

Conversion Analogique-Numérique et Numérique-Analogique

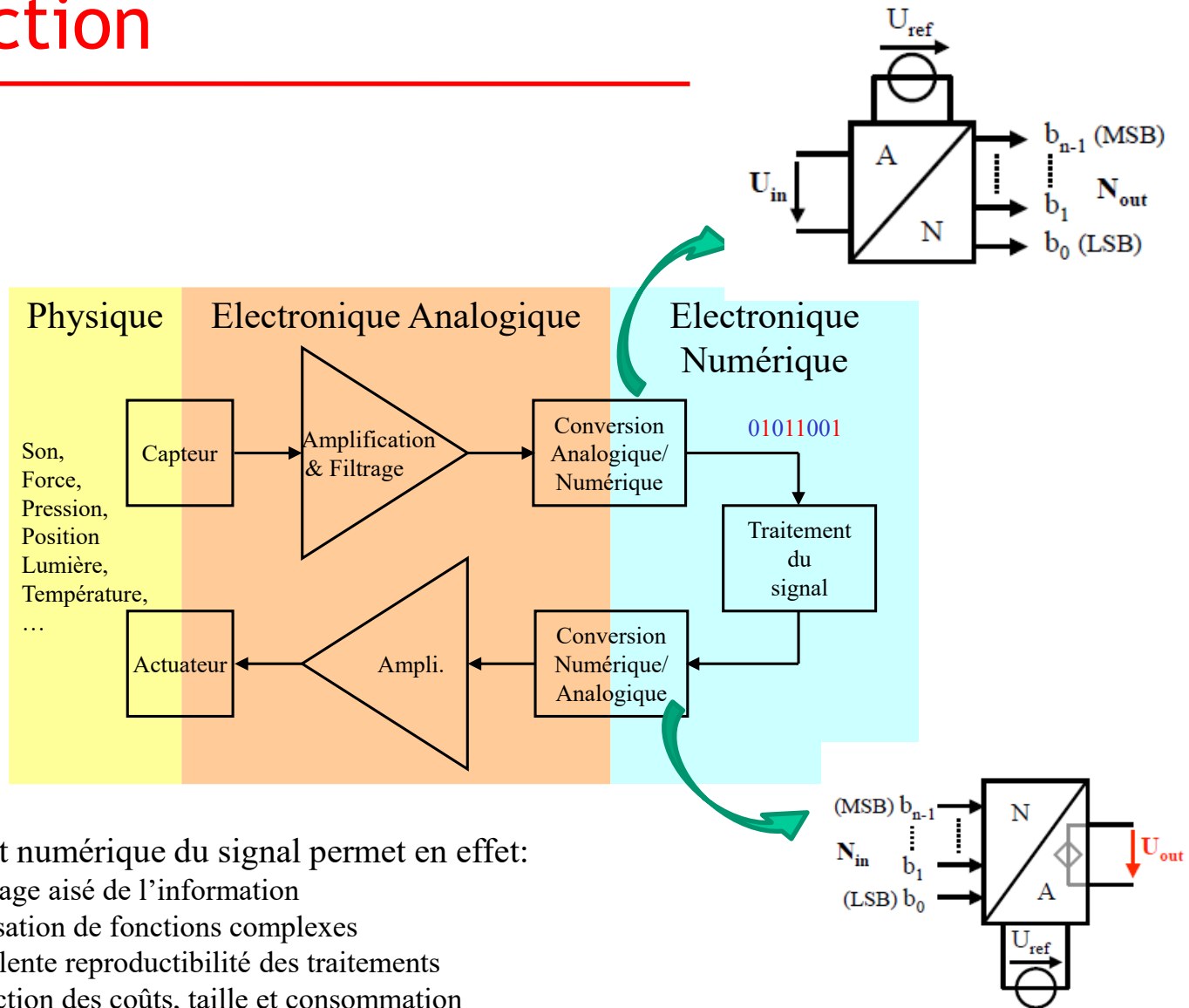
Systèmes électriques et électroniques

Adil KOUKAB

Sommaire

- Conversion Analogique-Numérique (CAN)
 - Echantillonnage
 - Quantification
- Architectures:
 - Convertisseur Flash
 - Convertisseur à approximations successives
- Conversion Numérique-Analogique-(CNA)
 - Génération de courants pondérés
 - Réseau R2R

Introduction



- Traitement numérique du signal permet en effet:
 - Stockage aisé de l'information
 - Réalisation de fonctions complexes
 - Excellente reproductibilité des traitements
 - Réduction des coûts, taille et consommation

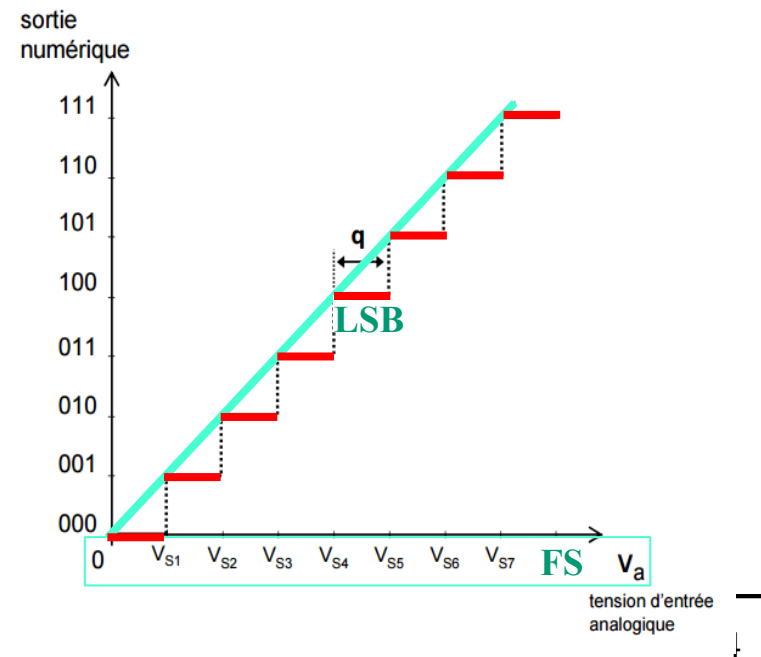
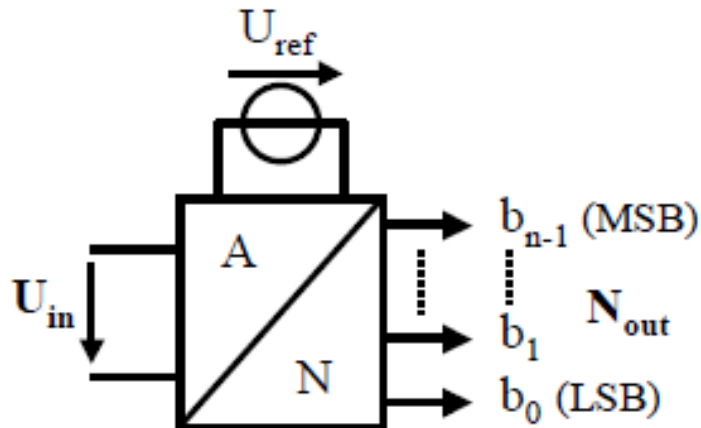
Conversion Analogique/Numérique: CAN

- Un convertisseur analogique/numérique transforme une grandeur analogique (**Ex la tension U_{in}**) en un nombre, généralement codé en binaire sur n bits (**Ex. $N_{out} \equiv (b_{n-1}, \dots, b_1, b_0)$** avec $b_i = 0$ ou 1) suivant la relation:

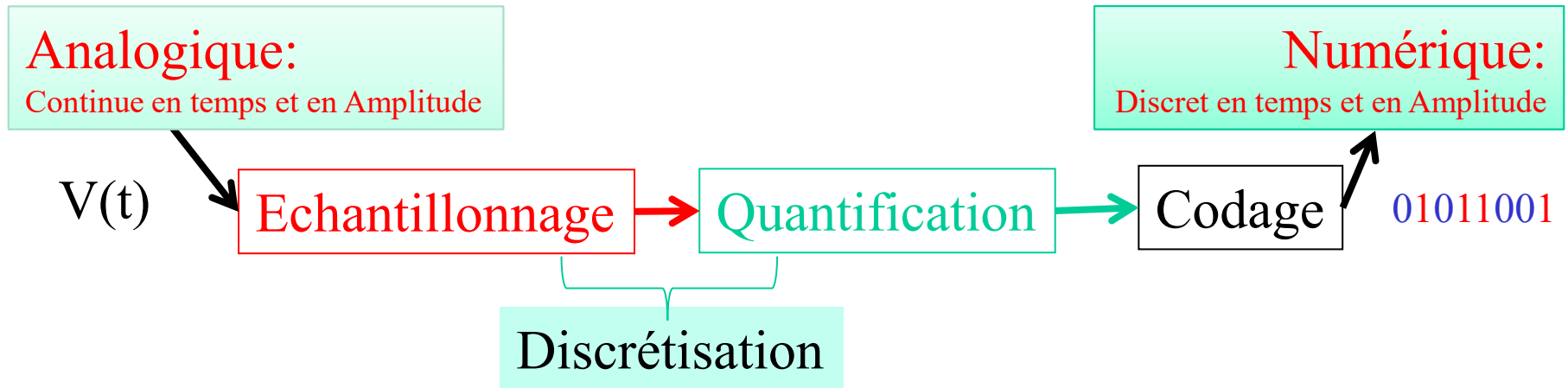
$$N_{out} = b_{n-1} 2^{n-1} + \dots + b_1 2^1 + b_0 2^0 = \text{Arrondie}(U_{in}/\text{LSB})$$

LSB (pour Least Significant Bit) = pas de quantification “ $\equiv$ l’unité de mesure”.

FS (Full Scale) = Pleine Echelle “ $\equiv$ gamme de mesure ici spécifiée par U_{ref} ”.



Conversion Analogique/Numérique

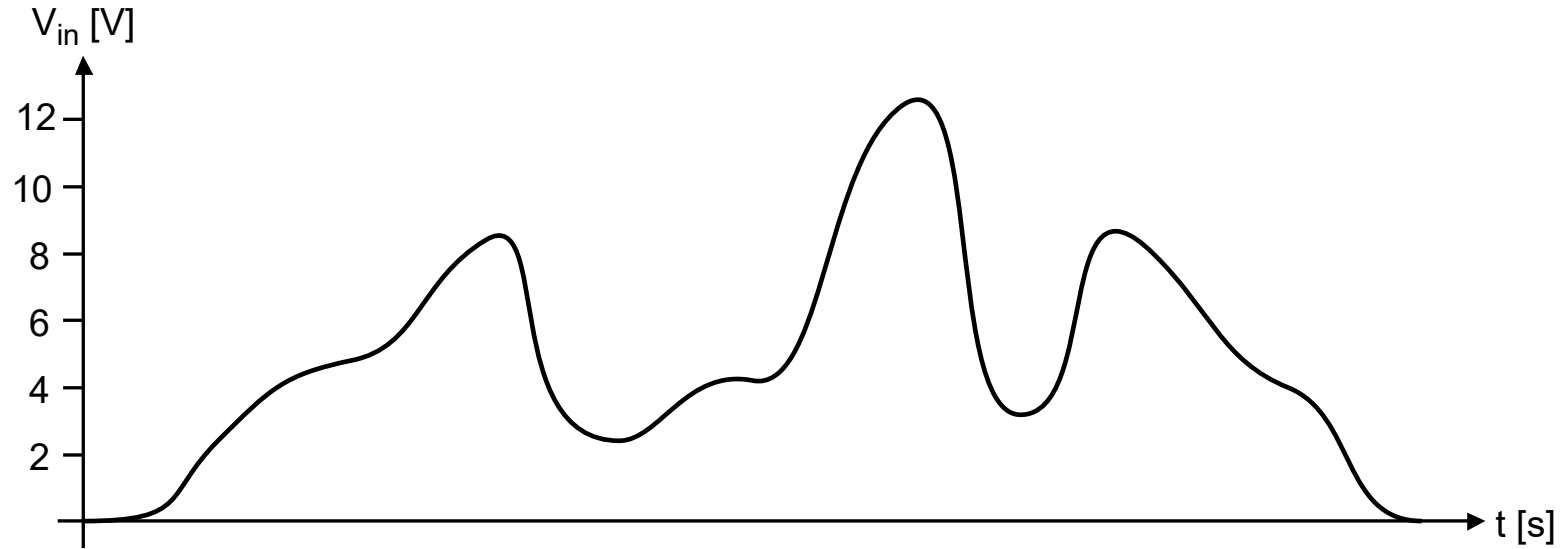


- Discrétisation
 - Echantillonnage
 - Quantification (L niveaux $L=2^N$):
 - N = nombre de bits du CAN

Codage

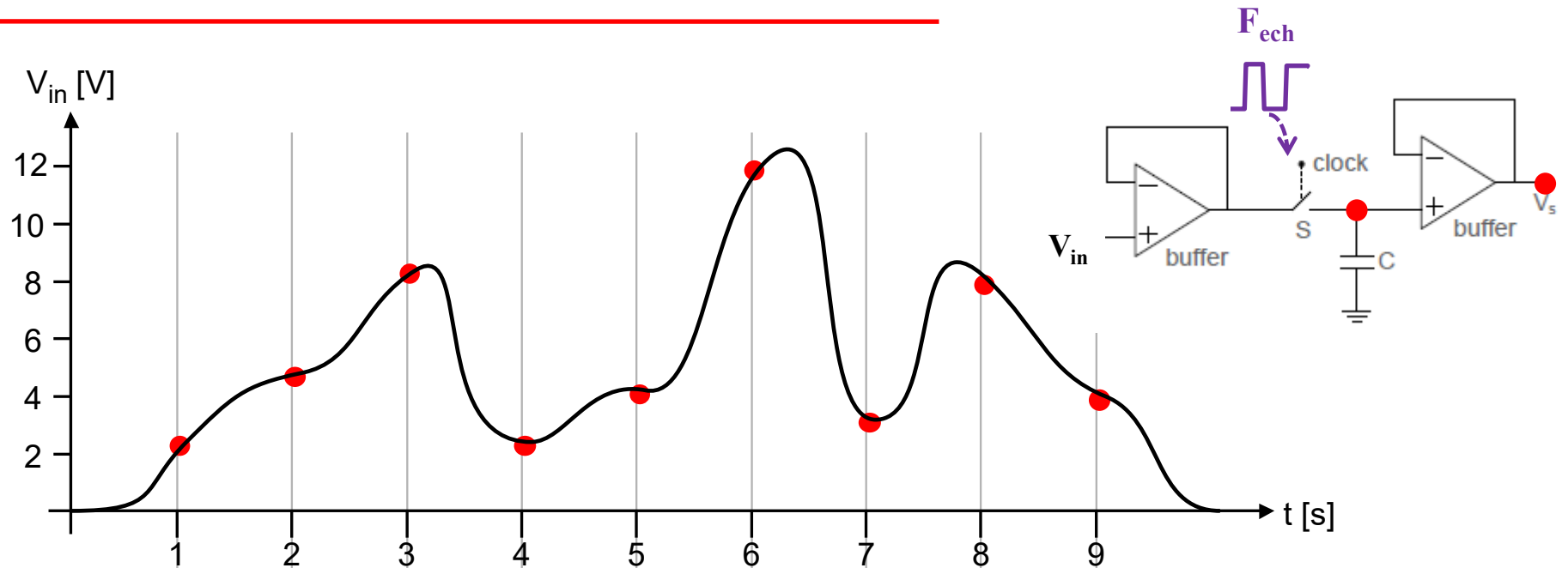
$$\text{Ex: } (L=8, N=3) \quad \mathbf{x} \equiv \mathbf{b}_2 \mathbf{b}_1 \mathbf{b}_0 \quad \text{c.à.d } x = \mathbf{b}_2 2^2 + \mathbf{b}_1 2^1 + \mathbf{b}_0 2^0$$

Conversion A/N



- Discrétisation du signal:
 - En *temps*: **Echantillonnage**
 - En *amplitude*: **Quantification**

Conversion A/N: Echantillonnage



- Le signal d'entrée est mesuré périodiquement
 - Frequence ou période d'échantillonnage constantes: $F_{ech}=1/T_{ech}$.

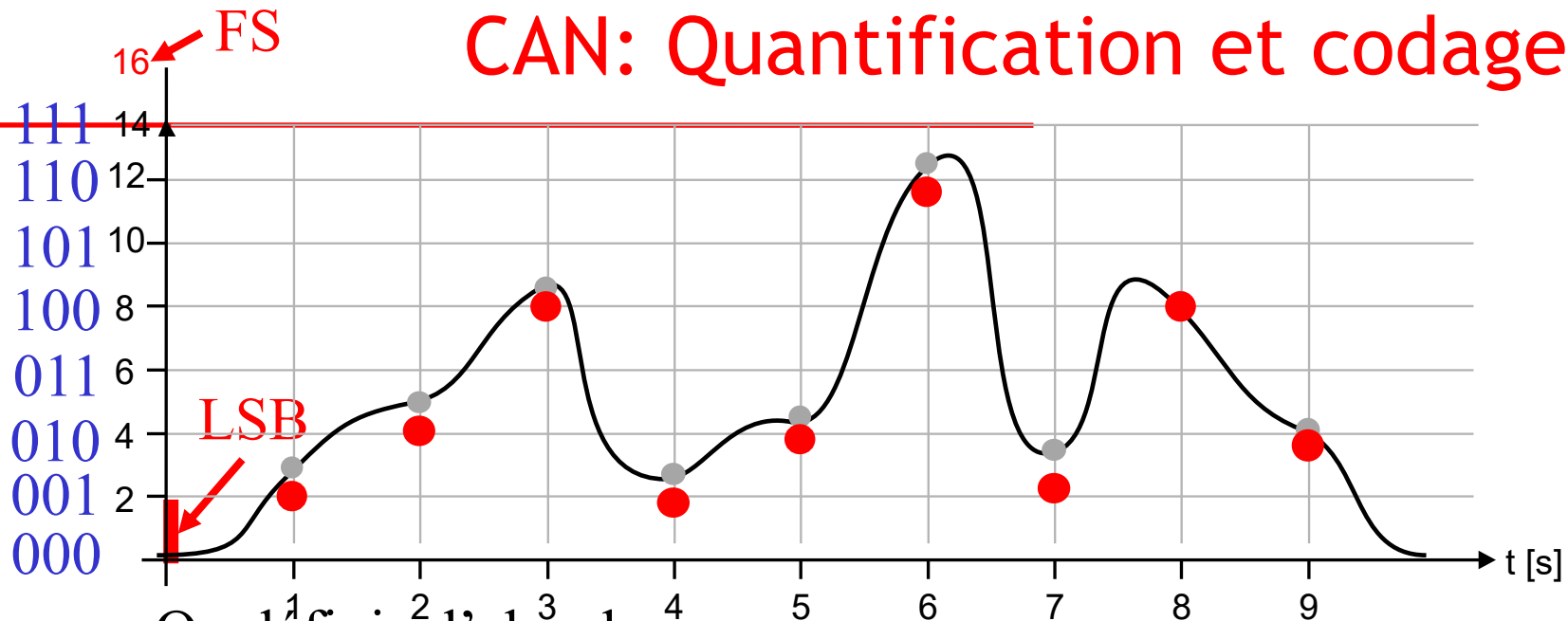
Théorème d'échantillonnage

- Théorème d'échantillonnage de Nyquist-Shannon:

$$F_{\text{échantillonnage}} > 2F_{\text{in;max}}$$

- Un signal $x(t)$ peut être représenté de manière univoque par une suite de valeurs échantillonnées si la fréquence d'échantillonnage, F_{ech} , est au moins deux fois plus élevée que la plus grande des fréquences, F_{max} , contenues dans le spectre.
 - Ex: fréquences audio que nous percevons s'étend de 20 Hz à 20 kHz: $\rightarrow$ le choix de la fréquence d'échantillonnage des CD est fixée à 44,1 kHz

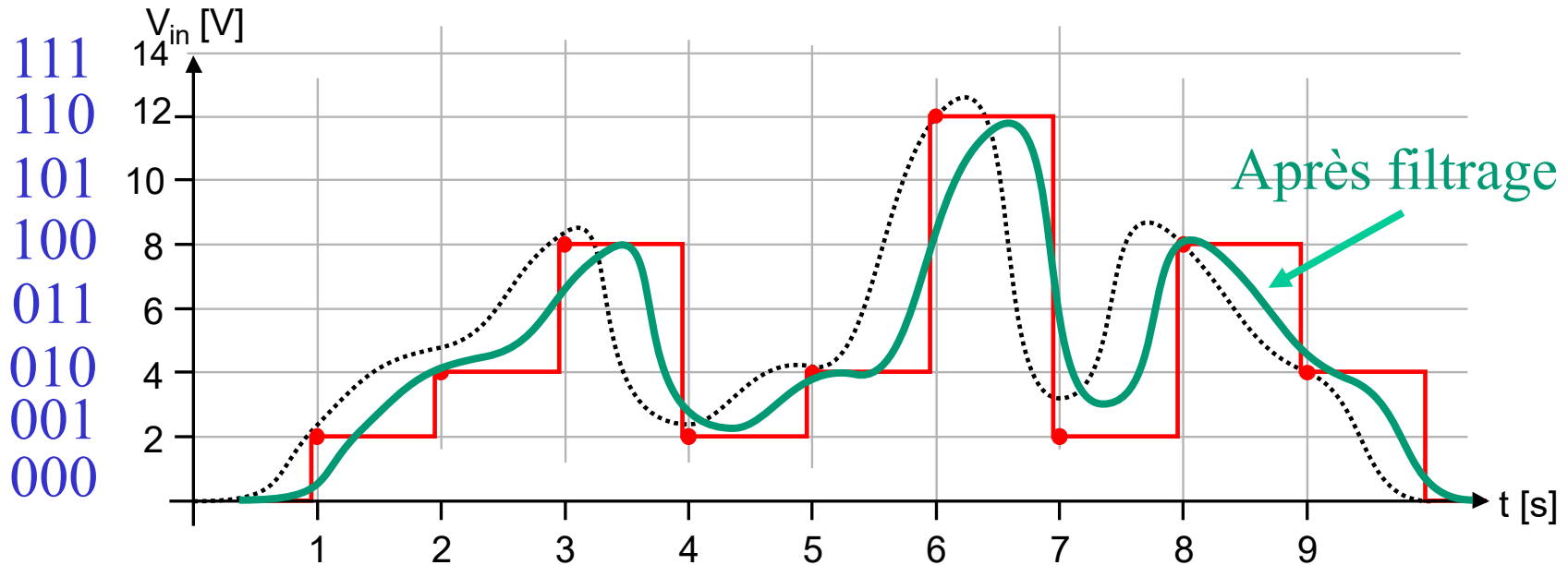
CAN: Quantification et codage



- On définit d'abord:
 - la plage de variation acceptable de la tension analogique d'entrée, appelée Pleine Echelle (FS pour Full-Scale en anglais; ici 16V).
 - Le pas de quantification (LSB pour Least Significant Bit; ici 2V): la précision du CAN.
 - $FS/LSB = 8 = 2^N$, avec N le nombre de bits du CAN aussi le nombre de codes binaires possible.
 - Pour chaque mesure, la valeur d'entrée est ramenée sur la valeur discrétisée (ici entière) immédiatement inférieure puis codé en binaire.

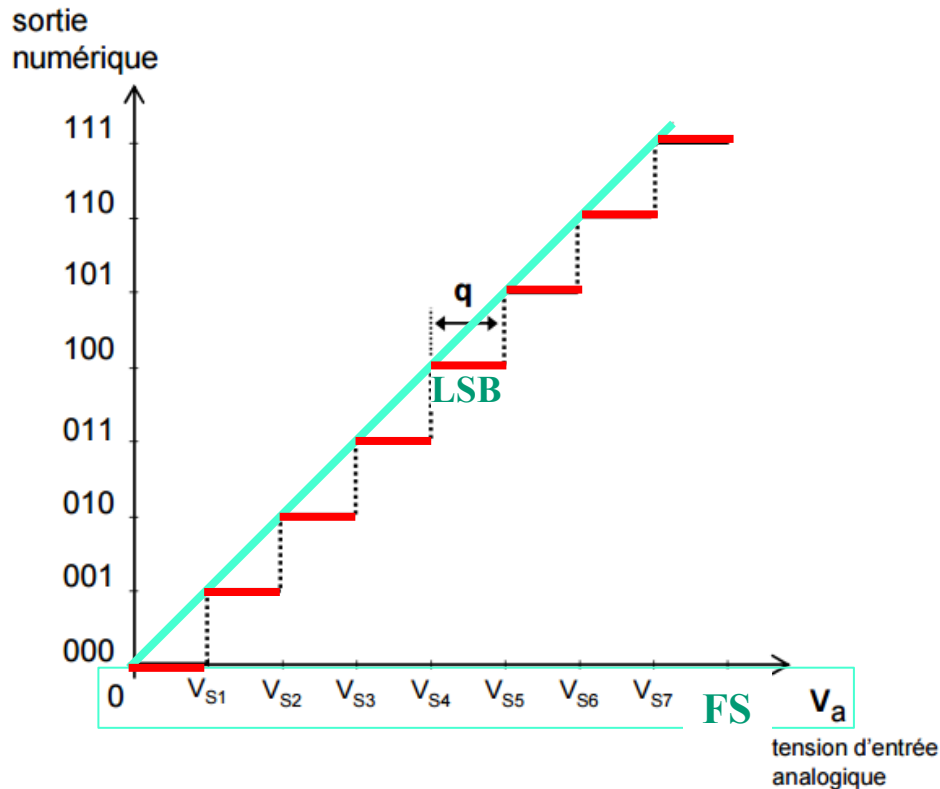
**Ex: pour FS=16V, LSB=2V, FS/LSB=8, N=3 et donc 0V $\equiv$ (000); 14V $\equiv$ (111)
11.4V $\equiv$? arrondi(11.4V/2V) = 5 = $1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \rightarrow 11.4V \equiv 101$.**

Signal numérique



- Le signal analogique peut être reconstruit par un convertisseur numérique/analogique et un filtre passe-bas.

Résolution du convertisseur



$$2^N = \frac{FS}{LSB}$$

$$N_{bits} = \log_2 \left(\frac{FS}{LSB} \right)$$

$$LSB = q$$
$$= \text{Err}_{\max} \text{ de Quantification}$$

- La plage d'entrée (Full scale) est discrétisée avec une certaine résolution.
- Plus les intervalles (LSB) sont petits (resp. N est grand), plus la mesure est précise.
LSB est dite erreur maximale de quantification ou de codage

Codage binaire

Binaire N = 3			Decimal
MSB (Most SB) b_2	b_1	LSB b_0	$Dec = \sum_{i=0}^{N-1} b_i 2^i$
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Codage thermométrique

Binaire N = 3			Décimal
Binaire N=3	Thermométrique $N_{th}=2^3-1$		$Dec = \sum_{i=0}^{Nth} b_{th,i}$
000	0000000		0
001	0000001		1
010	0000011		2
011	0000111		3
100	0001111		4
101	0011111		5
110	0111111		6
111	1111111		7

Sommaire

- Conversion Analogique-Numérique (CAN)
 - Echantillonnage
 - Quantification
- Architectures:
 - Convertisseur Flash
 - Convertisseur à approximations successives
- Conversion Numérique-Analogique-(CNA)
 - Génération de courants pondérés
 - Réseau R2R

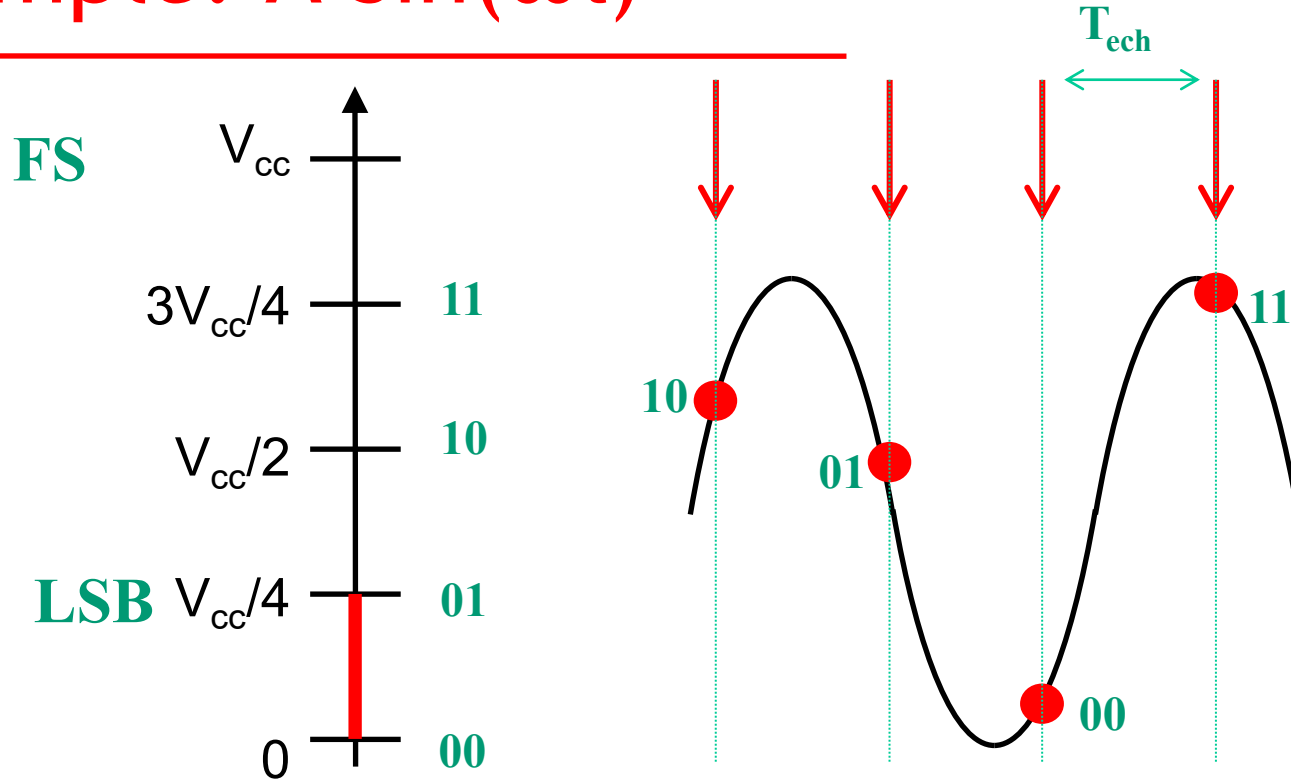
Convertisseurs Analogique-Numérique (CAN): Implémentations

- Différentes variantes possibles avec des compromis entre:
 - Vitesse de conversion
 - Précision
 - Consommation
 - Nombre de composants $\Leftrightarrow$ Taille du circuit $\Leftrightarrow$ Prix (\$)
- Exemples:
 - **Flash** (Très rapide, faible résolution 8 bits max, coûteux, très utilisé dans la vidéo).
 - **Approximations successives** (relativement rapide, précis 12bits-18bits et peu gourmand en énergie, utilisé un peu partout).
 - **Delta-sigma** (lent, très haute résolution 16-32 bits, très utilisé dans les capteurs).

Ex1: Convertisseur Flash

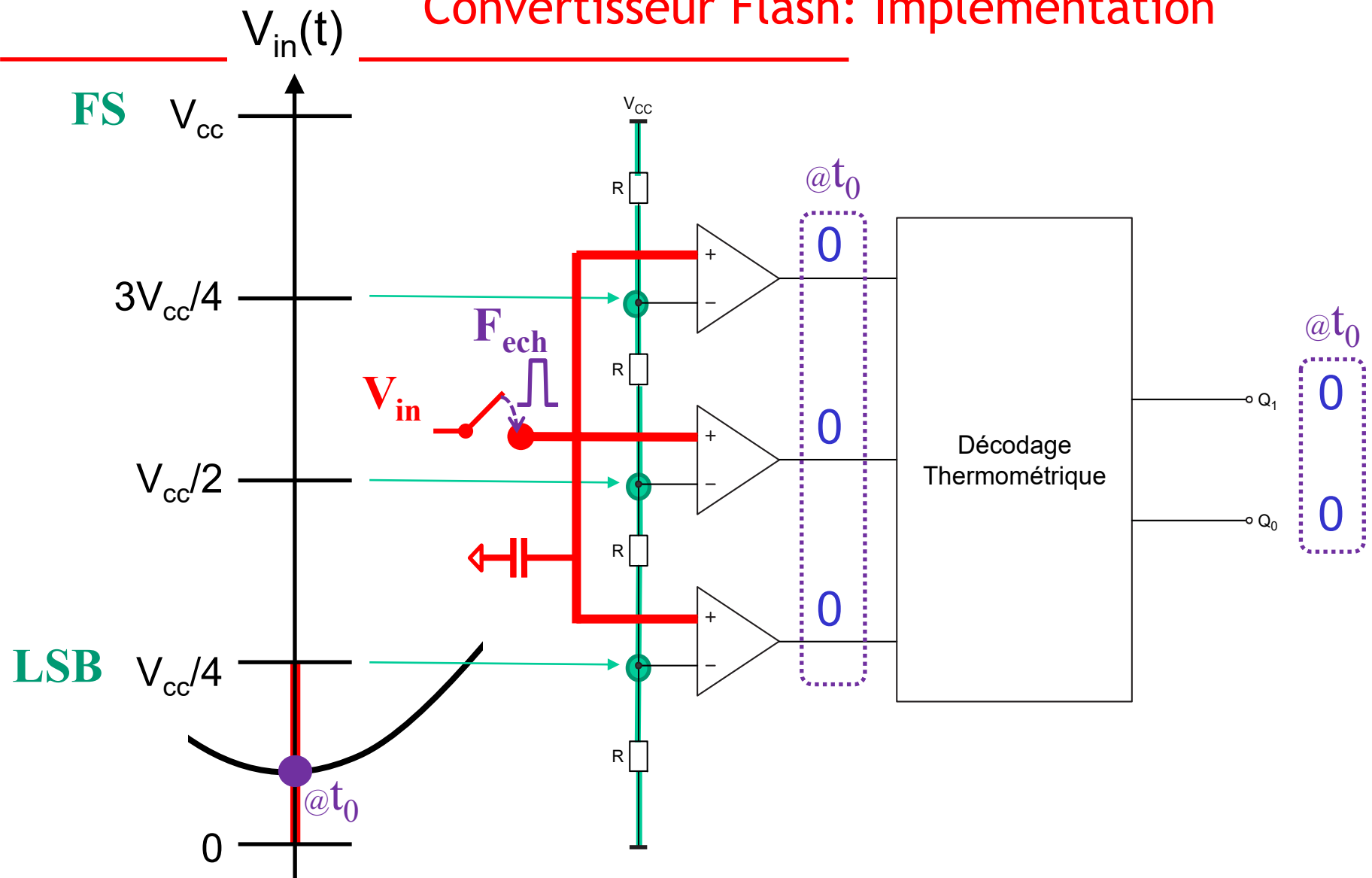
- Signal d'entrée comparé simultanément avec chaque seuil de quantification
- Nécessite $2^N - 1$ **comparateurs** et 2^N résistances
- **Conversion rapide**
- Résolution limitée en pratique à 8 bits (surface, consommation, pareillement, offset, bruit ...)

Cas simple: $A \sin(\omega t)$

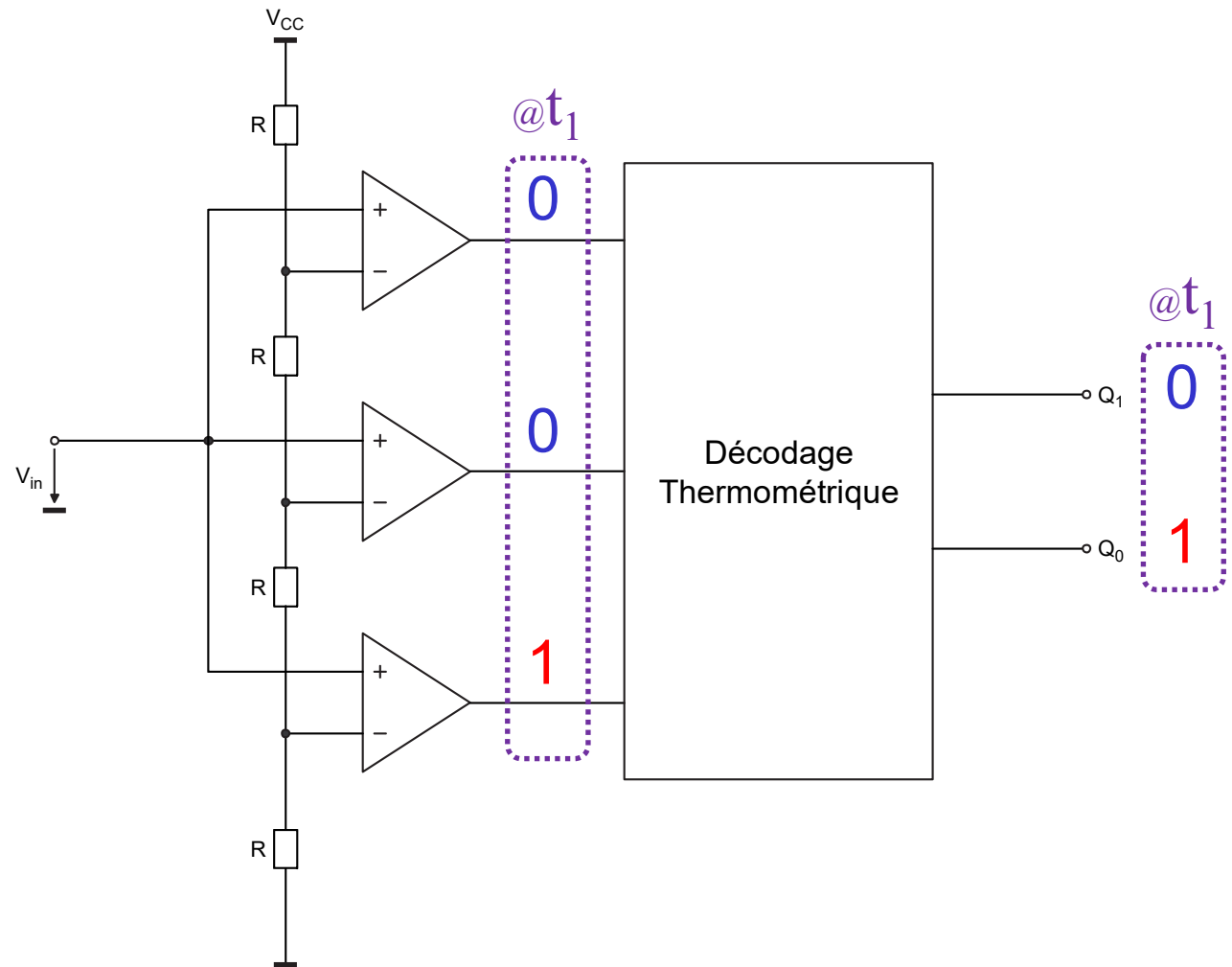
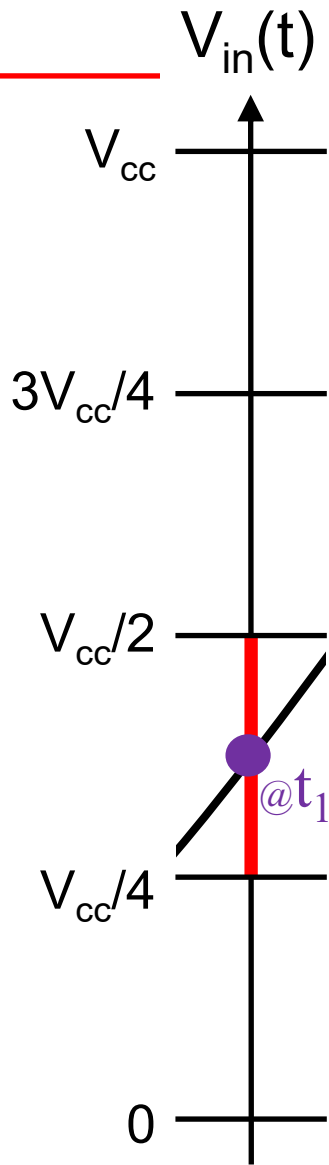


- On définit d'abord:
- la plage de variation de la tension analogique: $FS = V_{cc}$
- Le pas de quantification: $LSB = V_{cc}/4 \rightarrow$ l'erreur de quantification
- $FS/LSB = 2^N = 4 \rightarrow N$ le nombre de bits du CAN = 2; code binaire de 0 (00) à 3 (11)
- On définit aussi Fréquence d'échantillonnage : $1/T_{ech} > 2.F_{signal}$

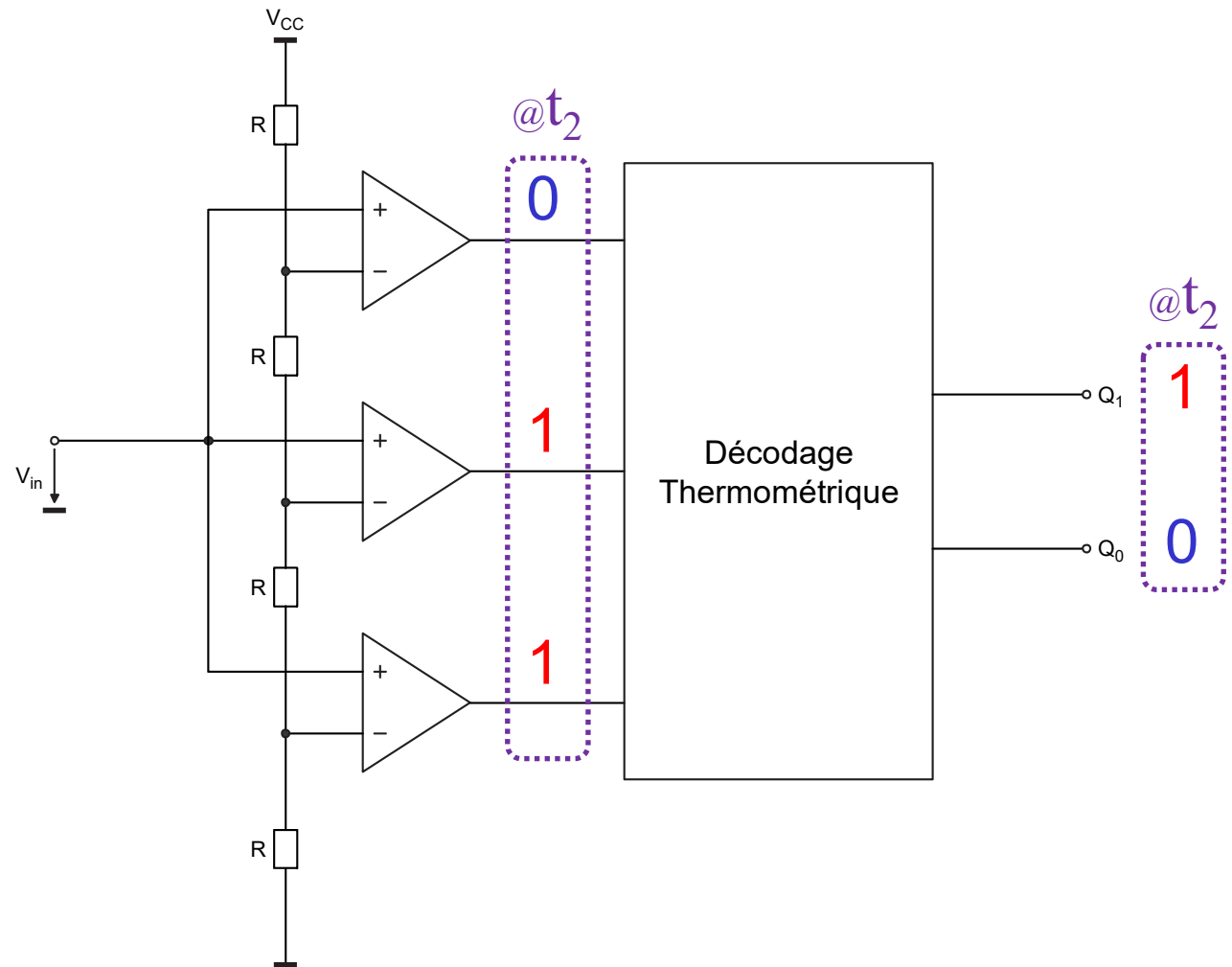
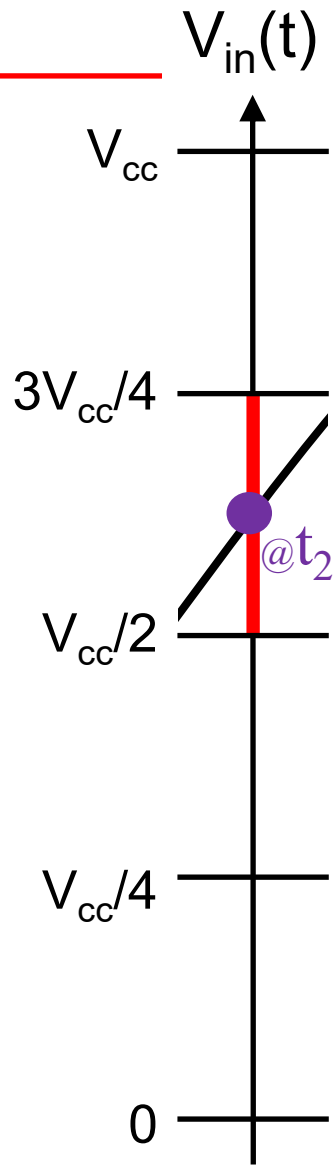
Convertisseur Flash: Implémentation



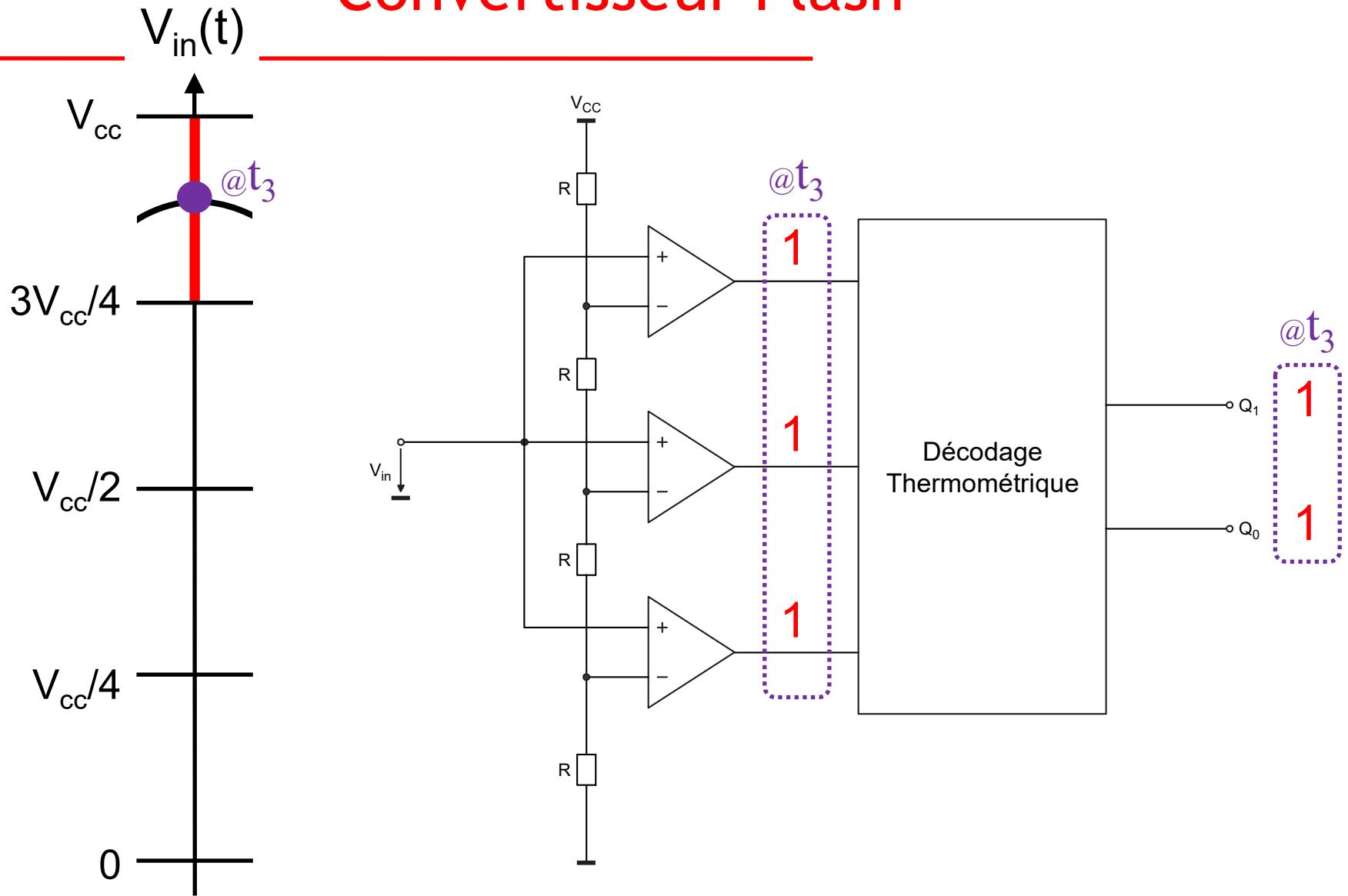
Convertisseur Flash



Convertisseur Flash



Convertisseur Flash



Performances du convertisseur Flash

- Avantages: Conversion rapide
 - (une comparaison → conversion).
- Inconvénients:
 - Complexité: Nécessite $2^N - 1$ comparateurs → surface, consommation élevées (pour 1 bit en plus on doit doubler le nombre de comparateurs.)
 - Résolution limitée en pratique à 8 bits (pareillement, offset, bruit ...)

Ex: Convertisseur à approximations successives (SAR)

- Signal d'entrée comparé successivement avec différents seuils choisis dichotomiquement
- Nécessite 1 seul comparateur
- Moins de surface et de consommation que le convertisseur Flash
- Conversion plus lente
(n comparaisons pour n bits de résolution)

Algorithme: Approximations successives (dichotomique)

Binaire			Décimal
b_2 4	b_1 2	b_0 1	$Dec = \sum_{i=0}^n d_i 2^i$
0	0	0 $b_0 = 0$	0
0	0 $b_1 = 0$	1 $b_0 = 1$	1
0	1 $b_1 = 1$	0 $b_0 = 0$	2
0 $b_2 = 0$	1	1 $b_0 = 1$	3
1 $b_2 = 1$	0	0 $b_0 = 0$	4
1	0 $b_1 = 0$	1 $b_0 = 1$	5
1	1 $b_1 = 1$	0 $b_0 = 0$	6
1	1	1 $b_0 = 1$	7

Approximations successives (Ex)

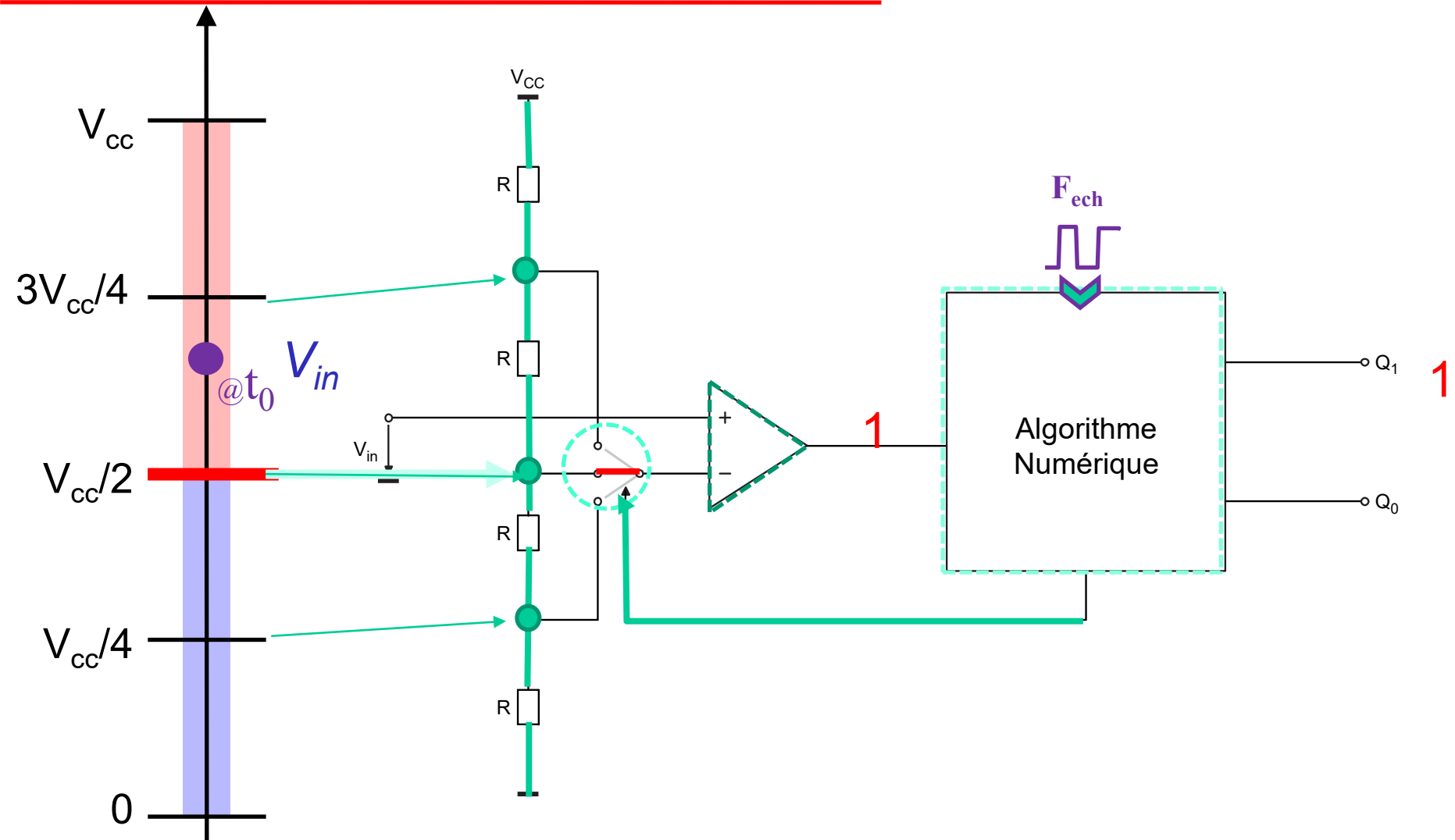
Binaire			Décimal
b_2 4	b_1 2	b_0 1	$Dec = \sum_{i=0}^N b_i 2^i$
0	0	0	0
0	0	1	1
0	1	0	2
0 $b_2 = 0$	1 $b_1 = 1$	1 $b_0 = 1$	3 (011)
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Diagram illustrating the successive approximation process for the decimal value 3.7 (indicated by a green arrow labeled 3.7). The process starts with the binary representation 000 and iteratively sets bits b_2 , b_1 , and b_0 based on the current value and the target 3.7.

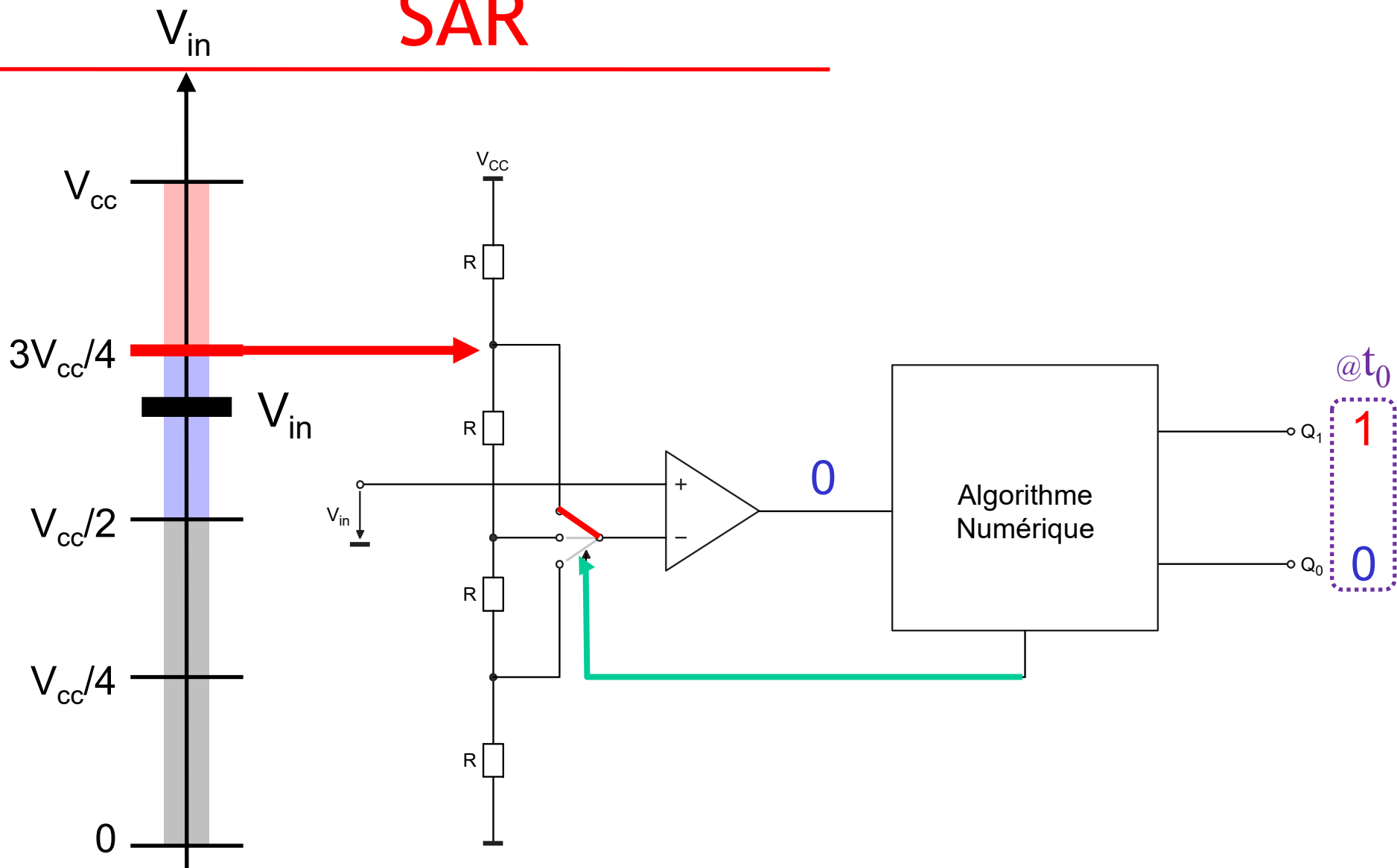
Annotations:

- Green arrow labeled 3.7 points to the first row (000).
- Green arrow labeled ≤ 4 points to the first row (000).
- Green arrow labeled $> 4-2$ points to the second row (001).
- Green arrow labeled $> 4-2+1$ points to the third row (010).
- Red arrow points from the third row (010) to the fourth row (011).

Convertisseur à approximations successives (SAR)



SAR



Performances du SAR

- **Avantages:**
 - **Faible complexité** (1 seul comparateur → surface, consommation faibles)
 - **Haute résolution possible** (up to 16 bits).
- **Inconvénients:**
 - Conversion relativement lente (n comparaisons pour n bits de résolution).

Sommaire

- Conversion Analogique-Numérique (CAN)
 - Echantillonnage
 - Quantification
- Architectures:
 - Convertisseur Flash
 - Convertisseur à approximations successives
- Conversion Numérique-Analogique-(CNA)
 - Génération de courants pondérés
 - Réseau R2R

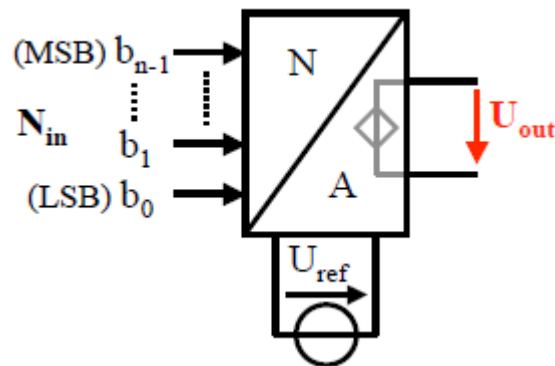
Conversion Numérique/Analogique (CNA)

- Un convertisseur numérique/analogique transforme un nombre, généralement codé en binaire sur n bits, en une grandeur analogique, généralement une tension (parfois un courant), suivant la relation:

$$U_{\text{out}} = (b_{n-1}2^{n-1} + \dots + b_12^1 + b_02^0) \text{ LSB}, \text{ avec } b_i = 0 \text{ ou } 1$$

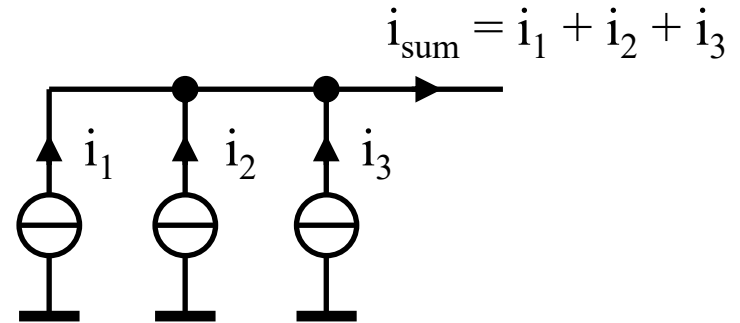
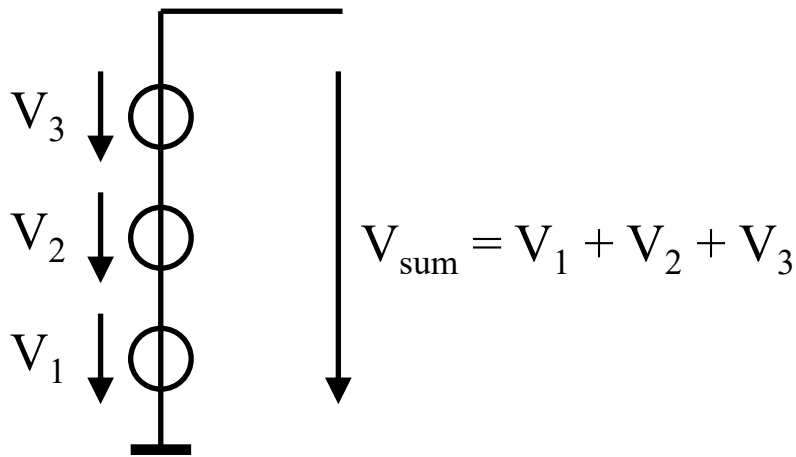
LSB (pour Least Significant Bit) = pas de quantification “ $\equiv$ l’unité de mesure”.

FS (Full scale) = Pleine Echelle “ $\equiv$ gamme de mesure ici spécifiée par U_{ref} ”.



- Rq: C’est un additionneur

Conversion Numérique/Analogique



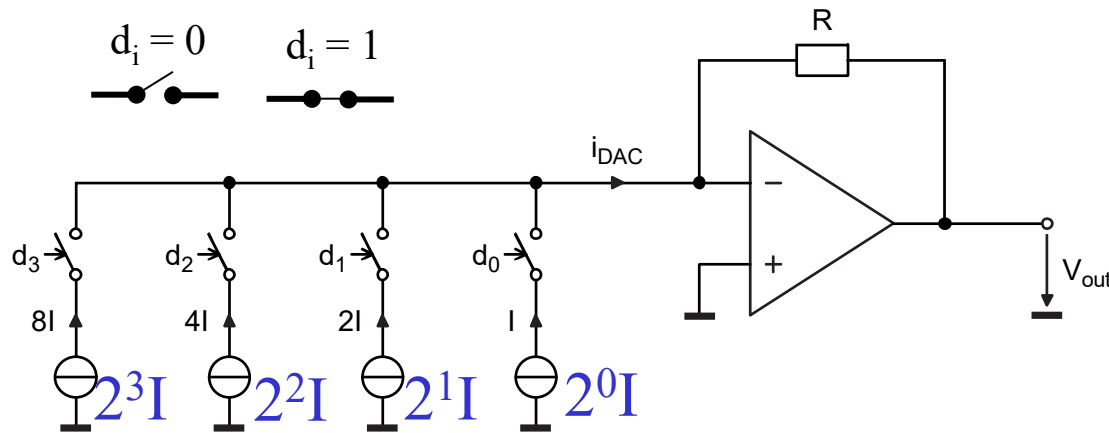
- Additionner des tensions
≡ Mise en série

Trop compliqué
voir impossible si n est trop grand
et/ou V_{cc} est petite

- Additionner des courants
≡ Mise en parallèle

Beaucoup plus simple
Et donc souvent privilégiée

Conversion N/A basée sur l'addition de courants

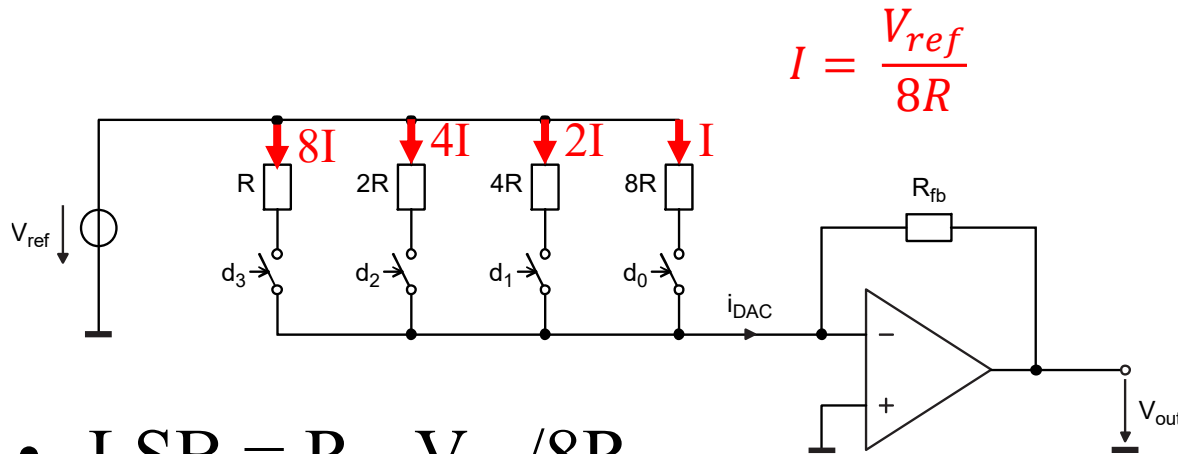


$$V_{out} = -Ri_{DAC} = -RI \sum_{i=0}^3 d_i 2^i$$

- V_{out} est la grandeur analogique (inversée) correspondant nombre numérique ($d_0, d_1, \dots, d_n$).
- L'unité de mesure étant $LSB = RI$.

Comment générer les courants pondérés (c.à.d. les 2^iI avec $i = 0, 1, \dots, n$) ?

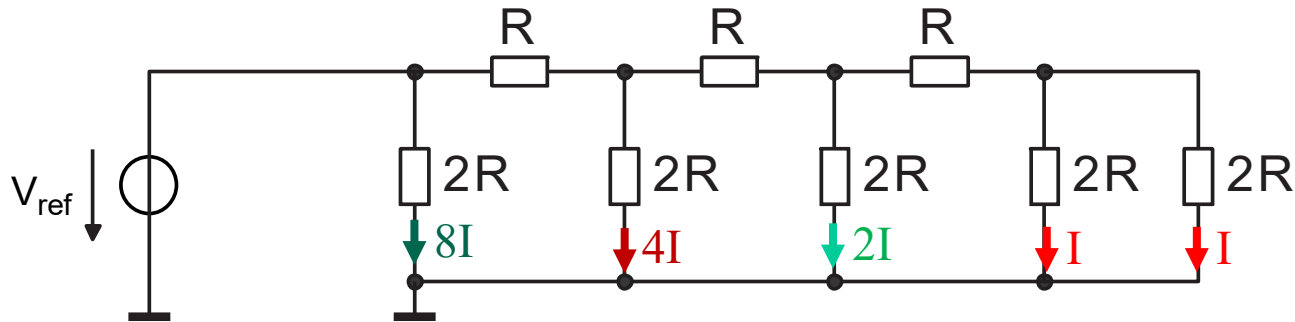
Génération de courants pondérés en puissances de 2



$$\begin{aligned} V_{out} &= -R_{fb} i_{DAC} \\ &= -R_{fb} \frac{V_{ref}}{8R} \sum_{i=0}^3 d_i 2^i \end{aligned}$$

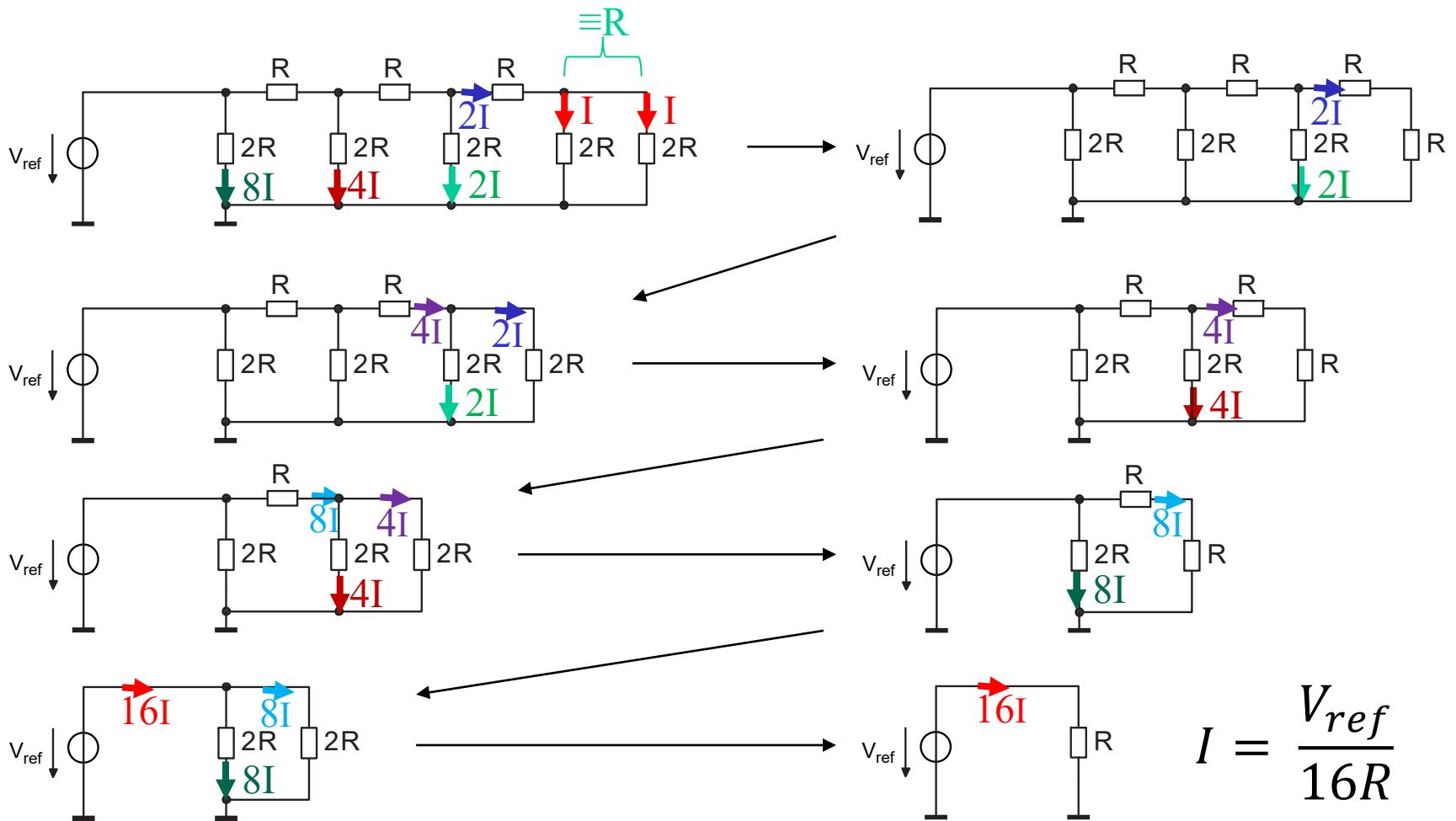
- $LSB = R_{fb} \cdot V_{ref} / 8R$.
- Problème: les valeurs des résistances sont très différentes les unes des autres et augmentent trop rapidement avec la résolution n .
- Une solution basée sur l'ajout de résistances identiques serait préférable.

Réseau R/2R

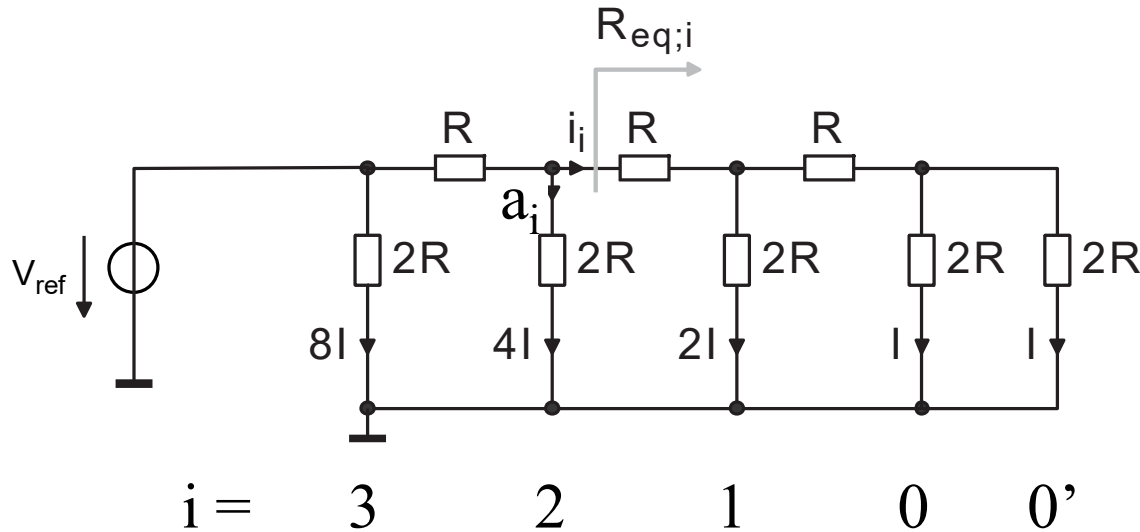


- La tension de référence est divisée par le réseau résistif.
- Démontrant que les courant dans chaque branche sont des courant pondérés (c.à.d. $2^i I$ avec $i = 0, 1, \dots, n$)?

Réseau R/2R: Simplification



Réseau R/2R: Division du courant



$$R_{eq;i} = 2R \quad \forall i$$

$$\Leftrightarrow$$

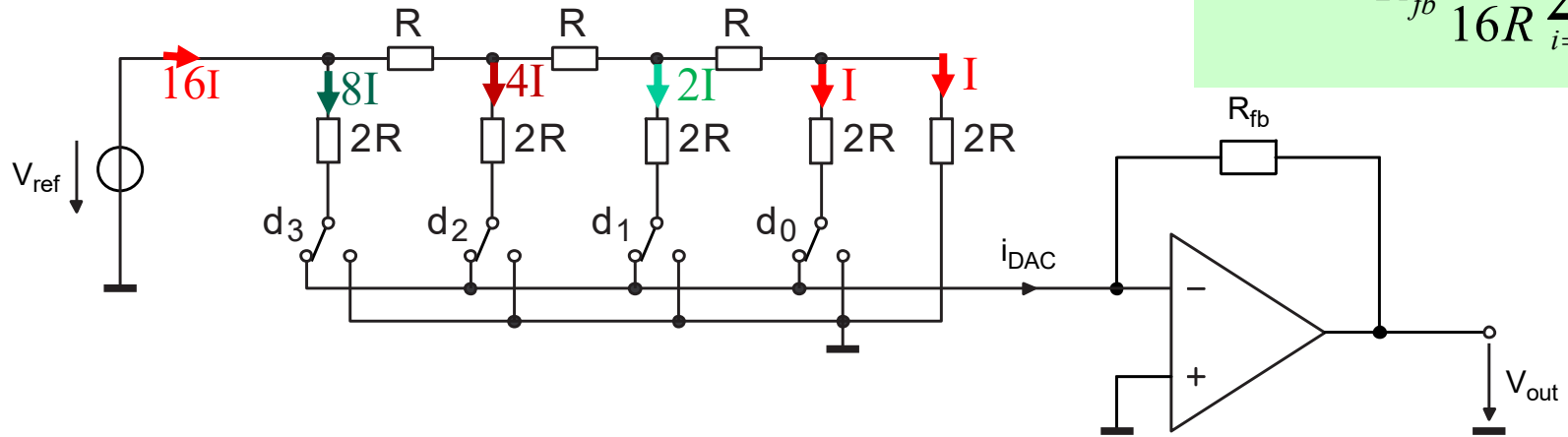
$$a_i = i_i$$

$$a_i = a_0 + \sum_{j=0}^{i-1} a_j = 2^i I$$

- La résistance équivalente vue depuis chaque embranchement vaut $2R$.
- Chaque étage divise le courant en 2.
- Les courants sont pondérés en puissances de 2.

Conversion N/A à réseau R/2R

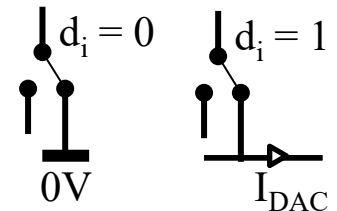
$$I = \frac{V_{ref}}{16R}$$



$$V_{out} = -R_{fb} i_{DAC}$$

$$= -R_{fb} \frac{V_{ref}}{16R} \sum_{i=0}^3 d_i 2^i$$

- Les switches aiguillent le courant de chaque branche dans la masse ou la masse virtuelle qui les somme.



- $LSB = R_{fb} \cdot I = R_{fb} \cdot V_{ref} / 16R$