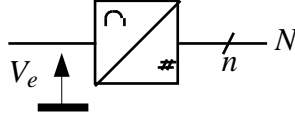


B13. Convertisseurs Analogique / Numérique (CAN)

Analog To Digital Converter (ADC)

- **Pleine échelle (full scale)** : Mode unipolaire : PE (FS) = V_{\max}
 Mode bipolaire : PE (FS) = $V_{\max} - V_{\min}$

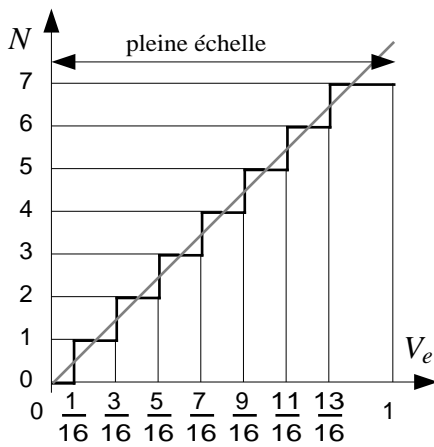
-  2^n intervalles de tension $\in [V_{\min}, V_{\max}] \rightarrow 2^n$ valeurs entières

- **Résolution** ou "pas de quantification" ou "incrément" : $\Delta V_e = q = \frac{PE}{2^n}$

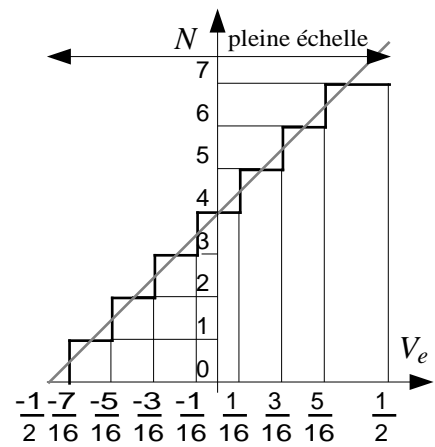
- **Caractéristique** (avec N arrondi à l'entier le plus proche) :

unipolaire :

bipolaire :



$$N = \left\lfloor \frac{V_e}{q} + 0,5 \right\rfloor$$



$$N = \left\lfloor \frac{V_e - V_{\min}}{q} + 0,5 \right\rfloor$$

- **Qualités et défauts des conversions A/N ou N/A**

- Temps d'établissement (*settling time*)

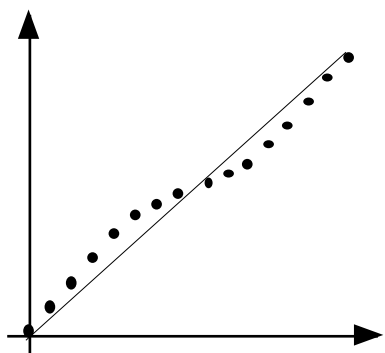
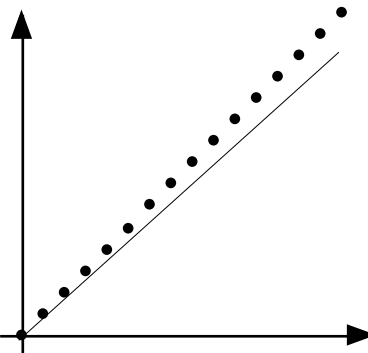
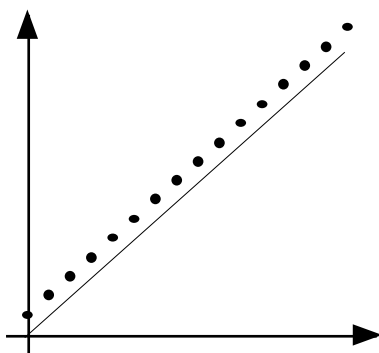
C'est le temps de réponse (ou rapidité) : temps entre l'application d'un échelon pleine échelle à l'entrée du convertisseur et le moment où la sortie entre et reste à l'intérieur d'une bande d'erreur en général équivalente à $\pm 1/2$ bit autour de la valeur finale.

- Précision (*accuracy*) : $p = \frac{\epsilon_{\max}}{V_{\max} - V_{\min}} = \frac{\text{erreur max}}{\text{pleine échelle}}$ en %

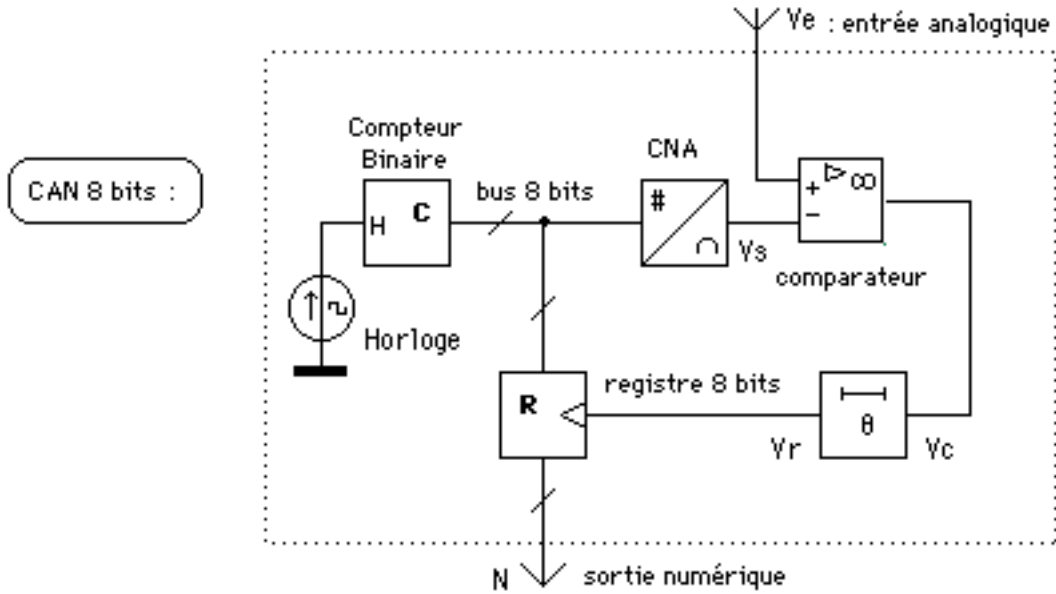
- Erreur de décalage :

- Erreur d'échelle :

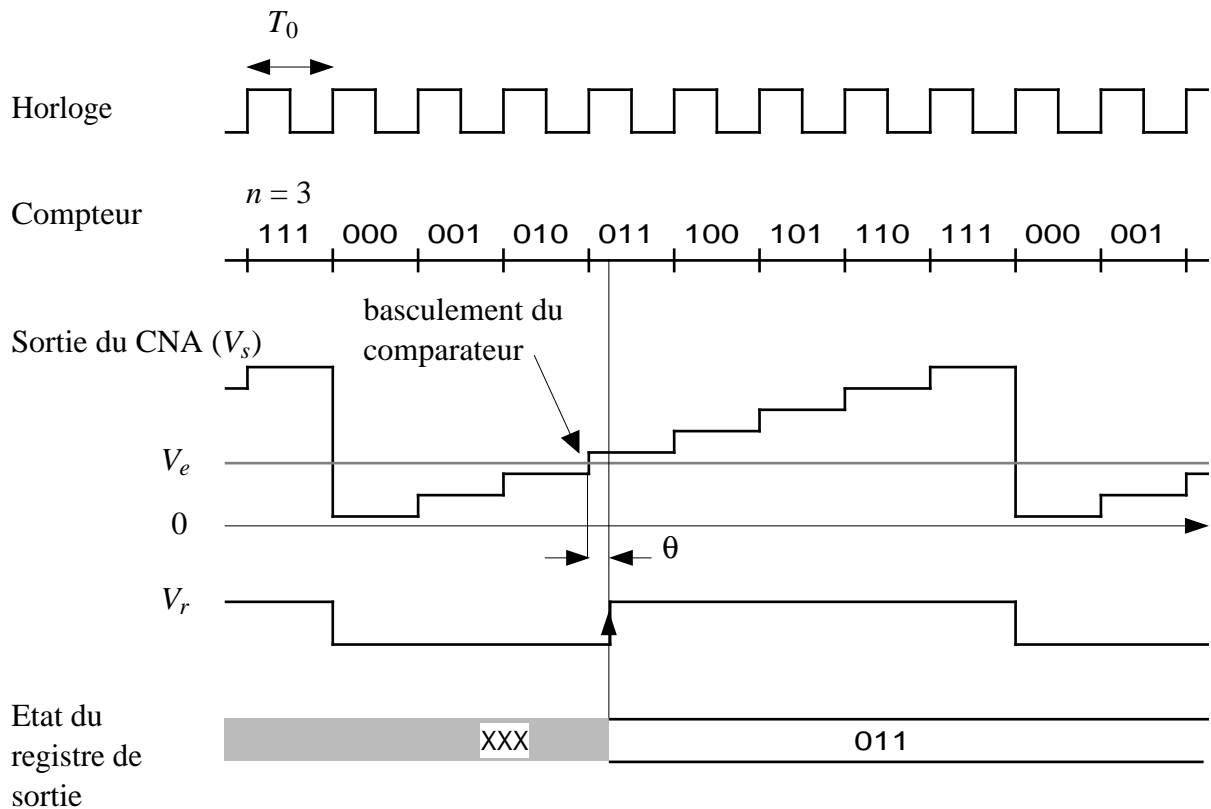
- Erreur de linéarité :



• CAN à rampe numérique

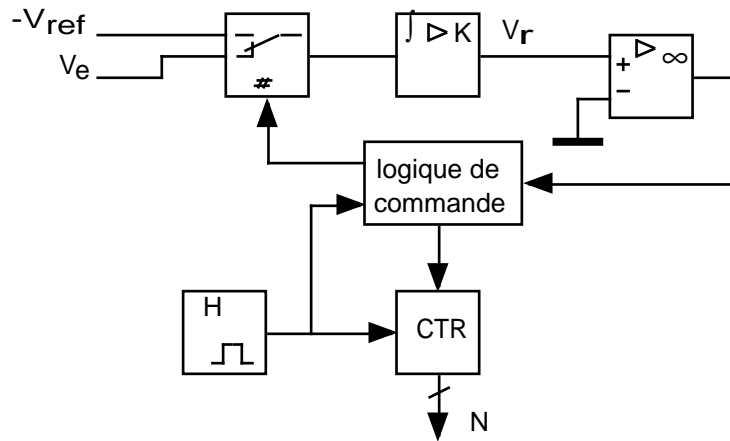


Le compteur binaire **C** est commandé par un signal d'horloge de fréquence f_H . Le bus interne du CAN (bus parallèle, par exemple à 8 bits) relie la sortie du compteur à l'entrée du CNA et à l'entrée du registre parallèle **R**. Lorsque la tension V_s générée par le CNA est égale à la tension analogique d'entrée V_e , le comparateur bascule et envoie un échelon de tension V_r (après un retard θ pour éviter un aléa logique) sur l'entrée d'horloge du registre **R**, qui mémorise ainsi l'état du bus. Le chronogramme est le suivant (exemple donné pour 3 bits) :



Temps de conversion maximal = durée de la rampe numérique.

• CAN à rampe analogique (type "double rampe")

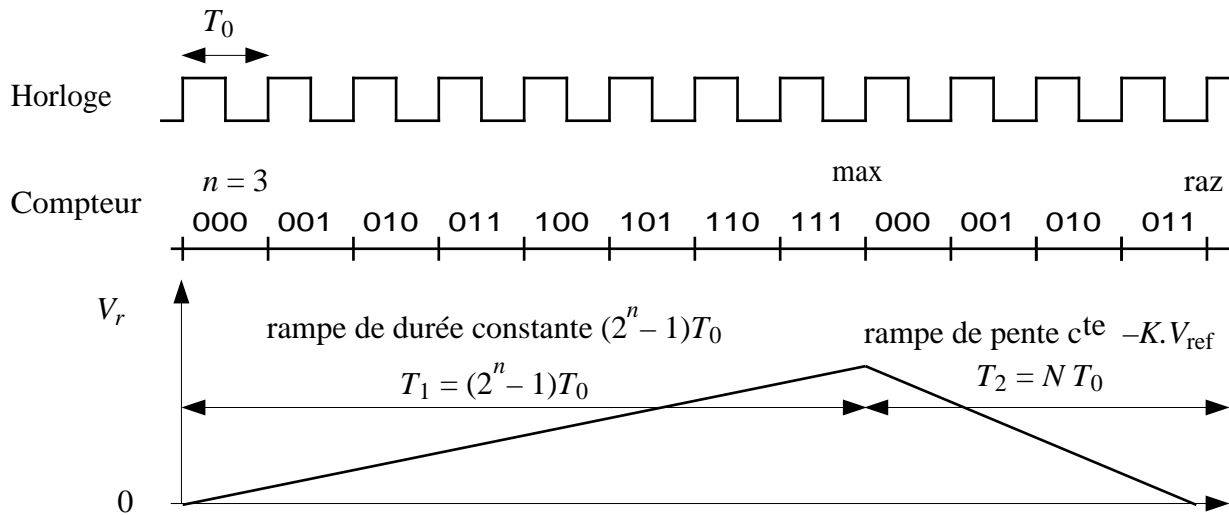


- 1ère phase de fonctionnement : **rampe à durée constante**. La tension analogique d'entrée V_e est appliquée à l'intégrateur, et simultanément le compteur est déclenché. L'intégrateur produit une tension en rampe V_r , qui atteint une valeur maximale telle que : $V_{rmax} = K.V_e T_1$ avec $T_1 = (2^n - 1)T_0$.

- 2ème phase de fonctionnement : **rampe à pente constante**. Lorsque le compteur atteint sa valeur maximale, la logique de commande connecte la référence négative $-V_{ref}$ à l'intégrateur, qui produit une rampe de pente décroissante $-K.V_{ref}$. Lorsque $V_r = 0$, le comparateur bascule et arrête le

compteur au bout du temps $T_2 = N.T_0 = \frac{K.V_e(2^n - 1)T_0}{K.V_{ref}}$.

- L'état final du compteur est donc : $N = \frac{V_e}{V_{ref}}(2^n - 1)$



Temps de conversion maximal (quand $V_e = V_{ref}$) $\approx 2^{n+1}.T_0$

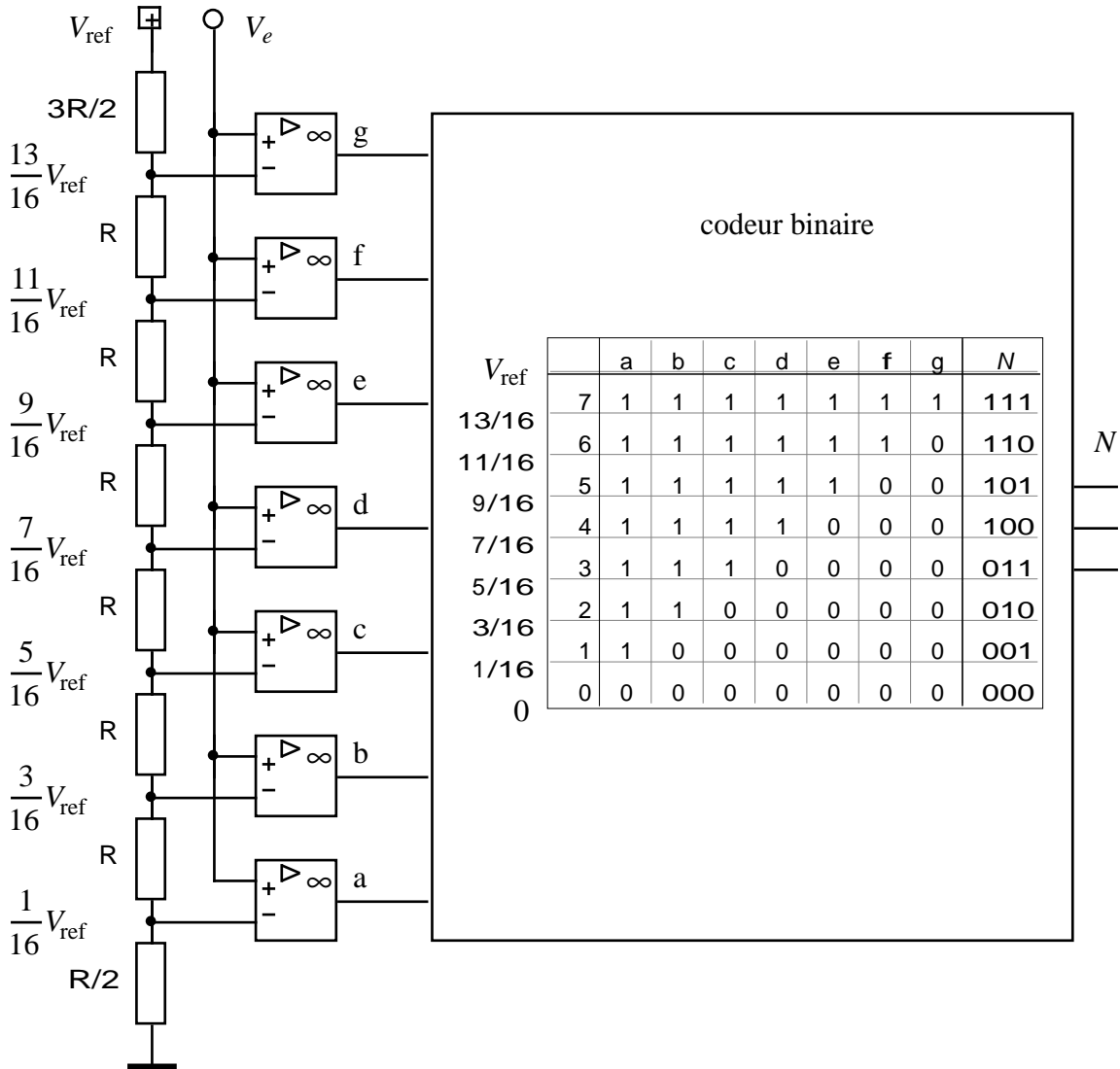
! On peut choisir judicieusement la durée d'intégration $T_1 = (2^n - 1)T_0$ du signal pour éliminer le bruit, par exemple un signal parasite alternatif à 50 Hz provenant du réseau : T_1 doit être un multiple de 20 ms, puisque l'intégrale d'une sinusoïde sur une période est nulle.

***** COMPLEMENTS *****

• **CAN parallèle à comparateurs ("flash")**

Un pont diviseur de tension constitué de 2^n résistances répartit une tension de référence V_{ref} en 2^n intervalles selon la caractéristique $N(V_e)$ indiquée plus haut. La tension à mesurer V_e est appliquée simultanément à $2^n - 1$ comparateurs. L'ensemble des comparateurs dont l'entrée de référence est portée à un potentiel inférieur à V_e sont à l'état haut, contrairement à ceux dont l'entrée de référence est portée à un potentiel supérieur à V_e . Une logique de codage décode les sorties des comparateurs en un nombre binaire N sur n bits.

Le temps de conversion est extrêmement faible, limité au temps d'établissement des comparateurs. Mais ce schéma est limité par sa complexité : avec $n = 8$ bits, il faut 256 comparateurs. Au-delà d'une certaine valeur, il devient très difficile à réaliser : pour chaque bit supplémentaire, il faut doubler les circuits tout en doublant également la précision.



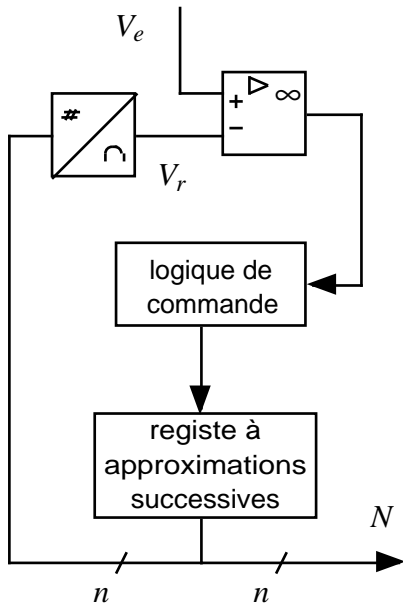
• **CAN à approximations successives (ou CAN par dichotomie)**

Le convertisseur compare la tension d'entrée V_e avec la sortie d'un CNA par pesées successives, comme le ferait une balance. La technique consiste à procéder par dichotomie, en divisant successivement par deux l'intervalle de tension dans lequel est mesurée V_e . Au départ, le bit de poids fort (MSB) est positionné à 1, les autres bits sont à 0. Par exemple, pour $n = 8$, le mot test $N = 1000\ 0000$ est envoyé au CNA, qui fournit la tension V_r :

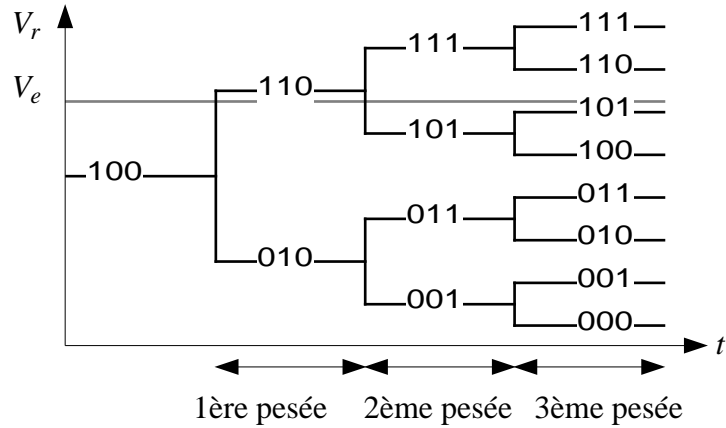
- si $V_r > V_e$, le MSB est mis à 0, et le bit suivant est positionné à 1 : on envoie $N = 0100\ 0000$ dans le CNA.
- si $V_r < V_e$, le MSB ne change pas, et l'on envoie $N = 1100\ 0000$ dans le CNA.

Etc. Cet algorithme est réalisé dans un "registre à approximations successives" (*Successive Approximation Register* ou *SAR*). Mais on peut aussi facilement l'implémenter dans un microcontrôleur, qui possède en général une entrée analogique reliée à un comparateur, et un CNA interne.

Schéma de principe :



Dichotomie pour $n = 3$ bits :



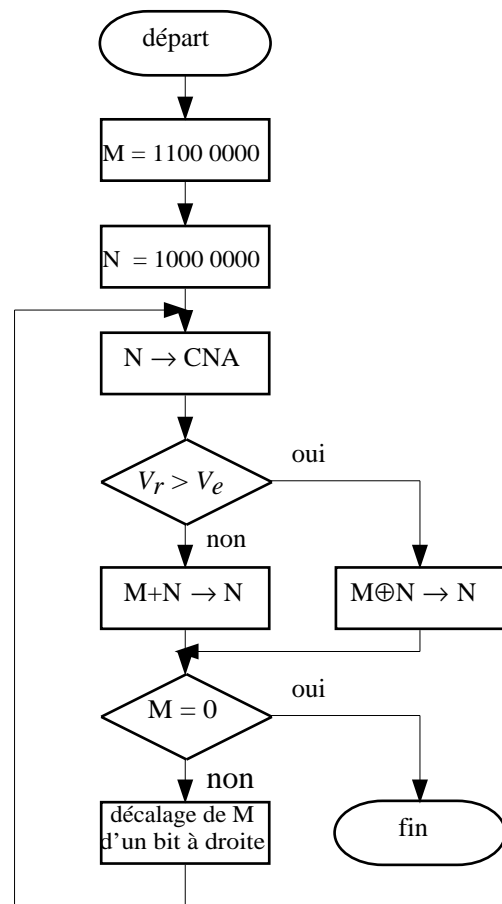
Le microcontrôleur commence par inscrire dans un registre le premier mot test $N = 1000\ 0000$ (et envoyé au CNA) et dans un second registre un mot de contrôle $M = 1100\ 0000$:

- si $V_r > V_e$ un OU Exclusif est effectué entre N et M , ce qui entraîne la mise à 0 du bit de poids fort de M et la mise à 1 du bit de poids 6 de N .
- si $V_r < V_e$ un OU simple est effectué entre N et M , ce qui conserve le bit de poids fort de N .

M est ensuite décalé vers la droite de 1 bit. Puis le cycle recommence, jusqu'à ce que tout 1 ait été chassé de M .

Exemple pour $n = 3$ (cf figure ci-dessus) :

- $N = 100 ; M = 110$
 - 1) $V_r < V_e \Rightarrow N = 110 + 100 = 110$
 $M = 011$
 - 2) $V_r > V_e \Rightarrow N = 011 \oplus 110 = 101$
 $M = 001$
 - 3) $V_r < V_e \Rightarrow N = 001 + 101 = 101$
 $M = 000$
 - 4) $V_r < V_e \Rightarrow N = 000 + 101 = 101$
- Fin



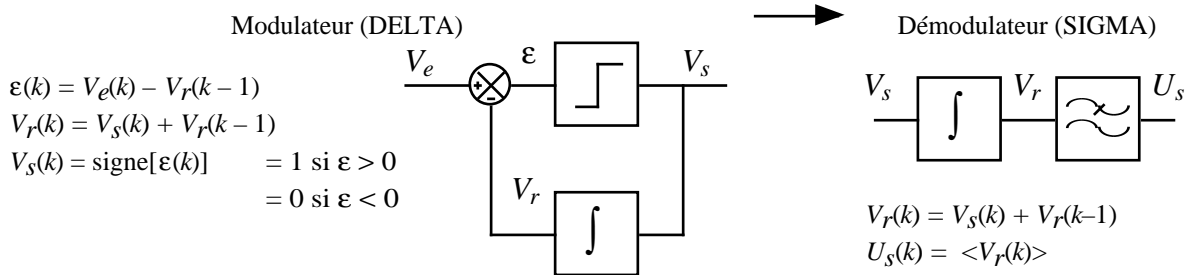
Temps de conversion : il faut n passes pour exécuter l'algorithme de dichotomie. L'intérêt de ce montage est que le temps de conversion est constant, et relativement faible.

• CAN Sigma-Delta ($\Sigma-\Delta$)

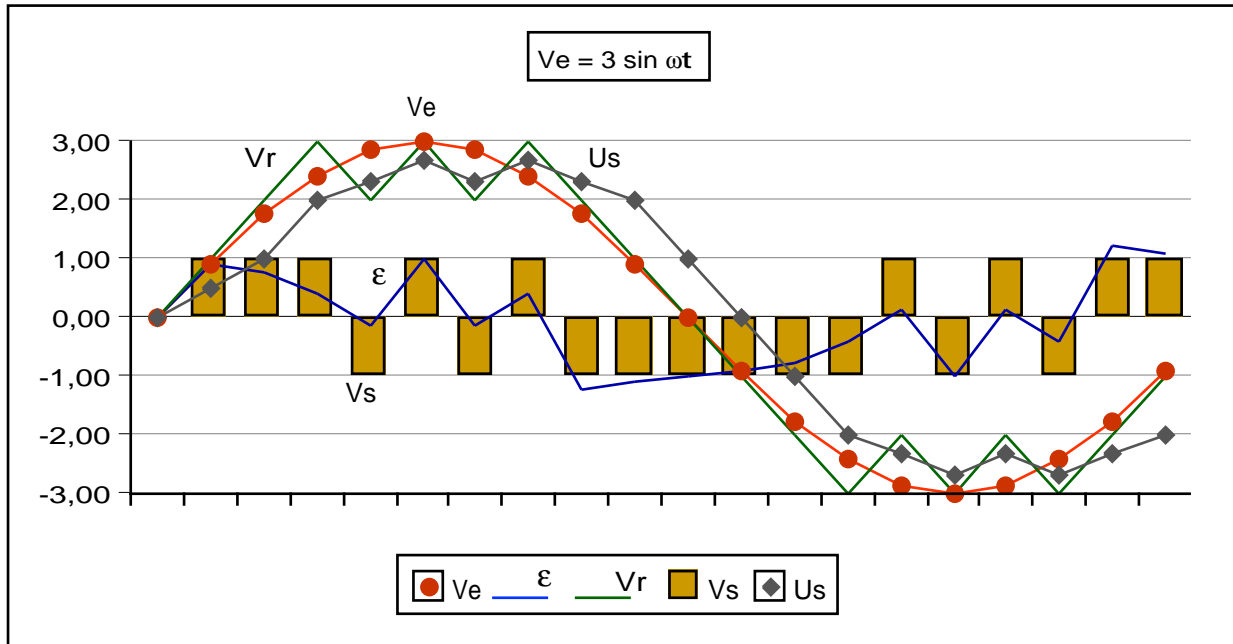
a) Modulation / Démodulation Delta-Sigma

Cette modulation consiste à coder les **variations** du signal V_e par une suite d'impulsions binaires ($V_s = 0$ ou 1). Pour que le codage s'effectue correctement, il faut que la fréquence d'échantillonnage soit assez élevée pour tenir compte des variations les plus rapides du signal.

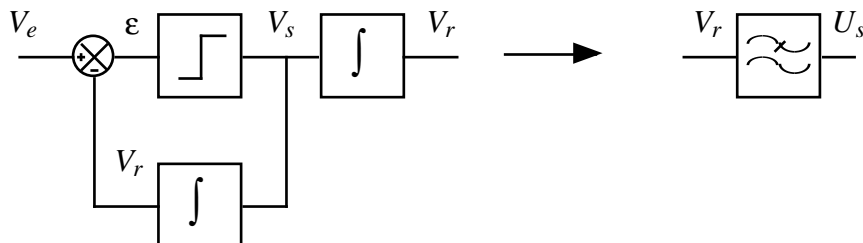
Le modulateur est constitué d'un comparateur linéaire (montage soustracteur), qui fournit un signal $\epsilon = \Delta V_e$ représentant la différence ("DELTA") entre la tension d'entrée à l'instant k et cette même tension à l'instant précédent $k-1$. Le signe de ϵ est codé par un comparateur binaire qui produit la tension de sortie V_s . Cette tension est à son tour intégrée (SIGMA) pour reconstituer une image de l'échantillon précédent $V_r = V_e(k-1)$ afin de calculer $\Delta V_e = V_e(k) - V_e(k-1)$.



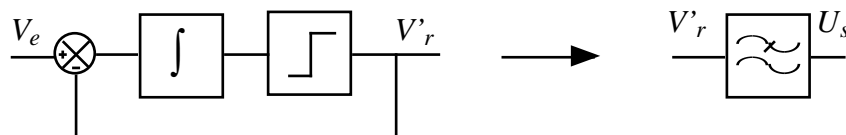
Simulation réalisée sur 20 échantillons :



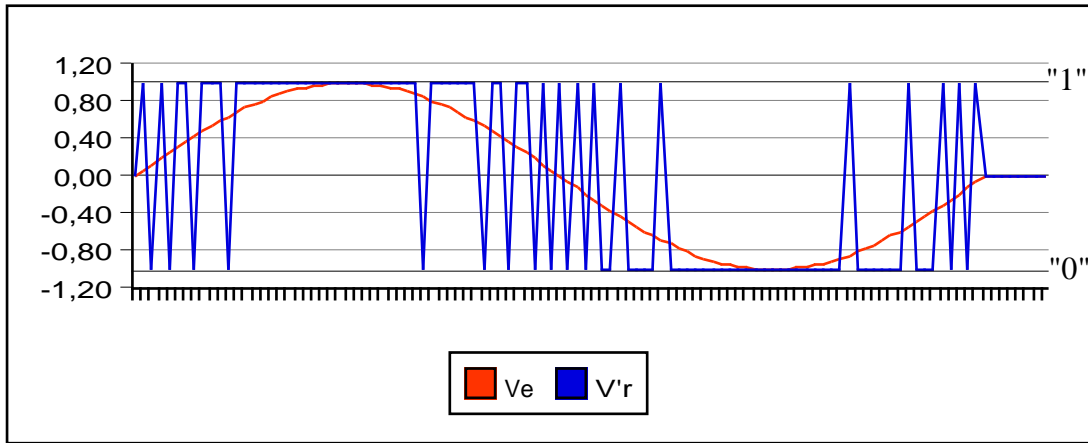
Ou encore, en réunissant les intégrateurs dans le modulateur :



soit finalement :



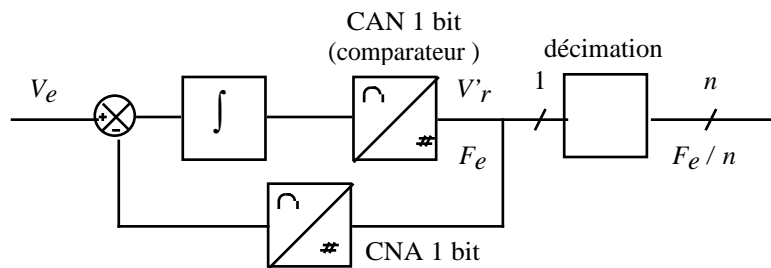
Simulation réalisée sur 100 échantillons :



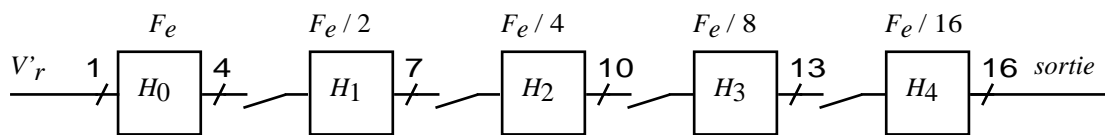
La tension V_r se présente sous la forme d'une suite de bits avec un débit de F_e bits/s, au lieu d'une suite d'échantillons analogiques. Sous sa forme la plus simple, elle équivaut à une tension modulée en largeur d'impulsions, avec un niveau de sortie binaire.

b) Convertisseur Delta-Sigma

Dans ce convertisseur, on remplace la suite de bits cadencés à la fréquence F_e qui constitue la tension V_r par une suite de mots de n bits cadencés à la fréquence F_e / n formant le signal de sortie U'_s . Tout en conservant le même débit d'information, on obtient ainsi un CAN n bits par un calcul purement numérique. L'algorithme utilisé est appelé "décimation".



Exemple (simplifié !) d'algorithme de décimation implémenté sur un CAN 16 bits avec un suréchantillonnage par 16 :



Le premier filtre H_0 est un filtre moyennneur par 16 (c'est donc un passe-bas de fréquence de coupure $F_e/16$) à fenêtre glissante (voir § *filtrage numérique*). Il en sort un signal cadencé à la fréquence F_e dont la résolution est en $1/2^4$ (puisque la moyenne de 16 échantillons de 1 bit peut prendre 16 valeurs distinctes), soit 4 bits.

Les autres filtres sont des filtres moyennneurs par 8 à fenêtre glissante. Chaque filtre permet donc d'augmenter la précision du signal obtenu par $1/2^3$, soit 3 bits supplémentaires par filtre. Au total, le résultat final est donc écrit avec $4 + 4 \times 3 = 16$ bits.

A la sortie des filtres H_1, \dots, H_4 on ne conserve à chaque fois qu'un échantillon sur deux : c'est le principe de la décimation. Cela revient à diviser la fréquence d'échantillonnage par 2 après chaque filtre.

Au total, le débit d'information global du système est de F_e bits/s à l'entrée, et de $16 \times F_e / 16$ à la sortie. Ce débit est donc conservé.

L'intérêt d'un CAN Delta-Sigma est d'être constitué de circuits purement numériques. Il est donc facile et peu onéreux de construire un convertisseur de grande précision. Mais l'exigence du suréchantillonnage impose une limite à la fréquence maximale de conversion. En outre, en continu, la présence d'un intégrateur entraîne un risque de dérive de la réponse du CAN au cours du temps. Au total, le domaine d'emploi typique de ces convertisseurs est le signal audio compris entre 20 Hz et 20 kHz, ainsi que le contrôle de vitesse des moteurs DC et AC.

• Discussion concernant le choix du pas de quantification d'un système numérique

1°) Cas des CNA

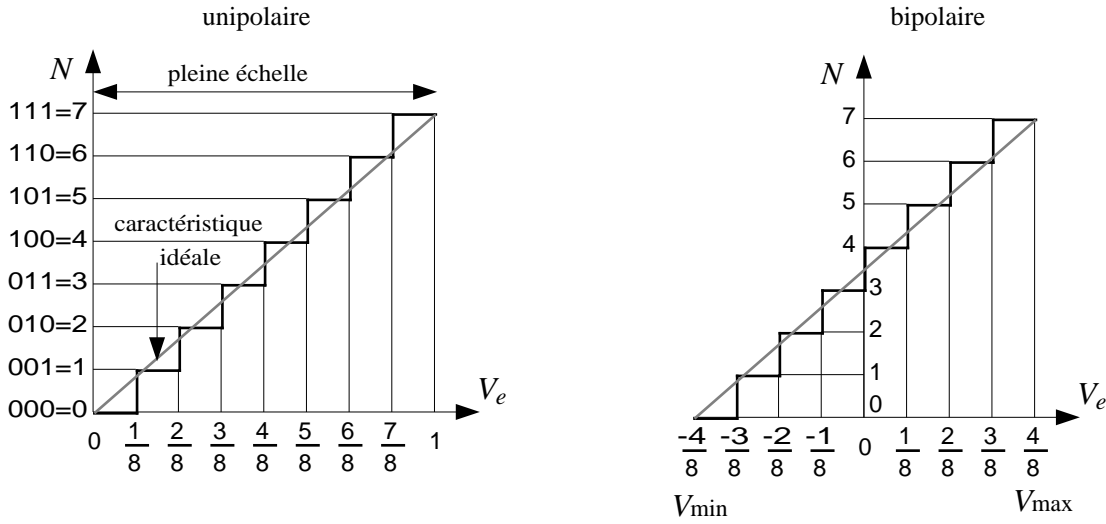
On a vu au §B12 que la fonction de transfert d'un CNA s'écrit :

$$V_s = q.N \text{ (CNA unipolaire) ou } V_s = q.N + V_{\min} \text{ (CNA bipolaire), avec } q = \frac{PE}{2^n - 1}$$

Pour n donné (nb de bits), il n'y a pas d'autre façon possible de choisir q , qui est fixé par le nombre d'intervalles $2^n - 1$ séparant 2^n valeurs entières distinctes.

2°) Cas des CAN : calcul avec troncature

Notons $\lfloor x \rfloor$ la fonction "partie entière de x " (définie pour x réel > 1). Une première solution consiste à faire le calcul par valeur inférieure : $N = \lfloor x \rfloor$. C'est une troncature.



$$q = PE / 8$$

$$V_e \in [0, q] \Rightarrow N = 0$$

$$V_e \in [q, 2q] \Rightarrow N = 1$$

$$\dots$$

$$V_e \in [7q, PE] \Rightarrow N = 7$$

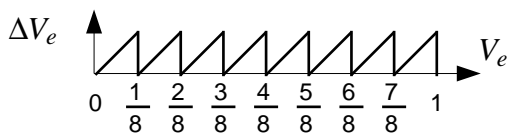
$$\Rightarrow N = \left\lfloor \frac{V_e}{q} \right\rfloor$$

$$N = \left\lfloor \frac{V_e - V_{\min}}{q} \right\rfloor$$

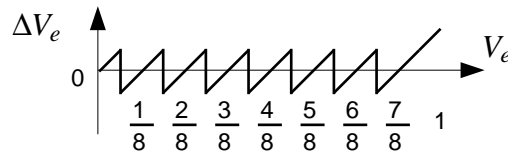
3°) Bruit de quantification d'un CAN

L'erreur de quantification des CAN ou "bruit de quantification" est l'erreur absolue systématique, différence entre la caractéristique idéale (valeur de la tension d'entrée) et la caractéristique réelle. En mode unipolaire, elle vaut : $\Delta V_e = V_e$

- $N.q$. Les graphes ci-dessous indiquent cette erreur pour 2^n pas :
 troncature : erreur < 1 LSB

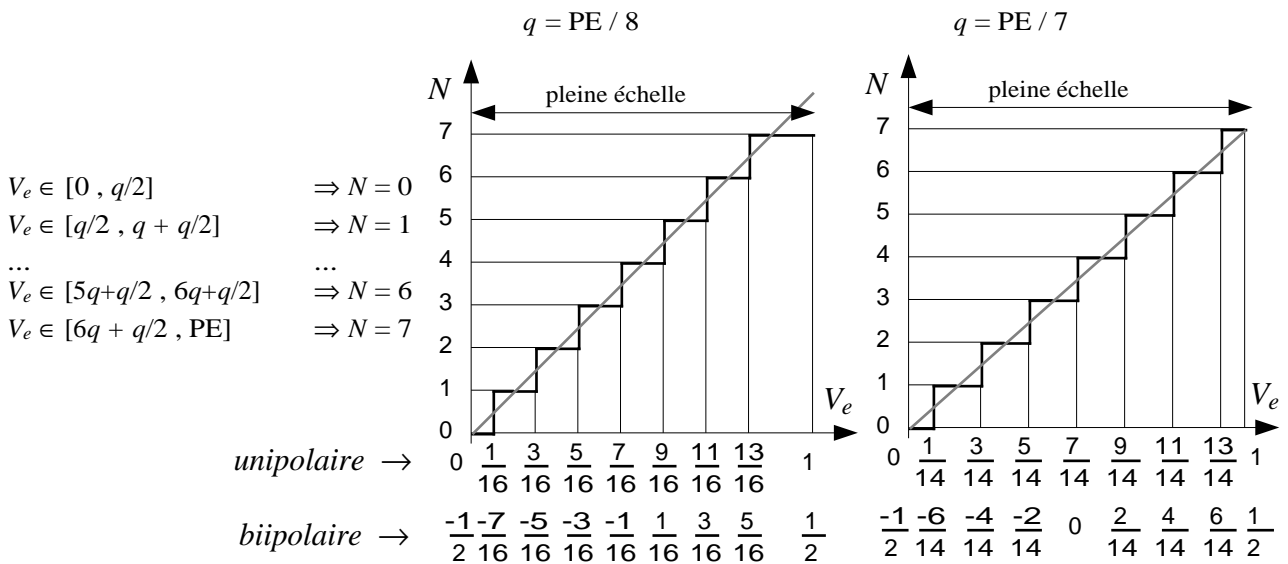


arrondi : erreur $< 1/2$ LSB



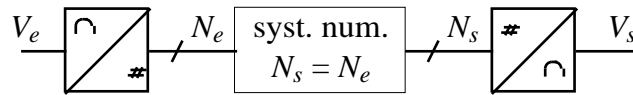
4°) Choix du pas de quantification

Clairement, le calcul le plus intéressant est l'arrondi à l'entier le plus proche : $N = \lfloor x + 0,5 \rfloor$, parce qu'il génère un bruit de quantification plus faible. C'est celui qui est généralement adopté (et présenté au début de ce chapitre). Mais cela implique que la caractéristique doit être décalée vers la gauche de "1/2 bit" pour s'approcher de la caractéristique idéale (cf fig. ci-dessous, à gauche). On remarque alors que, dans ce cas, il y a une dissymétrie entre les bornes minimale et maximale de la pleine échelle : à cause du décalage, le premier intervalle est de largeur $q/2$, alors que le dernier intervalle est de largeur $3q/2$. Pour éviter cette dissymétrie, on peut adopter un incrément $q = PE / (2^n - 1)$ au lieu de $PE / 2^n$ (fig. ci-dessous, à droite), mais cette solution présente l'inconvénient d'augmenter le bruit de quantification (voir 5°) :

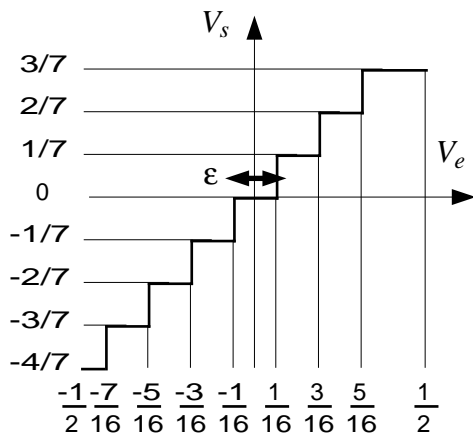


5°) Caractéristique de transfert d'un système numérique

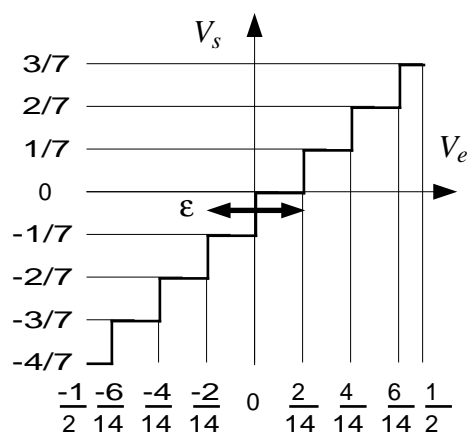
On considère un système numérique qui ne fait que recopier sur sa sortie le signal présent à l'entrée. On se place dans le cas de conversions bipolaires. On suppose que $n = 3$:



1ère solution :
CAN bipolaire $q = PE/8$
et CNA $q = PE/7$



2ème solution :
CAN bipolaire $q = PE/7$
et CNA $q = PE/7$



Soit $V_e(t) = \epsilon(t)$, tension résiduelle de bruit. Dans le premier cas (appelé "loi Américaine à mi-marche", utilisé aux Etats-Unis), tant que $|\epsilon| < 1/16$ de PE, la sortie V_s reste à zéro. Dans le second cas (appelé "loi Européenne" en téléphonie numérique), pour tout $|\epsilon|$ compris entre 0 et $1/7$ de PE, la sortie V_s oscille entre $-1/7$ et zéro. Cette dernière solution présente donc un facteur de bruit important : même en l'absence de tension d'entrée, la tension de sortie n'est pas nulle. Il est préférable de choisir des incréments distincts, égaux respectivement à $PE/2^n$ en entrée (au niveau du CAN) et $PE/(2^n - 1)$ en sortie (au niveau du CNA).