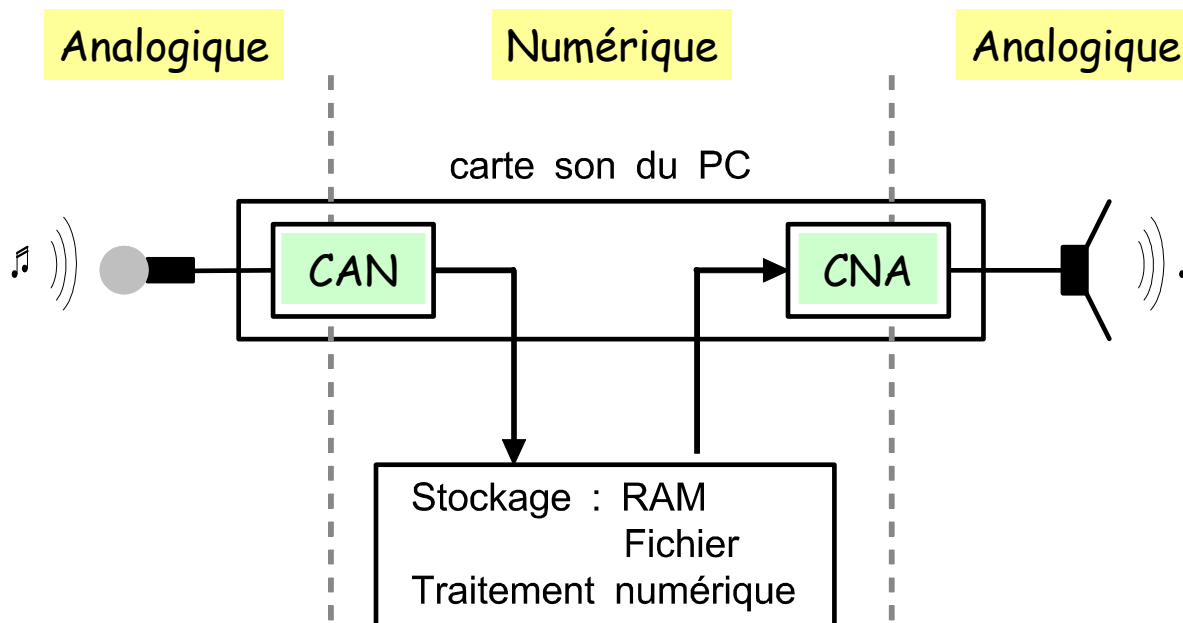


# 1/ CAN - CNA Exemple d'un enregistrement sonore



Convertisseur **A**nalogique **N**umérique : **A**nalog to **D**igital **C**onverter

🇫🇷 CAN : ADC 🇬🇧

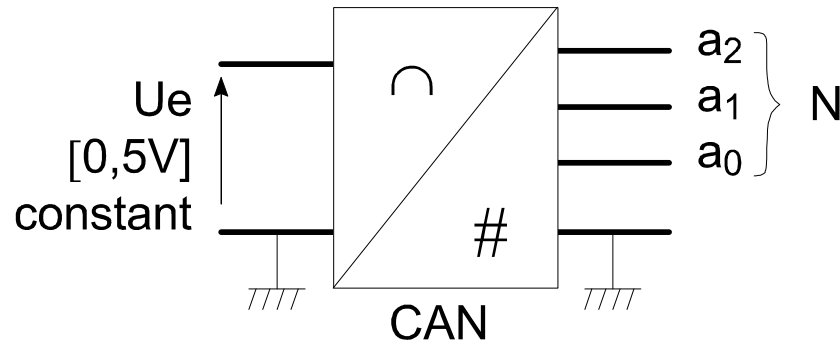
Convertisseur **N**umérique **A**nalogique : **D**igital to **A**nalog **C**onverter


🇫🇷 CNA : DAC 🇬🇧

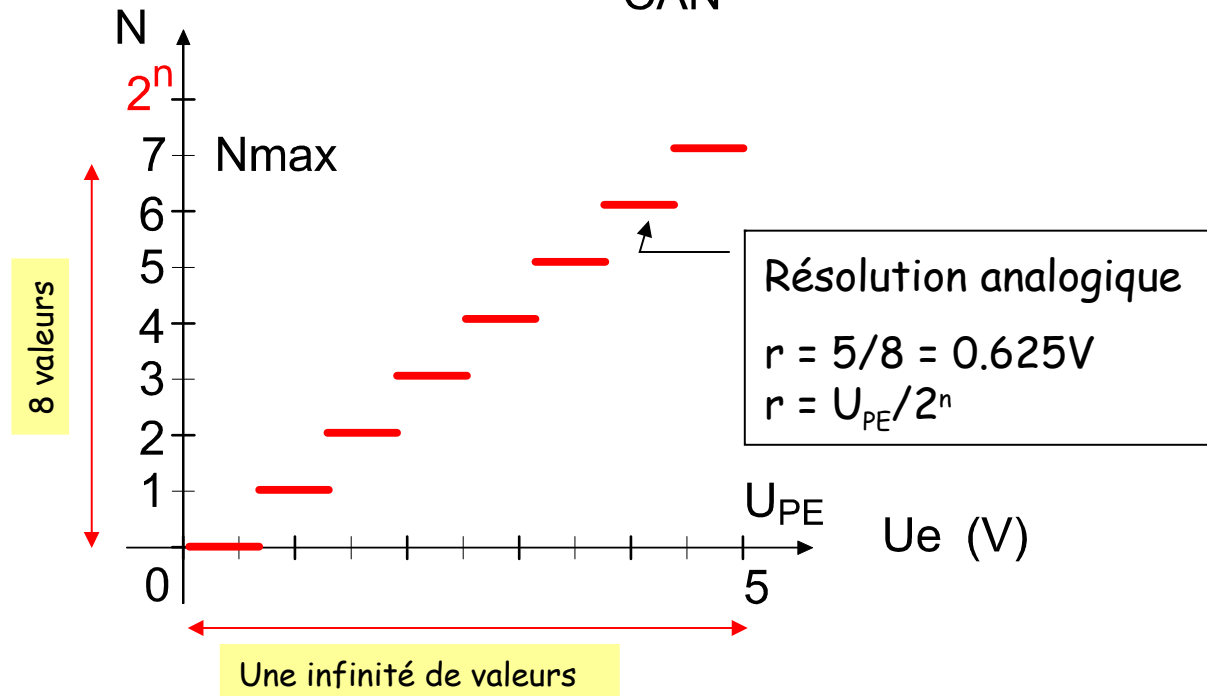
# 1.a/ Convertisseur Analogique Numérique



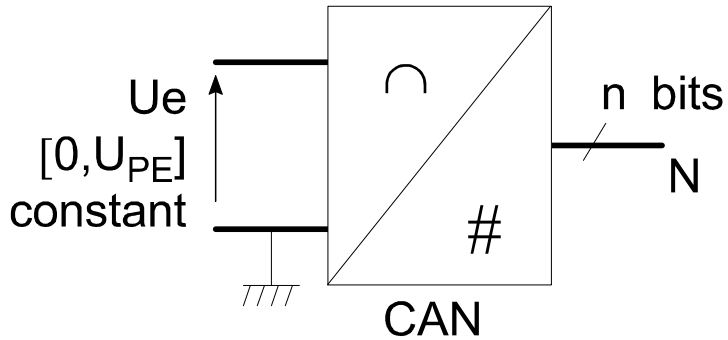
- Exemple d'un CAN 3 bits ( $n = 3$ )



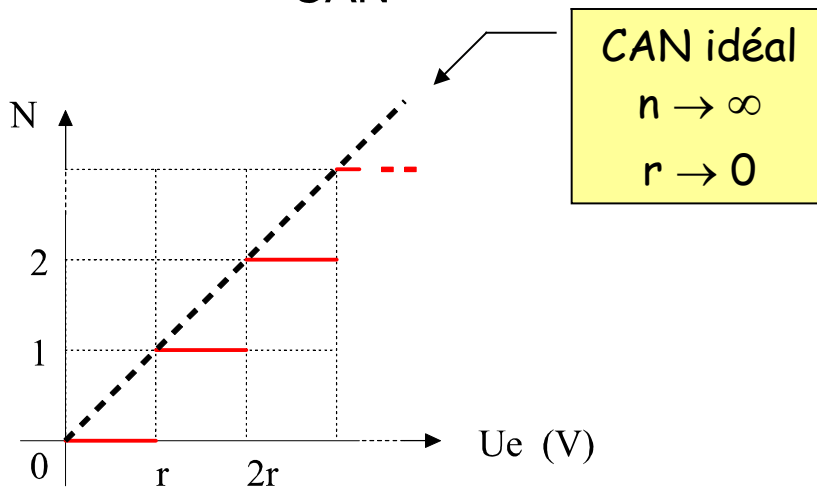
  
La conversion Analogique-Numérique introduit toujours une erreur de quantification



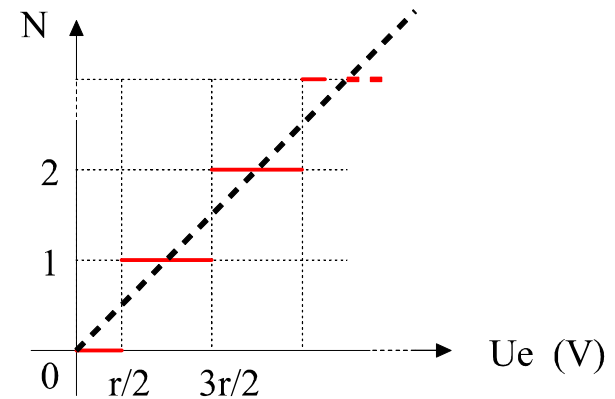
# 1.b/ Erreur de quantification



Résolution analogique :  $r = U_{PE}/2^n$   
 Résolution numérique : n bits



Erreur analogique : r  
 Erreur numérique : 1 LSB



Erreur analogique :  $\pm r/2$   
 Erreur numérique :  $\pm 1/2$  LSB

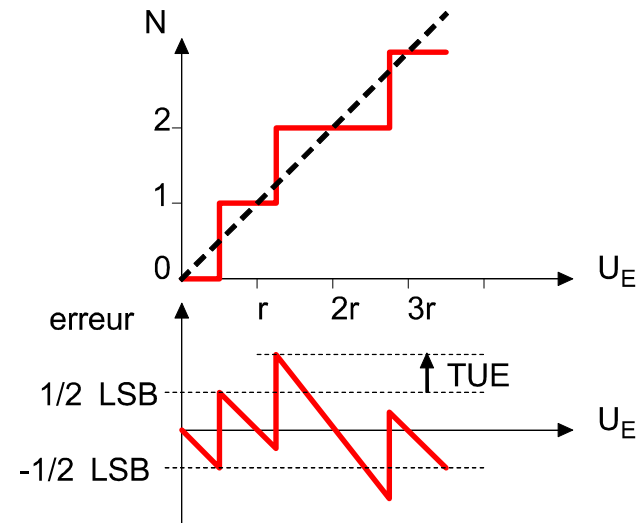
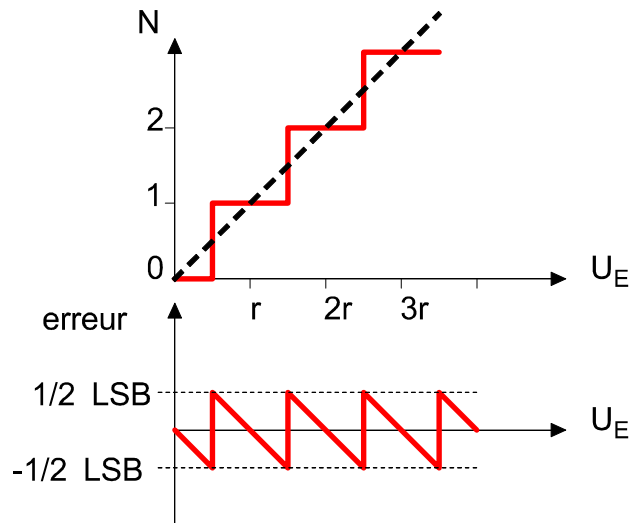
## 1.c/ Autres erreurs



À l'erreur de quantification, s'ajoutent d'autres erreurs  
linéarité, offset, gain ...

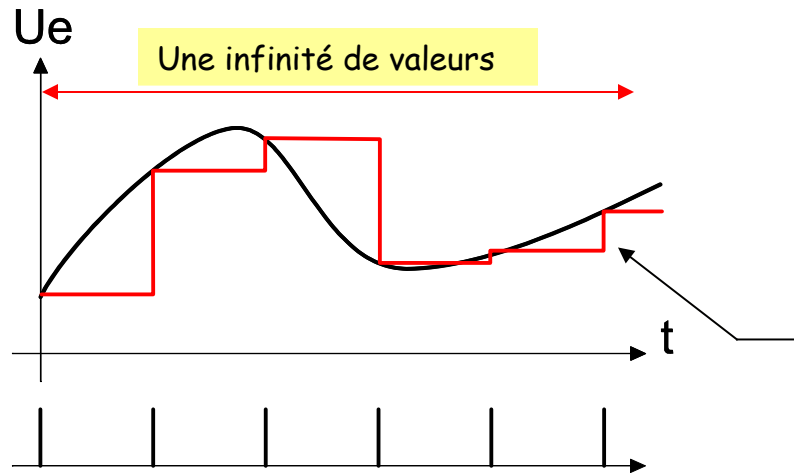
le constructeur fournit en général la valeur max

TUE : Total Unadjusted Error



Documents ( pdf ) : [CAN TLC549](#) [ADC08831](#)

## 1.d/ Convertir une tension variable



En amont du CAN se trouve un échantillonneur-bloqueur qui prélève régulièrement une valeur de  $U_e$  et bloque cette valeur jusqu'à l'échantillon suivant. ( mémoire analogique )

Tension d'entrée du CAN  
le temps de conversion doit être inférieur à  $T_e$

Commande de l'échantillonneur  
Période  $T_e$  , Fréquence  $F_e = 1/T_e$

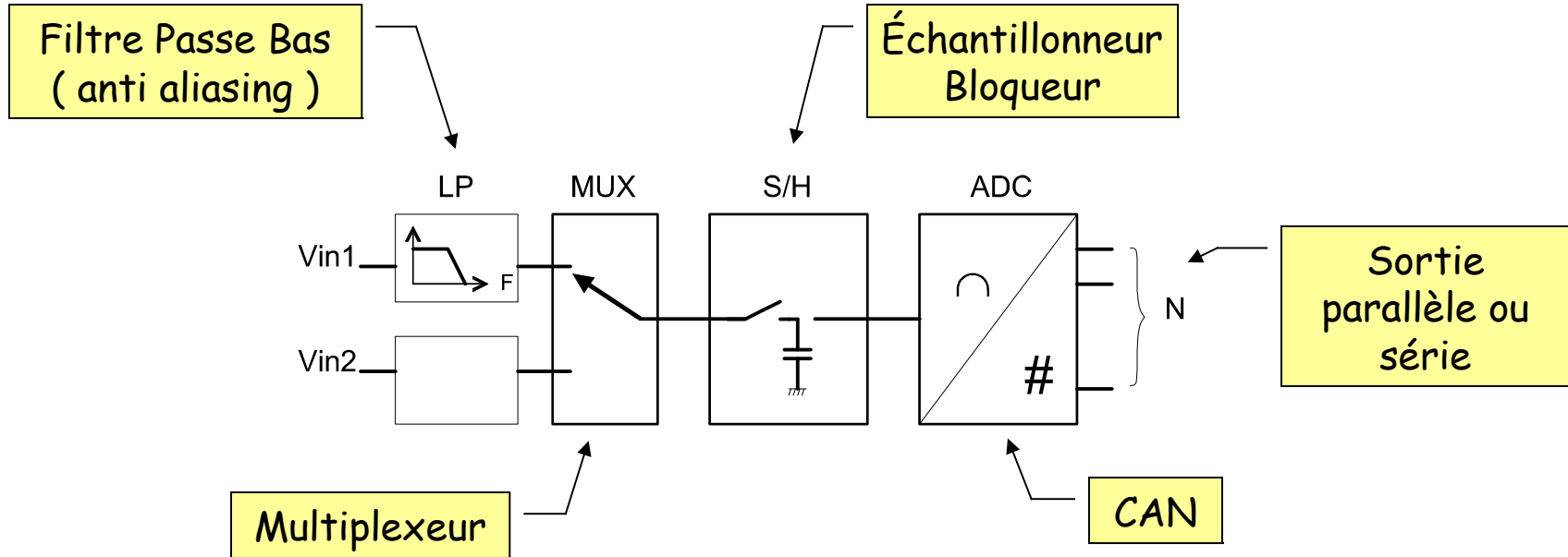
La conversion analogique numérique implique une double quantification :  
quantification temporelle ( échantillonnage )  
quantification en amplitude ( résolution )

Simulation crocodile :  
[Échantillonneur-bloqueur](#)

Document ( pdf ) :  
[CAN TLC549](#)



# 1.f/ Pour résumer...





## 2.a/ Technologie des CAN

Technologie	Temps de conversion	Exemple d'utilisation
Simple rampe	Lent ( ms )	Mesure sans précision
Double rampe Multi rampe	Lent ( ms )	Multimètre
Approximations successives	Rapide ( $\mu$ s )	Acquisition son
Flash ( ou CAN parallèle )	Très rapide ( ns )	Acquisition vidéo Oscilloscope numérique

Autres technologies : ( voir documentation )

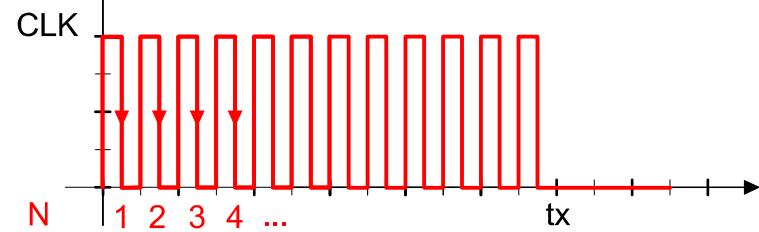
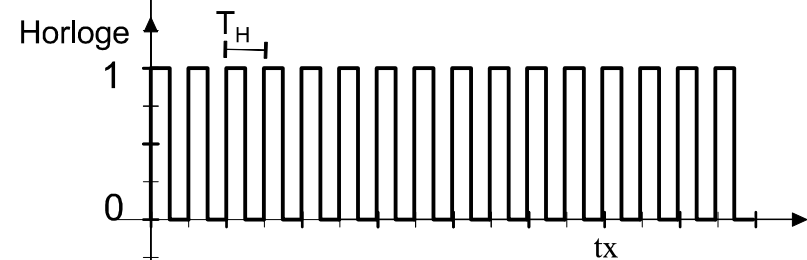
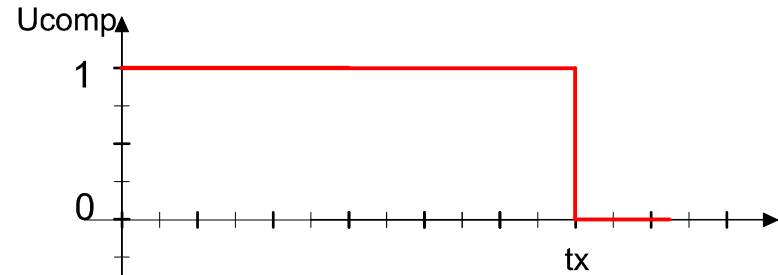
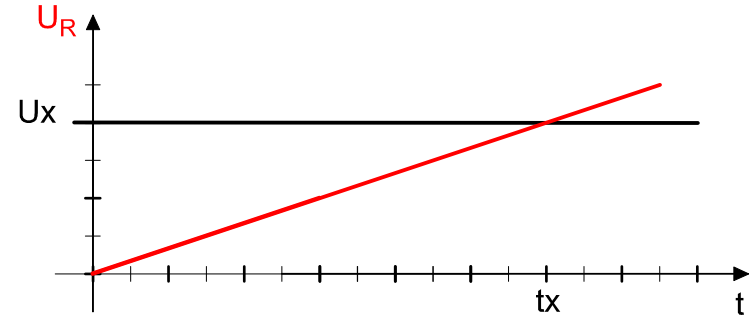
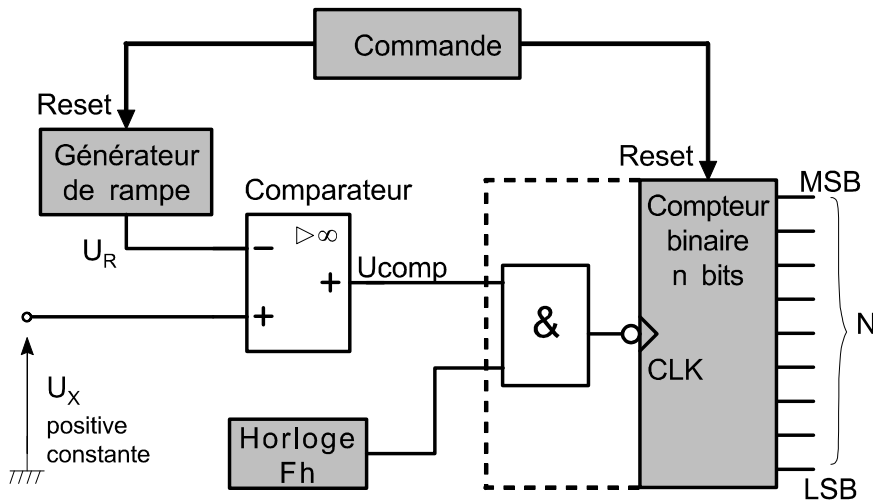
- CAN pipeline
- CAN Sigma-Delta (  $\Sigma-\delta$  ) à sur-échantillonnage



## 2.b/ CAN simple rampe



On effectue une conversion tension  $\rightarrow$  temps ,  
 puis une mesure du temps ( quantifiée )  
 par une horloge de période  $T_H$ .



Simulation Crocodile : **CAN 1 rampe**

$$U_R = a.t$$

$$tx = U_X/a$$

En fin de conversion :

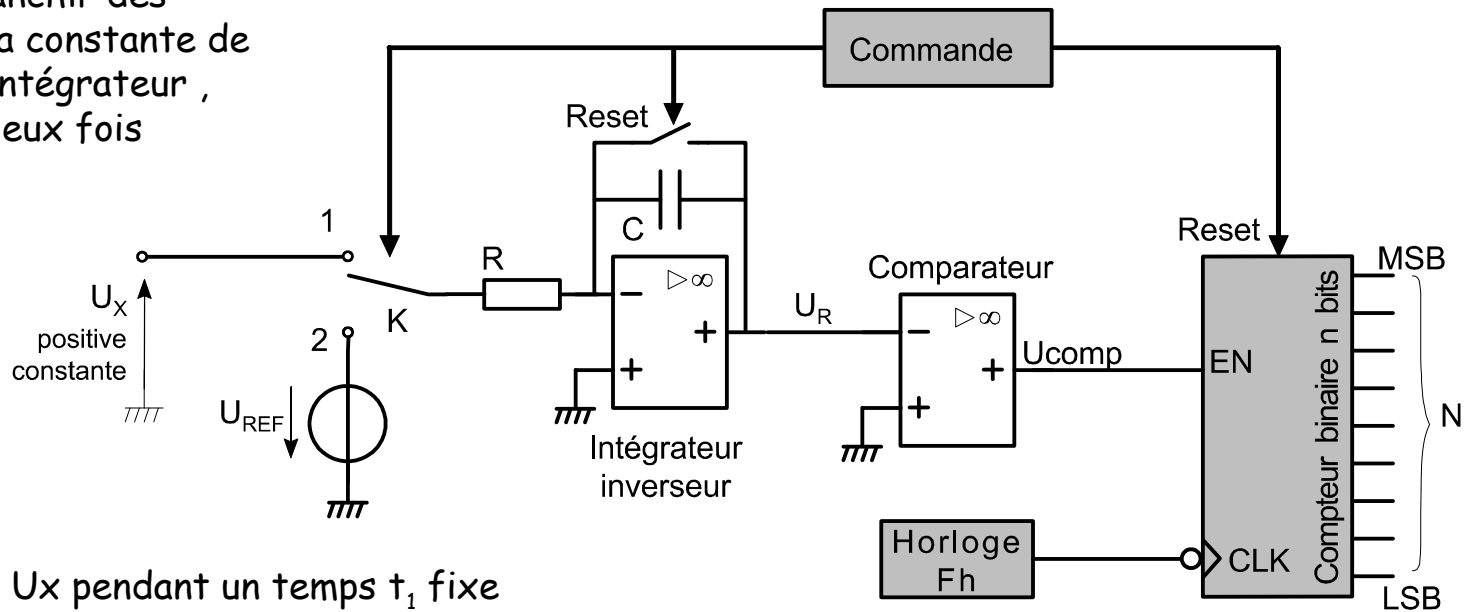
$$N = tx / T_H = U_X / (a.T_H)$$

Si a n'est pas constant  $\rightarrow$  erreur

## 2.c/ CAN double rampe



Pour s'affranchir des dérives de la constante de temps de l'intégrateur, on intègre deux fois



1. On intègre  $U_x$  pendant un temps  $t_1$  fixe

2. On intègre  $-U_{REF}$

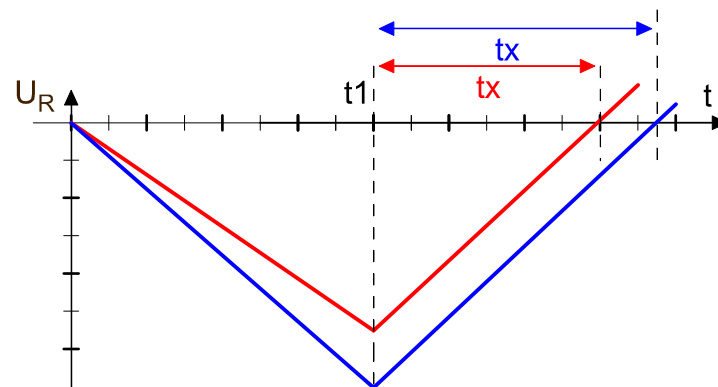
On mesure le temps  $t_x$  ( en unité  $T_H$  )  
que met  $U_R$  pour revenir à 0

$$U_{Rmax} = - U_x \cdot t_1 / RC$$

$$U_{Rmax} = - U_{REF} \cdot t_x / RC$$

$$t_x = t_1 \cdot U_x / U_{REF} \quad \text{indépendant de RC}$$

$$N = t_x / T_H = ( t_1 / T_H ) \cdot ( U_x / U_{REF} )$$



## 2.d/ Voltmètre numérique



Réalisation d'un Voltmètre numérique avec un CAN double rampe : CA3162  
 Affichage sur 3 afficheurs 7 segments avec un décodeur BCD/7seg : CA3161

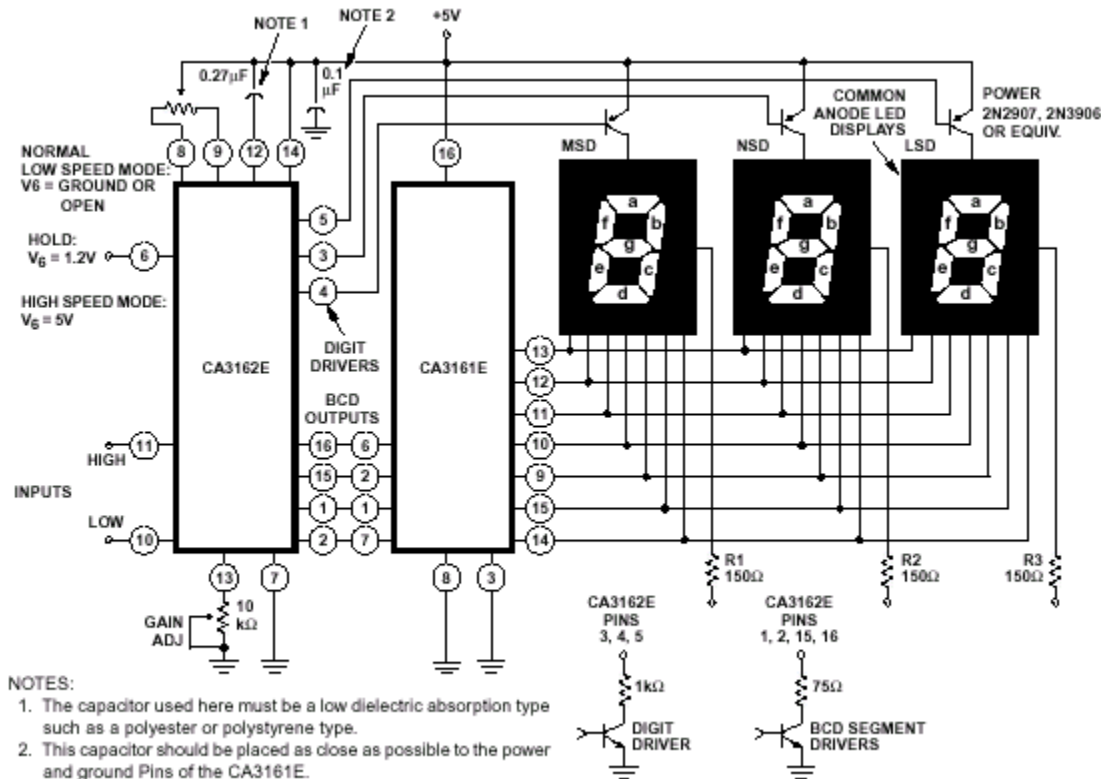
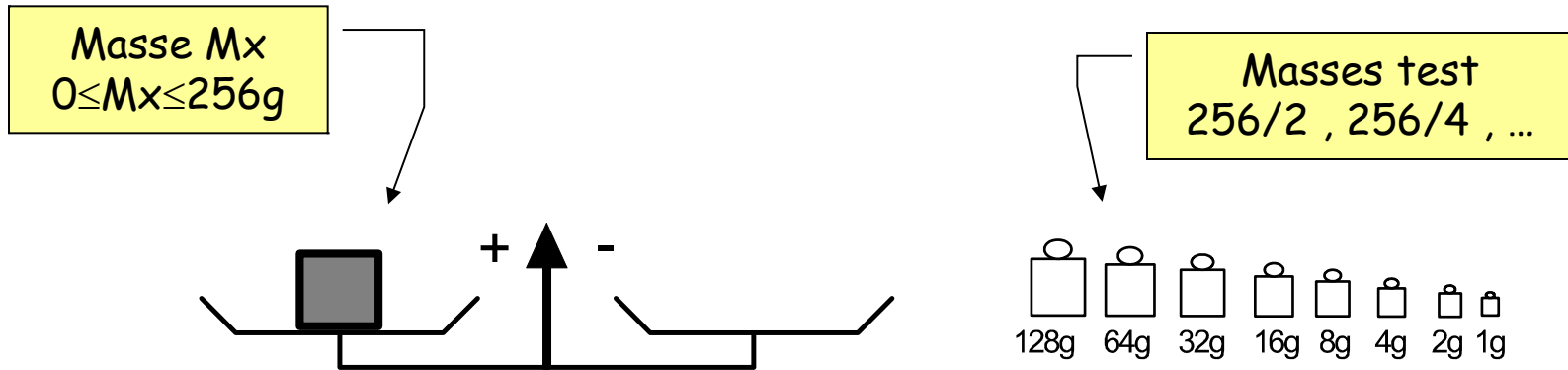


FIGURE 2. BASIC DIGITAL READOUT SYSTEM USING THE CA3162E AND THE CA3161E



## 2.e/ Principe d'une recherche par approximations successives

Principe de Dichotomie : on divise la plage de recherche par 2 à chaque étape :



1er test : on compare  $M_x$  et 128g ( le poids fort )  
- :  $M_x < 128g$  : on enlève la masse de 128g  
+ :  $M_x > 128g$  : on conserve la masse de 128g

2ème test : on ajoute 64g ...

On réalise une mesure de  $M_x$  en **8** tests  
avec une résolution de **1g**

## 2.f/ Approximations successives par transfert de charge



Ce sont les plus courants des CAN à approximations successives ,  
ils utilisent des transferts de charge dans un réseau de condensateurs pondérés.

Le "cerveau" de ces CAN est un registre :  
SAR = Successive Approximation Register

Simulation Crocodile : SAR transfert q

## 2.g/ Approximations successives avec un CNA



Exemple d'un CAN 3 bits

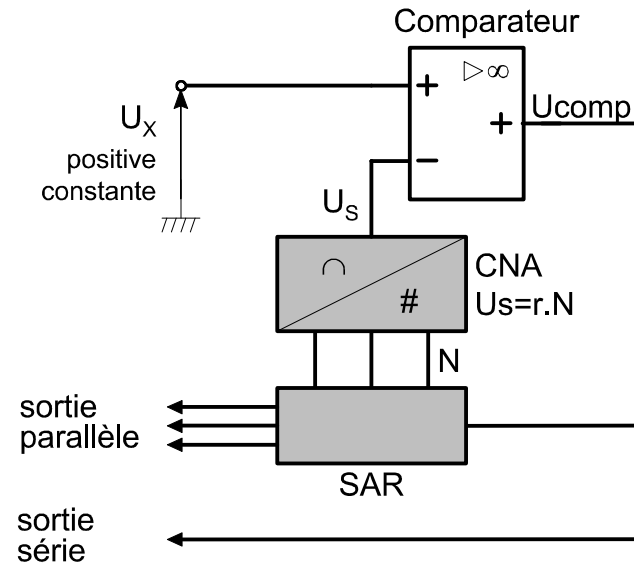
Ce CAN utilise un CNA !

La sortie du CNA est une tension analogique

$$U_s = r.N$$

On teste successivement les bits de N en débutant par le poids fort (MSB)

Le résultat du test est donné par le comparateur.



Exemple avec  $r=1V$ ,  $U_{PE}=8V$ ,  $U_x=4.5V$

N	$U_s=r.N$	$U_{comp}$	
100b=4	4V	1	Test du MSB: $4.5 > 4$ on garde MSB à 1 ...
110b=6	6V	0	et on teste le bit suivant: $4.5 < 6$ on remet le bit à 0
101b=5	5V	0	$4.5 < 5$ , le LSB = 0, le nombre cherché est 100b

Sortie série  
(poids fort en 1<sup>er</sup>)

Sortie parallèle : 100b

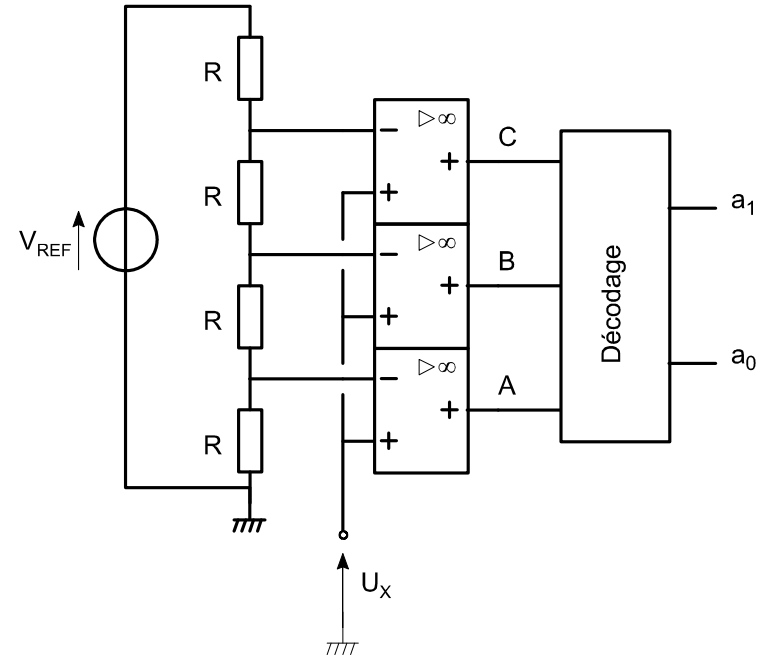
Pour un CAN de n bits il faudra n tests

## 2.h/ CAN flash



Exemple d'un CAN Flash à 2 bits

$U_X$	État de la sortie des comparateurs			N	
	A	B	C	$a_0$	$a_1$
$0 < U_X < V_{REF}/4$	0	0	0	0	0
$V_{REF}/4 < U_X < V_{REF}/2$	1	0	0	0	1
$V_{REF}/2 < U_X < 3V_{REF}/4$	1	1	0	1	0
$3V_{REF}/4 < U_X < V_{REF}$	1	1	1	1	1



Pour un CAN flash à n bits il faut  $2^n - 1$  comparateurs !

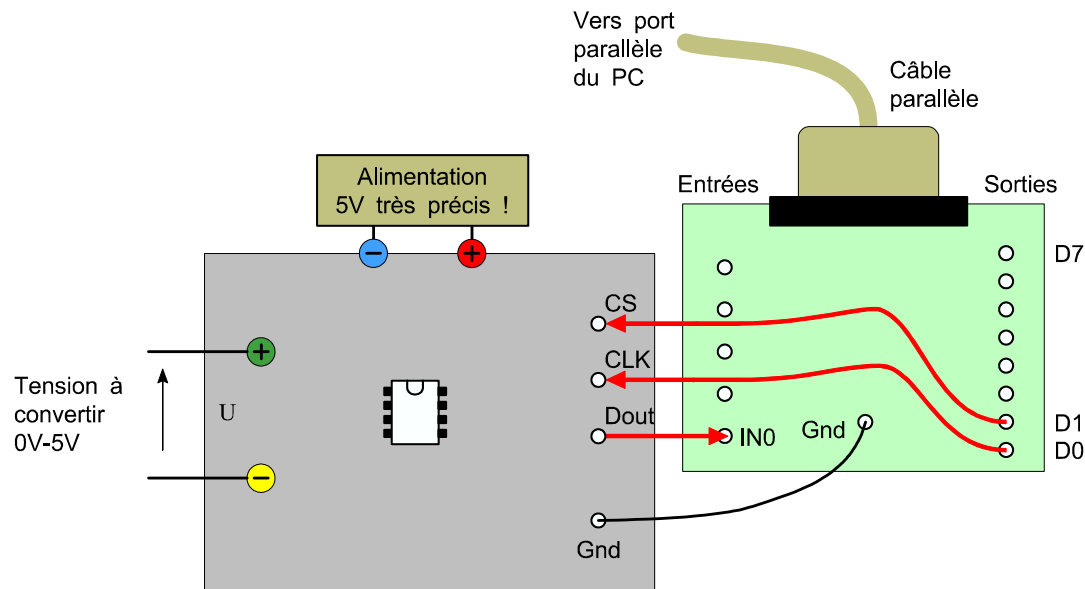
Document ( pdf ) : CAN flash 8 bits [AD9002](#)

## 2.i/ "Micro Cassy"



Exemple de réalisation d'une interface de mesure minimale avec le CAN à approximation successives : **ADC08831**

Interfaçage avec le port parallèle ou le port série du PC



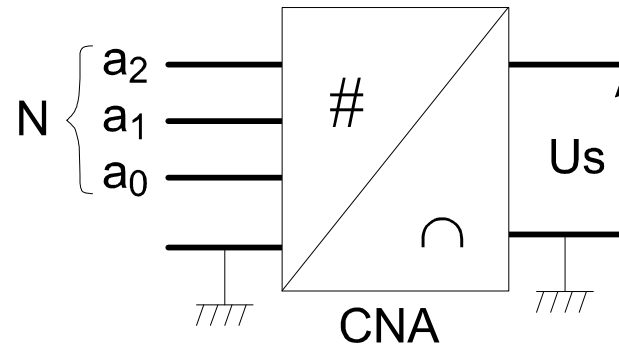
Les programmes associés :  
CAN paral  
CAN serie




### 3.a/ Convertisseur Numérique Analogique

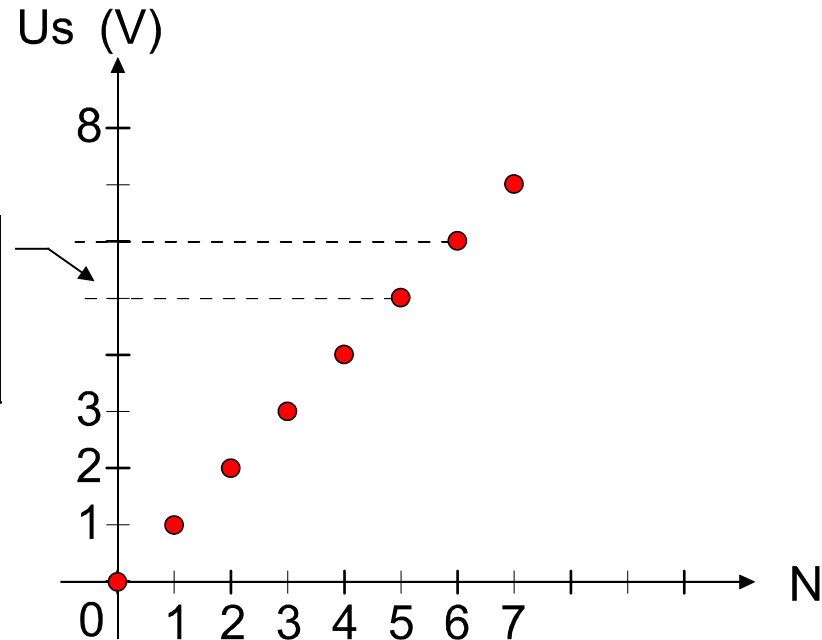


- Exemple d'un CNA 3 bits ( $n = 3$ )



  
Us ne peut pas prendre n'importe quelle valeur :  
$$Us = r \cdot N$$

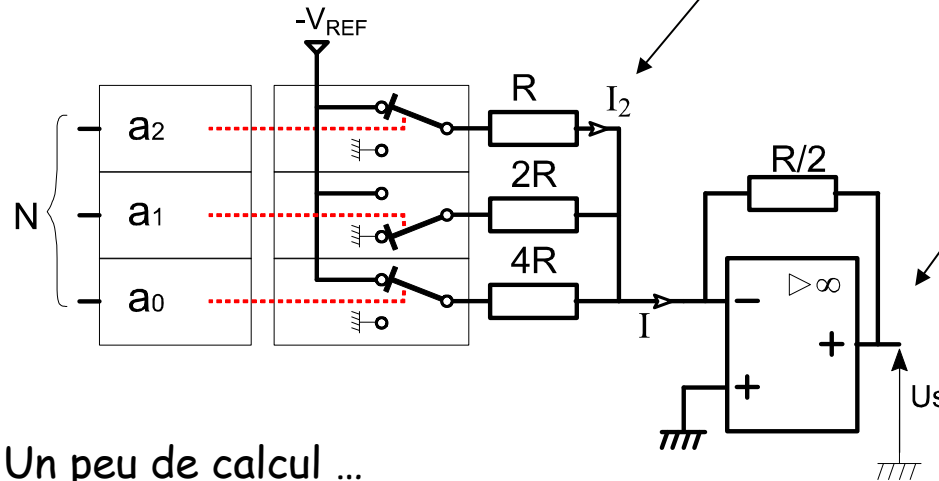
Résolution analogique  
 $r = 1V$



### 3.b/ CNA à résistances pondérées



CNA 3 bits  
( Schéma représenté pour  $N_2 = 101$  )



$a_2=0$  喊  $I_2=0$   
 $a_2=1$  喊  $I_2=-V_{REF}/R$   
 鍼  $I_2=(-V_{REF}/R) \cdot a_2$

En régime linéaire  
 $\varepsilon = V+ - V- = 0$   
 donc  $V- = 0$

Un peu de calcul ...

$$I = (-V_{REF}/R)a_2 + (-V_{REF}/2R)a_1 + (-V_{REF}/4R)a_0$$

$$U_s = -R/2 \cdot I = V_{REF} \cdot (a_2/2 + a_1/4 + a_0/8)$$

$$U_s = V_{REF} \cdot (4a_2 + 2a_1 + a_0) / 8 = V_{REF} \cdot N / 8$$

$U_s = V_{REF} \cdot N / 2^n$

Pleine échelle :  $U_{PE} = V_{REF} \cdot 2^{n-1} / 2^n \approx V_{REF}$

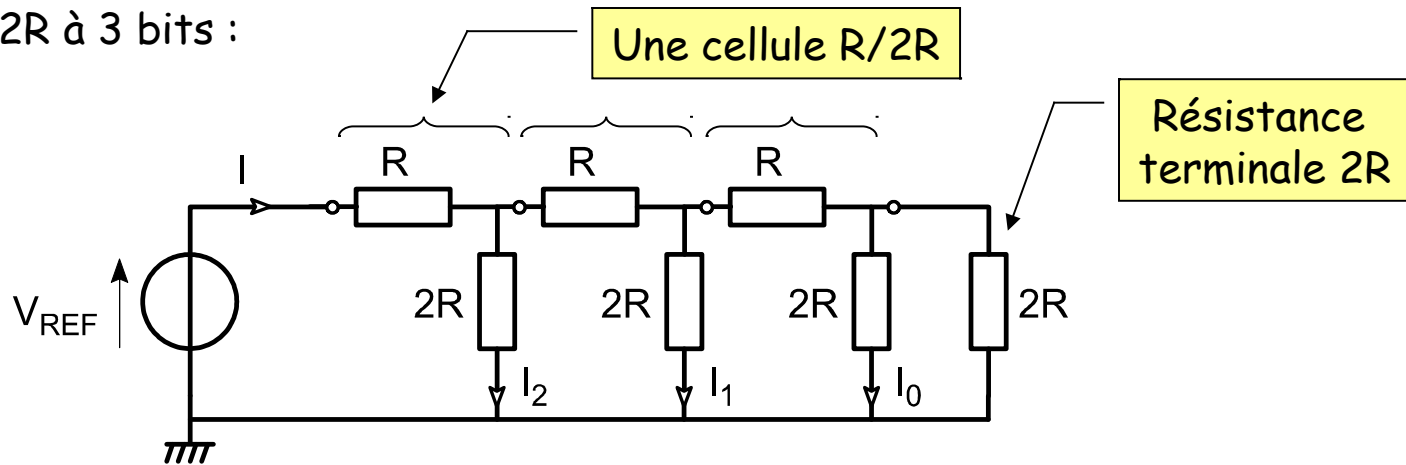
Résolution analogique :  $r = V_{REF} / 2^n \approx U_{PE} / 2^n$

Simulation crocodile :  
 CNA Rpond

### 3.c/ Réseau R/2R



Réseau R/2R à 3 bits :



Chaque cellule R/2R "voit" à sa droite une résistance équivalente de 2R.

Le générateur  $V_{REF}$  "voit" une résistance équivalente de 2R quelque soit le nombre de cellules.

$$I = V_{REF} / 2R$$

$$I_2 = I / 2 = V_{REF} / 4R$$

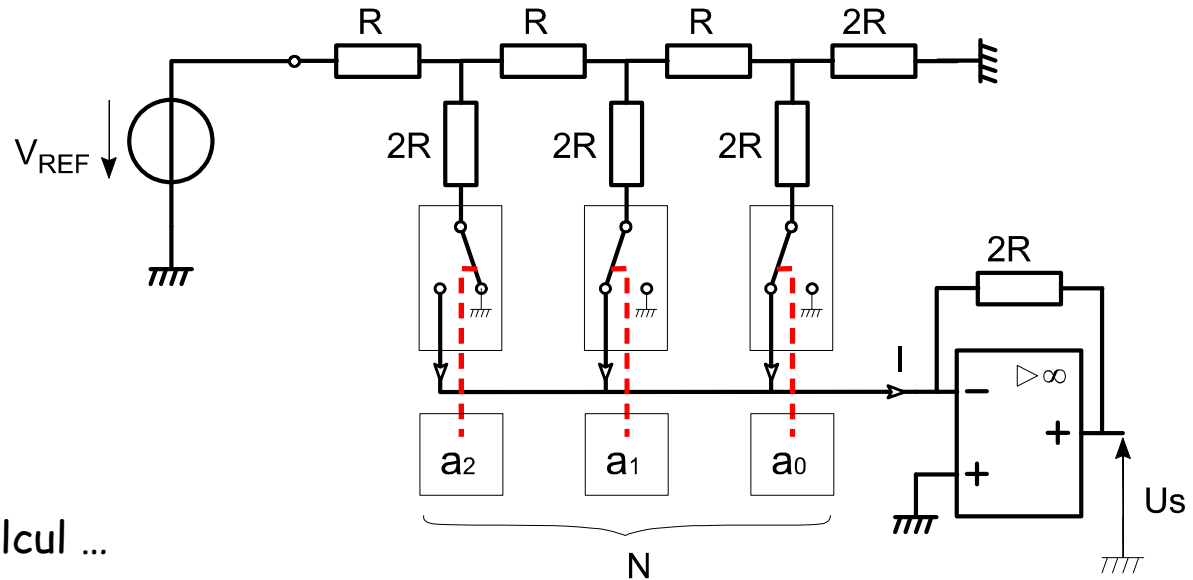
$$I_1 = I_2 / 2 = V_{REF} / 8R$$

$$I_0 = I_1 / 2 = V_{REF} / 16R$$

### 3.d/ CNA à réseau R/2R



CNA 3 bits  
( Schéma  
représenté pour  
 $N_2 = 011$  )



Encore un peu de calcul ...

$$I = I_2 \cdot a_2 + I_1 \cdot a_1 + I_0 \cdot a_0$$

$$I = (-V_{REF}/4R) \cdot a_2 + (-V_{REF}/8R) \cdot a_1 + (-V_{REF}/16R) \cdot a_0$$

$$U_s = -2R \cdot I = V_{REF} ( a_2/2 + a_1/4 + a_0/8 )$$

$$U_s = V_{REF} \cdot ( 4a_2 + 2a_1 + a_0 ) / 8 = V_{REF} \cdot N/8$$

$$U_s = V_{REF} \cdot N / 2^n$$

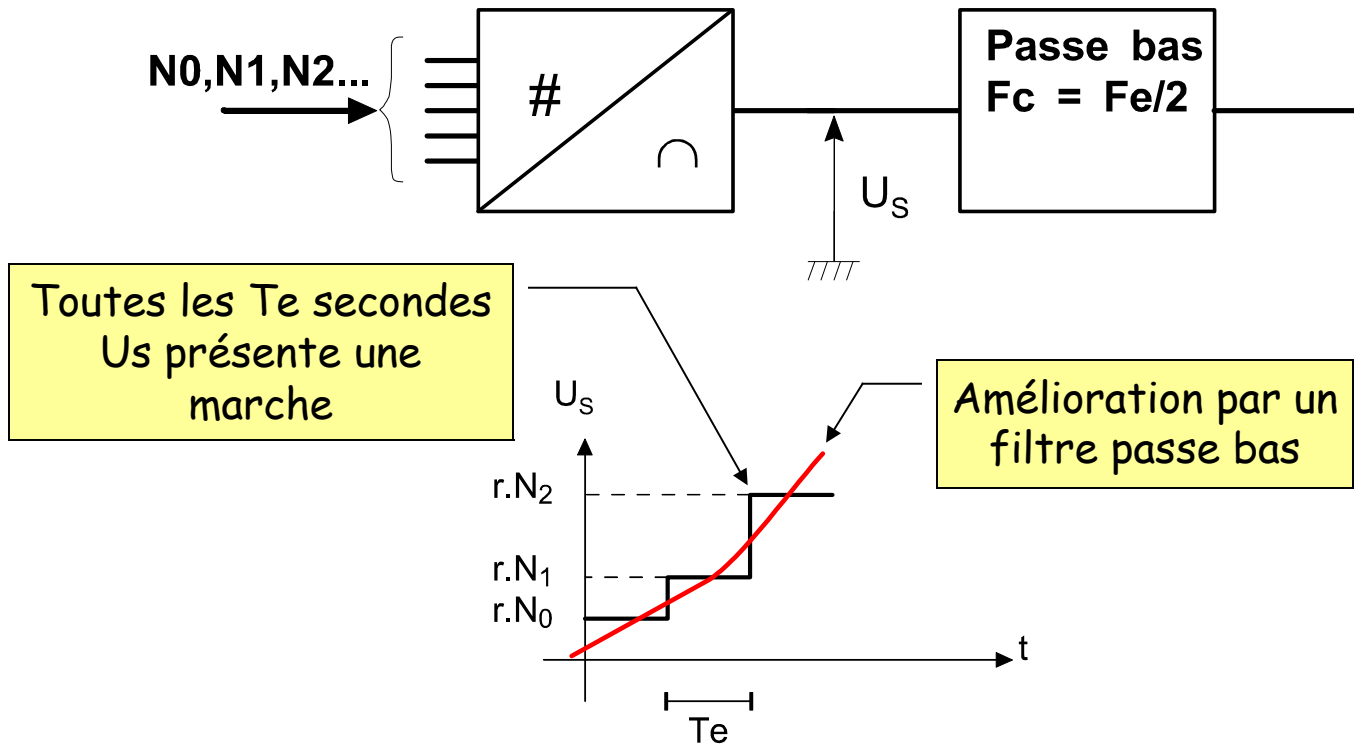
Simulation crocodile :  
CNA R2R

Document ( pdf ) :  
DAC0800

### 3.e/ Restitution d'un signal échantillonné



La séquence des nombres  $N_i$  est présentée à l'entrée du CNA à la fréquence  $F_e$ .



On peut aussi rajouter des valeurs intermédiaires de  $N$  par un calcul d'interpolation.

# Les bonus



Documents ( en Anglais ) sur les technologies de CAN :

Comparaison des différentes technologies : Comparison ADC

CAN pipeline : [Understanding Pipelined ADCs](#) ( pdf )

CAN Sigma Delta : Principles of Sigma Delta ADC

Quelques sites intéressants:

Comment ça marche l'informatique : <http://www.commentcamarche.net/>

Numération ( fait par un élève de MPI ) : <http://numeration.ifrance.com/numeration/>

Recherche de Data Sheet : <http://www.alldatasheet.com/>