

Remise à niveau en électronique - Documents

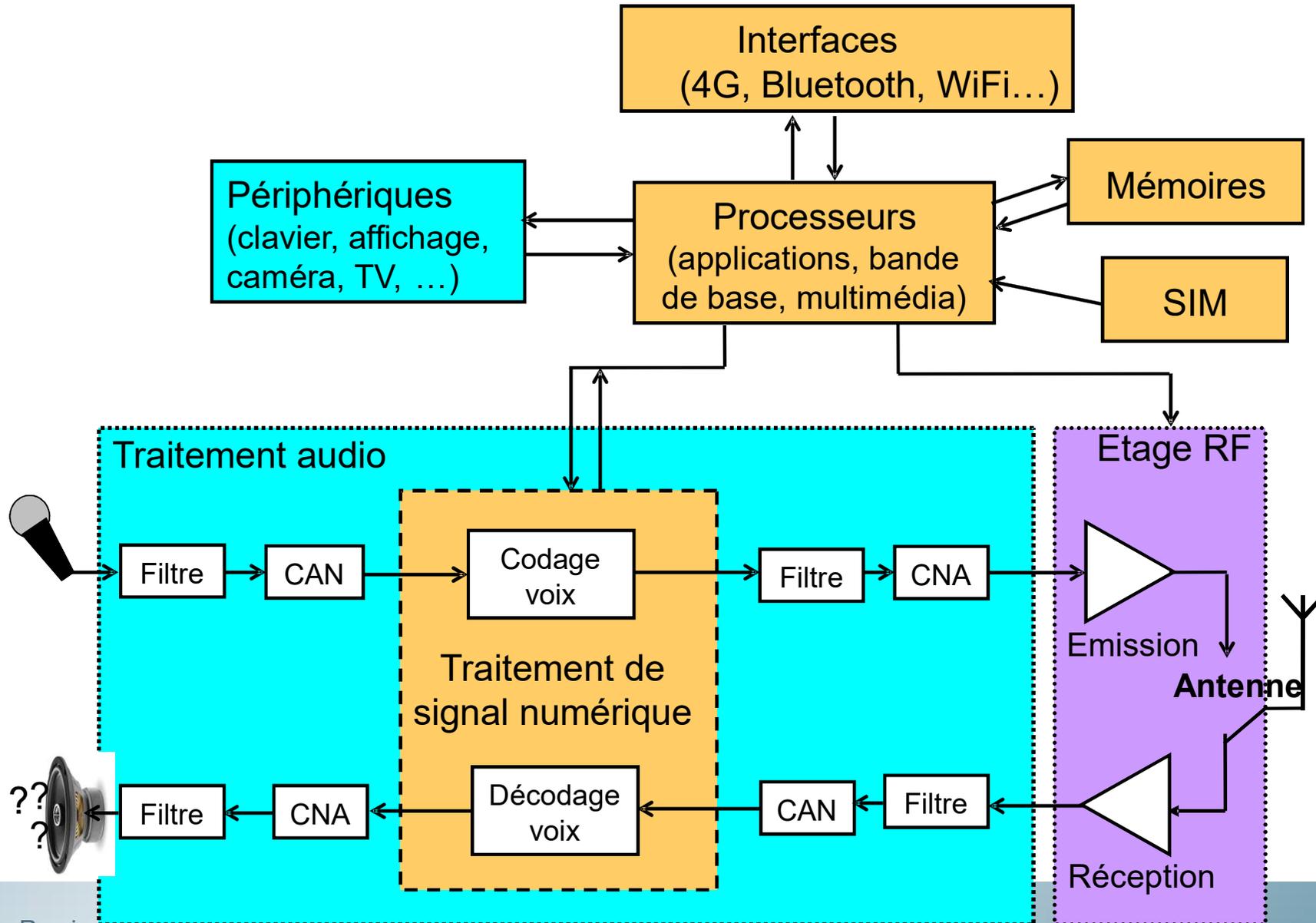
Alexandre BOYER

Alexandre.boyer@insa-toulouse.fr

www.alexandre-boyer.fr

Septembre 2022

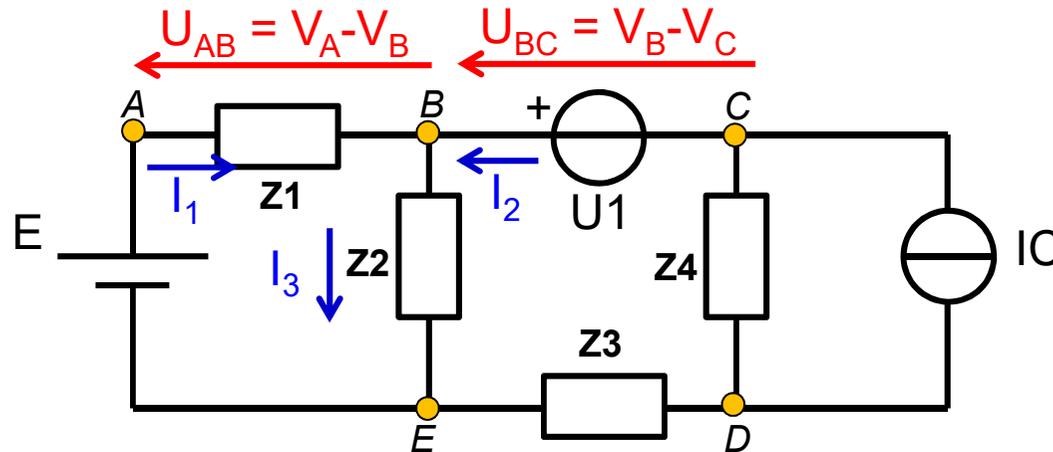
Electronique analogique vs. numérique



- 1. Tension, courant, impédance**
- 2. Régime harmonique et systèmes linéaires**
- 3. Filtre analogique**
- 4. Conversion analogique-numérique**
- 5. Circuits numériques – Portes de base**
- 6. Circuits numériques – structures générales**
- 7. Circuits numériques – Caractéristiques électriques**
- 8. Circuits numériques – Fonctions de base (registre, compteur, mémoires)**

1. Tension, courant, impédance

Lois élémentaires pour l'analyse des circuits

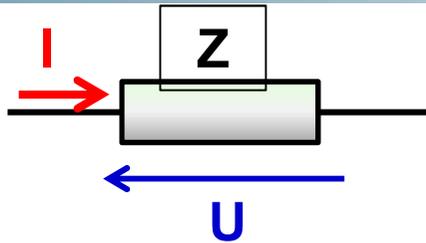


- ✓ Sources tension/courant et impédances connectés par l'intermédiaire de nœuds
- ✓ Convention générateur vs. récepteur

✓ Lois de Kirchoff :

- Loi des nœuds: En chaque nœud, la somme des courants entrants = somme des courants sortants. Ex: $I_1 + I_2 = I_3$
- Loi des mailles : La somme des tensions dans chaque maille = 0. Ex : $U_{AB} + U_{BE} + U_{EA} = 0$

- ✓ Circuit à composants linéaires → Théorème de superposition : la réponse d'un circuit à plusieurs excitations est la somme des réponses de ce circuit aux excitations appliquées individuellement..



Loi d'Ohm : $U = Z \times I$

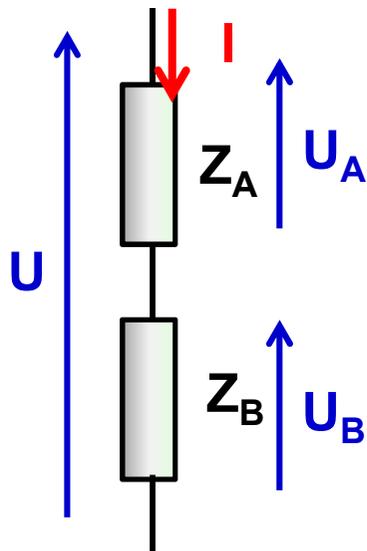
✓ Résistance : $U = R \times I$ \longrightarrow $Z = R$

✓ Condensateur : $I = C \times \frac{dU}{dt}$ \longrightarrow $Z = ?$

✓ Bobine (Inductance) : $U = L \times \frac{dI}{dt}$ \longrightarrow $Z = ?$



Mise en série



$$U = Z_{eq} I = U_A + U_B = (Z_A + Z_B) I$$

$$Z_{eq} = Z_A + Z_B$$

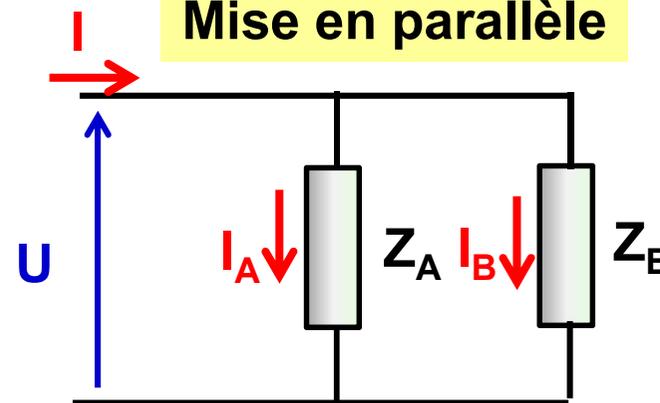
Pont diviseur de tension :

$$U_A = \frac{Z_A}{Z_A + Z_B} U$$

$$U_B = \frac{Z_B}{Z_A + Z_B} U$$

Remise a niveau electronique

Mise en parallèle



$$I = \frac{U}{Z_{eq}} = I_A + I_B = \frac{U}{Z_A} + \frac{U}{Z_B}$$

$$\frac{1}{Z_{eq}} = \frac{1}{Z_A} + \frac{1}{Z_B}$$

Pont diviseur de courant :

$$I_A = \frac{Z_B}{Z_A + Z_B} I$$

$$I_B = \frac{Z_A}{Z_A + Z_B} I$$

2. Régime harmonique et système linéaire



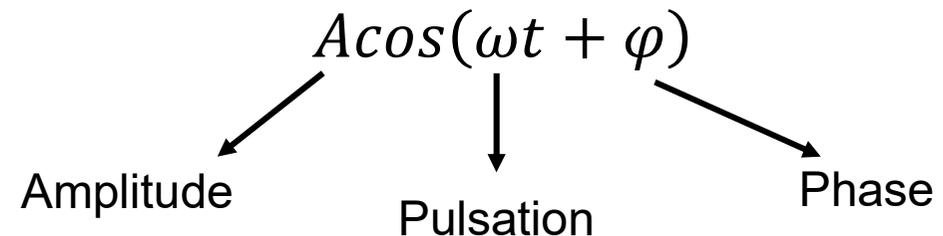
- ✓ Système linéaire à temps invariant :
 - Invariance temporelle
 - Linéarité : le principe de superposition s'applique

- ✓ Quels sont les systèmes dont les réponses les rendent forcément linéaires ?

- ✓ Un circuit R, L, C est-il linéaire à temps invariant ?

✓ Signaux invariants au « passage » à travers un système linéaire ?

✓ Signal (co)sinusoïdal
ou harmonique :

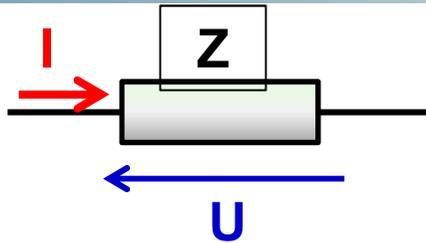


✓ Représentation complexe d'un signal (co)sinusoïdal :

$$\operatorname{Re}\left[A e^{j\varphi} e^{j\omega t}\right] = A \cos(\omega t + \varphi)$$

- ✓ Quel est l'effet d'un système linéaire sur un signal harmonique ?
 - Invariance de la fréquence (pulsation)
 - Seule l'amplitude complexe est modifiée (amplitude et phase)

- ✓ Intérêt du régime harmonique pour l'étude des systèmes linéaires ?



Loi d'Ohm : $U = Z \times I$

✓ Résistance : $U = R \times I \quad \longrightarrow \quad Z = R$

✓ Condensateur : $I = C \times \frac{dU}{dt}$

Régime harmonique ($\omega = 2\pi f$) : $I(\omega) = C \times j\omega U(\omega) \quad \longrightarrow \quad Z = \frac{1}{jC\omega}$

✓ Bobine : $U = L \times \frac{dI}{dt}$

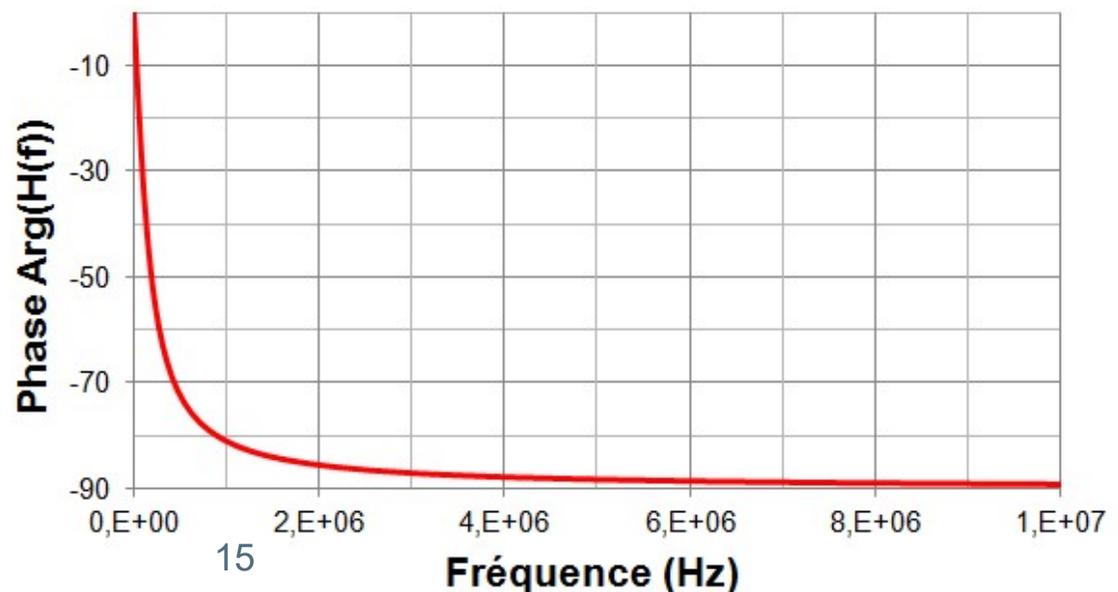
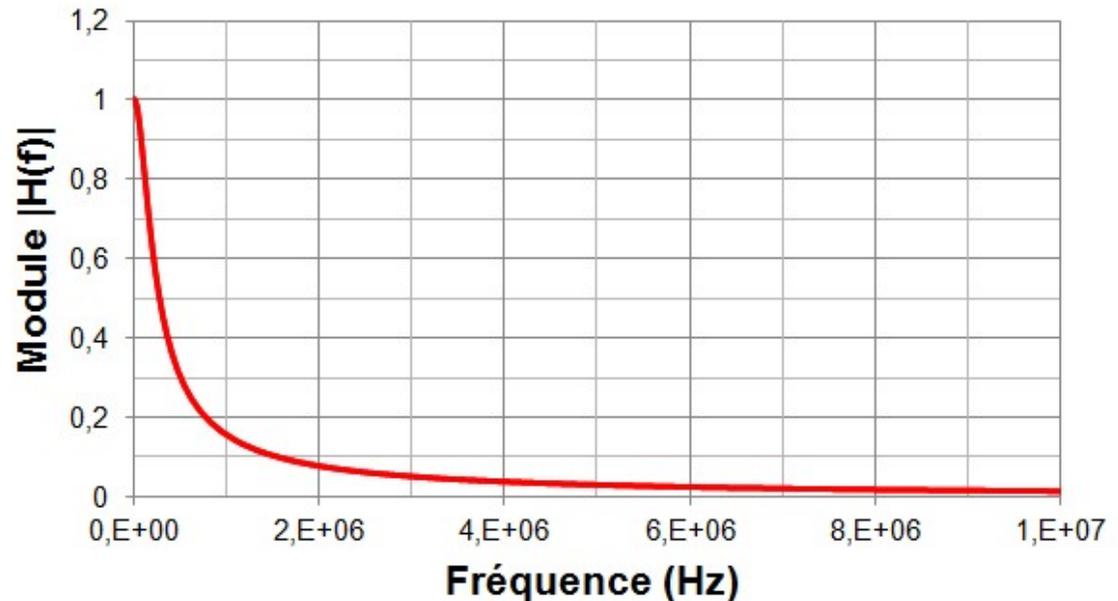
Régime harmonique ($\omega = 2\pi f$) : $U(\omega) = L \times j\omega I(\omega) \quad \longrightarrow \quad Z = jL\omega$

3. Filtre analogique

$$H(f) = \frac{1}{1 + j \frac{f}{f_c}}, f_c = \frac{1}{2\pi RC}$$

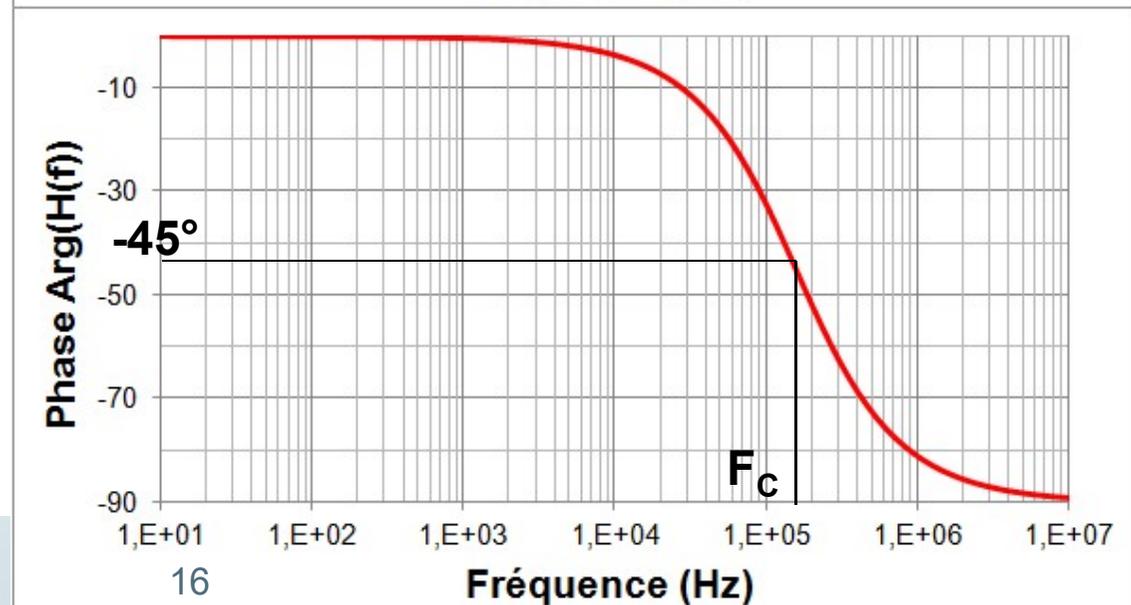
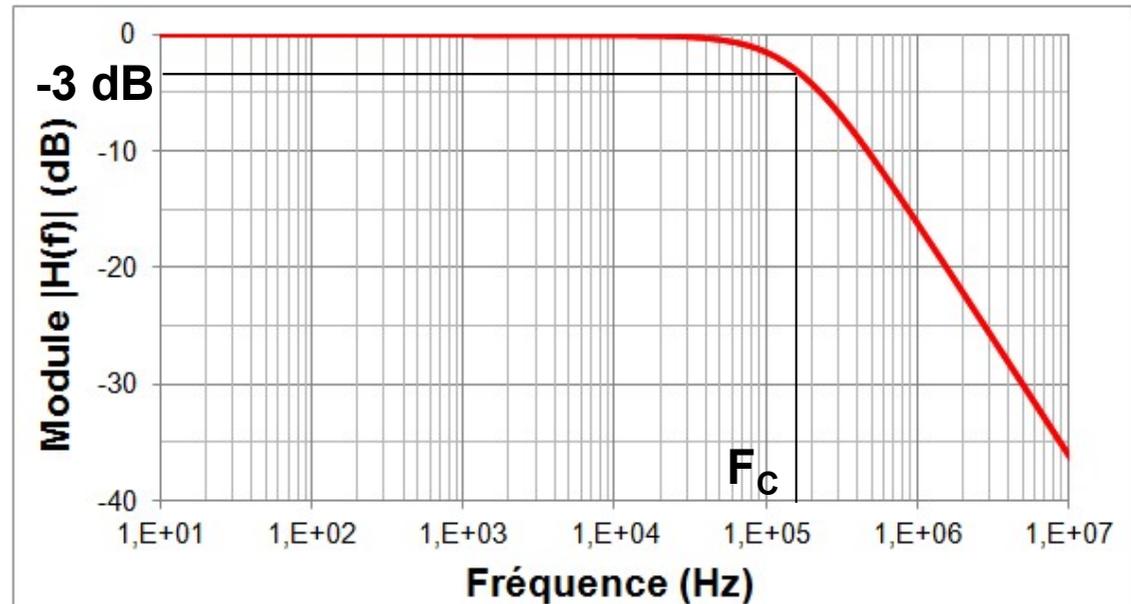
- ✓ Diagramme de Bode (représentation module – phase vs. Fréquence)
- ✓ Filtre passe-bas d'ordre 1 (RC) : R = 1 kΩ et C = 1 nF
- ✓ Fréquence de coupure :

$$F_c = \frac{1}{2\pi RC} = 159 \text{ kHz}$$

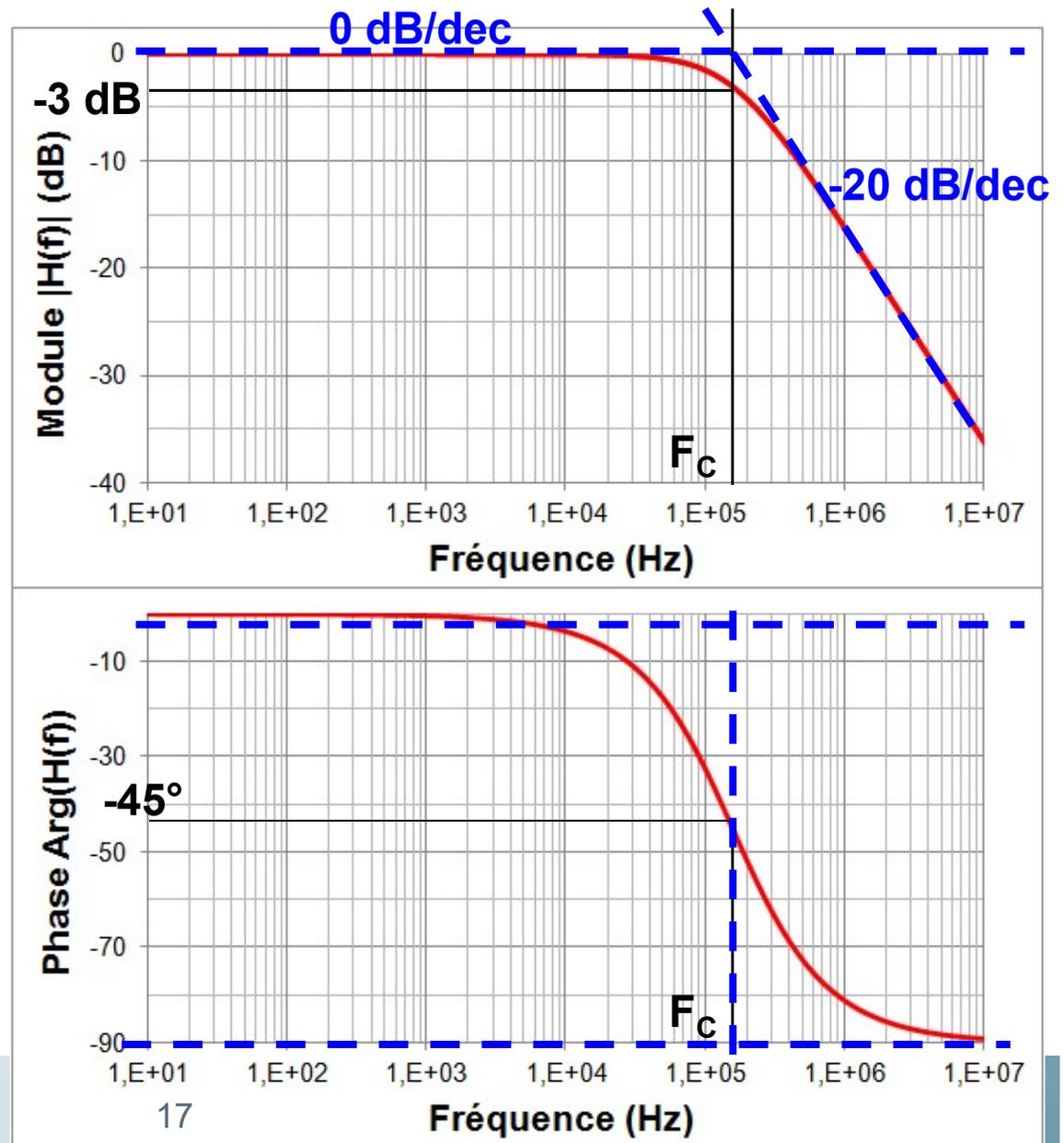


- ✓ Diagramme de Bode
(représentation module
– phase vs. Fréquence)
- ✓ Tracé log/log
- ✓ Filtre passe-bas d'ordre
1 (RC) : R = 1 kΩ et C
= 1 nF
- ✓ Fréquence de coupure

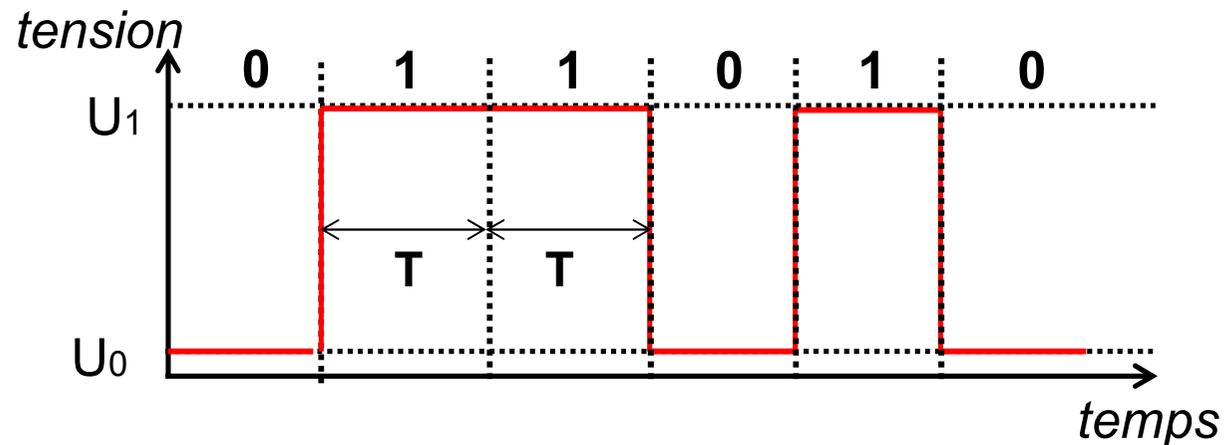
$$F_C = \frac{1}{2\pi RC} = 159\text{kHz}$$



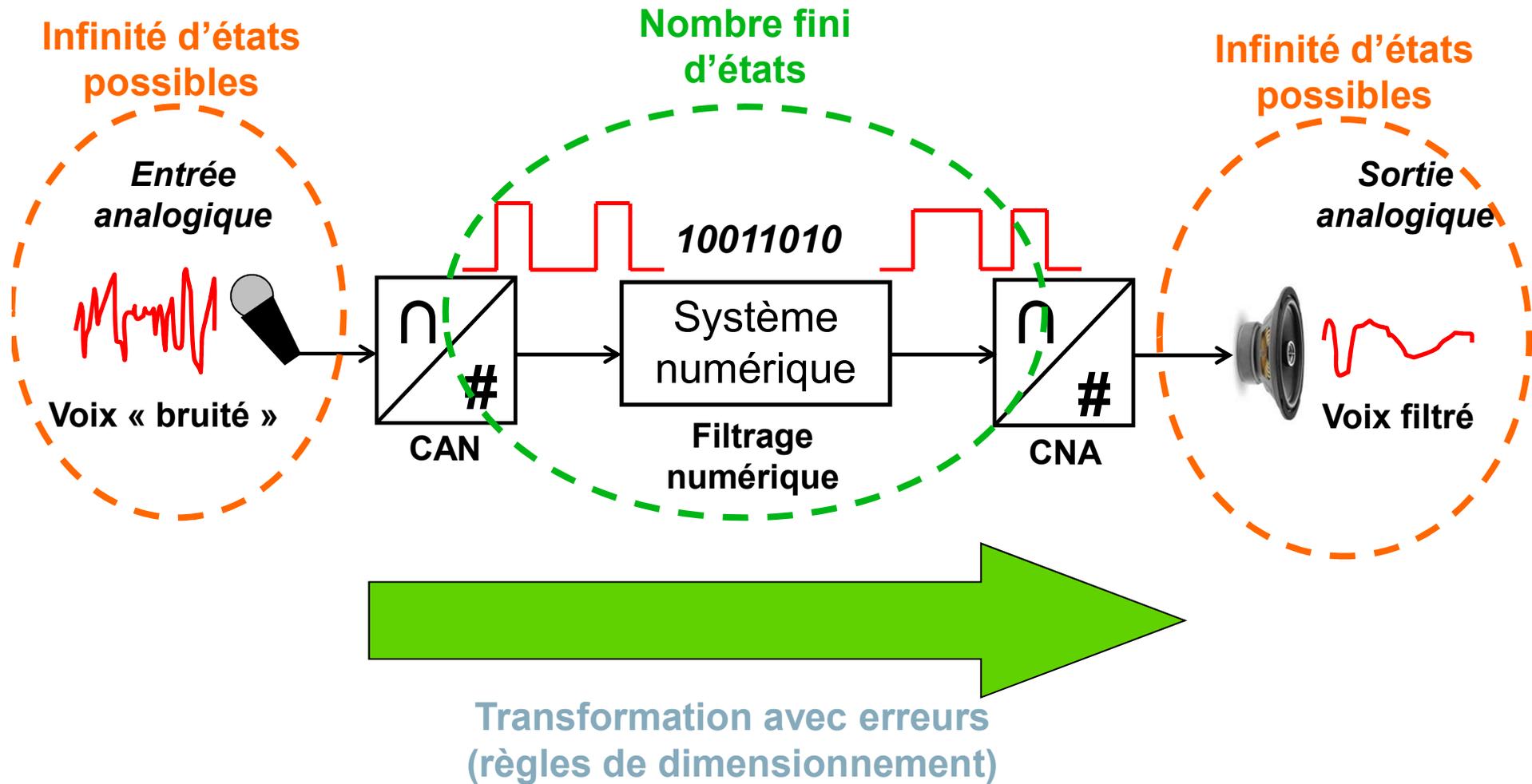
- ✓ Repérer les grandes tendances et les fréquences caractéristiques



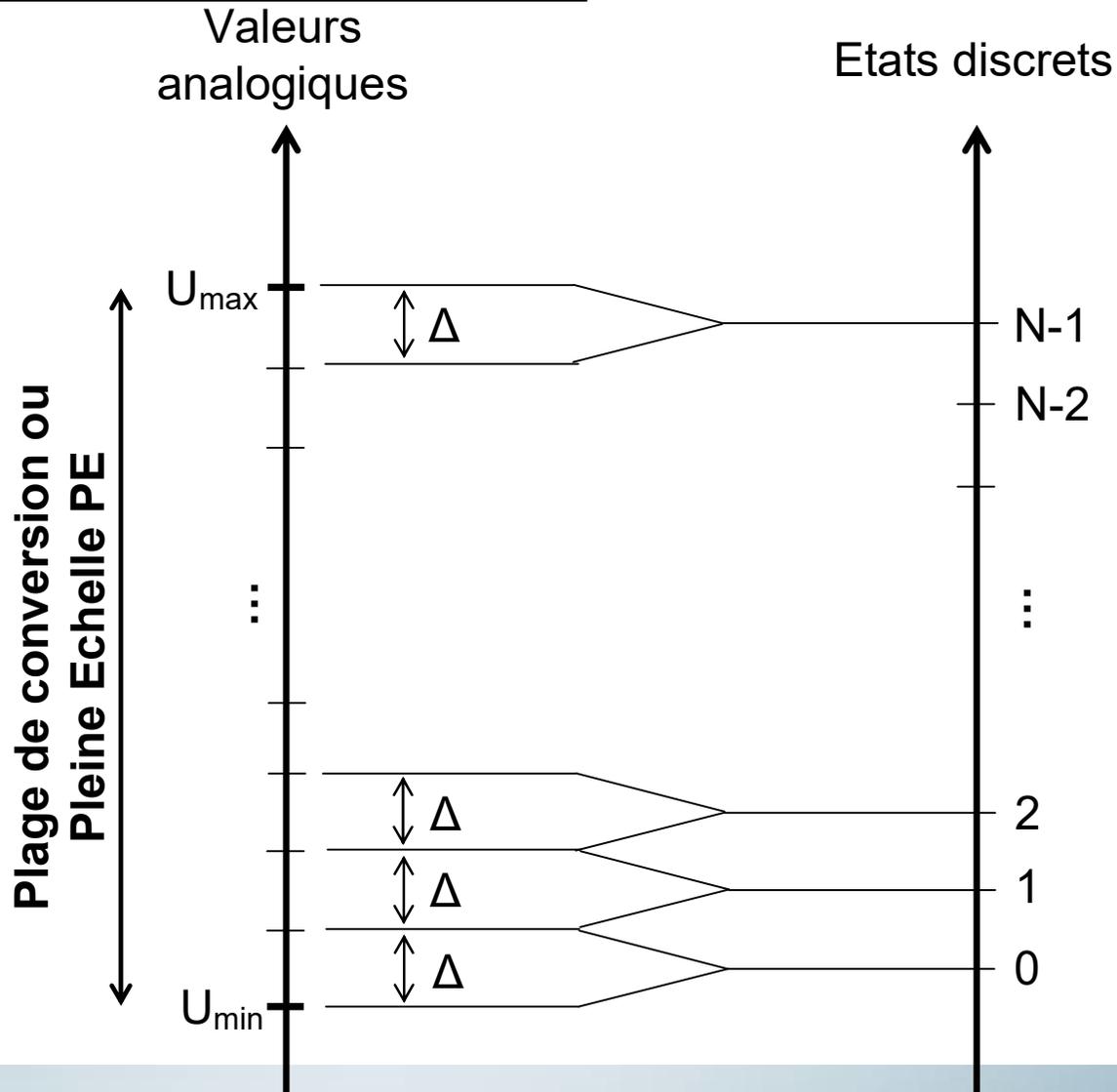
4. Conversion analogique-numérique



- ✓ Si la transmission des signaux binaires est cadencée à un rythme fixe de période T , le signal numérique est dit synchrone, produit par un **circuit synchrone**.
- ✓ Sinon, il s'agit d'un signal asynchrone, produit par un **circuit asynchrone**.



Quantification en amplitude



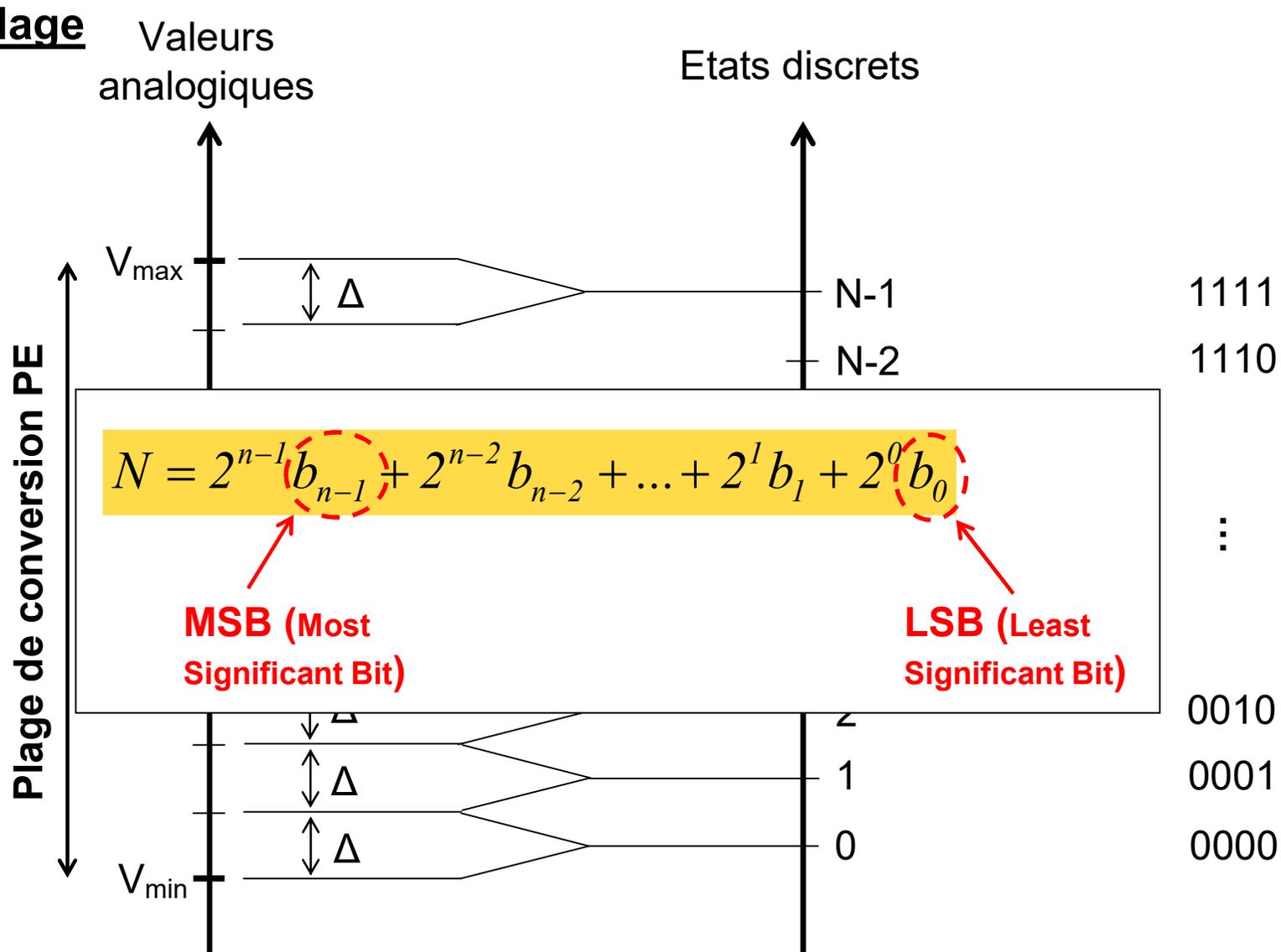
Nombre d'états :

$$N = 2^n, n \text{ entier}$$

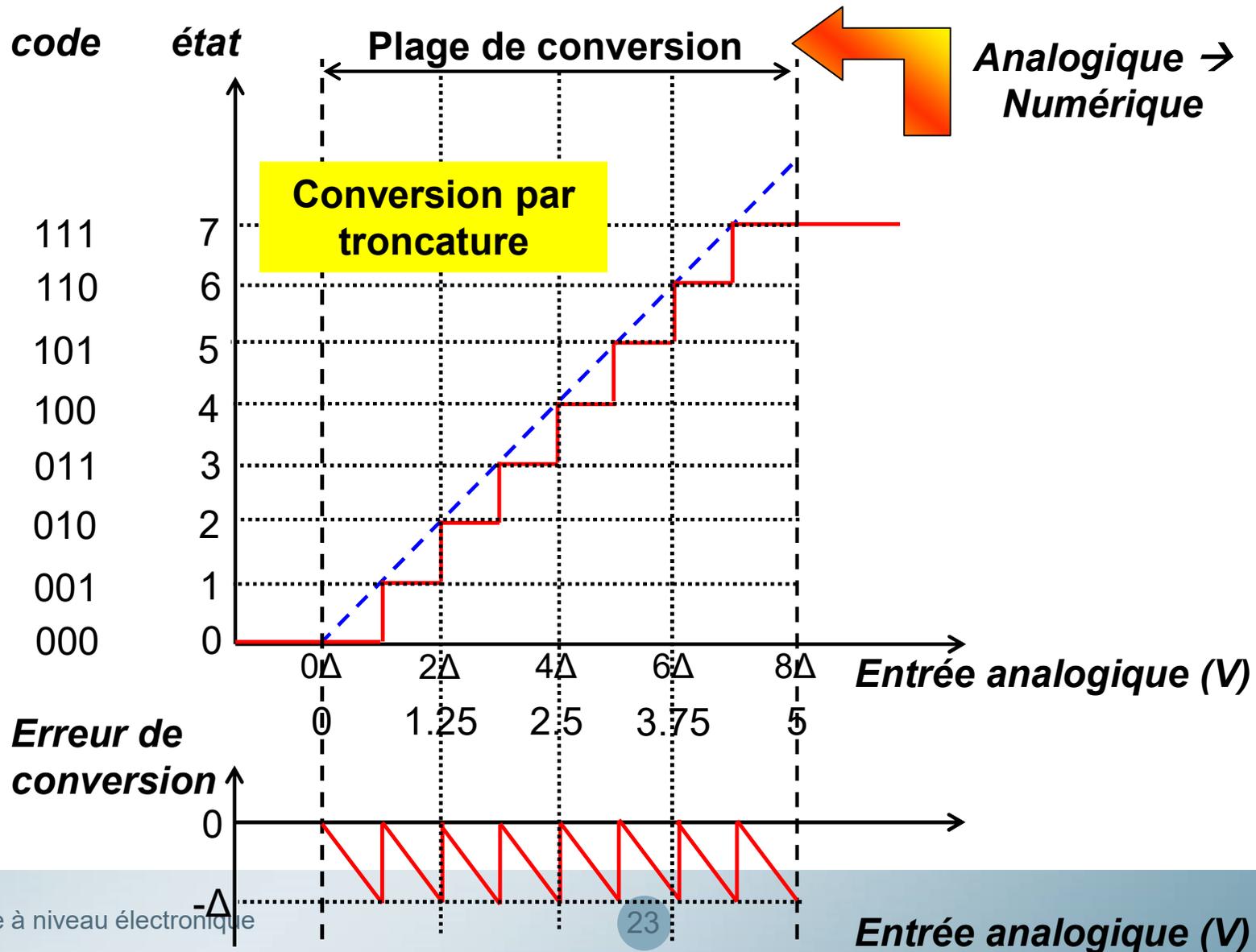
Pas de conversion
ou résolution :

$$\Delta = \frac{PE}{N} = \frac{PE}{2^n}$$

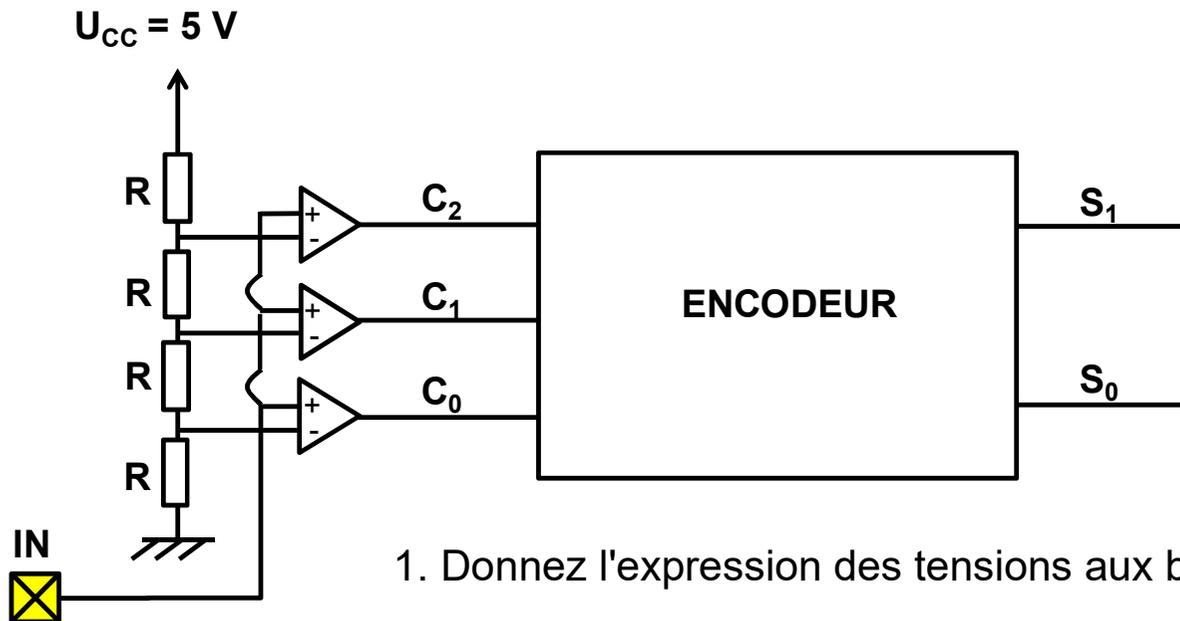
Codage



Erreur de conversion



Exercice – CAN Flash



1. Donnez l'expression des tensions aux bornes de chaque résistance R .
2. Donnez l'état C_i en sortie du comparateur n° i en fonction de la tension U_{in} .
3. En quoi ce dispositif forme t-il un CAN ? Quelle est sa pleine échelle ? Son pas de conversion ?
4. Quel est le rôle de l'encodeur ? Pourquoi a-t-il deux sorties ?
5. Proposez le circuit logique de cet encodeur.
6. Combien de composants élémentaires faudrait-il pour un CAN 8 bits ?

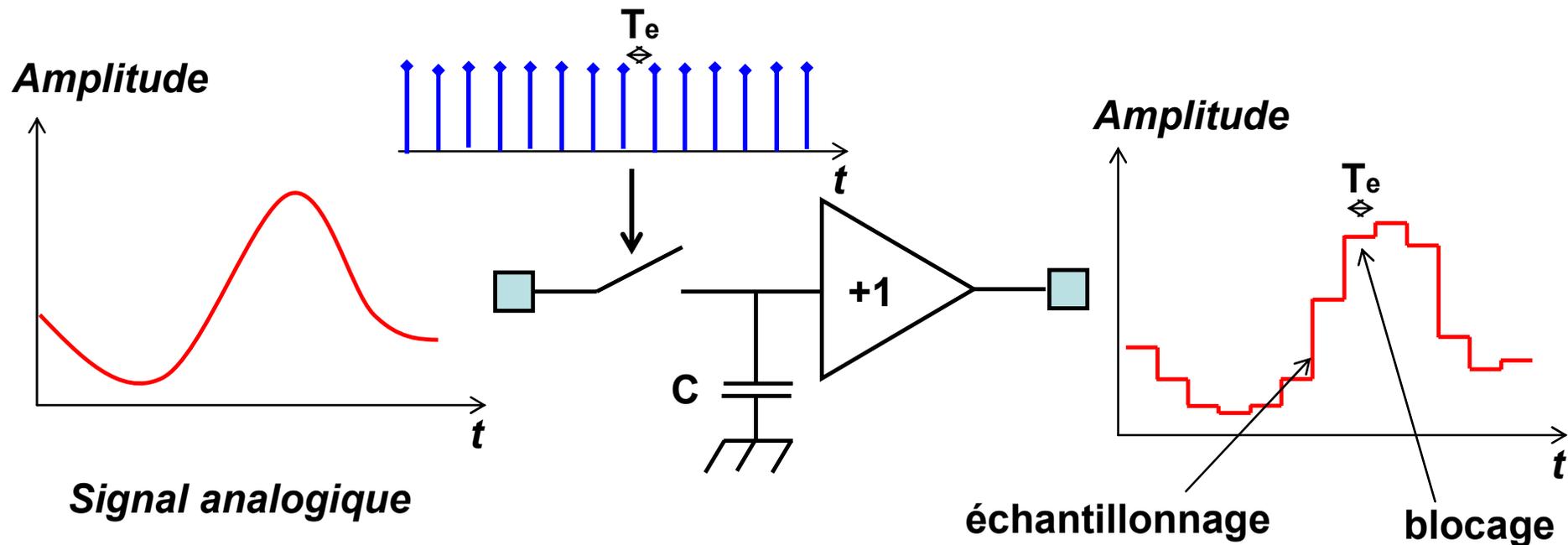
Exercice – CAN Flash

| Vin | N° état | C0 | C1 | C2 | S1 | S0 |
|-----------------|---------|----|----|----|----|----|
| < Ucc/4 | 0 | 0 | 0 | 0 | 0 | 0 |
| Ucc/4 - Ucc/2 | 1 | 1 | 0 | 0 | 0 | 1 |
| Ucc/2 - 3 Ucc/4 | 2 | 1 | 1 | 0 | 1 | 0 |
| > 3Ucc/4 | 3 | 1 | 1 | 1 | 1 | 1 |

$$5. S0 = \underline{C0} \cdot C1 \cdot C2 + C0 \cdot C1 \cdot C2 = C0 \cdot (C2 \cdot C3 + C2 \cdot C3) = C1 \cdot (\text{xnor}(C2, C3))$$

$$S1 = C1$$

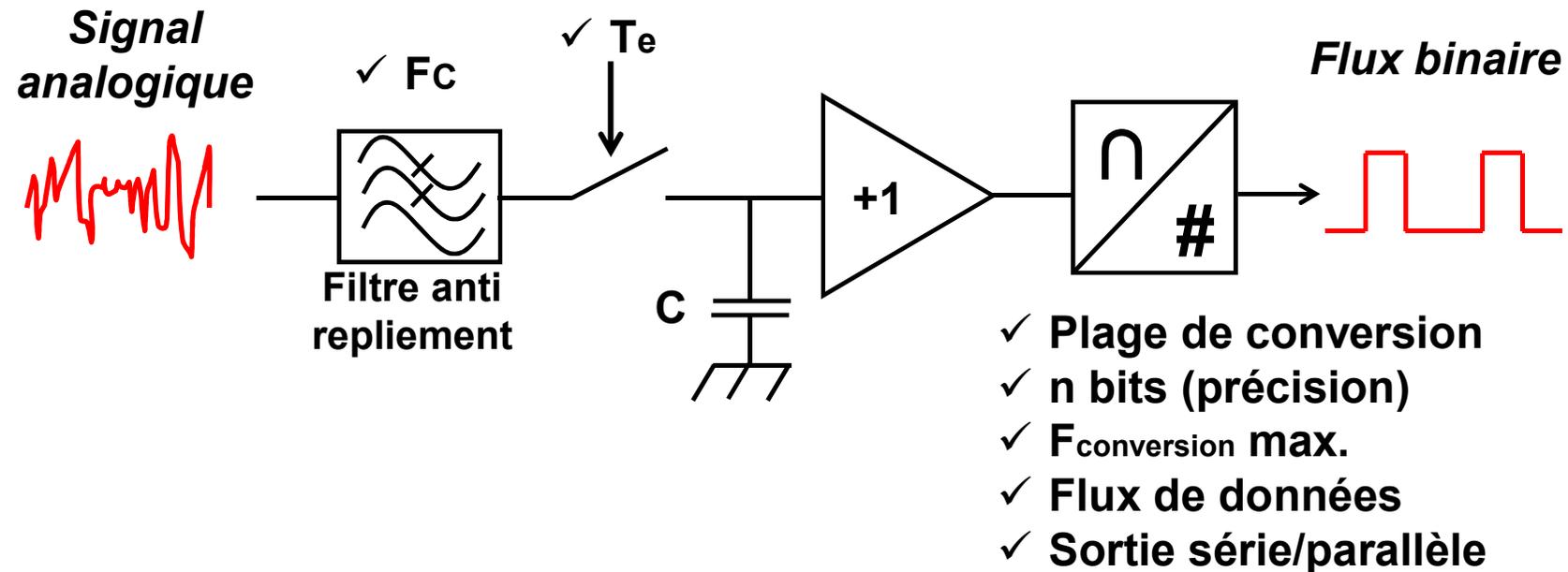
Echantillonneur - bloqueur



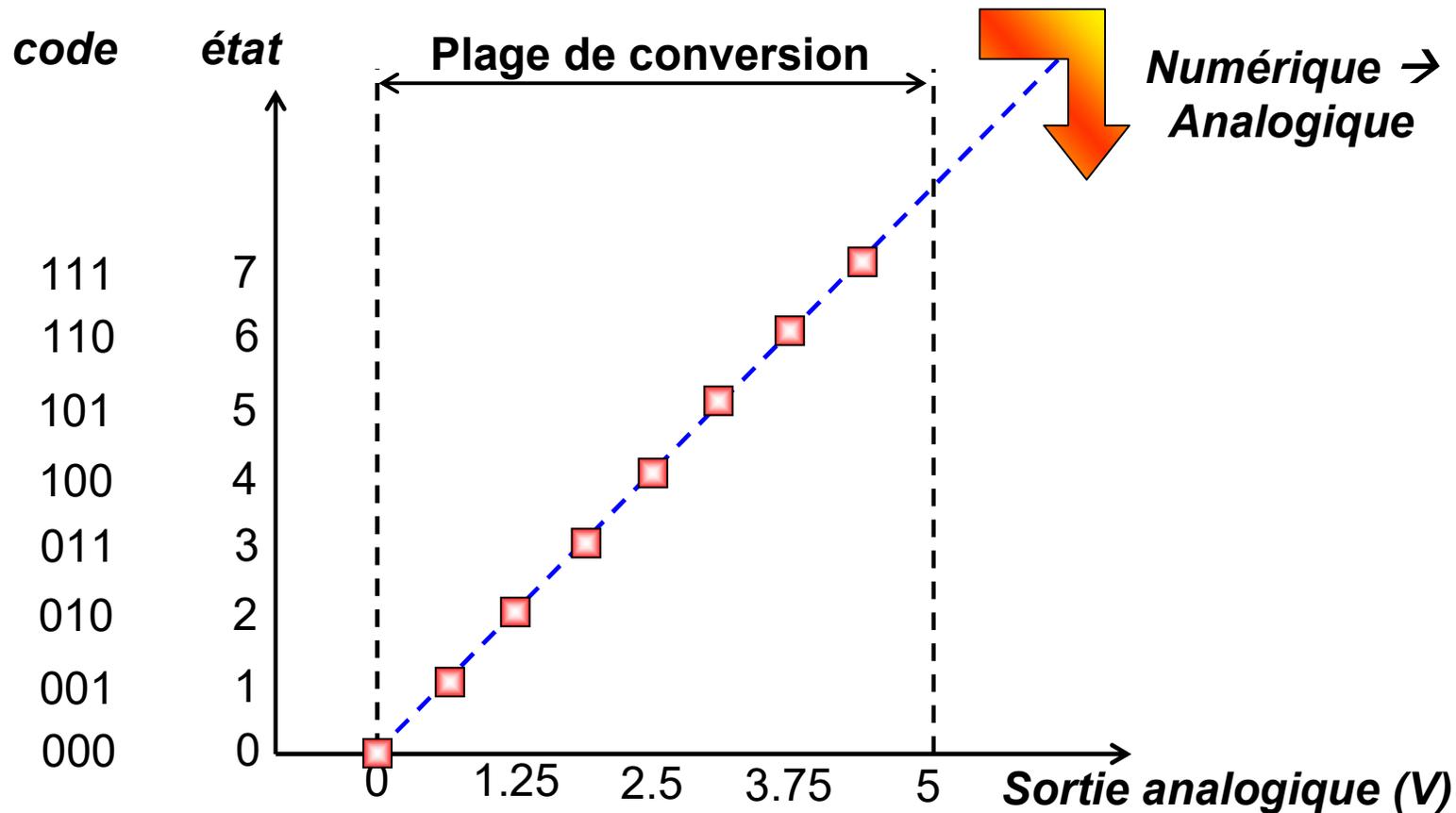
Fréquence d'échantillonnage F_e doit respecter le théorème de Nyquist-Shannon :

$$F_e = \frac{1}{T_e} \geq 2 \times f_{\max}$$

Chaîne de conversion analogique numérique

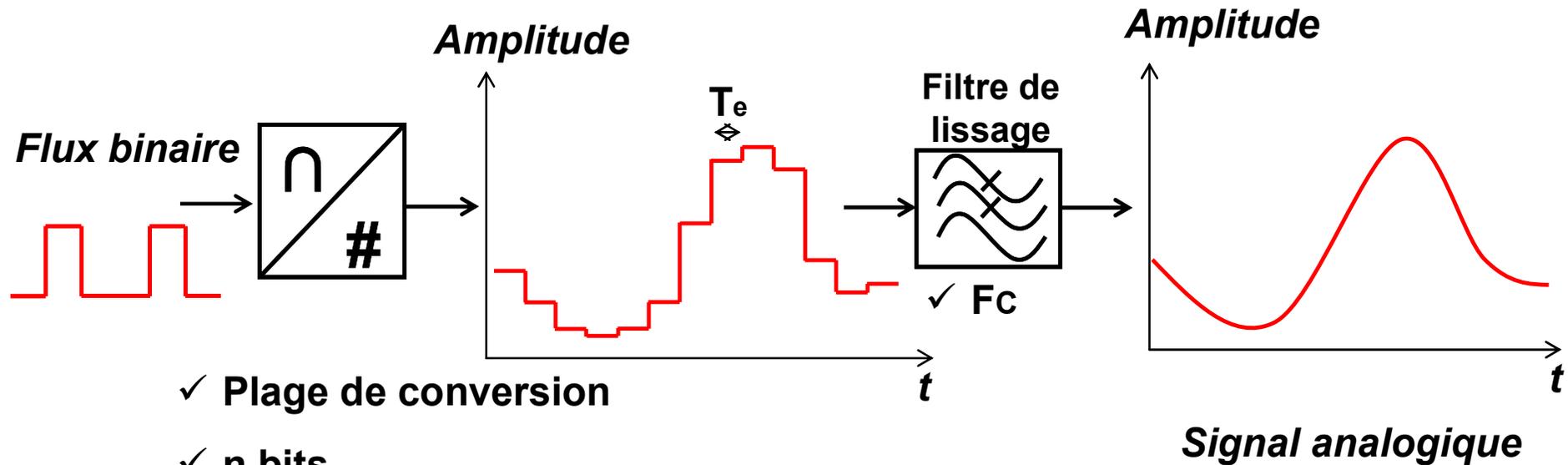


Convertisseur numérique-analogique



$$U_{CNA} = \Delta \times (2^{n-1} b_{n-1} + 2^{n-2} b_{n-2} + \dots + 2^1 b_1 + 2^0 b_0)$$

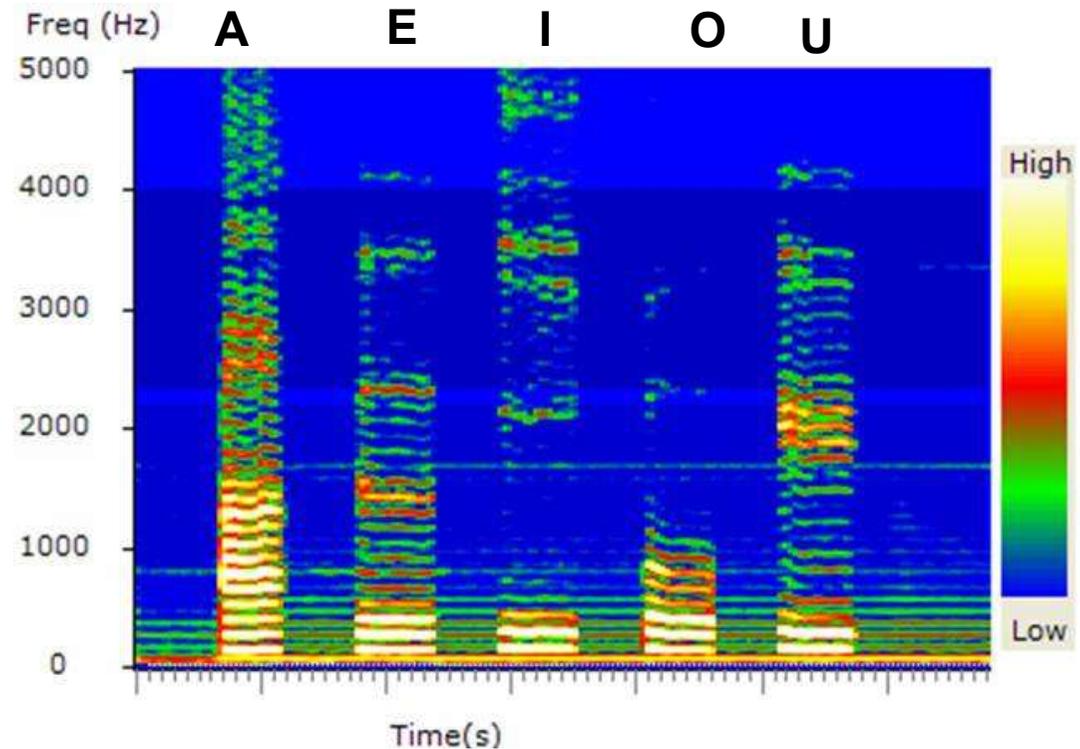
Chaîne de conversion numérique analogique



- ✓ Plage de conversion
- ✓ n bits
- ✓ Bande passante
- ✓ Entrée série/parallèle

Exercice – Numérisation voix humaine

1. Fréquence d'échantillonnage minimum ?
2. On veut une erreur de conversion $< 0.5\%$ de la PE. Sur quel nombre de bits doit-on coder les échantillons ?
3. Quelle est la résolution sachant que la PE du CAN est compris entre 0 et 5 V ?

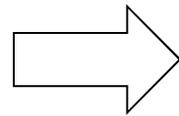


4. Le CAN renvoie le mot 0x1001 0110. Quelle tension était appliquée en entrée du CAN ? Peut-on la connaître avec précision ?
5. Quel est le débit binaire en sortie du CAN ?

Exercice – Numérisation voix humaine

$$resolution = \frac{PE}{2^n} = \frac{5}{2^8} = \frac{5}{256} = 19.53 \text{ mV}$$

$$code = 1 \times 2^7 + 1 \times 2^4 + 1 \times 2^2 + 1 \times 2^1 = 150$$



$$150 \times resolution < V_{in} < 151 \times resolution$$

$$2.9297 \text{ V} < V_{in} < 2.9492 \text{ V}$$

5. Circuits numériques – Portes de base

Etats logiques

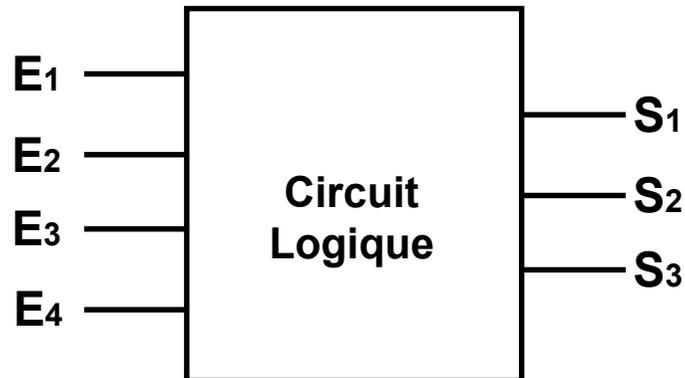
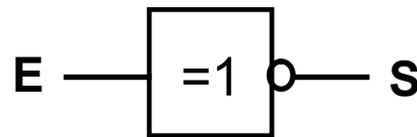
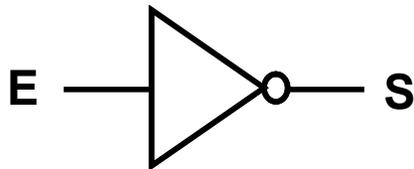


Table de vérité

| E1 | E2 | ... | E _n | S1 | S2 | ... |
|------|----|-----|----------------|----|----|-----|
| 0 | 0 | | 0 | 0 | 1 | |
| 1 | 0 | | 0 | 0 | 0 | |
| 1 | 1 | | 0 | X | 0 | |
| | | | | | | |
| 1 | 1 | | 1 | 1 | X | |

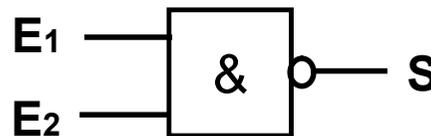
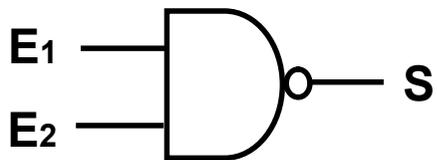
Portes à logique combinatoire

Inverseur $S = \overline{E}$



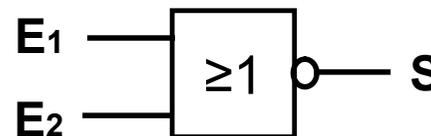
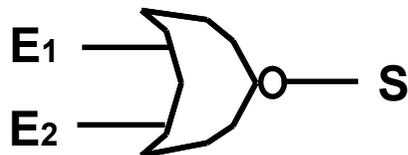
| E | S |
|---|---|
| 0 | 1 |
| 1 | 0 |

Porte NAND $S = \overline{E_1 \cdot E_2}$



| E1 | E2 | S |
|----|----|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

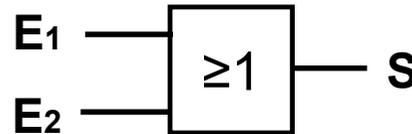
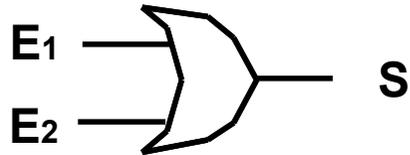
Porte NOR $S = \overline{E_1 + E_2}$



| E1 | E2 | S |
|----|----|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

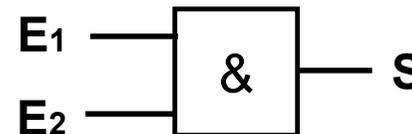
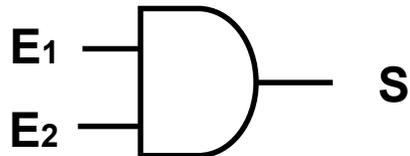
Portes à logique combinatoire

Ou $S = E_1 + E_2$



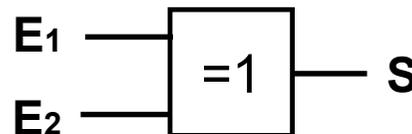
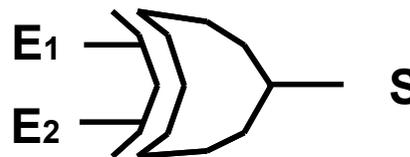
| E1 | E2 | S |
|----|----|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Et $S = E_1 \cdot E_2$



| E1 | E2 | S |
|----|----|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

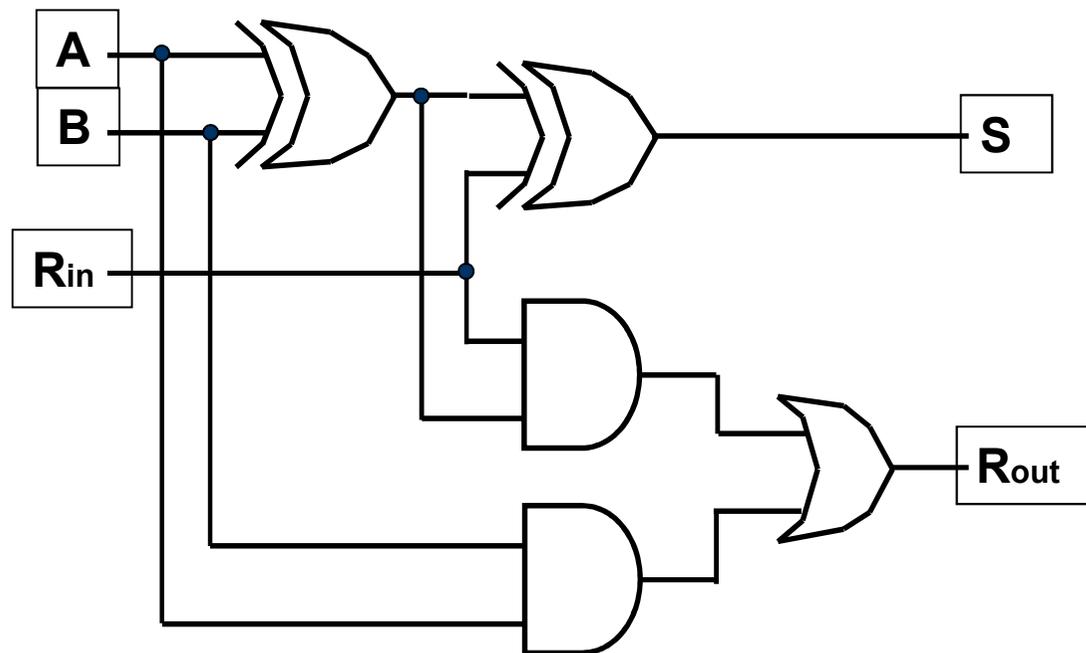
Ou exclusif (XOR) $S = E_1 \oplus E_2$



| E1 | E2 | S |
|----|----|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Portes à logique combinatoire

- ✓ Table de vérité de ce circuit ?
- ✓ Identifier sa fonction
- ✓ Que se passe t-il si on connecte en cascade ce circuit ?



Portes à logique combinatoire

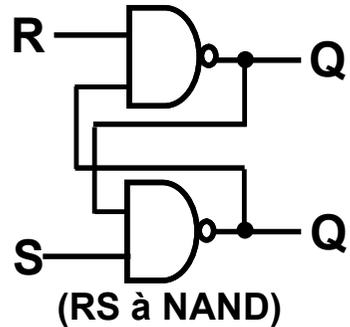
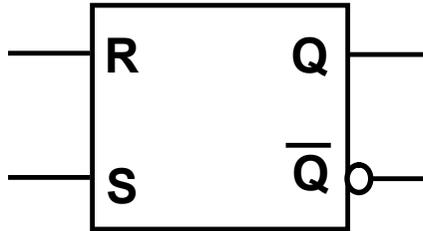
$$S = (A \text{ xor } B) \text{ xor } R_{in}$$

$$R_{out} = ((A \text{ xor } B) \text{ and } R_{in}) \text{ or } (A \text{ and } B)$$

| A | B | R _{in} | S | R _{out} |
|---|---|-----------------|----------|------------------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

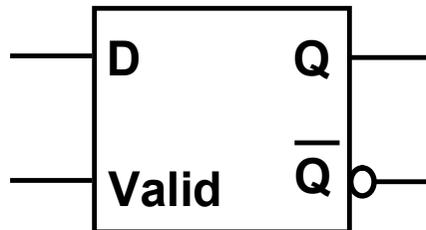
Portes à logique séquentielle – Bascules asynchrones

RS (Reset/Set)



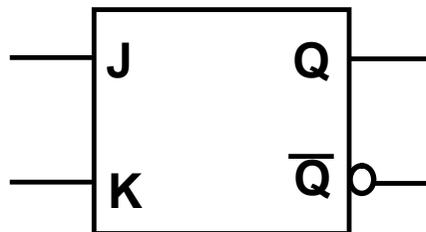
| R | S | Q_{N+1} | Q'_{N+1} |
|---|---|-----------|------------|
| 0 | 0 | Interdit | |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | Q_N | Q'_N |

D (Data)



| D | Valid | Q_{N+1} | Q'_{N+1} |
|---|-------|-----------|------------|
| 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |
| X | 0 | Q_N | Q'_N |

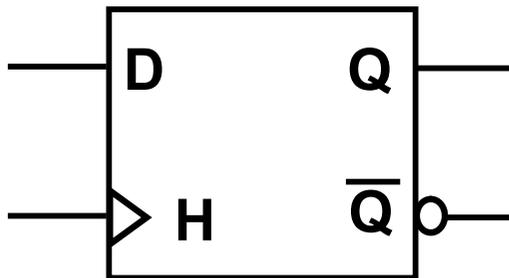
JK



| J | K | Q_{N+1} | Q'_{N+1} |
|---|---|-----------|------------|
| 0 | 0 | Q_N | Q'_N |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 0 | 0 | Q'_N | Q_N |

Portes à logique séquentielle – Bascules synchrones

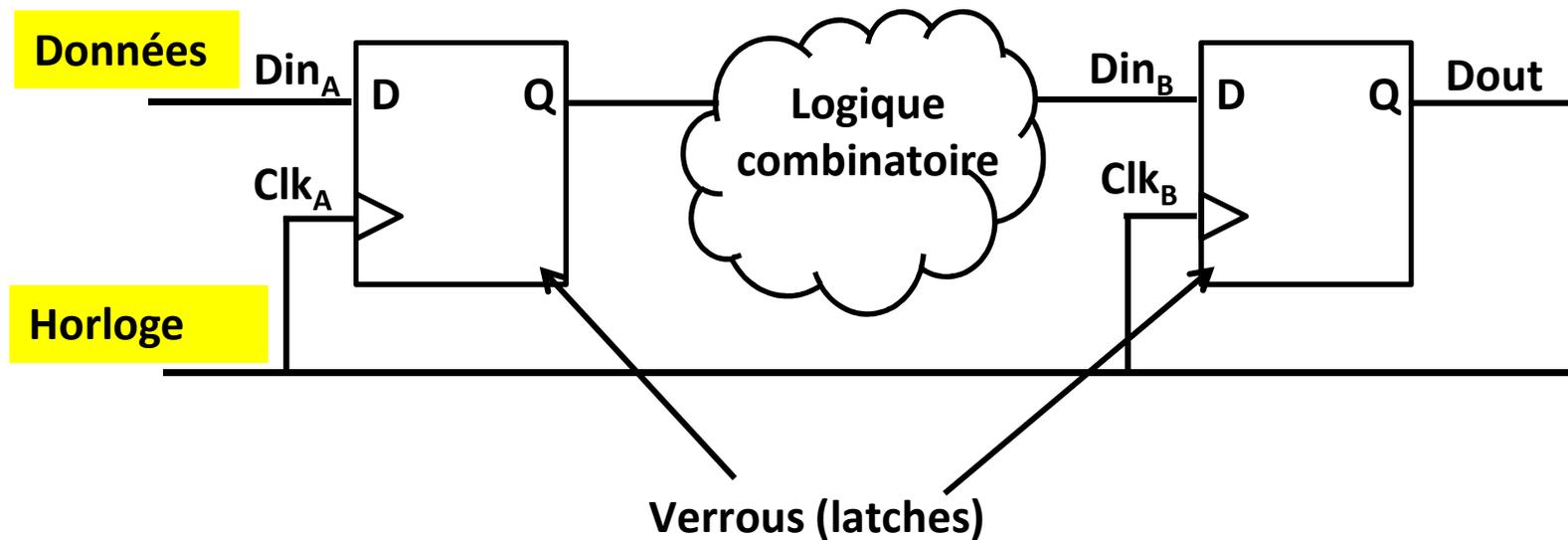
D à déclenchement



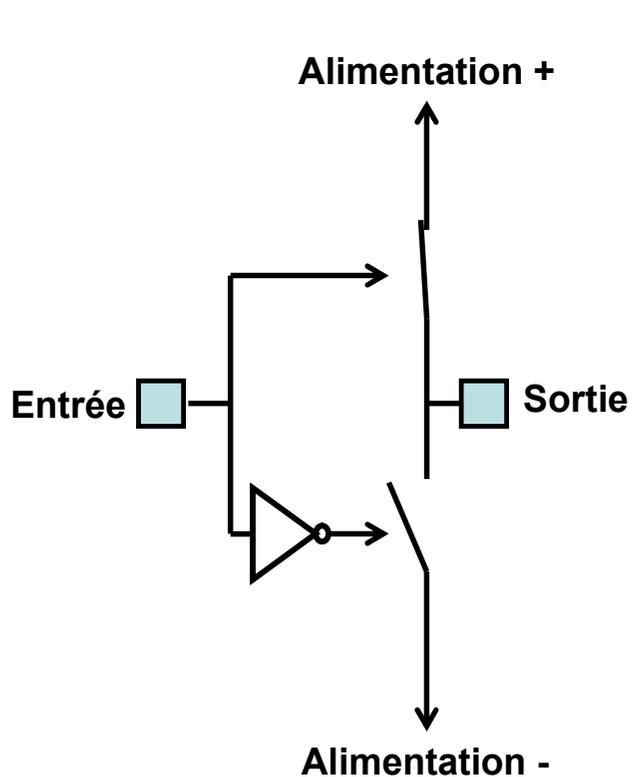
| D | H | Q_{N+1} | Q'_{N+1} |
|---|---|-----------|------------|
| 0 | ↑ | 1 | 0 |
| 1 | ↑ | 0 | 1 |
| X | ↓ | Q_N | Q'_N |

6. Circuits numériques – Structure générale

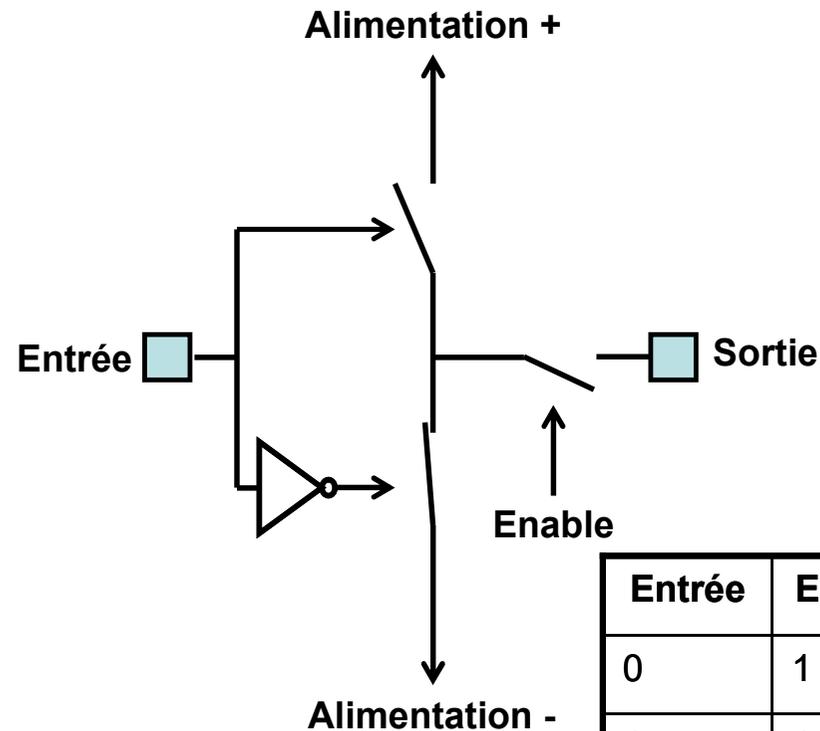
Structure circuit numérique synchrone



Sorties totem pole vs 3 états (tristate)



Sortie Totem pole

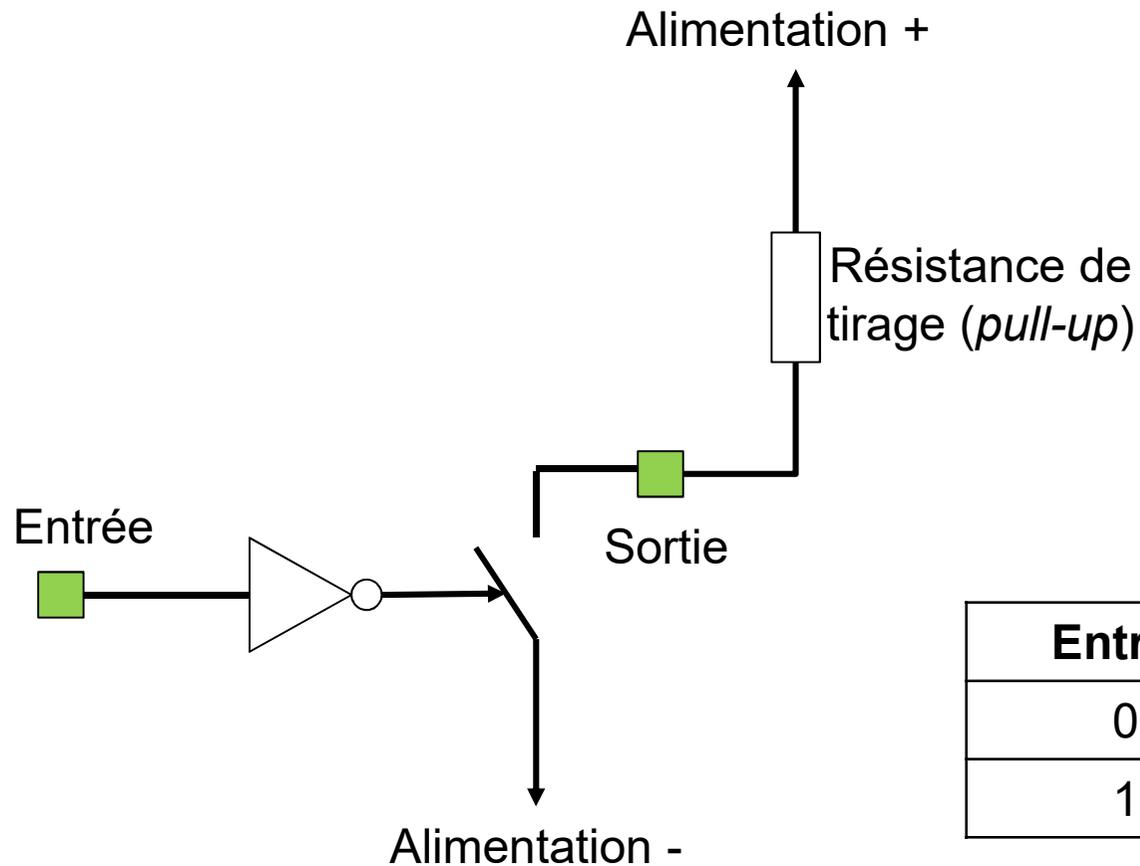


Sortie 3 états

| Entrée | Enable | Sortie |
|--------|--------|--------|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| X | 0 | Z |

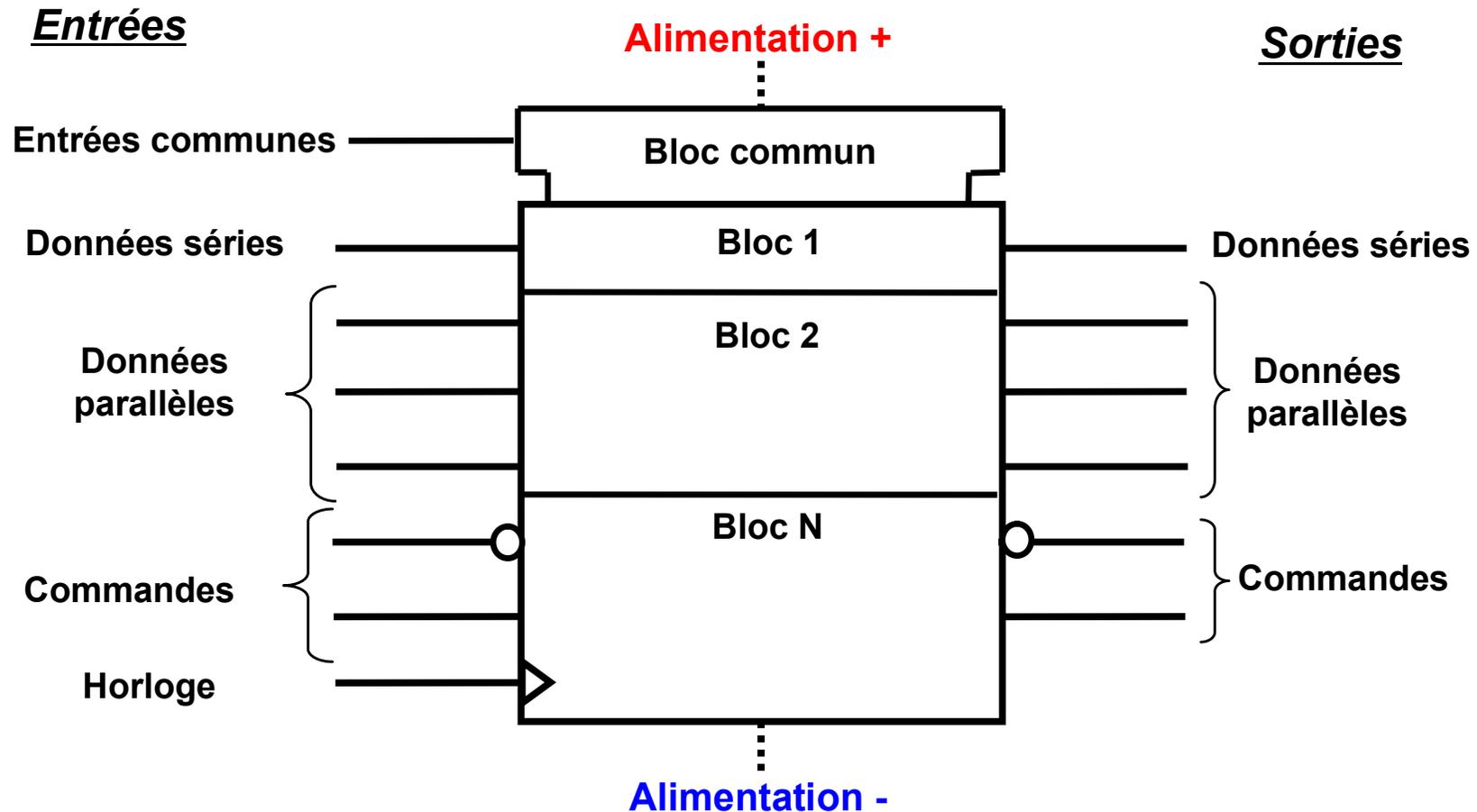
**Haute impédance
(High Z)**

Sortie open-drain



| Entrée | Sortie |
|--------|--------|
| 0 | 0 |
| 1 | 1 |

Structure générale d'un circuit numérique



- Négation logique en entrée/sortie
- ▷ Entrée dynamique (active sur front)

Structure générale d'un circuit numérique – exemples

SN74LVC245A Octal Bus Transceiver With 3-State Outputs

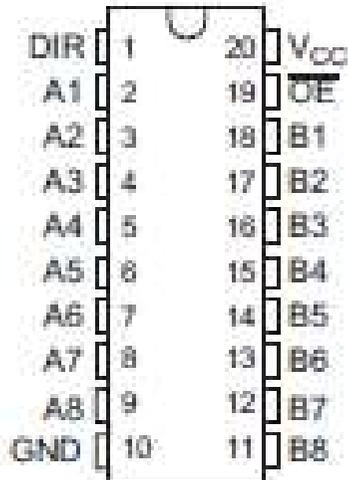


Table 1. Function Table

| INPUTS | | OPERATION |
|--------|-----|-----------------|
| OE | DIR | |
| L | L | B data to A bus |
| L | H | A data to B bus |
| H | X | Isolation |

Définir la nature et le rôle de chaque broche de ce circuit.

Synchrone / asynchrone ?

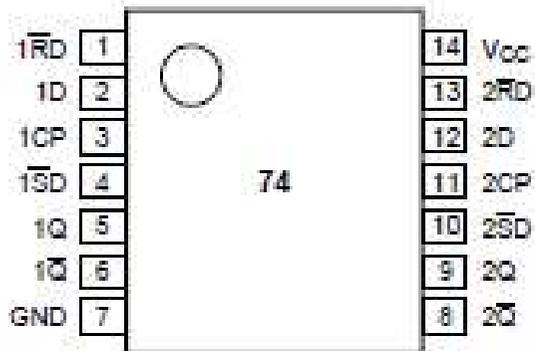
Sur quel niveau les entrées de commande sont-elles actives ?

Logique de commandes des E/S A et B ?

Structure générale d'un circuit numérique – exemples

74AHC74; 74AHCT74

Dual D-type flip-flop with set and reset; positive-edge trigger



| Control | | | Input | Output | | | |
|------------------|------------------|-------|-------|--------|-----------------|------------|-----------------------|
| \overline{nSD} | \overline{nRD} | nCP | nD | nQ | \overline{nQ} | nQ_{n+1} | \overline{nQ}_{n+1} |
| L | H | X | X | H | L | - | - |
| H | L | X | X | L | H | - | - |
| L | L | X | X | H | H | - | - |
| H | H | ↑ | L | - | - | L | H |
| H | H | ↑ | H | - | - | H | L |

Définir la nature et le rôle de chaque broche de ce circuit.

Synchrone / asynchrone ?

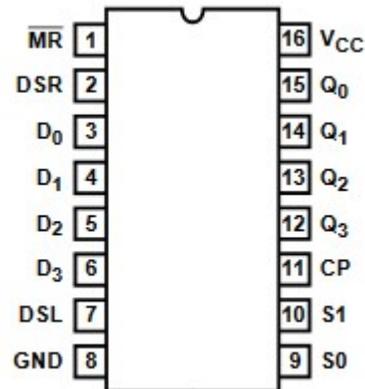
Sur quel niveau les entrées de commande sont-elles actives ?

Structure générale d'un circuit numérique – exemples

*CD54HC194, CD74HC194,
CD74HCT194*

TRUTH TABLE

High-Speed CMOS Logic
4-Bit Bidirectional Universal Shift Register



| OPERATING MODE | INPUTS | | | | | | | OUTPUT | | | |
|-------------------|--------|-----------------|----|----|-----|-----|-------|--------|-------|-------|-------|
| | CP | \overline{MR} | S1 | S0 | DSR | DSL | D_n | Q_0 | Q_1 | Q_2 | Q_3 |
| Reset (Clear) | X | L | X | X | X | X | X | L | L | L | L |
| Hold (Do Nothing) | X | H | l | l | X | X | X | q_0 | q_1 | q_2 | q_3 |
| Shift Left | ↑ | H | h | l | X | l | X | q_1 | q_2 | q_3 | L |
| | ↑ | H | h | l | X | h | X | q_1 | q_2 | q_3 | H |
| Shift Right | ↑ | H | l | h | l | X | X | L | q_0 | q_1 | q_2 |
| | ↑ | H | l | h | h | X | X | H | q_0 | q_1 | q_2 |
| Parallel Load | ↑ | H | h | h | X | X | d_n | d_0 | d_1 | d_2 | d_3 |

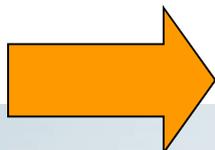
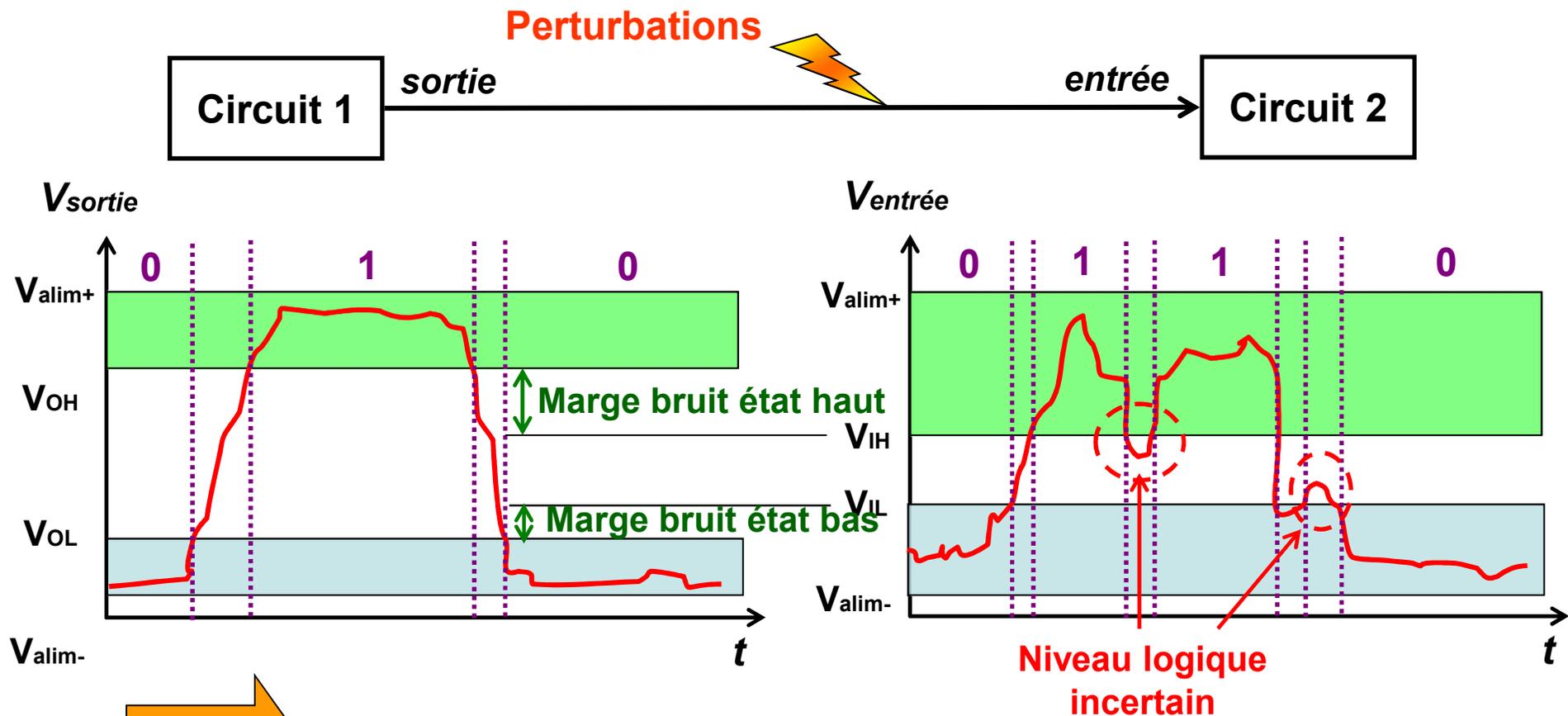
Mêmes questions ?

Rôle de circuit ?

7. Circuits numériques – Caractéristiques électriques

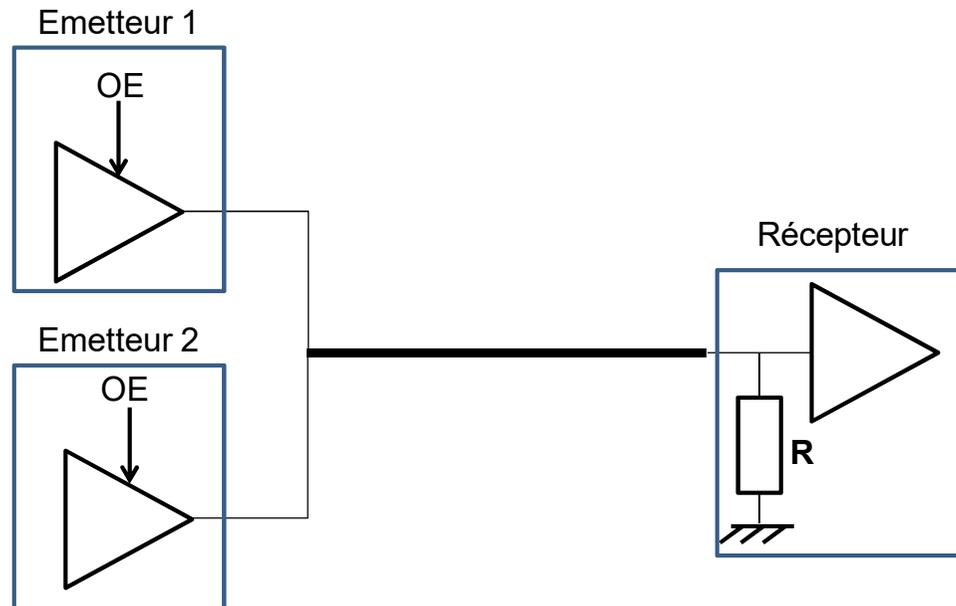
Niveaux logiques et marges de bruit

- ✓ Un signal numérique est associé à un état binaire si il appartient à une plage de tension, dépendante de la technologie.



PRECAUTION ?

Niveaux logiques et marges de bruit



On considère le bus digital ci-contre, sur lequel sont connectés 2 émetteurs et 1 récepteur.

Les caractéristiques de ces circuits sont :

- alimentation $V_{cc} = 5\text{ V}$
- $V_{il} = 0.8\text{ V}$ et $V_{ih} = 2\text{ V}$
- $V_{ol} = 0\text{ V}$ et $V_{oh} = V_{cc}$ (si $I_{out} = 0$)
- Courant de sortie max $I_{out} = 20\text{ mA}$
- Courant d'entrée $I_{in} = 0\text{ A}$

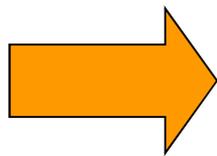
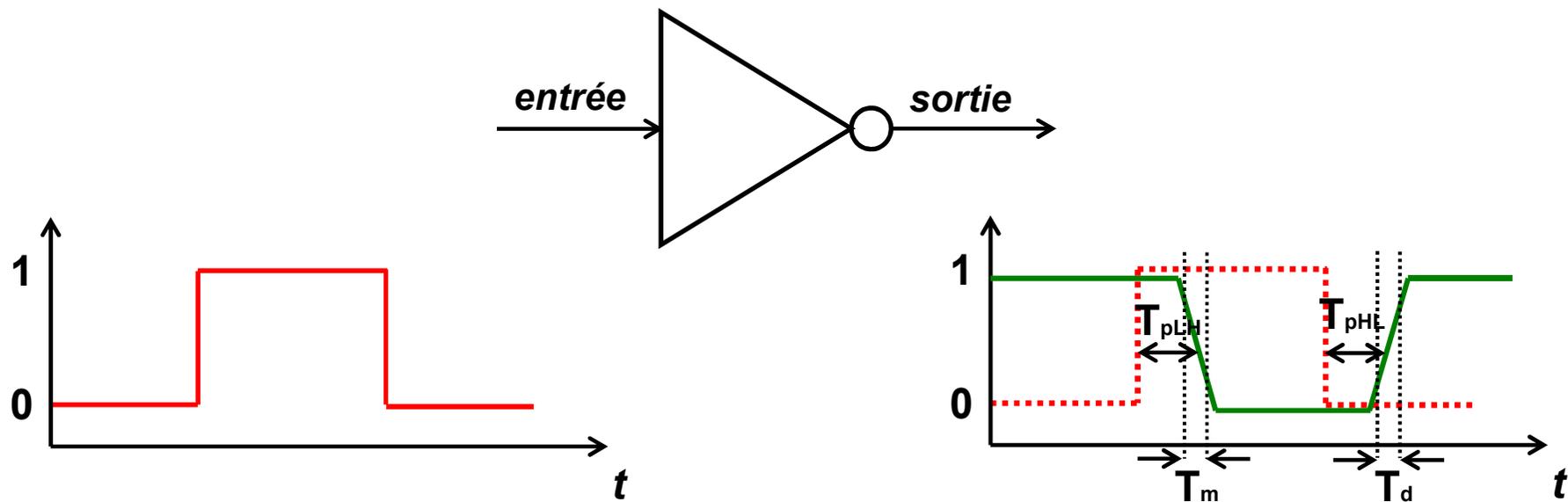
1. Pourquoi les sorties des émetteurs doivent-elles être trois états ?
2. Quel est le rôle de la résistance R ?
3. Quelle condition doit être vérifiée pour qu'un état logique bas transmis soit reçu sans erreur ?
Quelle est la marge de bruit ?
4. Même question pour un état logique haut ?
5. Proposez une valeur pour la résistance R.

Niveaux logiques et marges de bruit

$$V_{in} > V_{ih} \rightarrow R \cdot I_{outmax} > V_{ih} \rightarrow R > V_{ih}/I_{out} = 2/0.02 = 100 \text{ ohms}$$

Temps de commutation et de propagation

- Les circuits numériques présentent une certaine inertie face au changement. Leur réaction n'est pas instantanée.
- Les interconnexions entre circuits introduisent aussi un délai.
- Caractérisé par un temps de propagation T_p .

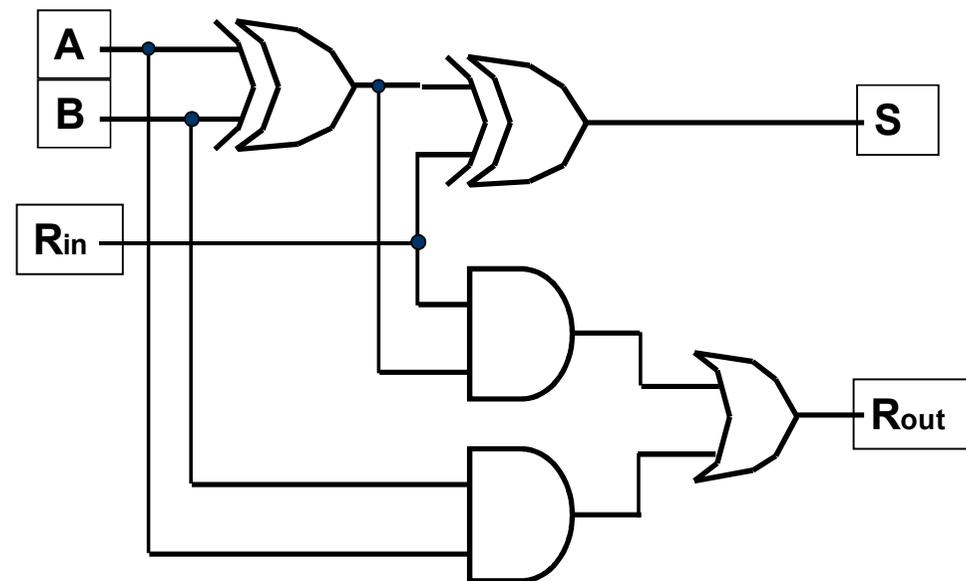


PROBLEMES ?

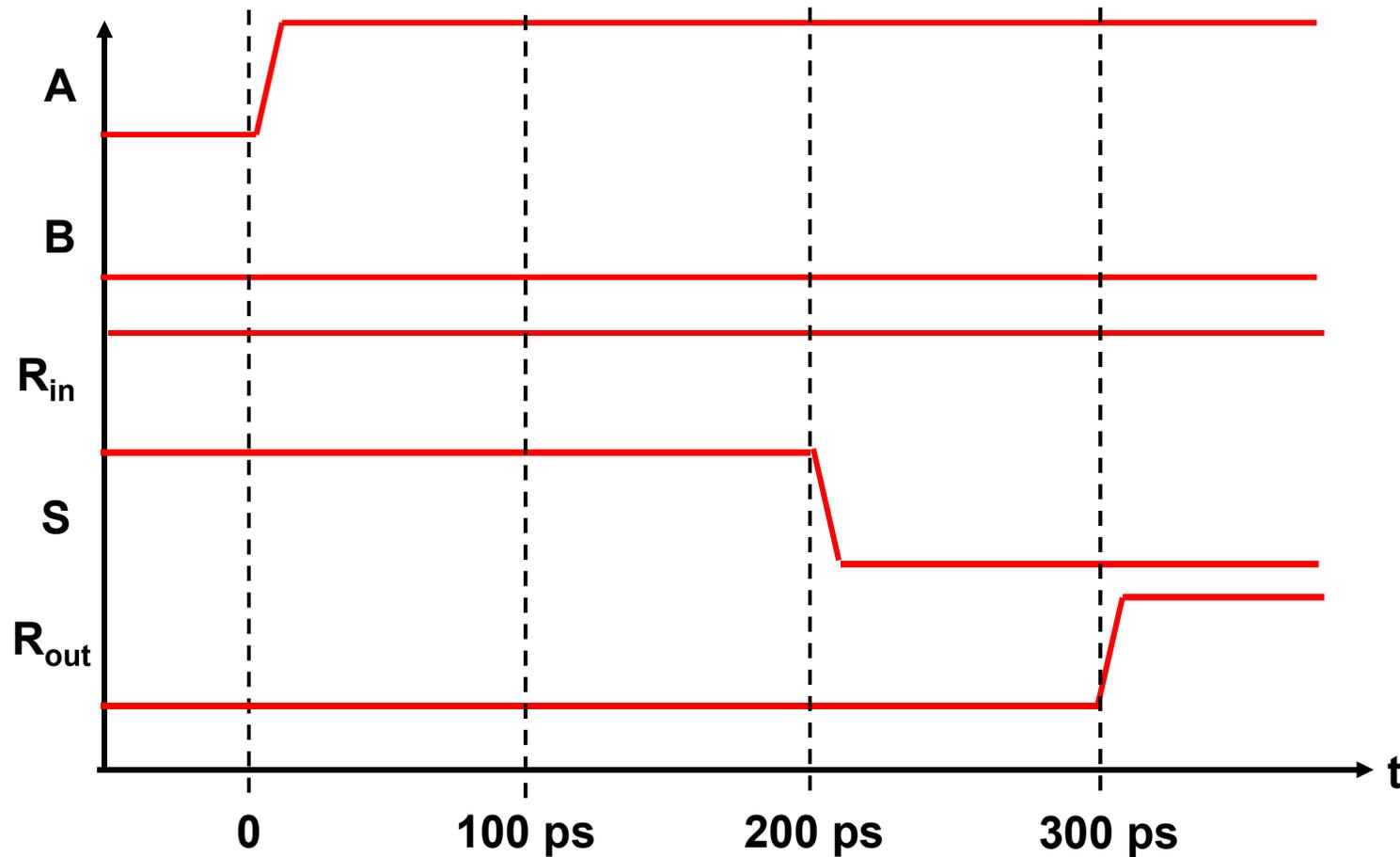
Temps de commutation et de propagation

Soit l'additionneur suivant. Les portes ont un temps de propagation $T_p = 100$ ps. A $t < 0$, les signaux d'entrée sont : $A = 0$, $B = 0$, $R_{in} = 1$. A $t = 0$, le signal A passe à l'état 1.

- Tracez l'évolution temporelle de S et Rout.
- Quel est le chemin critique ?
- Comment stabiliser le résultat du calcul pour le circuit en aval ?
- Quelle serait la fréquence maximale de fonctionnement d'un tel circuit ?

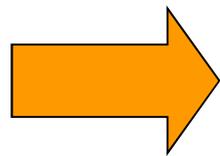
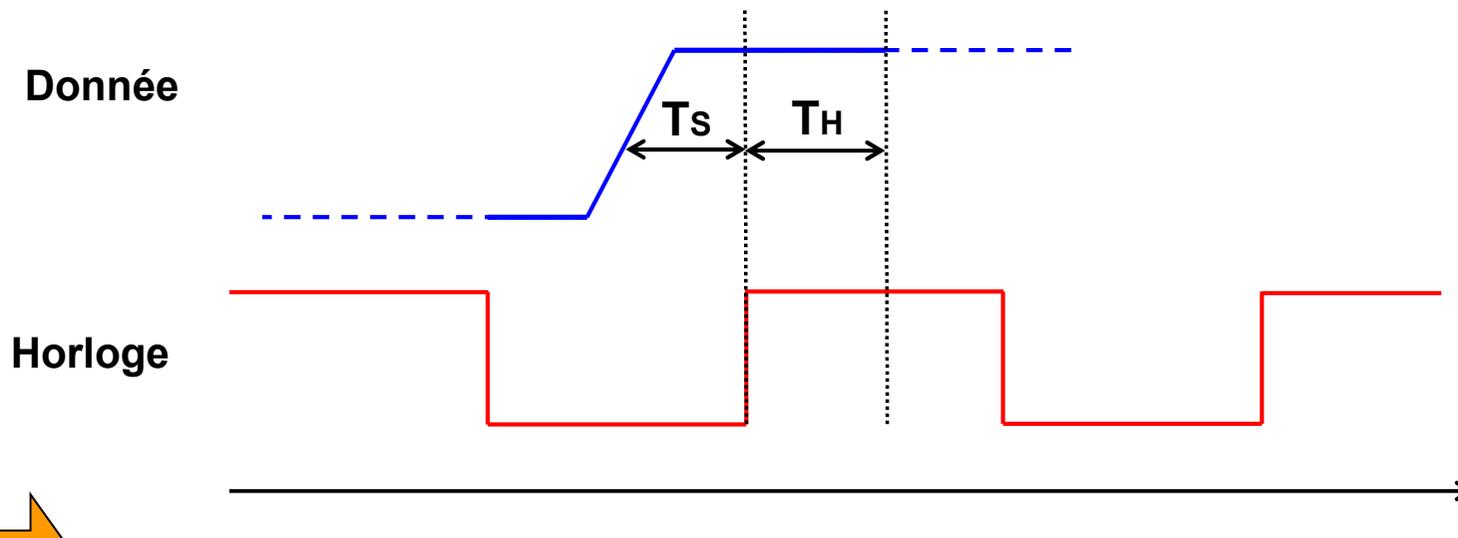


Temps de commutation et de propagation



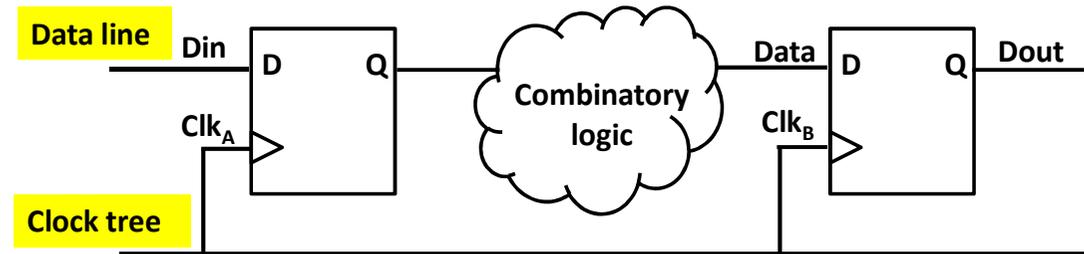
Temps de stabilisation / de maintien

- Les limitations temporelles des circuits numériques imposent le respect de délais à assurer entre les signaux afin d'éviter toute erreur logique :
 - ✓ Temps de stabilisation T_S (setup time)
 - ✓ Temps de maintien T_H (hold time)

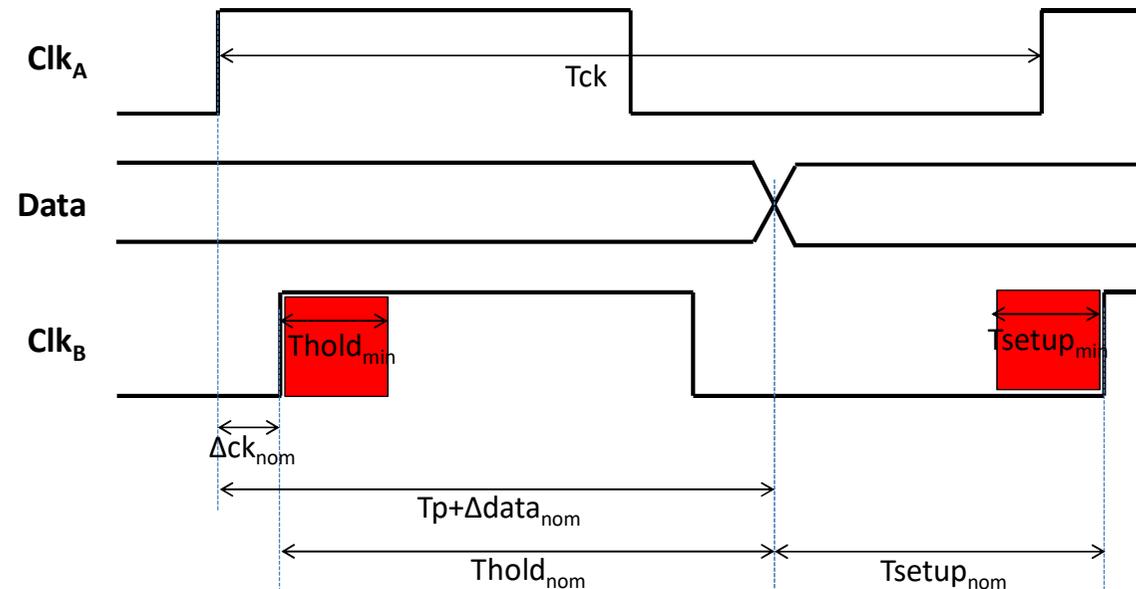


Limitation fréquentielle des circuits

Temps de stabilisation / de maintien



Timing diagram:



Fonctionnement assuré si les 2
marges suivantes sont positives :

$$M_{setup\ nom} = T_{setup\ nom} - T_{setup\ min} = T_{clk} + \Delta clk_{nom} - T_P - \Delta data_{nom} - T_{setup\ min} > 0$$

$$M_{hold\ nom} = T_{hold\ nom} - T_{hold\ min} = T_P + \Delta data_{nom} - \Delta clk_{nom} - T_{hold\ min} > 0$$

Temps de stabilisation / de maintien

On considère un circuit digital synchrone dont la profondeur logique max est égale à 10. La profondeur logique min est de 3. Le temps de propagation pour une porte logique ou une bascule est de 400 ps.

Le temps de stabilisation minimum est de 400 ps, tandis que le temps de maintien doit être d'au moins 300 ps. Le clock skew max. est estimé à +/- 40 ps.

On fait fonctionner ce circuit avec une fréquence d'horloge de 100 MHz.

Dans cette configuration, le circuit peut-il fonctionner sans violation des contraintes temporelles ?

Quelle est la fréquence maximale de fonctionnement ?

Temps de stabilisation / de maintien

Le chemin critique (chemin le plus lent dans le bloc de logique combinatoire) composé de 10 portes logiques de même temps de propagation TP : $dDatamax = 10 \cdot 0.4 = 4.4 \text{ ns}$.

Vérifions qu'à 100 MHz le circuit peut fonctionner sans erreur :

- Calcul du temps de set-up : $T_{setup\ nom} = T_{ck} + d_{ck} - T_p - d_{Datamax} = 10 + (+/- 0.04) - 0.4 - 4.4 = 5.64 \text{ ou } 5.56 \text{ ns}$
- Marge de setup : $M_{setup} = T_{setup\ nom} - T_{setup\ min} = 5.56 - 0.4 = 5.16 \text{ ns} > 0 \rightarrow$ pas de violation temporelle liée au temps de setup.
- Calcul du temps de hold : $T_{hold\ nom} = T_p + d_{Datamax} - d_{Ck} = 0.4 + 1.2 - (+/- 0.04) = 1.56 \text{ ou } 1.64 \text{ ns}$
- Marge de hold time : $M_{hold} = T_{hold\ nom} - T_{hold\ min} = 1.56 - 0.3 = 1.26 \text{ ns} \rightarrow$ pas de violation temporelle liée au temps de hold.

Le circuit fonctionne correctement à 100 MHz.

Temps de stabilisation / de maintien

Fréquence max → limite à partir de la quelle une violation temporelle apparait.

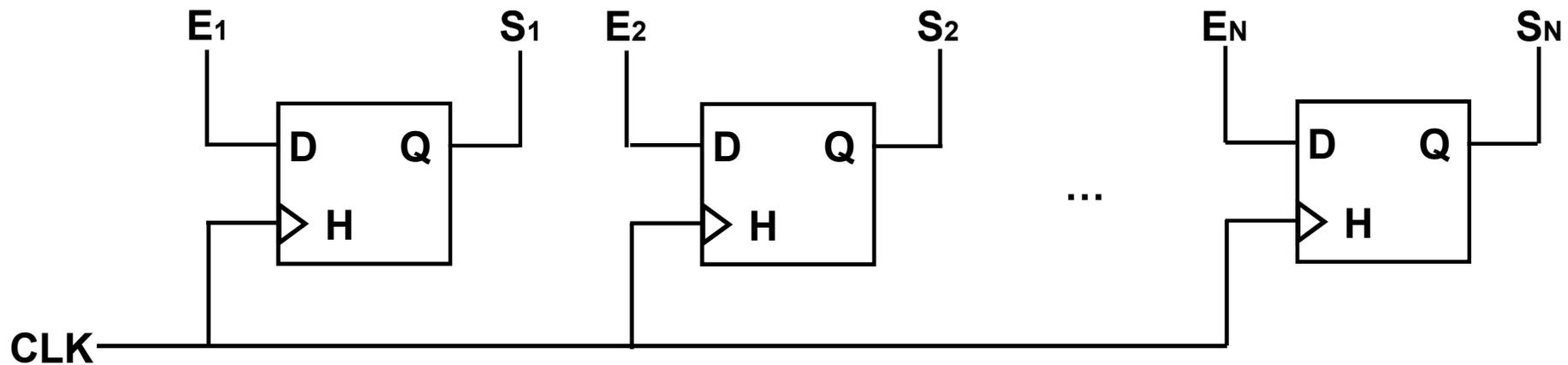
➤ La période d'horloge n'influe que sur le temps de set-up donc la fréquence max est liée à la marge de setup qui doit être positive :

➤ $T_{ck} + d_{Ck} - T_p - d_{Data_{max}} - T_{setup_{min}} > 0 \rightarrow T_{ck} > T_p + d_{Data} + T_{setup_{min}} - d_{Ck} =$
 $0.4 + 10 \cdot 0.4 + 0.4 - (+/- 0.04) = 4.84 \text{ ns}$

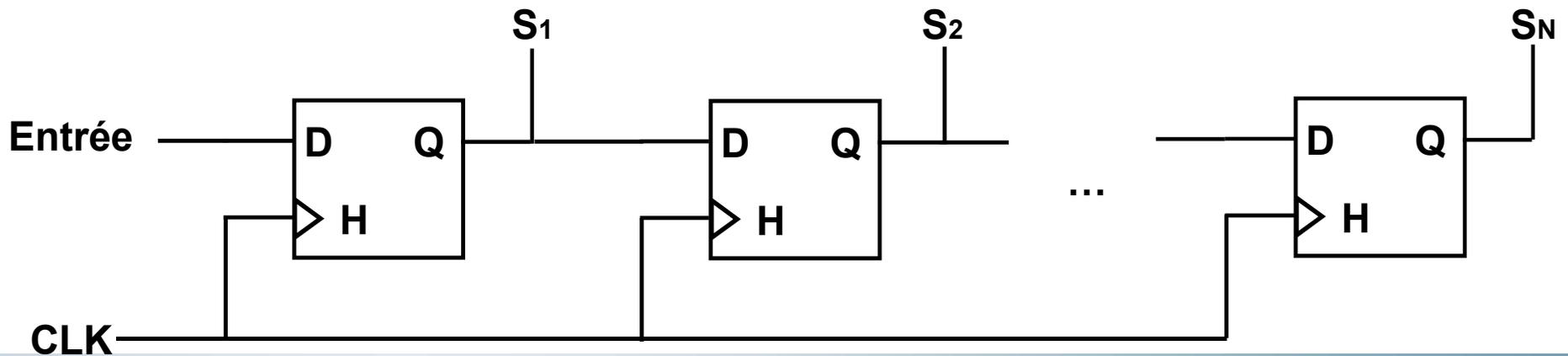
La période min d'horloge est donc égale à 4.84 ns d'où une fréquence max d'horloge de 206.61 MHz.

8. Circuits numériques – Fonctions de base

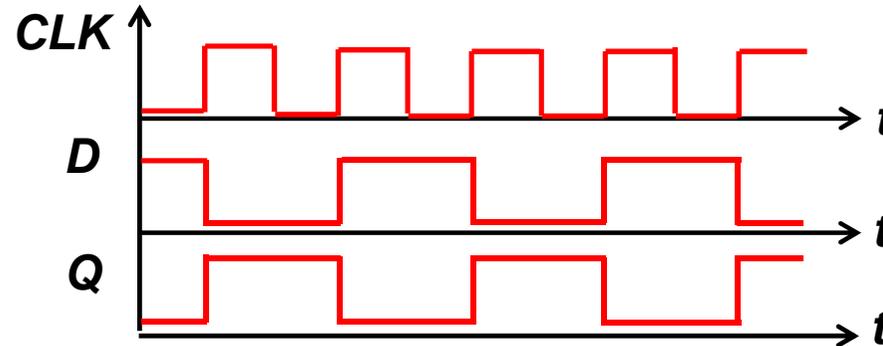
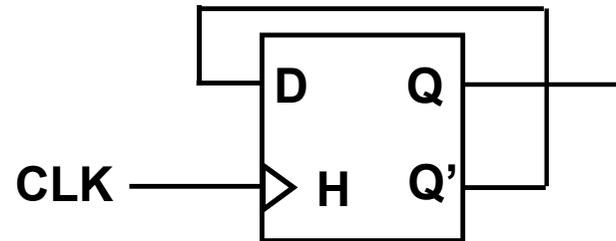
- Registre (entrées parallèles)



- Registre à décalage (entrée série)



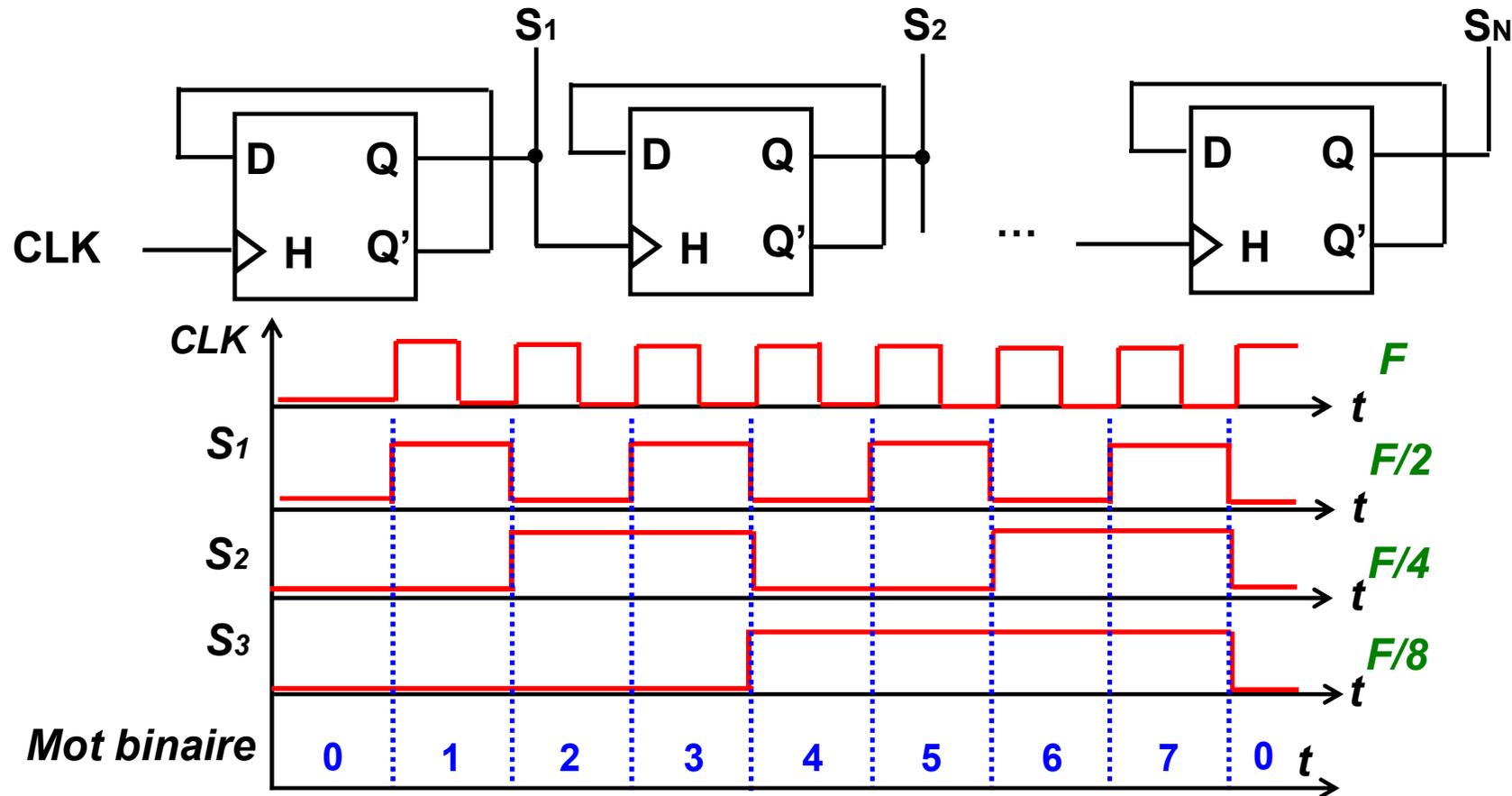
Diviseur de fréquence



$$F_D = F_{CLK}/2$$

Compteur asynchrone

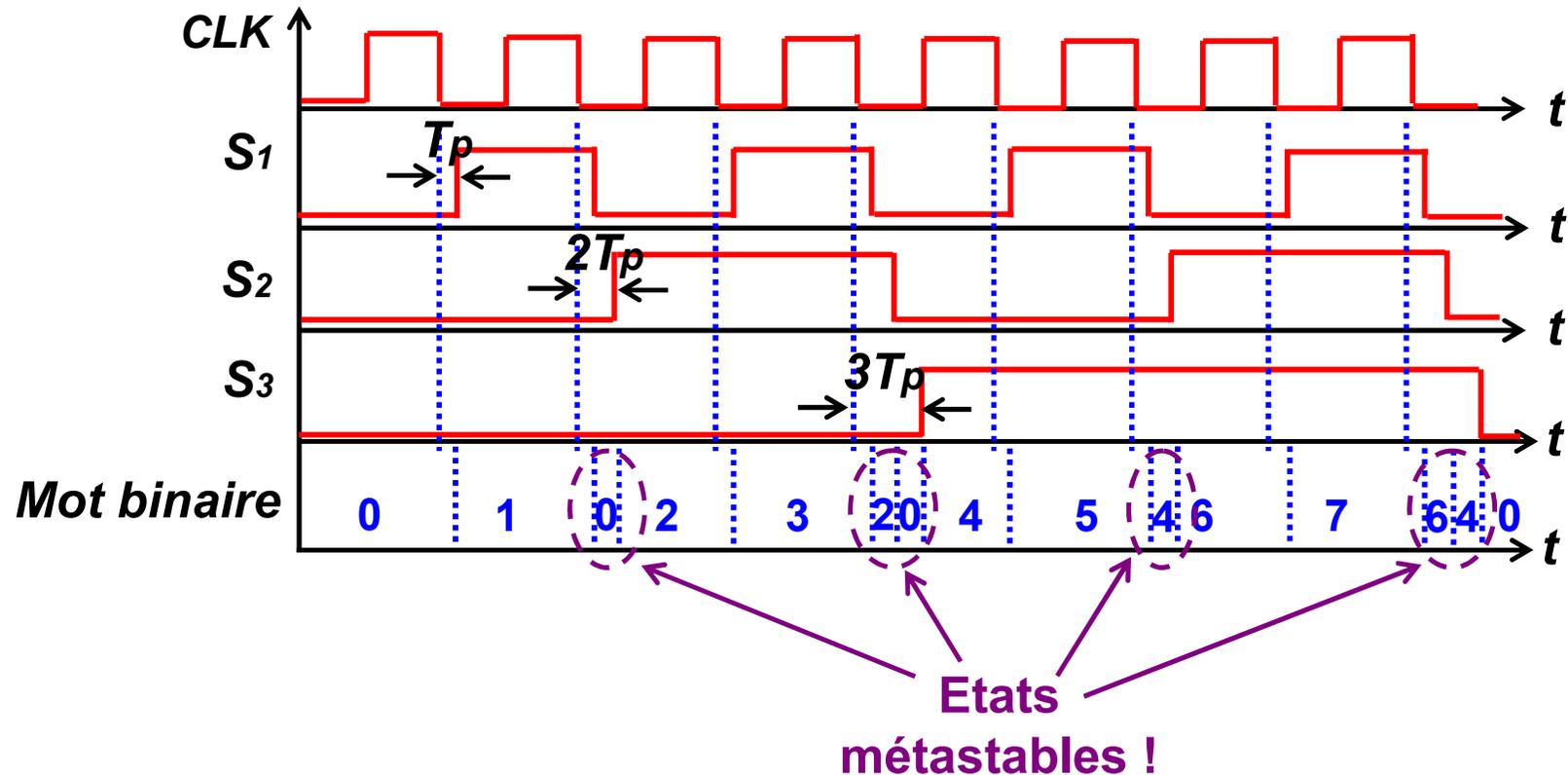
Qu'obtient-on si on chaine N diviseurs de fréquences ?



Comment réaliser un décompteur ?

Pourquoi est-il qualifié d'asynchrone ? Quelle est la conséquence ?

Compteur asynchrone - aléas



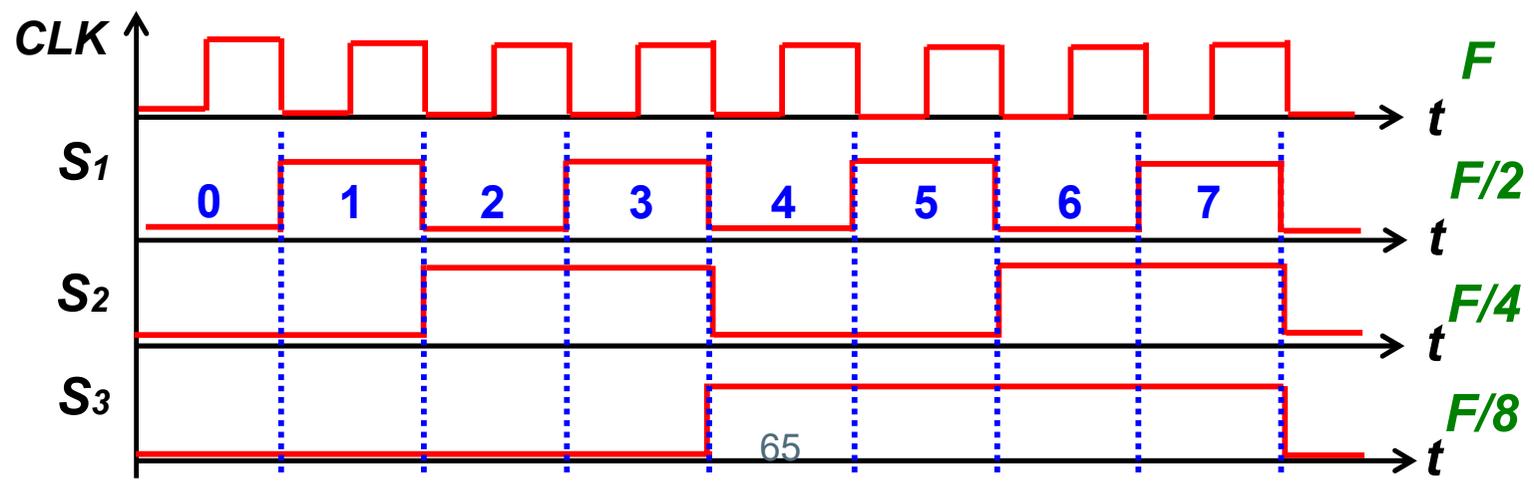
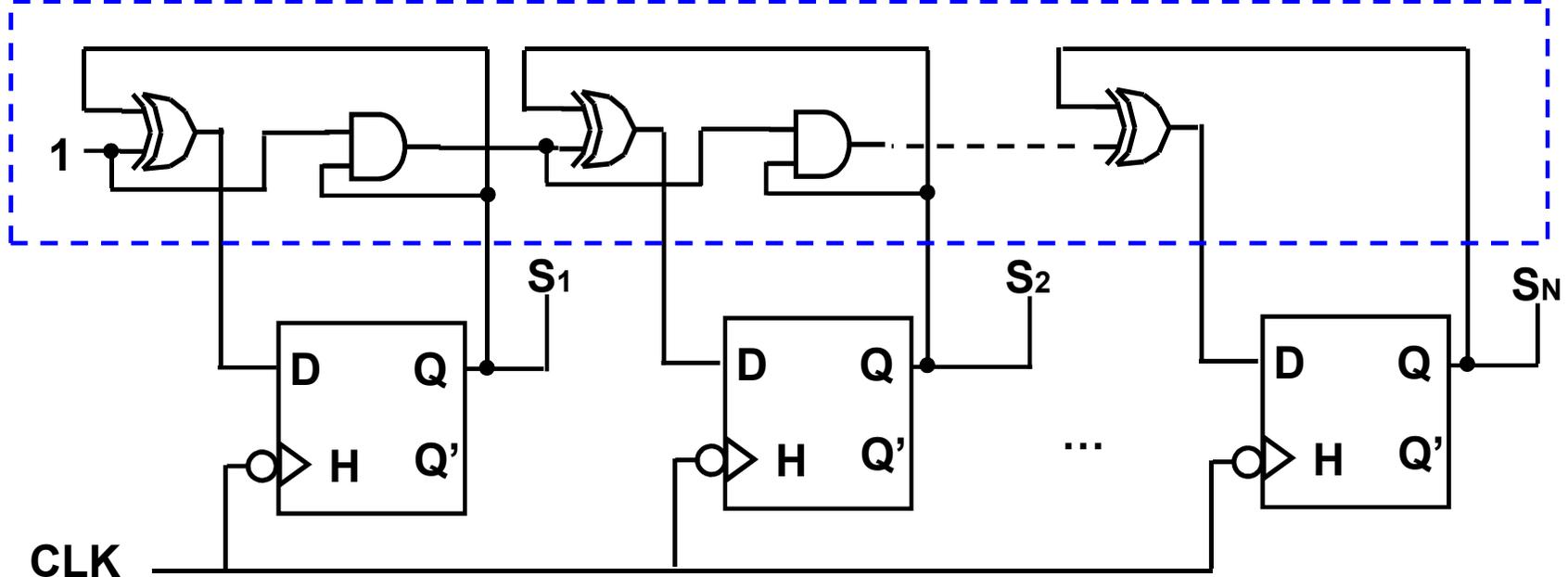
Fréquence max. de fonctionnement pour un compteur à N bascules :

$$F_{max} = \frac{1}{(N - 1) \times T_p}$$

$$F_{max} = \frac{1}{T_{pD} + T_{pXOR} + T_{pAND}}$$

Compteur synchrone

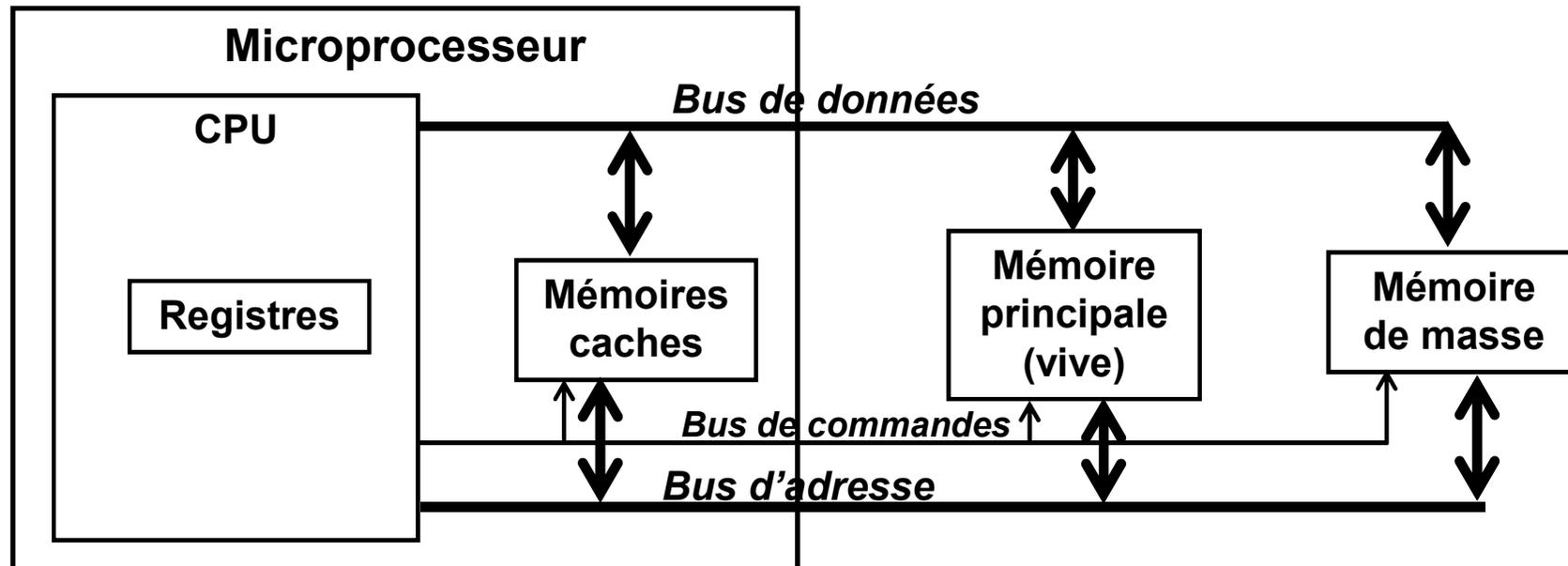
Logique combinatoire



Types de mémoire

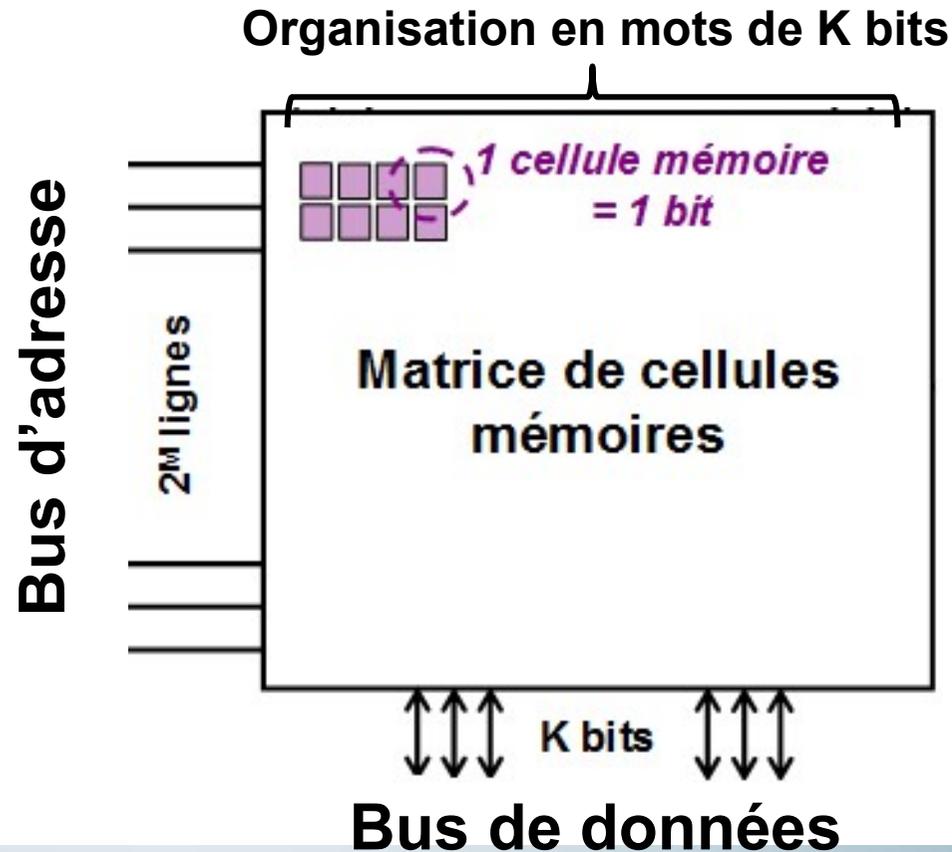
| Type | Volatile | Réinscriptible | Rafraichissement | Vitesse | Volume |
|---------------|----------|---------------------------------------|------------------|---------|--------|
| DRAM | Oui | Oui | Oui | + | ++ |
| SRAM | Oui | Oui | Non | +++ | + |
| DDR- SDRAM | Oui | Oui | Oui | ++ | ++ |
| EEPROM | Non | Oui ($10^5 - 10^6$) par secteurs | Non | + | 0 |
| Flash | Non | Oui (10^5) | Non | ++(+) | ++ |
| HDD | Non | Oui | Non | 0 | +++ |

Hierarchie de mémoires



Mémoire

- ✓ Une mémoire est une matrice de cellules de mémorisation.
- ✓ L'emplacement des données dans la mémoire est repérées par des adresses.



Adressage mémoire

Soit une mémoire pouvant stockée 65536 mots de 16 bits. L'adressage se fait par mot. La lecture et l'écriture se font par bloc de 16 mots. Elle présente en outre une contrainte d'alignement : seules des adresses multiples de 16 sont accessibles.

1. Combien de bits sont nécessaires pour l'adressage ?
2. Quelles sont les valeurs min et max de l'adresse ?
3. Quel est l'intérêt de lire/écrire par blocs ? De la contrainte d'alignement ?
4. Peut-on écrire 2 blocs à partir de l'adresse 0x45EF ? 0xFFF0 ?
5. On souhaite écrire 16 blocs à partir de l'adresse 0xFF00. Quelle est l'adresse du dernier mot ?

Adressage mémoire

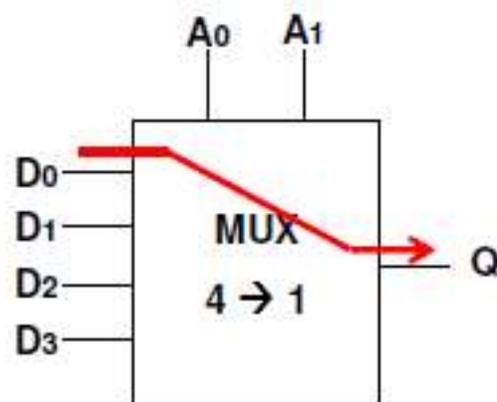
0x45EF → non : l'adresse en décimale est égale à 17903 qui n'est pas un multiple de 16.

0xFFFF0 → oui : l'adresse en décimale est égale à 65520 qui est un multiple de 16

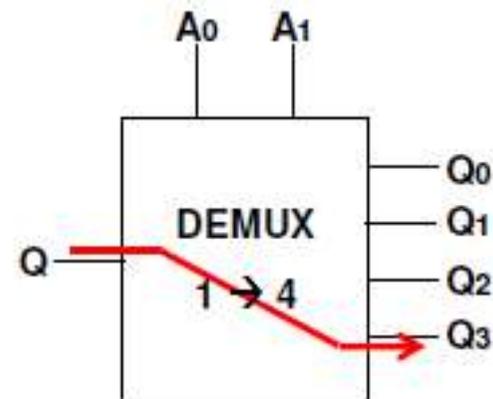
On peut écrire 16 blocs de 16 mots jusqu'à la limite de la mémoire (0xFFFF0 → 65535, donc le nb de blocs = $(65535 - 65520 + 1) / 16 = 16$). L'adresse du dernier mot est donc 0xFFFF.

Multiplexeur (Mux) / démultiplexeur (Demux)

- ✓ Rôle d'aiguilleur dans les circuits numériques : $N \rightarrow 1$ voie ou $1 \rightarrow N$ voies, où $N=2^M$
- ✓ Exemple :

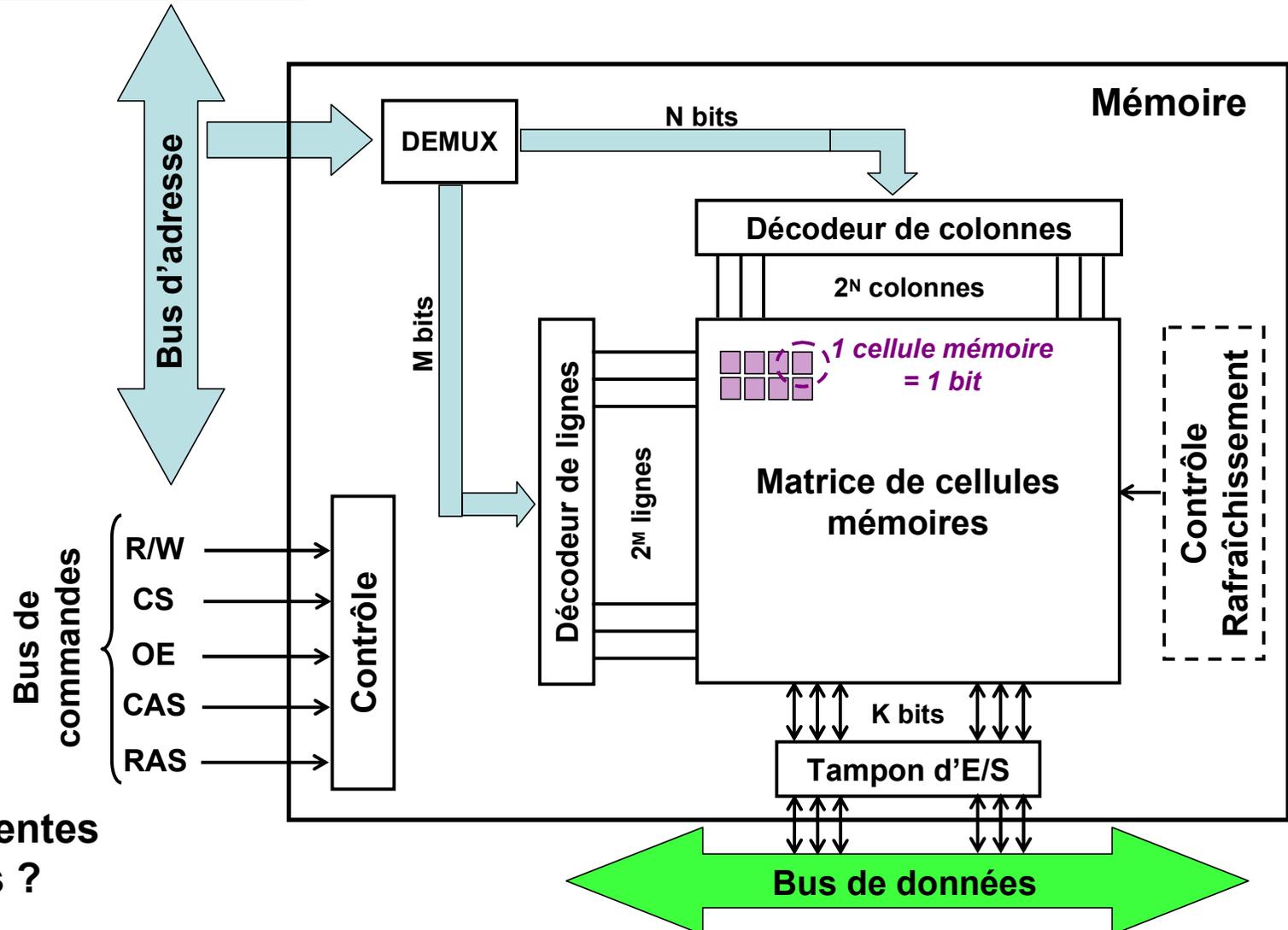


| A0 | A1 | Q |
|----|----|----|
| 0 | 0 | D0 |
| 1 | 0 | D1 |
| 0 | 1 | D2 |
| 1 | 1 | D3 |



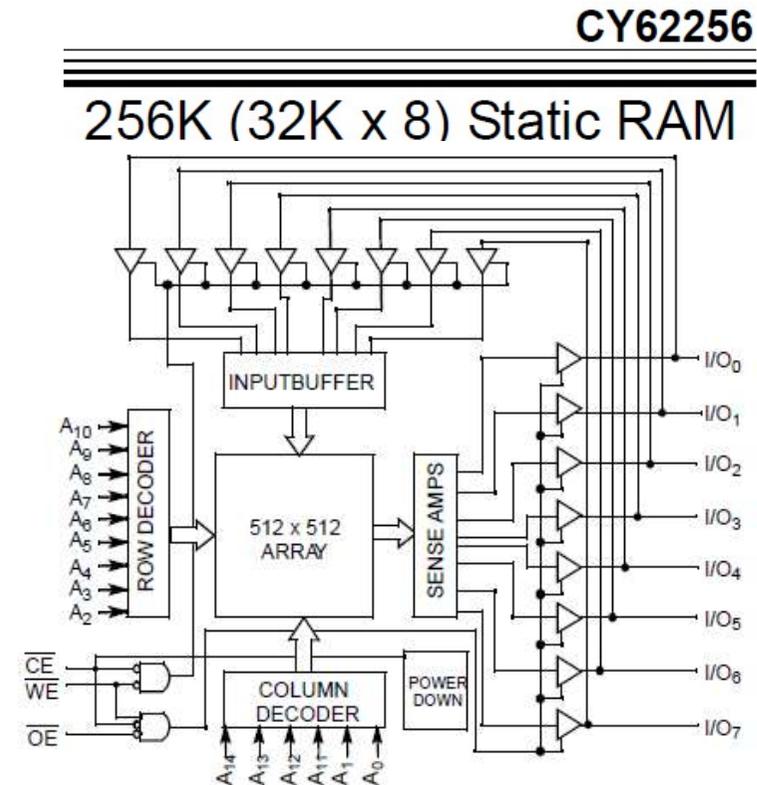
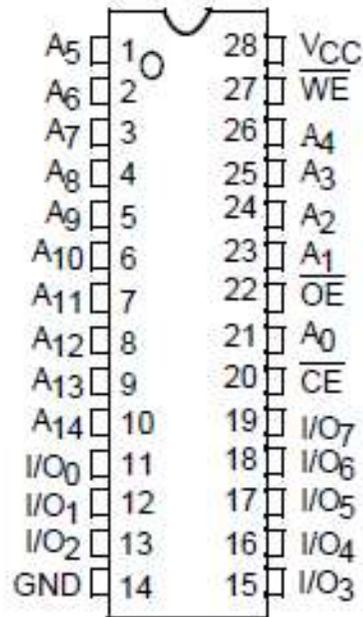
| A0 | A1 | Canal on |
|----|----|----------|
| 0 | 0 | Q0 |
| 1 | 0 | Q1 |
| 0 | 1 | Q2 |
| 1 | 1 | Q3 |

Architecture d'une mémoire



Rôle des différentes commandes ?

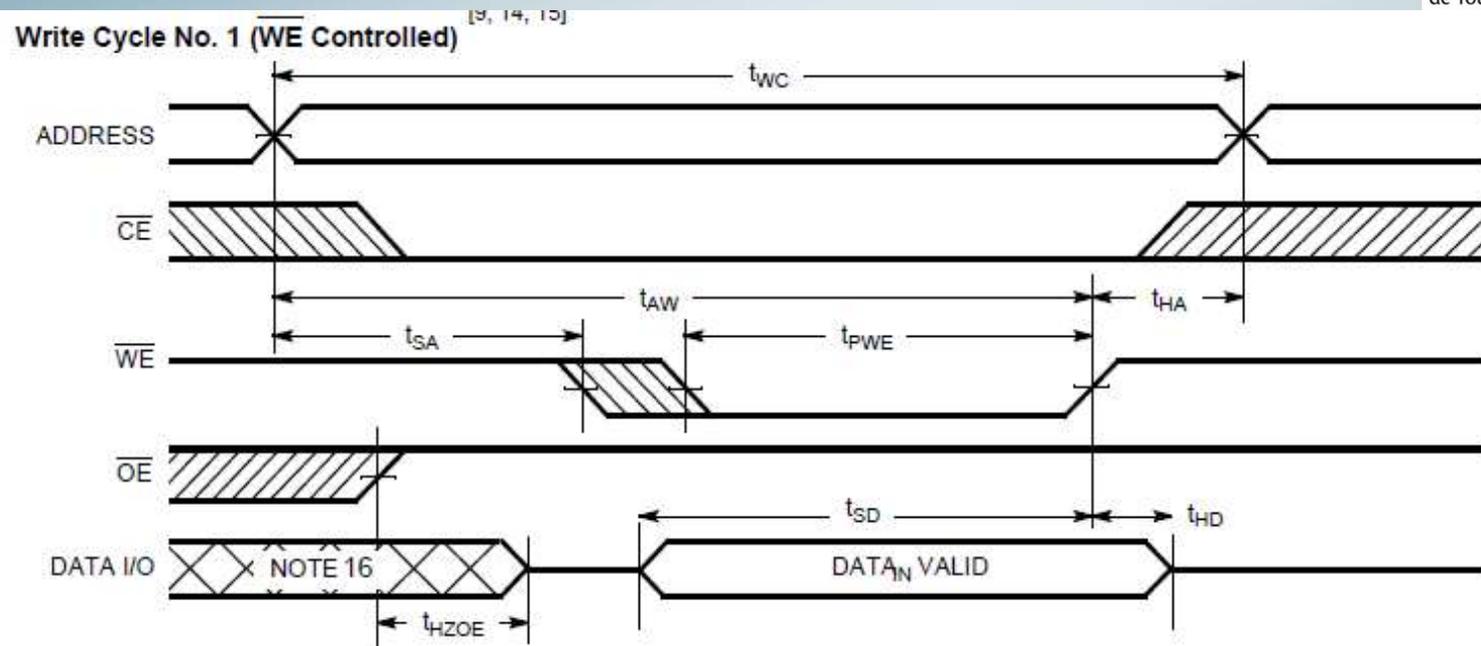
Exemple



Quelle est la taille exacte de la mémoire ?

Combien y a-t-il d'octets par ligne ? Par colonne ?

Exemple



| | | | |
|-----------|---|------------|--|
| T_{WC} | Write cycle time | T_{HA} | Address Hold from Write End (0 ns min) |
| T_{AW} | Address Set-up to Write End (60 ns min) | T_{HZOE} | OE HIGH to High-Z (25 ns max) |
| T_{SA} | Address Set-up to Write Start (60 ns min) | T_{SD} | Data Set-up to Write End (30 ns max) |
| T_{PWE} | /WE Pulse Width (50 ns min) | T_{HD} | Data Hold from Write End (0 ns min) |

Temps d'écriture minimum (/CE est laissé à 0) ?