonctions en série
 - $\rightarrow$ Les jonctions B1-C1, B2-E2 et B4-E4 conduisent $\rightarrow V_{B1} = 2.1V \rightarrow$ les jonction d'entrée bloquées
 - $\rightarrow$ Q2, Q4 saturé, $V_{B3} = 0.9V \rightarrow$ Q3 bloqué $\rightarrow$ la sortie est de l'ordre de 0.2V

Caractéristique de transfert

Si on fait varier la tensions d'entrée entre 0 et 5V. On obtient la caractéristique de la porte. D'après cette courbe on peut essayer de déterminer les niveaux logiques :



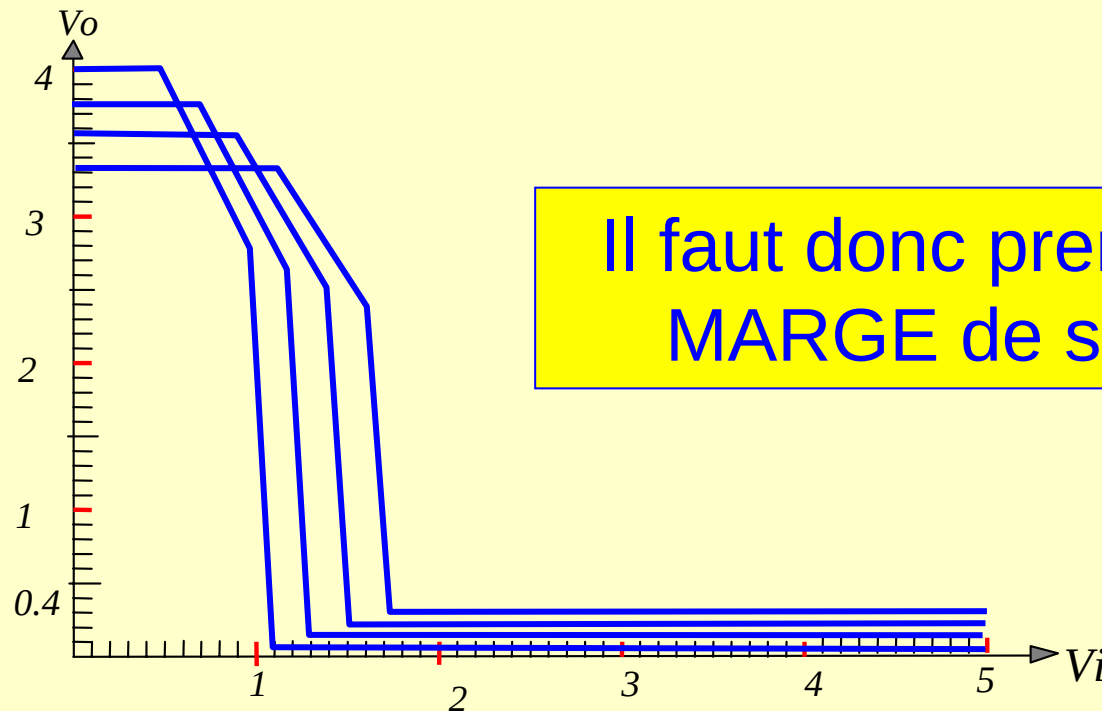
- Niveau **bas** en **entrée** : **[0V-1.2V]**
- Niveau **Haut** en **entrée** : **[1.3V-5VV]**
- Niveau bas en **sortie** : **0.2V**
- Niveau haut en **sortie** : **[2.8V – 4V]**

Dans la pratique

Si on trace cette caractéristique

- Pour **plusieurs** circuits 7400,
- Pour le même Circuit mais à **différentes températures**
- Pour le même Circuit mais à **différentes charges**

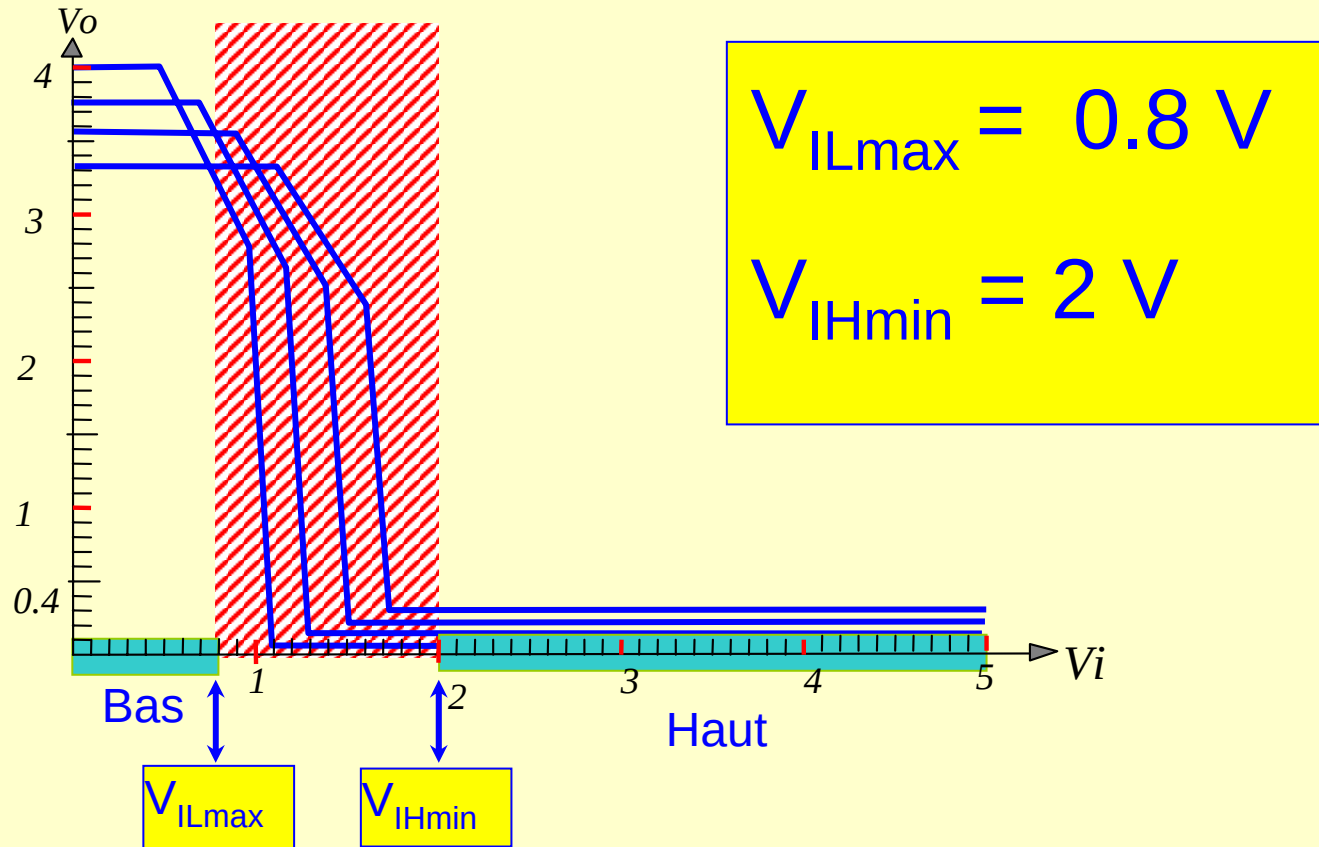
On obtient une dispersion des courbes



Il faut donc prendre une
MARGE de sécurité

Niveau logique TTL en entrée

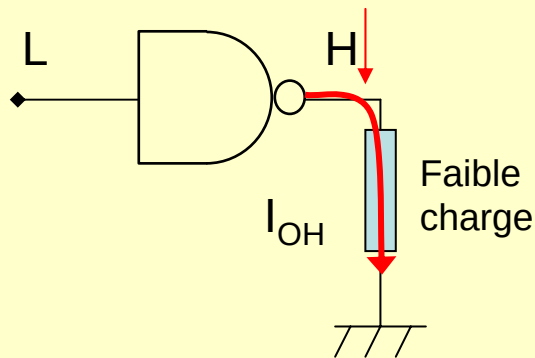
C'est nous qui appliquons les tension **d'entrée**,
C'est à nous d'éviter la zone de transition qui peut introduire des erreurs



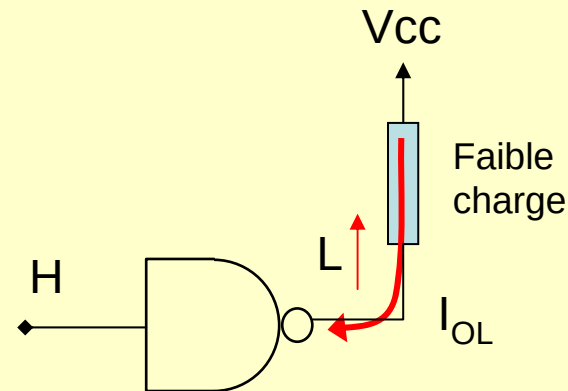
Les niveaux de sortie

C'est la porte qui impose la tension de **SORTIE** mais :

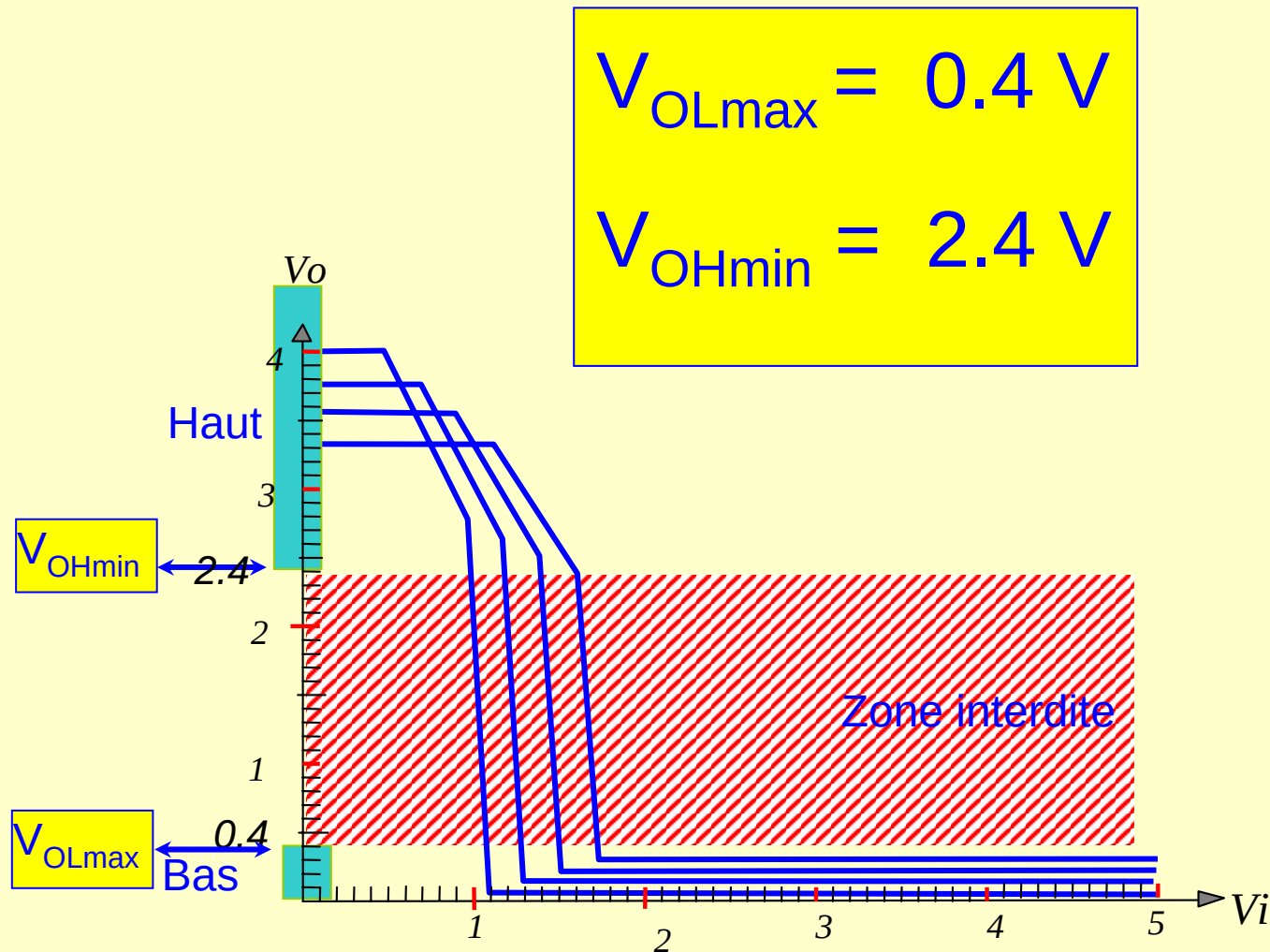
Si on demande trop de courant à une sortie au niveau haut, alors la tension de sortie baisse



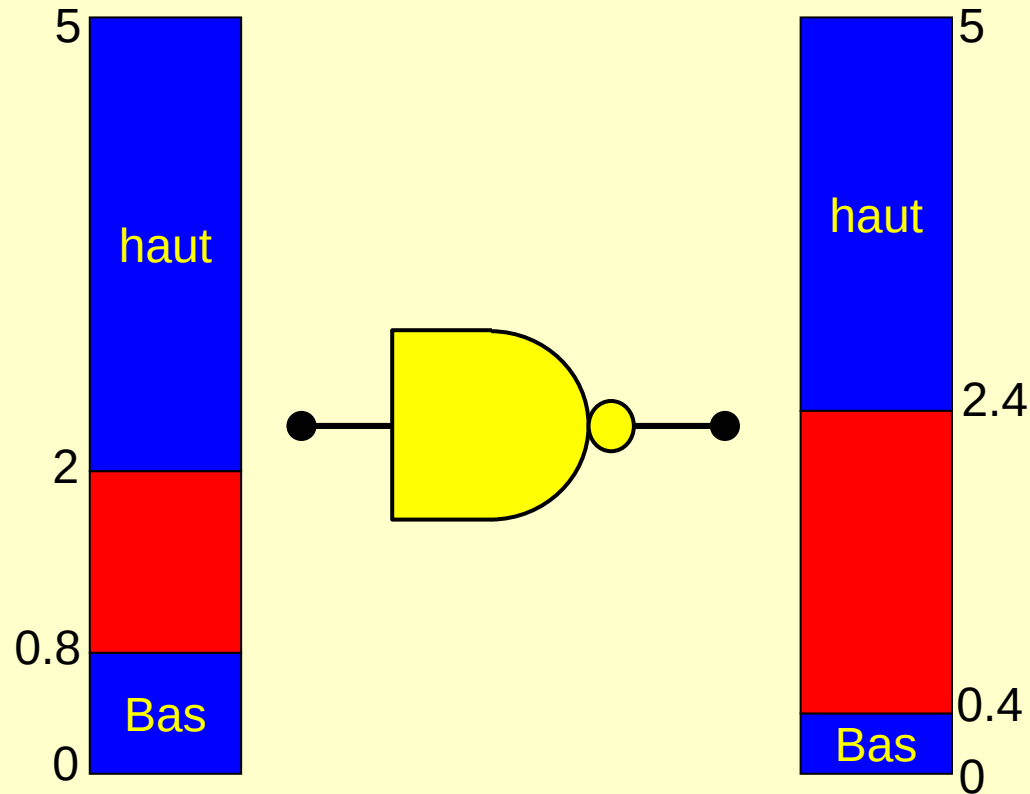
Si on injecte trop de courant à une sortie au niveau bas, alors la tension de sortie monte



Niveau logique TTL en Sortie

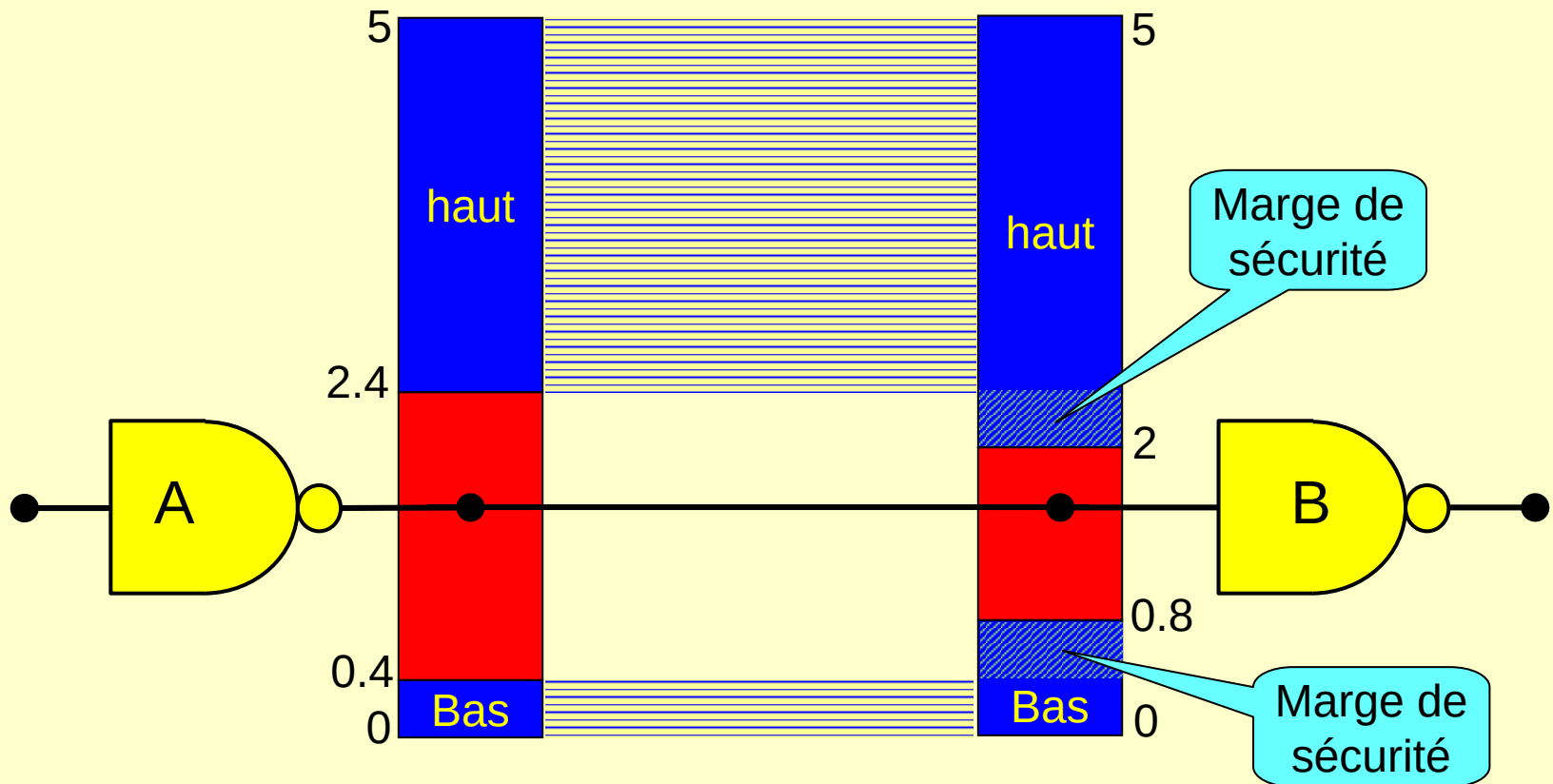


Niveaux logiques de la famille TTL 5V (récap)



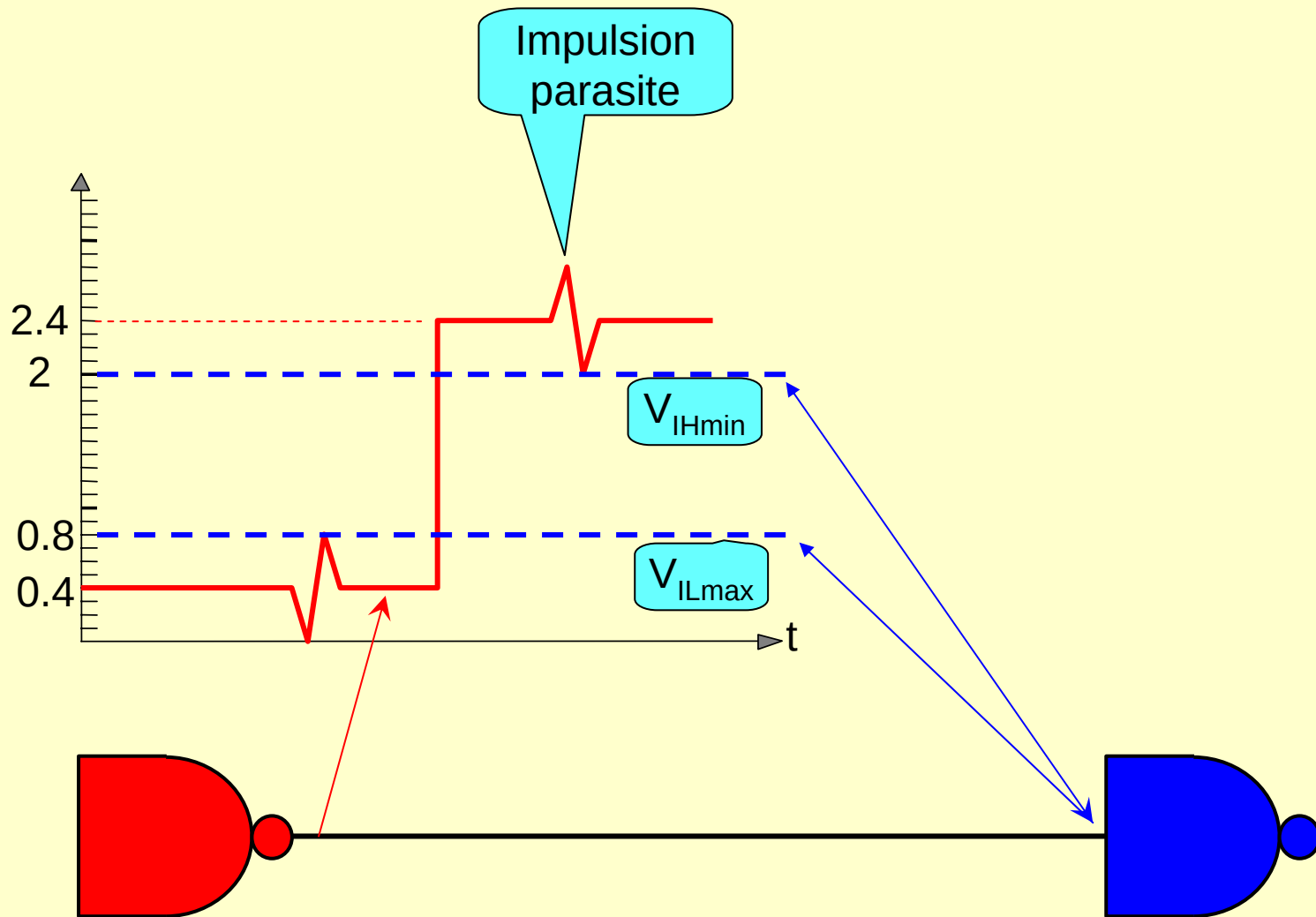
Immunité au bruit

Les signaux issus de la porte A doivent être interprétés correctement par la porte B

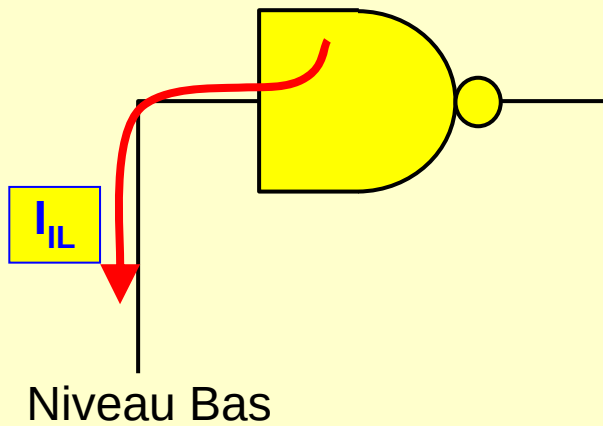


Immunité au bruit : $\Delta V_N = 0.4 \text{ V}$

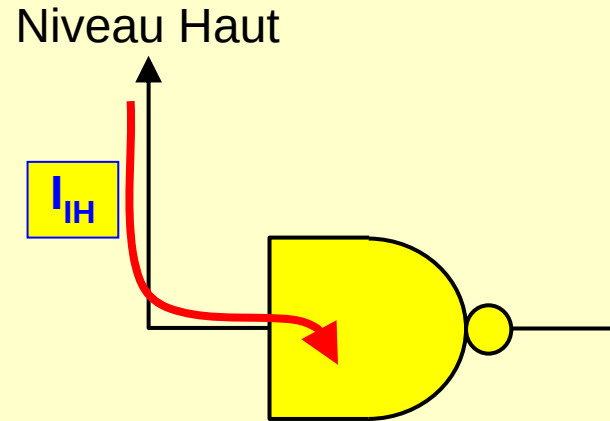
Autre façon de Voir l'immunité au bruit



Les courants d'entrée



$$I_{ILmax} = 1.6 \text{ mA}$$

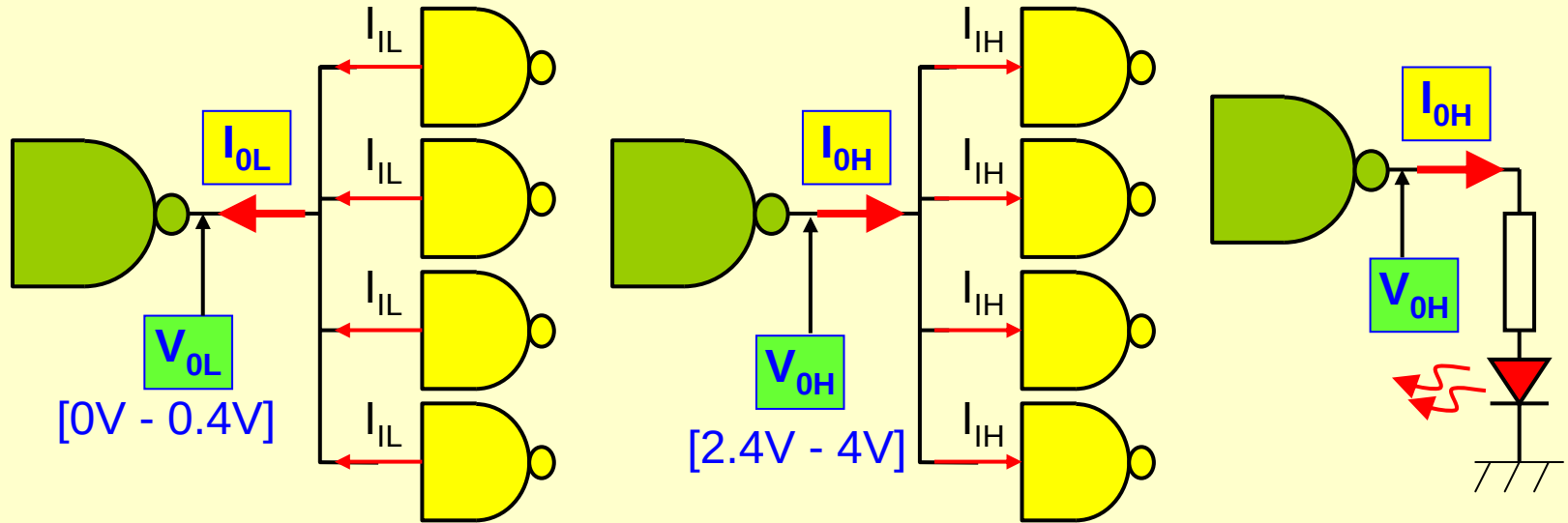


$$I_{IHmax} = 40 \text{ }\mu\text{A}$$

Les courants de sortie

Le courant de sortie dépend de la **charge** branchée à la sortie :

- Une ou plusieurs portes
- Charge résistive ou autre



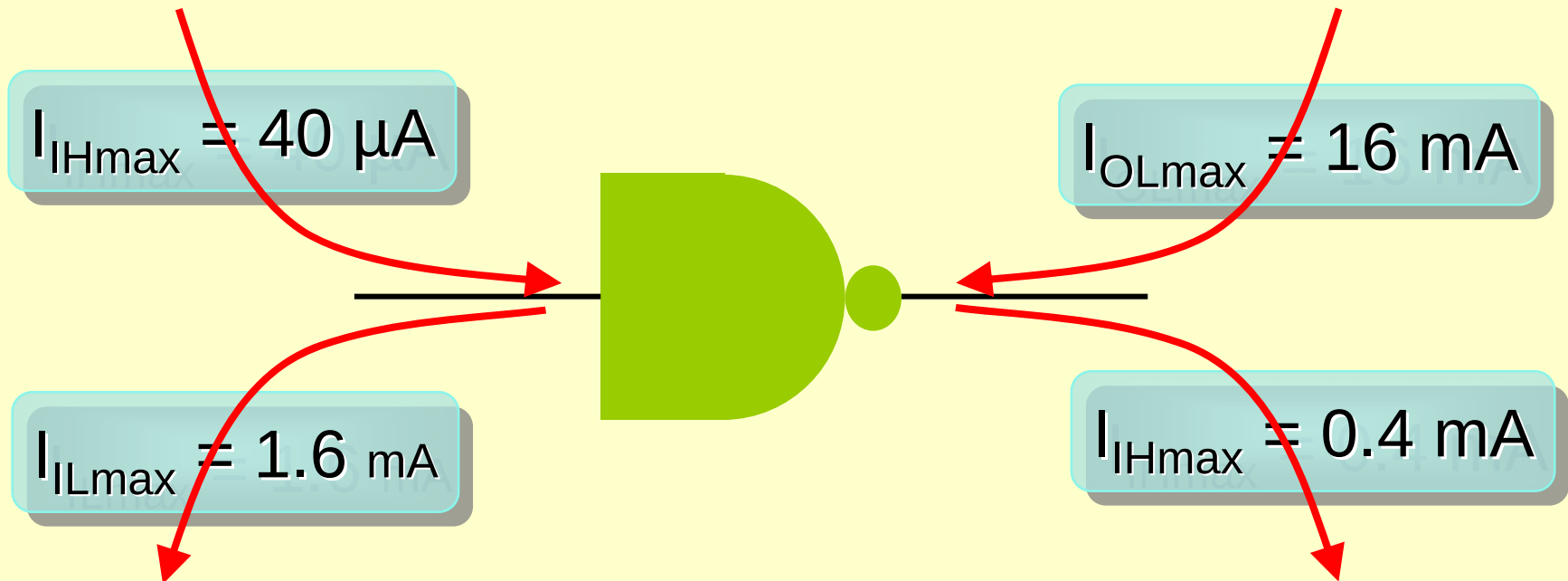
Si I_{OL} augmente
Alors V_{OL} augmente
Risque de dépasser 0.4 V
Ne sera plus interprétée
Comme niveau bas

Si I_{OH} augmente
Alors V_{OL} diminue
Risque de passer en dessous de 2.4 V
Ne sera plus interprétée
Comme niveau haut

Les constructeurs garantissent que dans le cas le plus défavorable :

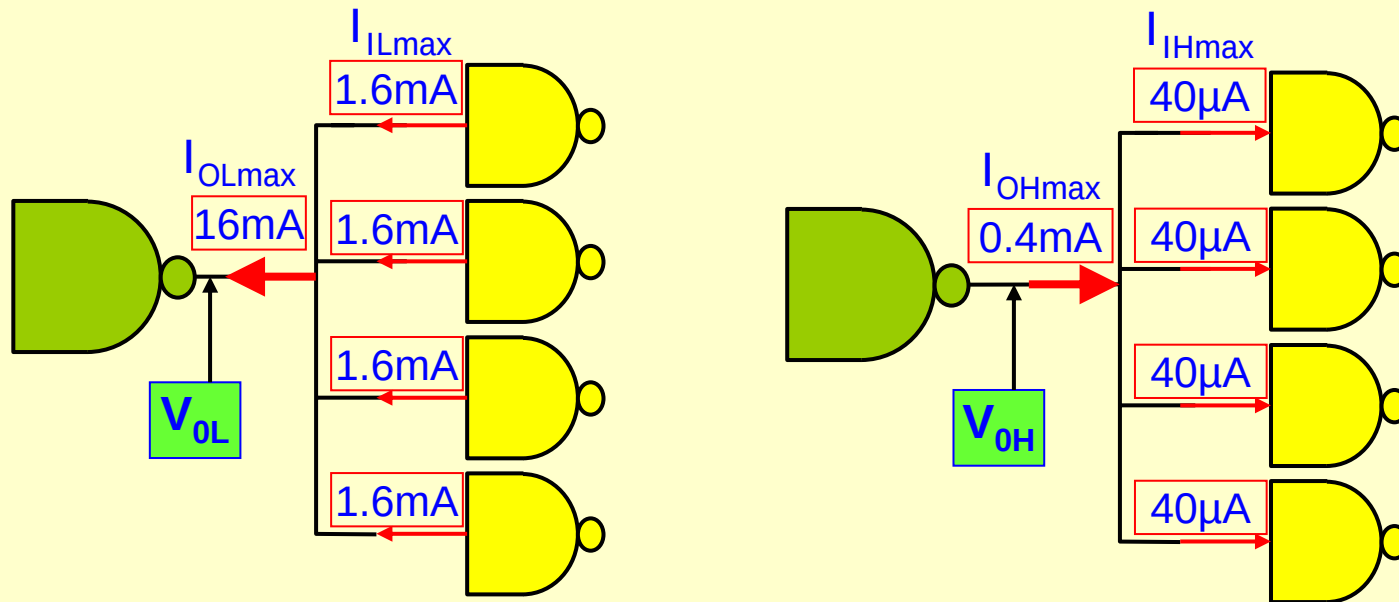
- **Niveau Bas** : la tension de sortie V_{OL} reste **inférieure** à $V_{OLmax} = 0.4 \text{ V}$ tant que le courant de sortie I_{OL} (injecté dans la porte) reste inférieur à **16 mA**
- **Niveau Haut** : la tension de sortie V_{OH} reste supérieure à $V_{OHmin} = 2.4 \text{ V}$ tant que le courant de sortie I_{OH} (fourni par la porte) reste inférieur à **0.4 mA**

Les courants de la famille TTL



Sortance

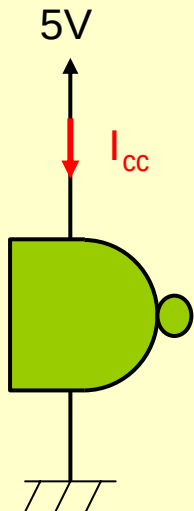
La **SORTANCE** représente le nombre max de portes que l'on peut brancher à la sortie d'une autre porte sans dégrader les niveaux logiques



$$\text{Sortance} = \frac{I_{OLmax}}{I_{ILmax}} = \frac{I_{OHmax}}{I_{IHmax}} = 10$$

Puissance Dissipée

La porte ne consomme pas le même courant quand sa sortie est au niveau haut et au niveau bas



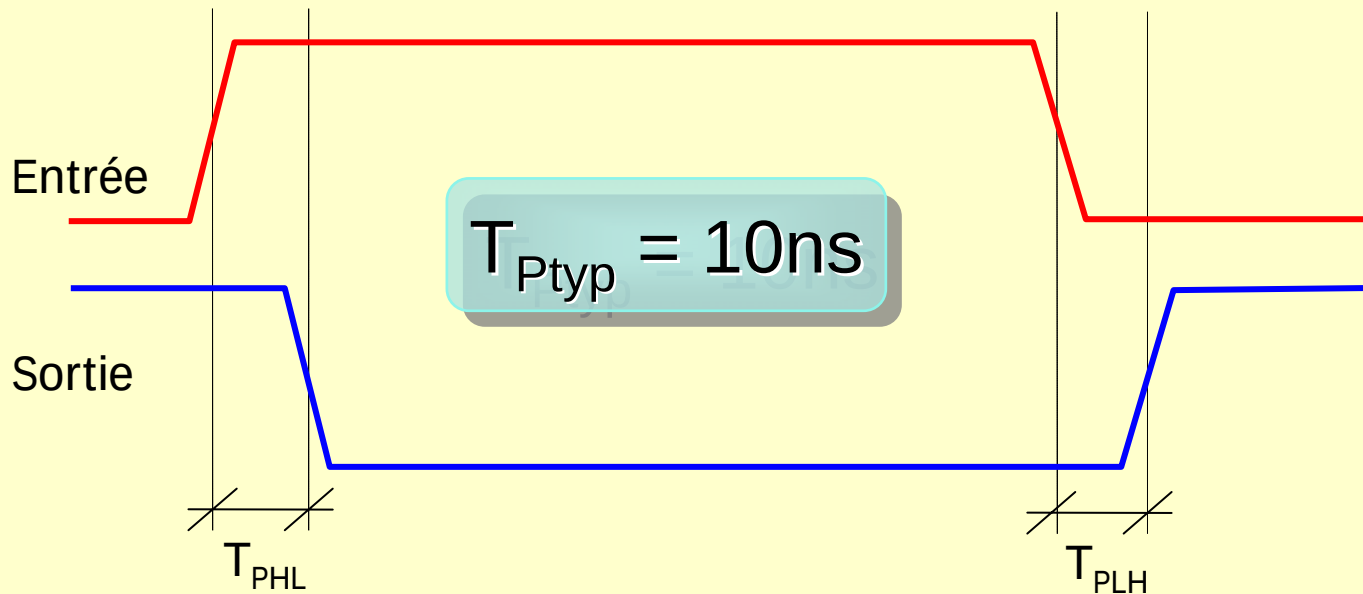
$$I_{CCHtyp} = 1mA$$

$$I_{CCLtyp} = 3mA$$

$$Pd_{typ} = \frac{1mA + 3mA}{2} \times 5V = 10 mW$$

La consommation augmente légèrement avec la fréquence. On peut considérer la valeur de 10 mW valable jusqu'à une fréquence de 1 MHz

Comportement dynamique



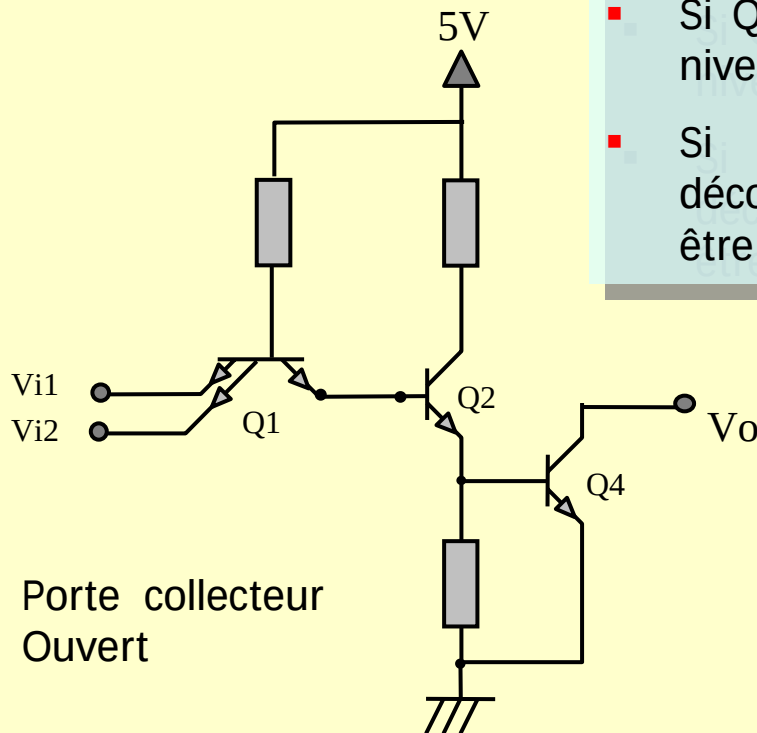
$$T_{PHLtyp} = 8ns$$

$$T_{PLHtyp} = 12ns$$

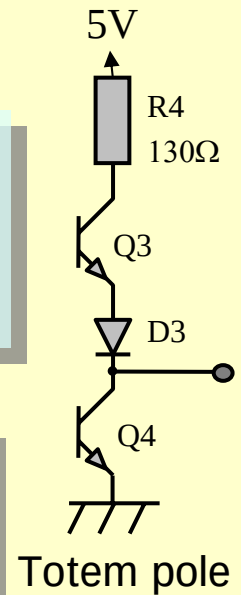
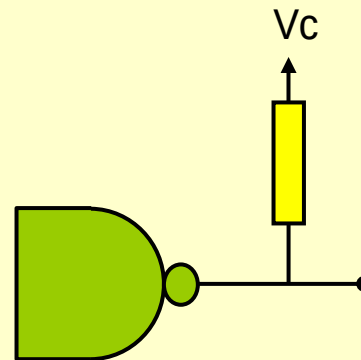
$$F_{max} = \frac{1}{T_{PLH} + T_{PHL}} = \frac{1}{20ns} = 50MHz$$

Porte collecteur Ouvert

La majorité des circuits TTL ont des sorties **totem-pole**
Certains circuits ont des sorties **Collecteur Ouvert**
La partie haute du totem-pole a été supprimée

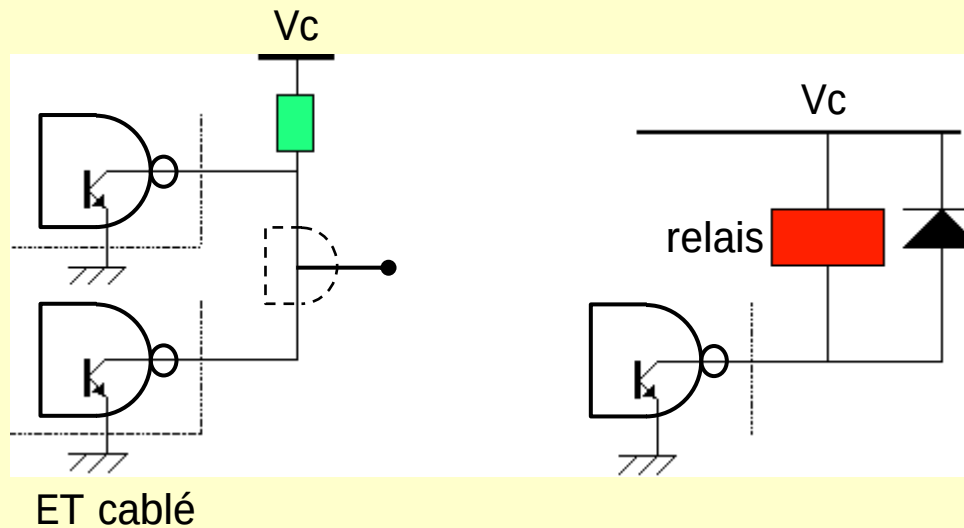


- Si Q4 est conducteur, la sortie est au niveau bas
- Si Q4 est bloqué, la sortie est déconnectée. Le niveau haut, peut être obtenu par une résistante externe



Avantage des portes collecteur ouvert

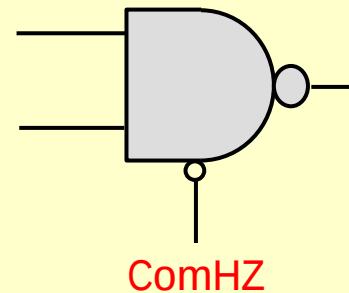
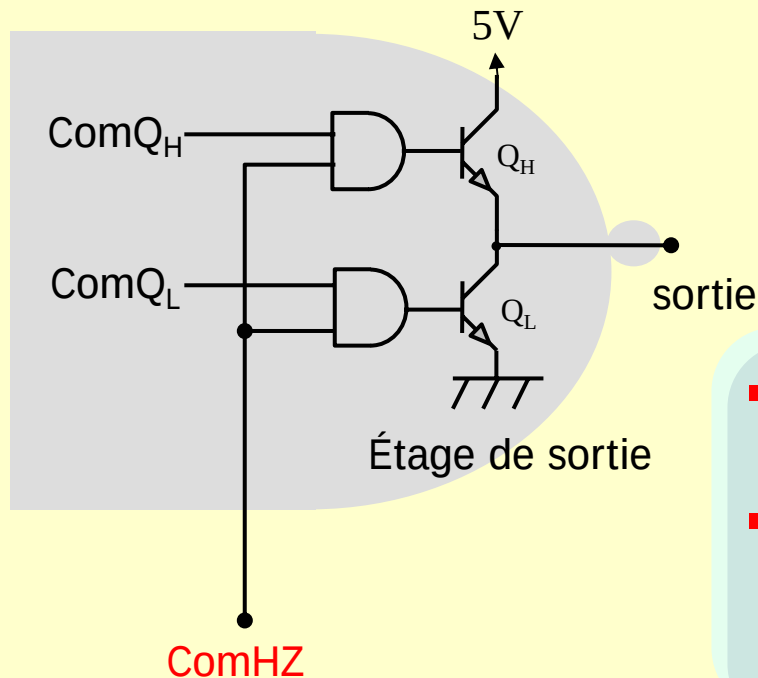
- La tension V_c peut aller jusqu'à 30v et le courant absorbé par la porte peut aller jusqu'à 40 mA. Cela permet par exemple de commander un Relais directement
- On peut connecter deux sorties pour faire un ET câblé : Pour que la sortie soit H, il faut que les deux sorties soient H



Porte à Sortie 3 états (Tristate)

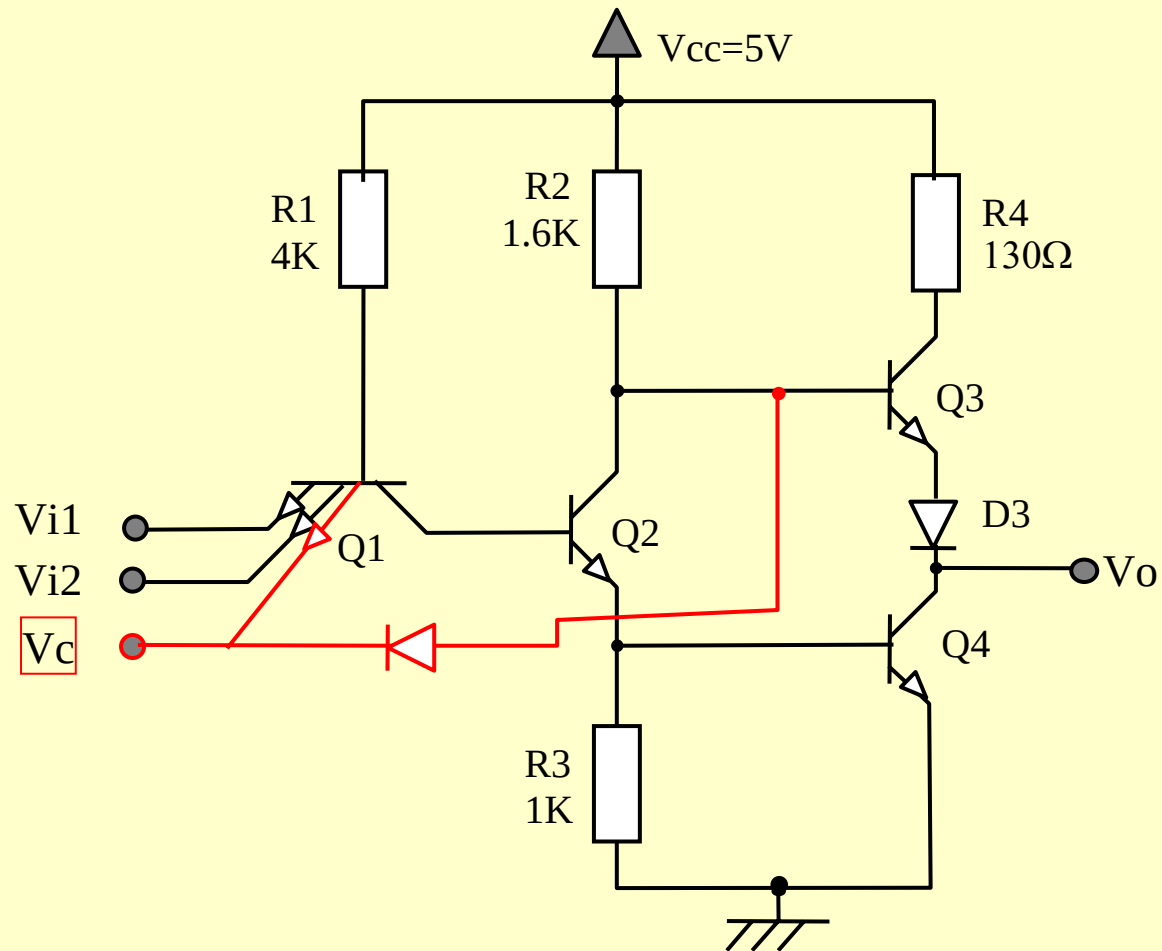
3 niveaux logiques :

- Niveau Bas (L)
- Niveau Haut (H)
- Niveau Haute Impédance (HZ)

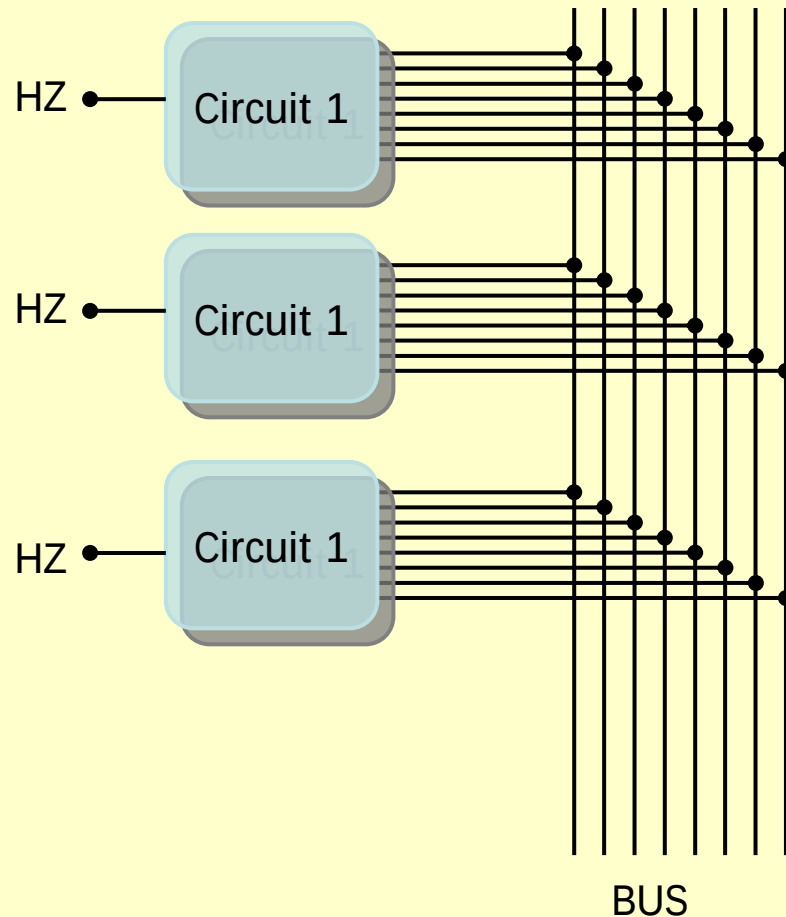


- $ComHZ = 1$: fonctionnement normal : sortie 2 états
- $ComHZ = 0$: 2 transistors bloqués, sortie déconnectée = haute impédance

TTL : Sorties 3 états (2)

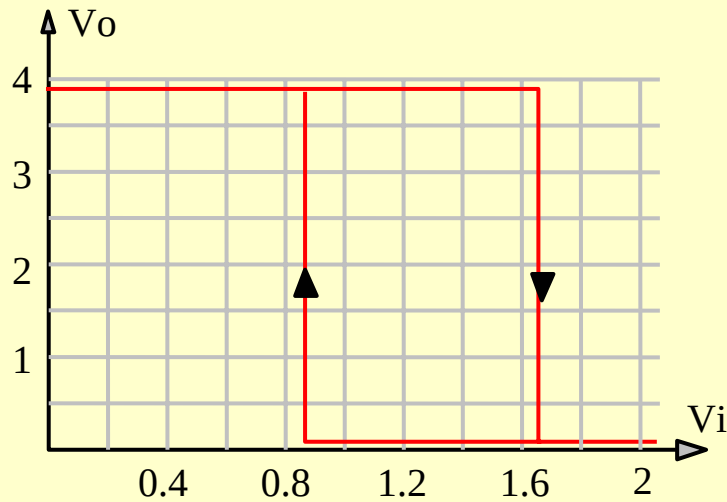
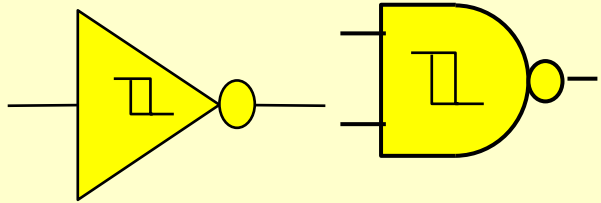


Utilisation des circuits tristate



Seul le circuit qui est sélectionné pour écrire dans le bus doit être en basse impédance, tous les autres doivent être déconnectés c.a.d en haute impédance

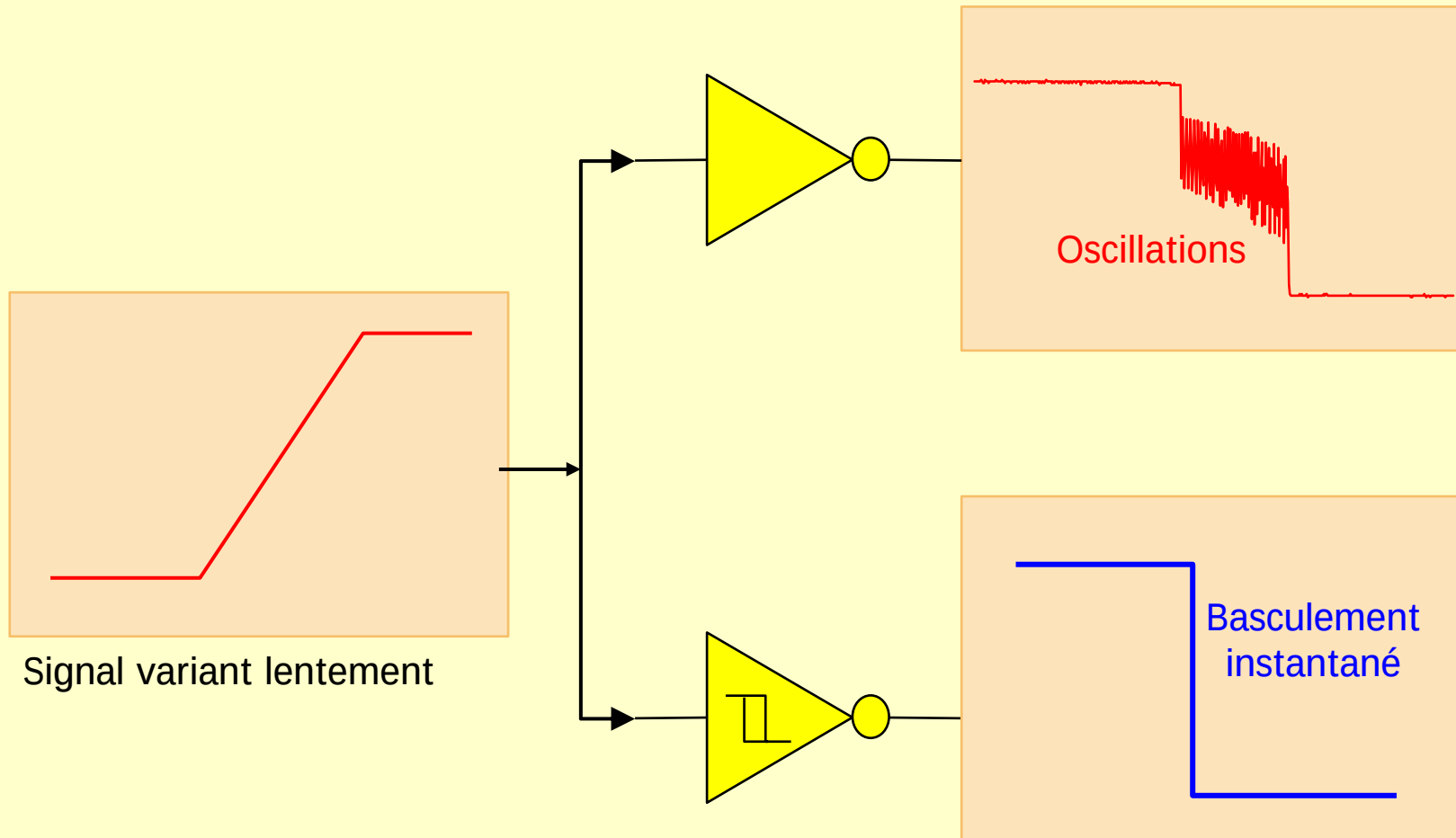
Porte avec entrée Trigger de Schmitt



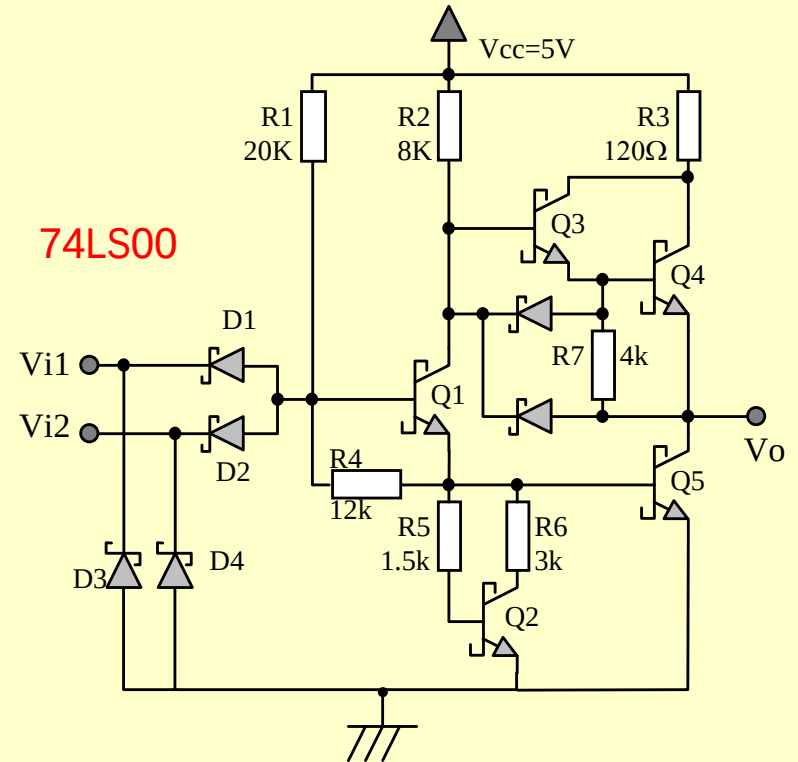
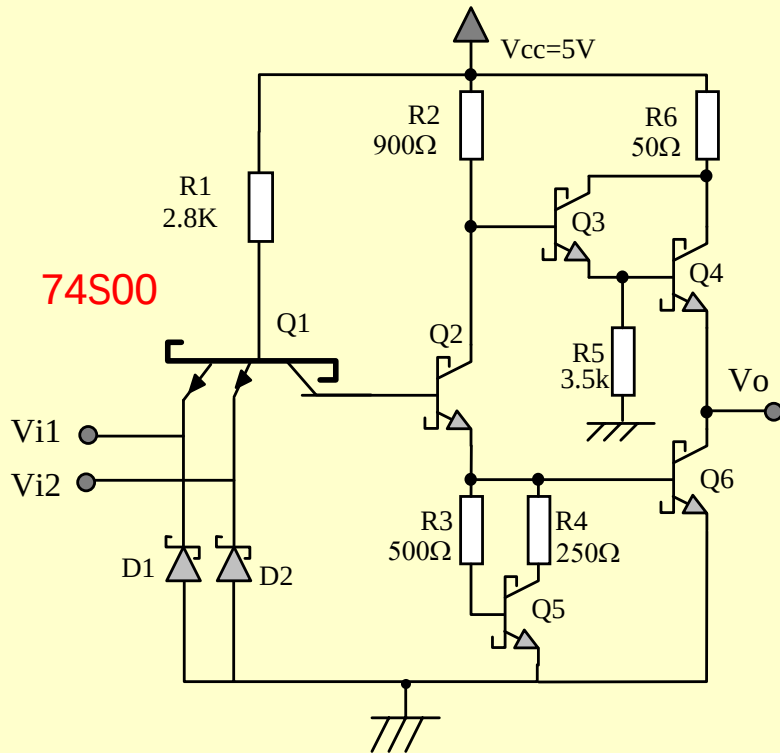
La caractéristique de transfert comporte un hystérésis. Le seuil de basculement H-L n'est pas le même que le seuil de basculement L-H

- Mise en forme des signaux,
- retardateur d'impulsions,
- élargisseur d'impulsions,
- oscillateurs

Mise en forme des signaux



TTL Schottky et Low Power Schottky



Ces technologies utilisent des diodes et des transistors Schottky qui ont la particularité de ne pas se saturer et de commuter plus rapidement. En baissant la valeur des résistances sur la 74S, on accélère la commutation mais on augmente la consommation.

Quelques courbes

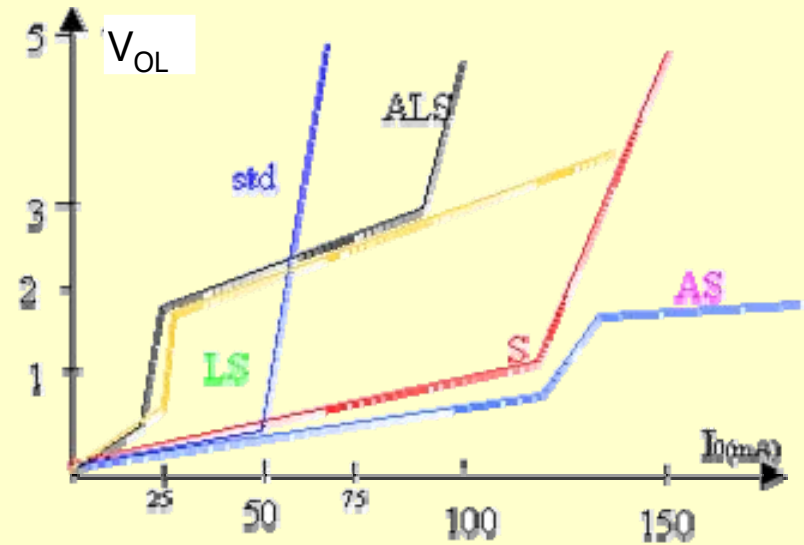
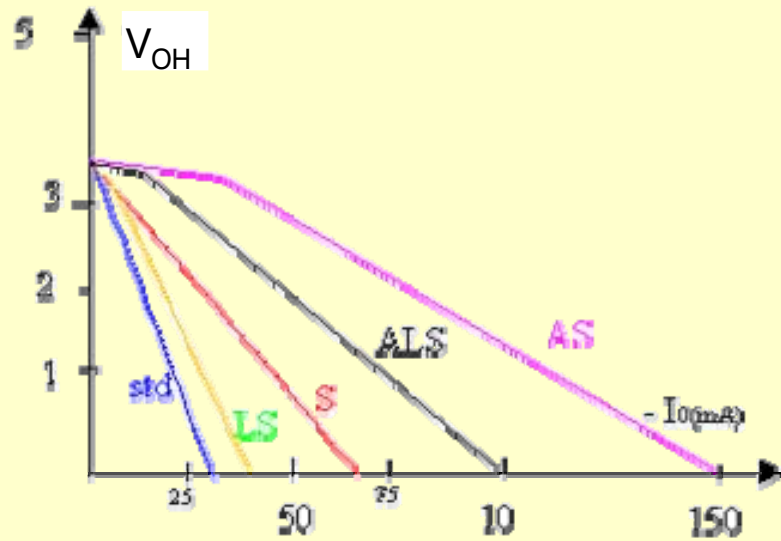
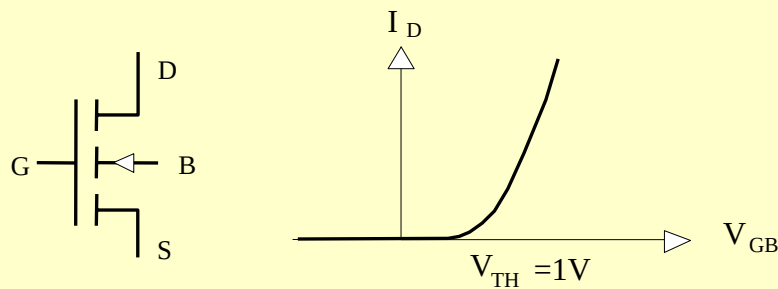


Tableau comparatif

	74	74S	74LS	74AS	74ALS
V_{ILmax}	0.8	0.8	0.8	0.8	0.8
V_{IHmin}	2	2	2	2	2
V_{OLmax}	0.4	0.5	0.5	0.5	0.5
V_{OHmin}	2.4	2.7	2.7	2.7	2.5
T_p (ns)	10	3	9.5	1.5	4
P_d (mW)	10	19	2	8.5	1
I_{ILmax} (mA)	1.6	2	0.4	0.5	0.2
I_{IHmax} (μA)	40	50	20	20	20
I_{OLmax} (mA)	16	20	8	8	8
I_{OHmax} (mA)	0.4	1	0.4	2	0.4
F_{max} (Mhz)	35	125	40	200	70

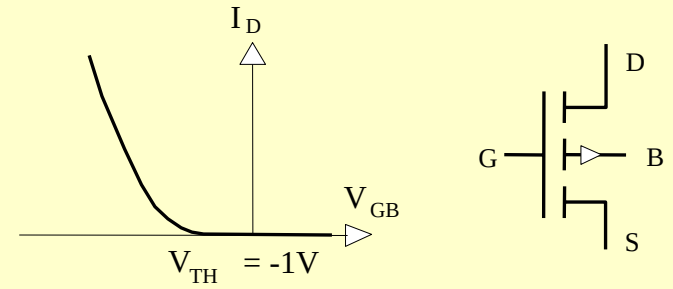
Famille CMOS (MOS Complémentaire)

- Fabriquée avec des MOS à enrichissement canal N et canal P
- Consommation quasi nulle en basse fréquence
- A l'origine très lente (série 4000), aujourd'hui, on sait fabriquer des circuit CMOS rapide
- Souplesse d'alimentation $3V < V_{dd}-V_{ss} < 18$
- Meilleure immunité au bruit



$V_{GS} < V_{TH} \rightarrow$ **OFF , $R = 10^{10} \Omega$**

$V_{GS} \gg V_{TH} \rightarrow$ **ON , $R = qq\ 100^{enes} \Omega$**

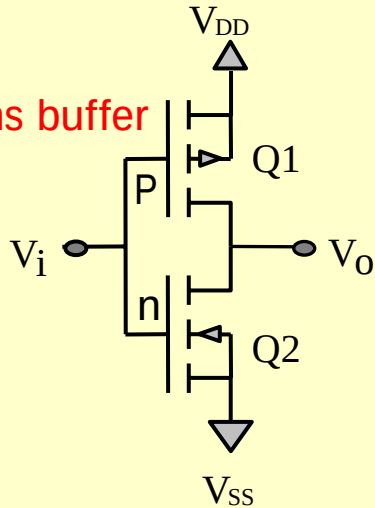


$V_{GS} > V_{TH} \rightarrow$ **OFF , $R = 10^{10} \Omega$**

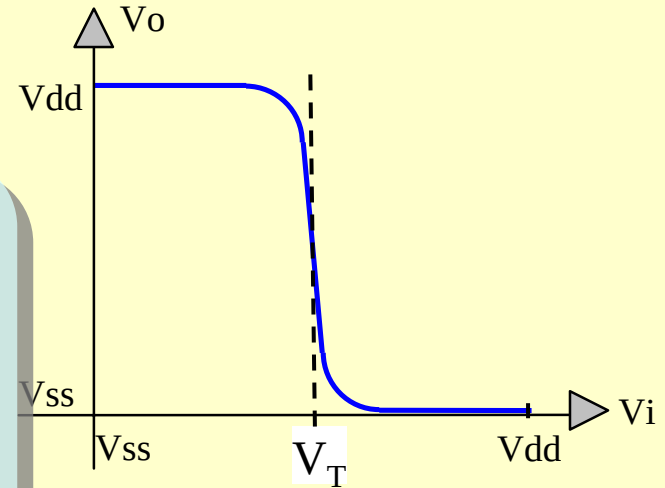
$V_{GS} \ll V_{TH} \rightarrow$ **ON , $R = qq\ 100^{enes} \Omega$**

Structure de la porte élémentaire

Sans buffer

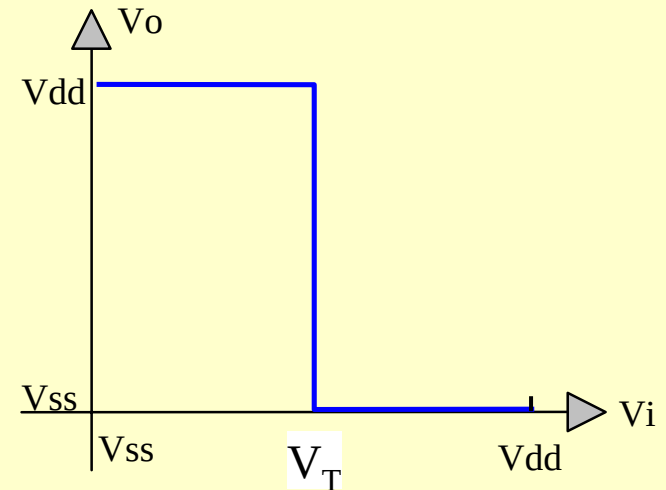
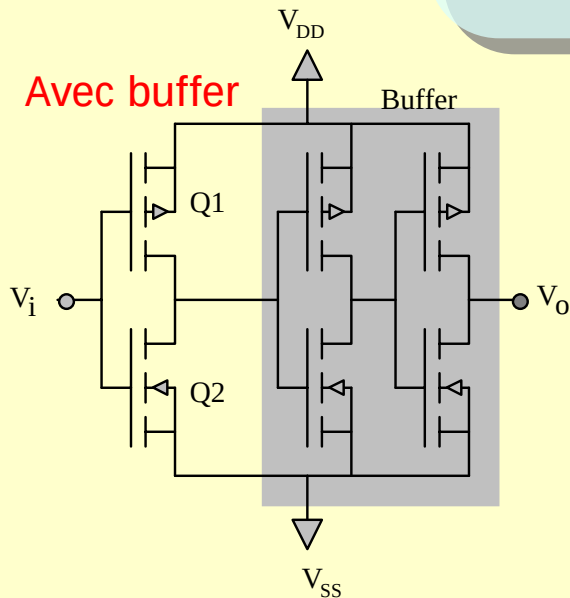


- $V_{SS} = 0$ pour simplifier
- $V_i = 0$, $V_{GS1} = -V_{dd}$, $V_{GS2} = 0$,
Q1:ON, Q2:OFF, $V_o = V_{dd}$
- $V_i = V_{dd}$, $V_{GS1} = 0$, $V_{GS2} = V_{dd}$,
Q1:OFF, Q2:ON, $V_o = 0$

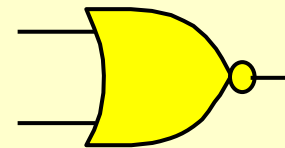
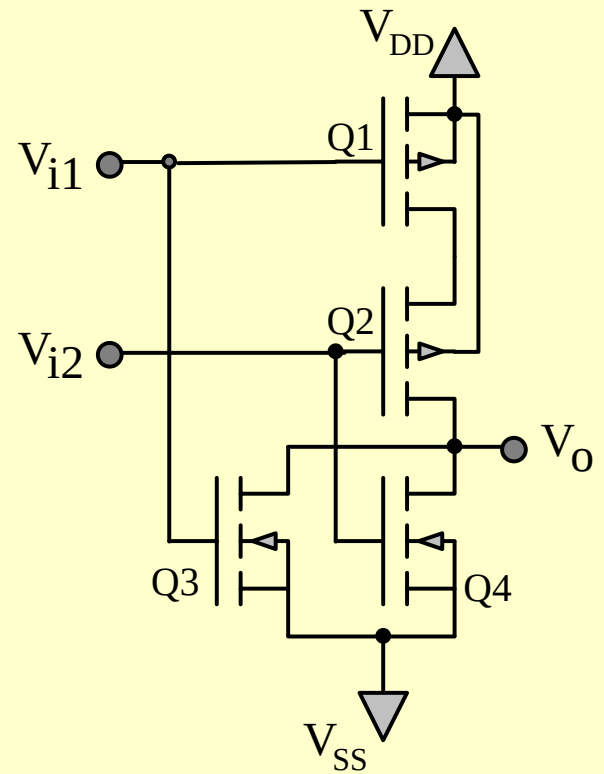
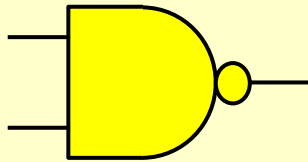
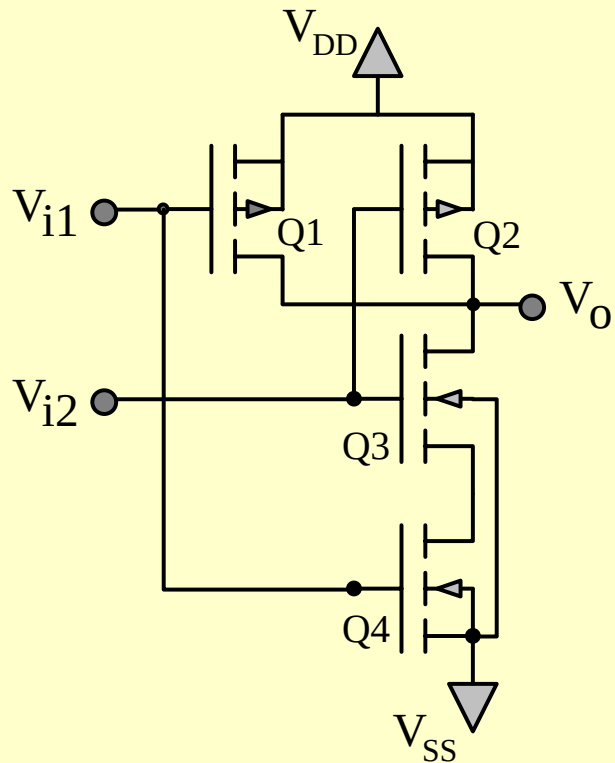


$$V_T = \frac{V_{DD} + V_{SS}}{2}$$

Avec buffer

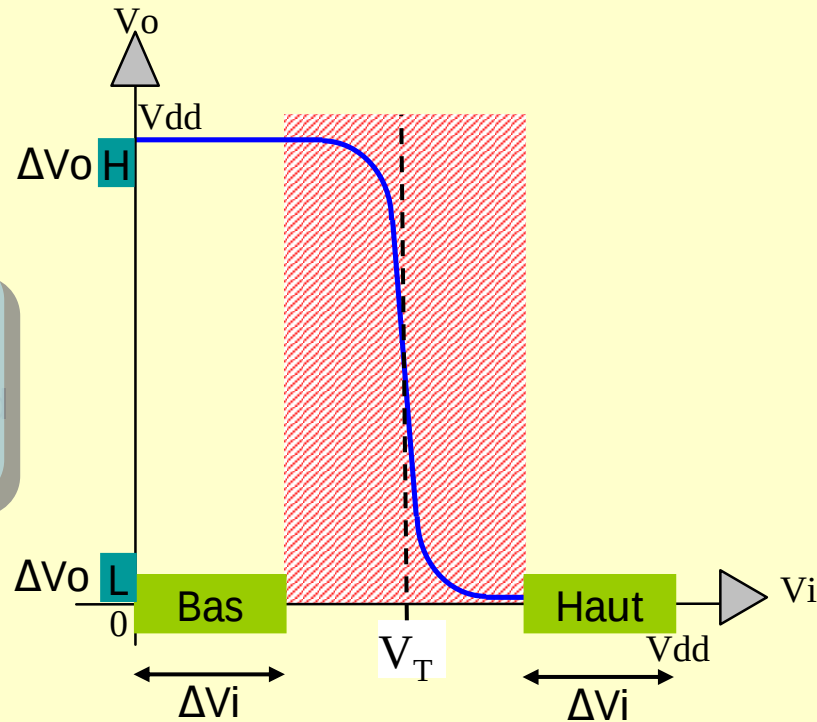


Porte NAND et NOR



Niveaux logiques

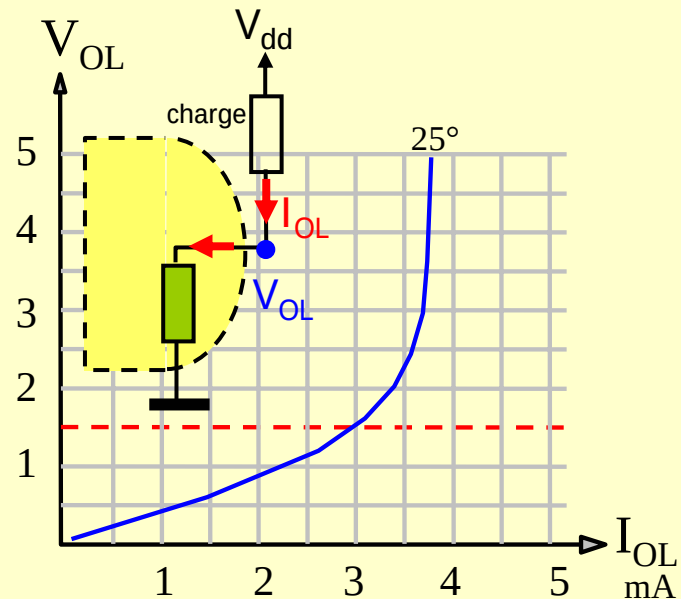
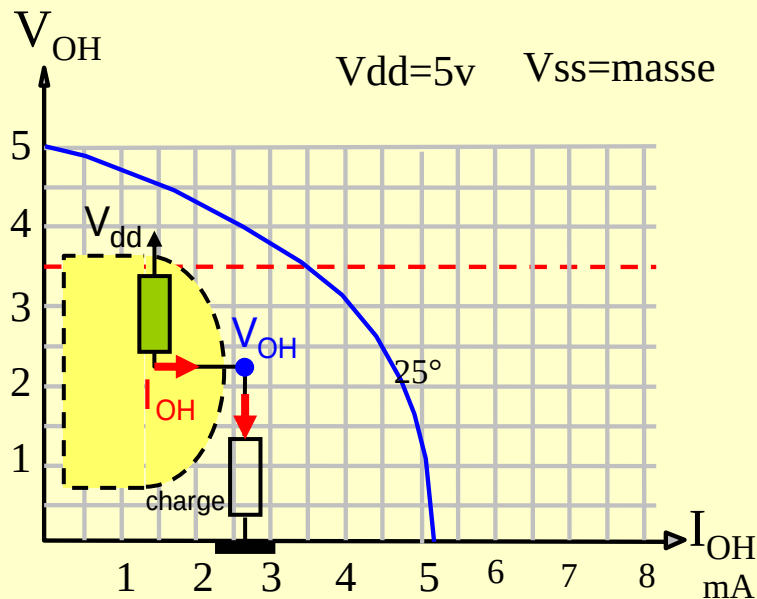
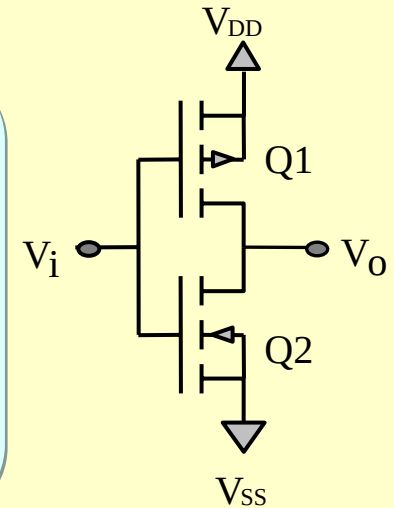
- $I_o < 1\mu A$: $\Delta V_o = 0.05V$
- $I_o < 0.5mA$: $\Delta V_o = 10\%V_{dd}$
- $I_o > 0.5mA$: voir courbes



- Sans buffer : $\Delta V_i = 20\% V_{DD}$
- Avec buffer : $\Delta V_i = 30\% V_{DD}$

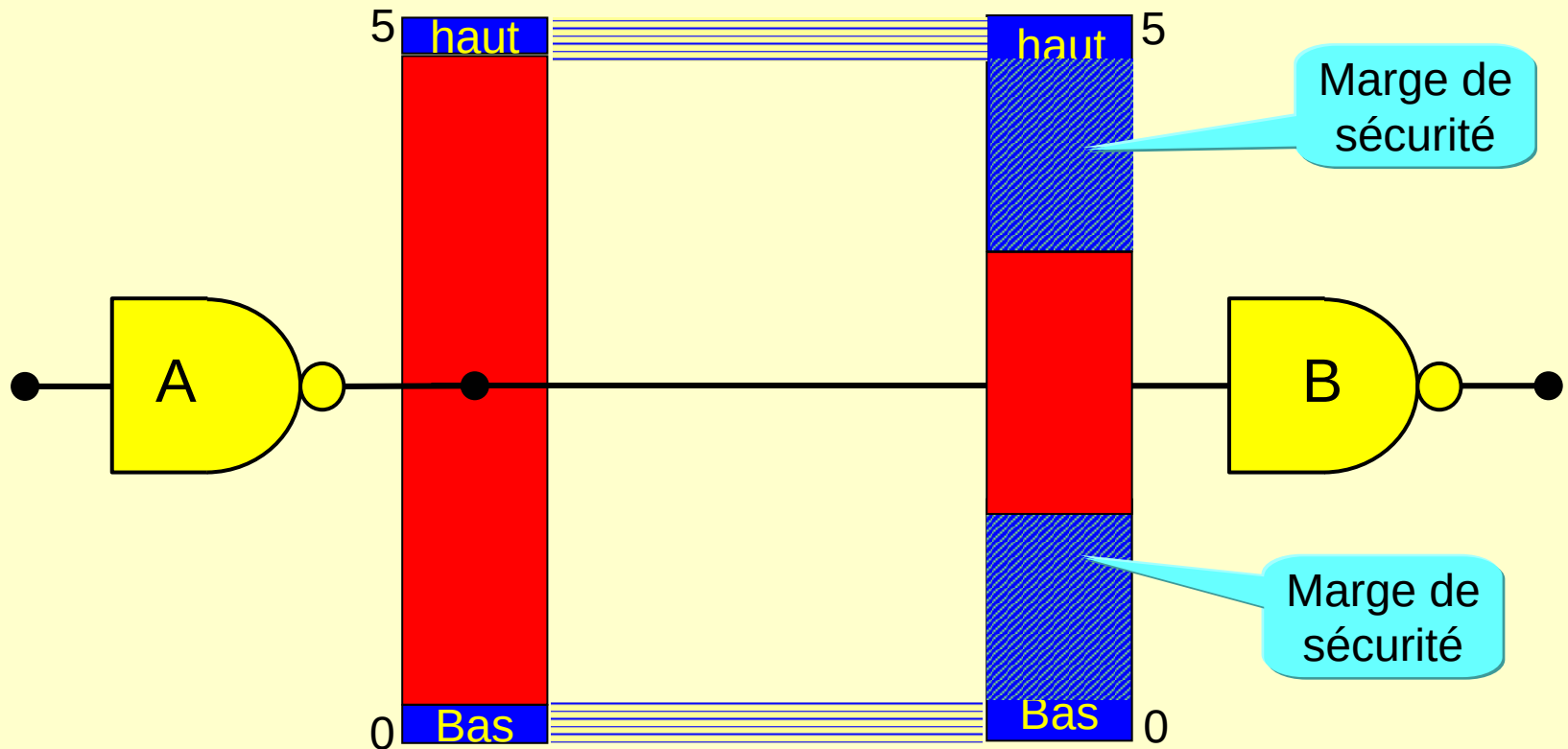
Courants de la famille CMOS

- Le courant d'entrée est **nul** car on attaque sur des grilles isolées
- Le courant de sortie circule à travers la résistance $R_{\text{DS(on)}}$ du MOS conducteur.
- La chute de tension dans cette résistance fait que V_o dépend fortement de I_o



Immunité au bruit

Les signaux issus de la porte A doivent être interprétés correctement par la porte B



Immunité au bruit : $\Delta V_N = \Delta V_i - \Delta V_o$

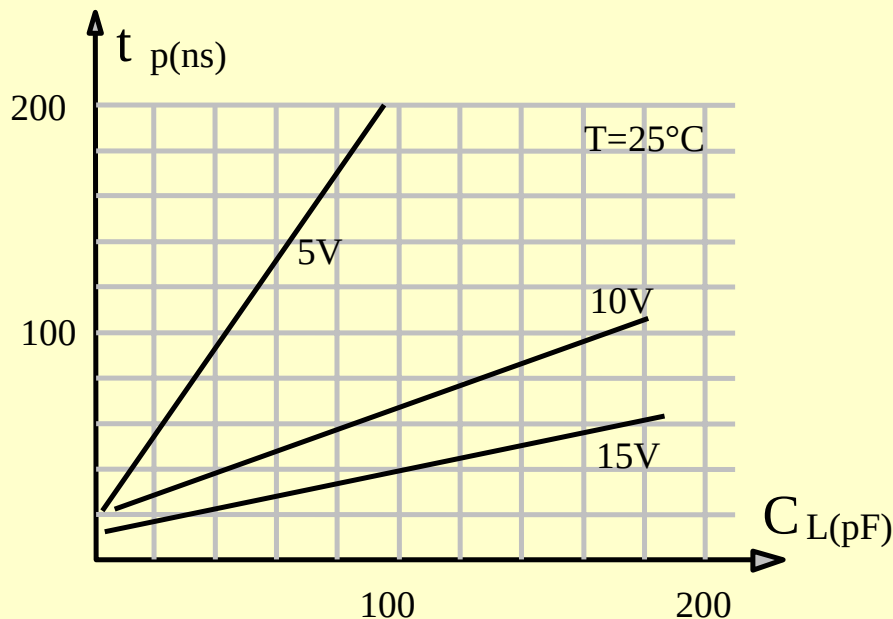
On peut augmenter l'immunité en augmentant l'alimentation

Sortance

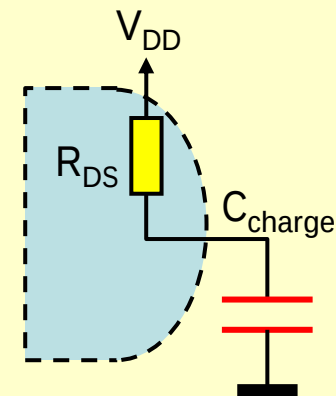
- La sortance n'est pas limitée par les courants puisque le courant d'entrée des portes CMOS est nul.
- Mais chaque porte branchée ajoute sa capacité (7.5 pF) qui augmente la capacité de charge ce qui détériore le temps de propagation
- Dans la pratique, on évite de dépasser une sortance de 50

Comportement dynamique

- Le temps de propagation dépend du circuit RC constitué de R_{DS} et de la capacité de charge
- A chaque basculement, il faut charger ou décharger C à travers la résistance R_{DS} . ($\tau = R_{DS} C$)
- Il faut Diminuer C et augmenter V_{DD} (pour diminuer R_{DS})

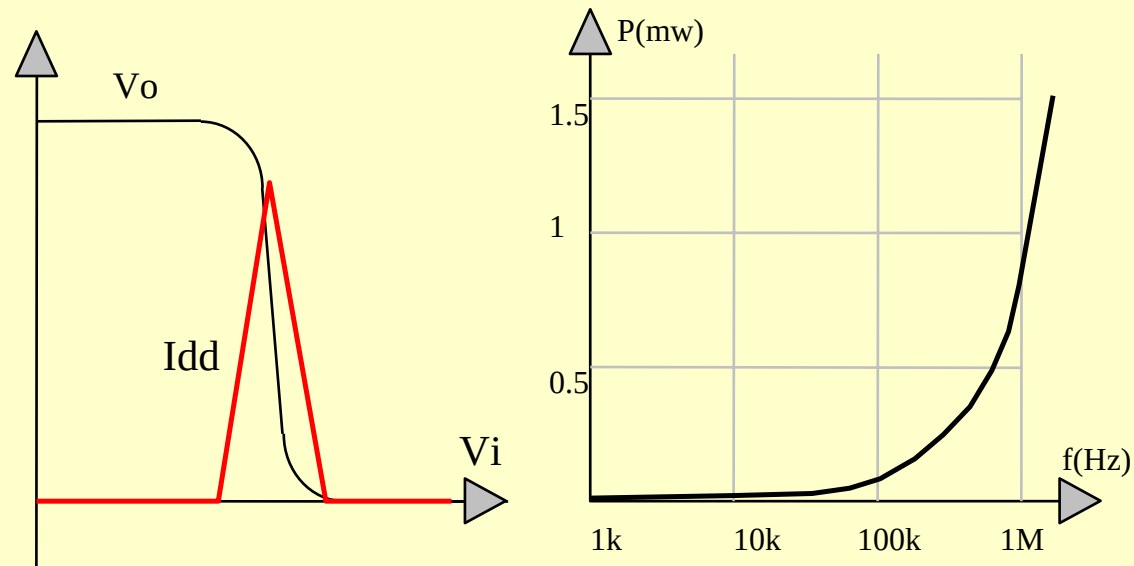


$$R_{DS} = \frac{1}{k(V_{GS} - V_{th})}$$



Consommation

- En **statique** la porte ne **consomme rien** un des deux transistors est bloqué
- Quand $V_i = V_T$ (basculement), les deux transistors conduisent simultanément et un courant circule entre V_{dd} et la masse
- En **haute fréquence**, on a beaucoup de basculement, donc la **consommation augmente**



Quelques Variantes de la famille CMOS

Serie 4000 : CMOS standard (l'ancêtre)

74C : série 4000 avec brochage TTL

74AC : advanced CMOS

74ACT : advanced CMOS compatible TTL

74HC : High-Speed CMOS Logic

74HCT : High-Speed CMOS Logic compatible TTL

74AHC : Advanced High-Speed CMOS Logic

74AHCT : Advances High-Speed CMOS Logic compatible TTL

74BCT : BiCMOS technology

74ABT : Advanced BiCMOS

74LV : Low Voltage HCMOS Technology

74LVC : Low Voltage CMOS

74ALVC : Advanced Low Voltage CMOS

74LVT : Low Voltage Technology

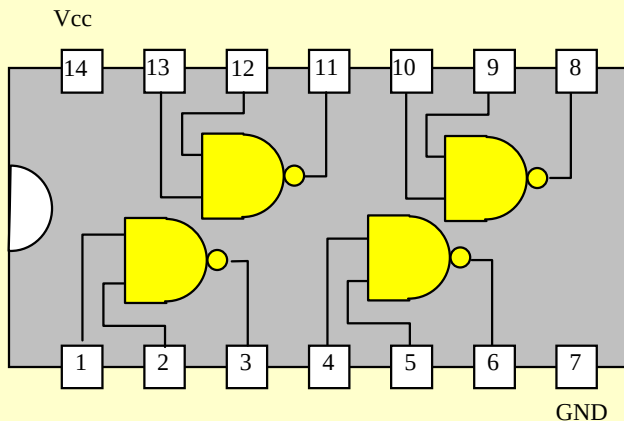
74ALVT : Advanced Low Voltage Technology

74ALB : Advanced Low voltage BiCMOS

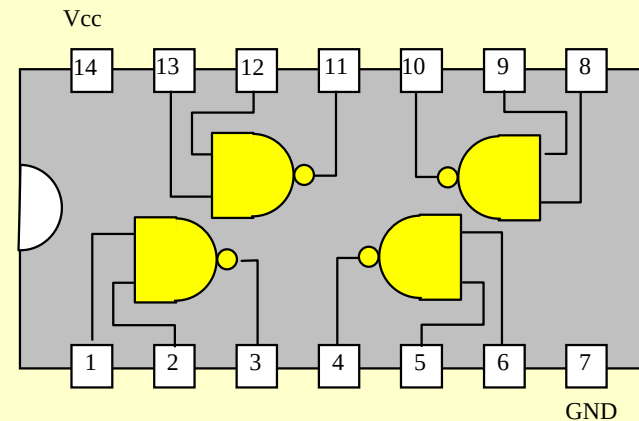
74CBTLV : Low Voltage Bus Switches (Crossbar technology)

Pinout de la famille 74C

Les circuits de la famille 74C sont les mêmes que les circuits de la série 4000 sauf qu'ils ont le même brochage que la famille TTL



7400
74C00
74xx00

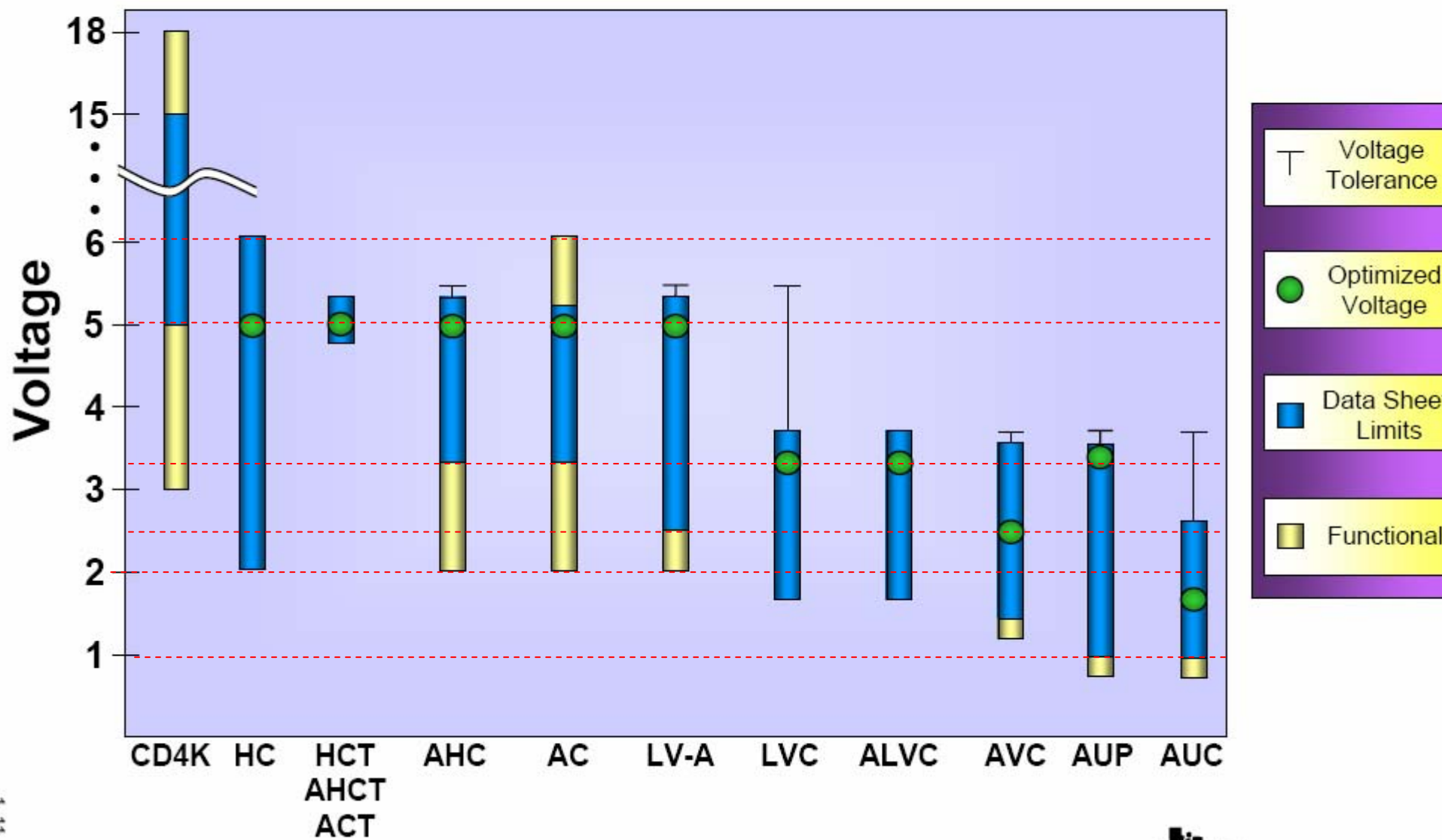


CD 4011

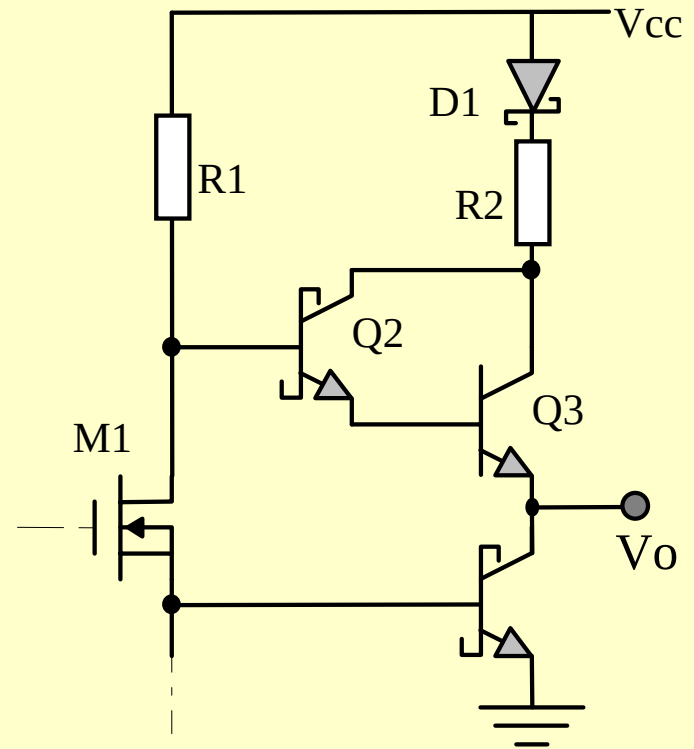
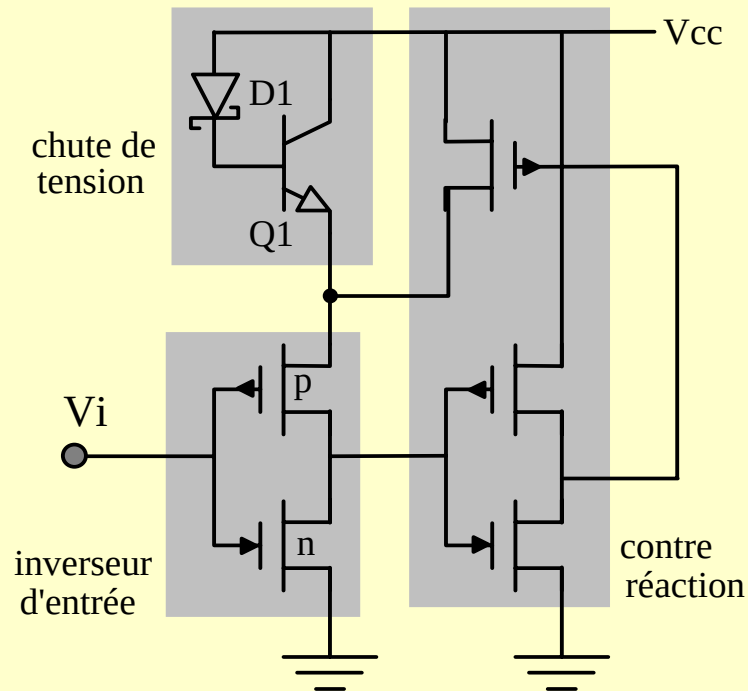
Les famille High Speed Cmos et Advanved CMOS HC, HCT, AHC, AHCT, AC, ACT

- Même structure que la famille CMOS
- Transistors à grille silicium
- Technologie de fabrication plus avancée 3, 2 et 1 μm
- Capacité de grille plus faible $\rightarrow$ rapidité
- Faible consommation comme la CMOS
- Plus rapide, similaire TTL
- Courant de sortie plus important $\rightarrow$ driving capability
- Alimentation max ramenée à 6V

CMOS Voltage Roadmap

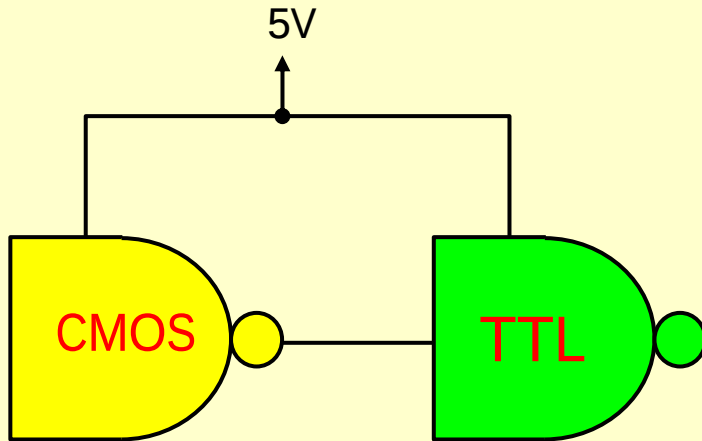


Famille BiCMOS : BCT, ABT ...

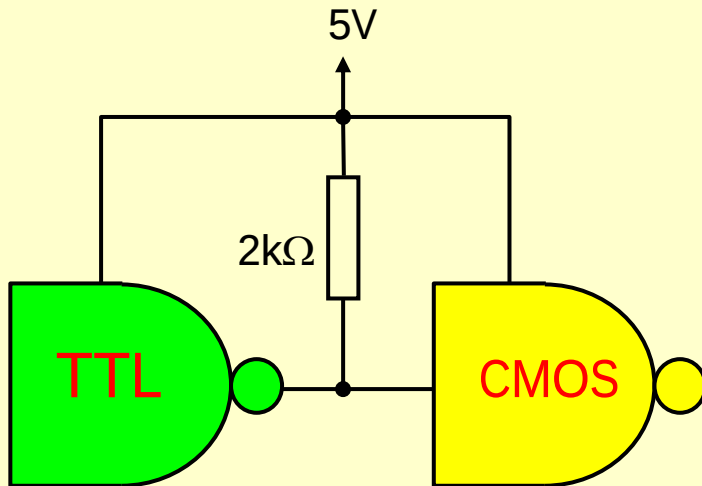


- Avantages de CMOS : Faible consommation, courant d'entrée nul
- Avantage de Bipolaire : Courant de sortie important (Driving capability)

Interface TTL-CMOS-TTL

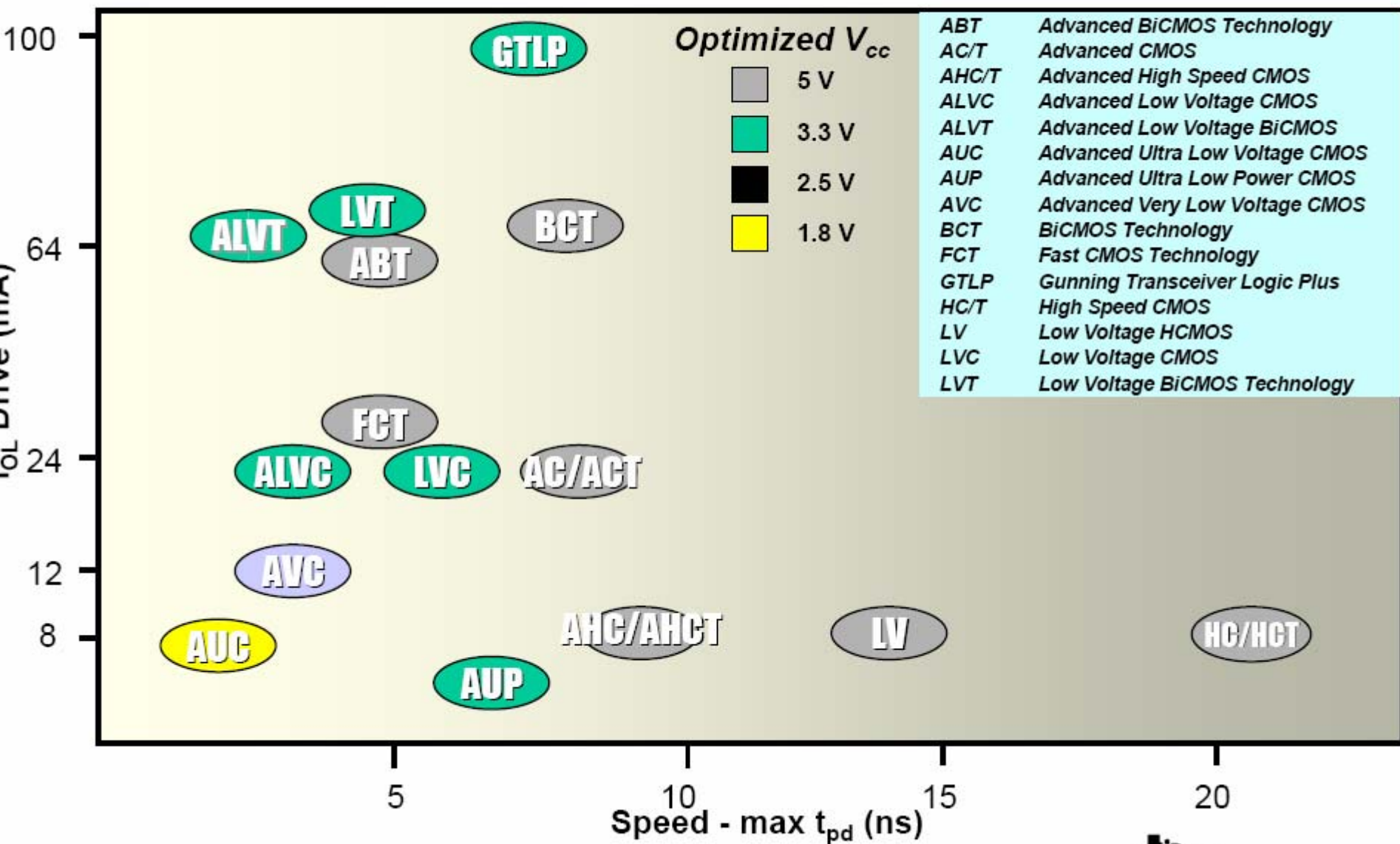


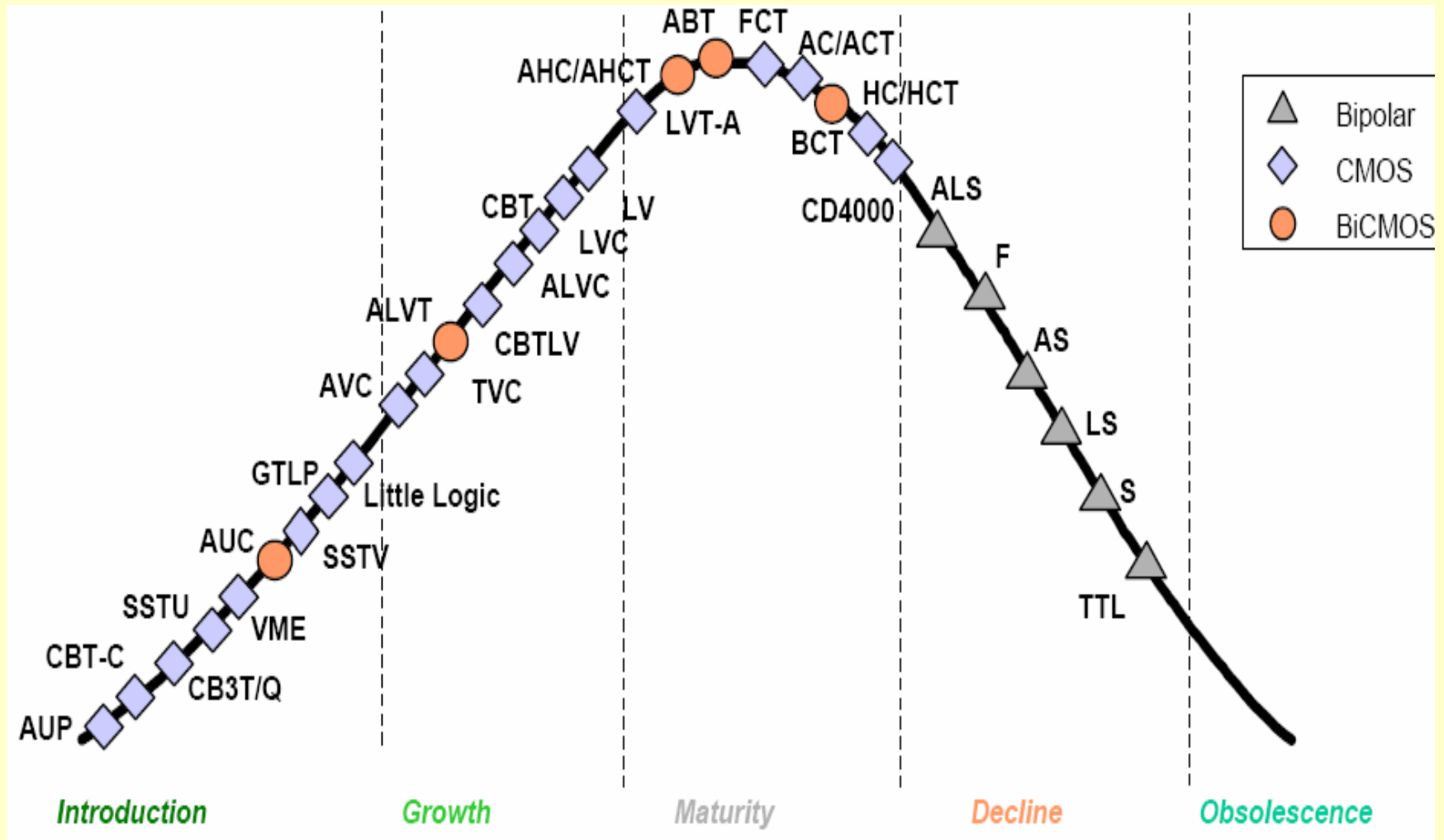
- CMOS $\rightarrow$ TTL
- Aucune interface n'est nécessaire



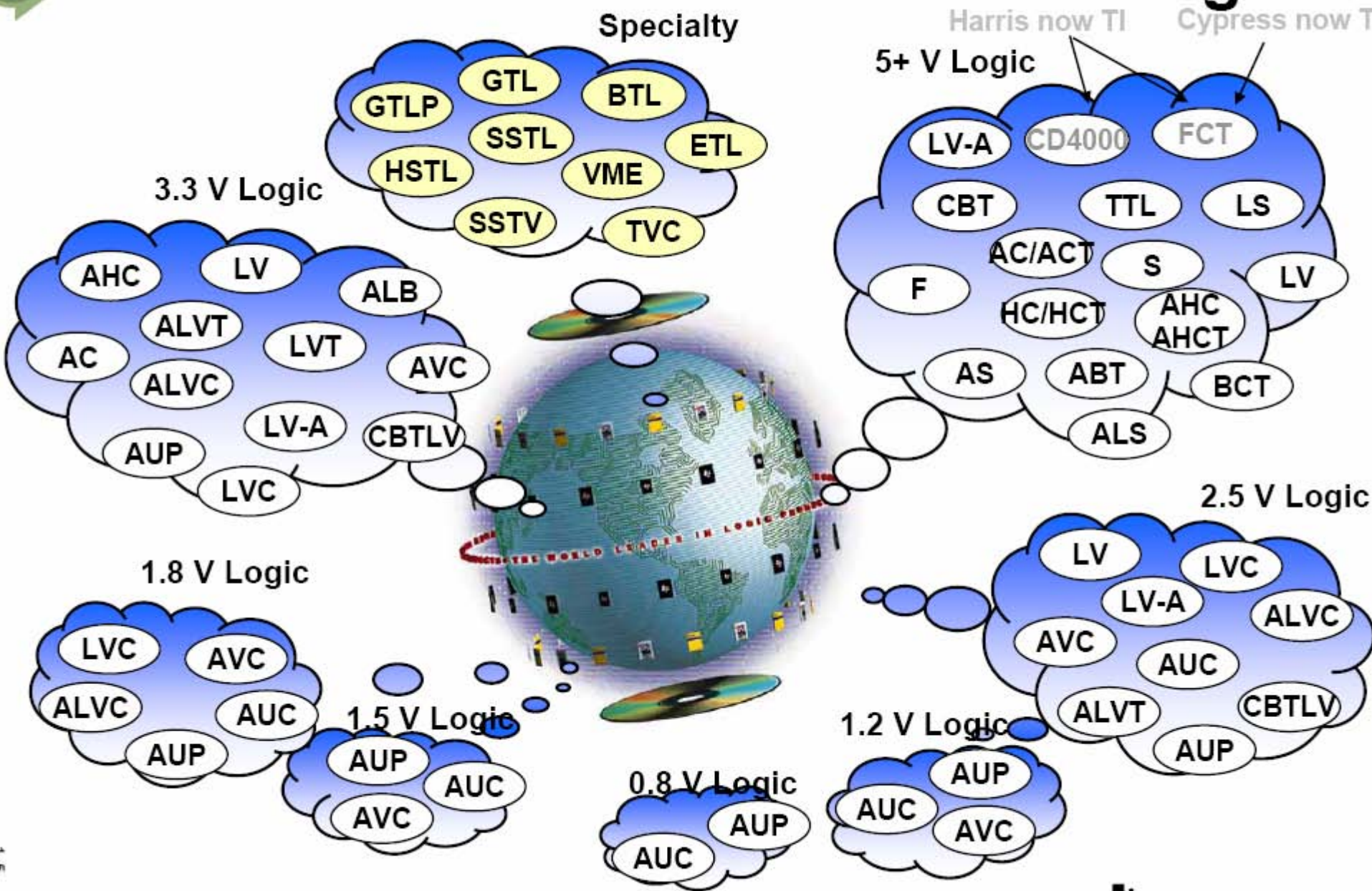
- TTL $\rightarrow$ CMOS
- V_{OH} garanti par TTL = 2.4V insuffisant pour CMOS
- La résistance de pull up garanti
 - Niveau haut > 4.6V
 - Niveau bas < 0.4V

Family Performance Positioning



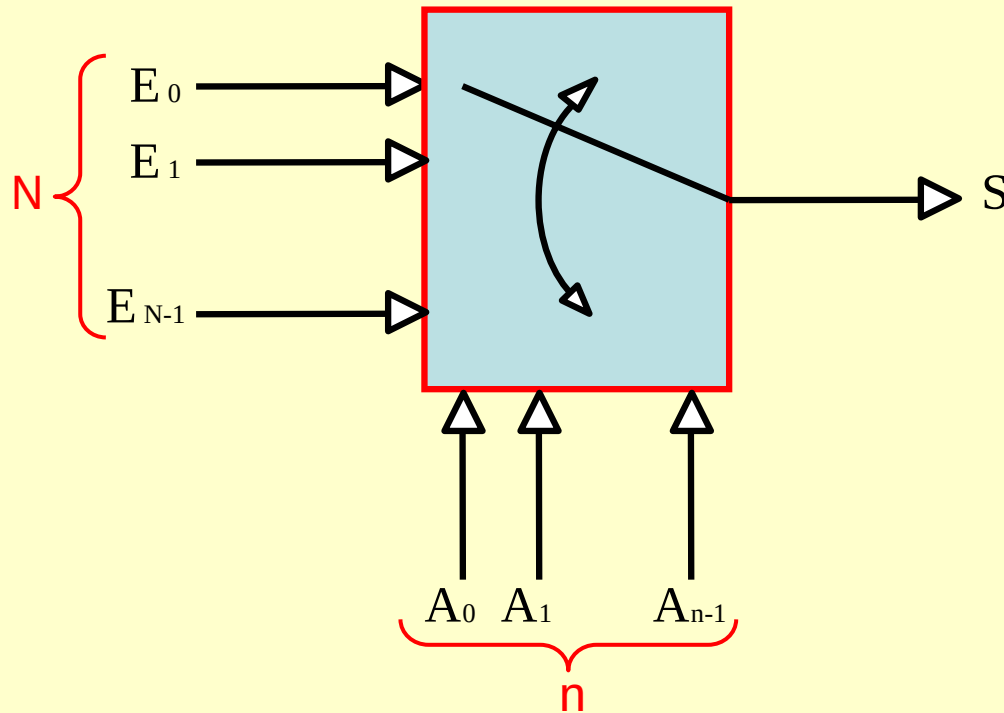


Welcome to the World of TI Logic



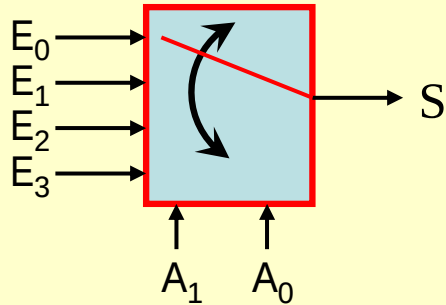
Les circuits Combinatoires Usuels

Les multiplexeurs

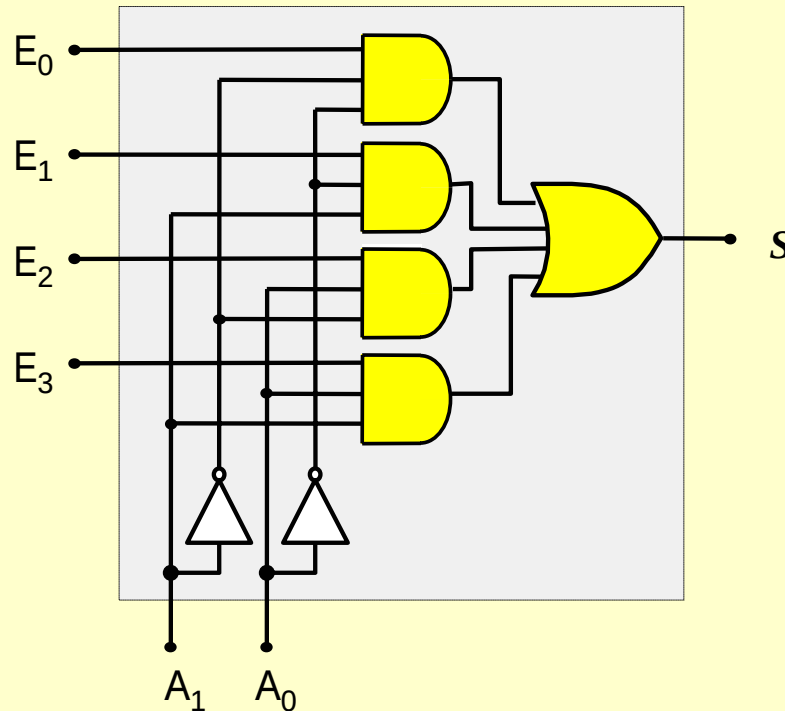


- Choisir une voie parmi N
- Le choix se fait par n entrées adresse
- Il faut que $2^n \geq N$

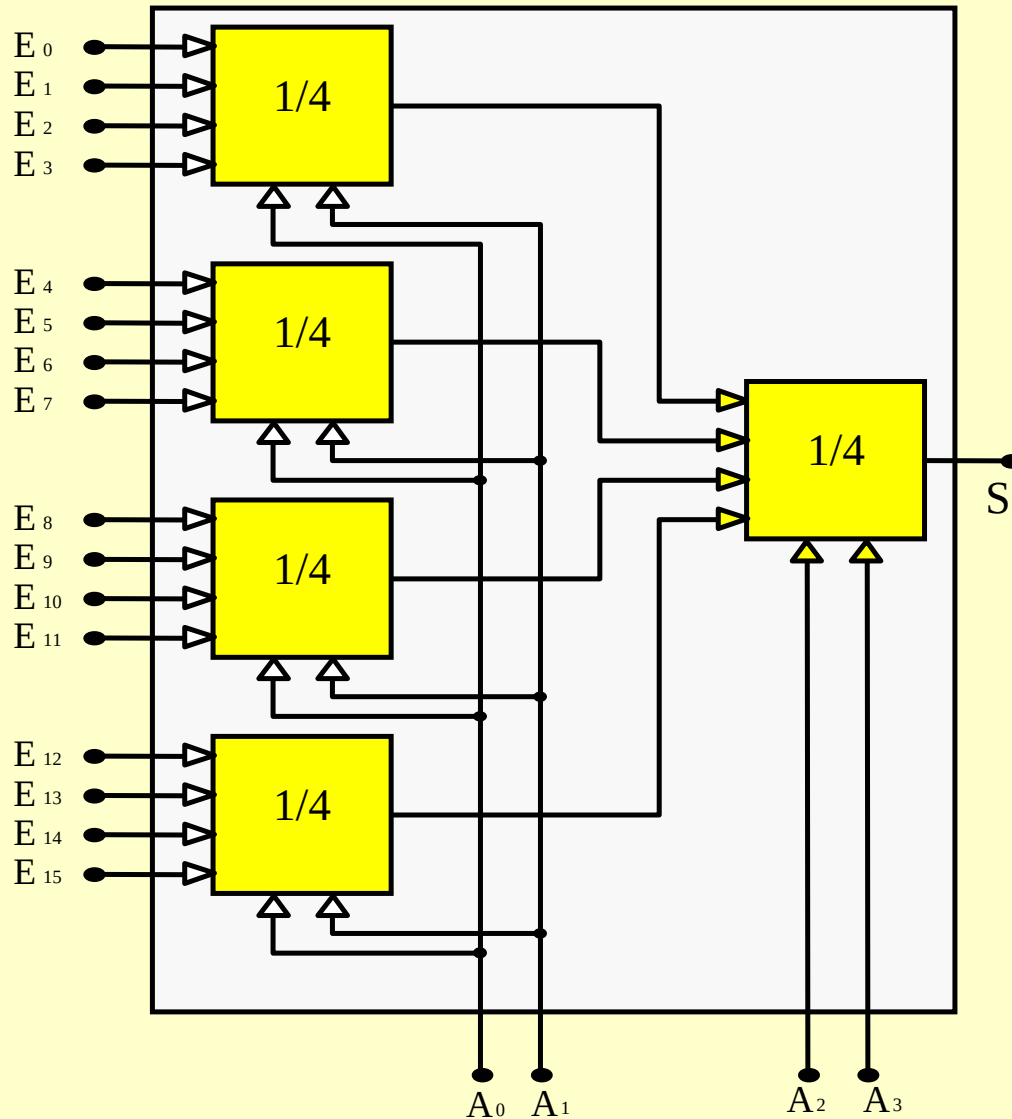
Multiplexeur 1 parmi 4



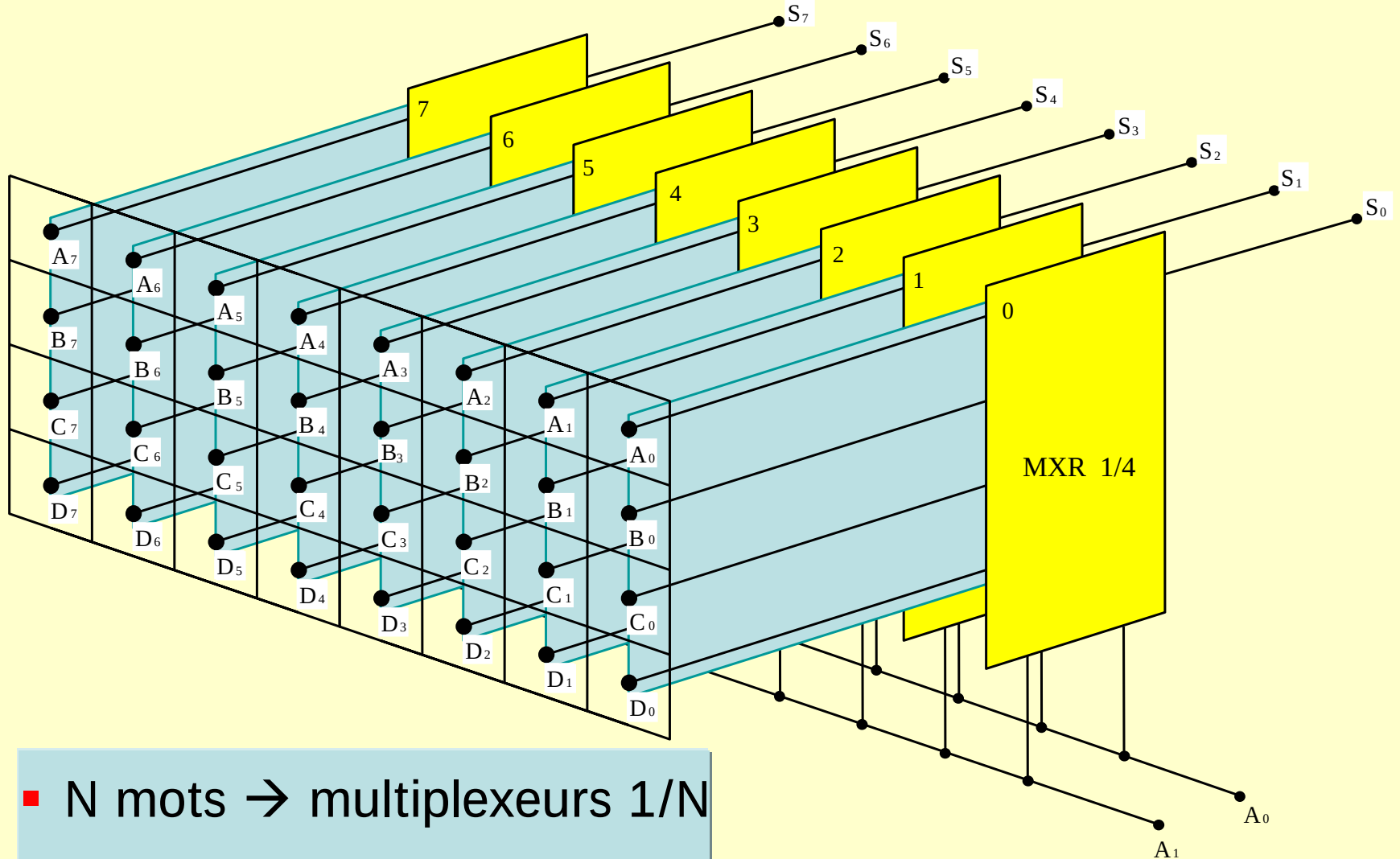
$$S = E_0 \overline{A_0} \overline{A_1} + E_1 A_0 \overline{A_1} + E_2 \overline{A_0} A_1 + E_3 A_0 A_1$$



Association de multiplexeurs

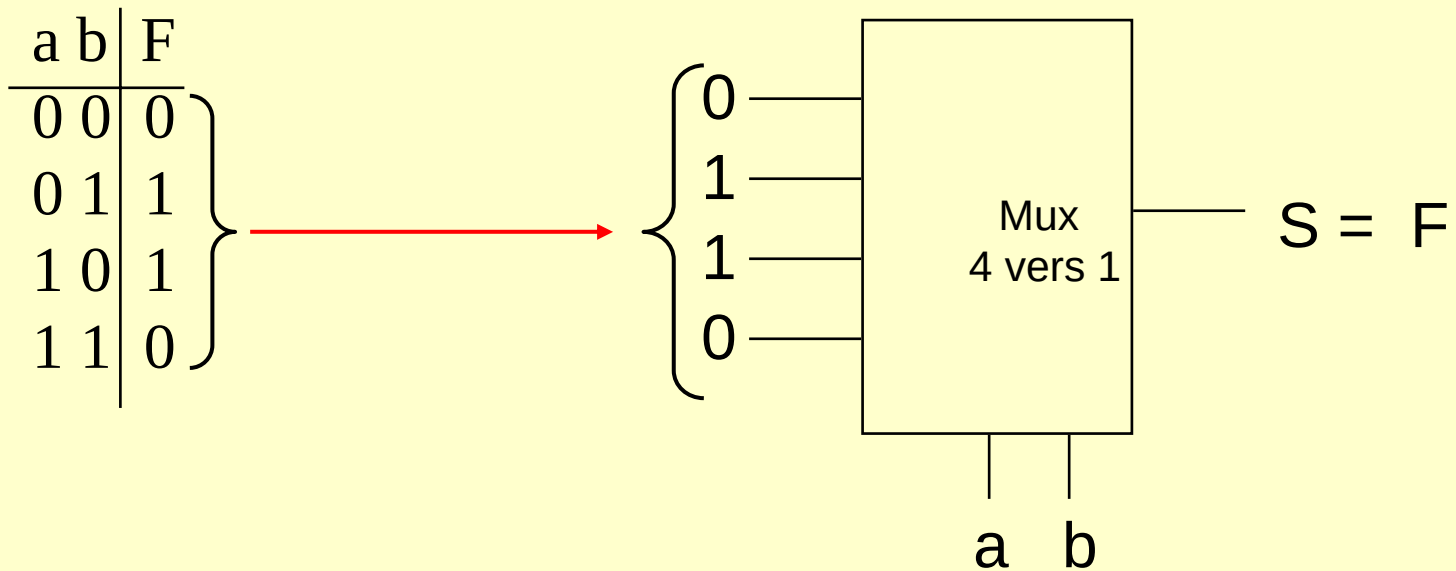


Sélectionner un mot parmi 4



- N mots $\rightarrow$ multiplexeurs 1/N
- M bits $\rightarrow$ M multiplexeurs

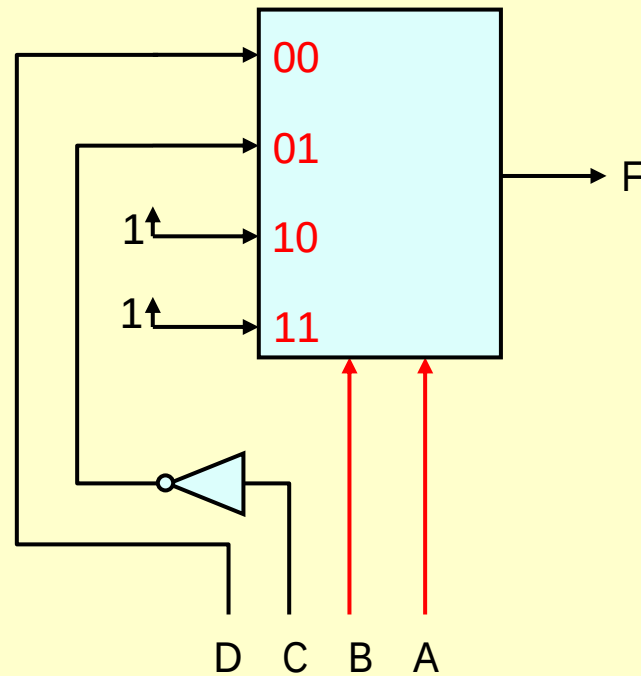
Réalisation de fonctions



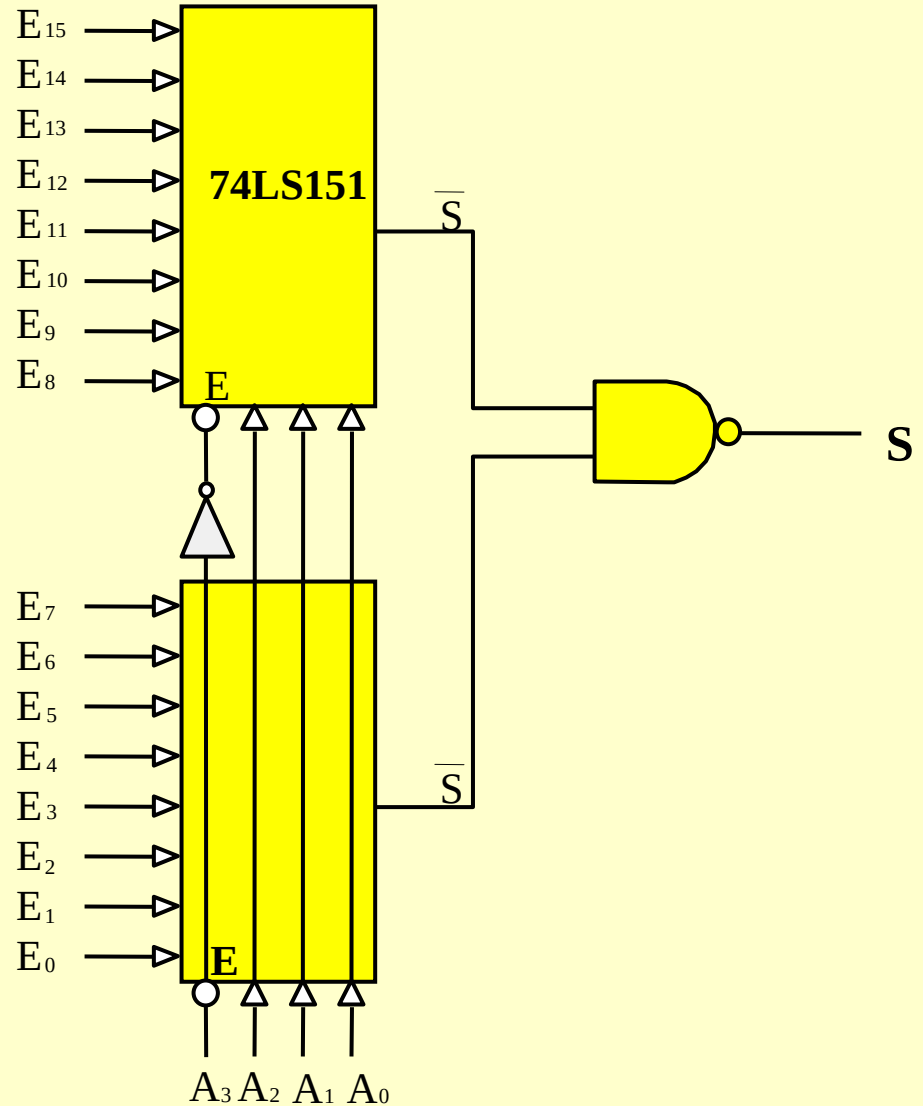
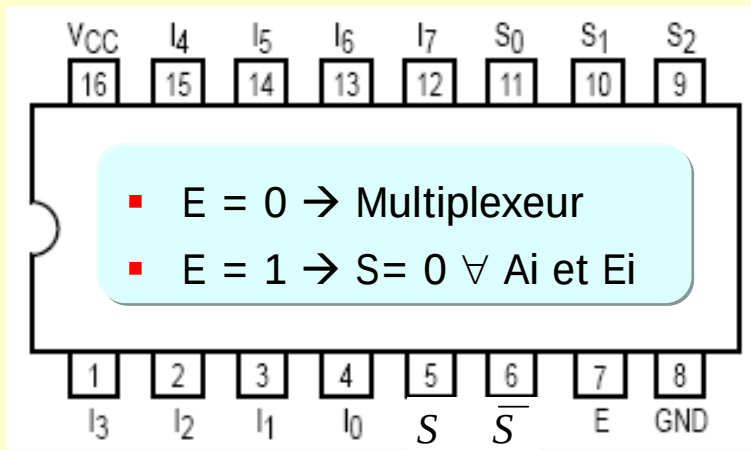
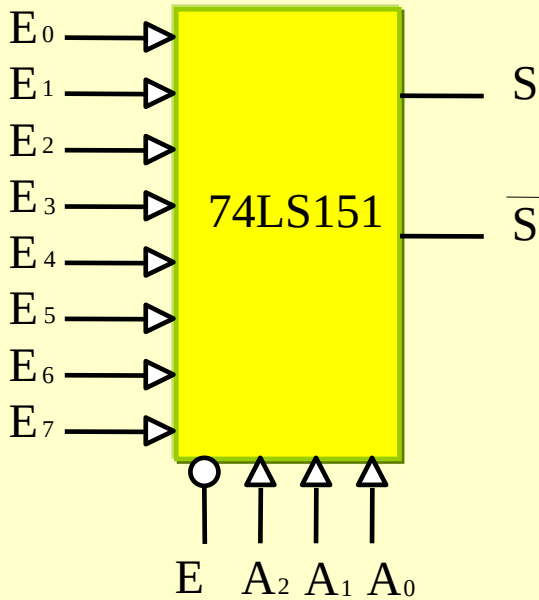
- Toute fonction logique de n variables est réalisable à l'aide d'un multiplexeur 1 parmi 2^n
- On utilise la première forme canonique, la fonction est définie pour tous les monomes

Réalisation d'une fonction simplifiée

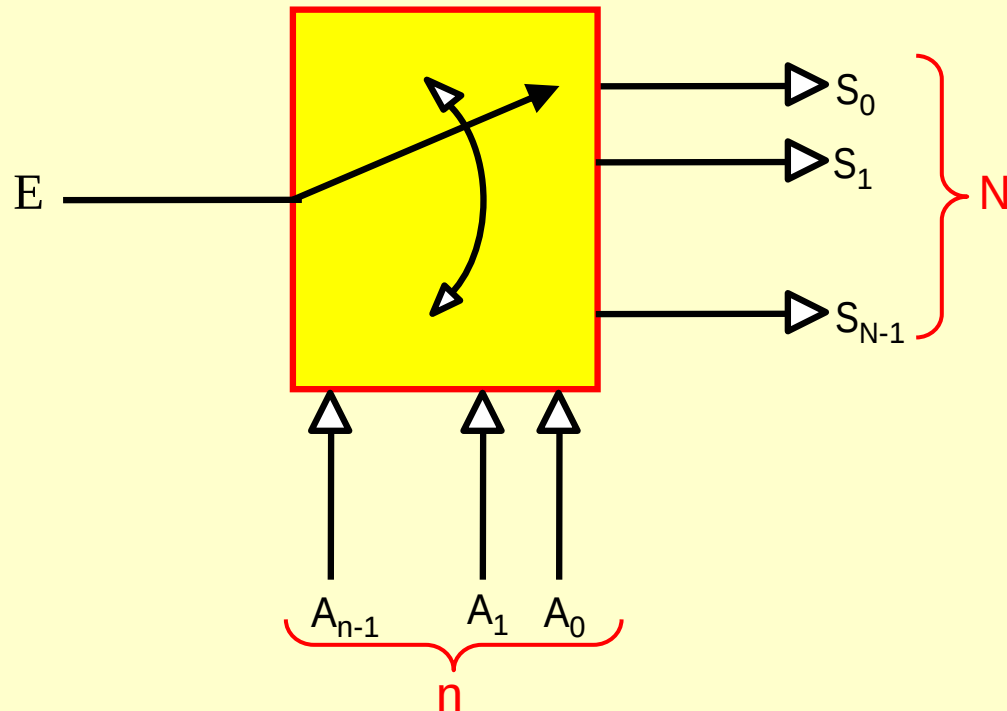
$$F = B + A\overline{C} + \overline{A}D$$



Multiplexeur du commerce : exemple



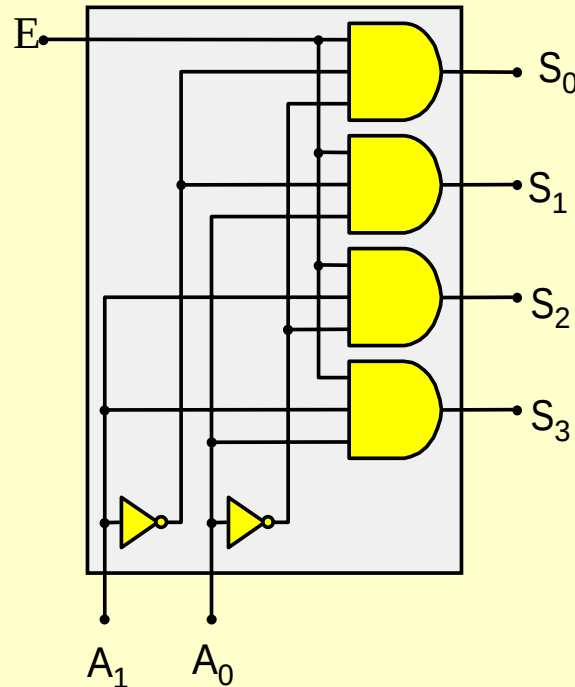
Les démultiplexeurs



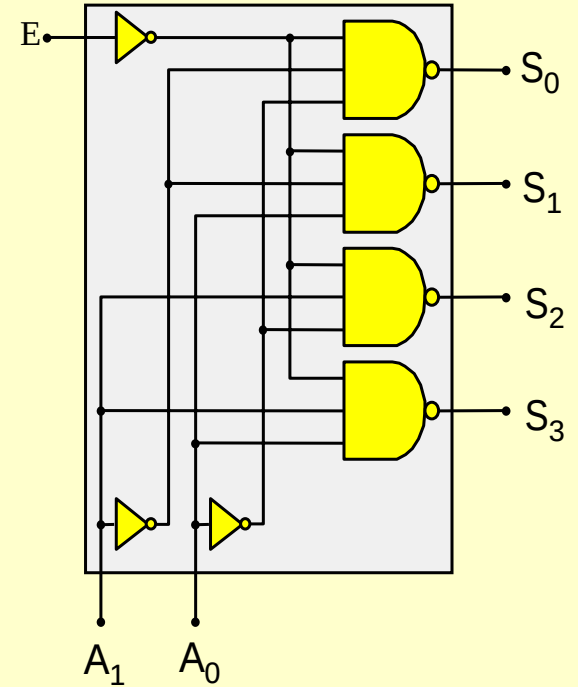
- Aiguille l'entrée vers une des **N sorties**
- Le choix se fait par **n** entrées adresse
- Il faut que **$2^n \geq N$**

Démultiplexeur 1 / 4

$$\begin{aligned} S_0 &= E \overline{A_1} \overline{A_0} \\ S_1 &= E \overline{A_1} A_0 \\ S_2 &= E A_1 \overline{A_0} \\ S_3 &= E A_1 A_0 \end{aligned}$$



Sortie non
sélectionnée
= 0

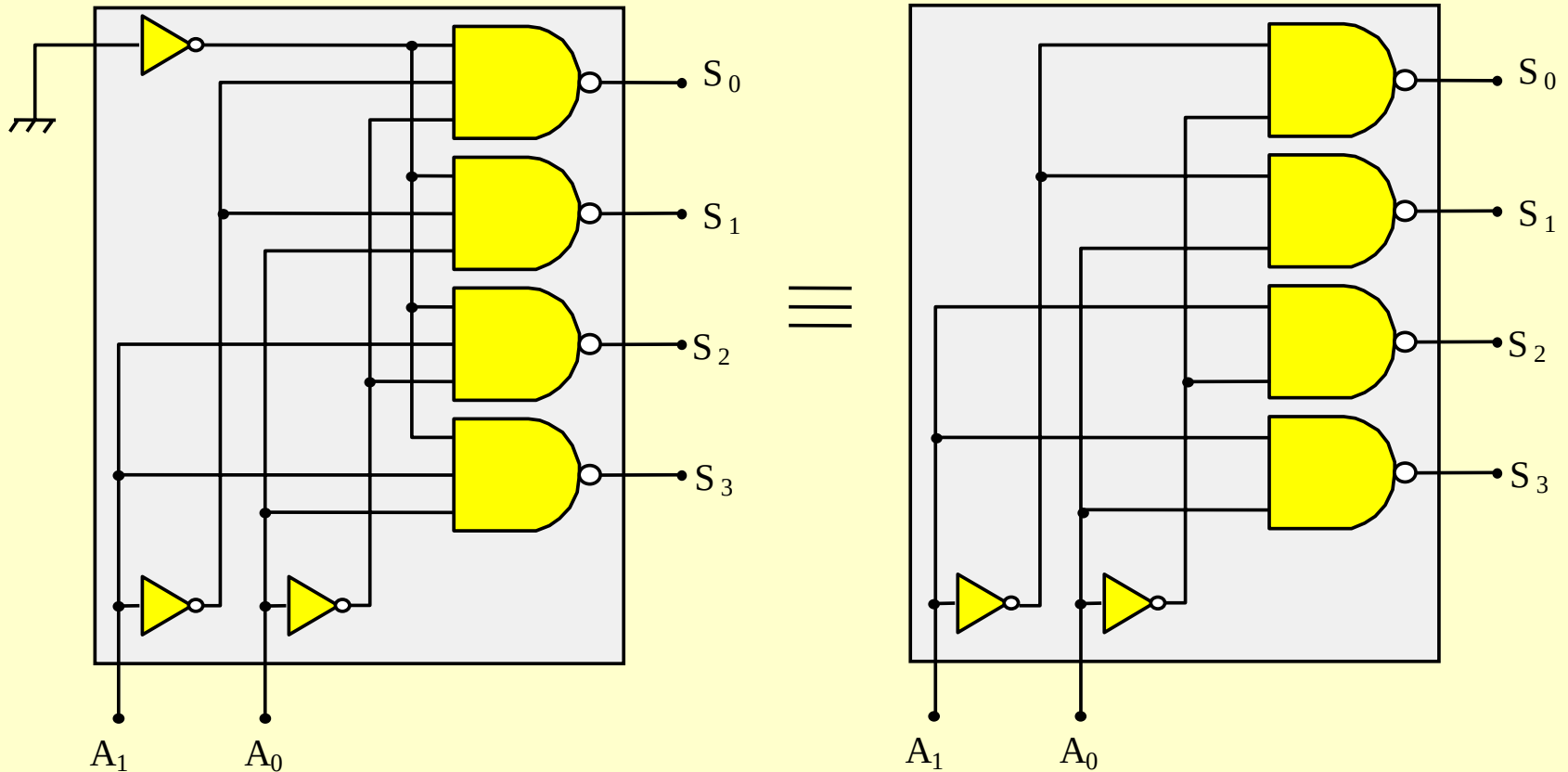


Sortie non
sélectionnée
= 1

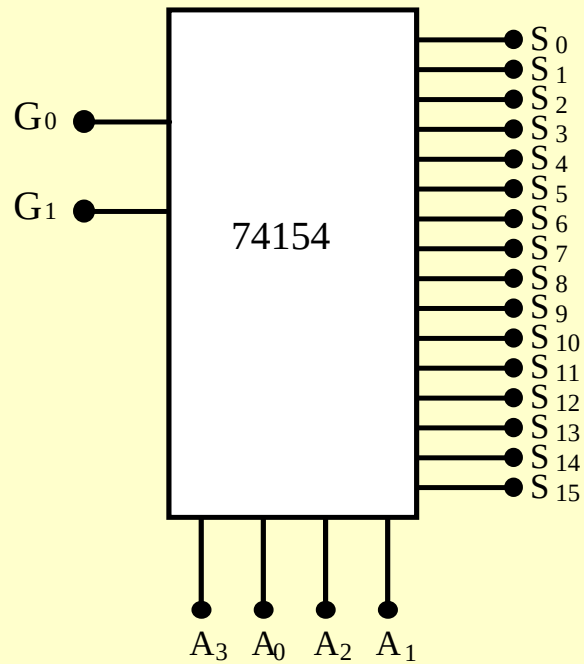
Décodeur

Un décodeur est un démultiplexeur particulier

- La sortie sélectionnée = 0
- Les sorties non sélectionnées = 1

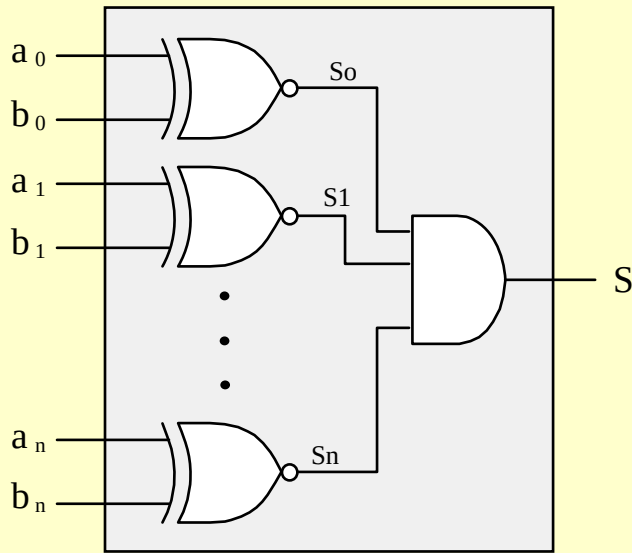
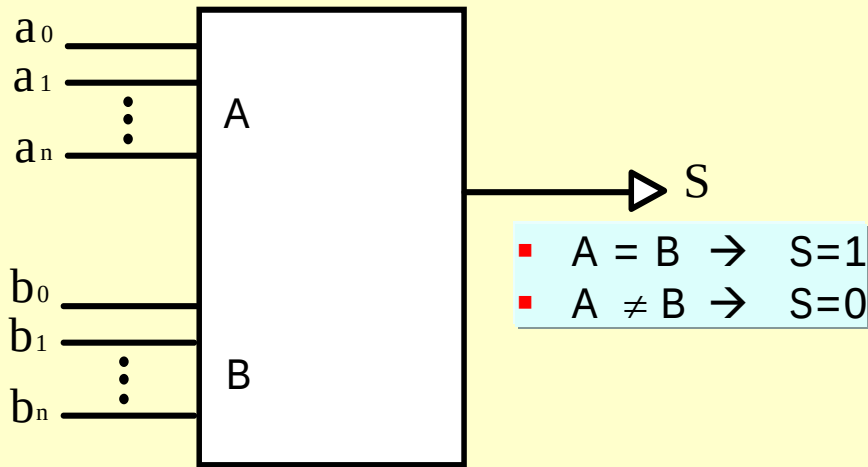


Décodeur / Démultiplexeur du commerce



G_1	G_0	
0	0	décodeur
0	entrée	Démultiplexeur
entrée	0	
1	1	Inhibé : toutes les sorties = H

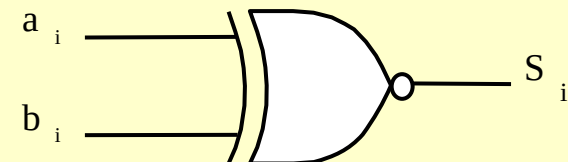
Les circuits d'identification : Comparateurs



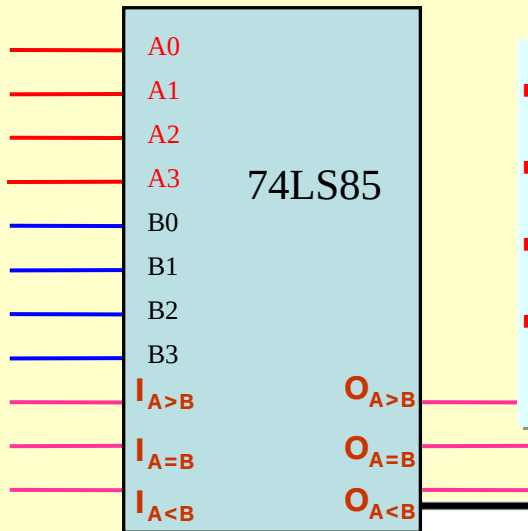
On sait que deux nombres $A = a_n \dots a_3 a_2 a_1 a_0$ et $B = b_n \dots b_3 b_2 b_1 b_0$ sont égaux si tous les bits de même poids sont égaux. L'élément de base est donc le comparateur élémentaire de 2 bits :

$a_i \backslash b_i$	0	1
0	1	0
1	0	1

$$S_i = \overline{a_i} \overline{b_i} + a_i b_i = \overline{a_i \oplus b_i}$$



Comparteur du commerce : 74LS85



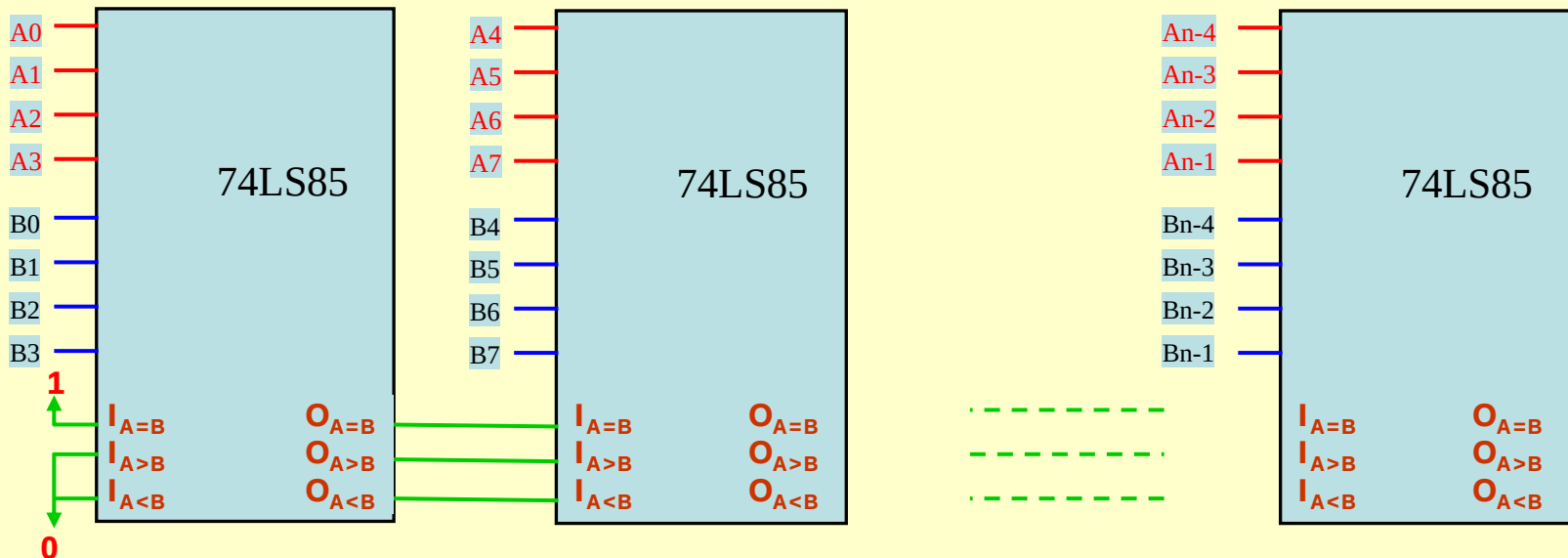
- Comparateur de 2 mots de 4 bits
- $A > B \rightarrow O_{A>B} = H, O_{A=B} = L, O_{A<B} = L$
- $A < B \rightarrow O_{A<B} = H, O_{A=B} = L, O_{A>B} = L$
- $A = B \rightarrow O_{A=B}$ est connectée à $I_{A=B}$
pour les autres, sorties voir tableau

COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A ₃ ,B ₃	A ₂ ,B ₂	A ₁ ,B ₁	A ₀ ,B ₀	I _{A>B}	I _{A<B}	I _{A=B}	O _{A>B}	O _{A<B}	O _{A=B}
A ₃ >B ₃	X	X	X	X	X	X	H	L	L
A ₃ <B ₃	X	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ >B ₂	X	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ <B ₂	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ >B ₁	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ <B ₁	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ >B ₀	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ <B ₀	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	L	L	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	H	L	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	X	X	H	L	L	H
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	H	L	L	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	L	L	H	H	L

Cascadage des 74LS85

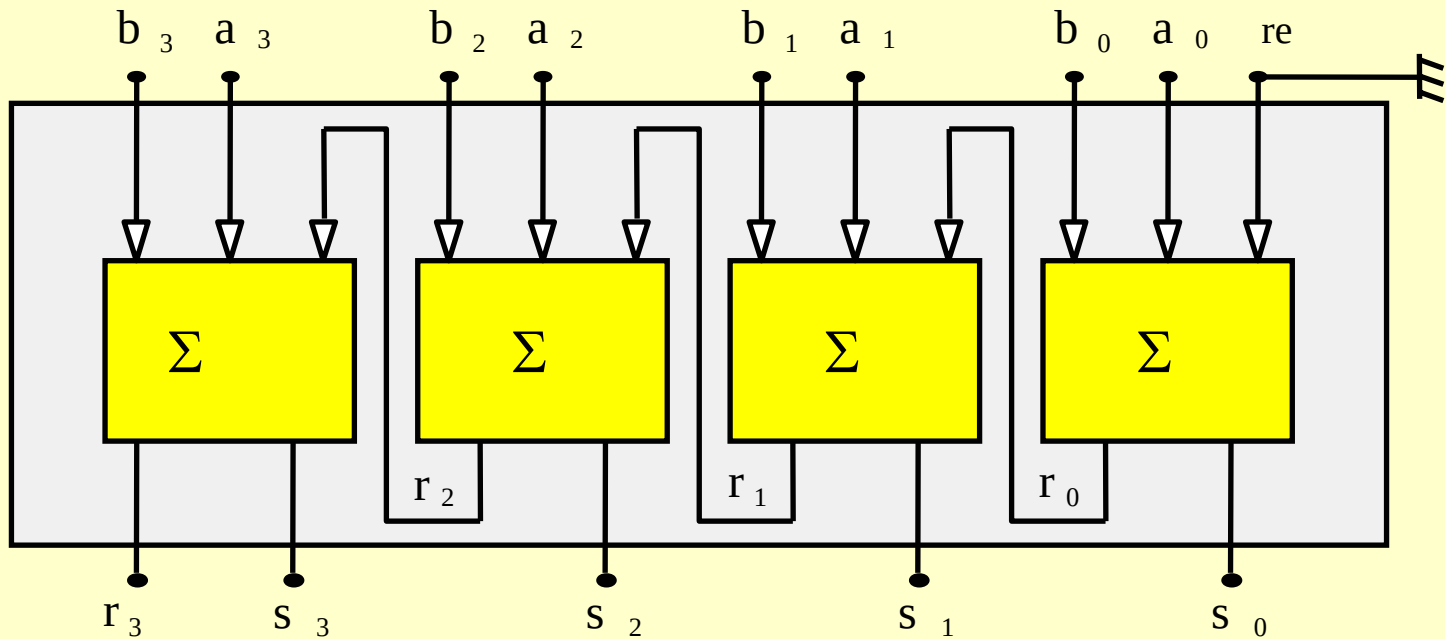
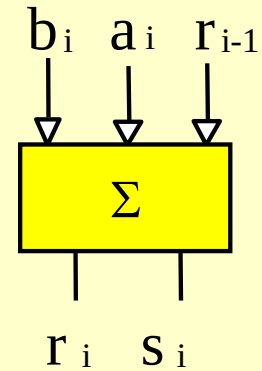
Pour comparer Deux mots de plus de 4 bits chacun :

- Pour le comparateur de plus faible poids :
 $I_{A<B} = L$, $I_{A>B} = L$, $I_{A=B} = H$
- Pour les autres, connecter les entrées de cascadage aux sortie coresponsables



Les circuits Arithmétique : Additionneur

$$\begin{array}{r}
 r_{n-2} \quad r_1 \quad r_0 \\
 a_{n-1} \quad a_2 \quad a_1 \quad a_0 \\
 b_{n-1} \quad b_2 \quad b_1 \quad b_0 \quad + \\
 \hline
 r_{n-1} \quad s_{n-1} \quad s_2 \quad s_1 \quad s_0
 \end{array}$$



Additionneur élémentaire

$r_{i-1} \backslash a_i b_i$	00	01	11	10
0	0	1	0	1
1	1	0	1	0

S_i

$r_{i-1} \backslash a_i b_i$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

R_i

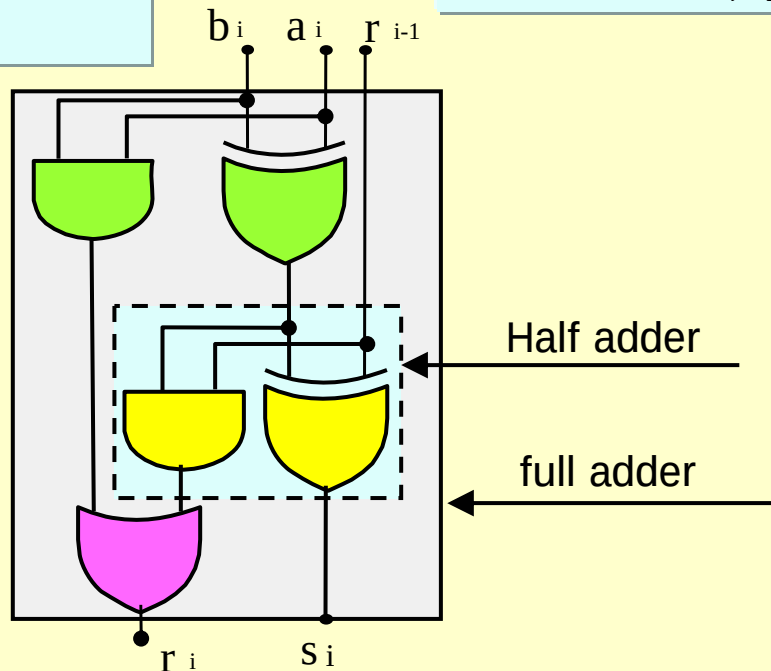
$$s_i = \overline{r_{i-1}}(a_i \overline{b_i} + \overline{a_i} b_i) + r_{i-1}(\overline{a_i} \overline{b_i} + a_i b_i)$$

$$s_i = \overline{r_{i-1}} a_i \oplus b_i + r_{i-1} \overline{a_i} \oplus b_i$$

$$s_i = a_i \oplus b_i \oplus r_{i-1}$$

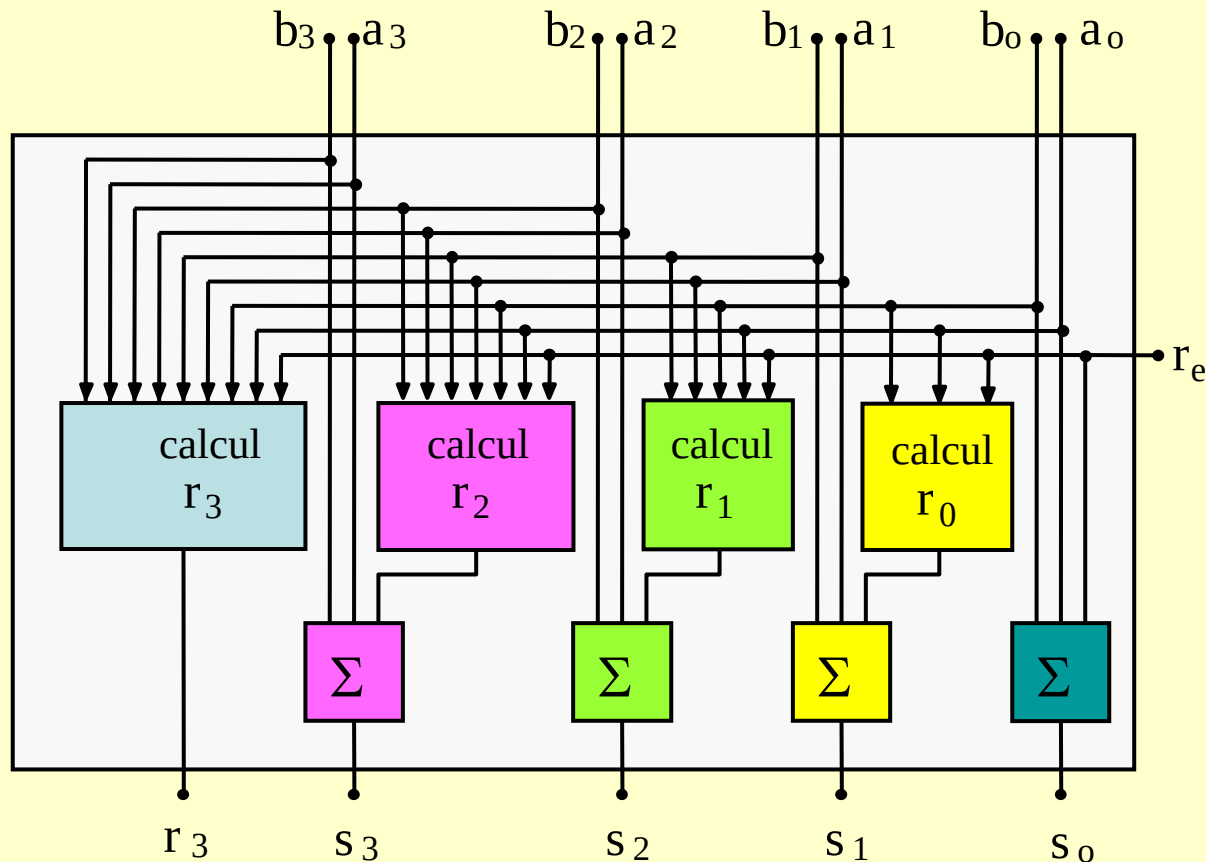
$$r_i = a_i b_i + \overline{a_i} b_i r_{i-1} + a_i \overline{b_i} r_{i-1}$$

$$r_i = a_i b_i + r_{i-1}(a_i \oplus b_i)$$



Additionneur à retenue anticipée

Pour réduire le temps de calcul dû à la propagation de la retenue, chaque étage calcule sa propre retenue entrante sans attendre les étages précédents



Bloc de génération de la retenue

$$r_i = a_i b_i + (a_i + b_i) r_{i-1}$$

$a_i b_i = G_i$: **Génération** de la retenue : la retenue est générée localement

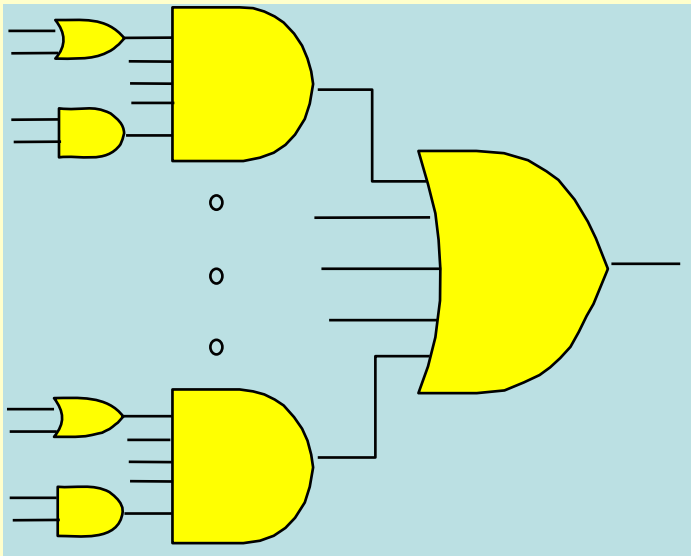
$a_i + b_i = P_i$: **Propagation** de la retenue : la retenue vient de l'étage de droite

$$r_0 = G_0 + P_0 r_e$$

$$r_1 = G_1 + P_1 r_0 = G_1 + P_1 G_0 + P_1 P_0 r_e$$

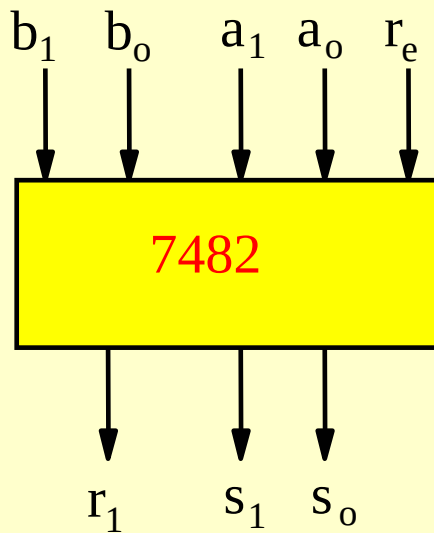
$$r_2 = G_2 + P_2 r_1 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 r_e$$

$$r_3 = G_3 + P_3 r_2 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 r_e$$

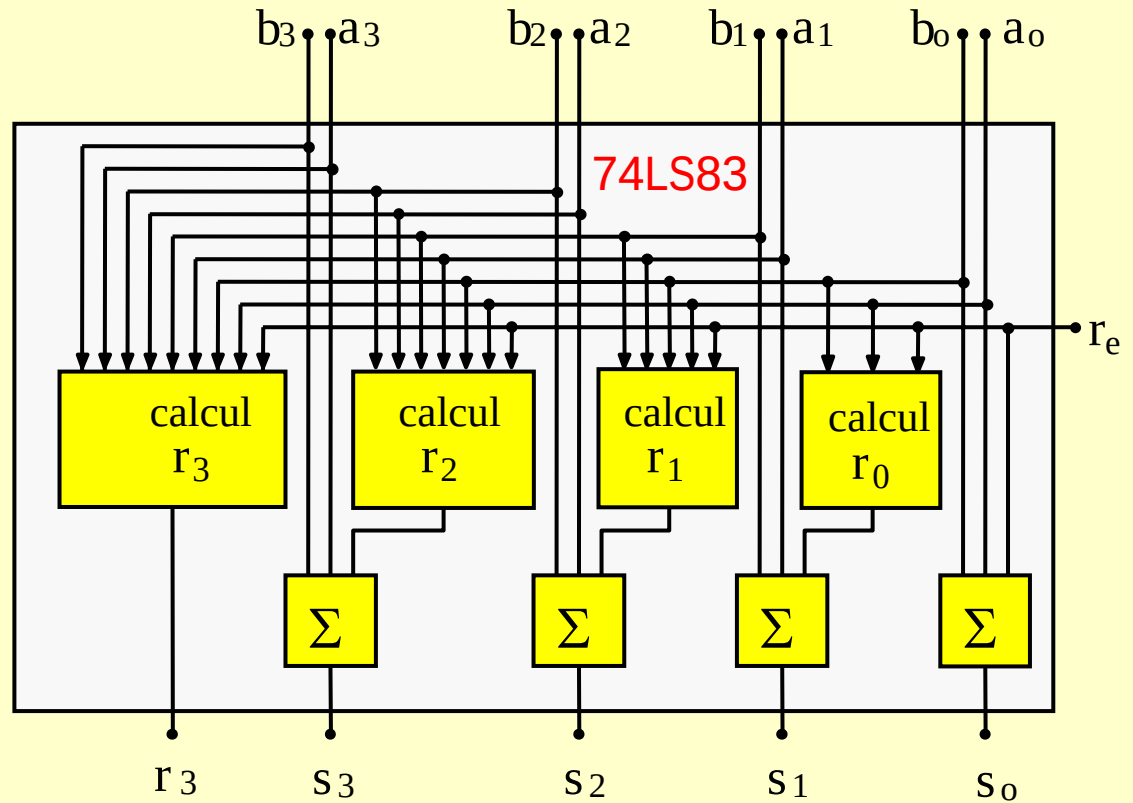


Le temps de calcul de la retenue est le même $\forall$ l'étage. Il est égal à 3 temps de propagation d'une porte

Exemple additionneurs du commerce

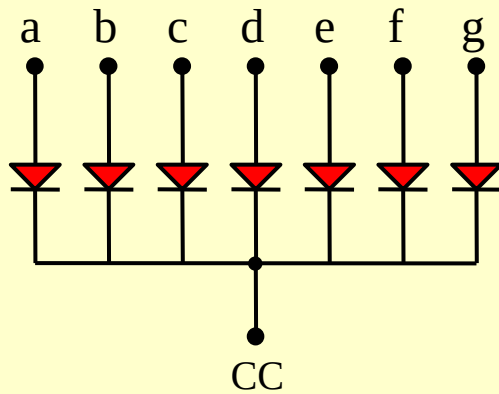
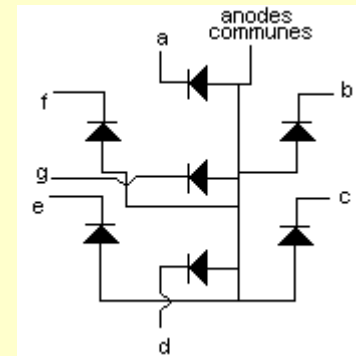
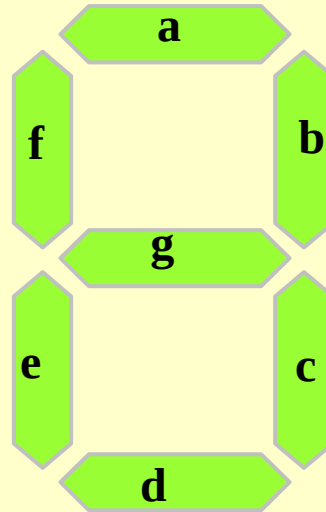
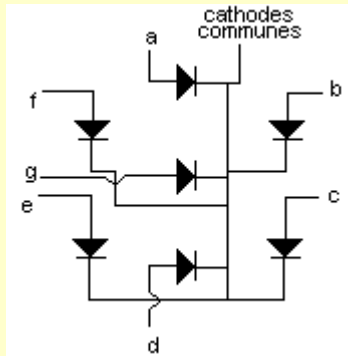


Additionneur
série 2 bits

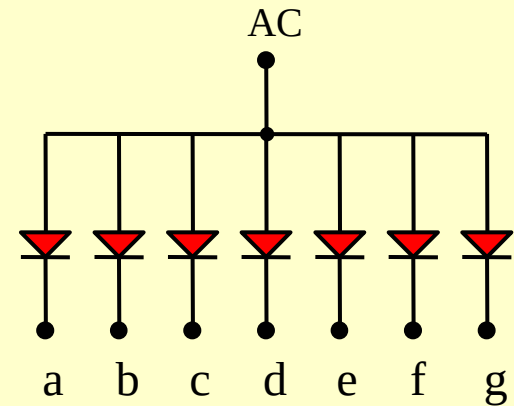


Additionneur à retenue
anticipée 4 bits

Afficheurs 7 segments

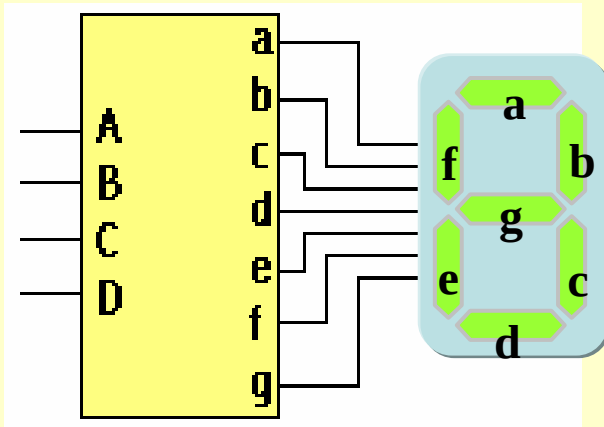


Se commandent
par niveau haut



Se commandent
par niveau bas

Décodeur BCD 7 segments – Cathode commune

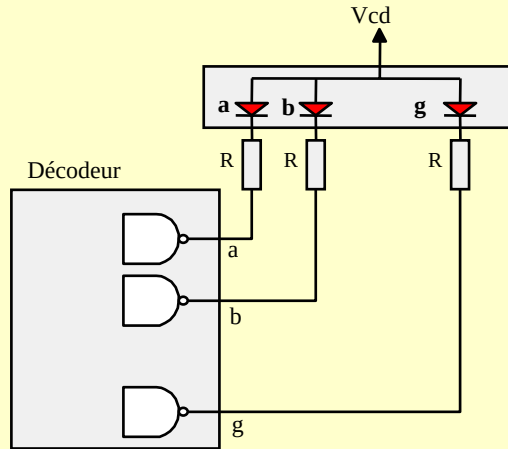


D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	1	1	1	1	1	1	0	8
0	0	0	1	0	1	1	0	0	0	0	9
0	0	1	0	1	1	0	1	1	0	1	8
0	0	1	1	1	1	1	1	0	0	1	8
0	1	0	0	0	1	1	0	0	1	1	8
0	1	0	1	1	0	1	1	0	1	1	8
0	1	1	0	0	0	1	1	1	1	1	8
0	1	1	1	1	1	1	0	0	0	0	8
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	0	0	1	1	8

BA \ DC	00	01	11	10
00	1	0	1	1
01	0	1	1	0
11	x	x	x	x
10	1	1	x	x

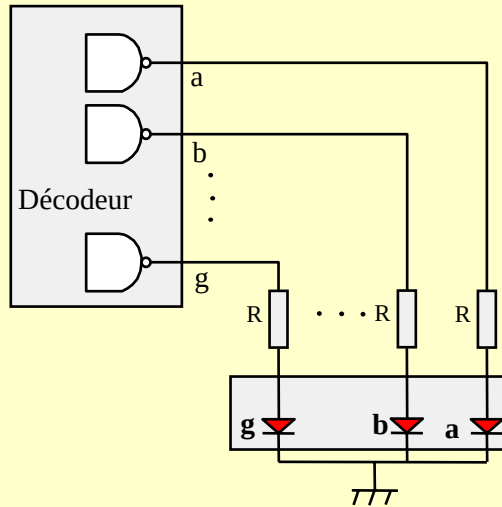
$$\begin{aligned}
 a &= \overline{A}\overline{C} + \overline{A}B + D + AC & e &= \overline{A}\overline{C} + \overline{A}B \\
 b &= \overline{A}\overline{B} + \overline{C} + AB & f &= \overline{C}\overline{B} + D + \overline{A}\overline{B} + \overline{A}C \\
 c &= \overline{B} + A + C & g &= \overline{B}\overline{C} + D + \overline{A}B + \overline{B}C \\
 d &= \overline{A}\overline{B}C + \overline{A}\overline{C} + \overline{A}B + \overline{B}\overline{C}
 \end{aligned}$$

Pilotage des afficheur 7 segments

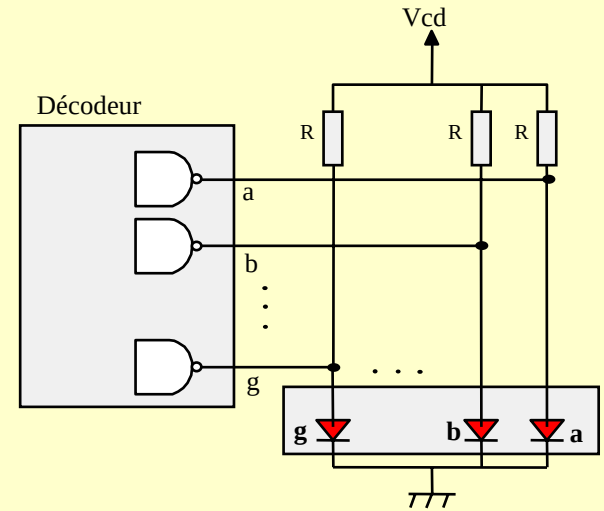


- Anode Commune
- Commande par niveau bas
- Décodeur à sortie totem-pole ou collecteur ouvert

$$R = \frac{V_{cd} - V_D}{I_{dn}}$$



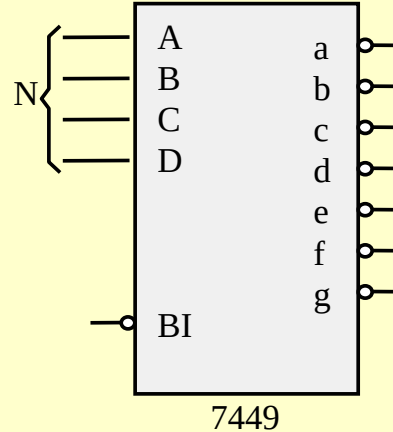
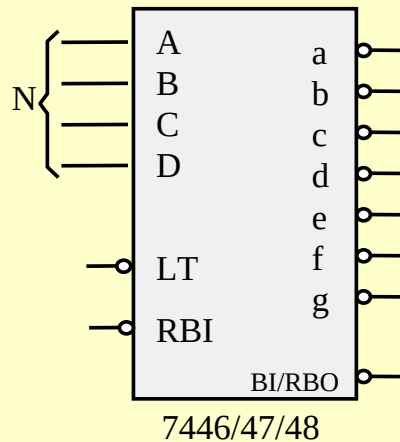
- Cathode Commune
- Commande par niveau haut, sortie totem-pole
- Faible éclaircissement
- Résistances internes au décodeur = simplicité d'utilisation



- Cathode Commune
- Commande par niveau haut, sortie OC
- Consommation même si l'afficheur est éteint

$$R = \frac{V_{cd} - V_D}{I_{dn}}$$

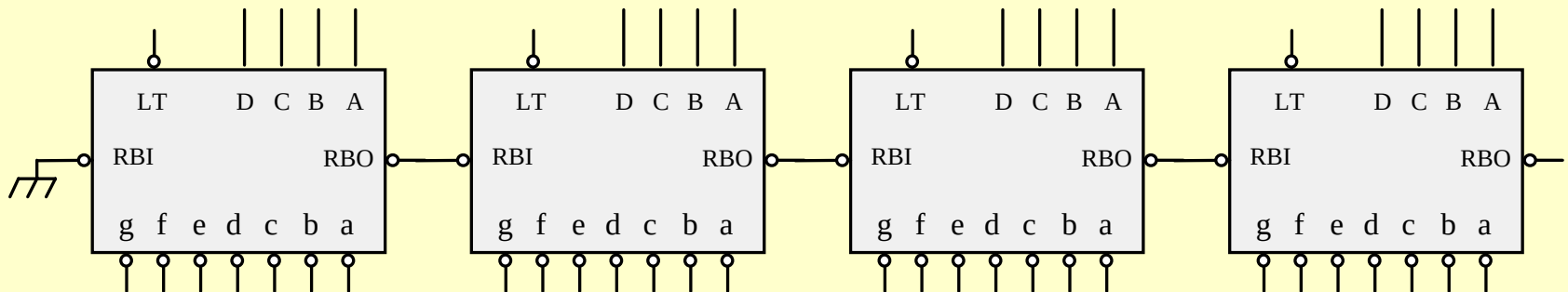
Décodeurs BCD-7seg du commerce



- 7446 /47 : commande par niveau **bas**, Sortie collecteur ouvert
- 7448 : commande par niveau **haut**. Résistances intégrées
- 7449 : commande par niveau **haut**, sortie collecteur Ouvert
- 7446 : Vcdmax = 30 V
- 7447 : Vcdmax = 15 V
- 7449 : Vcdmax = 5 V

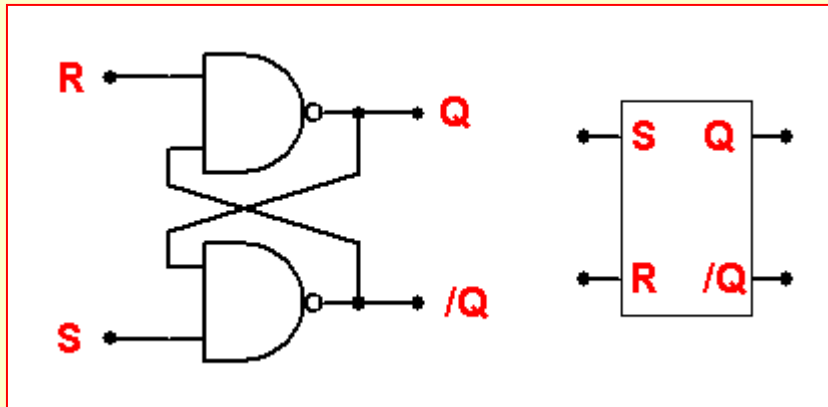
L'entrée RBI et la sortie RBO permettent de ne pas afficher les zéros de gauche

RBI	0	N $\neq$ 0 , il est affiché et RBO = 1 N = 0, l'afficheur est éteint et RBO = 0
	1	N affiché $\forall$ sa valeur, RBO = 1

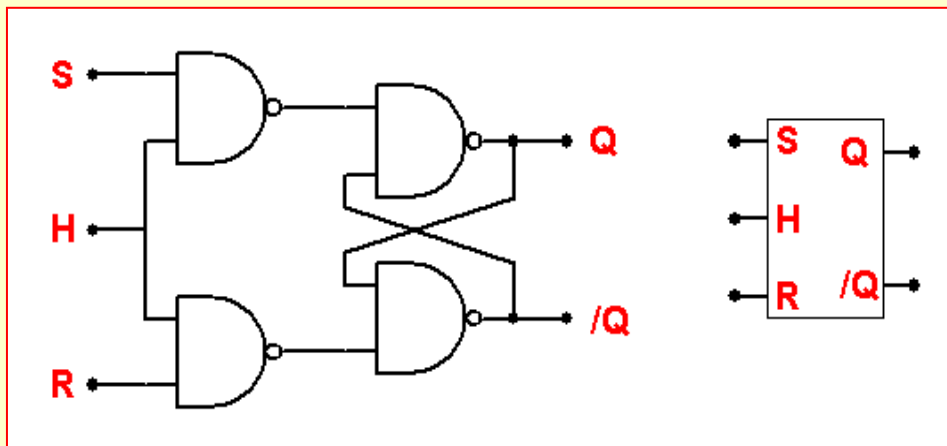


Les circuits Séquentiels Usuels

Bascule RS et RSH



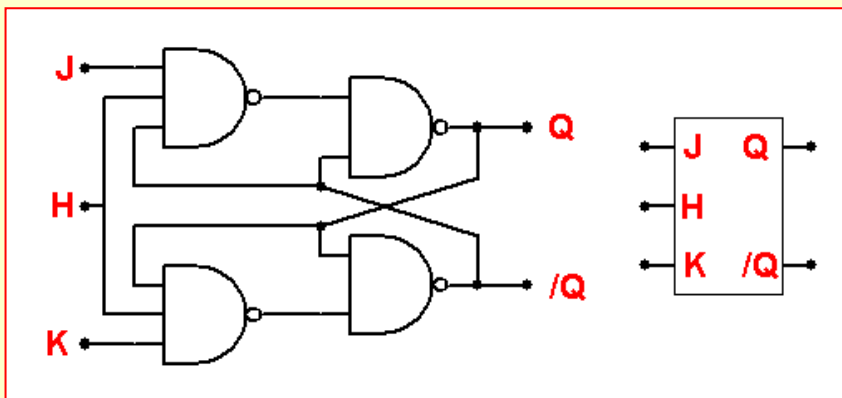
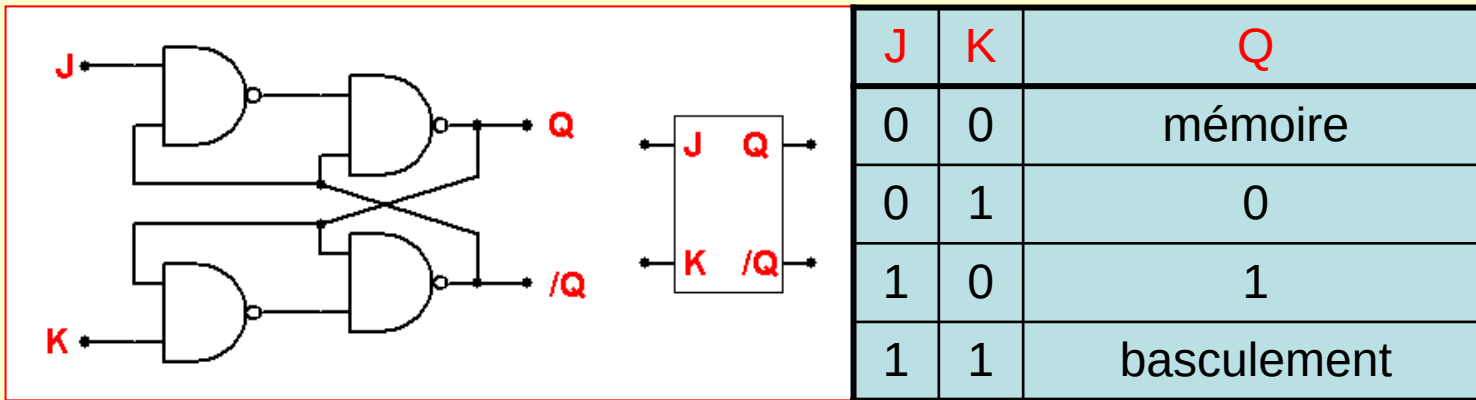
S	R	Q
0	0	indéterminé
0	1	0
1	0	1
1	1	mémoire



- H=1 → Table de vérité
- H=0 → mémoire

Bascule JK et JKH

La bascule JK lève l'indétermination du cas entrées = 00



- H=1 → Table de vérité
- H=0 → sorties inchangées

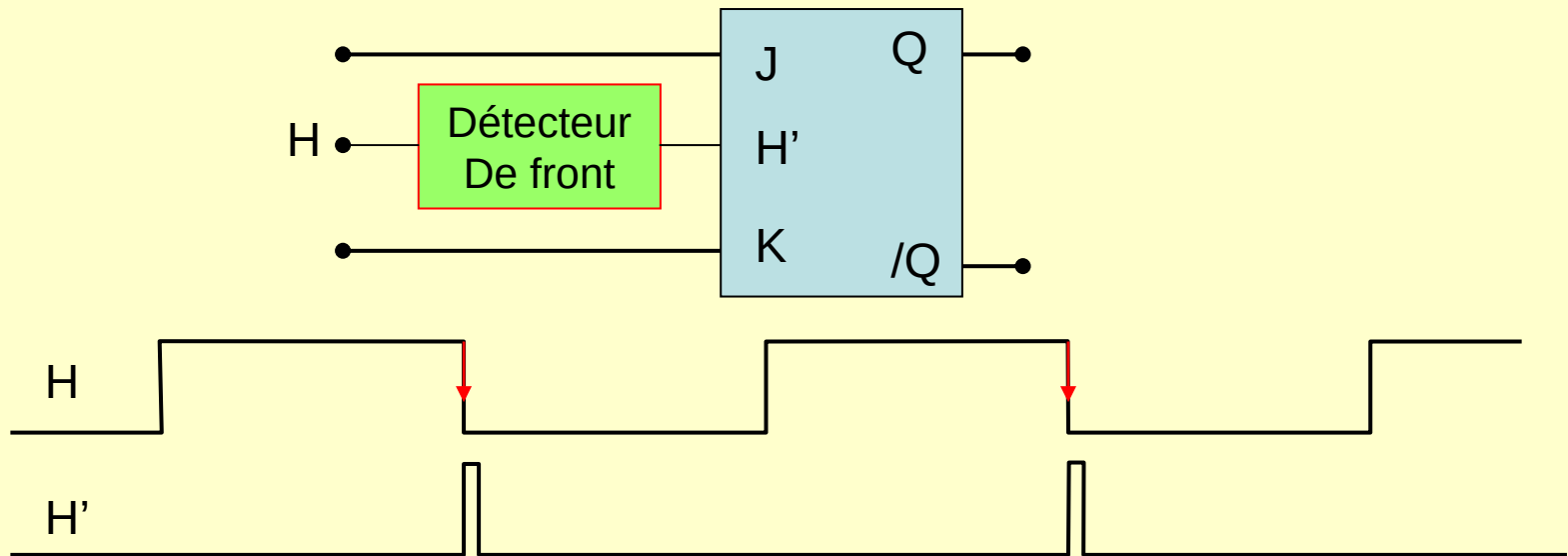
Les bascules réagissant sur front d'horloge

Bascule en permanence dans l'état **Mémoire** sauf pendant les fronts de l'horloge

Deux techniques :

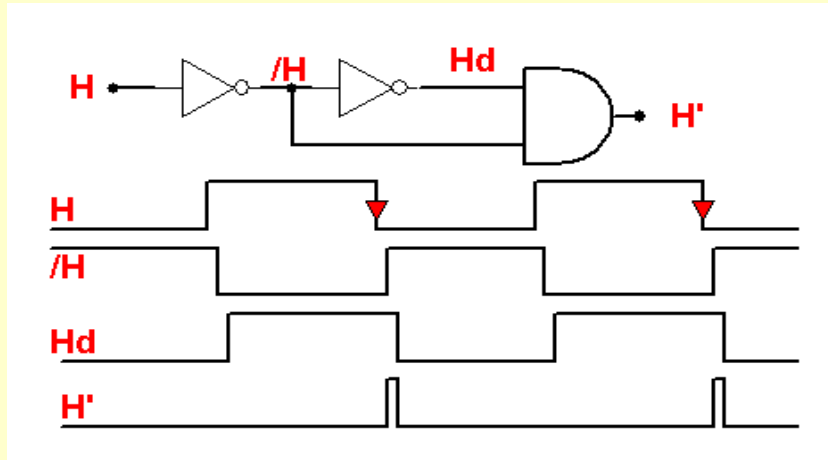
- Utiliser un détecteur de front sur l'entrée horloge
- Utiliser une structure Maître Esclave

Bascule JK avec détecteur de Front descendant

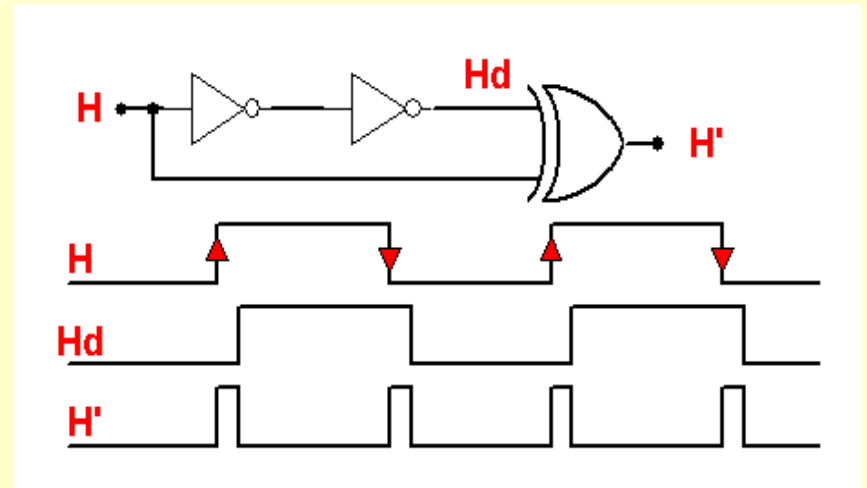
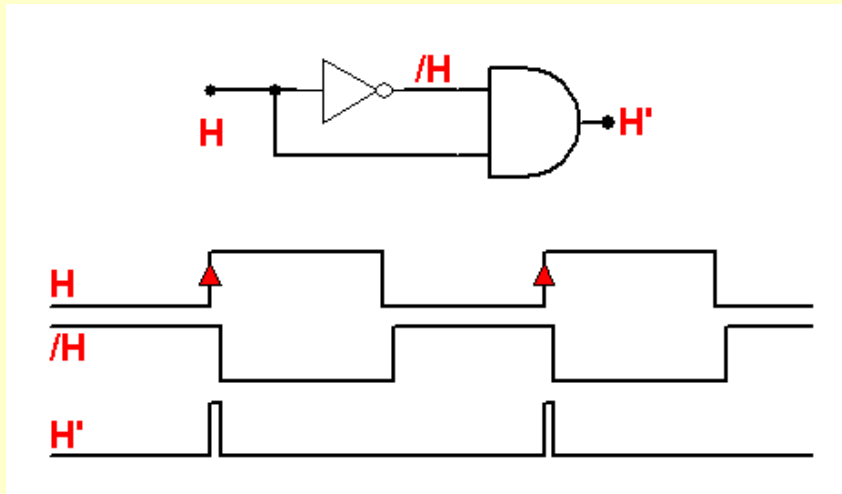


H	J	K	Q
↓	0	0	mémoire
↓	0	1	0
↓	1	0	1
↓	1	1	basculement

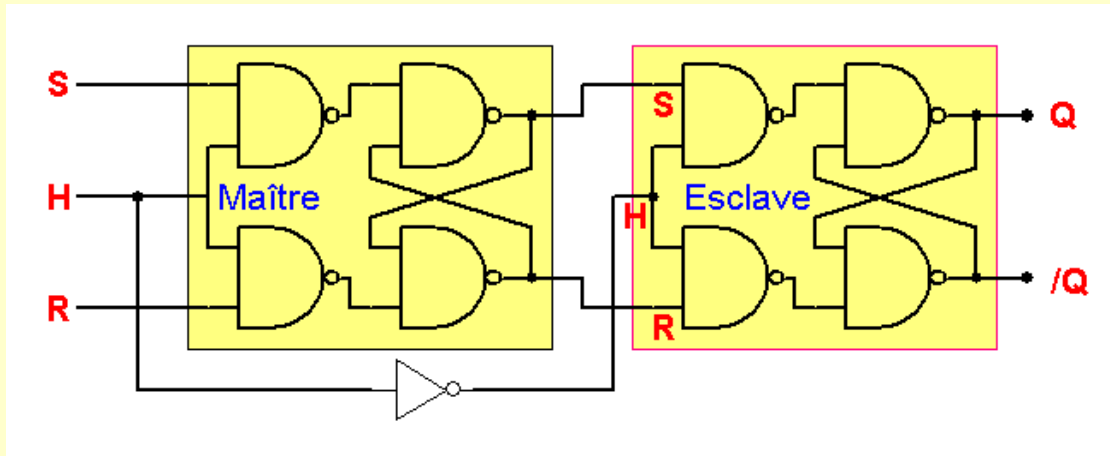
Exemples de détecteurs de front



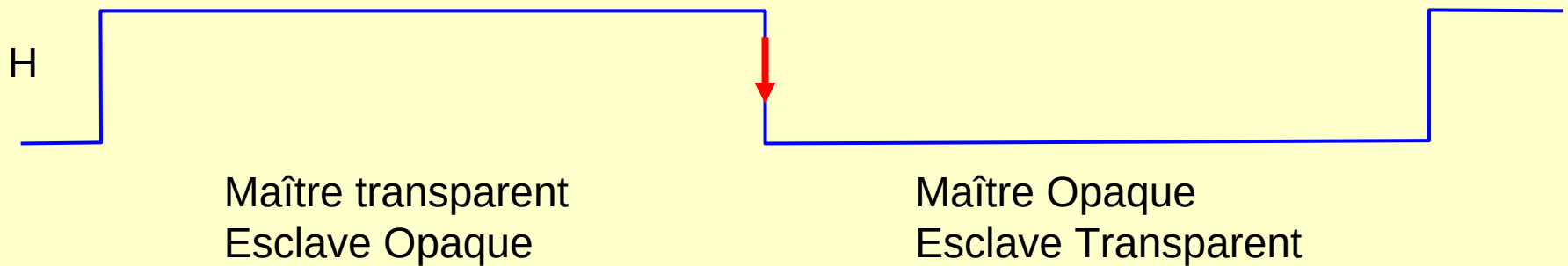
On exploite le retard élémentaire des portes logiques



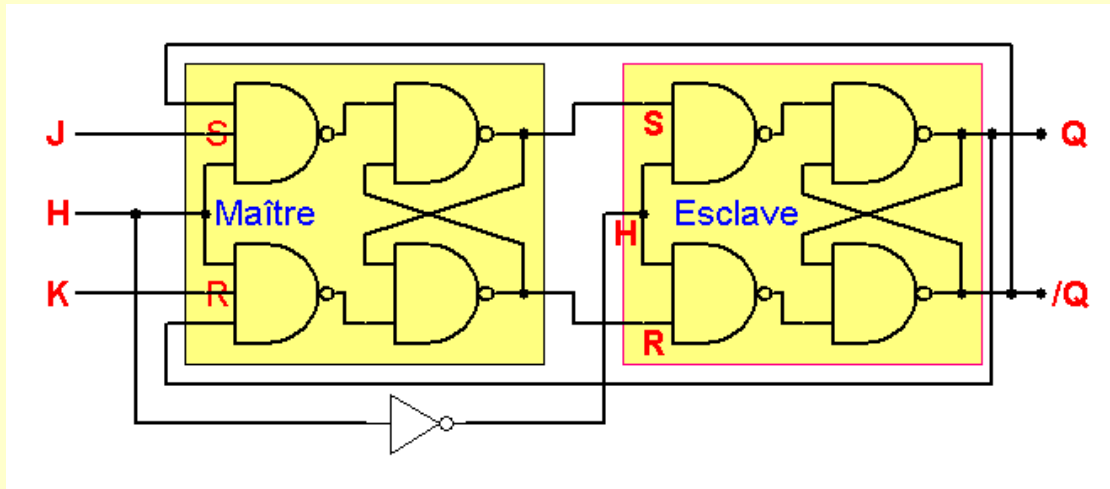
Bascule RS Maître Esclave



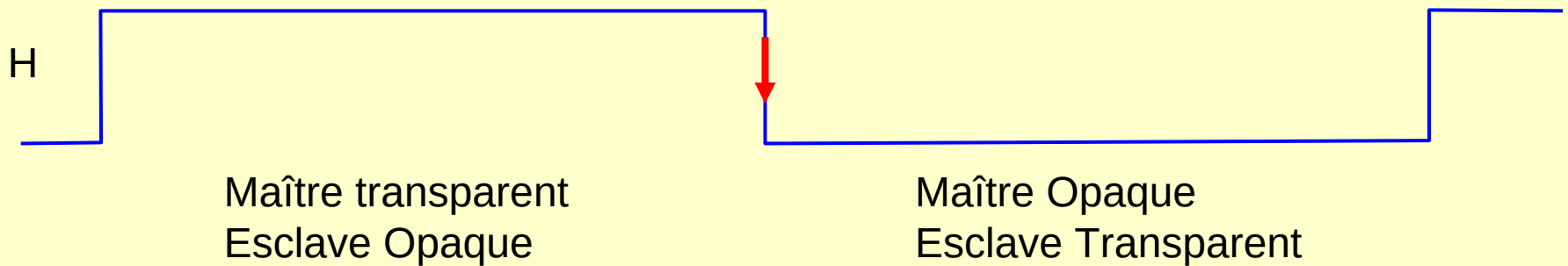
H	S	R	Q
↓	0	0	?
↓	0	1	0
↓	1	0	1
↓	1	1	mémoire



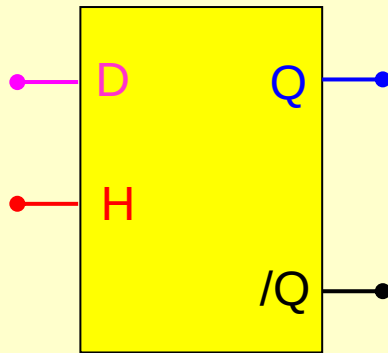
Bascule JK Maître Esclave



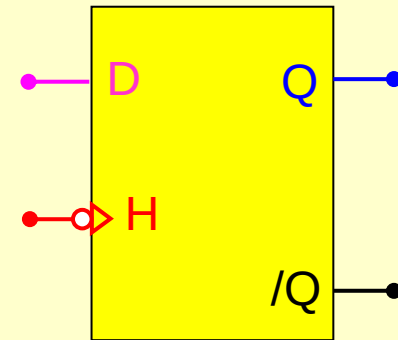
H	S	R	Q
↓	0	0	mémoire
↓	0	1	0
↓	1	0	1
↓	1	1	basculement



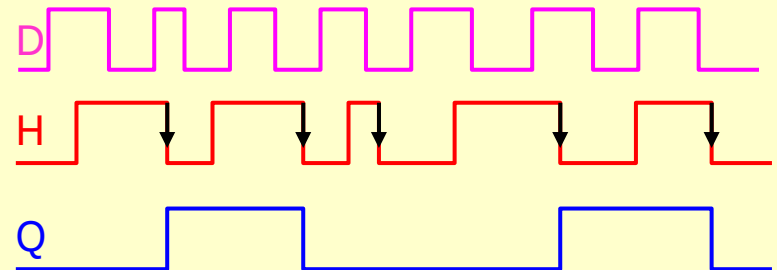
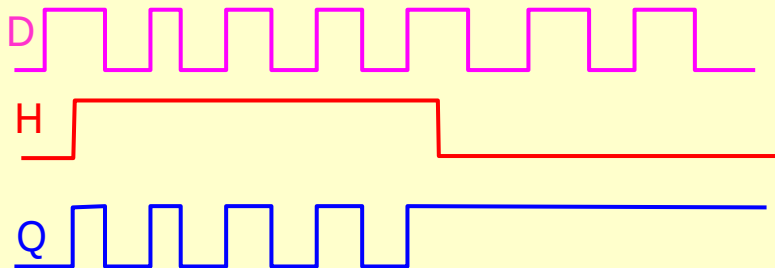
Bascule D



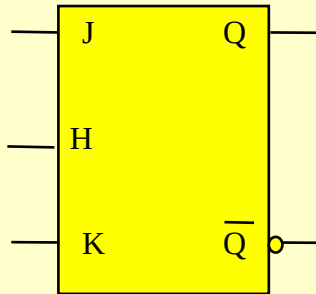
- $H = 1$: Q suit D
- $H = 0$: Q reste bloqué : état mémoire



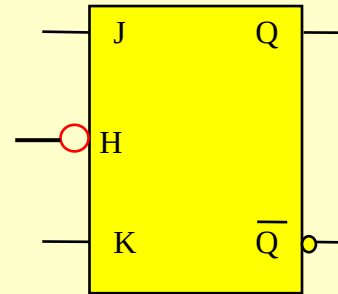
Pendant le **front descendant** de H, Q prend la valeur de D et la garde jusqu'au prochain front



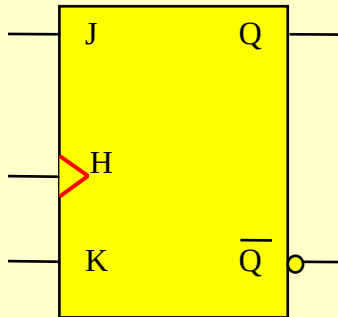
Convention de dessin



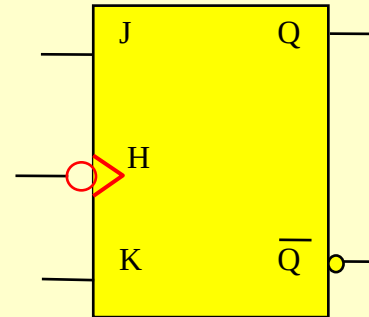
Bascule réagissant sur
niveau haut de H (latch)



Bascule réagissant sur
niveau bas de H (latch)



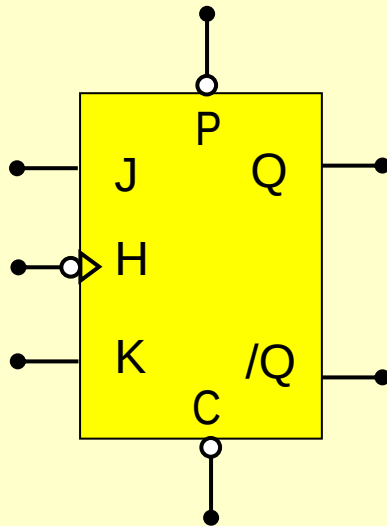
Bascule réagissant sur
front montant de H



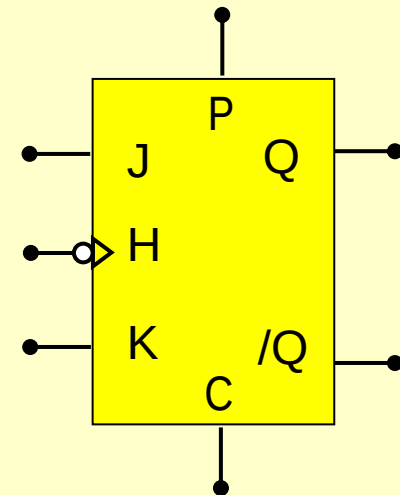
Bascule réagissant sur
front descendant de H

Entrée de forçage Clear et Preset

- L'entrée **C** force la sortie à **0** $\forall J, K, H$
- L'entrée **P** force la sortie à **1** $\forall J, K, H$
- Il ne faut pas activer C et P simultanément

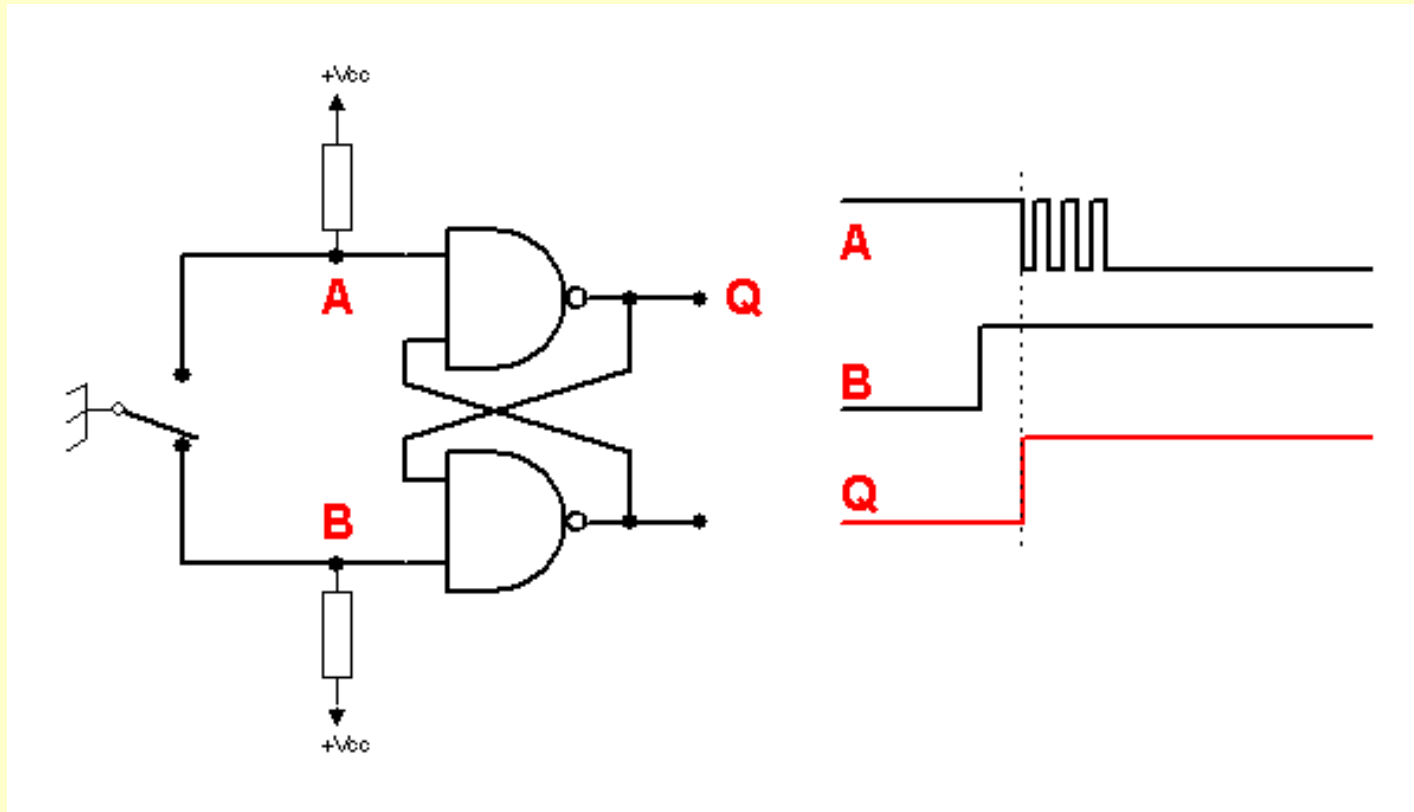


C et P actifs au
niveau bas



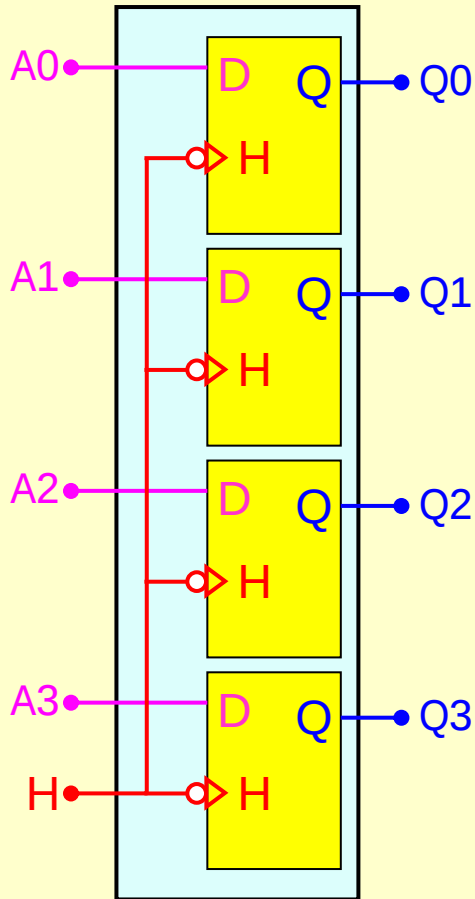
C et P actifs au
niveau haut

Application : Circuit Anti-rebond



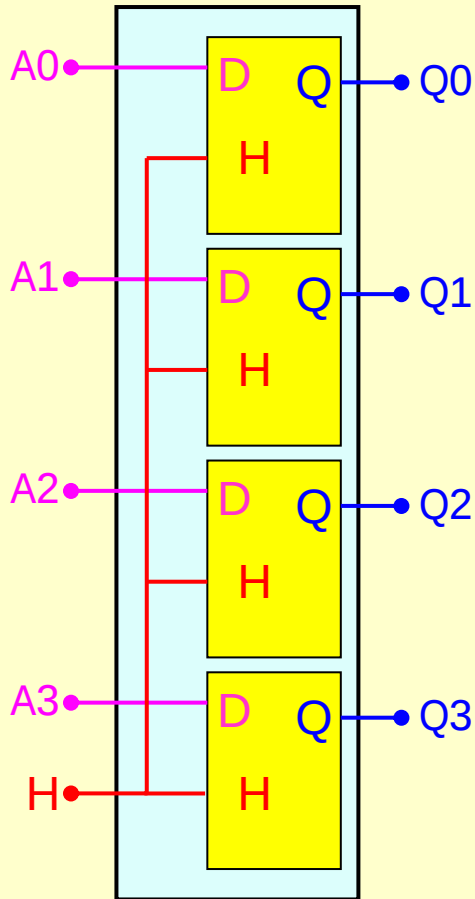
Les Registres

Registres



- Un registre est une association de N bascules réalisant la fonction de **mémoire N bits**
- Au front d'horloge, l'information $A_3A_2A_1A_0$ est copiée dans $Q_3Q_2Q_1Q_0$ et y reste jusqu'au prochain front

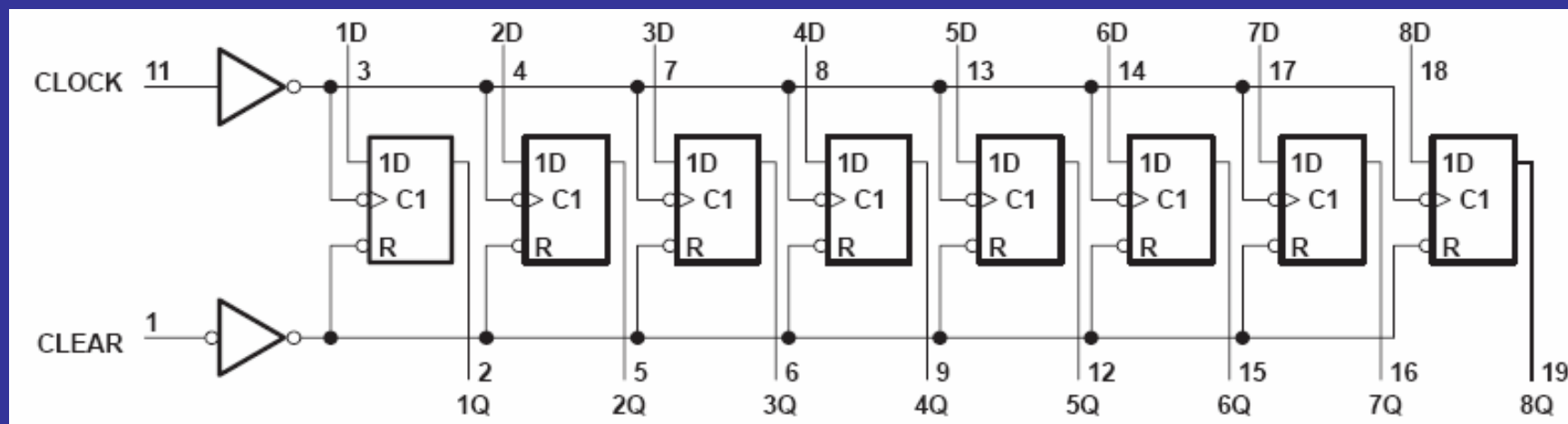
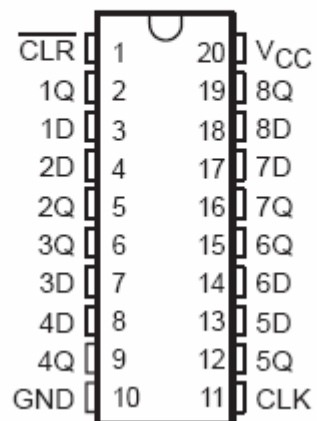
Registre Latch



- Tant que l'horloge est à 1, les sortie Q suivent les entrée A
- Quand l'horloge passe à zéros, les sorties Q restent bloquées (mémoire) jusqu'à ce que l'horloge passe de nouveau à 1

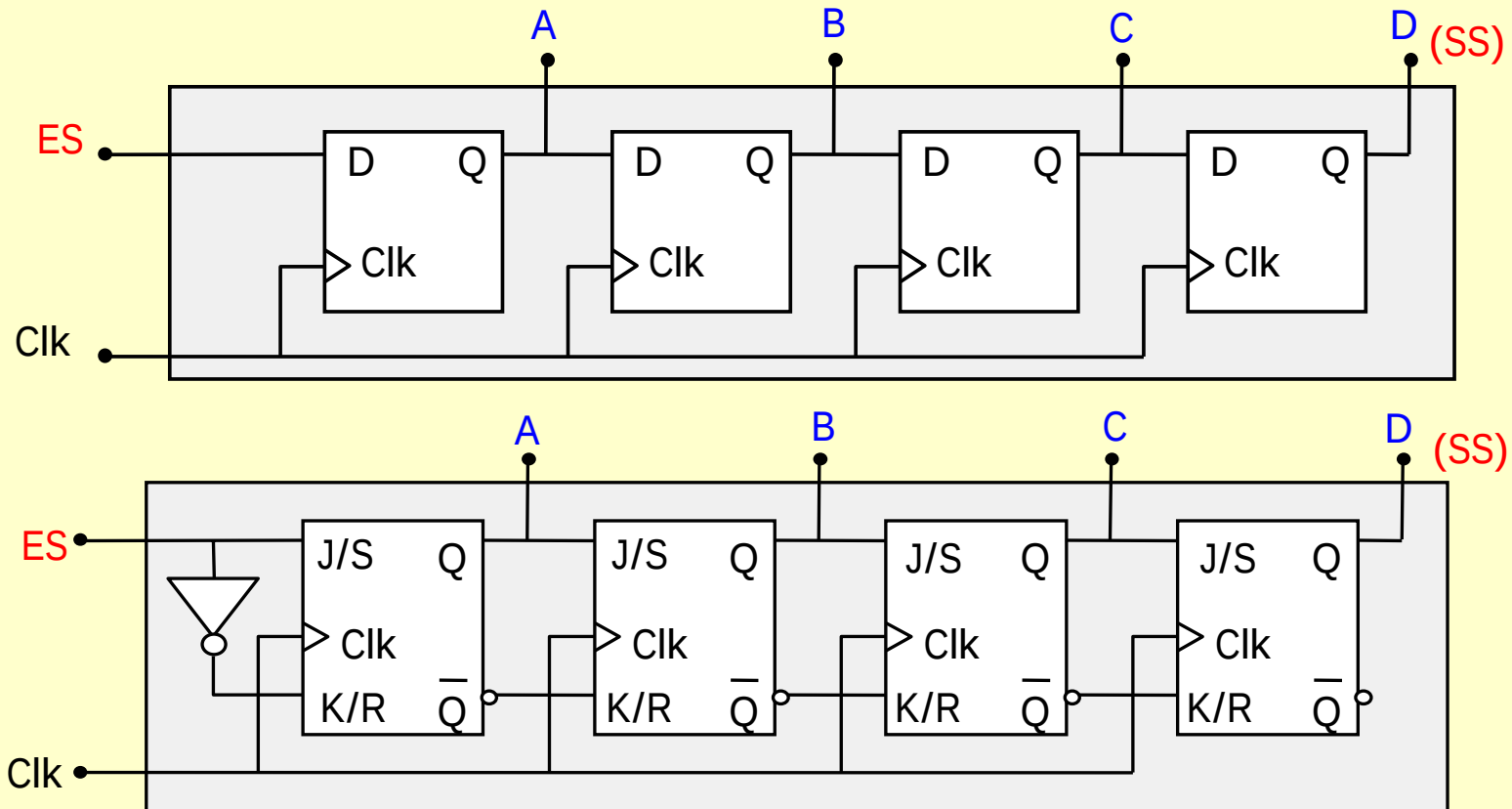
Exemple de registre commercialisé

SN54273, SN74LS273 ... J OR W PACKAGE
 SN74273 ... N PACKAGE
 SN74LS273 ... DW OR N PACKAGE
 (TOP VIEW)

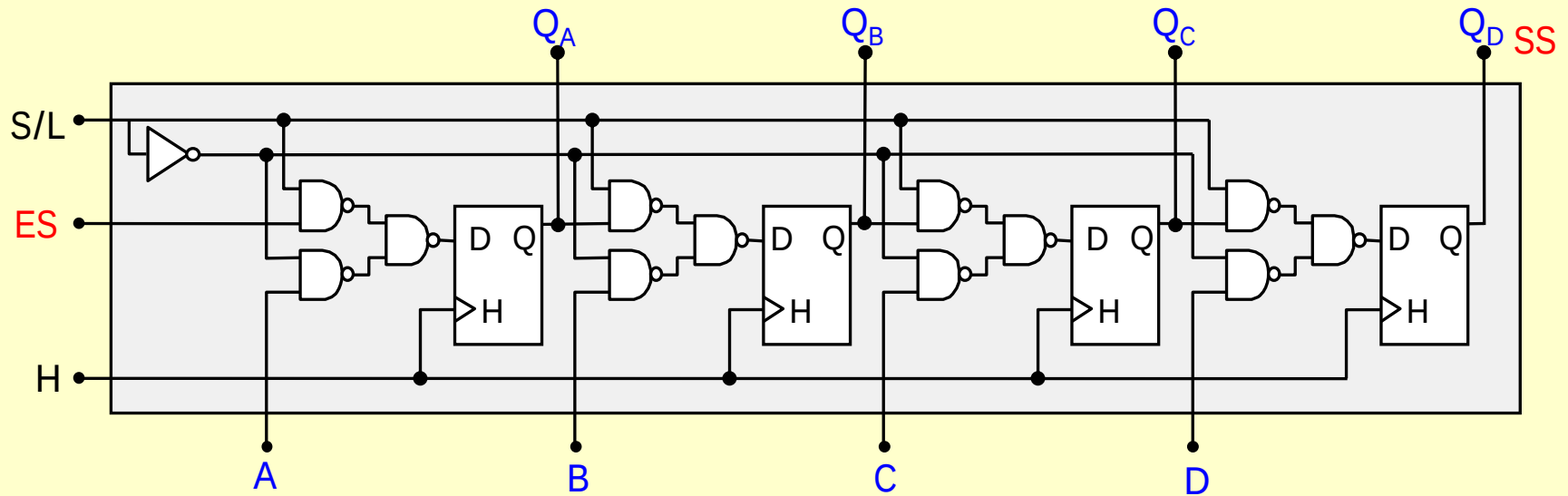


Registres à décalage

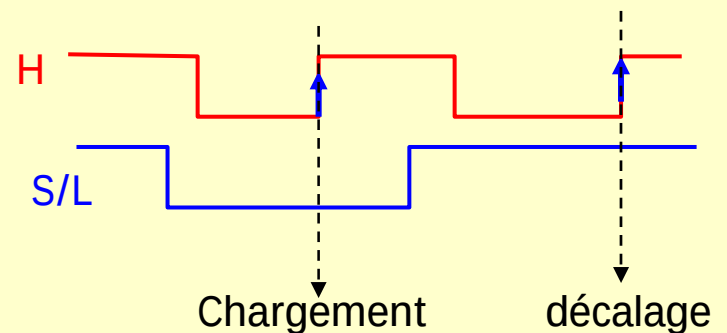
- A chaque coup d'horloge, le contenu d'une bascule est remplacé par celui de la bascule à sa gauche → tous les bits sont alors décalés vers la droite
- ES constitue l'entrée série,
- ABCD constituent les sorties parallèle,
- SS est la sorties série



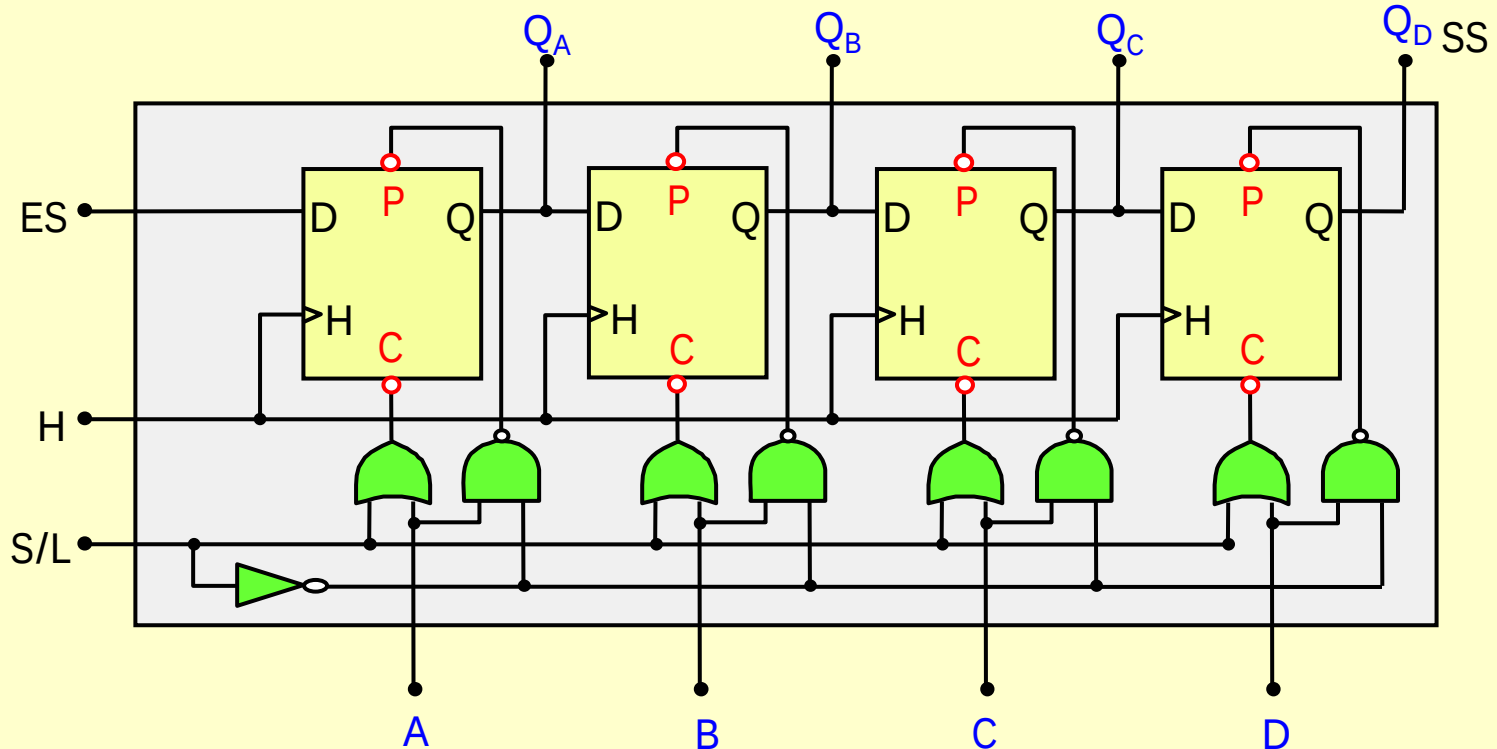
Registre à décalage avec chargement parallèle **synchrone**



- **ES** : Entrée séries
- **A B C D** : entrées parallèles
- $Q_A Q_B Q_C Q_D$: sorties parallèles
- **SS** : sorties série
- S/L : entrée de control :
 - S/L = 1 : décalage
 - S/L = 0 : chargement parallèle **synchrone** (au coup d'horloge)



Registre à décalage avec chargement parallèle asynchrone



$S/L = 0 \rightarrow$ chargement parallèle sans attendre le coup d'horloge

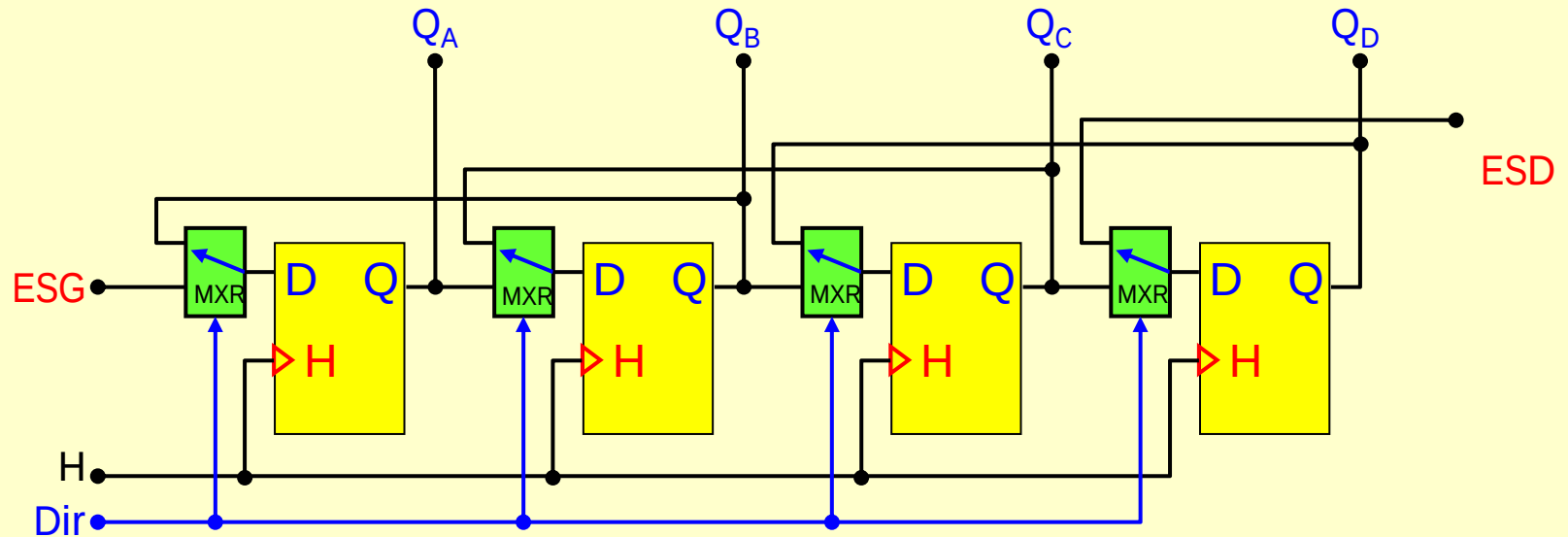
$S/L = 1 \Rightarrow C = P = 1 \Rightarrow$ bscules libres $\Rightarrow$ décalage

$S/L = 0 \Rightarrow C = A, P = \bar{A}$

Si $A = 0 \Rightarrow C = 0, P = 1$, r`az de la bascule $\Leftrightarrow$ chargement de A

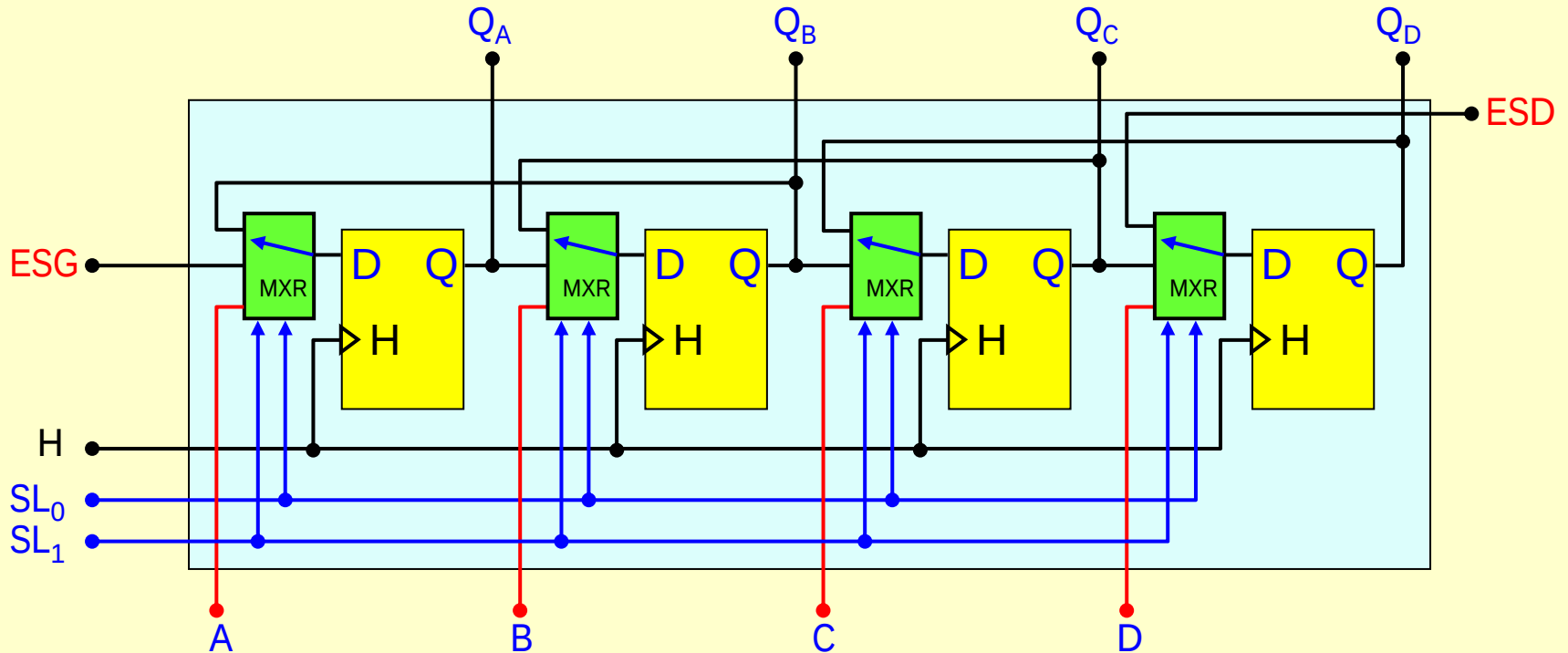
Si $A = 1 \Rightarrow C = 1, P = 0$, r`au de la bascule $\Leftrightarrow$ chargement de A

Registre à décalage bidirectionnel



- **ESG** : entrée série gauche
- **ESD** : entrée série droite
- $Dir = 0 \Rightarrow$ décalage à gauche
- $Dir = 1 \Rightarrow$ décalage à droite

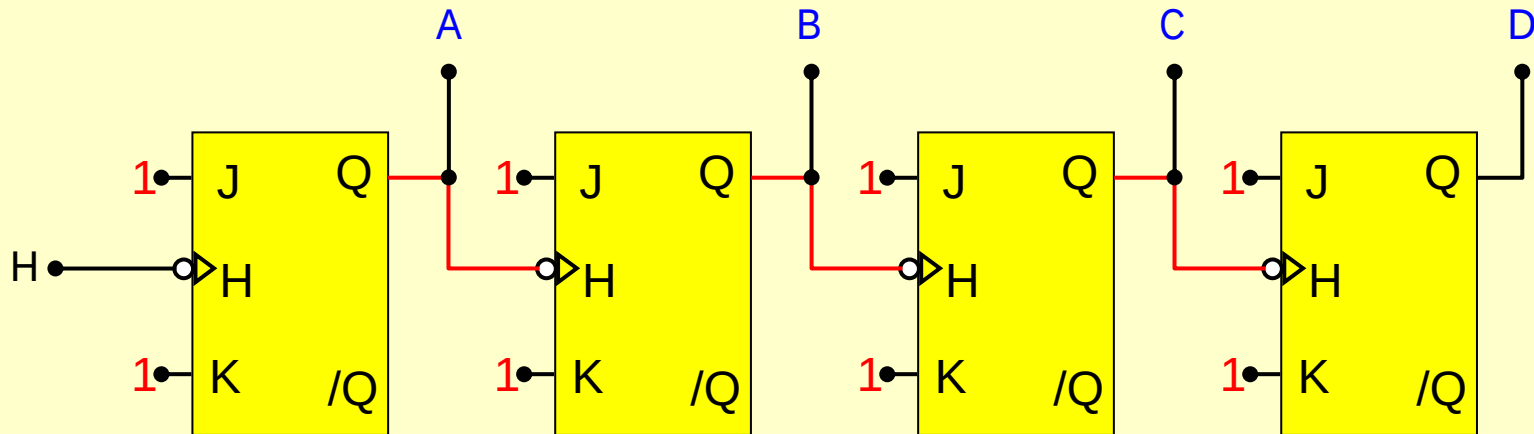
Registre à décalage Universel



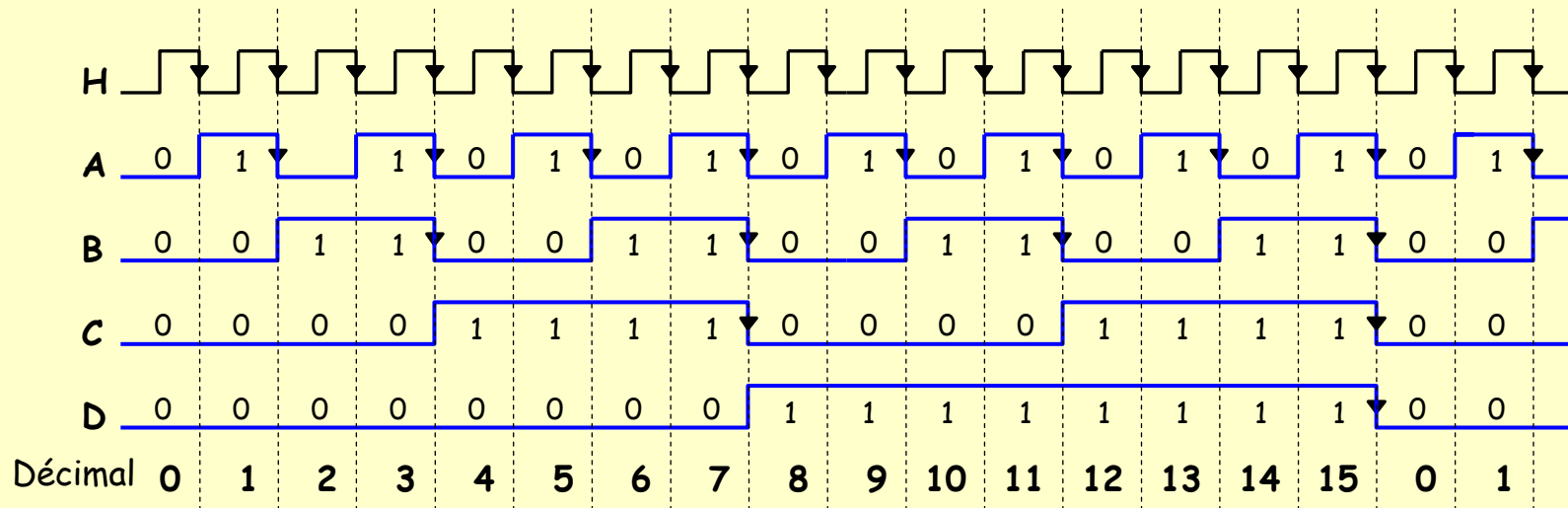
SL_1	SL_0	
0	0	Décalage à gauche
0	1	Décalage à droite
1	0	Chargement parallèle synchrone
1	1	Non utilisé

Les Compteurs Asynchrones

Compteur Asynchrone

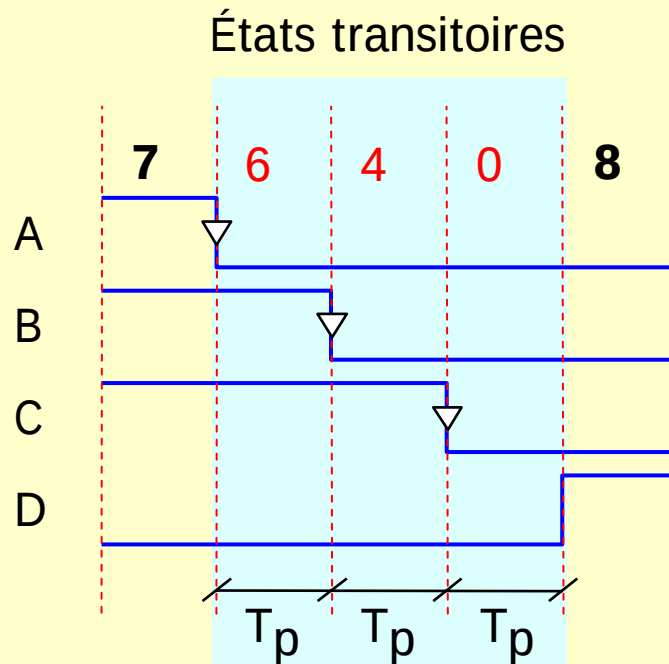


$J = K = 1 \rightarrow$ Basculement à chaque coup d'horloge

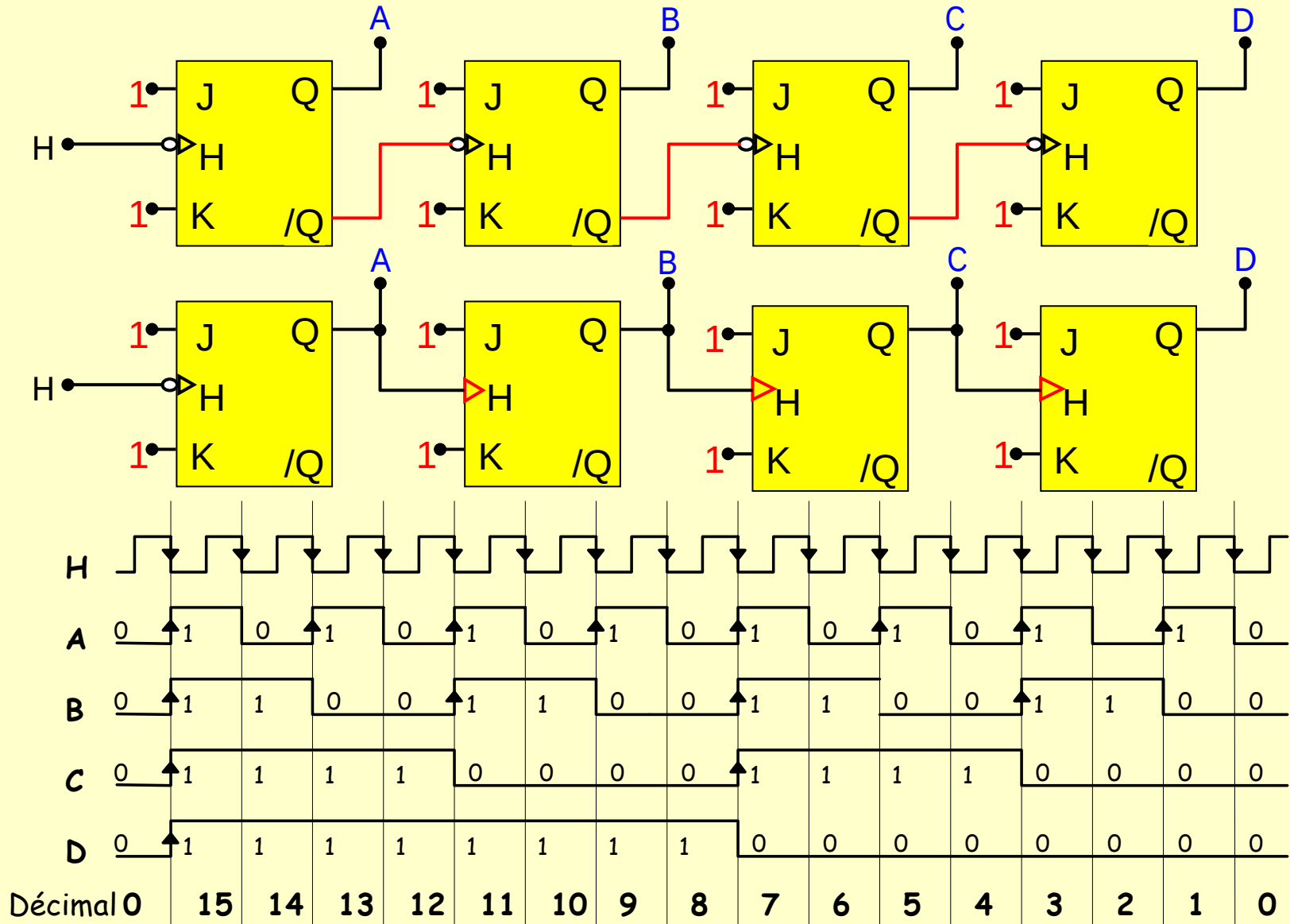


Pourquoi asynchrone ?

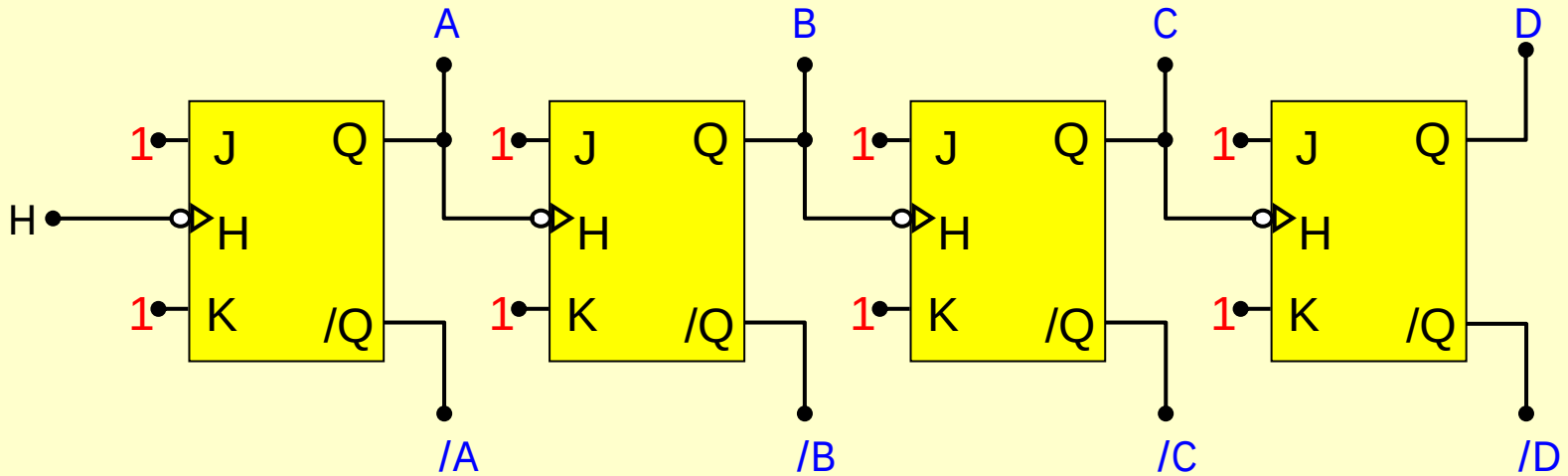
Les bascules ne changent pas toutes en même temps.
Chaque bascule change un temps de propagation après la
bascule précédente ce qui engendre des états transitoires



Décompacteur asynchrone

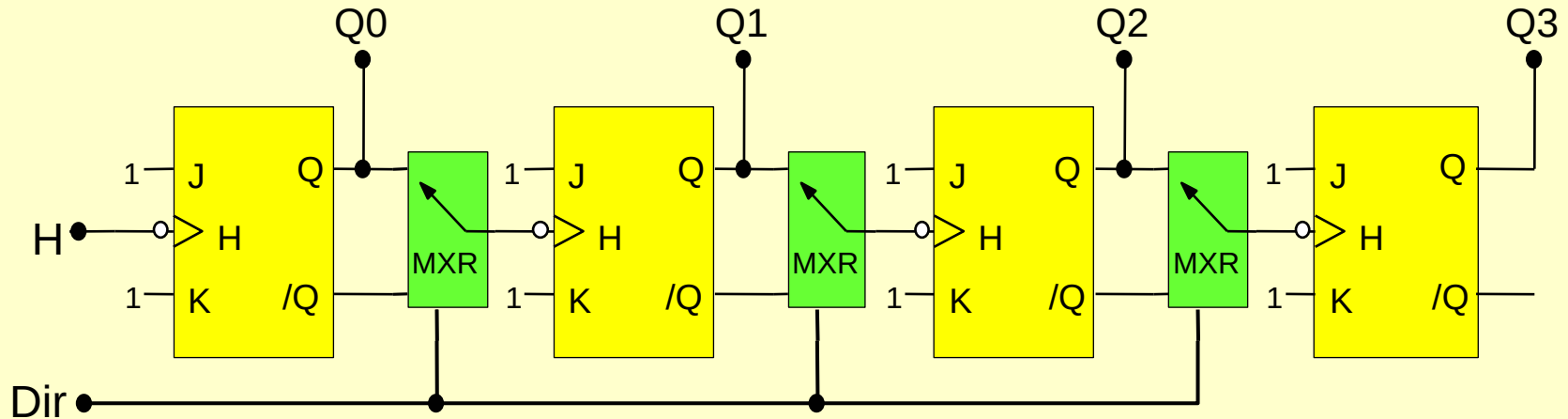


Décompteur , autre méthode



	D	C	B	A	/D	/C	/B	/A	
0	0	0	0	0	1	1	1	1	15
1	0	0	0	1	1	1	1	0	14
2	0	0	1	0	1	1	0	1	13
3	0	0	1	1	1	1	0	0	12
4	0	1	0	0	1	0	1	1	11
5	0	1	0	1	1	0	1	0	10
6	0	1	1	0	1	0	0	1	9
7	0	1	1	1	1	0	0	0	8
8	1	0	0	0	0	1	1	1	7
9	1	0	0	1	0	1	1	0	6
10	1	0	1	0	0	1	0	1	5
11	1	0	1	1	0	1	0	0	4
12	1	1	0	0	0	0	1	1	3
13	1	1	0	1	0	0	1	0	2
14	1	1	1	0	0	0	0	1	1
15	1	1	1	1	0	0	0	0	0

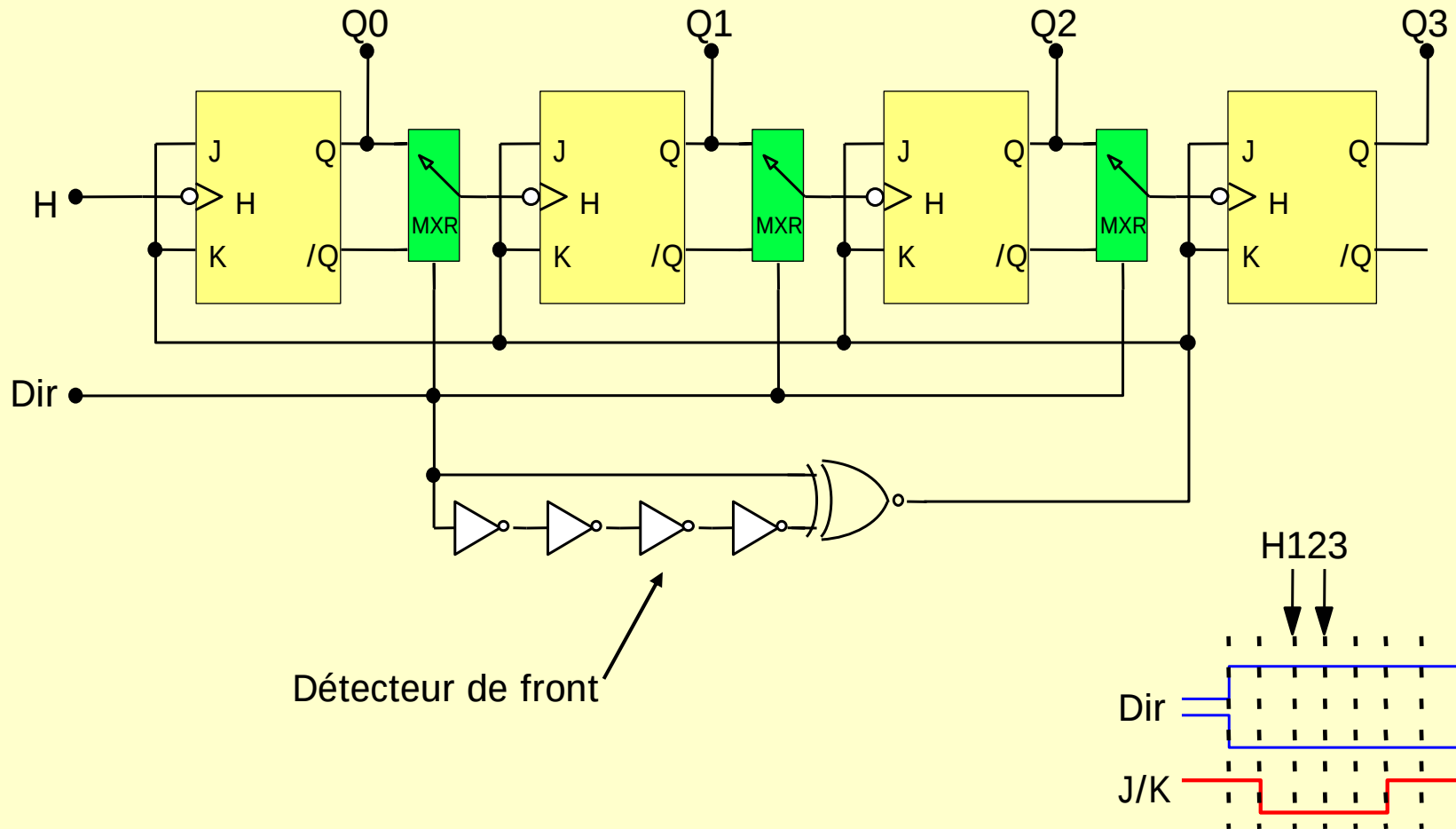
Compteur décompteur Asynchrone



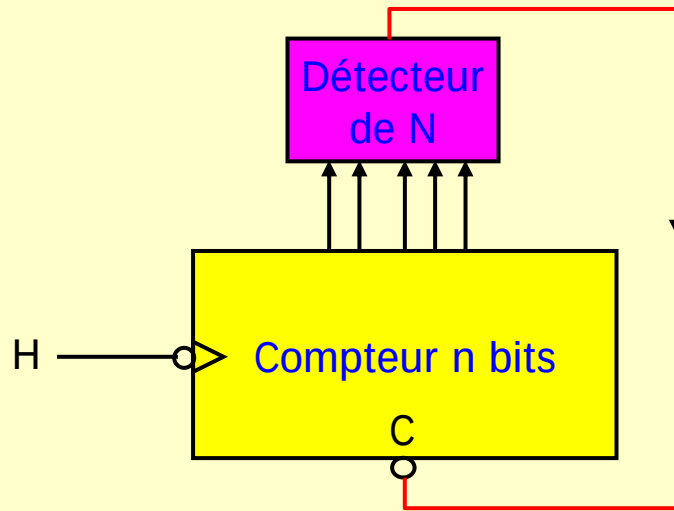
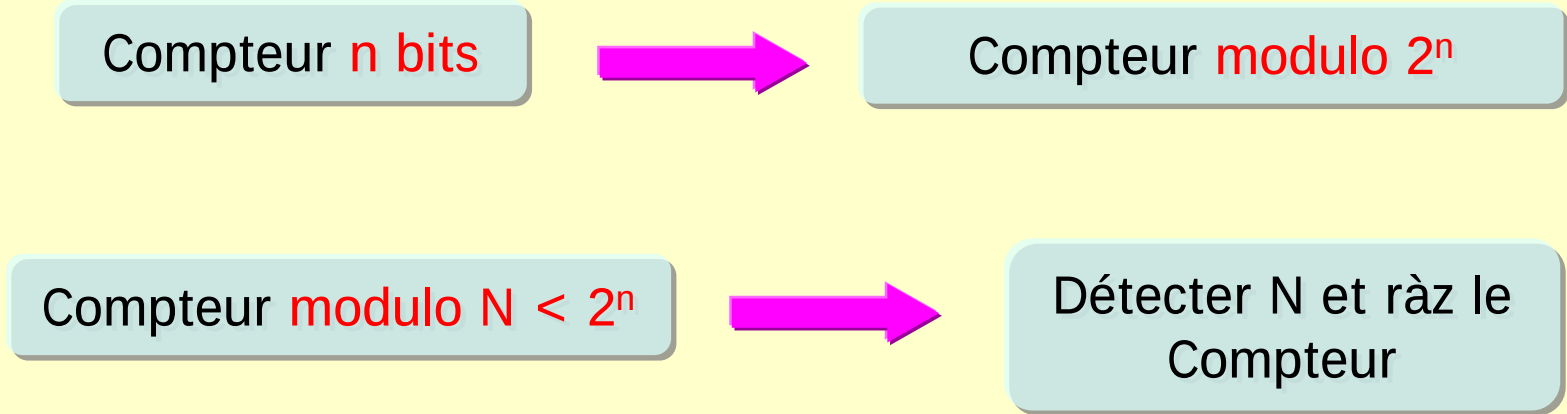
Au changement de direction, si $Q=1$, le passage à $/Q$ génère un front descendant provoquant un changement d'état de la bascule suivante ce qui peut être indésirable.

Compteur décompteur Asynchrone Amélioré

L'idée est de mettre les bascules en mode mémoire ($J = K = 0$) pendant le basculement des horloges entre Q et /Q

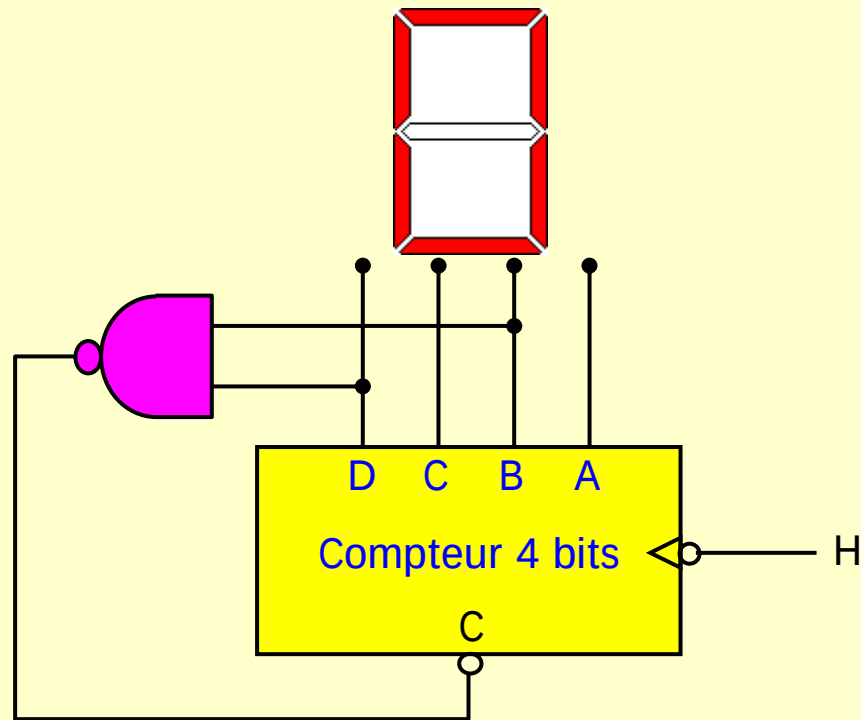


Compteur modulo N



Compteur modulo 10 : Décade

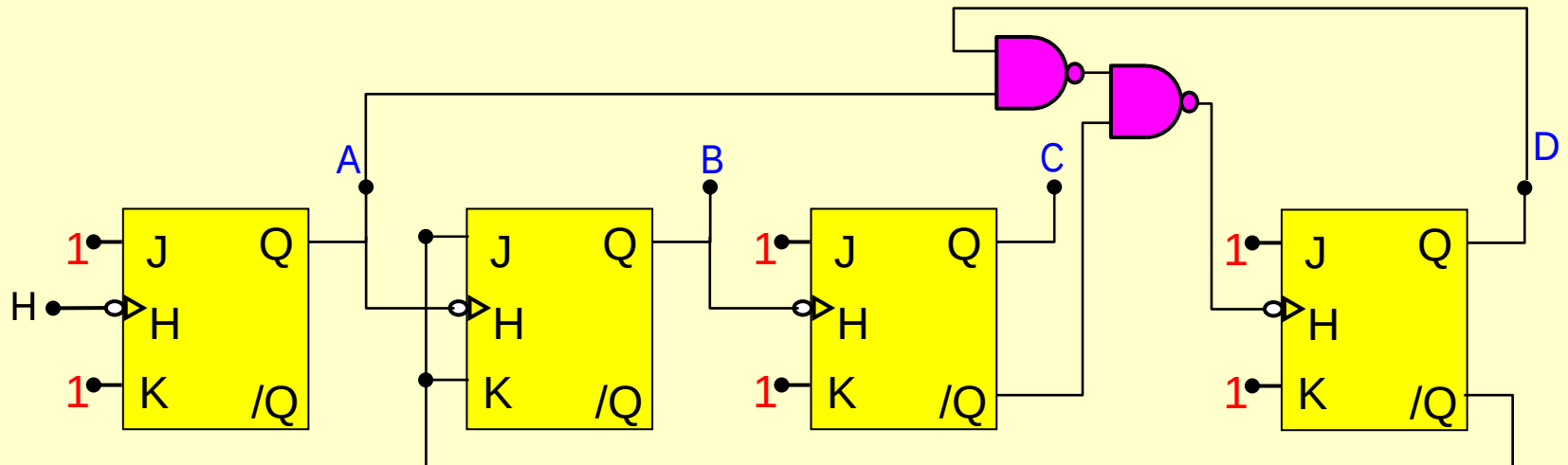
On détecte le 10 = **1 0 1 0** et on s'en sert pour remettre le compteur à 0



Réalisation d'une décade par action sur les entrées JK

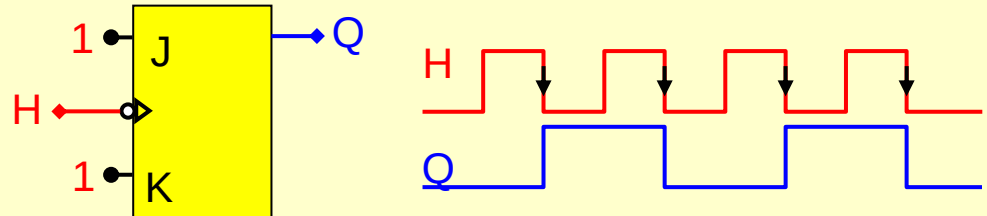
	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
0	0	0	0	0

- Le bit A change d'état à chaque impulsion d'horloge, on utilise une bascule avec $H_A = \text{Clk}$, $J=K=1$
- Le bit B change à chaque front descendant de A sauf pour la transition $9 \rightarrow 0$, dans ce cas on va forcer la bascule à l'état mémoire $\rightarrow H_B = A$, $J = K = /D$
- Le bit C ne pose pas de problème, il change à chaque front descendant de B $\rightarrow H_C = B$, $J=K=B$
- L'horloge de la bascule D ne peut être commandée par le bit C seul car celui-ci n'a pas de transition pour le passage $9 \rightarrow 10$, il faut trouver un signal qui a un front descendant après 7 et après 9, le signal $C+DA$ présente cette condition

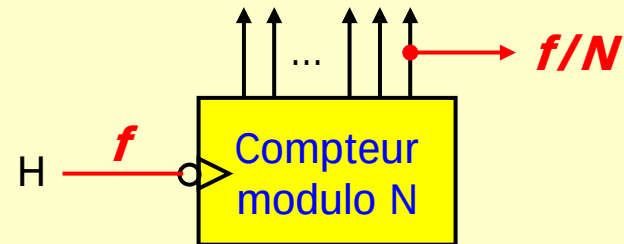


Division de fréquence

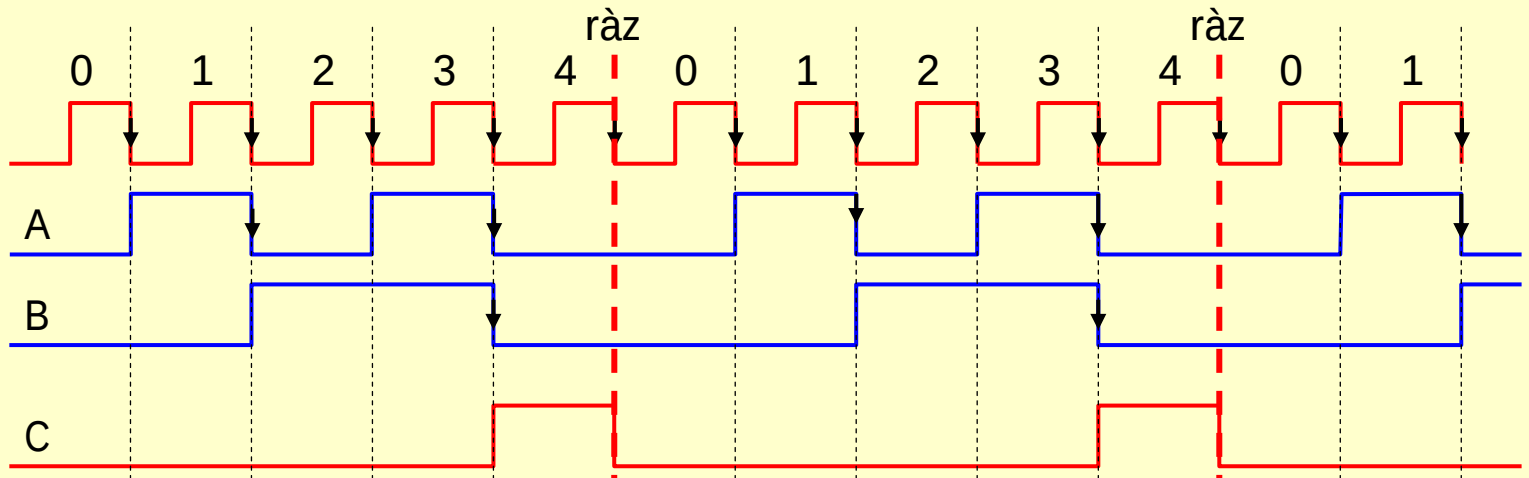
Une bascule divise la fréquence par 2



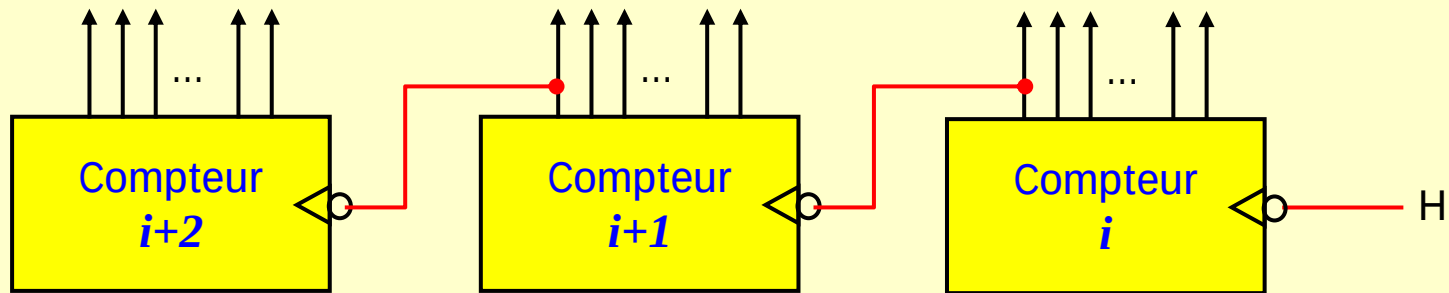
- Une compteur modulo N divise la fréquence par N
- Fréquence du MSB = fréquence de l'horloge / N



Compteur modulo 5



Mise en cascade des compteur asynchrones

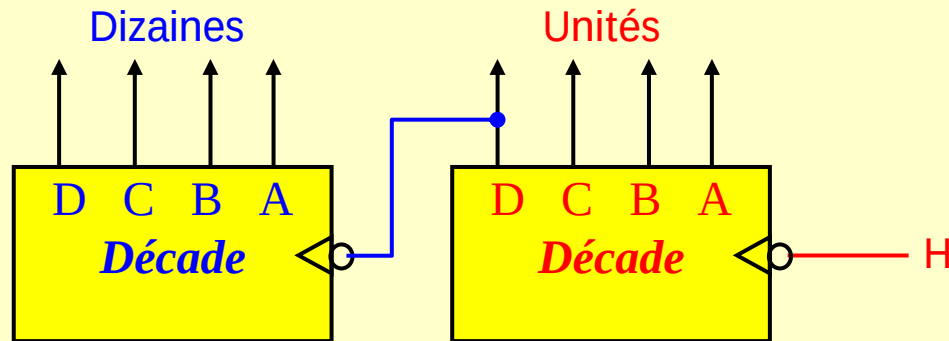


- Le MSB du compteur i est appliqué à l'horloge du compteur $i+1$ (suivant)
- Chaque fois qu'un compteur repasse à 0, le compteur suivant est incrémenté

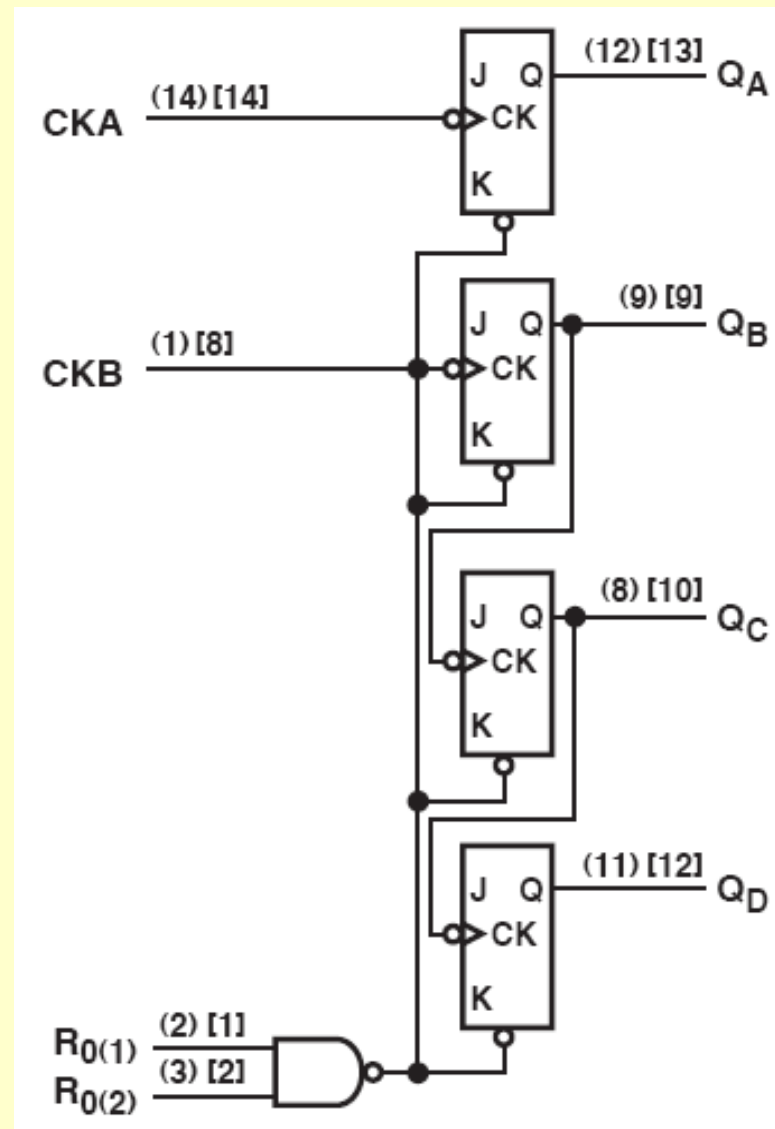
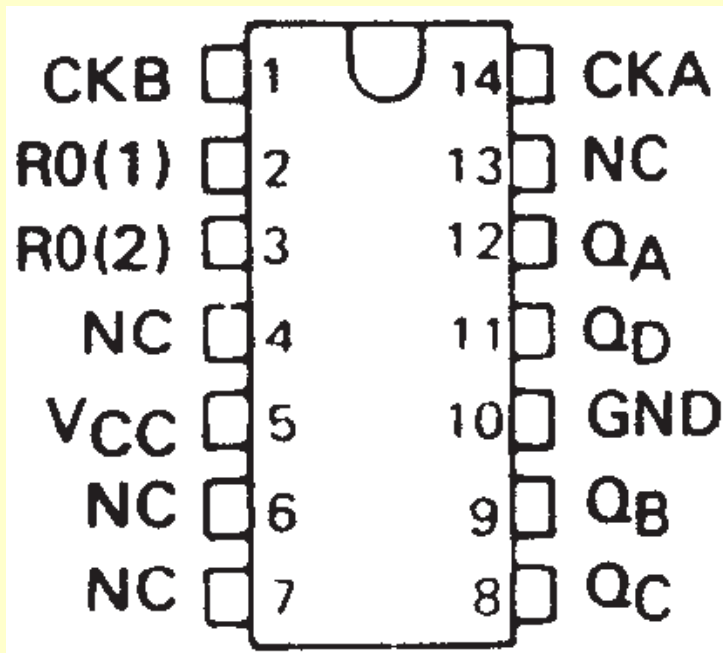
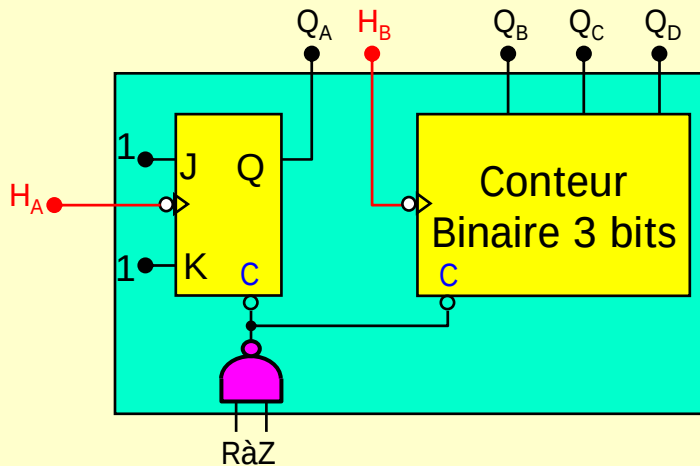
Compteur BCD

On obtient un compteur BCD en cascadeant des compteurs modulo 10

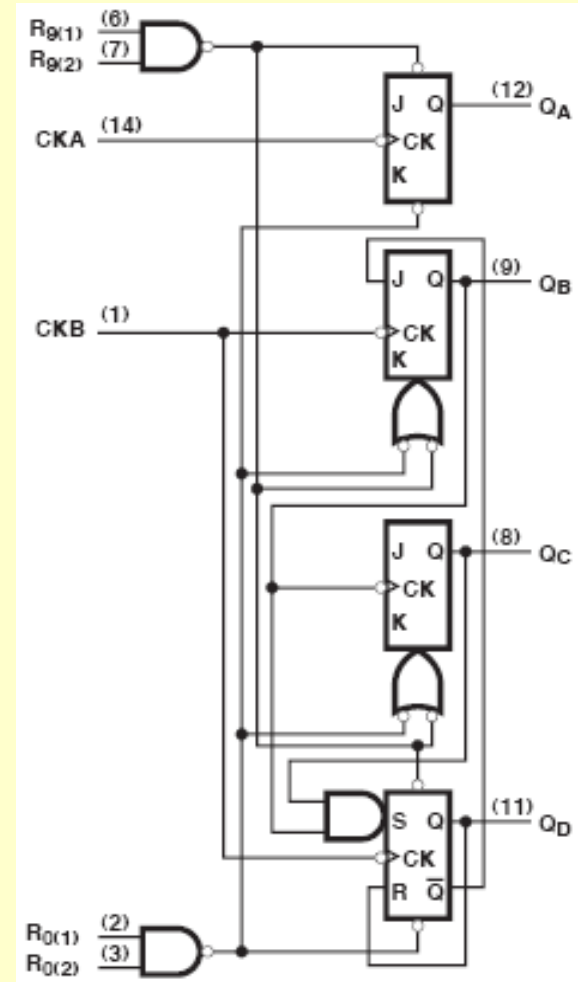
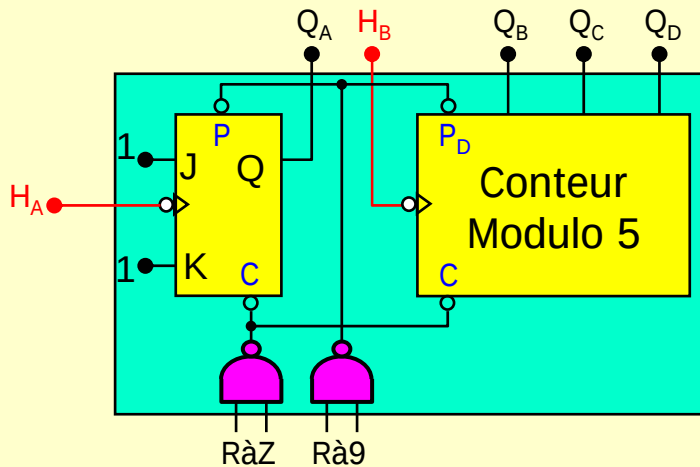
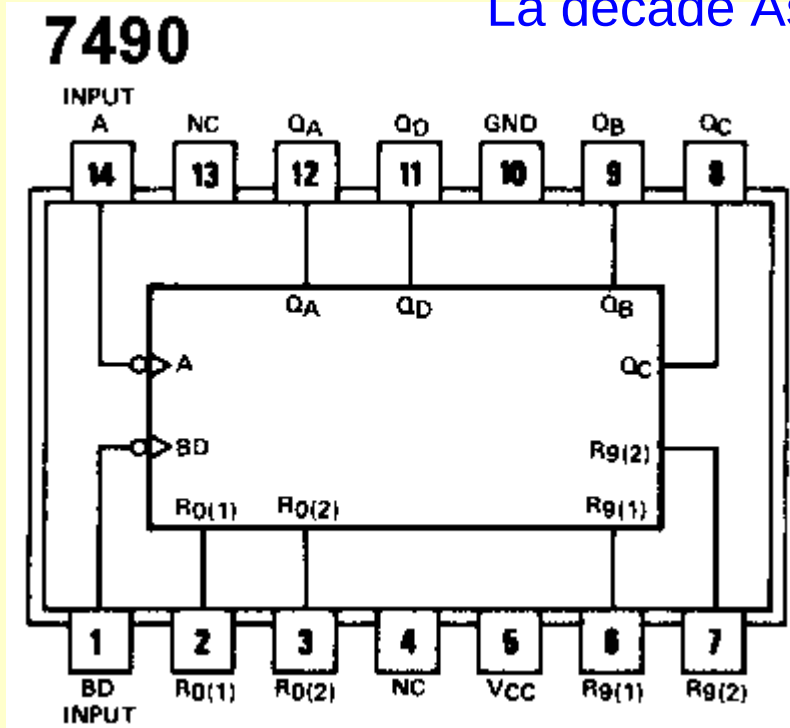
0	0
0	1
0	2
0	3
0	4
0	5
0	6
0	7
0	8
0	9
1	0
1	1
1	2
1	3
1	4
1	5
1	6
1	7
1	8
1	9
2	0
2	1
...	...



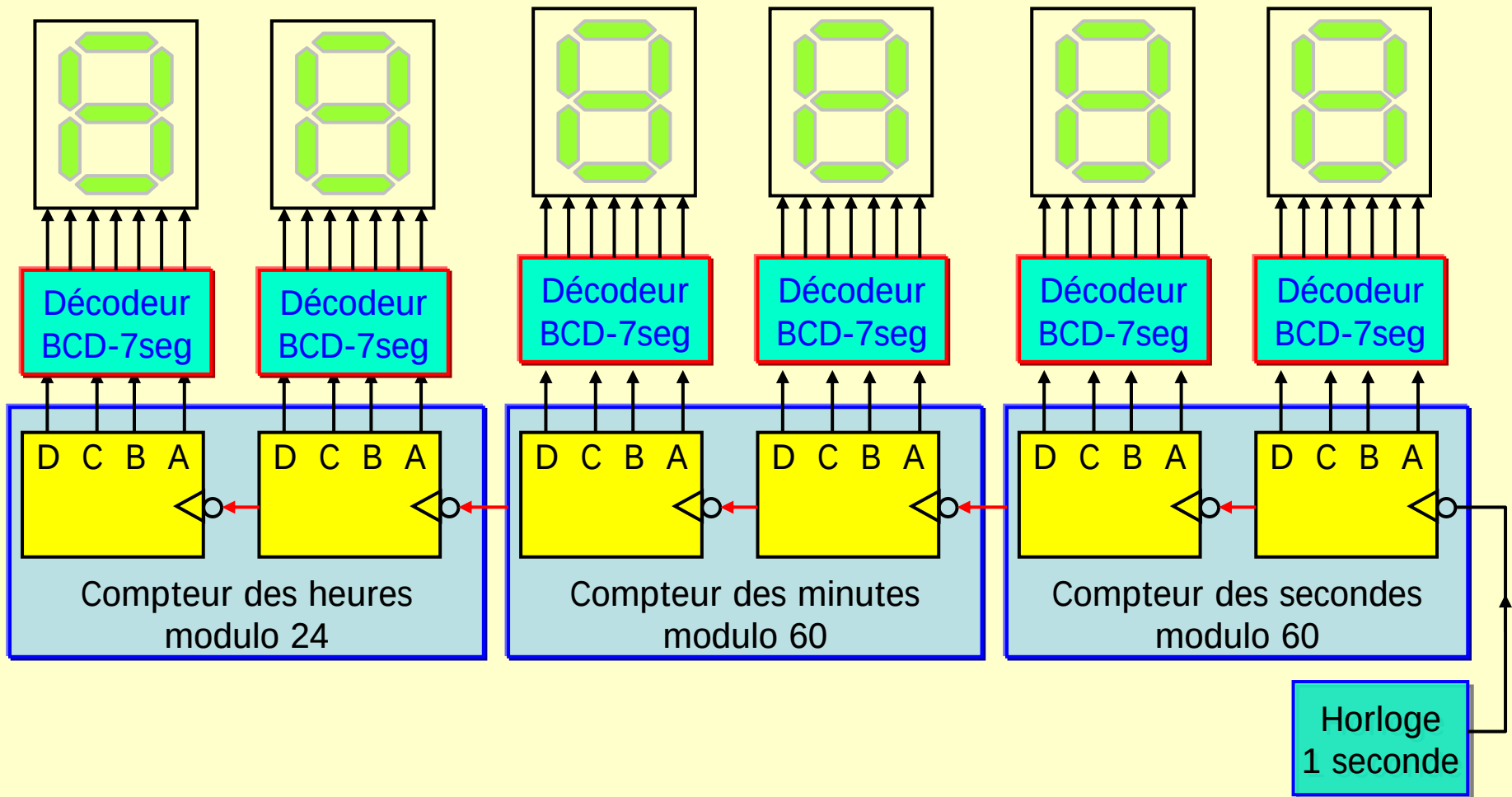
Le compteur Binaire 74LS93



La décade Asynchrone 74LS90



Horloge secondes , minutes Heures

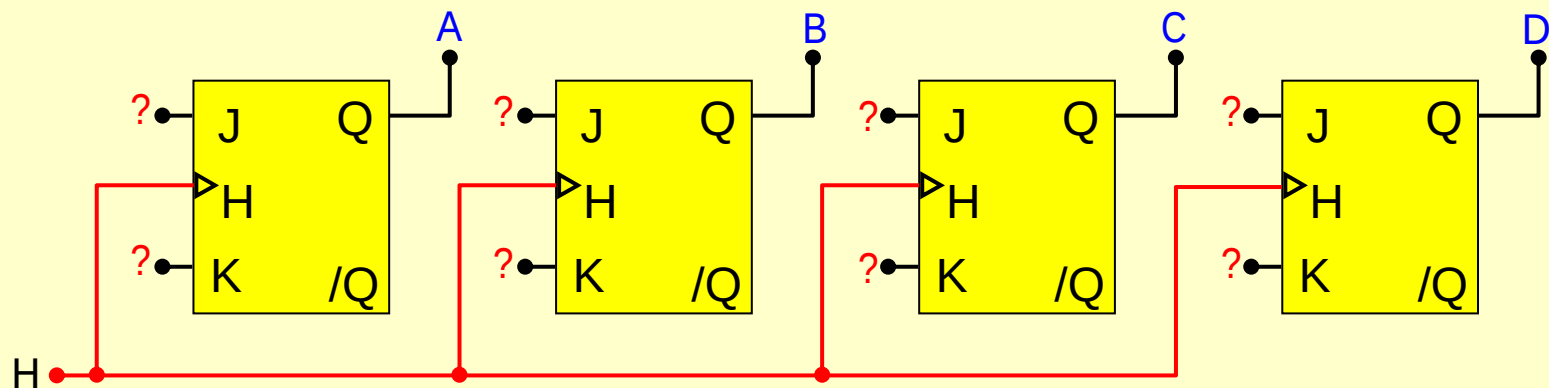


Les Compteurs Synchrones

Le principe des compteurs synchrones

Toutes les bascules reçoivent la **même horloge**

Au coup d'horloge, toutes les bascules changent en même temps → **Pas d'états transitoires**



Que doit on faire des **J** et des **K** pour que ça compte ?

Compteur synchrone suite

- ❑ On connaît la séquence de comptage, donc,
- ❑ Au coup d'horloge, on sait pour chaque bascule si elle doit changer d'état ou non,
- ❑ On en déduit le J et le K qu'il faut lui appliquer
- ❑ Pour ça, on utilise la table des transitions

H	Q_n	Q_{n+1}	J	K	J	K
	0	→ 0	0 0	0 1	0	x
	0	→ 1	1 1	1 0	1	x
	1	→ 0	1 0	1 1	x	1
	1	→ 1	0 1	0 0	x	0

Synthèse d'un compteur Synchrone 4 bits

	D	C	B	A	J _A	K _A	J _B	K _B	J _C	K _C	J _D	K _D
0	0	0	0	0	1	x	0	x	0	x	0	x
1	0	0	0	1	x	1	1	x	0	x	0	x
2	0	0	1	0	1	x	x	0	0	x	0	x
3	0	0	1	1	x	1	x	1	1	x	0	x
4	0	1	0	0	1	x	0	x	x	0	0	x
5	0	1	0	1	x	1	1	x	x	0	0	x
6	0	1	1	0	1	x	x	0	x	0	0	x
7	0	1	1	1	x	1	x	1	x	1	1	x
8	1	0	0	0	1	x	0	x	0	x	x	0
9	1	0	0	1	x	1	1	x	0	x	x	0
10	1	0	1	0	1	x	x	0	0	x	x	0
11	1	0	1	1	x	1	x	1	1	x	x	0
12	1	1	0	0	1	x	0	x	x	0	x	0
13	1	1	0	1	x	1	1	x	x	0	x	0
14	1	1	1	0	1	x	x	0	x	0	x	0
15	1	1	1	1	x	1	x	1	x	1	x	1

Synthèse d'un compteur Synchronisé 4 bits (suite)

On utilisant les tables de Karnaugh, on obtient :

$$J_A = 1$$

$$J_B = A$$

$$J_C = AB$$

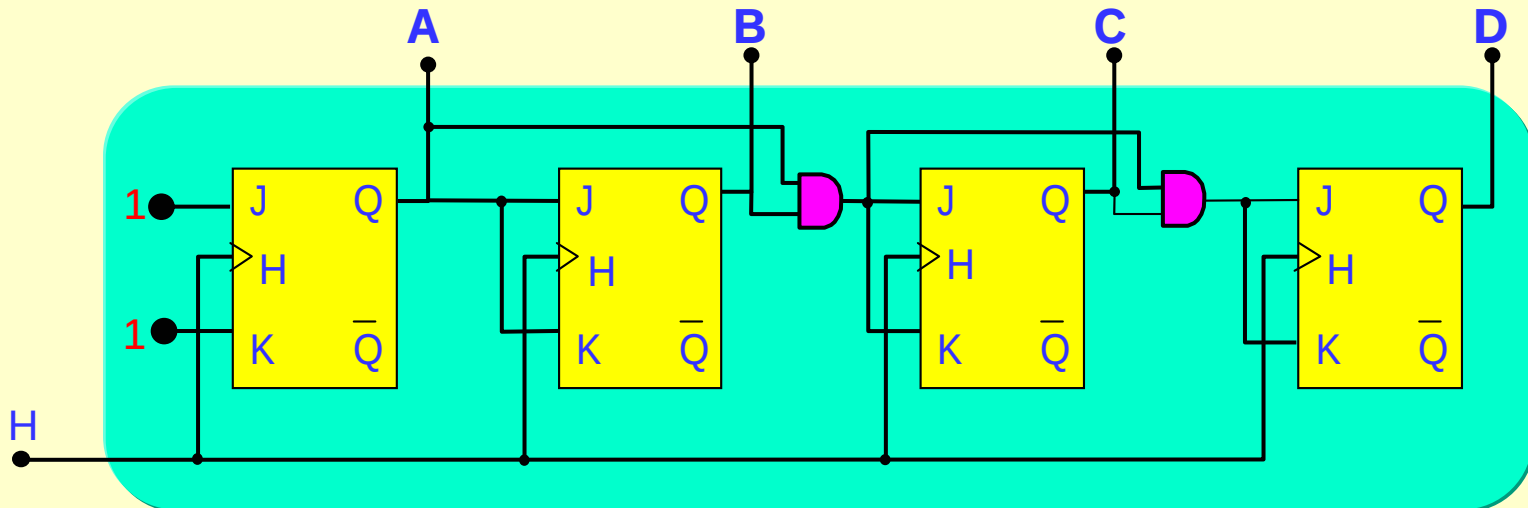
$$J_D = ABC$$

$$K_A = 1$$

$$K_B = A$$

$$K_C = AB$$

$$K_D = ABC$$



Synthèse d'un décompteur Synchrone 4 bits

Avec une étude similaire, on obtient :

$$J_A = 1$$

$$J_B = \bar{A}$$

$$J_C = \bar{A} \bar{B}$$

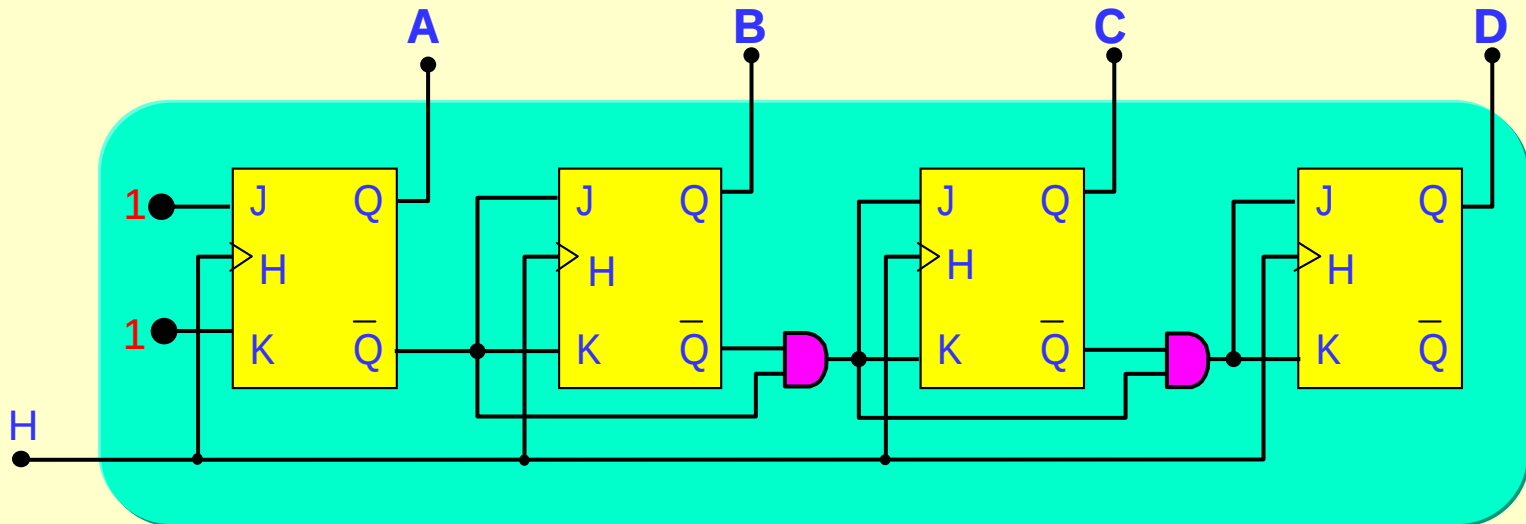
$$J_D = \bar{A} \bar{B} \bar{C}$$

$$K_A = 1$$

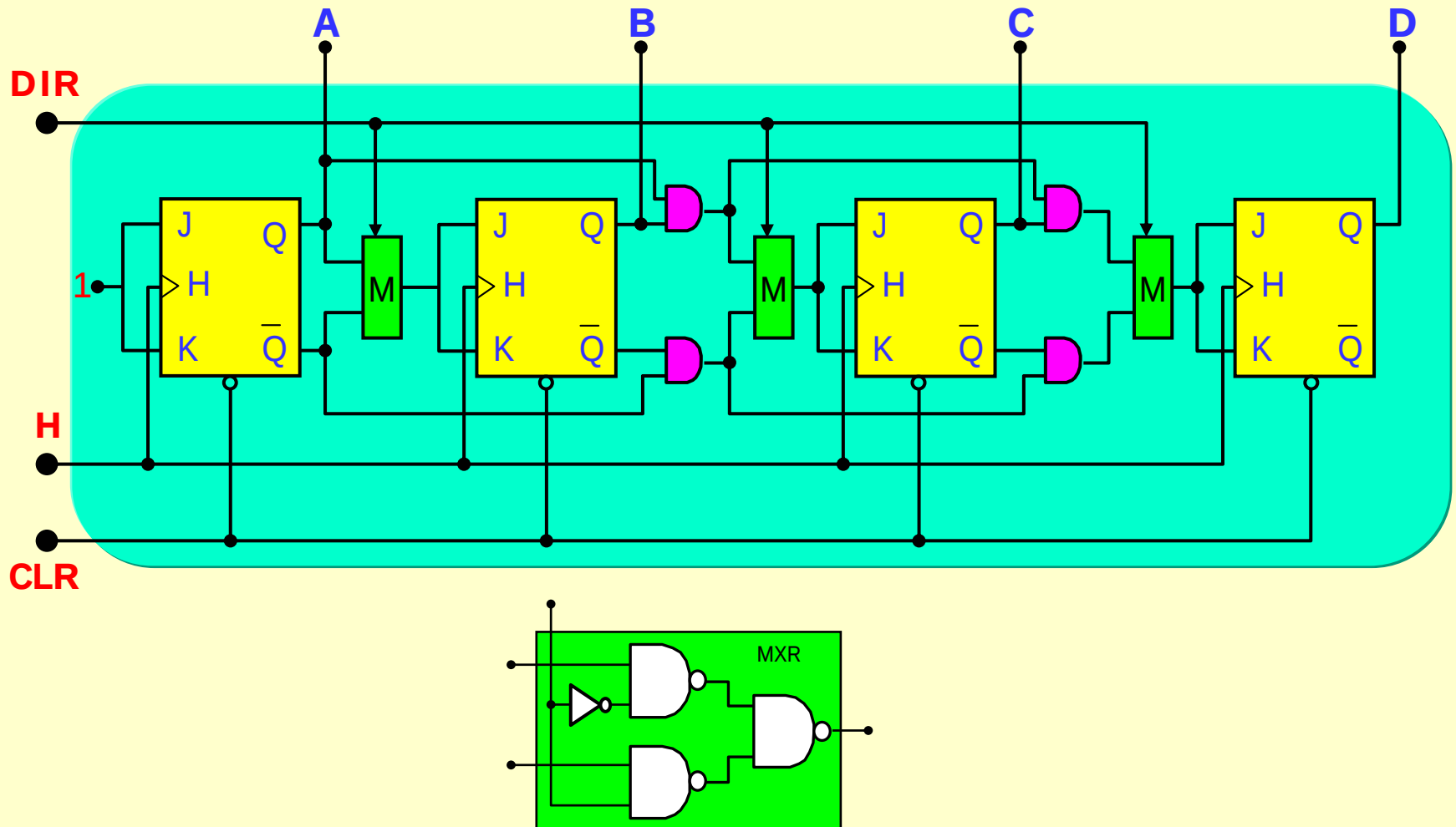
$$K_B = \bar{A}$$

$$K_C = \bar{A} \bar{B}$$

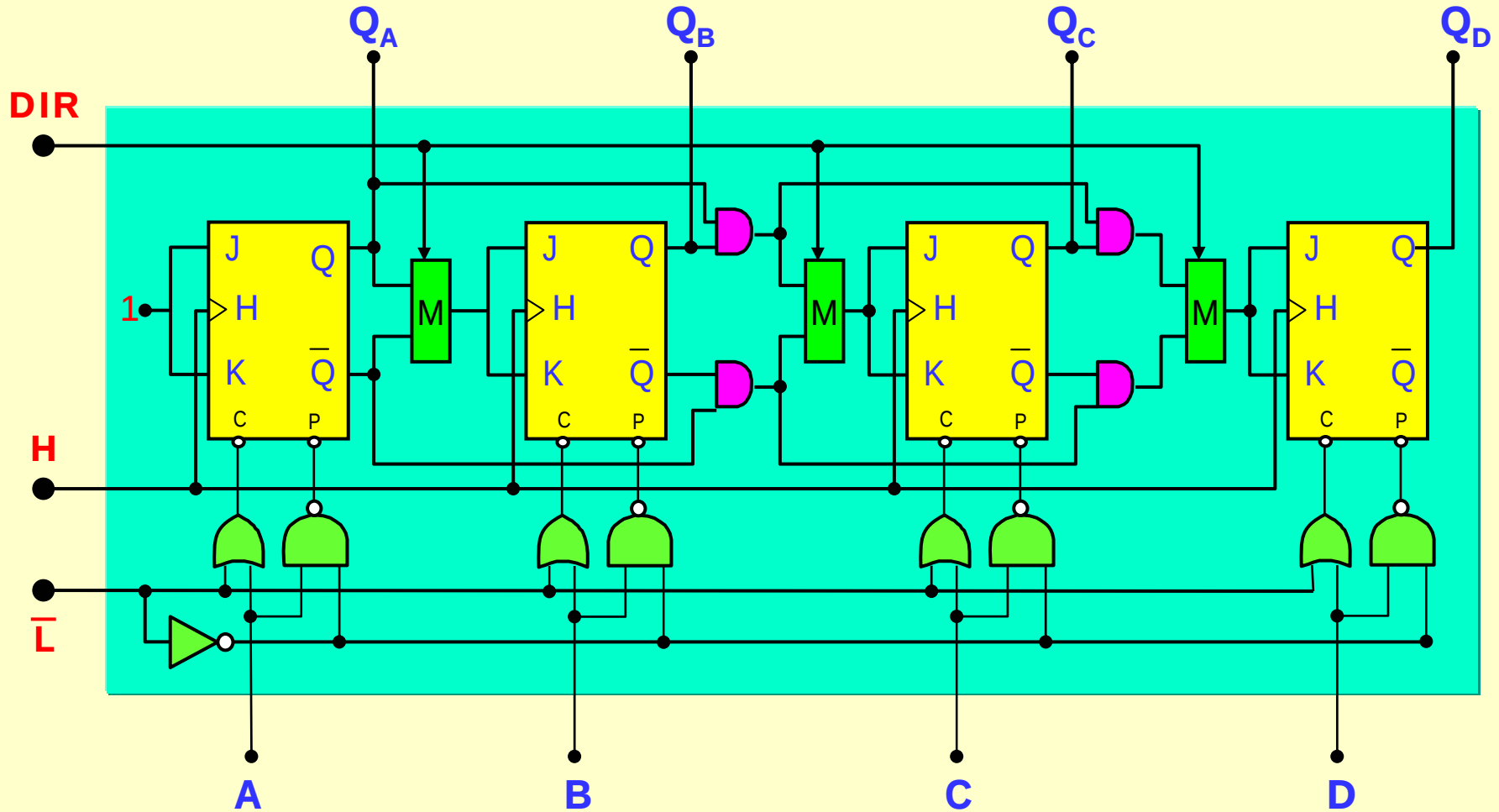
$$K_D = \bar{A} \bar{B} \bar{C}$$



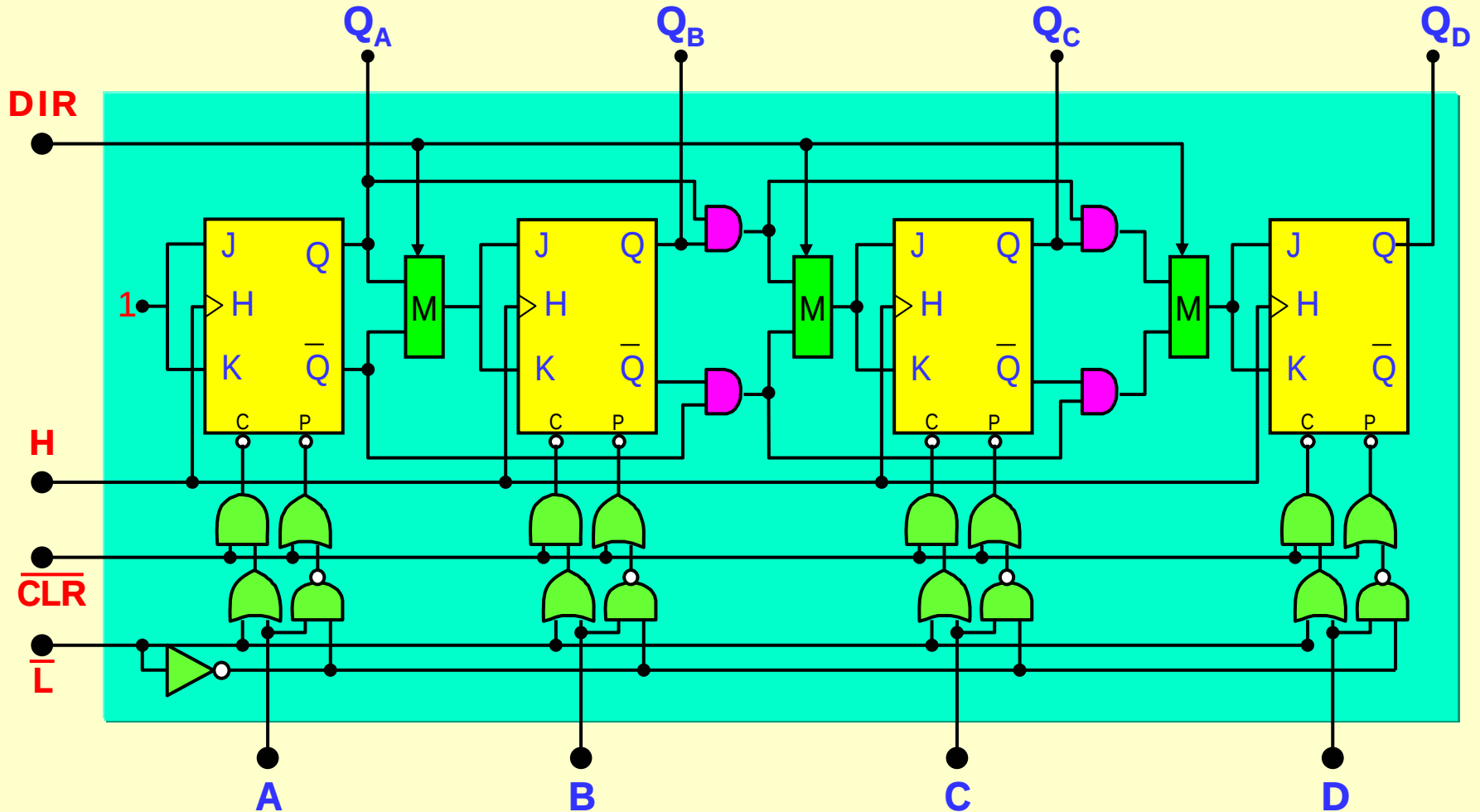
Compteur décompteur Synchrones



Compteur décompteur Synchrone avec chargement parallèle



Compteur décompteur Synchronisé Universel



Synthèse d'une **décade** Synchrones

	D	C	B	A	J _A	K _A	J _B	K _B	J _C	K _C	J _D	K _D
0	0	0	0	0	1	x	0	x	0	x	0	x
1	0	0	0	1	x	1	1	x	0	x	0	x
2	0	0	1	0	1	x	x	0	0	x	0	x
3	0	0	1	1	x	1	x	1	1	x	0	x
4	0	1	0	0	1	x	0	x	x	0	0	x
5	0	1	0	1	x	1	1	x	x	0	0	x
6	0	1	1	0	1	x	x	0	x	0	0	x
7	0	1	1	1	x	1	x	1	x	1	1	x
8	1	0	0	0	1	x	0	x	0	x	x	0
9	1	0	0	1	x	1	1	x	0	x	x	0

$$J_A = 1$$

$$K_A = 1$$

$$J_B = A\bar{D}$$

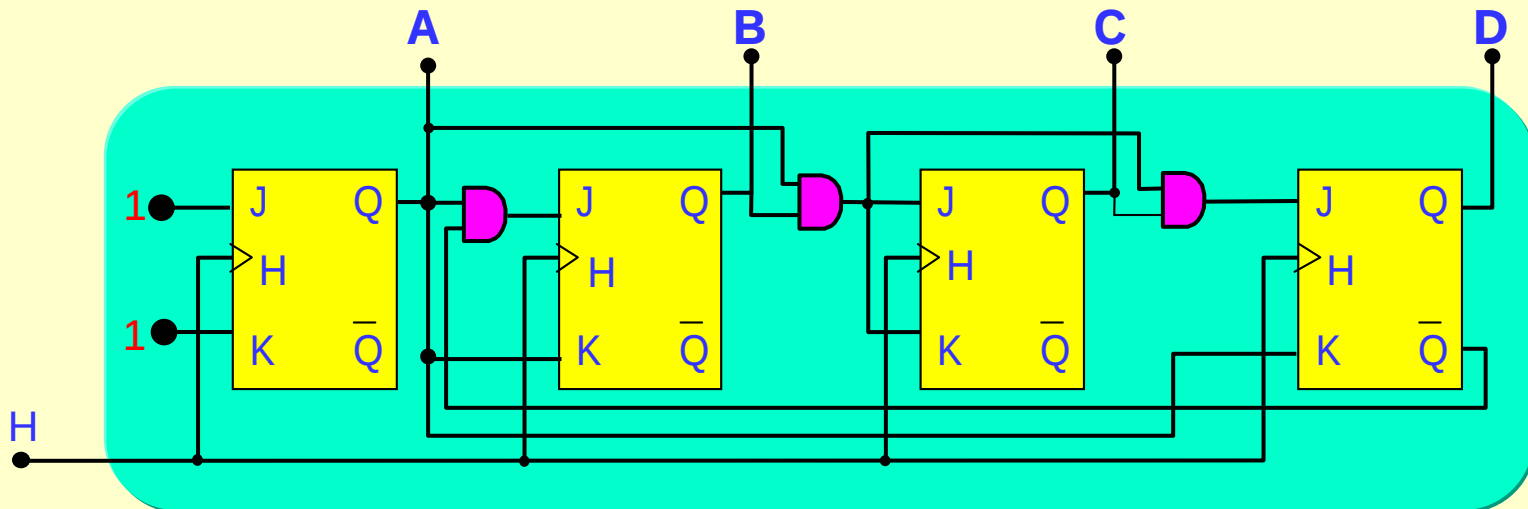
$$K_B = A$$

$$J_C = AB$$

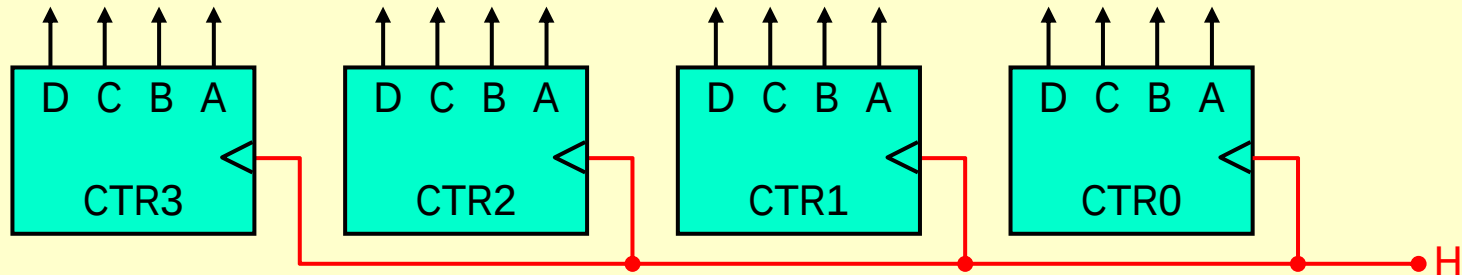
$$K_C = AB$$

$$J_D = ABC$$

$$K_D = A$$



Mise en cascade des compteurs synchrones



La mise en cascade doit être **synchrone**. Tous les compteurs doivent recevoir la **même Horloge**

Dans ces conditions, tous les compteurs fonctionneront simultanément et on n'aura pas le comptage désiré.

Il faut qu'un compteur ne s'incrémente que pendant le débordement du compteur précédent.

Mise en cascade des compteurs synchrones (2)

On va rajouter à chaque compteur une entrée de **validation V** et une sortie de **retenue R**

L'**entrée** de validation V permettra de le contrôler :
 $V=1 \rightarrow \text{Comptage}$, **$V=0 \rightarrow \text{arrêt}$**

La **sortie** de retenue R passe à 1 pour indiquer que le compteur est arrivé en fin de cycle.

- ☐ Compteur 4 bits, **$N=15 \rightarrow R=1$, $N \neq 15 \rightarrow R=0$**
- ☐ Compteur par 10, **$N=9 \rightarrow R=1$, $N \neq 9 \rightarrow R=0$**

Compteur Synchrone avec E/S de cascading

