



Institut
Mines-Telecom

Electronique des Systèmes Embarqués

Chadi Jabbour

Convertisseur Analogique
Numérique



Introduction

Principe et erreur de Quantification

Source d'erreur et métrique d'évaluation

Architectures principales

Conclusion

Introduction

Principe et erreur de Quantification

Source d'erreur et métrique d'évaluation

Architectures principales

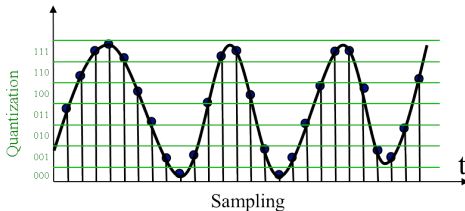
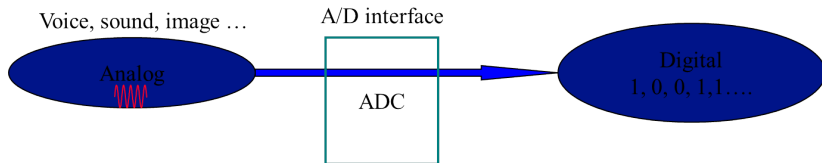
Conclusion

Le monde réel est analogique



Tous les systèmes de com. (filaire, sans fil, sur fibre), tous les systèmes de détection (radar, capteur de distance ...), tous les systèmes audio, un moment ou un autre, sont analogiques

Interface



Convertisseur Analogique Numérique

Le Convertisseur Analogique Numérique est l'interface entre le monde réel analogique et le monde numérique de calcul.

Introduction

Principe et erreur de Quantification

Source d'erreur et métrique d'évaluation

Architectures principales

Conclusion

La conversion analogique numérique requiert 2 étapes principales:

- ▶ Une discrétisation dans le temps ou un échantillonnage
- ▶ Une discrétisation en valeur ou la quantification qui consiste à réduire l'ensemble des valeurs traitées à un ensemble de valeurs connues.

Convertisseur Analogique Numérique

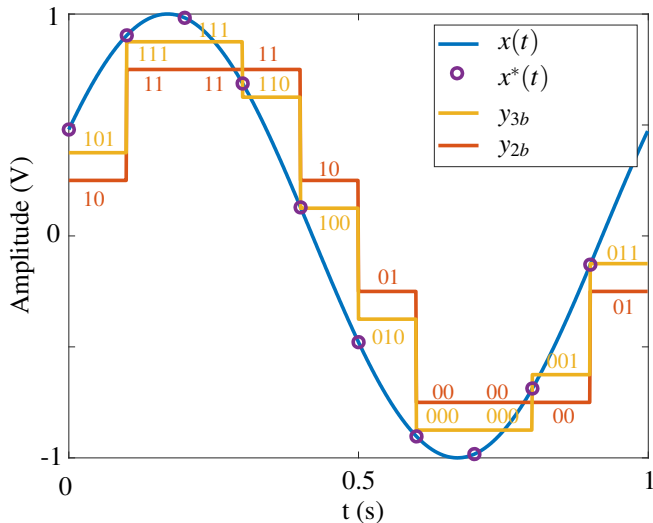
On utilise l'acronyme CAN pour désigner les Convertisseurs Analogique Numérique, ainsi que ADC pour *Analogue to Digital Converter*.

Fréquence d'échantillonnage

La fréquence d'échantillonnage est notée f_e , la notation f_s pour *Sampling frequency* est aussi souvent utilisée.

Exemple de quantification

$$f_{\text{signal}} = 1 \text{ Hz} \quad A_{\text{Signal}} = 1 \text{ V} \quad f_e = 10 \text{ Hz}$$



Définition

La grandeur analogique $x[k]$ est transformée en un signal num. (b_1, b_2, \dots, b_{nb})

$$x[k] = b_1[k] \frac{PE}{2} + b_2[k] \frac{PE}{4} + \dots + b_n[k] \frac{PE}{2^{nb}} + \frac{PE}{2^{nb+1}} + e$$

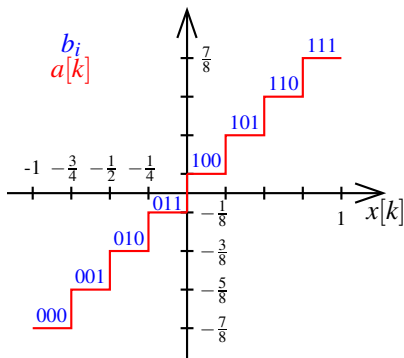
$$x[k] = N[k] \frac{PE}{2^{nb}} + \frac{PE}{2^{nb+1}} + v_{min} + e = N[k]q + \frac{q}{2} + v_{min} + e[k],$$

- ▶ PE : la pleine échelle du convertisseur, égale à la différence entre la valeur max. v_{max} et min. v_{min} supportées par le convertisseur.
- ▶ e : l'erreur de quantification du convertisseur comprise entre $\pm q/2$
- ▶ N : la sortie numérique du convertisseur
- ▶ nb : le nombre de bit ou la résolution du CAN
- ▶ b_1 : le bit de poids le plus fort (MSB : *Most Significant Bit*) et b_{nb} : le bit de poids le plus faible (LSB : *Least Significant Bit*).
- ▶ Pas de quantification ou quantum $q = \frac{PE}{2^{nb}}$.

On peut ainsi définir la grandeur $a[k]$ qui correspond à l'estimation de $x[k]$ avec:

$$a[k] = v_{min} + N[k]q + \frac{q}{2} = x[k] - e[k]$$

Exemple



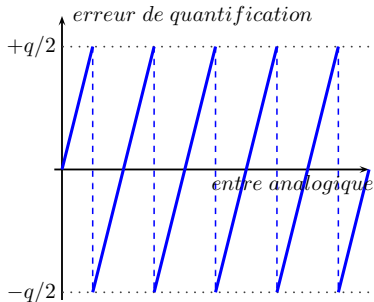
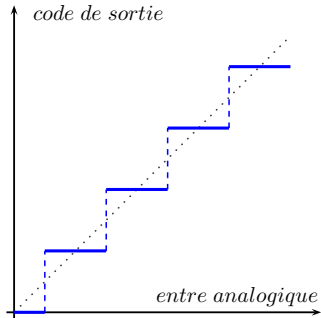
$$nb=3 ; PE=2 \text{ V} ; q=0.25 \text{ V}$$

Pour $x[k] = -0.53 \text{ V} \implies N[k] = 1 ; b_i[k] = 001$

$a[k] = -1 + 0.25 + 0.125 = -0.625 \implies$

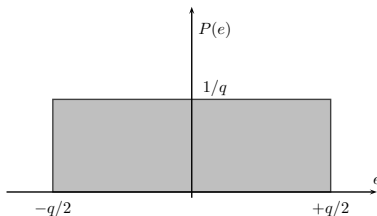
$e[k] = x[k] - a[k] = 0.095 \text{ V}$

Principe



Quand le nombre de bits est suffisamment élevé, on peut approximer que l'erreur de quantification est uniformément distribuée entre $\pm \frac{q}{2}$

Erreur de quantification



$$\sigma^2 = \frac{1}{q} \int_{-q/2}^{+q/2} e^2 de = \frac{q^2}{12}.$$

$$SQNR = \frac{\text{puissance du signal}}{\text{puissance du bruit}}$$

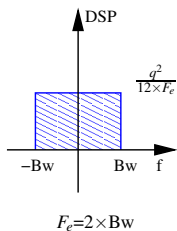
$$SQNR = \frac{\int_{-f_e/2}^{+f_e/2} |\text{Signal}(f)|^2 df}{\int_{-f_e/2}^{+f_e/2} dspe df}$$

$$\text{Si Signal} = \text{Amp} \sin(2\pi ft), \implies SQNR = \frac{3}{2} \cdot 2^{2nb} \cdot \left(\frac{2 \cdot \text{Amp}}{PE}\right)^2$$

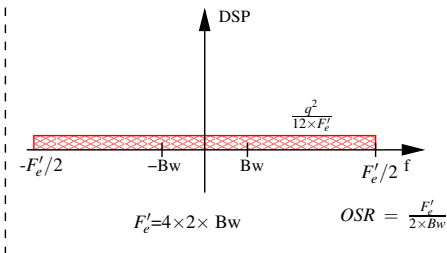
En dB:

$$SQNR_{dB} = 10 \log(SQNR) = 1,76 + 6,02nb + 20 \log\left(\frac{2 \cdot \text{Amp}}{PE}\right)$$

Sur-échantillonnage



a)



b)

Effet du sur-échantillonnage

Quand on sur-échantillonne, la puissance du bruit de quantification reste constante mais est étalée sur une bande de fréquence plus large

L'expression générale du SQNR devient

$$SQNR_{dB} \approx 6.02nb + 1.76 + 20 \log_{10}\left(\frac{2 \cdot Amp}{PE}\right) + 10 \log_{10}\left(\frac{f_e}{2 \cdot Bw}\right)$$

Introduction

Principe et erreur de Quantification

Source d'erreur et métrique d'évaluation

Architectures principales

Conclusion

La précision des CANs (et de tous les systèmes électroniques) est dégradée par deux sources principales:

- ▶ Le bruit modélisé par un signal aléatoire ajouté au signal utile
 - ▶ Bruit blanc: Densité Spectrale de Puissance (DSP) constante ou uniforme
 - ▶ Bruit coloré: DSP non uniforme
- ▶ Les distorsions sont des erreurs dont l'amplitude dépend du signal d'entrée
 - ▶ Distorsion linéaire assimilable à du filtrage
 - ▶ Distorsion non linéaire

- ▶ Bruits blancs:
 - ▶ Bruit thermique du à l'agitation des porteurs dans les résistances et transistors
 - ▶ Bruit de quantification
 - ▶ Bruit de phase qui cause la gigue d'horloge ou *jitter*

- ▶ Bruits colorés
 - ▶ Bruit de scintillement, *Flicker noise* ou bruit $1/f$
 - ▶ Pop corn noise

Les causes des distorsions non linéaires sont diverses:

- ▶ Saturation
- ▶ Interférence entre symbole
- ▶ Désappariements des composants ou *mismatch*

Les non-linéarités causent:

- ▶ Une dégradation de la résolution
- ▶ L'apparition de nouvelles composantes dans le spectre aux multiples de la fréquence d'entrée.

Le rapport Signal à Bruit SNR permet d'évaluer les dégradations dues au bruit:

$$SNR_{dB} = 10 \log_{10} \left(\frac{P_{signal}}{P_{Bruit}} \right)$$

Le THD permet d'évaluer les dégradations dues aux non linéarités

$$THD_{dB} = -10 \log_{10} \left(\frac{P_{signal}}{P_{Distorsion}} \right)$$

Le SNDR ou SINAD (*Signal to noise and distortion ratio*) permet d'évaluer les dégradations dues aux 2

$$SNDR_{dB} = 10 \log_{10} \left(\frac{P_{Signal}}{P_{Bruit} + P_{Distorsion}} \right)$$

Exemple de spectre - AD7682

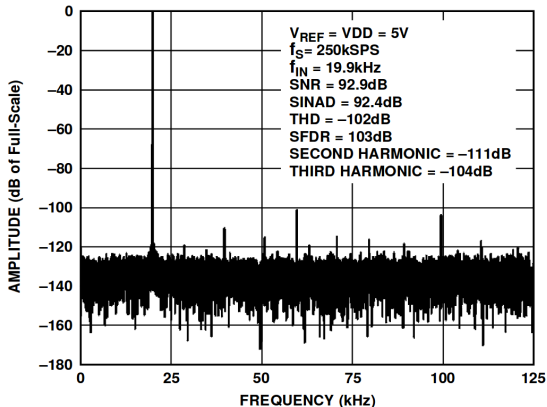


Figure 10. 20 kHz Fast Fourier Transform (FFT), $V_{REF} = V_{DD} = 5V$

SAR ADC AD7682 de Analog Devices

Spécifications des CANs

- ▶ Résolution: SNR, SNDR ou SINAD, THD, Dynamic range (DR) , Effective number of bits (ENOB) $ENOB = \frac{SNDR-1.76}{6.02}$
- ▶ Linearité: SFDR, THD, IIP3, IIP2, DNL, INL, ...
- ▶ Vitesse: Data rate ($2 \times Bw$ if no oversampling)
- ▶ Consommation de Puissance

Figure de mérite:

- ▶ Walden FoM (the lower the better)
Typical values: 10 - 50 fJ per conversion step

$$FoM_{Walden} = \frac{Power\ Consumption}{2 \cdot Bw \cdot 2^{ENOB}}$$

- ▶ Thermal FoM (the higher the better)
Typical values: 160 - 180 dB

$$FoM_{Thermal} = 10 \log \left(\frac{BW}{P} \right) + SNDR$$

Autre caractéristiques des CANs

- ▶ Robustesse par rapport aux variations de la température, de tension d'alimentation, de vieillissement ...
- ▶ Impédances d'entrée et de sortie
- ▶ Interface d'entrée/sortie (type de connecteurs, protocole, niveau de tension ...)
- ▶ Dimension, prix, pérenité du produit ...

Introduction

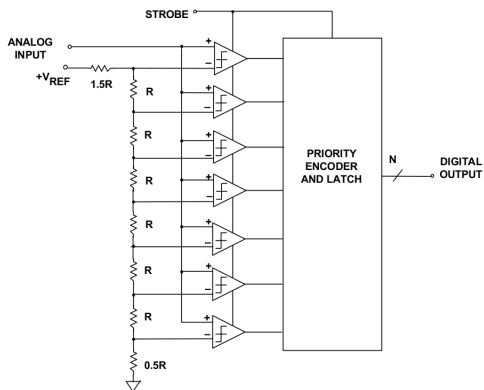
Principe et erreur de Quantification

Source d'erreur et métrique d'évaluation

Architectures principales

Conclusion

Architectures de CAN - Flash

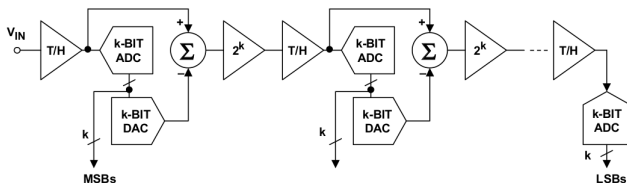


- ▶ Un Flash N_b bits nécessite 2^{N_b-1} comparateurs (résolution limitée)
- ▶ La conversion se fait en un seul cycle d'horloge (pas de retard)
- ▶ Le codeur convertit le code thermomètre de sortie en un code binaire

La précision est limitée par les performances du comparateur (Offset, métastabilité, hystérésis ...) et le matching entre résistances.

CAN Pipeline

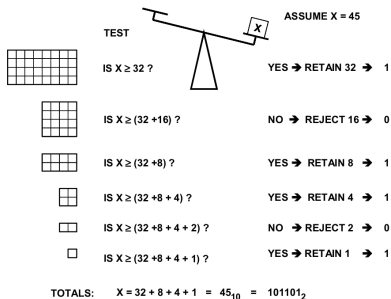
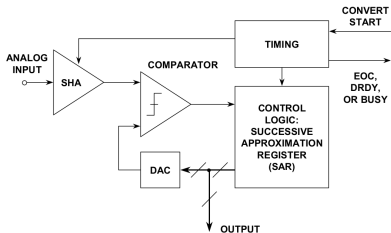
La conversion se fait à l'aide de plusieurs étages permettant d'atteindre une résolution plus élevée qu'un Flash avec moins de comparateurs



- ▶ Le premier étage effectue une quantification grossière qui est affinée par les étages suivants
- ▶ Un Sample&Hold ou Track&Hold et des amplificateurs sont nécessaires entre les étages pour amplifier l'erreur à PE

SAR ADC

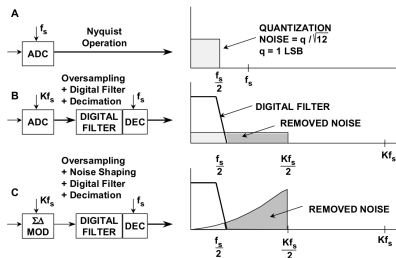
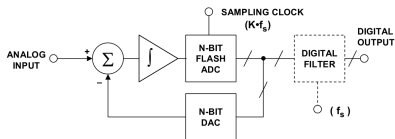
CAN à approximations succesives basée sur la dichotomie



- ▶ La résolution est donnée par le nombre de cycles de conversion
- ▶ Le convertisseur numérique analogique est critique pour la consommation et la performance dans cette architecture

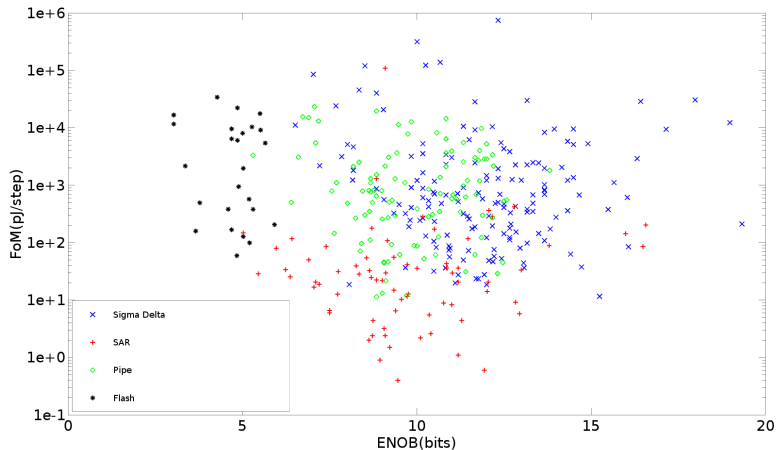
CAN $\Delta\Sigma$

Les CAN $\Delta\Sigma$ utilisent le sur-échantillonnage et la mise en forme du bruit pour atteindre des résolutions élevées.



- ▶ Le bruit de quantification est “poussé” hors de la bande utile
- ▶ Un filtre de décimation est requis en sortie pour revenir à la cadence de Nyquist

Architectures de CAN - Comparaison



FOM vs ENOB in conferences ISSCC and VLSI over the last two decades

Architectures de CAN - Comparaison

	Flash	Pipeline	SAR	$\Delta\Sigma$
Vitesse	Très élevé	élevé	Moyen	Faible
Résolution	Faible	élevé	élevé	Très élevé
Cons. de Puis.	Très élevé	élevé	Très Faible	Faible
Surface	Très élevé	élevé	Faible	Moyen
Désavantages	Résolution Limitée. Nb comparateurs ↗ exp.	Limité par le S/H	Limité par le matching du DAC ou CNA	Instabilité et horloge haute fréquence

Introduction

Principe et erreur de Quantification

Source d'erreur et métrique d'évaluation

Architectures principales

Conclusion

Conclusion

- ▶ Le CAN (ou ADC en anglais) est un élément clé dans l'interfacage des SE
- ▶ Principales spécifications: f_e , PE, résolution, interfaces d'entrée de sortie ...
- ▶ Plusieurs d'architectures sont disponibles, le choix se fait en fonction des spécifications
- ▶ Le principe des Convertisseurs Numérique Analogiques (ou DAC en anglais) est similaire mais avec des contraintes différentes

Exercice: capteur cardiaque

Nous souhaitons faire le *reverse engineering* d'un CAN proposé par un concurrent. Ce CAN est optimisé pour la numérisation du signal de sortie d'un capteur cardiaque. On sait que le signal numérisé à la sortie du CAN a un SNR de 40 dB. On sait aussi que 3 types de bruit y sont présents: le bruit thermique, le bruit $1/f$ *flicker noise* et le bruit de quantification. La bande d'intérêt du signal est comprise entre 0 et 50 Hz.

1. Vu que le *flicker noise* a un comportement en $1/f$, ceci veut dire que sa puissance $\rightarrow \infty$ quand $f \rightarrow 0$, pourquoi cela n'est pas un problème?
2. En analysant la *datasheet* du composant et en s'appuyant sur d'autres méthodes qu'on ne détaillera pas car l'espionnage industriel est interdit, on détermine les PSD des bruits:

$$PSD_{Flicker} = \frac{2.1 \times 10^{-7}}{f} \text{V}^2/\text{Hz} \quad ; \quad PSD_{Therm} = 10^{-7} \text{V}^2/\text{Hz}$$

Calculer les puissances du bruit thermique et $1/f$ sur $1 \mu\text{Hz}$ à 50 Hz.

3. Sachant que le signal utile a une puissance de 0.1V^2 , calculer la puissance maximale autorisée pour le bruit total à la sortie du CAN. En déduire la puissance maximale autorisée pour le bruit de quantification et le budget de bruit adopté par le concurrent pour son CAN.
4. Calculer le nombre de bits nécessaire pour le CAN sans sur-échantillonnage avec une pleine échelle de 1 V.

1-Pour étudier des fréquences proches de zéro, il faut avoir une durée de fonctionnement très longue. En fonction de comment le système est dimensionné, la fréquence nécessaire pour dégrader le SNR au dessous d'une valeur limite peut nécessiter des dizaines ou des centaines d'années de fonctionnement.

2- Pour calculer les puissances de bruit, il suffit d'intégrer le bruit sur la bande d'intérêt

$$P_{Flicker} = \int_{10^{-6}}^{50} PSD_{Flicker}(f) \cdot df = 2.1 \times 10^{-7} \ln\left(\frac{50}{1e-6}\right) = 3.72 \times 10^{-6} \text{ V}^2$$

$$P_{Therm} = \int_{10^{-6}}^{50} PSD_{Therm}(f) \cdot df = 10^{-7} \times (50 - 10^{-6}) \simeq 5 \times 10^{-6} \text{ V}^2$$

3- Nous savons que:

$$SNR_{glob-dB} = 10 \log \left(\frac{P_{signal}}{PB - total} \right)$$

On en déduit que:

$$PB - total = \frac{P_{signal}}{10^{SNR_{glob-dB}/10}} = \frac{0.1}{10^4} = 10^{-5} \text{ V}^2$$

Vu que le bruit global est la somme des 3 bruits thermique, 1/f et et quantification, on en déduit que:

$$PB_{Quant} = PB_{total} - (PB_{Therm} + PB_{flicker}) = 1.27 \times 10^{-6} \text{ V}^2$$

Le budget de bruit est tout simplement la contribution de chacun des bruits: le bruit thermique a une contribution de 50%, 37% pour le bruit 1/f et 13% pour le bruit de quantification.

4-On sait que le bruit que quantification est donné par :

$$PB_{quant} = \frac{q^2}{12} = \frac{PE^2}{12 \cdot 2^{2nb}}$$

Avec $PE=1 \text{ V}$ et $PB_{Quant} = 1.27 \times 10^{-6} \text{ V}^2$, on obtient $nb = 8$;

Merci pour votre attention

Questions ?