ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2015-ENST-00xx

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TÉLÉCOM ParisTech

Spécialité « Electronique et Communications »

présentée et soutenue publiquement par

Gaël Kamdem De Teyou

le 31 Août 2015

Calibration aveugle et adaptative des convertisseurs analogique-numérique entrelacés temporellement

Directeur de thèse : **Patrick LOUMEAU**
Co-encadrement de la thèse : **Hervé PETIT**

Jury

M. Philippe Benabes, Professeur, Supélec, Gif sur Yvette	Président
M. Dominique DALLET, Professeur, ENSEIRB, Bordeaux	Rapporteur
M. Dominique MORCHE, Directeur de Recherche, CEA-Leti, Grenoble	Rapporteur
M. Stéphane PAQUELET, Responsable de Laboratoire, B-COM, Rennes	Examinateur
M. Patrick LOUMEAU, Professeur, Télécom ParisTech, Paris	Directeur de thèse
M. Hervé PETIT, Maitre de Conférences, Télécom ParisTech, Paris	Co-Directeur de thèse

TÉLÉCOM ParisTech
école de l'Institut Mines-Télécom - membre de ParisTech
46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

T H È S E

Adaptive and Blind Background Calibration of Channel Mismatches in Time-Interleaved ADCs

Gaël Kamdem De Teyou

September 23, 2015

Acknowledgment

Several persons have supported me during this PhD research. First of all, I would like to thank deeply my supervisors, Patrick Loumeau and Hervé Petit for their openness, guidance and support during the course of this work. I am also grateful to Renesas Design France which funded this research and where I had the opportunity to work during my PhD. Especially, I want to thank warmly Stephane Paquelet and Yann Le Guillou of Renesas who initiated this work and who help me to understand the basics of ADCs.

My sincere thanks are due, to Philippe Benabes, Dominique Dallet and Dominique Morche members of this jury, for their availability and who accepted to examine this work during summer.

I am extremely thankful to Hussein Fakhoury and Chadi Jabbour of Télécom ParisTech for useful discussions and comments during this work. Also thanks to my friends and colleagues at Télécom ParisTech for the fun we had together.

Special thanks go to my fiancée Lorie, for her patience, encouragement, and support throughout my research. My brothers Harold, Maxime and Jordan for their moral support. Above all, I would like to thank my parents for a never-ending support and for all they did for me.

Abstract

Wireless receivers for emerging Software Define Radio require faster and more accurate converters. A popular way to achieve this is by designing Time-Interleaved ADCs (TI-ADCs). First low speed and high resolution sub-ADCs are built, then they are interleaved, working alternatively as if they were a single ADC but working at a much higher rate. TI-ADCs have emerged as a good way to provide high speed data converters from relatively slow circuits. However, unfortunately in this kind of architecture, new errors emerge and give rise to nonlinear distortion which significantly degrade the resolution of the overall TI-ADC. These errors come from discrepancies between the individual sub-ADCs in the system and are commonly referred to as channel mismatch errors. They consist of gain, time-skew, bandwidth and offset mismatch errors and they should be mitigated.

There are two possible ways to deal with channel mismatches. The first is to complexify the analog circuit design of the ADC in order to reduce the magnitude of the original mismatches at the cost of more power consumption and area. But this increases the time-to-market of the ADC. The second solution is to alleviate the design and to correct the errors with a calibration technique.

In this work, a new digital blind calibration technique is proposed for TI-ADCs which is able to correct the gain, time-skew, bandwidth and offset mismatches. The technique can be divided into two independent steps. The first step is estimation and it consists in identifying the mismatches from the measurements. In this step we use a rational fractional delay and a low-pass filter to estimate adaptively the different mismatches. The second step is the compensation which consists in reducing the errors due to channel mismatches. This stage is based on the development of a matrix approach to find the suitable filters to apply at the output of different channels.

This technique was tested on 14 bits ADCs from Analog Devices and the results show the effectiveness of the technique.

Contents

Notations

Symbols

For all this report, we use the following symbols.

Symbol	**Explanation**
q	Quantization step, also called Least Significant Bit
M	Number of time interleaved ADC
N	Resolution of the ADC
m	Index identifying the m^{th} ADC
f_s	Sampling frequency of the TI-ADCs
$\frac{f_s}{M}$	Sampling frequency of an individual ADC
Ts	Sampling period of the TI-ADCs
G_m	Gain of the m^{th} ADC
O_m	Offset of the m^{th} ADC
$\xi[n]$	Jitter of the ADC at the instant nT_s
τ_m^s	Time Skew of the m^{th} ADC
$h_m(t)$	Impulse response of the m^{th} ADC
$INL_m(i)$	Integral Non-Linearity of code i of the m^{th} ADC
$DNL_m(i)$	Differential Non-Linearity of code i of the m^{th} ADC.

Abbreviations

For all this report, we use the following abbreviations.

Abbreviation	**Explanation**
ADC	Analog to Digital Converter
DTFT	Discrete Time Fourier Transform
ENOB	Effective Number Of Bits
PLL	Phase Locked Loop
PDF	Probability Density Function
PSD	Power Spectral Density
SC	Switched Capacitor
SDR	Software Defined Radio
SFDR	Spurious Free Dynamic Range
S/H	Sample and Hold
SNDR	Signal to Noise and Distortion Ratio
SNR	Signal to Noise Ratio
TI-ADCs	Time Interleaved Analog to Digital Converters
THD	Total Harmonic Distortion

French Summary

0.1 Introduction

Il y a une augmentation permanente du débit de données dans les standards de communication sans fil. C'est ainsi qu'on est passé de 14.4 Kbits/s en 2G pour atteindre 384 Kbits/s en 3G, puis 1 Gbits/s en 4G et plusieurs Gbits/s sont attendus en 5G bien qu'elle ne soit pas encore standardisée. Ceci a été rendu possible grâce à un certain nombre d'opérations sur la couche physique parmi lesquelles l'augmentation de la largeur de bande. C'est ainsi qu'on est passé de 200 KHz par porteuse en 2G pour pouvoir atteindre dans certains cas 100 MHz en 4G grâce à l'agrégation de porteuses, c'est-à-dire une largeur de bande 500 fois plus élevée qu'en 2G. Cette augmentation de la largeur de bande nécessite des Convertisseurs Analogique-Numérique (CAN) plus rapides car le signal doit être échantillonné au niveau du récepteur à au mois deux fois la largeur de bande.

Pour assurer la mobilité de l'usager afin qu'il puisse se connecter partout, le terminal doit intégrer plusieurs bandes et les anciens standards doivent coexister avec les nouveaux. Ceci constitue un gros challenge au niveau de l'architecture de récepteur. La Fig. 1 par exemple illustre l'architecture d'un récepteur superhétérodyne conventionnel. Il y a une partie de la RF qui est partagée puis il y a une chaine de réception spécifique à chaque canal. Ainsi la complexité d'un tel récepteur en termes de coût matériel augmente drastiquement avec le nombre de standards et de canaux. La solution pour régler ce problème est la radio logicielle (Software Defined Radio). Elle comprend une chaine de réception principalement réalisée en numérique (logiciel) et dans une moindre mesure en matériel. La même infrastructure matérielle peut alors être réutilisée pour les différents standards mais avec cette fois-ci un logiciel qui est spécifique au standard et au service souhaité. Ceci est rendu possible en déplaçant le CAN autant que possible vers l'antenne de façon à échantillonner le signal le plus tôt possible et effectuer le reste de traitements dédiés en numérique comme le montre la Fig. 2. Cependant déplacer le convertisseur vers l'antenne augmente les contraintes sur celui-ci qui doit être plus rapide tout en conservant une bonne résolution.

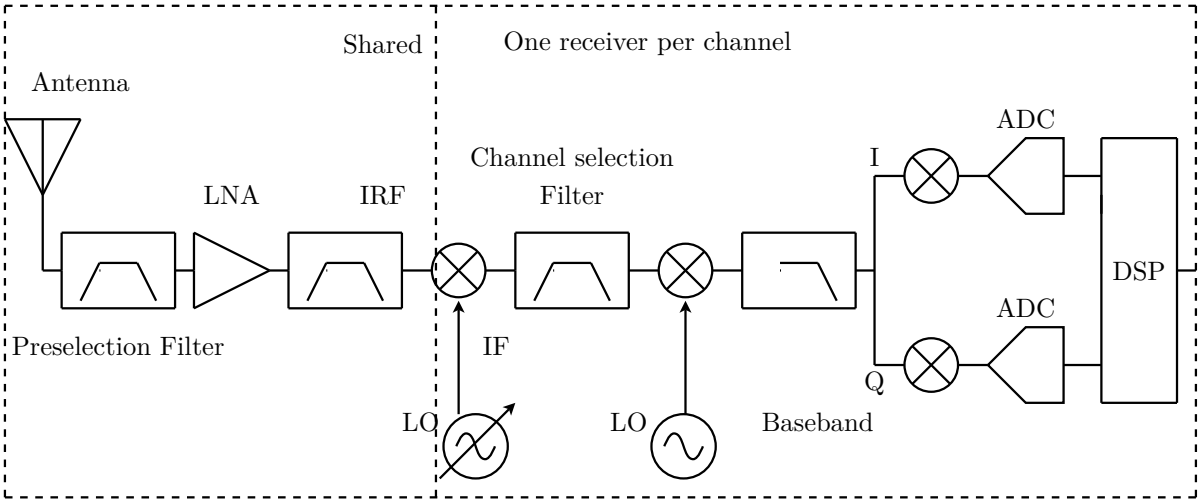


Figure 1: Architecture d'un récepteur superheterodyne conventionnel.

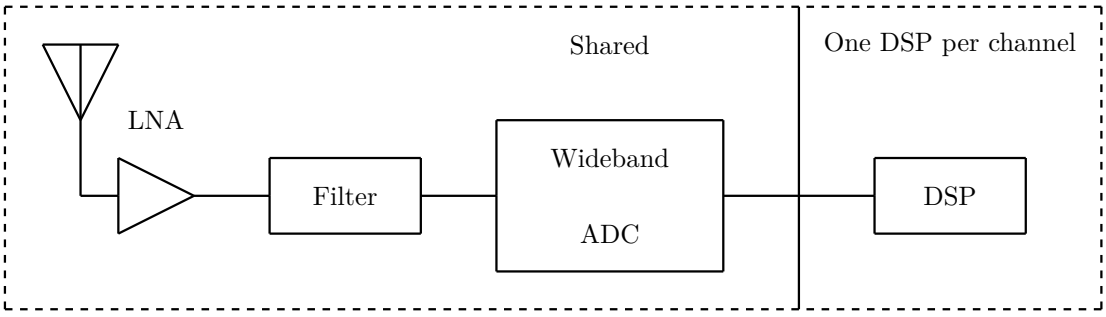


Figure 2: Radio logicielle idéale.

La Fig. 3 montre un état de l'art des CAN sur les 18 dernières années. Nous avons représenté les CAN à approximations successives qui sont très appréciés pour leur bonne résolution et leur faible consommation en puissance. Nous pouvons remarquer que dans l'ensemble leur fréquence d'échantillonnage est limitée à la centaine de mégahertz. Si nous visons des vitesses d'échantillonnage supérieures comme par exemple 230 MHz pour le LTE-A il faudrait une autre solution. On peut envisager les Convertisseurs Analogique-Numérique à Entrelacement Temporel (CANET) qui ont une vitesse d'échantillonnage qui va de la centaine de mégahertz à quelques gigahertz. Cependant on remarque que les CANETs ont une résolution qui est plus faible que celle des CAN autonomes. Ceci est du au fait que l'entrelacement introduit d'autres erreurs qui n'étaient pas présentes dans les CAN autonomes. Ces erreurs proviennent des disparités entre les différents CAN et sont communément appelés « mismatchs ». On distingue principalement les mismatchs de gain, time-skew, bande et offset qui doivent être corrigés à travers une calibration.

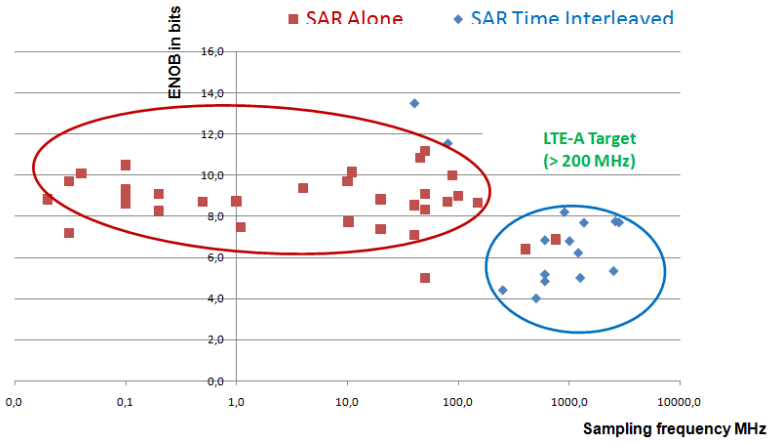


Figure 3: Stand alone ADCs performance survey from 1997 to 2015.

La Fig. 4 montre l'architecture d'un CANET qui est constitué de plusieurs CAN qui fonctionnent alternativement l'un après l'autre et l'ensemble se comporte comme un CAN unique mais qui a une fréquence d'échantillonnage M fois supérieure à celle d'un CAN de chaque voie. Ce type d'architecture est utilisé dans les applications comme les communications sans fil, l'instrumentation, l'aérospatial et les stations de base pour le militaire. Comme avantage nous avons la vitesse et comme inconvénient il y a l'apparition de nouvelles erreurs appelées mismatchs qu'il faut calibrer.

On peut classer les techniques de calibration en plusieurs types suivant le mode d'opération et l'emplacement de la correction. Il existe les calibrations

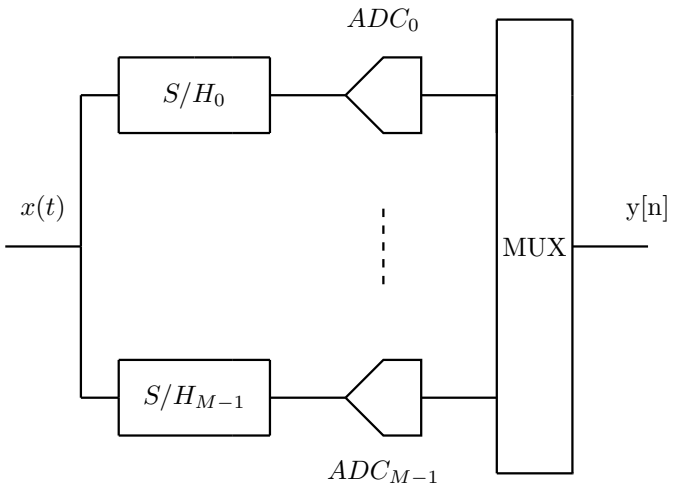


Figure 4: Time Interleaved ADCs architecture.

différées dans lesquelles un signal de test connu est injecté à l'entrée du convertisseur puis le signal de sortie est mesuré puis comparé au signal de l'entrée. Ceci permet l'identification des mismatchs et la correction des erreurs. Ce type de méthodes est adapté pour les applications comme l'instrumentation et la mesure où on peut envisager une phase de diagnostique permettant de calibrer le convertisseur puis une phase de fonctionnement où le CAN marche normalement. Les méthodes de calibration différées ne sont pas adaptées pour les applications temps réel comme les communications où le convertisseur doit toujours être en cours de fonctionnement. La solution alternative est d'utiliser les méthodes en ligne où la calibration opère en toile de fond tandis que le CAN fonctionne. Les méthodes de calibration en ligne sont basées sur des techniques aveugles pour estimer les mismatchs. En réalité l'estimation n'est jamais totalement aveugle car on connait tout de même quelques informations statistiques sur le signal comme sa densité spectrale de puissance et sa puissance totale. Si la correction utilise une rebouclage analogique sur le front-end du CAN on qualifie la calibration de mixte. Les méthodes mixtes permettent d'obtenir de très forts niveau de correction mais elles augmentent le temps de conception du circuit car il faut modifier le circuit électronique du convertisseur. L'alternative est d'utiliser les méthodes totalement numériques dans lesquelles la correction est effectuée à l'aide d'un circuit numérique. Dans cette thèse nous intéresserons aux calibrations en ligne et totalement numériques.

Le gain, le time-skew et l'offset ont été largement explorés mais le mismatch de bande n'a été analysé que tout récemment. De plus La plupart des techniques de calibration du mismatch de bande ne prennent pas en compte les mismatch de gain et de time-skew. En effet les erreurs dues au mismatch de bande peuvent se combiner de façon constructive ou destructive avec les erreurs dues aux mismatch de gain et de time-skew comme nous allons le démontrer dans cette thèse. Donc par souci d'optimalité il faudrait traiter ces trois mismatchs en même temps.

Cette thèse a deux volets. Le premier c'est un volet modélisation où nous proposons des modèles qui permettent d'analyser les erreurs dans les CAN autonomes. Nous aurons une attention spéciale sur la modélisation de la nonlinéarité dans les Echantillonneur/Bloqueur (E/B) bootstrappés, puis suivrons les modèles de bruits dans les CAN et les modèles d'erreurs dans les structures entrelacées. Le second volet consiste en méthode de calibration des mismatchs de gain, time-skew et bande dans les CANETs. L'estimation se fait en aveugle, de façon adaptative et est basée sur un filtrage passe-bas combiné avec un filtre à retard fractionnaire. La compensation des erreurs est basée sur le développement d'une approche matricielle pour réduire l'effet des mismatchs.

0.2 Modèles d'erreurs dans les CANs

La principale erreur analysée dans ce chapitre est la nonlinéarité de l'E/B. Un E/B doit vérifier un certain nombre de contraintes parmi lesquelles les contraintes de consommation en puissance, de vitesse, de nonlinéarité et de dynamique de signal d'entrée. Parmi ces contraintes là, la contrainte de nonlinéarité est difficile à atteindre car de par sa structure de base l'E/B est non linéaire. En effet la résistance de transistor-commutateur est donnée par la relation (1) qui dépend de la tension grille-source qui elle-même dépend du signal d'entrée. Cette dépendance de la tension grille-source avec le signal d'entrée crée des nonlinéarités dans le circuit qui distordent le signal.

$$R_{on} = \frac{1}{\mu C_{ox} \frac{W}{L} (V_{gs} - V_{th} - \frac{V_{ds}}{2})} \tag{1}$$

La technique la plus populaire pour corriger cette nonlinéarité est de modifier le circuit de l'E/B et de rajouter un circuit dit « Bootstrap » comme le montre la Fig. 6. Des simulations en 65nm CMOS montrent que le montage du bootstrap n'est pas suffisant car la résistance de transistor-commutateur continue de varier avec le signal d'entrée. Cependant on peut noter que la variation devient linéaire avec le signal d'entrée comme le montre la Fig. 5. Pour une variation linéaire de la résistance avec le signal d'entrée et en considérant une capacité de maintien idéale on démontre que le signal de sortie se compose d'un offset à la fréquence 0, du signal fondamental et d'une harmonique de second ordre. L'harmonique 2 étant l'harmonique dominante, elle peut être supprimée grâce à une structure différentielle. Cependant les mismatchs entre les deux voies de la structure différentielle limitent la correction et il est possible d'exprimer l'harmonique 2 résiduelle en fonction du niveau de mismatch.

Les mismatchs étant aléatoires, nous faisons également une étude statistique de cette distorsion. Cette étude nous permet d'obtenir la densité de probabilité de la distorsion et de trouver de combien il faut réduire les mismatchs pour obtenir une distorsion seuil avec un niveau de confiance spécifique. Pour avoir une distorsion de l'ordre de 100 dB avec un niveau de confiance de 99.9 % il

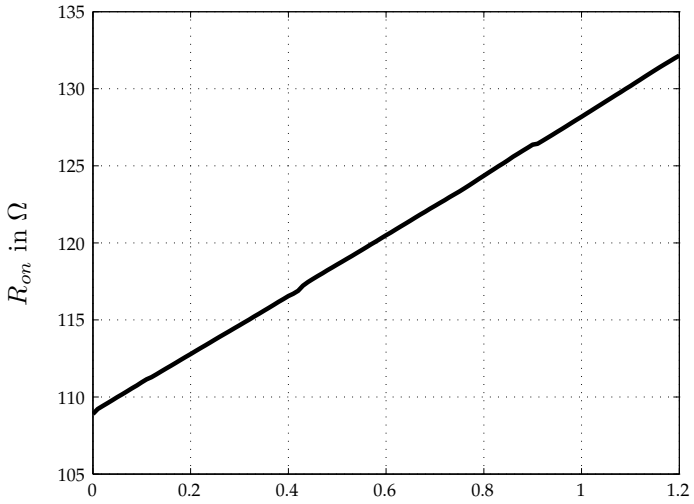


Figure 5: Simulation results of the on-resistance of a bootstrapped S/H as a function of the input signal in 65 nm CMOS process with supply voltage of $v_{dd} = 1.2\,\mathrm{V}$.

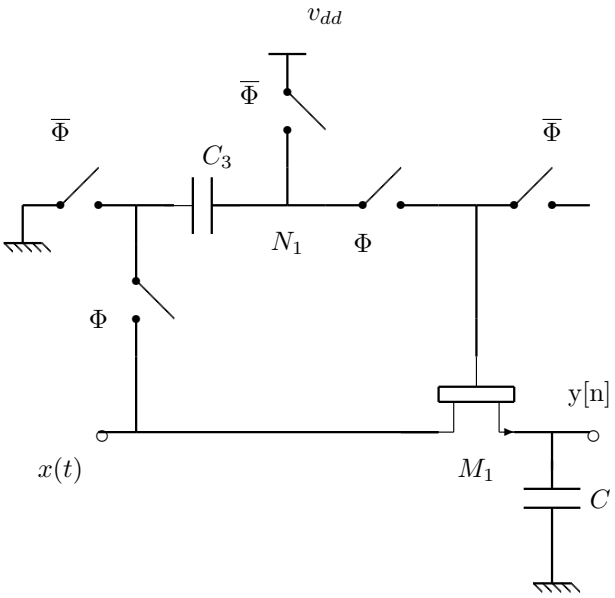


Figure 6: Logical structure of the bootstrap circuit.

faudrait que les mismatch soient plus petits que 1 %. Ceci fournit au concepteur analogique des outils pour dimensionner son circuit et atteindre spécifications technique désirées.

Il est à noter que les imperfections comme la nonlinéarité sont spécifiques à chaque CAN et doivent être corrigées. En les corrigeant on obtient un CAN linéaire dont le modèle mathématique est donné par les formules (2) et (3) où on distingue une composante de filtrage, une composante d'offet et une composante de bruit.

$$y[n] = y_{AC}[n] + y_{DC}[n] + v_{noise}[n] = \Big(h(\mathbf{p}, .) \star x\Big)(nT_s) + O + v_{noise}[n] \quad (2)$$

$$\begin{aligned} Y(f) &= Y_{AC}(f) + Y_{DC}(f) \\ &= f_s \sum_{k=-\infty}^{+\infty} \Big\{H(\mathbf{p}, .)X(.) + O\delta(.)\Big\}\Big|_{f-kf_s} \end{aligned} \quad (3)$$

Ce modèle servira de base pour l'analyse des CANETs.

0.3 Modèles de bruit dans les CANs

Dans ce chapitre nous étudions les bruits dans les CANs en termes de densité de probabilité et de densité spectrale de puissance. Nous intéressons de façon sommaire au bruit thermique et au bruit de scintillement, puis de façon détaillée au bruit thermique et au bruit créé par la gigue. Le bruit thermique est gaussien et sa densité spectrale de puissance se blanchit au fur et à mesure que le temps d'acquisition devient grand devant la constante de temps de l'E/B. La gigue est la variation aléatoire des instants d'échantillonnage et elle constitue une des erreurs les plus importantes dans les CAN. Elle comprend la gigue à l'ouverture et la gigue d'horloge. La gigue d'horloge est causée par le bruit de phase de l'horloge qui peut être un oscillateur libre ou une boucle à verrouillage de phase. Le bruit créé par la gigue d'horloge dans le cas d'un oscillateur libre suit un processus de Wiener et a une densité spectrale de puissance en forme d'une Lorentzienne. La gigue à l'ouverture est causée par la distribution du signal d'horloge (buffers). Il s'agit d'un processus banc gaussien et sa densité spectrale de puissance est quasiment blanche.

0.4 Modèles d'erreurs dans les CANETs

Dans ce chapitre nous étudions en détail les CANETs. Le signal à la sortie du système est donné par les relations (4) et (5). On peut noter une partie idéale qui consiste en des repliements aux multiples de la fréquence d'échantillonnage du signal ce qui est normal car l'échantillonnage se traduit dans le domaine spectral par la périodisation du spectre du signal. Cependant on note aussi l'apparition d'une composante perturbatrice aux diviseurs de la fréquence d'échantillonnage. Cette composante provient des mismatchs entre les différentes voies.

$$\frac{Y_{DC}(f)}{f_s} = \sum_{k=-\infty}^{+\infty} \overbrace{\Big[1 + \frac{1}{M}\sum_{m=1}^{M-1} \delta O_m(.)\Big] O_0 \delta(f - k f_s)}^{Regular\ part} + \sum_{\substack{k\neq 0[M] \\ -\infty}}^{+\infty} \frac{1}{M} \underbrace{\left[\sum_{m=1}^{M-1} \zeta^{-mk} \delta_m O\right] O_0 \delta(f - k\frac{f_s}{M})}_{Spurious\ part} \quad (4)$$

$$\frac{Y_{AC}(f)}{f_s} = \sum_{\substack{k\neq 0[M] \\ -\infty}}^{+\infty} \frac{1}{M} \underbrace{\left[\sum_{m=1}^{M-1} \zeta^{-mk} \delta H_m(.)\right] H_0(.)X(.)}_{Spurious\ part}\Big|_{f-k\frac{f_s}{M}} + \sum_{k=-\infty}^{+\infty} \overbrace{\Big[1 + \frac{1}{M}\sum_{m=1}^{M-1} \delta H_m(.)\Big] H_0(.)X(.)}^{Regular\ part}\Big|_{f-kf_s} \quad (5)$$

Nous pouvons également exprimer le mismatch de fonction de transfert comme combinaison du mismatch de gain, time-skew et bande à travers la relation (6). Cette relation montre que le mismatch de gain agit sur la partie réelle de la fonction de transfert, le mismatch de time-skew agit sur la partie imaginaire de la fonction de transfert, donc ces deux mismatch sont orthogonaux l'un l'autre. Par contre le mismatch de fonction de transfert agit à la fois sur la partie réelle et sur la partie imaginaire. Par conséquent le mismatch de bande peut se combiner de façon constructive ou destructive avec les mismatch de gain et de time-skew. C'est pourquoi dans cette thèse nous traiterons ces trois mismatch simultanément.

$$\delta H_m(f) = \delta g_m Q_0(f) + \delta t_m Q_1(f) + \delta f_{c_m} Q_2(f) \quad (6)$$

$$\text{where} \begin{cases} Q_0(f) = 1 \\ Q_1(f) = j2\pi f T_s \\ Q_2(f) = \frac{j\frac{f}{f_c}}{1+j\frac{f}{f_c}} \end{cases} \quad (7)$$

Les mismatchs étant aléatoires, le SFDR sera aussi une variable aléatoire qu'on peut caractériser. C'est ainsi que nous pouvons calculer d'une part la densité de probabilité du SFDR pour une profondeur d'entrelacement paire et impaire qui est donnée respectivement par les formules (9) et (8). Les figures (7), (8) et (9) montrent la distribution statistique du SFDR pour des mismatchs de gain, time-skew et bande. On peut voir que la distribution statistique suit bien la densité de probabilité analytique calculée.

D'autre part nous pouvons calculer de combien il faut réduire l'amplitude des mismatch pour obtenir le SFDR désiré avec un niveau de confiance spécifique. La forumule (10) nous dit qu'avec un niveau de mismatch ayant un ecart type de

sigma σ_a, on peut obtenir un $SFDR(\eta)$ avec un niveau de confiance de η. Ceci permet au concepteur analogique de mesurer l'effort de calibration à fournir pour atteindre les performances désirées.

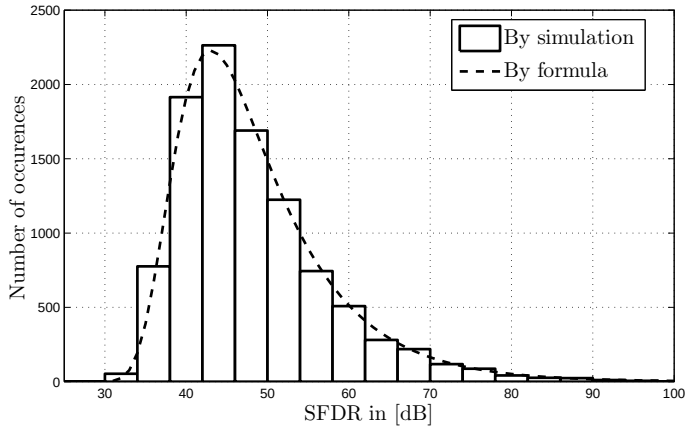


Figure 7: PDF of a two-channel TI-ADCs with a gain mismatch of 1 % and with a input sinusoid of amplitude 1 V

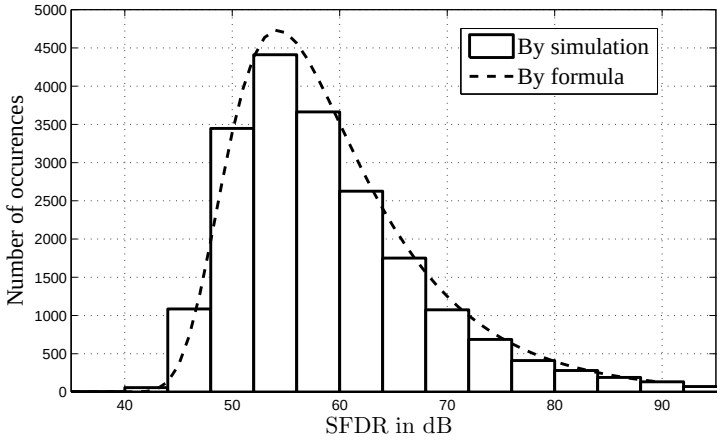


Figure 8: PDF of a two-channel TI-ADCs with a time-skew mismatch of 1 %, a sampling frequency of 320 MHz and with a input sinusoid of amplitude 1 V and a frequency of 137 MHz

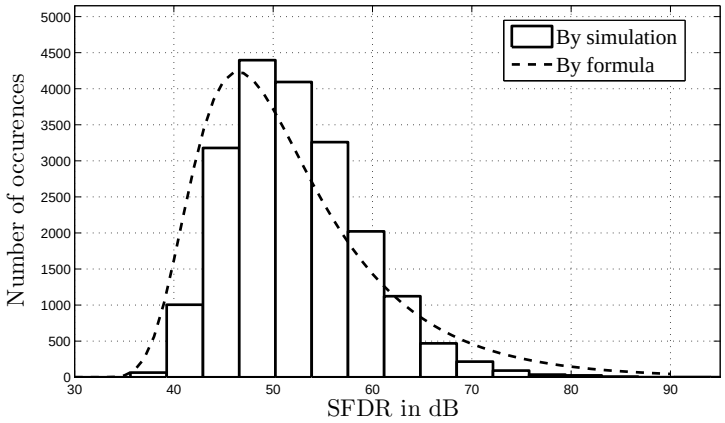


Figure 9: PDF of a two-channel TI-ADCs with a bandwidth mismatch of 1 %, a sampling frequency of 320 MHz and with a input sinusoid of amplitude 1 V, a frequency of 137 MHz and a cutoff frequency of 160 MHz

$$\begin{aligned} p(SFDR) &= \frac{1}{F'\Big[F^{-1}(SFDR)\Big]} f_{max}\Big(F^{-1}(SFDR)\Big) \\ &= \alpha \frac{M-1}{2} s e^{-s}(1-e^{-s})^{\frac{M-3}{2}}\Big|_{s=\frac{M}{\sigma_a^2 C_a}10^{-\frac{SFDR}{10}}} \end{aligned} \tag{8}$$

$$p(SFDR) = \alpha s(1-e^{-s})^{\frac{M-4}{2}}\Big[\frac{M-2}{2}e^{-s}erf(\frac{s}{2})+\frac{1}{\sqrt{2\pi s}}e^{-s}(1-e^{-\frac{s}{2}})\Big]\Big|_{s=\frac{M}{\sigma_a^2 C_a}10^{-\frac{SFDR}{10}}} \tag{9}$$

$$SFDR_{dB}(\eta) = -10\log_{10}\frac{C_a\sigma_a^2}{M} - 10\log_{10}K(\eta, M) \tag{10}$$

0.5 Calibration des CANETs

La calibration des CANETs que nous proposons se décompose en deux parties indépedantes: l'estimation et la correction. L'estimation utilise un filtrage passe-bas combiné avec un filtre à retard fractionnaire pour estimer simultanément les mismatchs de gain, time-skew et bande à travers l'expression du mismatch de fonction de transfert en fonction du mismatch de gain, time-skew et bande comme illustré sur la Fig 10. Cette estimation est aveugle c'est-à-dire que le signal d'entrée est inconnu. On suppose juste qu'il est stationnaire et à bande limitée. La Fig. 11 montre le temps de convergence pour les mismatchs de gain, time-skew et bande. On peut voir qu'avec 10-K échantillons on peut avoir les estimées des trois mismatchs.

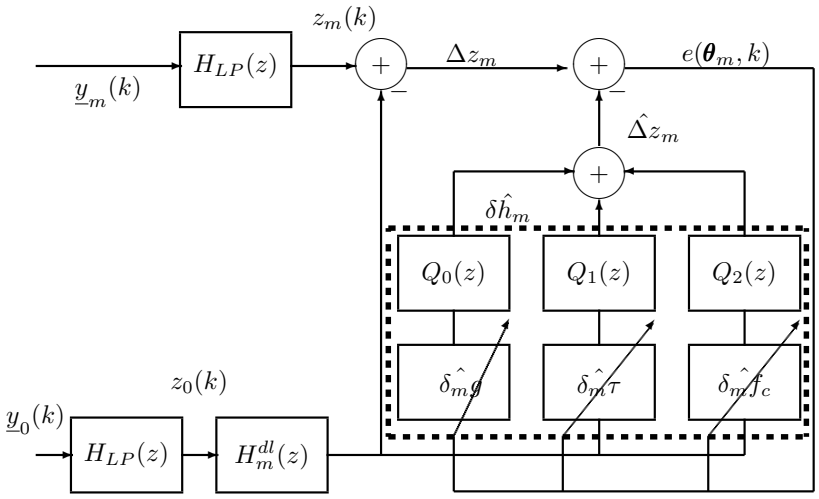


Figure 10: Adaptive filtering structure

La compensation utilise une représentation vectorielle des signaux de sortie pour identifier la matrice responsable de la distorsion, puis cette matrice là est inversée pour corriger les erreurs. La Fig. 12 montre la structure de compensation avec deux voies. On peut noter que par simulation nous avons des corrections de l'ordre de 35 dB comme le montre la Fig. 12 où est represent le SFDR avant at après calibration pour une fréquence d'échantillonnage de 340 MHz avec des mismatchs de gain de 1%, time-skew de -1% et bande de 2%. Cependant au fur et à mesure qu'on se rapproche de la fréquence de Nyquist les performances s'évanouissent. Ceci est du au fait que la bande du signal s'agrandit, ce qui diminue la portion de bande sur laquelle porte l'estimation d'une part et d'autre part la correction devient de moins en moins précise car le mismatch de fonction de transfert n'est plus négligeable devant un (le mismatch de fonction de transfer est un filtre passe haut).

Des mesures ont été faites sur une carte FPGA Zynq SoC sur laquelle est connecté un émetteur/récepteur contenant deux CAN que nous avons entrelacés. Le banc de mesure est illustré la Fig. 13. La Fig. 15 montre le spectre du signal

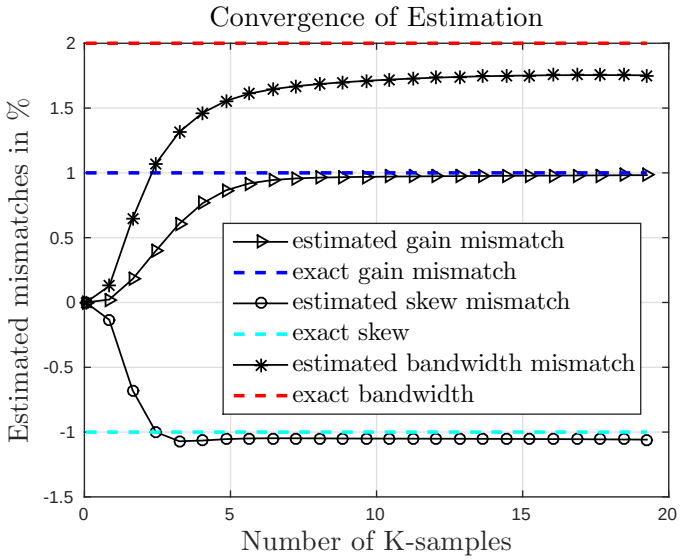


Figure 11: Simulation of the Convergence of mismatch estimation

de sortie avant et après calibration. On peut voir que le spur du aux mismatchs a été réduit de 38 dB. La Fig. 14 montre le niveau de correction en fonction de la taille des filtres utilisés. On constate que la correction augmente au fur et mesure que l'ordre des filtres augmente. Mais à partir d'un ordre de 51 la correction ne s'améliore plus et ceci correspond à une compensation de 38 dB. Ce chapitre se termine par une synthèse sur un circuit numérique. La puissance consommée est d'environ 10 mW et la surface de 0.035 mm^2.

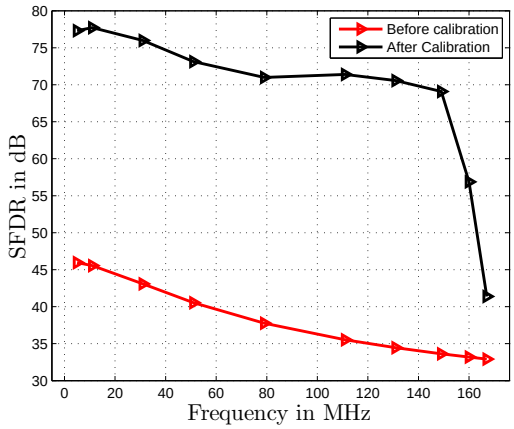


Figure 12: SFDR before and after correction with a two-channel TI-ADCs with 1% gain mismatch, -1% time-skew mismatch and 2% bandwidth mismatch. The sampling frequency is 340 MHz.

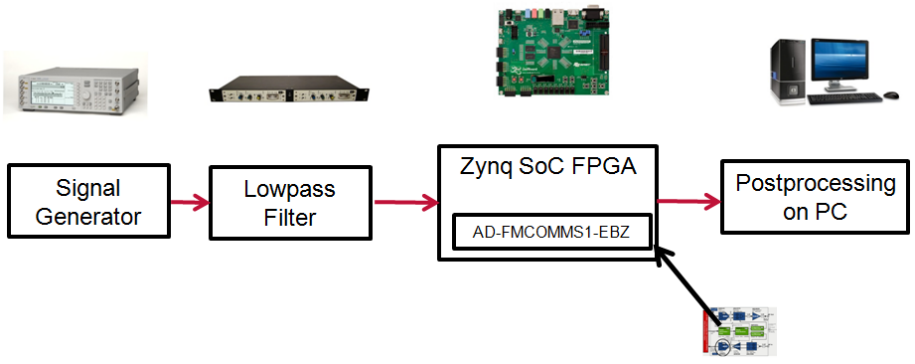


Figure 13: Test bench used for the measurements

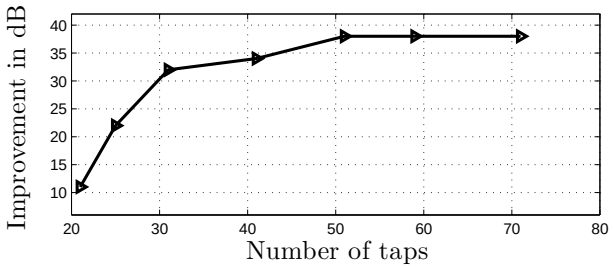
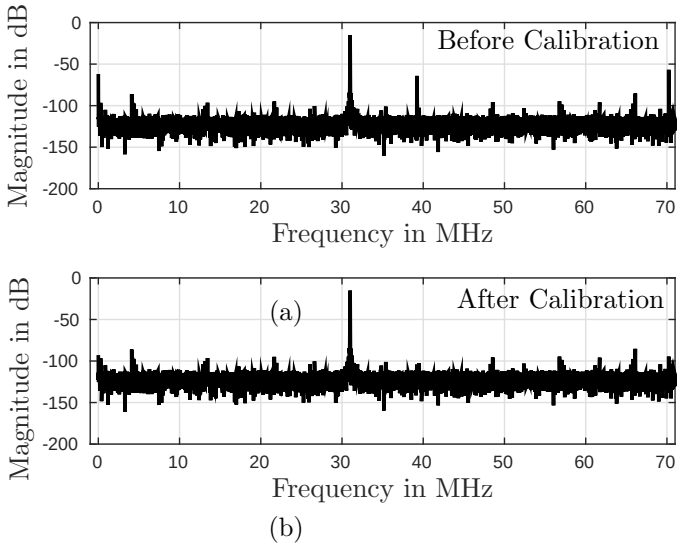


Figure 14: Reduction of spurious magnitude with the number of taps of correction filters



(b)

Figure 15: Measurement results of the output spectrum before and after calibration

0.6 Conclusion

Dans ce travail, nous avons proposé une modèle déterministe et statistique pour analyser la nonlinéarité dans les E/B bootstrappés pour les signaux single-ended et pour les signaux différentielles. Ceci nous a permis de montrer que dans les structures bootstrap l'harmonique 2 est dominante et peut être réduite par un montage différentiel. Dans les structures différentielles, la densité de probabilité de la distorsion de second ordre a été calculée et nous avons également relié le niveau de distorsion en fonction du mismatch et du niveau de confiance. Ce chapitre se conclut avec le modèle mathématique du CAN qui sera utilisé pour l'entrelacement.

Ensuite dans le second chapitre, nous avons modélisé les bruits dans les CAN en termes de densité de probabilité et densité spectrale de puissance. Une attention particulière a été portée sur le bruit thermique et la gigue.

Puis dans le troisième chapitre, nous avons modélisé les erreurs dans les structures entrelacées. Le signal de sortie a été représenté comme une composante idéale et une composante perturbatrice crée par les mismatchs entre les différentes voies. Une analyse statistique nous a également permis d'exprimer le SFDR désiré en fonction du niveau de mismatch et du niveau de confiance.

Le dernier chapitre présente la méthode de calibration que nous proposons pour calibrer le convertisseur. Elle se base sur une estimation aveugle des mismatchs à travers un filtrage passe-bas et une filtre à délai fractionnaire. La compensation utilise une représentation matricielle du CANET pour corriger les erreurs. Les résultats montrent qu'on peut obtenir des corrections entre 30 et 40 dB et que l'estimation converge avec moins de 10-K échantillons.

Comme perspectives, on peut envisager la prise en compte de la nonlinéarité de l'E/B dans l'entrelacement et utiliser la méthode établie dans ce travail pour analyser d'autres erreurs comme l'injection de charge et l'excursion d'horloge.

Chapter 1

Introduction

1.1 Background and Motivation

The continuing demand for ever-higher wireless data rate system has resulted in higher bandwidth standards such as LTE-Advanced or IEEE 802.11.ac requiring for instance bandwidth up to 160 MHz. This increases the requirements on Analogue-to-Digital Converters (ADCs) because in radio receivers, the signal should be sampled at a rate equal at least to the bandwidth.

Fig. 1.1 shows a conventional radio receiver (superheterodyne). The complexity of this architecture grows linearly with the number of standards and channels. This is a strong limitation because mobile devices should integrate multiple bands and cohabitation of old standards such as GSM with new ones such as LTE-Advanced is desirable in order to enable users to connect worldwide. This has resulted in the definition of Software Defined Radio (SDR) concept. The idea behind SDR is that the same hardware architecture can be reconfigured to handle any radio standard. It is achieved by replacing conventional analog signal processing such as channel selection or filtering in conventional radio receiver by digital signal processing. This is done by pushing the ADC close to the antenna as shown on Fig. 1.2.

1.2 High Speed ADCs Design Challenges

Replacing analog parts such as mixers with digital ones requires faster and more accurate converters. However, requirements on high speed is often contradictory with high accuracy and low power consumption. To illustrate it, Fig. 1.3 shows a recent survey across popular stand alone ADC architectures (Pipeline, Flash, Sigma-Delta and Successive Approximation Register) from 1999 to 2014 [1]. It can be noticed that when the resolution increases, the speed decreases.

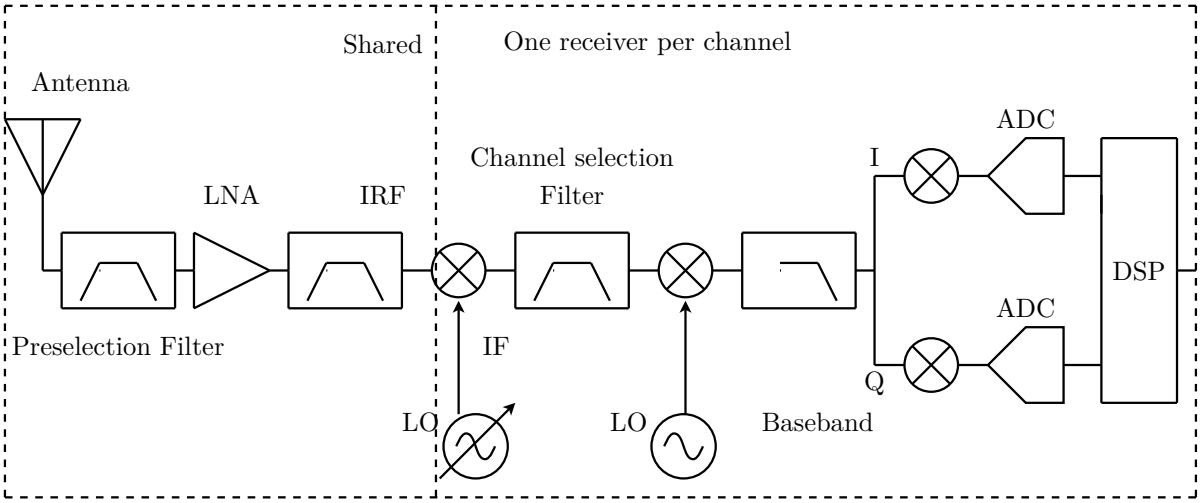


Figure 1.1: Conventional superheterodyne architecture.

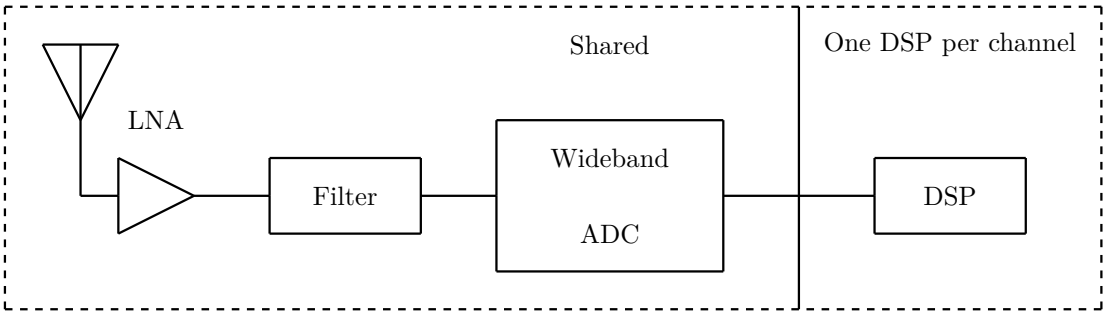


Figure 1.2: The ideal software defined radio architecture.

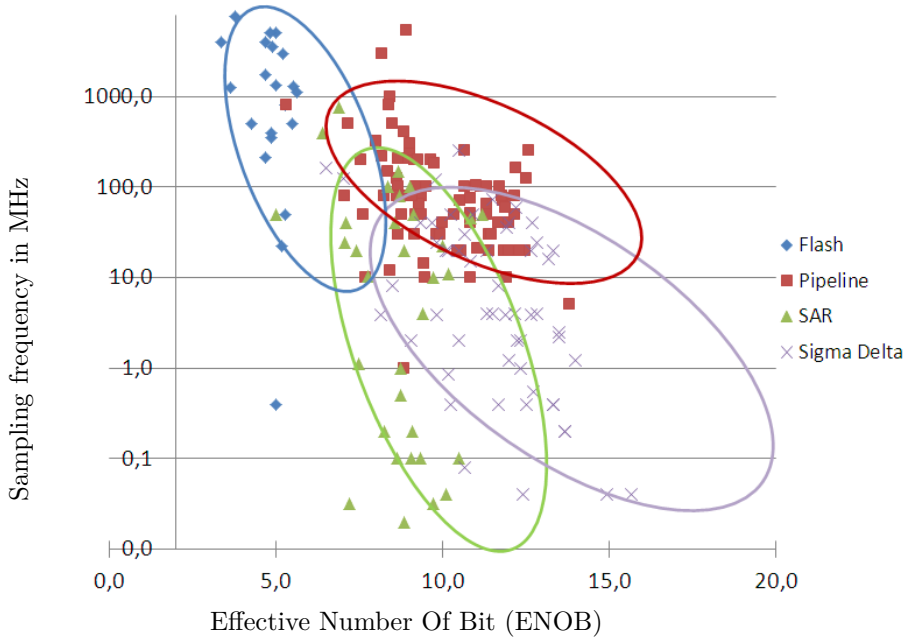


Figure 1.3: Stand alone ADCs performance survey from 1999 to 2014.

1.3 Time-Interleaved ADCs

A possible way to achieve a high product speed-resolution with a reasonable power consumption is by designing Time-Interleaved ADCs (TI-ADCs) [2]. The first step consists in building high resolution sub-ADCs at relatively low speed. For mobile handset applications, the power-efficiency is a critical issue. Therefore Pipeline and Successive Approximation Register (SAR) architectures emerge as candidates for this purpose since they can achieve simultaneously high resolution with a low power. Then the second step is to interleave these sub-ADCs, so that they work alternatively as if they were effectively a single high resolution ADC but working at a much higher rate. This is shown on Fig. 1.4.

1.4 Related work on TI-ADCs

Time-Interleaved (TI) architectures have emerged as a good way to provide high speed data converters with relatively slow circuits. Unfortunately in this kind of architecture, new errors emerge and give rise ton nonlinear distortion which significantly degrade the resolution of the overall TI-ADC. These errors come from discrepancies between the individual sub-ADCs in the system and are commonly referred to as channel mismatch errors. They consist of gain, time-skew, bandwidth and offset mismatch errors and they should be mitigated [2] [3].

There are two possible ways to deal with channel mismatches. The first is to complexify the analog circuit design of the ADC in order to reduce the magnitude of the original mismatches at the cost of more power consumption and area. But this increases the time-to-market of the ADC. The second solution is to alleviate the design and to correct the errors with a calibration technique. Depending of the nature of the input signal, calibration techniques can be classified into two different categories.

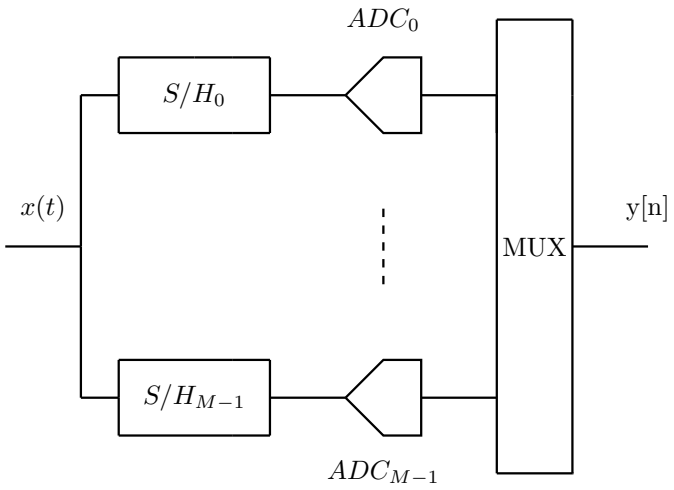


Figure 1.4: Time Interleaved ADCs architecture.

Foreground calibration techniques estimate the channel mismatches by interrupting the normal TI-ADCs operation and applying a known signal like a sinewave at its input [4]. Then the output of the TI-ADCs is compared to the expected output that would have been obtained with no mismatches. In this way, the effect of each mismatch can be measured and corrected. The drawback of foreground calibration techniques is that the ADC has to be taken offline every time that the calibration is carried out. In addition the mismatches may change due to temperature variations and aging. As a consequence for applications such as mobile communication systems, foreground calibration techniques are not suitable because the ADC has to be always on. However they can easily be used in applications such as instrumentation where the device can be calibrated off before being used.

In background calibration techniques, the ADC operation continues when the calibration is being performed. The mismatches are continuously estimated and corrected. These techniques can be subdivided into semiblind and blind. Semiblind background calibration techniques combine the input signal with a test signal that will be used for the calibration [5]. In blind background calibration techniques, no test signal is used [6]. Blind background calibration techniques are the most difficult to design because they should track and adjust to the changing operation conditions of ADCs in demanding environments with rapidly changing temperatures. In addition, they should work with no informations or with little a priori informations on the input signal such as statistics.

When the calibration uses a feedback to the analog front-end of the ADC, it is a mixed signal calibration [7][8]. When it is done entirely in the digital domain, the calibration is said to be fully digital [9][10][5]. Mixed signal calibration techniques are popular in current TI-ADCs chips but fully digital calibrations are more and more desired because they require no custom redesign of the analog

front-end of the ADC [8].

Several works have been done on correcting the gain, time-skew and offset mismatches in TI-ADCs [11] [12] [10] [7] [13] [6] [14] [15] [16] , but little work has been done on bandwidth mismatches. In [5] [17] [18], some bandwidth mismatch calibrations are proposed for two channels ADCs but they dont take into account the time-skew and gain mismatch. Indeed bandwidth mismatch is frequency dependent and is likely to combine constructively or destructively with time-skew and gain mismatch as we will demonstrate in this thesis. Therefore these three mismatches should be treated jointly for more optimality. In [19], a calibration method for gain, time-skew and bandwidth mismatches using a feedforward equalizer is proposed, but the algorithm takes a long time to converge. In [20], a calibration of frequency response mismatch is proposed by modeling transfer function as polynomial with variable order differentiators and coefficients. This was done for only two channels and the decomposition in differentiator filters is more accurate for time-skew mismatch correction than for bandwidth mismatch. In [21], a fully digital frequency response mismatch compensation algorithm for TI-ADCs is proposed using correlations between TIADC samples at two different frequency shifted images for mismatch estimation. In addition most of these calibration techniques were only tested with behavioral simulations, they have not been synthesized on chips in order to measure power and area consumption.

1.5 Goal, Contribution and Thesis Organization

As pointed out in section 1.4, most of the calibration methods of the state of art are either suboptimal since they don't handle all the mismatches together, limited to a number of channels, have a low convergence speed, or have not been implemented on chips to evaluate the consumed resources. The main objective of this work is to propose a new calibration technique that overcomes these limitations. We propose a fully digital blind background calibration of gain, time-skew, bandwidth and offset mismatches in TI-ADCs. The contributions of this work are:

- An original time and frequency domain representation which defines the transfer function mismatch and models it as a combination of the gain, time-skew and bandwidth mismatch and formulates the problem of calibration.
- The demonstration of a relationship between the gain, time-skew and bandwidth mismatch errors which highlights the necessity of a joint calibration of these three mismatches for more optimality.
- An adaptive and simultaneously blind estimation of the gain, time-skew and bandwidth mismatches. The mismatches can be estimated with an accuracy of respectively 98%, 94% and 88% for gain, time-skew and bandwidth mismatches. The algorithm converges with less than 10K-samples which is faster compared to the state of art to our knowledge.
- A joint compensation of the gain, time-skew, bandwidth and offset mismatch errors simultaneously. Our technique was tested on a two-channels

ADCs board from Analog Devices and measurement results show that the linearity can improved by almost 40 dB.

- A calibration scheme that can be applied to any interleaved factor with a high flexibility, thus reducing the time-to-market of TI-ADCs.
- A statistical characterization of noise in ADCs in terms of Probability Density Function (PDF) and Power Spectral Density (PSD).
- A deterministic and a statistical model in time and in frequency domain of bootstrapped S/H both for single ended and differential architectures. For differential bootstrapped S/H the mismatches between the p-channel and the n-channel should be less than 1% to obtain substantial second harmonic distorsion in the order of 100 dB.

This thesis is organized as follow. Chapter 2 gives an analysis of non-idealities of a single ADC and the analog variables responsible for this non-idealities are precisely identified. Then a mathematical model of the output of a single ADC in function of its gain, time-skew, bandwidth and offset is proposed. Chapter 3 complements this model of a single ADC with a statistical description of noises in ADCs in terms of Probability Density Function (PDF) and Power Spectral Density (PSD). These noises include quantization, thermal, jitter and flicker noises. Chapter 4 proposes a general framework describing simultaneously all mismatches together in TI-ADCs. Statistical laws are analytically derived. They convert SFDR and THD into matching requirement and therefore provide key rules for TI-ADCs designers. In chapter 5, we present the digital calibration we propose for channel mismatches correction. Numerical simulations and measurements are carried out to verify the correctness of the algorithm. Performances are also presented for ASIC synthesis in term of power and area consumption. Conclusion and perspectives of this work are in chapter 6

The following publications were done during the course of this work:

[7] **G. Kamdem De Teyou**, Hervé Petit and Patrick Loumeau. "Calibration of all mismatches in Time-Interleaved ADCs", *IEEE Transactions on Circuits and Systems - II*, submitted 2015.

[6] **G. Kamdem De Teyou**, Hervé Petit and Patrick Loumeau. "Adaptive and Joint Blind Calibration of Gain, Time-skew and Bandwidth Mismatch Errors in Time-Interleaved ADCs", *IEEE Electronic Letters*, accepted 2015.

[5] **G. Kamdem De Teyou**, Hervé Petit and Patrick Loumeau. "Adaptive and Digital Blind Calibration of Transfer Function Mismatch in a Time-Interleaved ADCs", *IEEE New Conference on Circuit and Systems*, 2015.

[4] **G. Kamdem De Teyou** et al. "Statistical Analysis of Noise in Broadband and High Resolution ADCs", *IEEE International Conference on Electronic Circuit and Systems*, 2014.

[3] **G. Kamdem De Teyou** et al. "Statistical Analysis of Harmonic Distorsion in Bootstrapped Sample and Hold Circuit", *IEEE PRIME Conference*,

2014.

[2] **G. Kamdem De Teyou** et al. "Mismatch Requirement Analysis in Bootstrapped S/H", *GdR System on Chip- System In Package Symposium*, 2014.

[1] S. Paquelet, **G. Kamdem De Teyou** and Y. Le Guillou "TI-ADCs SFDR Requirements Analysis", *IEEE New Conference on Circuit and Systems*, 2013.

Chapter 2

Analysis of Analogue to Digital Converters

Introduction

Analogue-to-Digital Converters (ADCs) are the fundamental interface between the physical world where signals are analog and digital processing circuits widely used because of their noise immunity, reconfigurability and flexibility. They sample continuous-time analogue signal and convert it into a discrete digital representation. The conversion process consists always into two steps: sampling which is discretization in time and quantization which is discretization in amplitude. ADCs have become a key element in almost all applications of electronics such as radar, radio receiver, instrumentation, audio etc...

This chapter gives an analysis of the internal behavior of a single ADC. First, common ADC performances specifications are given. Then S/H architectures are explored with their non-idealities. The principal limitations include time-skew, bandwidth limitation and the signal dependency of the on-resistance. A precise analysis of this last limitation is provided. The bootstrap technique is well known to reduce this signal dependency of this on-resistance but some nonlinearities remain due to parasitic capacitances, mobility degradation and back gate effect. This results in a second order harmonic spurious which can be reduced with a differential architecture. However mismatch between channels limits this technique. In this chapter, we provide a deterministic model in time and in frequency domain of bootstrapped S/H both for single ended and differential architectures. A statistical analysis of bootstrapped S/H for differential signal in function of mismatches is proposed and also the probability for the Harmonic Distortion (HD) to be lower than a critical value for any mismatch dispersion. Therefore, for a level of performance determined by a minimum HD and its probability of achievement we can specify the required mismatch dispersion. This practical information becomes of relevant importance to establish robust design with safe margins.

In section 2.4, the most popular quantization architectures are analysed regarding the speed, the accuracy, the area and the power consumption. The last section of this chapter proposes a mathematical model of the output of an

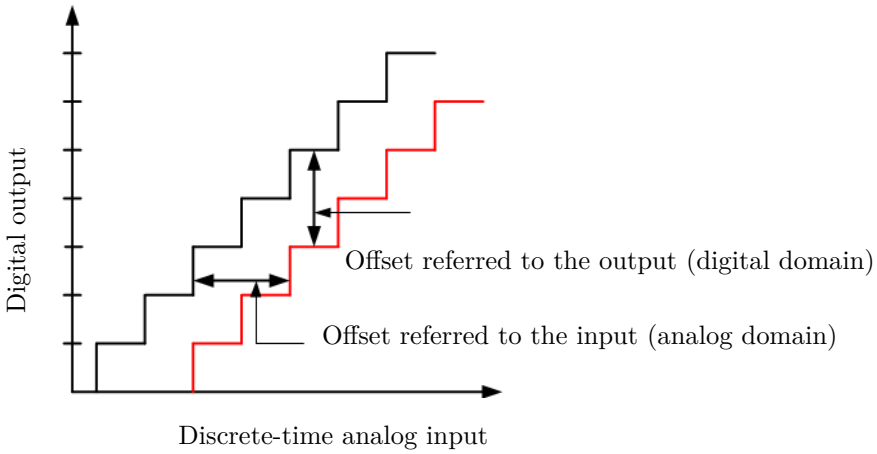


Figure 2.1: Static characteristic of an ADC with offset

ADC in function of its gain, time-skew, bandwidth and offset.

2.1 ADC Performance Specifications

ADC performance specifications quantify the errors that are caused by the ADC itself. ADC performance specifications are generally categorized in two ways: DC accuracy and dynamic performance.

2.1.1 DC Accuracy

The DC specifications for the converter tell how the device performs for steady-state analog inputs.

Offset error

Ideally, the output code for 0V is 0. But in practice for a real ADC, this is not the case. Offset error as illustrated on Fig. 2.1 is the constant shift in tension introduced by the ADC across its characteristic. Offset can be positive or negative and it is a common problem with ADC. Offset error can be easily compensated by calibration.

Gain error

The gain error of an ADC indicates how well the slope of an actual transfer function matches the slope of the ideal transfer function. Fig. 2.2 shows the static characteristic of an ADC with gain error.

Integral and Differential Non-Linearities error

Nonlinearities errors are local variations of code transition levels which can not be expressed linearly. They are defined after correcting for linear (offset and gain) errors. It consists of Integral Non-Linearity (INL) and Differential Non-Linearity (DNL).

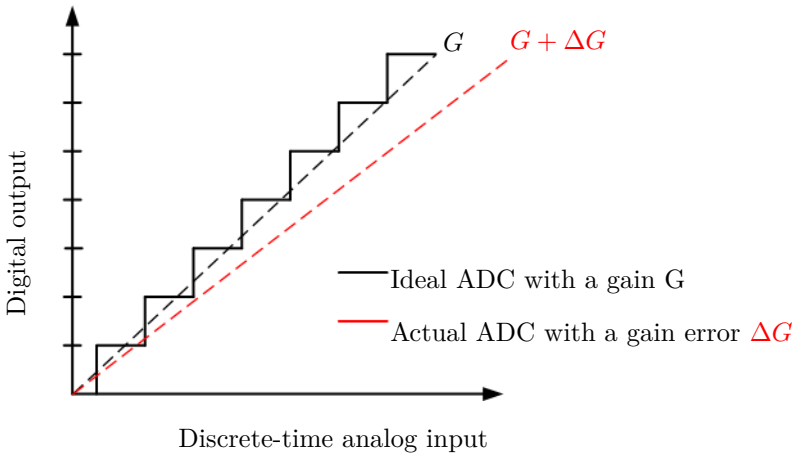


Figure 2.2: Static characteristic of an ADC with gain error

The INL is the distance between the actual decision level and the decision level of an ideal ADC that has been gain and offset corrected expressed in LSB units. The INL measures the deviation of the characteristic from a straight line.

$$INL_m = \frac{V_m - V_{o_m}}{q} \tag{2.1}$$

Where $V_{o_m} = (m - \frac{1}{2})q$ is ideal transition level of code m and $V_m = (m - \frac{1}{2})q + qINL_m$ is the real transition level of code m.

The DNL expresses the difference between the actual and the ideal code bin widths in LSB units. If DNL exceeds 1 LSB then there is the possibility to have a missing code at the output.

$$DNL_m = \frac{V_{m+1} - V_m}{q} - 1 = INL_{m+1} - INL_m \tag{2.2}$$

Figure 2.3 shows the characteristic of an ADC with non-linearities.

2.1.2 Dynamic Performances

Dynamic performances tell how much noise and distortion have been introduced into the sampled signal and the accuracy of the converter for a given input frequency and sampling rate.

Spurious Free Dynamic Range

The Spurious Free Dynamic Range (SFDR) is the ratio of the rms of the fundamental signal to the rms of the strongest spurious regardless of where it comes from in the spectrum .

$$SFDR = 20\log_{10}\left(\frac{S}{S_{max}}\right) \tag{2.3}$$

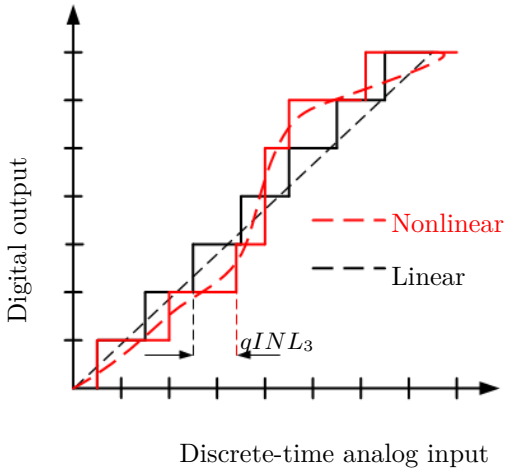


Figure 2.3: Static characteristic of an ADC with nonlinearities

Where S is the rms of the fundamental signal and S_{max} the rms of the highest spur which may or may not be an harmonic of the fundamental signal.

Total Harmonic Distortion

The Total Harmonic Distortion (THD) is the ratio of the rms of the fundamental signal to the root-sum-square of its harmonics :

$$THD = 20\log_{10}\left(\frac{S}{D}\right) \tag{2.4}$$

Where D is the root-sum-square of all harmonic components.

Signal to Noise Ratio

The Signal to Noise Ratio (SNR) is defined as the ratio of the power of the full scale fundamental signal by the total power of noise:

$$SNR = 20\log_{10}\left(\frac{S}{N_0}\right) \tag{2.5}$$

Where N_0 is the rms of noise. For an ideal ADC with a sine input, the noise consists only of quantization noise and the SNR is given by [22]:

$$SNR = 6.02N + 1.76 \tag{2.6}$$

Where N is the resolution of the ADC. In practice the measured SNR is inferior to this theoretical value and we should take into account the others sources of noise excluding harmonic distorsion.

Signal to Noise and Distorsion Ratio

Signal to Noise and Distorsion Ratio (SNDR) is defined as the ratio of the rms of the signal amplitude to the rms of all other spectral components including harmonics, but not DC [23]. SNDR is a good indicator for the overall performance of the ADC because it includes all components which make noise and distorsion.

$$SNDR = 20\log_{10}\left(\frac{S}{N_0 + D}\right) \tag{2.7}$$

Where D is the rms of harmonic distortions , N_0 is the total noise excluding DC component and harmonic distortions and S the rms of the fundamental signal. SNDR, THD and SNR are linked by the relation:

$$SNDR = -10\log_{10}\left(10^{-THD/10} + 10^{-SNR/10}\right) \tag{2.8}$$

Effective Number of Bit

If we solve (2.6) in N considering noise and all distorsion components , we obtain the Effective Number Of Bits (ENOB) defined as :

$$ENOB = \frac{SNDR - 1.76}{6.02} \tag{2.9}$$

Figure of Merit

A popular indicator used to compare ADC is the Figure Of Merit (FOM). It is given by :

$$FOM = \frac{P}{2^{ENOB} f_s} \ \ (pJ/step) \tag{2.10}$$

Where f_s is the sampling rate and P the power consumption. This parameter is commonly used in published report as it is based on measured quantities and calculates something that has a meaningful value (energy per conversion step).

2.2 Analysis of a Basic CMOS Sample and Hold

In ADCs, quantization is not instantaneous. The signal should be maintained to a constant value to process quantization. This is the function of the Sample and Hold (S/H) circuit which samples the voltage of the continuous-time signal and holds its value at a constant level for a minimum period.

The simplest S/H consists of two buffer amplifiers, a transistor-switch and a storage capacitor as illustrated in figure 2.4. During the sampling mode, the transistor-switch is ON and the input signal charges or discharges the hold capacitor so that the voltage $y(t)$ across the capacitor is practically proportional to the input voltage $x(t)$. This stage goes from $nT_s - \beta T_s$ to nT_s, with βT_s the acquisition time which is taken as a fraction of the sampling period. The circuit is governed by a first order linear Ordinary Differential Equation (ODE) with constant coefficients:

$$y(t) + R_{on}C\frac{dy(t)}{dt} = x(t) \text{ and } nT_s - \beta T_s \leq t \leq nT_s \tag{2.11}$$

With R_{on} the on-resistance of the transistor-switch. During the hold mode, the switch is OFF and the input signal is disconnected from the capacitor. The voltage across the capacitor is stored as the sampled value $y[n]$. The hold mode goes from nT_s to $(n+1)T_s - \beta T_s$:

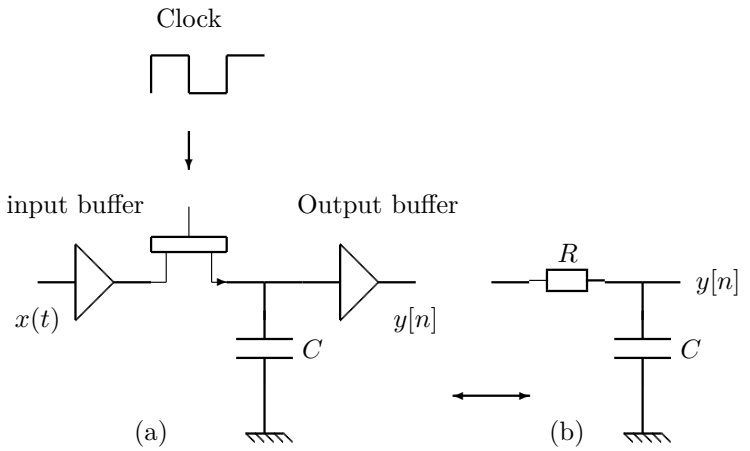


Figure 2.4: Open-loop S/H diagram (a) and its equivalent first order model (b).

$$y(t) = y[n] = y(nT_s) \text{ and } nT_s \leq t \leq (n+1)T_s - \beta T_s \tag{2.12}$$

There are several non-idealities in this circuit.

2.2.1 Time-skew

Time-skew t_0 is a short and constant delay between the moment when sampling has to be done and the moment when it is really done. Indeed on die, clock signal must be driven to S/H. For this purpose, buffers are inserted along clock path to regenerate and amplify clock signal in order to ensure satisfactory edge at arrival as shown on Fig. 2.5. A typical buffer consists in two inverters in cascade.

Time-skew comes from propagation delay of clock signal and transition time in inverters. Time-skew is in the order of some few nanosecond (ns). On a single ADC, time-skew produces no error but only acts as a fixed delay on sampling process. But in a Time-Interleaved ADCs (TI-ADCs), it can produce significant errors as we will see in chapter 4.

2.2.2 Bandwidth limitation

S/H of Fig. 2.4 can be reduced to an RC low-pass filter which has a time constant $\tau = R_{on}C$. To ensure accurate sampling, the sampling duration βT_s should be several time bigger than the time constant. A common metric is the number of time constants N_{on} as defined in [24]:

$$N_{on} = \frac{\beta T_s}{R_{on}C} \tag{2.13}$$

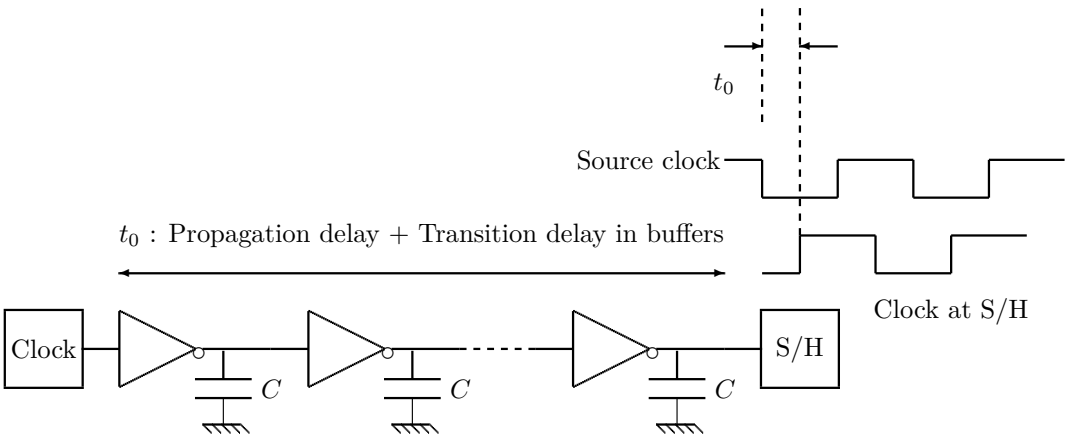


Figure 2.5: Clock distribution circuit of a single ADC.

Precision S/H is typically designed with $N_{on} \geq 7$ [24]. The maximum allowable sampling frequency in this context is obtained from (2.13) as:

$$f_s = \frac{\beta}{N_{on} R_{on} C} \tag{2.14}$$

2.2.3 Signal dependency of the on-resistance

In the sampling mode of the S/H of Fig. 2.4, the on-resistance of the transistor switch is approximately given by [25]:

$$R_{on} = \frac{1}{\mu C_{ox} \frac{W}{L} (V_{gs} - V_{th} - \frac{V_{ds}}{2})} \tag{2.15}$$

Where W, L, C_{ox}, V_{gs}, V_{th} and V_{gs} are respectively the width and length of the MOS transistor, the gate capacitance per unit area, the gate-to-source voltage, transistor threshold voltage and the drain-to-source voltage. $\frac{V_{ds}}{2}$ is most of the time negligible with respect to $V_{gs} - V_{th}$. The source S of the transistor is connected to the input signal $x(t)$. Therefore the gate-to-source voltage v_{gs} becomes dependent of the input signal. As a result, the whole resistance R_{on} is signal dependent and this will cause significant distortion in the sampled voltage on the hold capacitor.

The bootstrap circuit

Some techniques have been proposed on the analog side to mitigate this distortion. For example the use of a PMOS switch in parallel with an NMOS switch [26]. But the most popular is the clock bootstrapping [27] [28] [29] [30] which removes a significant portion of nonlinearities by making the value of the transistor-switch gate-source voltage as independent as possible of the input signal. The logical structure of the bootstrapped circuit is shown in Fig. 2.6.

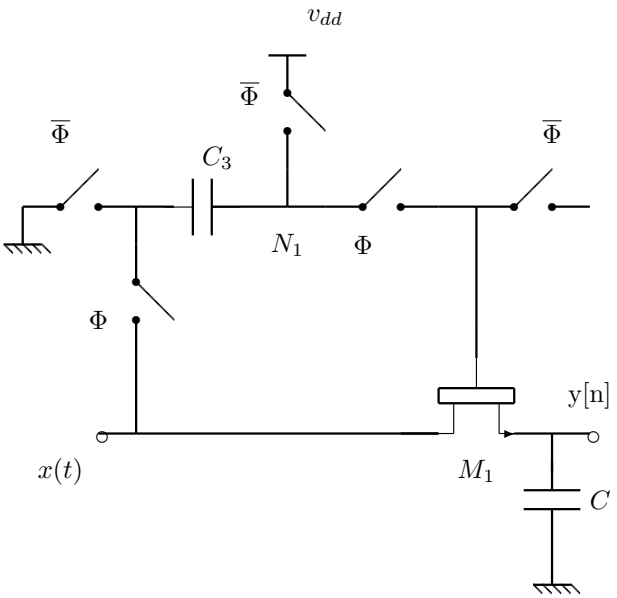


Figure 2.6: Logical structure of the bootstrap circuit.

When the sampling switch M_1 is OFF, $\Phi = 0$ and C_3 is precharged to v_{dd}. When $\Phi = 1$, a constant voltage equal to v_{dd} is established between the gate and the source of M_1. Ideally, R_{on} is now independent of the input signal. But in practice, parasitic capacitance at node N_1, mobility degradation and back gate effect limit the linearity that can be achieved. At a first order we can consider that R_{on} still depends of the input signal, but with a weakly linear dependency:

$$R_{on}(x) = b_{on} + a_{on}x \tag{2.16}$$

With a_{on} and b_{on} two constants which depend on the circuit. Simulation results confirm this assumption as we can see on Fig. 2.7 which shows the on-resistance of a bootstrapped circuit in function of the input signal in 65 nm CMOS technology. We see that the on-resistance varies quasi linearly with the input signal.

Therefore, without loss of generality we consider that the time constant τ varies linearly with the input signal as written in (2.17). b is the static component of the time constant and a is the dynamic component of the time constant. Montecarlo simulation of a bootstrapped circuit in 65 nm CMOS process shows that b and a can be modeled as random variables normally distributed as we can see on Fig. 2.8 and Fig. 2.9.

Simulation results show that a and b are highly correlated as shown on Fig. 2.10 where a is plotted in function of b. A correlation coefficient $\rho(a, b) = 0.9$ was obtained between a and b. As a consequence, we can consider that a can be expressed linearly in function of b as in (2.17). In others words, this means that a can be determined knowing b. In Appendix . A, this correlation is explained through some analytical calculation.

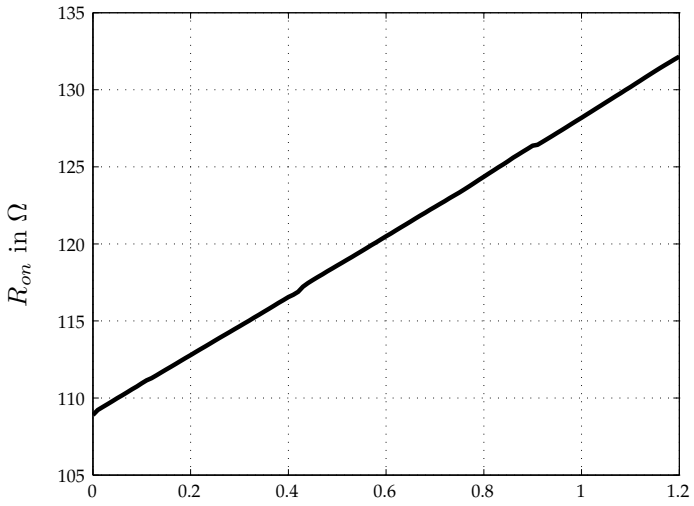


Figure 2.7: Simulations result of the on-resistance of a bootstrapped S/H as a function of the input signal in 65 nm CMOS process with supply voltage of $v_{dd} = 1.2\,\mathrm{V}$.

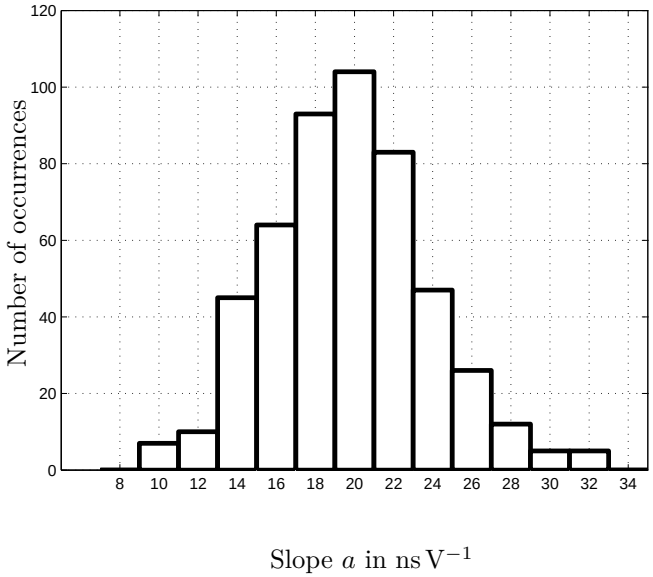


Figure 2.8: Statistical distribution of the slope a in 65 nm CMOS process with supply voltage of $v_{dd} = 1.2\,\mathrm{V}$.

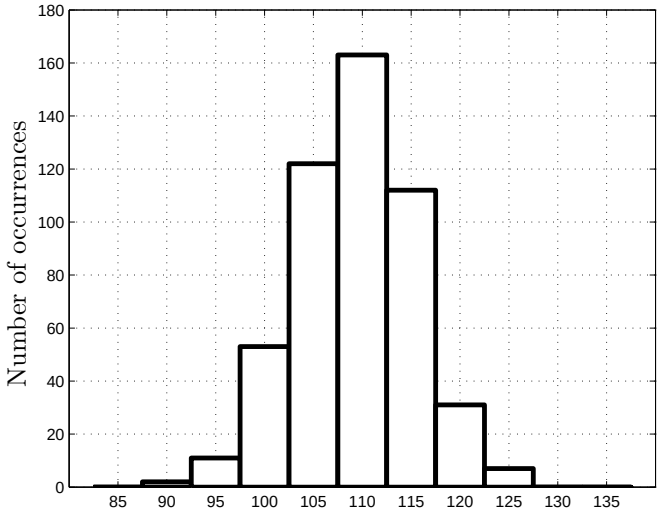


Figure 2.9: Statistical distribution of the constant resistance b_{on} in 65 nm CMOS process with supply voltage of $v_{dd} = 1.2\,\mathrm{V}$.

$$\begin{aligned} \tau(x) = R_{on}C &= b + ax \\ a &= \alpha + \beta b \end{aligned} \tag{2.17}$$

For the simulation of Fig. 2.10, we obtained $\alpha = -39.5\,\mathrm{ns\,V^{-1}}$ and $\beta = 0.5\,\mathrm{V^{-1}}$.

Single ended architecture

We can solve the ODE of (2.11) by considering the linear dependency of the time constant $\tau(x)$ with the input signal x of (2.17). For this purpose we make the following assumptions:

- The sampling duration βT_s is several times bigger than the time constant τ in order to ensure accurate sampling.
- The S/H is weakly nonlinear i.e $ax \ll b$ in (2.17)
- There is no memory effect, i.e the charge of the hold capacitor is set to zero after each sample

The n^{th} sample at the S/H output can be written as a desired component and an undesired component (see Appendix A.2):

$$y[n] = \overbrace{(h \star x)(nT_s)}^{desired} - \underbrace{\frac{a}{b}\Big\{(h \star x^2)(nT_s) - \Big(h \star \Big[x.(h \star x)\Big]\Big)(nT_s)\Big\}}_{undesired\ component} \tag{2.18}$$

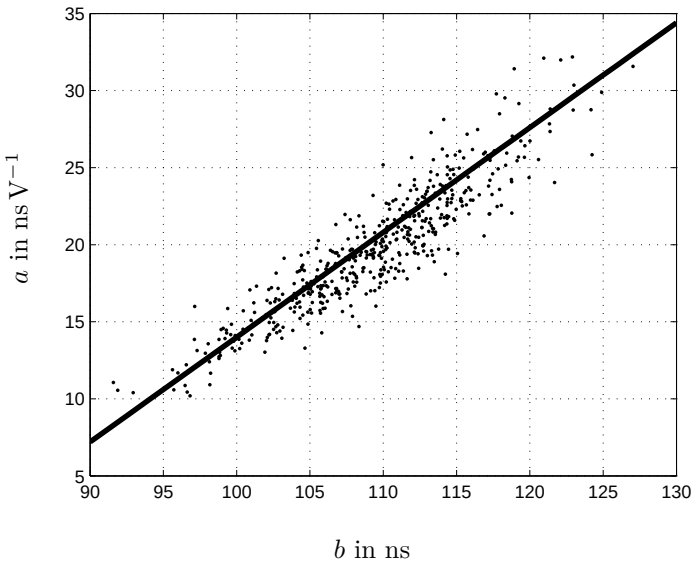


Figure 2.10: Correlation between a and b in 65 nm CMOS process with supply voltage of $v_{dd} = 1.2\,\mathrm{V}$.

With h the impulse response of the linear S/H which is obtained by taking the static component b of the time constant $\tau(x)$:

$$h(t) = \frac{1}{b}\exp(-\frac{t}{b})u(t) \tag{2.19}$$

$u(t)$ in (2.19) is the Heaviside step function. The desired part is what should be obtained if the on-resistance were totally constant while the undesired component comes from the variations of the on-resistance with the input signal $x(t)$. As $a \to 0$, the resistance becomes constant and the undesired component decreases. (2.18) is far more simple than what has been obtained in [31].

Taking the particular case of a sinusoidal input $x(t) = A\sin(2\pi f_0 t)$, the Discrete Time Fourier Transform (DTFT) $Y(f)$ of $y[n]$ is:

$$Y(f) \simeq \overbrace{\frac{aA^2}{4b}\Big[H(f_0) + H(-f_0) - 2\Big]}^{Offset} + \overbrace{\frac{1}{2}AH(f_0)\delta(f - f_0)}^{desired\ part} + \underbrace{\frac{aA^2}{4b}H(2f_0)K(f_0)\delta(f - 2f_0)}_{second\ harmonic} \tag{2.20}$$

$H(f)$ in (2.20) is the transfer function of the S/H and $K(f) = H(f) - 1$.

We notice that firstly the S/H exhibits an offset which can easily be removed digitally. Secondly the nonlinearity is mostly characterized by the presence of a dominant harmonic of second order which will degrade the dynamic performances such as SFDR and SNDR.

Frequency	According to (2.20)	By simulation
HD_2 in dB	61.25	61.20

Table 2.1: Simulation results of a S/H with $\tau(x) = (1.25e-10 + 2.33e-11\ x$. The input signal is $x(t) = 0.6\sin(2\pi f_o t)$, $f_o = 20\,\text{MHz}$, $f_s = 300\,\text{MHz}$.

Fig. 2.11 presents the simulation of the output spectrum of a S/H with a time constant varying linearly with the input signal and Tab. 2.1 compares the simulation results with (2.20). Considering a sinusoidal input, the worst spurious is effectively at $2f_0$. The Second Harmonic Distortion (HD_2), defined as the ratio of the power of fundamental signal to the power of the second harmonic matches very well with (2.20).

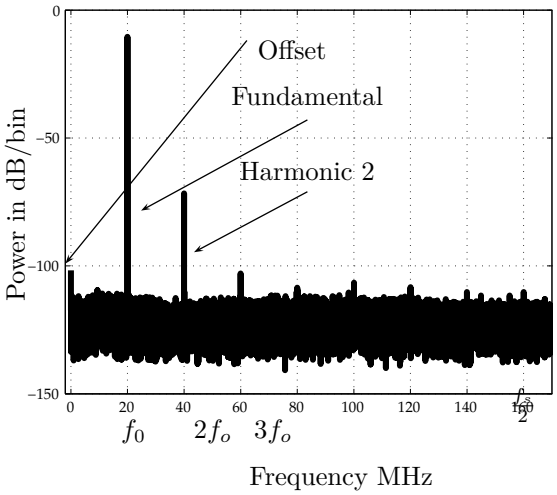


Figure 2.11: Output spectrum of a S/H with $\tau(x) = (1.25e-10 + 2.3e-11\ x$. The input signal is $x(t) = 0.6\sin(2\pi f_o t)$, $f_o = 20\,\text{MHz}$, $f_s = 300\,\text{MHz}$ and the number of fft points is 16384

.

Differential architecture

As the worst spurious created by nonlinearity in a bootstrapped S/H is the second harmonic, it can be mitigate with a differential architecture. To do it, for an input signal $x(t)$, we send the signal $x_1(t) = \frac{1}{2}x(t)$ on the first channel and on the second channel we send the signal $x_2(t) = -\frac{1}{2}x(t)$. Fig. 2.12 shows a differential bootstrap S/H circuit.

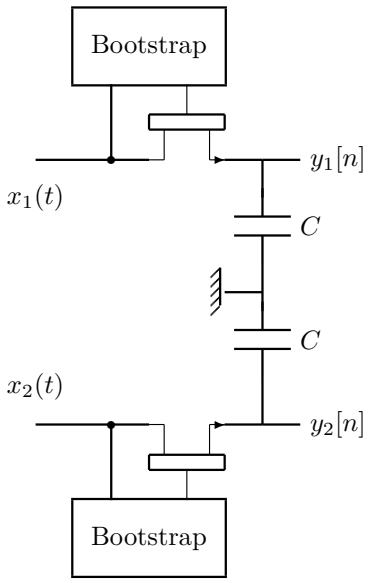


Figure 2.12: Differential Bootstrap S/H circuit.

From (2.20), the second harmonic of the $i_{1,2}^{th}$ channel is obtained by dividing the amplitude by two :

$$H_{2i} = \frac{a_i A^2}{16 b_i} H_i(2f_0) K_i(f_0) \tag{2.21}$$

Where $H_i(f) = \frac{1}{1+j2\pi f b_i}$ is the transfer function of the i^{th} channel and $K_i(f) = H_i(f) - 1$. The time constant of channel $i_{1,2}$ is $\tau_i = b_i + a_i x_i(t)$. The second harmonic of the whole S/H is obtained by making the difference of the second harmonic of channels 1 and 2:

$$\begin{aligned} H_2 &= H_{21} - H_{22} \\ &\simeq \frac{\lambda A^2}{16} \overline{H}(2f_0)\overline{K}(f_0)\Big[\frac{\theta}{\lambda} + \overline{K}(2f_0) + \overline{H}(f_0)\Big]\frac{\Delta b}{\overline{b}} \end{aligned} \tag{2.22}$$

$\overline{b}$ is the average value of b, $\overline{H} = \frac{1}{1+j2\pi f\overline{b}}$, the average transfer function, $\overline{K} = \overline{H} - 1$, $\lambda = \frac{\alpha}{\overline{b}} + \beta$ and $\theta = \frac{\alpha}{\overline{b}}$. α and β are defined in (2.17)

The HD_2 can be derived from (2.22) and (2.20) as :

$$HD_2 = 10\log_{10}\Big[|c|^2 / \Big(\frac{\Delta b}{\overline{b}}\Big)^2\Big] \tag{2.23}$$

With c the complex number given by:

$$c = \frac{8\overline{H}(f_0)}{\lambda A \overline{H}(2f_0)\overline{K}(f_0)\Big[\frac{\theta}{\lambda} + \overline{K}(2f_0) + \overline{H}(f_0)\Big]} \tag{2.24}$$

Frequency	$\frac{\Delta b}{\overline{b}} = 0.062$	$\frac{\Delta b}{\overline{b}} = -0.041$
Simulated HD_2	77.09	80.23
HD_2 with (2.23)	76.96	80.52

Table 2.2: Simulation results of a differential bootstrapped S/H with $\overline{\beta} = 0.125\,\text{ns}$. The input signal is $x(t) = 0.6\sin(2\pi f_o t)$, $f_o = 20\,\text{MHz}$ and $f_s = 300\,\text{MHz}$.

Table. 2.2 compares some simulation results with (2.23). As we can see this analytical model match very well and the HD_2 is considerably better than what was obtained in section 2.2.3.

Probabilistic Description of the Second Harmonic Distortion

Device mismatches are inherent to any manufacturing processes whatever the technology is (CMOS, SiGe, ...) and are commonly described by random variables normally distributed. Since these mismatches are responsible for channel mismatch errors, it is convenient to rely on a probabilistic characterization of the time constant mismatches. As a consequence, the harmonic distortion is modeled by a statistical distribution related to the standard deviation σ of time constant mismatch as in [3].

Readily, $\frac{\Delta b}{\overline{b}} = \sigma\epsilon$. With $\epsilon \sim \mathcal{N}(0,1)$. From (2.23) the HD_2 can be rewritten as:

$$HD_2 = 10\log_{10}\left(\frac{|c|^2}{\sigma^2\epsilon^2}\right) = F(\epsilon^2) \tag{2.25}$$

As the HD_2 is function of the random variable ϵ^2, it is also a random variable whose dispersion depends on standard deviation of mismatches σ. The random variable ϵ^2 follows a chi-squared distribution with one degree of freedom χ_2^1 and the function F is strictly monotone. Therefore the Probability Density Function (PDF) p of the HD_2 can be obtained from that of ϵ^2 with a change of variables. All calculations done, we find that the PDF of the HD_2 is:

$$\begin{aligned} p(HD_2) &= \frac{1}{F'\Big[F^{-1}(HD_2)\Big]}\chi_2^1\Big(F^{-1}(HD_2)\Big) \\ &= \sqrt{\frac{C}{2\pi}}\frac{\log(10)}{10\sigma}10^{-\frac{HD_2}{20}}\exp^{-\left[\frac{C}{2\sigma^2}10^{-\frac{HD_2}{10}}\right]} \end{aligned} \tag{2.26}$$

Fig. 2.13 shows the statistical distribution of the HD_2 obtained by simulation and the calculated PDF. Both curves match almost.

Reliability of HD_2

Using an analogy to the usual yield, we introduce the robustness criterion $HD_2(\eta)$, that states the HD_2 remains higher than this threshold value with

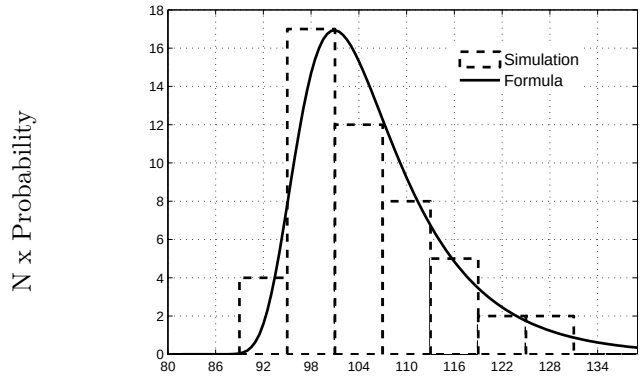


Figure 2.13: Statistical distribution of second harmonic distortion of a differential bootstrapped S/H in 65 nm CMOS process with an input signal $x(t) = 0.6\sin(2\pi f_o t)$, $f_o = 20\,\text{MHz}$, $f_s = 300\,\text{MHz}$, a relative mismatch of 1.2 % and N = 50 points.

a probability $1-\eta$.

$$\begin{aligned}\eta &= p\Big(HD < HD(\eta)\Big)\\ &= 1 - p\Big(\epsilon^2 < F^{-1}(HD(\eta)\Big)\\ &= 1 - \text{erf}\left(\sqrt{\frac{\Big[F^{-1}(HD(\eta))\Big]^2}{2}}\right)\end{aligned} \tag{2.27}$$

erf is the Gauss error function.

Inverting (2.27) with respect to $SFDR(\eta)$ gives :

$$HD_2(\eta) = -10\log_{10}\frac{\sigma^2}{|c|^2} - 10\log_{10}K(\eta) \tag{2.28}$$

With $K(\eta) \simeq \frac{\pi}{2}(1-\eta)^2$. (2.28) states the HD_2 remains higher than the threshold value $HD_2(\eta)$ with the probability $1-\eta$ allowing to control the reliability of any mismatch calibration process like in [3].

Fig. 2.14 shows the harmonic distortion law as a function of the standard deviation of mismatches for a frequency of 20 MHz. To obtain a HD_2 of 100 dB with a reliability of 0.999, the mismatch should be less or equal than 1 %.

2.2.4 Charge Injection and Clock Feedthrough

Others non-idealities of S/H circuits are charge injection and clock feedthrough. Fig. 2.15 represents the equivalent model of the transistor in the triode region. C_{gs} and C_{gd} are respectively the gate-source and the gate-drain parasitic capacitance.

When the transistor turns OFF, the channel charge Q_{ch} is dispersed into the source and drain. The channel charge depends on the gate-source voltage [32]:

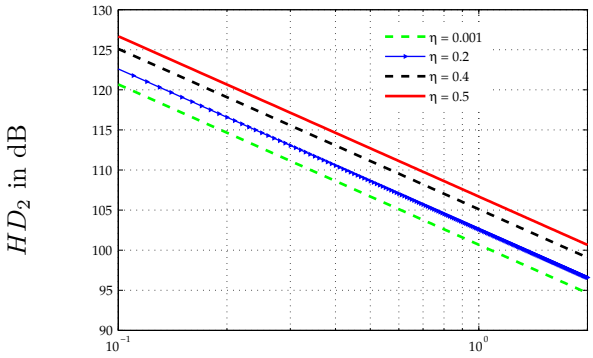


Figure 2.14: Harmonic distortion law

$$Q_{ch} = -WLC_{ox}(V_{gs} - V_{th}) \tag{2.29}$$

Only the fraction injected in the drain will produce an error which will depend linearly on the input signal. If we denote k_{ch} this fraction, then the charge injection error is given by :

$$\Delta y_{ch} = \frac{k_{ch}Q_{ch}}{C} = -\frac{k_{ch}WLC_{ox}(V_{gs} - V_{th})}{C} \tag{2.30}$$

The charge injection appears as a gain error and an offset. Similarly, the clock feedthrough offset error Δy_{clk} due to the gate-drain parasitic capacitance C_{gd} is given by [32] :

$$\Delta y_{clk} = -\frac{C_{gd}}{C_{gd} + C}V_{dd} \tag{2.31}$$

Table. 2.3 gives some order of magnitude of the ON resistance, charge injection gain and clock feedthrough offset error in 28 nm HPL process.

2.3 Others Sample and Hold architectures

2.3.1 Close loop S/H

The signal-dependent charge injection mentioned above can be avoided by operating the switch at a constant potential, which can be realized by enclosing the switch in a feedback loop to create a virtual ground. Fig.2.16 shows a basic closed-loop S/H circuit following this idea [33].

The drawback of this architecture is that it uses two opamps in tracking mode with a positive feedback loop. Therefore a heavy compensation is needed in order to avoid instability. This naturally reduces the speed of the circuit. Further more, for an interleaving factor of M, $2M$ opamps will be needed. This will consume a large portion of the total power as well as a large chip area.

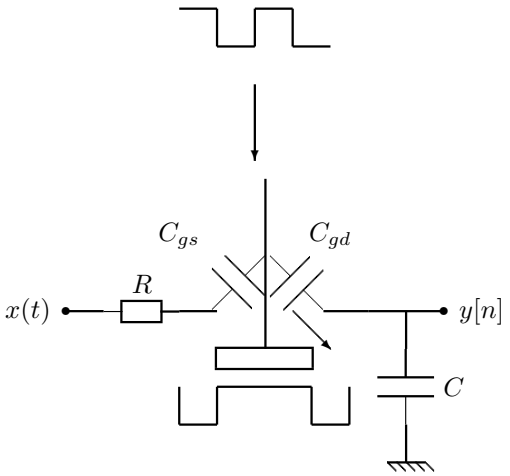


Figure 2.15: Charge injection and clock feedthrough.

Parameter	Symbol or formula	Value	Unit
NMOS transistor width	W	1.4	µm
NMOS transistor length	L	30	nm
Overlap length	x_d	4	nm
Oxide thickness	t_{ox}	1.4	nm
Oxyde Permittivity	ϵ_{ox}	3.51×10^{-13}	$\mathrm{F\,cm^{-1}}$
Gate capacitance per unit area	$C_{ox} = \epsilon_{ox}/t_{ox}$	25	$\mathrm{fF\,\mu m^{-2}}$
Threshold voltage	V_{th}	0.536	V
Supply voltage	V_{dd}	1	V
Gate-drain parasitic capacitance	$C_{gd} = \epsilon_{ox} W(L + 2x_d)/2t_{ox}$	0.4	fF
Electron mobility	μ	400	$\mathrm{cm^2\,V^{-1}\,s}$
Hold capacitor	C	4	pF
Charge injection coefficient	k_{ch}	0.5	
Charge injection gain	$G_{ch} = k_{ch} W L C_{ox}/C$	1.3×10^{-4}	
Clock feedthrough offset	$y_{clk} = \frac{C_{gd}}{C_{gd}+C} V_{dd}$	0.2	V
On resistance	$R_{on} \simeq \left[\mu C_{ox} \frac{W}{L}(V_{dd} - V_{th})\right]^{-1}$	46	Ω

Table 2.3: Parameter values used to model S/H non-idealities in 28 nm HPL technology

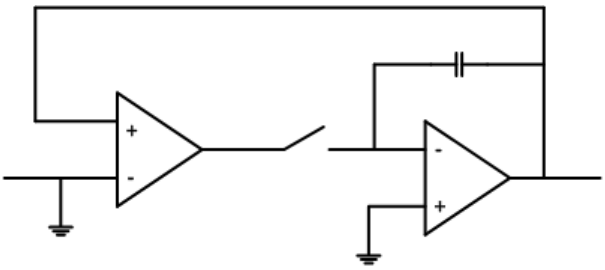


Figure 2.16: Typical close loop S/H circuit

2.3.2 Switched Capacitor S/H

A commonly used S/H is Switched Capacitor (SC) S/H shown on Fig. 2.17. The sampling is performed passively i.e without the opamp making the signal acquisition fast. The clocks CLK_1 and CLK_2 have a frequency f_s and are in phase opposition. When CLK_1 is high, CLK_2 is low, and the input signal charges the capacitor. It is the sampling mode. During the Hold mode, CLK_1 is low, CLK_2 is high and the sample loaded accross C is transfered to the output. To avoid charge injection, CLK_{1p} is slightly advanced to CLK_1. This is known as bottom plate sampling technique [34] [33].

2.3.3 Double Sampling S/H

The double-sampling is a technique to double the sampling rate of Switched Capacitor (SC) S/H circuit with only a minor increase in power consumption [35] [36]. Fig. 2.18 shows a double sampling SC S/H. The clocks CLK_1 and CLK_2 have a frequency f_s and are in phase opposition. When CLK_1 is high, CLK_2 is low, and the first sample is loaded accross C_1. Then when CLK_1 is low, CLK_2 is high, the first sample loaded accross C_1 is forwarded to the ADC for quantization and the second sample is loaded accross C_2 and so on. As a result, every $T_s/2$ there is a sample at the output of the ADC and the sampling frequency is thus doubled. To avoid charge injection, bottom plate sampling technique is used.

2.4 Quantization

After the sampling process, the amplitude of the different samples can theoretically take any value in a continue range of values. Quantization is the conversion of a discrete-time analogue signal into a digital signal. All the values at the output of the quantizer are multiple of an elementary quantity q called quantization step or Least Significant Bit (LSB). The LSB and the resolution N are related by (2.32) :

$$q = \frac{V_{max} - V_{min}}{2^N - 1} \tag{2.32}$$

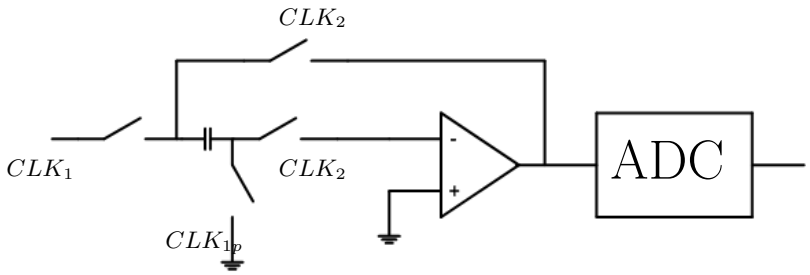


Figure 2.17: Switch Capacitor S/H circuit

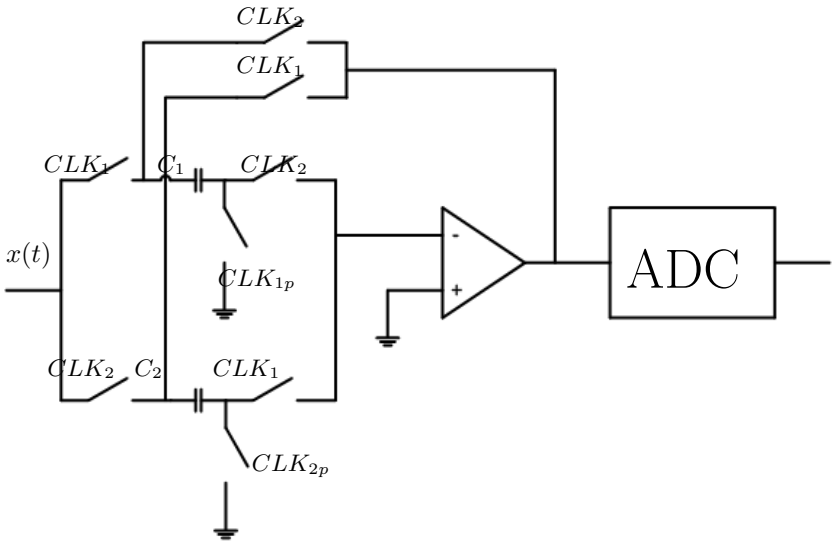


Figure 2.18: Double sampling technique

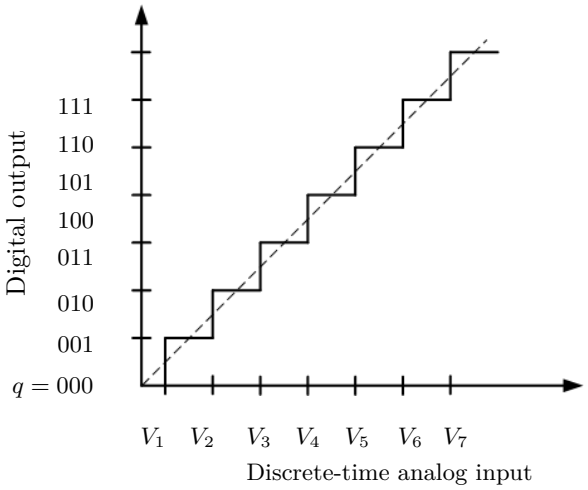


Figure 2.19: Quantization characteristic

Where V_{min} and V_{max} respectively are the upper and lower extremes of the voltages that can be coded.

Quantization structures can be implemented in many different ways. Different architectures are suitable for some specific applications regarding the speed, the accuracy, the area and the power consumption. The most popular quantization architecture are: Flash, Pipeline, Sigma-Delta and Successive Approximation Register (SAR).

2.4.1 Flash Architecture

Flash converters [18] [13] [37] [38] [39], are the simplest and fastest converters in the ADC family. The parallel nature of Flash converter makes it suitable for high speed high bandwidth applications. The drawback of this architecture is that it is power hungry, consumes significant die area and offers only low to moderate output resolution. This limit Flash converters to high frequency applications such as data satellite communication, radar processing, sampling oscilloscopes and high-density disk drive.

In a Flash converter, a resistance ladder is used to generate voltage reference levels. Then a constant voltage V_{ref} is applied to the whole resistance ladder, and the voltage levels between the resistances are used as reference levels. The analog input signal or the output of the S/H is then compared to the reference levels from the resistance ladder to determine which level is closest. This means that to get a precision of N bits in the ADC, 2^N resistances and $2^N - 1$ comparators are required.

Since all reference levels are compared to the analog signal simultaneously, the conversion time is constant, independent of the number of bits. This also means that a Flash ADC is very fast. However, the drawback is that the hardware grows exponentially with the number of bits. The power consumption also grows exponentially.

ADC type	Resolution	Sampling rate	Characteristic
Flash	4 to 8 bits	100 MHz to 5 GHz	• High speed • High bandwidth • High power consumption • Large area • Matching difficulties
Pipeline	12 to 16 bits	10 MHz to 100 MHz	• High throughput • Moderate bandwidth • Low power consumption • Moderate area • Self calibration technique
SAR	10 to 16 bits	50 kHz to 5 MHz	• Very high resolution and accuracy • Low bandwidth • Low power consumption
Sigma-Delta	14 to 20 bits	100 kHz to 500 MHz	• High output resolution • Moderate to high speed • High bandwidth • Moderate power consumption • On Chip digital filtering

Table 2.4: Comparisons of differents converters achitectures

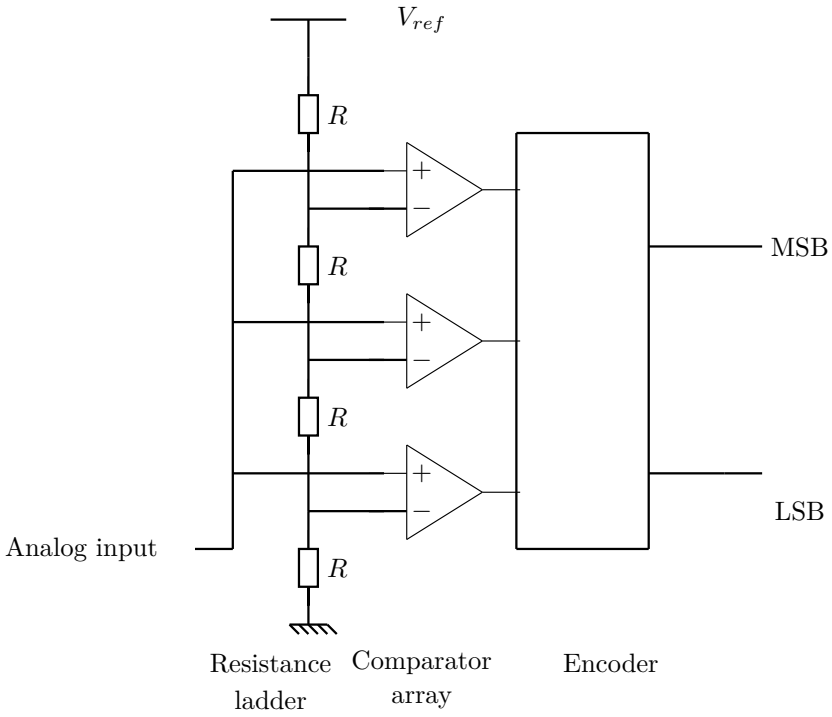


Figure 2.20: A 2 bits Flash converter.

2.4.2 Successive Approximations Architecture

Successive Approximation Register (SAR) architectures are very popular for high resolution and low to medium speed applications. Current architectures have capabilities of sampling at several megahertz with a resolution ranging from 9 to 18 bits[37].

The basic architecture of the SAR ADC consists of a DAC, an analog comparator and a SAR logic [40][41]. Fig. 2.21 shows a 4 bit SAR using a binary weighted capacitor DAC array. The DAC capacitor array has four binary-weighted ($C_i = 2^i C_0$, for i=0 to 3) and one redundant ($Cr = C_0$) capacitors.

In the figure, the switches are shown in the sampling mode where the analog input is constantly charging or discharging the parallel combination of all capacitors.

The hold mode is initiated by opening S_g and by connecting $Sr, S_0 ... S_3$ to ground. The voltage at node X becomes $-V_{in}$.

After that, S_3 is connected to V_{ref} and this forces V_x to be:

$$V_x = -V_{in} + \frac{C_3}{C_{total}} V_{ref} = -V_{in} + \frac{1}{2} V_{ref} \tag{2.33}$$

Where $C_{total} = 2^N C_0$ denotes the total capacitance of the DAC capacitor array. If V_x is negative then V_{in} is greater than $V_{ref}/2$ and we have $D_3 = 1$. Otherwise we have $D_3 = 0$ and the top plate of C_3 will be reconnected to ground. For an

N bits SAR ADC, this process iterates N times, with a smaller capacitor being switched each time, until the conversion is finished. In the i^{th} iteration, V_x can be expressed by:

$$V_x(i) = -V_{in} + \frac{C_{IN}(i)}{C_{total}} V_{ref} \tag{2.34}$$

Where $C_{IN}(i)$ is the total capacitance connected to V_{ref} at the i^{th}. At the end of conversion, V_x approaches zero. The transition level of code $m = (D_{N-1}...D_0)$ is:

$$V_m = \frac{\sum_{i=0}^{N-1} D_i C_i}{C_{total}} . V_{ref} \tag{2.35}$$

Due to process variations, capacitors are not exactly matched and we have:

$$C_i = 2^i C_o (1 + \epsilon_i) \tag{2.36}$$

Where ϵ_i is the mismatch of capacitor C_i. Finally (2.35) can be written as:

$$\begin{aligned} V_m &= \frac{\sum_{i=0}^{N-1} D_i 2^i C_0}{2^N C_0} . V_{ref} + \frac{\sum_{i=0}^{N-1} D_i 2^i \epsilon_i C_0}{2^N C_0} . V_{ref} \\ &= V_{ideal}(m) + q \underbrace{\sum_{i=0}^{N-1} D_i 2^i \epsilon_i}_{INL_m} \end{aligned} \tag{2.37}$$

In (2.37) we recognize the expression of the INL of the ADC which depends directly of the N capacitor mismatches of the DAC. This shows that the DAC is a critical component because the linearity of the ADC is limited by the DAC. Further more the SAR ADC speed is also limited by the settling time of the DAC. This makes the accuracy of the SAR converter imposed by the accuracy of the DAC [42].

(2.37) shows that mismatch ϵ_{N-1} on the MSB capacitor is the most significant, then follows ϵ_{N-2} and so on. These capacitor mismatches can be estimated directly by measuring the DNL of the ADC:

$$\begin{aligned} DNL(0) &= \epsilon_0 \\ DNL(2^1 - 1) &= 2\epsilon_1 - \epsilon_0 \\ ... &= ... \\ DNL(2^{N-1} - 1) &= 2^{N-1}\epsilon_{N-1} - ... - \epsilon_0 \end{aligned} \tag{2.38}$$

2.4.3 Pipelined Architecture

Pipeline converters are very popular architectures with conversion speeds from few tens of megahertz to few hundred of megahertz. They have resolution capabilities from 6 to 16 bits. Owing to their high resolution capability and moderate to high sampling rate, they are more widely used in Charge Coupling Device

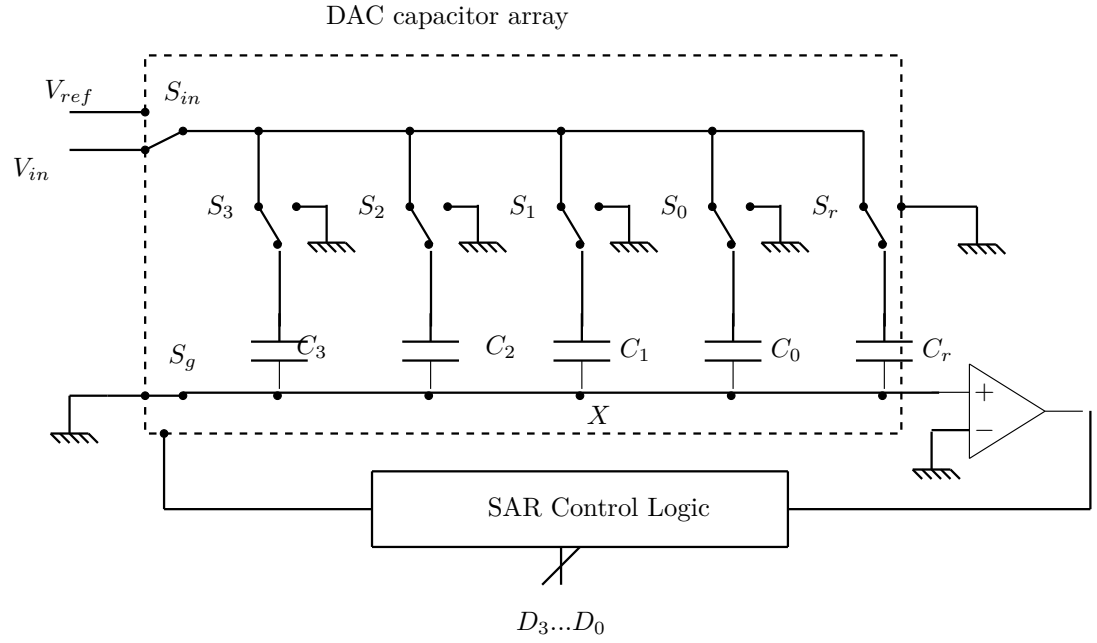


Figure 2.21: Successive Approximation Register.

(CCD), imaging, digital receiver, communication base stations, cable modem and Ethernet device. However for high speed applications beyond the GHz range, Flash topology is still the architecture of choice. A typical pipeline converter is shown on Fig. 2.22 [18] [37].

The converter divides the conversion task into several consecutive stages. Each of them consists of a sample and hold circuit, an m-bit sub-ADC (usually a Flash ADC), and an m-bit DAC . The sample and hold, DAC, subtraction block and amplification form an arithmetic unit called the multiplying digital-to-analog converter (MDAC). First, the sample and hold of the first stage requires the input signal. Then the n-bit Flash converter converts the sampled signal to digital data. The conversion result forms the most significant bits of the digital output. This same digital output is sent to an m-bit digital to analog converter, and its output is subtracted from the original sampled signal. The residual analog signal is then amplified and sent to the next stage in the pipeline to be sampled and converted as it was in the first stage. This process is repeated through as many stages are necessary to achieve the desired resolution.

2.4.4 Delta-Sigma Architecture

The block diagram of a basic Delta Sigma ($\Delta\Sigma$) converter is shown on Fig. 2.23. The elementary DS converter is a one bit sampling system. An analog signal applied to the input of the converter needs to be relatively slow so the converter

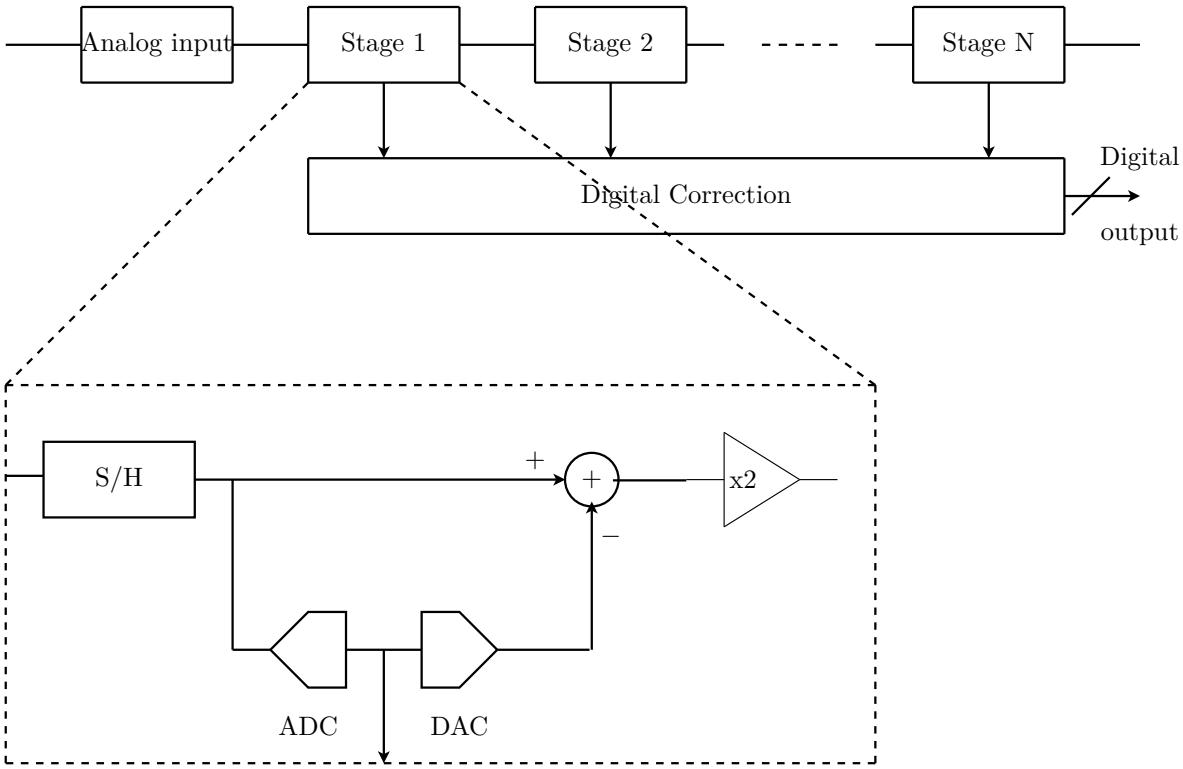


Figure 2.22: Pipeline Converter Architecture.

can sample it multiple times, a technique known as oversampling. The sampling rate can be hundreds of times faster than the digital results at the output ports. Each individual sample is accumulated over time and averaged with the other input-signal samples through the digital/decimation filter to produce a high-resolution slower digital code [37] [18] [43].

2.4.5 Summary on quantization architecture

The performances of SAR, Pipeline, Flash and SD are summarized in Table. 2.4. More details can be found in [37] [38] [43] [18] [44].

From Table. 2.4, Pipeline and SAR architectures emerge as candidates for low power high resolution ADCs. In the past decades, Pipeline ADCs have dominated over SAR ADCs because analog design was easier and capacitors were occupying too much die and area and were lower quality than today. However today as CMOS technology evolves, analog design becomes a very challenging task forcing the designers to use as much digital solution as possible. Further more capacitors become better in term of matching and density. As a result SAR ADCs is attracting more and more designers and are suitable to achieve high resolution with a high power efficiency due to their dominantly digital nature, scalable architecture and to the steady improvement in matching and

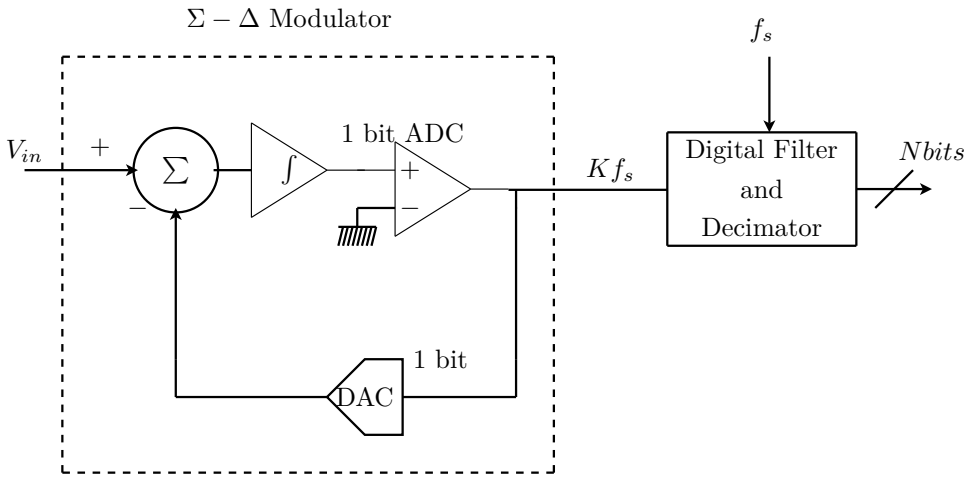


Figure 2.23: Delta Sigma ADC.

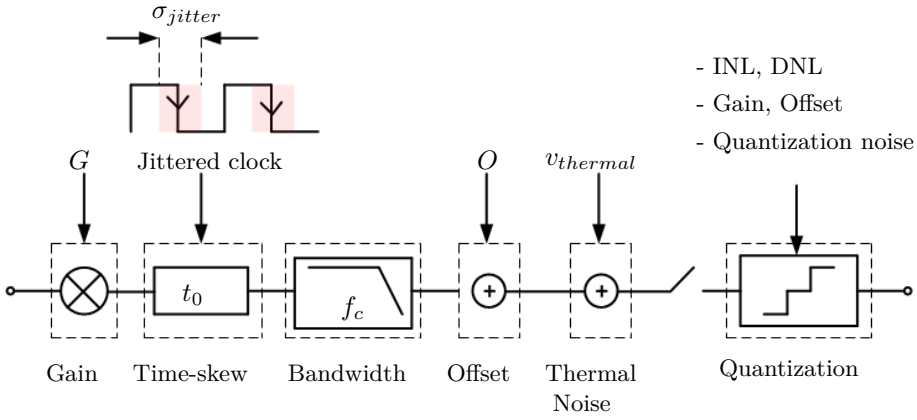


Figure 2.24: Mathematical model of a single ADC

density of Metal-Finger Capacitors(MFC). But the drawback is that they suffer from speed limitations.

2.5 Summary and mathematical model at the ADC output

Typical S/H analyzed above can be reduced at a first order approximation to as low-pass filters with a cutoff frequency $f_c = \frac{1}{2\pi R_{on} C}$ where R_{on} is the combined resistance of all transistors and buffers and C the hold capacitor. G, t_0 and O are respectively the gain, the time-skew and offset of the ADC. We denote $h(\mathbf{p}, t)$ the impulse response of the equivalent low-pass filter parameterized by the vector $\mathbf{p} = (G, t_0, f_c)$ and $H(\mathbf{p}, f)$ its transfer function. Taking into the

account the memory effect of the hold capacitor and the acquisition time βT_s, it can be found that:

$$H(\mathbf{p}, f) = H^{mem}(f)\ \frac{G}{1 + j\frac{f}{f_c}} e^{-j2\pi f t_0} \tag{2.39}$$

Where $H^{mem}(f) = \frac{1-\alpha e^{-j2\pi\beta f T_s}}{1-\alpha e^{-j2\pi f T_s}}$ and $\alpha = e^{-\frac{\beta T_s}{R_{on}C}}$. The demonstration is detailed in Appendix A.4. For ADCs of concern, the acquisition time βT_s is much more greater than the time constant $R_{on}C$ (at least 7 times). Therefore, $H^{mem}(f) \simeq 1$. Nonetheless, α can be taken into account if case arises.

Figure. 2.24 shows the mathematical model of a single ADC. The n^{th} sample at the ADC output output can be splitted into an AC component, a DC component and a noise component :

$$y[n] = y_{AC}[n] + y_{DC}[n] + v_{noise}[n] = \Big(h(\mathbf{p}, .) \star x\Big)(nT_s) + O + v_{noise}[n] \tag{2.40}$$

where $\star$ is the convolution operator.

Without loss of generality, the noise component v_{noise} can be considered independently as an additive contribution which will be analyzed later in chapter 3. The Discrete Time Fourier Transform (DTFT) of $y[n]$ is given by:

$$\begin{aligned} Y(f) &= Y_{AC}(f) + Y_{DC}(f) \\ &= f_s \sum_{k=-\infty}^{+\infty} \Big\{H(\mathbf{p}, .)X(.) + O\delta(.)\Big\}\Big|_{f-kf_s} \end{aligned} \tag{2.41}$$

The filtering component Y_{AC} is made up of replicas of the fundamental signal at multiple of f_s and the DC component appears as a line spectrum at multiple of f_s. This DTFT of the output of a single ADC will be compared to that of a TI-ADCs. It will enable us to highlight spurious components resulting from channel mismatch errors.

2.6 Chapter conclusion

In this chapter a performance analysis of ADC was presented in terms of DC specifications and dynamic performance. DC specifications include gain, offset, INL and DNL and they tell how accurately the quantization is done. Dynamic performance tell how much noise and distortion have been introduced into the signal and are commonly described by THD, SFDR, SNDR and SNR.

Then basic CMOS S/H architectures were studied as well as their non-idealities. These non-idelities consist of clock feedthrough, charge injection, bandwidth limitation and nonlinearity of the on resistance. Clock feedthrough can easily be removed digitally since it appears as an offset. Charge injection can be canceled with a dummy switch or with the bottom plate sampling technique. The popular method to remove the nonlinearity of the on resistance is

the clock bootstrap technique. Unfortunately some nonlinearities remains due to the parasitic capacitances, the backgate effect and the mobility degradation. A deterministic model describing booststrapped S/H circuits both for single ended and differential architecture was proposed. For single ended architecture, the second harmonic is dominant and it can be mitigated with a differential architecture. However mismatches between channels should significantly be reduced to obtain high performances in differential architecture.

After that, the most popular quantization architectures were introduced regarding the speed, the accuracy, the area and the power consumption. Pipeline and SAR have emerged as good candidates for low power TI-ADCs. A model linking the nonlinearities of the SAR ADC to DAC capacitor mismatches was also proposed.

Finally, the expression of the signal at ADC output was derived as the combination of deterministic component and a noise component. The deterministic component is entirely described with the gain, time-skew, bandwidth and offset of the ADC. The next chapter will provide a statistical description of the noise component.

Chapter 3

Noise modeling in ADCs

3.1 Introduction

In previous chapter, we studied S/H and quantization architectures. We saw that ADCs can be characterized by some non-idealities such as nonlinearities, gain, time-skew, bandwidth and offset. Another problem is that all individual ADCs are naturally disturbed by noises which constitute a severe barrier to the achievement of high resolution. Noises include thermal, flicker, jitter and quantization noise. The goal of chapter document is double. On the one hand, it aims to study probability density function and power spectral density of these noises. On the other hand we aim to propose a general framework describing design requirements in terms of thermal noise, jitter and power consumption for low noise ADC.

3.2 Signal model

The n^{th} sample at the ADC output can be written as :

$$y[n] = \Big(h(\mathbf{p},.) \star x\Big)(nT_s) + O + \overbrace{v_{thermal}[n] + v_{flicker}[n] + v_{jitter}[n] + +v_q[n]}^{v_{noise}[n]} \tag{3.1}$$

Where T_s is the sampling period, $v_{thermal}$, $v_{flicker}$, v_{jitter} and v_q are respectively the thermal, the flicker, the jitter and quantization noise. O the offset of the ADC and $h(\mathbf{p},.)$ is the impulse response of the ADC parameterized by $\mathbf{p} = (G, \tau^s, f_c)$.

The total noise power is given by :

$$\overline{v_{noise}^2} = \overline{v_{thermal}^2} + \overline{v_{flicker}^2} + \overline{v_{jitter}^2} + \overline{v_q^2} \tag{3.2}$$

3.3 Quantization Noise

3.3.1 Total power

A first order expression of the total power of quantization noise is [45]:

$$\overline{v_q^2} = \frac{q^2}{12} + \frac{q^2}{3}\sum_{m=0}^{2^N-1} P_m(INL_{m+1}^2 + INL_{m+1}INL_m + INL_m^2) \tag{3.3}$$

Where P_m is the probability of occurence of code m and N the number of bits. If the ADC is linear, then quantization noise is reduced to $\frac{q^2}{12}$ and the SNR for a sinusoidal input signal is given by:

$$SNR_q = 6.02N + 1.76 \tag{3.4}$$

Quantization noise is systematic and in practice we should also take into account the other noise sources. However, most of the time, the ADC is designed such that the major source of noise is quantization.

3.3.2 Probability Density Function

The Probability Density Function (PDF) of the quantization noise of a linear ADC is [46]:

$$p(v_q) = \begin{cases} \frac{1}{q} + \frac{1}{q}\sum_{n\neq 0} \Phi_x\left(\frac{2\pi n}{q}\right)\exp\left(\frac{-j2\pi n v_q}{q}\right) & -q/2 < v_q < q/2 \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

Where Φ_x is the characteristic function of the input signal. The PDF of the quantization noise depends on the statistical properties of the input signal. If Φ_x verifies $\Phi_x(\frac{2\pi n}{q}) = 0$ for all $n \neq 0$, then the quantization noise is uniform.

For the particular case of a zero mean gaussian input signal with a standard deviation σ_x, the PDF of quantization noise becomes :

$$p(v_q) = \begin{cases} \frac{1}{q}\Big[1 + 2\sum_{n\neq 0} \cos\left(\frac{2\pi n v_q}{q}\right)\exp\left(-\frac{2\pi^2 n^2 \sigma_x^2}{q^2}\right)\Big] & -q/2 < v_q < q/2 \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

As the ratio σ_x/q becomes large, the PDF of quantization noise becomes uniform.

3.3.3 Power Spectral Density

There is no general expression for the PDF of the quantization noise. For a gaussian input signal with a standard deviation σ_x, quantization noise becomes white as the ratio σ_x/q becomes large [46].

3.4 Thermal noise

By definition, thermal noise is an electronic noise generated by thermal agitation of charge carriers inside an electrical conductor at thermal equilibrium, which

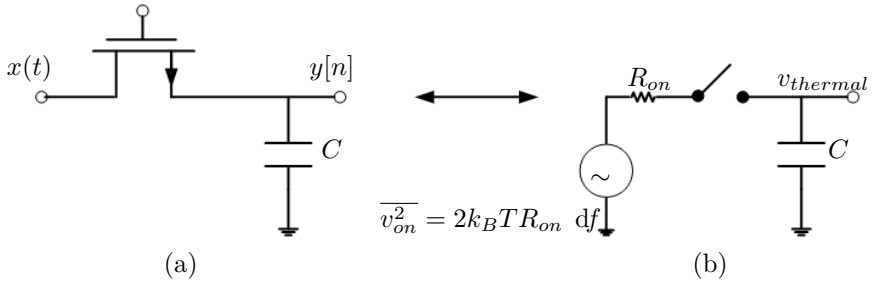


Figure 3.1: (a): Typical S/H circuit. (b): Thermal noise source in a basic S/H.

happens regardless of any applied voltage. In an ideal resistor R, thermal noise is white, gaussian and its Power Spectral Density (PSD) is $2k_BTR$, where k_B is Boltzmann's constant in joules per kelvin and T is the resistor's absolute temperature in kelvins [47].

To examine the impact of thermal noise in any circuits, the first step is to set all input voltages to zero and add noise source associated to each resistance of the circuit. Then the second step consists in evaluating the circuit output voltage in these conditions. Fig. 3.1 (b) shows the circuit for analysis of basic S/H thermal noise of Fig. 3.1 (a). R_{on} is the on-resistance of the transistor switch and v_{on} is the white gaussian noise source that model thermal noise of R_{on}.

3.4.1 Probability Density Function

Thermal noise $v_{thermal}$ at S/H output is gaussian and its total power is $\frac{k_BT}{C}$:

$$v_{thermal} \sim \mathcal{N}\Big(0, \frac{k_BT}{C}\Big) \tag{3.7}$$

See proof in Appendix D.0.1

3.4.2 Power Spectral Density

The Power Spectral Density (PSD) of sampled thermal noise at S/H output is given by:

$$S_{thermal}(f) = \frac{1}{f_s}\frac{k_BT}{C}\frac{1-\exp\Big(-4N_{on}\Big)}{1-2\exp\Big(-2N_{on}\Big)\cos(2\pi\frac{f}{f_s})+\exp\Big(-4N_{on}\Big)} \tag{3.8}$$

Where N_{on} is the number of time-constants defined as the ratio of the sampling duration to the time constant of the sample and hold. The demonstration is detailed in Appendix D.0.2. To ensure accurate sampling N_{on} should be large. For example:

$$N_{on} \geq 7 \longrightarrow \exp\Big(-\frac{T_s}{R_{on}C}\Big) \leq 8.3\times10^{-7} \ll 1. \tag{3.9}$$

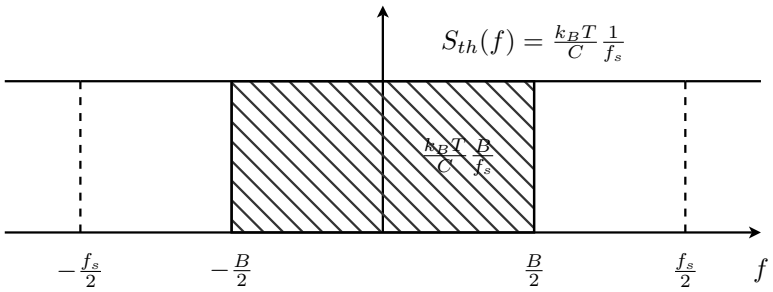


Figure 3.2: Power spectral density of thermal noise

Therefore the power of thermal noise is uniformly distributed in the frequency domain and :

$$S_{thermal}(f) \simeq \frac{1}{f_s} \frac{k_B T}{C} \tag{3.10}$$

For a bandlimited signal, thermal noise power integrated in signal bandwidth is $\frac{k_B T}{C} \frac{B}{f_s}$, with B the bandwidth of the signal.

Thermal noise decreases when the hold capacitor increases and so the designer can size the hold capacitor sufficiently large to mitigate as possible the thermal noise. However increasing the hold capacitor limits the sampling frequency of the S/H. Indeed, for sampling accuracy purposes, the maximum sampling frequency is limited to :

$$f_{s\,max} = \frac{1}{2 N_{on} R_{on} C} \tag{3.11}$$

Therefore the designer has to face a trade-off between the thermal noise, the sampling frequency and the number of time constants. Considering for example a sinusoidal signal $x(t) = A \sin(2\pi f_o t)$, the Signal to Noise Ratio ($SNR_{thermal}$) and the Effective Number Of Bit ($ENOB_{th}$) due to thermal noise inside the nyquist band are respectively:

$$\begin{aligned} SNR_{thermal} &= 10 \log_{10} \left(A^2 / \frac{k_B T}{C} \right) \\ ENOB_{th} &= \frac{10 \log_{10} \left(A^2 / \frac{k_B T}{C} \right) - 1.76}{6.02} \end{aligned} \tag{3.12}$$

Fig. 3.3 shows the evolution of the sampling frequency and the ENOB in function of the hold capacitor. Targetting and $ENOB_{thermal}$ of 14 bit, the hold capacitor should be 6 pF and the sampling frequency should be less than 1 GHz.

3.5 Jitter Noise

Uncertainty of sampling times in S/H cause an error given by :

$$v_{jitter}[n] = x(nT_s + \epsilon[n]) - x(nT_s + \epsilon[0]) \tag{3.13}$$

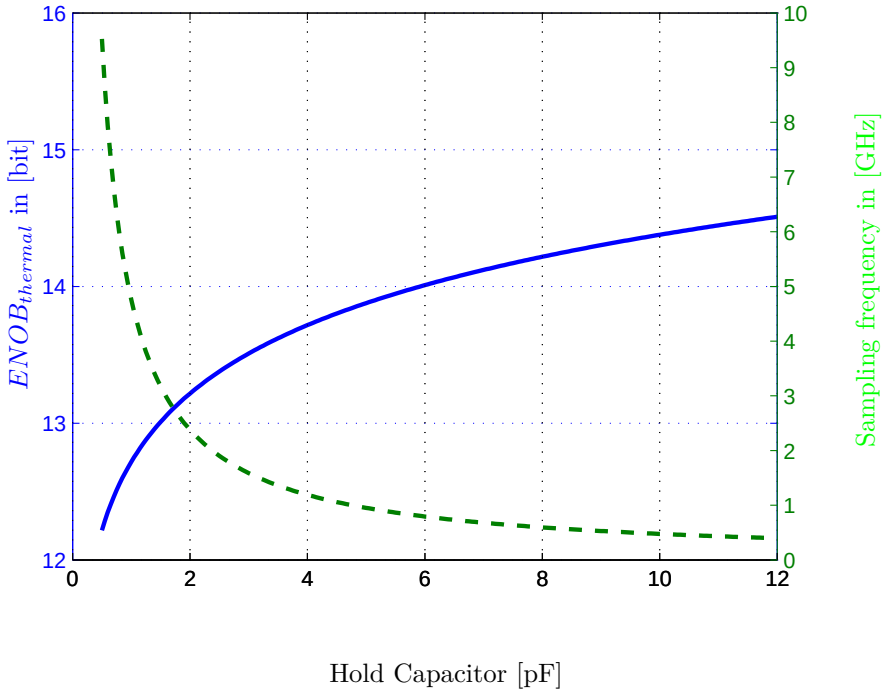


Figure 3.3: Sampling frequency and ENOB due to thermal noise vs Hold capacitor. Simulation done with $N_{on} = 7$, $R_{on} = 15$ and $A = 0.75Vpp$

Where $\epsilon[n]$ is a random time shift at time nT_s. The jitter ξ is defined as the sample instant variations :

$$\xi[n] = \{\epsilon[n] - \epsilon[0]\} \sim \mathcal{N}(0, \sigma^2[n]) \tag{3.14}$$

In general, $\sigma \ll T_s$. For example, considering our wideband applications, $f_s \sim 307.2\,\text{MHz} \rightarrow T_s = 3.25\,\text{ns}$. Typical orders of magnitude of σ is some picoseconds. Taking $\sigma = 1\,\text{ps}$ gives $\sigma = 3 \times 10^{-4} T_s$. So (3.13) can be approximated to :

$$v_{jitter}[n] \simeq \xi[n] \frac{\partial x(t)}{\partial t}\Big|_{t=nT_s} \tag{3.15}$$

3.5.1 Signal to Noise Ratio

Taking the particular case of a sinusoidal input $x(t) = A\sin(2\pi f_o t)$, the total power of jitter noise is derived from (3.15) as :

$$\begin{aligned} \overline{v_{jitter}^2} &= \mathbb{E}\Big[2\pi f_o \xi[n] A\cos(2\pi f_o t)\Big]^2 \\ &= 2\pi^2 f_o^2 A^2 \sigma^2 \end{aligned} \tag{3.16}$$

And the Signal to Noise Ratio (SNR_{jitter}) and the Effective Number Of Bit ($ENOB_{jitter}$) due to jitter are respectively:

$$\begin{aligned} SNR_{jitter}[\text{dB}] &= -20\log_{10}(2\pi f_o \sigma) \\ ENOB_{jitter} &= \frac{-20\log_{10}(2\pi f_o \sigma) - 1.76}{6.02} \end{aligned} \tag{3.17}$$

(3.4) shows that the SNR and ENOB decrease with the jitter and the input signal frequency as plot on Fig. 3.4. This is a significant limiting factor for high

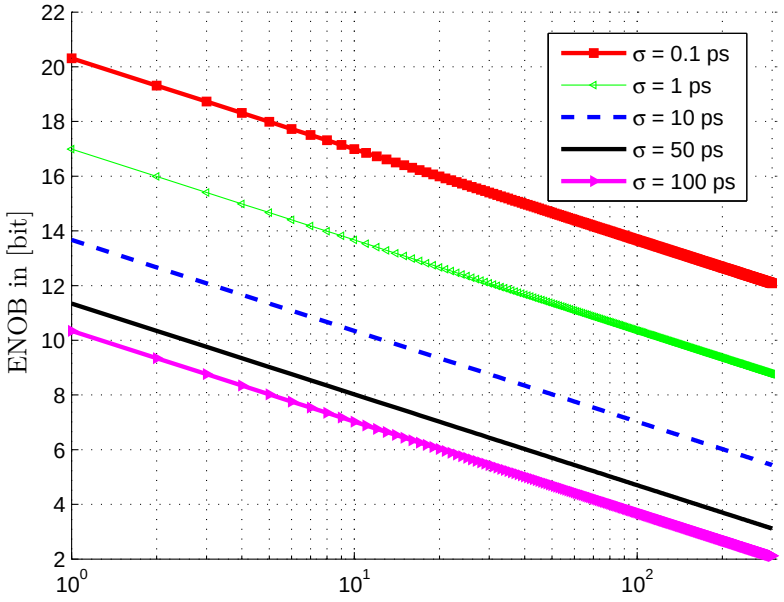


Figure 3.4: ENOB due to jitter vs input frequency

input frequencies. For instance, in order to achieve 14 bit $ENOB_{jitter}$ when sampling a 100 MHz signal, the total jitter must be less than 75 fs. Achieving such a low jitter is very challenging. Indeed low jitter is often contradictory with low power consumption. For example, the PLL Figure Of Merit (FOM) is defined by [48] :

$$FOM_{PLL} = 10\log_{10}\left[\left(\frac{\sigma_{PLL}}{1\,\mathrm{s}}\right)^2.\frac{P_{PLL}}{1\,\mathrm{mW}}\right] \tag{3.18}$$

Where P_{PLL} is the power consumption of PLL. Taking a $FOM_{PLL} = -240\,\mathrm{dB}$ (very close to the state of art) leads to a power consumption of up to 178 mW which is very enormeous for mobile handset applications.

3.5.2 Composition of jitter

The total amount of jitter is the combination of the clock jitter ξ_{clk} and the aperture jitter ξ_{apt} which are independent [49]:

$$\begin{aligned} \xi[n] &= \xi_{clk}[n] + \xi_{apt}[n] \\ \sigma^2 &= \sigma_{clk}^2 + \sigma_{apt}^2 \end{aligned} \tag{3.19}$$

σ_{clk} and σ_{apt} are respectively the standard deviation of the clock and the aperture jitter in seconds. The Clock jitter is time uncertainty caused by the clock generator (PLL, oscillators) while the aperture jitter stands for uncertainty caused by clock distribution (buffer, S/H circuitry).

3.5.3 Aperture jitter

The clock buffer slew rate (SR) translates the thermal noise into time uncertainty called aperture jitter which increases as clock slew rate decreases as depicted on Fig. 3.5 and which is given by:

$$\sigma_{apt} = \frac{v_{thermal}}{SR} \tag{3.20}$$

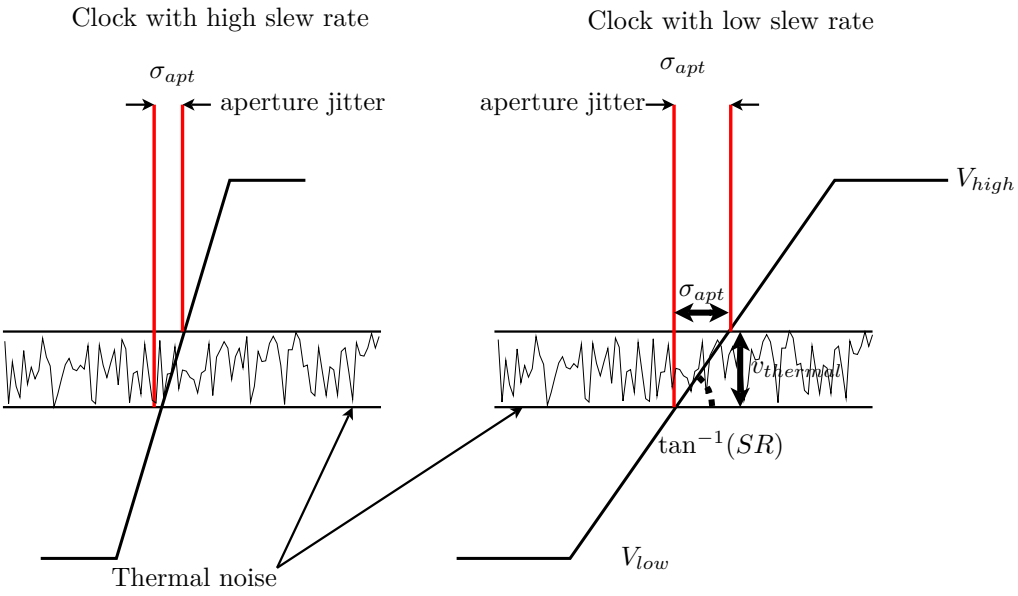


Figure 3.5: Aperture jitter in S/H circuit

The Clock Slew rate SR in $\mathrm{V\,ns^{-1}}$ is defined as the slope of the clock signal at transition instants :

$$SR = \frac{V_{high} - V_{low}}{t_{rising}} \tag{3.21}$$

V_{high}, V_{low} and t_{rising} are respectively the high level, the low level and the rising time of the clock as depicted on Fig.3.5. An ideal clock has an infinite slew rate while a real clock has a finite slew rate. For SysClk (DigRF), V_{high} = 1.2 V, V_{low} = 0.3 V and t_{rising} = 2 ns (measured between 30 % and 70 % of V_{high}) and this gives $SR = 0.24\,\mathrm{V\,ns^{-1}}$ [50].

A capacitor of 4 pF and a temperature of 300 K give a thermal noise of 0.032 mV rms. Considering SysClk (DigRF) which has a slew rate of $SR = 0.2\,\mathrm{V\,ns^{-1}}$, we obtain σ_{apt} = 0.13 ps rms.

Probability density function of aperture jitter

The aperture jitter ξ_{apt} is commonly modeled by a white gaussian process : i.e we have the following properties $\forall n,\ m[\,]$:

- $\xi_{apt}[n] \sim \mathcal{N}(0, \sigma^2_{apt})$
- $\mathbb{E}\Big[\xi_{apt}[n]\xi_{apt}[m]\Big] = \sigma^2_{apt}\delta(n-m)$

Power spectral density of sampling noise due aperture jitter

In [51], the PSD of the noise created by aperture jitter in sampling process has been established for a periodic signal. Based on that and using the above sta-

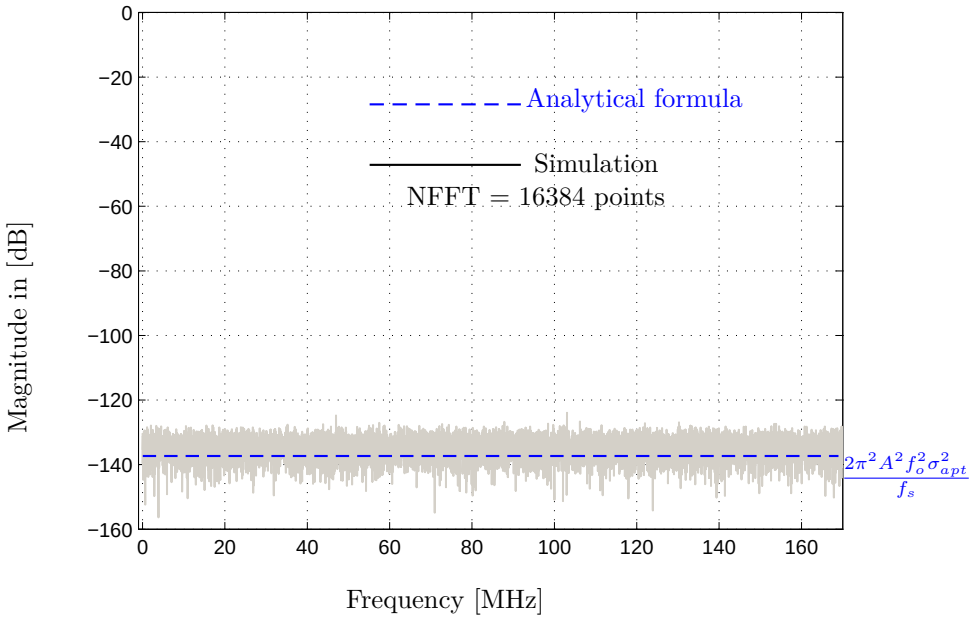


Figure 3.6: Power spectral density of aperture jitter noise in S/H with $f_o = 80\,\text{MHz}$, $\sigma_{apt} = 0.13\,\text{ps rms}$, $f_s = 307.2\,\text{MHz}$ and $A = 0.75\,\text{V}_p$

tistical properties, we establish the PSD of any Wide Sense Stationnary Signal (WSS) $x(t)$ as :

$$S_{jitter\_apt}(f) = \overbrace{S_{xx}(f)(1 - e^{-2\pi^2 f^2 \sigma_{apt}^2})^2}^{Frequency\ dependent\ component} + \overbrace{\frac{1}{f_s}\int_{-\infty}^{+\infty} S_{xx}(f_1)(1 - e^{-4\pi^2 f_1^2 \sigma_{apt}^2})\,\mathrm{d}f_1}^{White\ component} \tag{3.22}$$

The demonstration is detailed in Appendix E. One can see that the aperture jitter is almost white because the frequency dependent component is negligible when compared to the white component in high frequencies. For the particular case of a sinusoidal input $x(t) = A\sin(2\pi f_o t)$, we have :

$$\begin{aligned} S_{jitter\_apt}(f) &\simeq \overbrace{(\pi f_o \sigma_{apt})^4 A^2 \Big[\delta(f + f_o) + \delta(f - f_o)\Big]}^{Frequency\ dependent\ component} + \overbrace{\frac{2(\pi f_o \sigma_{apt})^2 A^2}{f_s}}^{White\ component} \\ &\simeq \overbrace{\frac{1}{f_s} 2(\pi A f_o \sigma_{apt}) A^2}^{White\ component} \end{aligned} \tag{3.23}$$

SysClk (DigRF) buffer slew rate is $0.24\,\text{V ns}^{-1}$ and with a thermal noise of $v_{thermal} = 0.03\,\text{mV}$ it produces an aperture jitter $\sigma_{apt} = 0.13\,\text{ps rms}$. Simulation results of noise resulting from sampling a sinusoidal signal with this clock is presented on Fig. 3.6. We notice that the PSD is effectively white and the shape matches very well with (3.23)

3.5.4 Clock jitter

Clock jitter is time uncertainty coming from clock phase noise. The clock signal $CK(t)$ controlling the S/H can be written as :

$$CK(t) = g\Big[\omega_s\Big(t + \frac{\phi_{OUT}(t)}{\omega_s}\Big)\Big] = g[\omega_s(t + \epsilon(t))] \tag{3.24}$$

Where g is a 2π periodic function, ϕ_{OUT} the clock phase noise and ϵ the random time shift defined in 3.15.
The rms clock jitter after n clock cycles as defined in (3.15) is [52] :

$$\begin{aligned}\sigma^2_{clk}[n] &= \frac{1}{\omega_s^2}\mathbb{E}\Big[\Big|\phi_{OUT}(nT_s) - \phi_{OUT}(0)\Big|^2\Big] \\ &= \frac{2}{\omega_s^2}\Big[R_{\phi_{OUT}}(0) - R_{\phi_{OUT}}(nT_s)\Big] \\ &= \frac{4}{\omega_s^2}\int_{-\infty}^{+\infty} S_{\phi_{OUT}}(\Delta f)\sin^2(\pi\Delta f n T_s)d(\Delta f)\end{aligned} \tag{3.25}$$

Δf is the offset frequency. $R_{\phi_{OUT}}$ and $S_{\phi_{OUT}}$ are respectively the autocorrelation function and the PSD of phase noise. The shape of $S_{\phi_{OUT}}(\Delta f)$ depends on how the clock is made. We will analyze the cases of a free-running oscillator .

Jitter noise caused by a free-running oscillator in ADC

$\longrightarrow$ Probability Density Function :

From equation (3.25), and considering that free-running oscillator phase noise profile is accurately modeled by $\frac{1}{f^2}$ characteristic, we observe that clock jitter linearly increases with cycles. More generally it can be shown that in a free-running oscillator, clock jitter is a wiener process with gaussian increments i.e we have the following properties :

- $\epsilon_{clk}[1] = \lambda_1$
- $\epsilon_{clk}[n] = \sum_{i=1}^{n}\lambda_i$
- $\lambda_i \sim \mathcal{N}(0, \sigma^2_{clk})$
- $\mathbb{E}[\lambda_i\lambda_j] = \sigma^2_{clk}\delta(i-j)$

$\longrightarrow$ Power Spectral Density of jitter noise in S/H:

In [51], the PSD of the jitter noise has been established for a periodic signal. Based on that and using the above statistical properties, we establish the PSD of any Wide Sense Stationnary Signal (WSS) $x(t)$ as :

$$S_{jitter\_clk}(f) \simeq \overbrace{\int_{-\infty}^{+\infty} S_{xx}(f_1)\frac{f_1^2\sigma^2_{clk}f_s}{\pi^2 f_1^4\sigma^4_{clk}f_s^2 + (f-f_1)^2}\,\mathrm{d}f_1}^{Lorentzian\ spectrum} \tag{3.26}$$

Assuming a sinusoidal $x(t) = A\sin(2\pi f_o t)$, $S_{xx}(f) = \frac{A^2}{4}\Big[(\delta(f-f_o) + \delta(f+f_o)\Big]$. Then(3.26) becomes :

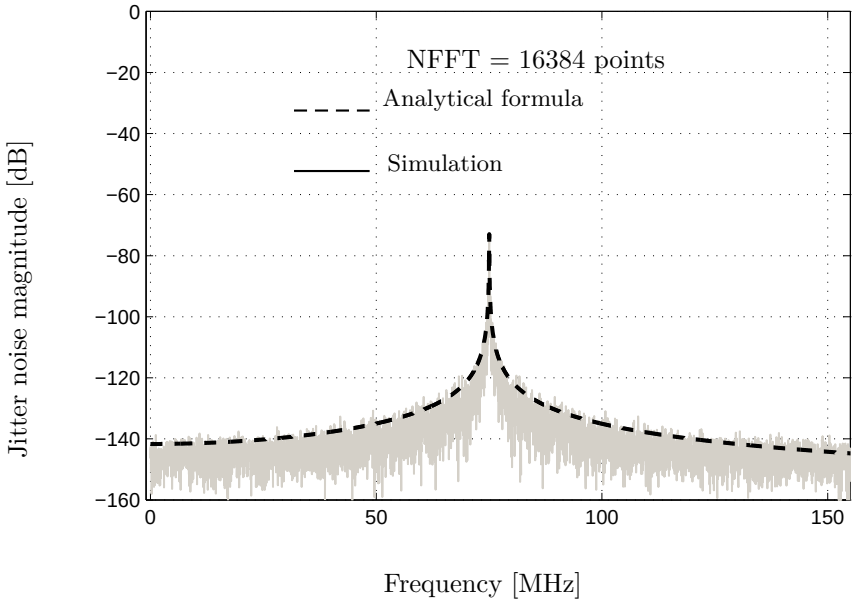


Figure 3.7: Power spectral density of noise resulting from sampling a sinusoidal signal $x(t) = 0.6\sin(2\pi f_0 t)$ with a clock generated by a free-running oscillator. $f_o = 75\,\text{MHz}$, $\sigma_{clk} = 82$ fs, $f_s = 300\,\text{MHz}$

$$S_{jitter\_clk}(f) = N_{fo}(f) + N_{-f_o}(f) \tag{3.27}$$

N_{f_o} is the lorentzian curve given by :

$$N_{f_o}(f) = \frac{A^2}{4}\frac{f_o^2\sigma_{clk}^2 f_s}{\pi^2 f_o^4\sigma_{clk}^4 f_s^2 + (f - f_o)^2}$$

N_{-f_o} is obtained by replacing f_o by $-f_o$ is N_{f_o} .

Fig. 3.7 shows simulation results of noise resulting from sampling a sinusoidal signal with a clock whose jitter follows a Wiener process. We see that the shape of simulation matches very well with analytical formulas (3.27) and (3.5.4).

3.6 Flicker Noise

Flicker noise is a gaussian noise source appearing from the random trapping of charge carriers at the oxide-silicon interface of MOSFETs [53]. The Power Spectral Density (PSD) of flicker noise is given by:

$$S_{flicker}(f) = \frac{K}{C_{ox}WL}\cdot\frac{1}{f} \tag{3.28}$$

Where K, C_{ox}, L and W are respectively the flicker noise coefficient, the gate oxide capacitance per unit area, the channel length and width of the transistor. The importance of flicker noise is analysed by measuring the *corner frequency*

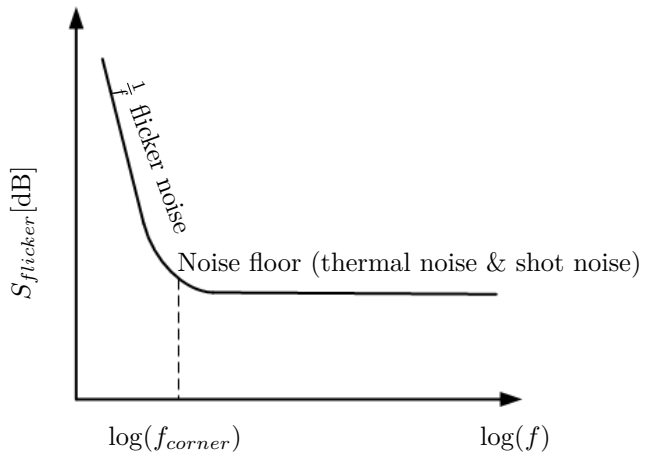


Figure 3.8: Power spectral density of flicker, thermal and shot noise

[54] which is the frequency where it becomes far negligible with respect to white noise (thermal and eventually shot noise) as represented on Fig. 3.8.

As example, ADC of [55] has $\frac{1}{f}$ noise of 0.64 pW/f and a high frequency noise floor of 0.062 pW Hz^{-1}.

3.7 Conclusion

In this chapter we have analyzed statistical properties of noises that occur in single ADC. First we demonstrated how INL/DNL increase quantization noise. For a linear ADC, the total power of quantization noise is equal to the irreductible power of $\frac{q^2}{12}$ where q is the LSB. For gaussian input signals, quantization noise becomes white and uniformly distributed in the range $[-q/2\ q/2]$ as the LSB becomes negligible compared to the standard deviation of the signal.

Secondly we saw that thermal noise is gaussian and has a PSD which becomes white when the number of time constants in the sampling duration increases. Thermal noise can be mitigated by sizing the hold capacitor sufficiently large. But usually there is a tradeoff between thermal noise, sampling frequency and the number of time constants.

Thirdly we saw that jitter noise is a significant limiting factor in ADC especially in wideband ADC. It consists of aperture jitter and clock jitter. Aperture jitter stands for uncertainty caused by clock distribution circuitry. It can be modeled by a white gaussian process which variance depends of clock slew rate and thermal noise magnitude. With a capacitor of 4 pF, the slew rate must be higher than 0.2 V ns^{-1} to ensure that aperture jitter is lower than 0.13 ps. Noise caused by aperture jitter in sampling process has a white PSD.

Clock jitter is time uncertainty caused by the clock generator (PLL, oscillators). For free-running oscillators, there is accumulation of clock jitter with

cycles and in sampling process they cause a noise which PSD is a lorentzian function.

In a PLL the input reference jitter is dominant within the PLL BW and the VCO jitter is dominant outside the PLL BW. When High bandwith and high dynamic range ADC system are considered, ultra low jitter performance is achievable by using some jitter cleaner techniques but at the cost of high power consumption.

Flicker noise is gaussian and has a PSD which decreases with frequency.

Chapter 4

Time-Interleaved ADCs modeling

4.1 Introduction

In the previous chapters, we analyzed non-idealities and noises on stand alone ADCs. However, in order to increase the sampling rate of ADCs beyond a certain process technology limit, Time Interleaved (TI) ADCs have been proposed [2]. In this kind of architecture, several sub-ADCs work in an interleaved manner as if they were effectively a single ADC but working at a much higher rate. The overall sampling frequency is the frequency of one sub-ADC multiplied by the interleaved factor. Fig. 4.1 shows a TI-ADCs architecture. TI-ADCs find applications in electronic systems such as radar, radio receiver and high speed instrumentation.

Time-Interleaved (TI) architectures have emerged as a good way to provide high speed and high resolution data converters with relatively slow circuits. Unfortunately in this kind of architecture, new errors emerge and give rise to nonlinear distortion which significantly degrade the resolution of the overall TI-ADCs. These errors come from discrepancies between the individual sub-ADCs in the system and are commonly referred to as channel mismatch errors. They consist of gain, time-skew, bandwidth and offset mismatch errors.

TI-ADCs have been widely studied for deterministic mismatches [2] [18] [5]. However considering random character of manufacturing process, resulting mismatches and spurious become random variables and their description involves necessarily a statistical modeling. [56] makes a general analysis of random time-skew mismatch. In [57], Probability Density Function (PDF) of both Signal-to-Noise Ratio (SNR) and Spurious-Free-Dynamic Range (SFDR) are explicitly calculated for random gain, time-skew and offset mismatch. [58] provides formula of the expected Signal-to-Noise and Distortion Ratio (SINAD) for random gain, time-skew and offset mismatch. The novelty of this chapter consists in:

- generalizing the deterministic mismatch model including the bandwidth and showing its coupled effect with the time-skew and bandwidth

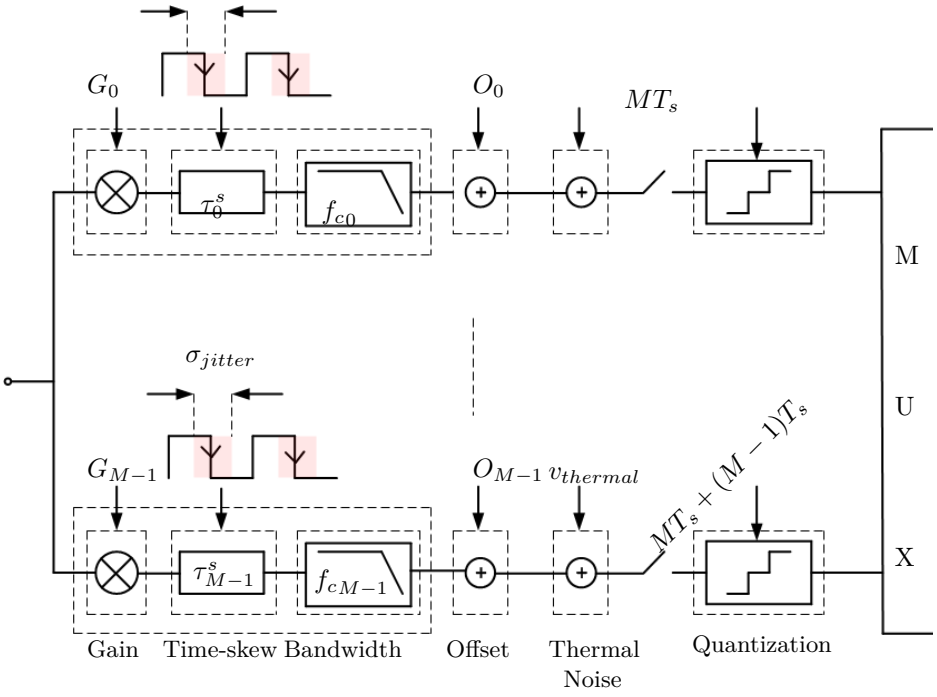


Figure 4.1: M Time-Interleaved ADCs

- determining the probability density function of SFDR for all mismatches including bandwidth
- determining the probability for the SFDR to be lower than a critical value for any mismatch dispersion.

Therefore, for a level of performance determined by a minimum SFDR and its probability of achievement we can specify the required mismatch dispersion. This practical information becomes of relevant importance to establish robust design with safe margins.

4.2 Time Interleaved ADCs Architecture

In the interleaved architecture of Fig. 4.1, each sub-ADC samples the signal at the period MT_s and there is a time shift of T_s between the clock of two consecutive channels as shown on Fig. 4.2 which represents the clock diagram of the different sub-ADCs.

4.2.1 Clock

The TI-ADCs source clock has a frequency $f_s = \frac{1}{T_s}$ and can be generated with an oscillator. A PLL can also be used in order to have a low jitter but at the cost of a higher power consumption.

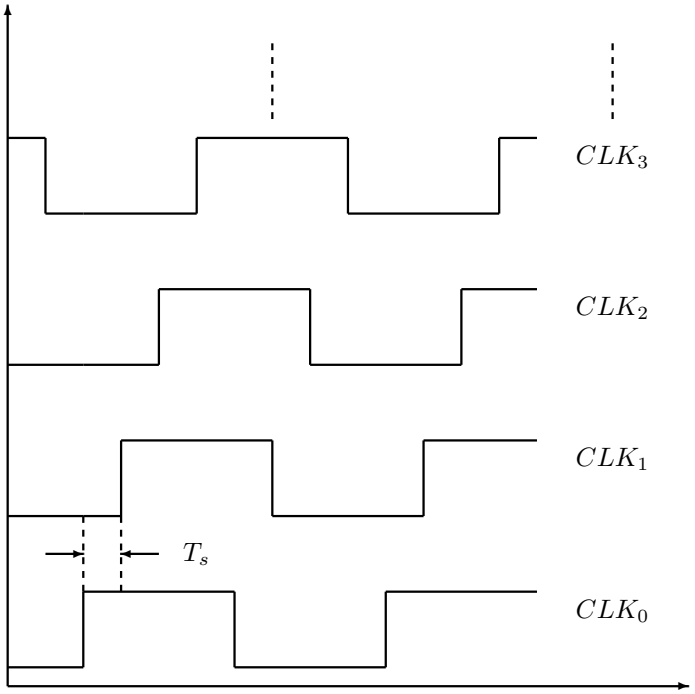


Figure 4.2: Clock diagram of the sub-ADCs

4.2.2 Phase generator

The function of the phase generator is to derive the sub-ADCs clocks of frequency $\frac{f_s}{M}$ from the original clock of frequency f_s as shown on Fig. 4.3. A Delay-Locked Loop (DLL) can be used for this purpose [59] but Shift Registers (SR) are better than DLL because they generate less jitter for a given power budget [60]. Fig. 4.4 shows phase generator for 4 sub-ADCs using SR.

4.2.3 Buffers

Once each sub-ADC clock has been generated, it has to be distributed efficiently to each sub S/H. A typical way to drive clock signal from phase generator to sub-S/H is to insert buffers along clock path as in Fig. 4.5. The function of buffer stages is to regenerate the clock signal in order to ensure satisfactory edge rate at S/H. A buffer consists of two inverters as represented on Fig. 1.4.

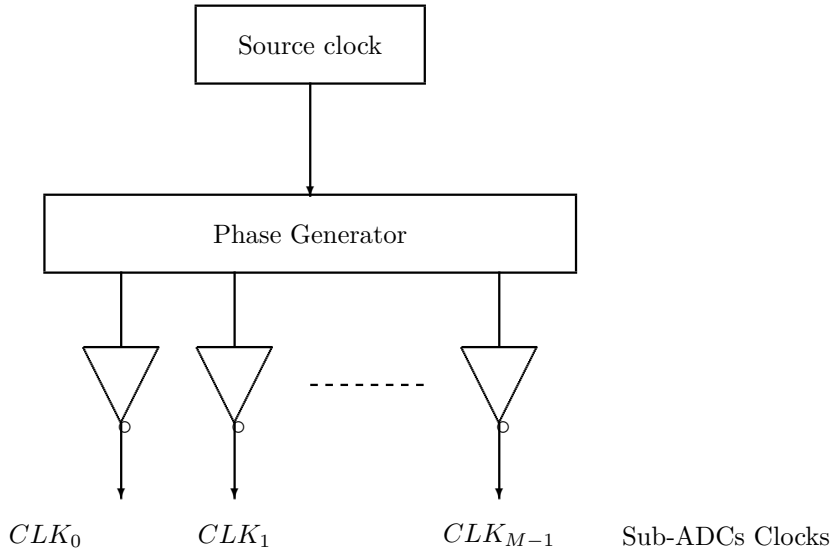


Figure 4.3: Sub-ADCs clock created by a phase generator

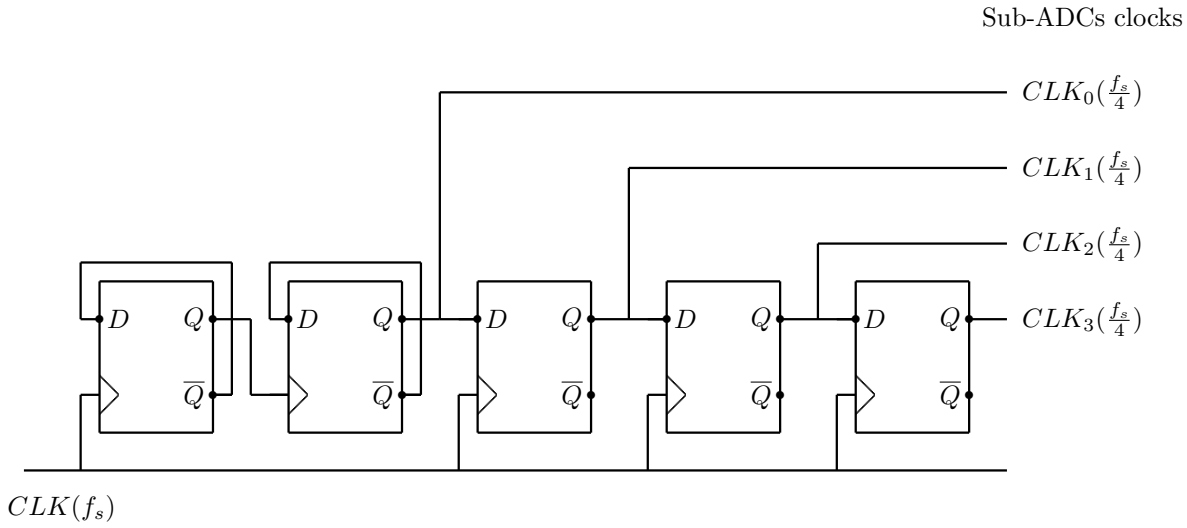


Figure 4.4: Shift Registers Phase generator for 4 sub-ADCs

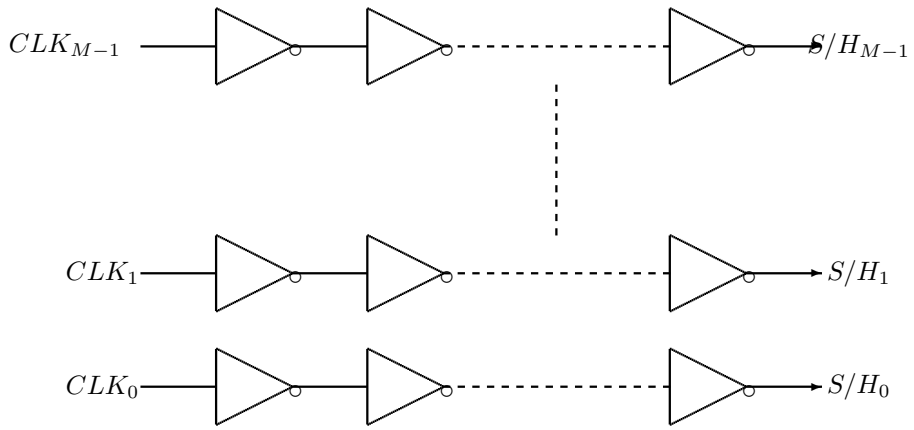


Figure 4.5: Clock distribution with inverters

4.3 Time Domain Analysis

Let's use de following notations:

- $m \in \{0, 1, ...M-1\}$ denotes a given channel.
- $\zeta = \exp\left(\frac{j2\pi}{M}\right)$, the M^{th} unitary root.
- G_m, t_m, f_{cm}, O_m, h_m and H_m are respectively the gain, time-skew, cut-off frequency, offset, impulse response and transfer function of channel m.

By extending (2.40) to interleaving, the AC and the DC components of the TI-ADCs output are:

$$y_{AC}[n] = (h_{n[M]} \star x)(nT_s) \qquad y_{DC}[n] = O_{n[M]} \tag{4.1}$$

4.4 Frequency domain representation

Ideally, all the sub-ADCs should be identical. But in practice there are some mismatches among them. Let's use the following notations for the mismatches:

- $g_m = (1 + \delta g_m)g_0$, with δg_m the relative gain mismatch of channel m. Similarly we define δf_{cm}, δh_m, δH_m and δO_m.
- The relative time-skew mismatch of channel m, δt_m is defined respective to the sampling period T_s: $\delta t_m = (t_m - t_0)/T_s$.

Channel 0 is considered as the reference channel in our analysis. Using (4.1) and the definition of mismatches, the Discrete-Time Fourier Transform (DTFT) Y_{DC} and Y_{AC} of y_{DC} and y_{AC} can be found:

	Gain	Time-skew	Bandwidth	Offset
Q_a	1	$-j2\pi f T_s$	$\frac{j\frac{f}{f_c}}{1+j\frac{f}{f_c}}$	$\frac{\overline{O}}{\lvert\overline{H}\sqrt{P}\rvert}$

Table 4.1: Expression of Q_a

$$\frac{Y_{DC}(f)}{f_s} = \sum_{k=-\infty}^{+\infty} \overbrace{\left[1 + \frac{1}{M}\sum_{m=1}^{M-1}\delta O_m(.)\right] O_0 \delta(f - k f_s)}^{Regular\ part} + \sum_{\substack{k\neq 0[M] \\ -\infty}}^{+\infty} \frac{1}{M} \underbrace{\left[\sum_{m=1}^{M-1} \zeta^{-mk}\delta_m O\right] O_0 \delta(f - k\frac{f_s}{M})}_{Spurious\ part} \quad (4.2)$$

$$\frac{Y_{AC}(f)}{f_s} = \sum_{\substack{k\neq 0[M] \\ -\infty}}^{+\infty} \frac{1}{M} \underbrace{\left[\sum_{m=1}^{M-1} \zeta^{-mk}\delta H_m(.)\right] H_0(.)X(.)}_{Spurious\ part}\Bigg|_{f-k\frac{f_s}{M}} + \sum_{k=-\infty}^{+\infty} \overbrace{\left[1 + \frac{1}{M}\sum_{m=1}^{M-1}\delta H_m(.)\right] H_0(.)X(.)}^{Regular\ part}\Bigg|_{f-kf_s} \quad (4.3)$$

Where $\zeta = e^{j\frac{2\pi}{M}}$ is the M^{th} unitary root. The demonstration is detailed in Appendix B. By reassembling (4.2) and (4.3), $Y(f)$ is written as a sum of a regular part and a spurious part:

$$Y(f) = Y_{regular}(f) + Y_{spur}(f) \quad (4.4)$$

The regular part is made up of replicas of the fundamental signal at multiple of f_s and it corresponds to what would be obtained with a single ADC sampled at f_s. The spurious part consists of replicas at fractions of f_s and comes from discrepancies between each channel. The spurious part of the DC component Y_{DC} appears as a line spectrum independent of the signal at fixed frequencies $k\frac{f_s}{M}$.

4.5 Pairing between mismatches

Based on (2.39), the relative transfer function mismatch δH_m can be written as a function of the gain, time-skew and bandwidth mismatch through a logarithmic derivative:

$$\delta H_m(f) = \delta g_m Q_0(f) + \delta t_m Q_1(f) + \delta f_{cm} Q_2(f) \quad (4.5)$$

$$\text{where} \begin{cases} Q_0(f) = 1 \\ Q_1(f) = j2\pi f T_s \\ Q_2(f) = \frac{j\frac{f}{f_c}}{1+j\frac{f}{f_c}} \end{cases} \tag{4.6}$$

Q_0 is a unit filter and is associtated to the gain mismatch. Q_1 is a differentiator and is associated to the time-skew mismatch. Q_2 is a high pass filter and is associated to the bandwidth mismatch. The advantage of (4.3) and (4.5) is that gain, time-skew and bandwidth mismatches are aggregated in a single mismatch δH_m. (4.5) and (4.6) show that gain mismatch occurs on the real part of δH_m. Time-skew mismatch is frequency dependent and occurs on the imaginary part of δH_m. Bandwidth mismatch is frequency dependent and can combine constructively or destructively with gain and time-skew mismatches.

4.5.1 Spur power analysis

Let $x(t) = \sqrt{P}\exp(j2\pi f_o t)$ be the input signal. Based on (4.3), Y_{AC} spurs appear at frequencies $f_o + k\frac{f_s}{M}$. The power of the k^{th} spur of Y_{AC} is :

$$P_k^{AC} = \frac{1}{M^2}\left|\sum_{m=0}^{M-1}\zeta^{-mk}\delta_m H(f_o)\right|^2 |\overline{H}(f_o)|^2 P \tag{4.7}$$

Likewise, (4.2) gives Y_{DC} k^{th} spur power :

$$P_k^{DC} = \frac{1}{M^2}\left|\sum_{m=0}^{M-1}\zeta^{-mk}\delta_m O\right|^2 O_0^2 \tag{4.8}$$

Let's denote $Q_a(f_o)$ in (4.5) the filter associated to a where a stands for either gain, time-skew or bandwidth. For the special case of offset which is totally independent of the input signal, we define $Q_o(f) = \frac{O_0}{|H(f_o)|\sqrt{P}}$. We note $\epsilon_m^a \in \{\delta g_m, \delta t_m, \delta f_{cm}, \delta O_m\}$ the relative mismatch associated to a and $C_a = |Q_a|^2$. Assuming mismatch a is predominant, the k^{th} spur power is :

$$P_k^{spur} = \frac{C_a}{M^2}\left|\sum_{m=0}^{M-1}\zeta^{-mk}\epsilon_m^a\right|^2 |\overline{H}(f_o)|^2 P \tag{4.9}$$

Fig. 4.6 and Table 4.2 compare some simulation results with (4.9) when using a 4 TI-ADCs with bandwidth mismatches. Considering a sinusoidal input, spurs appear effectively at frequencies $\pm f_o + k\frac{f_s}{M}$ and their power match with (4.9).

4.5.2 Dynamic specifications of TI-ADCs

The signal to the k^{th} spur ratio (SSR_k) referred to the TI-ADC output is defined by:

$$SSR_k = \frac{|\overline{H}(f_o)|^2 P}{P_k^{spur}} = \frac{M^2}{C_a Z_a(k)} \tag{4.10}$$

Frequency	Spur power according to (4.9) [dB]	Simulated spur power [dB]
$-f_o + \frac{f_s}{2}$	-64.81	-64.81
$f_o - \frac{f_s}{4}$	-72.62	-72.71
$-f_o + 3\frac{f_s}{4}$	-72.62	-72.71

Table 4.2: Simulation results of a 4 TI-ADC with bandwidth mismatch for an input signal $x(t) = 1.5\ \sin(2\pi f_o t)$, $f_o = 146.29$ MHz and $f_s = 320$ MHz. $[f_{co}\ f_{c1}\ f_{c2}\ f_{c3}] = [4.01\ 3.94\ 4.12\ 3.86]$ GHz.

$$\text{where} \quad Z_a(k) = Z_a(M-k) = \left| \sum_{m=0}^{M-1} \zeta^{-mk} \epsilon_m^a \right|^2 \tag{4.11}$$

The Spurious-Free-Dynamic Range (SFDR) of a TI-ADC is derived from (4.10) as :

$$SFDR = \min_{k=1,\ldots,\lfloor \frac{M}{2} \rfloor} SSR_k \tag{4.12}$$

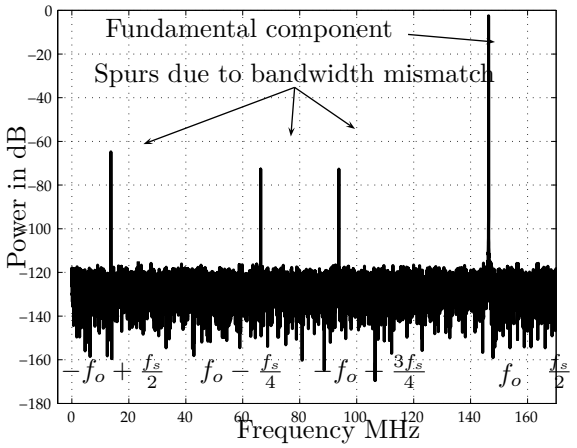


Figure 4.6: Output spectrum of a 4 TI-ADCs with bandwidth mismatch for an input signal $x(t) = 1.5\ \sin(2\pi f_o t)$, $f_o = 146.29$ MHz and $f_s = 320$ MHz. $[f_{co}\ f_{c1}\ f_{c2}\ f_{c3}] = [4.07\ 3.80\ 3.98\ 3.85]$ GHz.

4.6 Probabilistic Description of Mismatches

4.6.1 Motivation

Device mismatches are caused by Process, Voltage and Tempretaure (PVT) variations and are commonly described by random variables normally distributed.

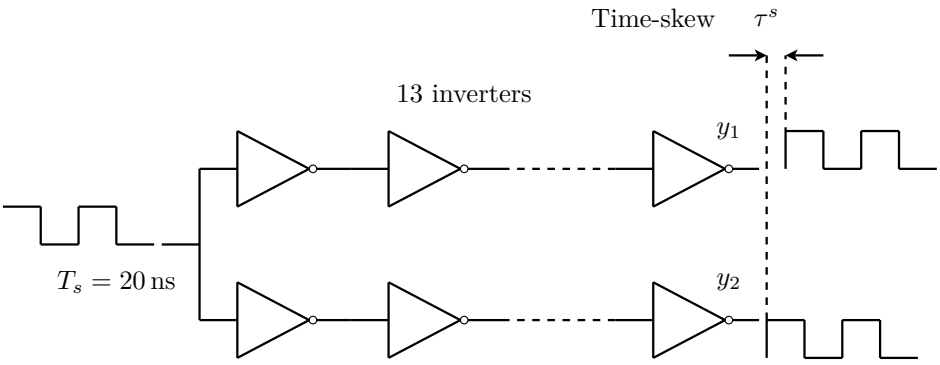


Figure 4.7: Two chains of 13 inverters clocked at 50 MHz in 65 nm process

For example, Fig. 4.7 represents the clock distribution circuitry for 2 channels in 65 nm process. The original clock has a frequency of 50 MHz. Ideally, the outputs y_1 and y_2 should be identical since both channels have the same input. However, due to PVT variations this is not the case. Fig. 4.8 shows the distribution of delay mismatch between the two outputs obtained by running Monte Carlo simulations. The standard deviation of this time-skew mismatch is approximately 30 ps which represents 0.15 % of the clock period.

To understand it, let's consider a simple rough model of the CMOS inverter delay of Fig. 4.9 [61]:

$$t_{inv} = \frac{V_{dd} C_{load}}{I} = \frac{C_{load} V_{dd}}{\mu C_{ox} \frac{W}{L} (V_{dd} - V_T)} \tag{4.13}$$

This delay is function of the inverter load C_{load}, supply voltage V_{DD} and the transistor threshold voltage V_T which depends also of the temperature T. Any variation of these parameters from one inverter to another will result in delay mismatch at the output of the chain. Thus, using this rough model and considering N_{inv} inverters per branch, time-skew mismatch can be modeled by a gaussian variable $\delta t_{inv} \sim \mathcal{N}\left(0, \sqrt{N_{inv}} \frac{\sigma_{inv}}{T_s}\right)$ with σ_{inv} given by:

$$\sigma_{inv} \simeq t_{inv} \left(\frac{\Delta C_{load}}{C_{load}} - \frac{V_T}{V_{DD}(V_{DD} - V_T)} \Delta V_{DD} + \frac{1}{V_{DD} - V_T} \Delta V_T \right) \tag{4.14}$$

More generally we are going to consider that gain, time-skew, bandwidth and offset mismatches are random. Since these mismatches are responsible for channel mismatch errors, it is convenient to rely on a probabilistic characterization of the gain, offset, time-skew and bandwidth mismatches. As a consequence, the SFDR is modeled by a statistical distribution related to the standard deviation σ_a of a specific mismatch as in [57].

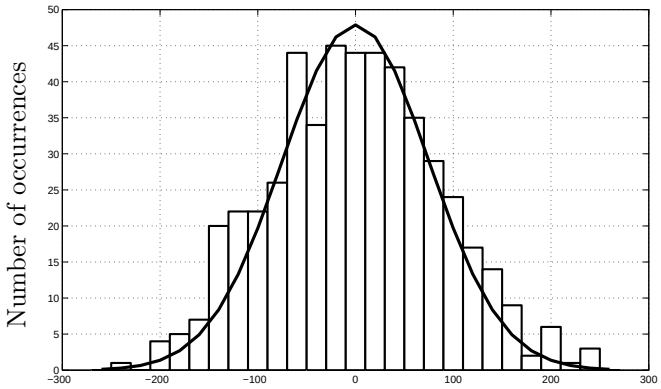


Figure 4.8: Time-skew mismatch between two channel of 13 inverters in 65 nm process at a frequency of 50 MHz

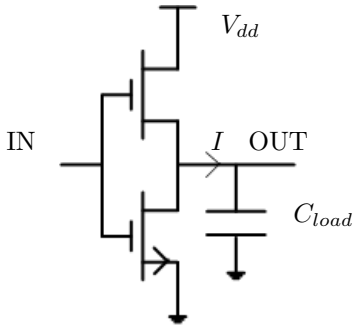


Figure 4.9: Typical CMOS inverter

Readily, $\epsilon_m^a = \sigma_a \xi_m^a$ with $\xi_m^a \sim \mathcal{N}(0,1)$. (4.11) becomes :

$$Z_a(k) = \sigma_a^2 \left| \sum_{m=0}^{M-1} \zeta^{-mk} \xi_m^a \right|^2 = \sigma_a^2 S_a(k) \tag{4.15}$$

4.6.2 Probability Density Function of SFDR and THD

In the previous sections, we saw that channel mismatches can be described by random variables normally distributed. As the SFDR is also function of these mismatches, it is important to rely on a probabilistic description of the SFDR. In the following analysis, we are going to consider that mismatch a is predominant where a stands for either gain, time-skew, bandwidth of offset mismatch. In section 4.6.2, we will provide a general framework to describe the PDF of all mismatches including bandwidth. Therefore this will extends the work presented in [57] where bandwidth mismatch was not considered.

In practice, the analog designer is not only interested in PDF of the SFDR.

Indeed he also wants to establish safe margins for his design by looking for the probability that the SFDR is greater than the targeted value. Therefore using an analogy to the usual yield, we will introduce a robustness criterion $SFDR(\eta)$ which states that the SFDR remains higher than the threshold value with a probability η. Therefore this extends the work presented in [58][56] where only mean value were determined.

Probability Density Function of SFDR

The SFDR is defined as the ratio of the fundamental signal to the worst spurious. So we have:

$$\begin{aligned} SFDR &= 10\log_{10}\left[\frac{P_{signal}}{P_{max}^{spur}}\right] \\ &= 10\log_{10}\left[\max_{k=1,\dots,\lfloor\frac{M}{2}\rfloor} SSR_k\right] \\ &= 10\log_{10}\left[\frac{M^2}{\sigma_a^2 C_a \max_{k=1,\dots,\lfloor\frac{M}{2}\rfloor} S_a(k)}\right] \\ &= 10\log_{10}\left[\frac{M^2}{\sigma_a^2 C_a S_{max}}\right] \\ &= F(S_{max}) \end{aligned} \tag{4.16}$$

In (4.16), the function F is monotone. Let's call f_{max} the PDF of S_{max} which is calculated in Appendix. C.2. Using (4.16) and a variable change, we find that for M odd:

$$\begin{aligned} p(SFDR) &= \frac{1}{F'\Big[F^{-1}(SFDR)\Big]} f_{max}\Big(F^{-1}(SFDR)\Big) \\ &= \alpha\frac{M-1}{2} s e^{-s}(1-e^{-s})^{\frac{M-3}{2}}\Big|_{s=\frac{M}{\sigma_a^2 C_a}10^{-\frac{SFDR}{10}}} \end{aligned} \tag{4.17}$$

With $\alpha = -\frac{\log 10}{10}$ and $C_a = |Q_a|^2$ is a frequency dependent coefficient depending of the mismatch analyzed. For M even, the PDF of the SFDR is given by:

$$p(SFDR) = \alpha s(1-e^{-s})^{\frac{M-4}{2}}\Big[\frac{M-2}{2}e^{-s}erf(\frac{s}{2}) + \frac{1}{\sqrt{2\pi s}}e^{-s}(1-e^{-\frac{s}{2}})\Big]\Big|_{s=\frac{M}{\sigma_a^2 C_a}10^{-\frac{SFDR}{10}}} \tag{4.18}$$

With erf the Gauss error function.

Fig. 4.10, 4.11 and 4.12 compare the above PDF with simulation results of a two-TIADCs with gain, time-skew and bandwidth mismatches. We can notice that the statistical distribution of SFDR matches pretty well with the analytical PDF calculated above.

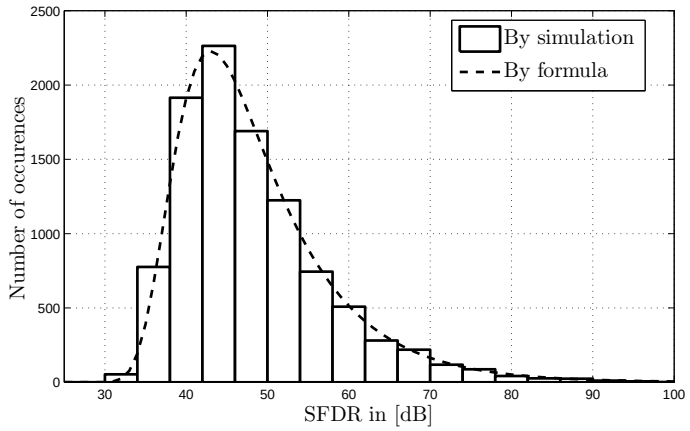


Figure 4.10: PDF of a two-channel TI-ADCs with a gain mismatch of 1 % and with a input sinusoid of amplitude 1 V

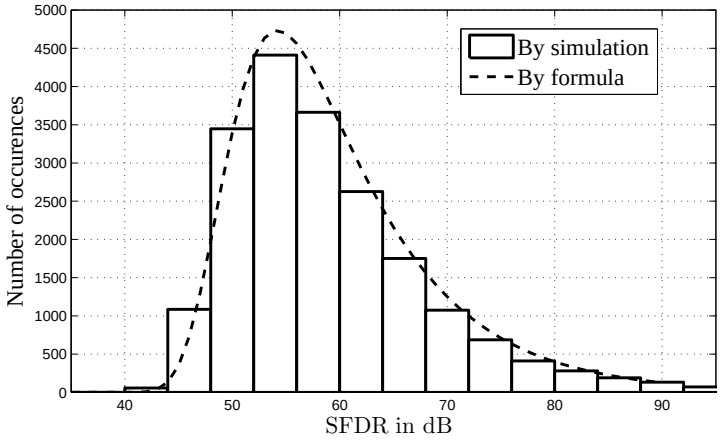


Figure 4.11: PDF of a two-channel TI-ADCs with a time-skew mismatch of 1 %, a sampling frequency of 320 MHz and with a input sinusoid of amplitude 1 V and a frequency of 137 MHz

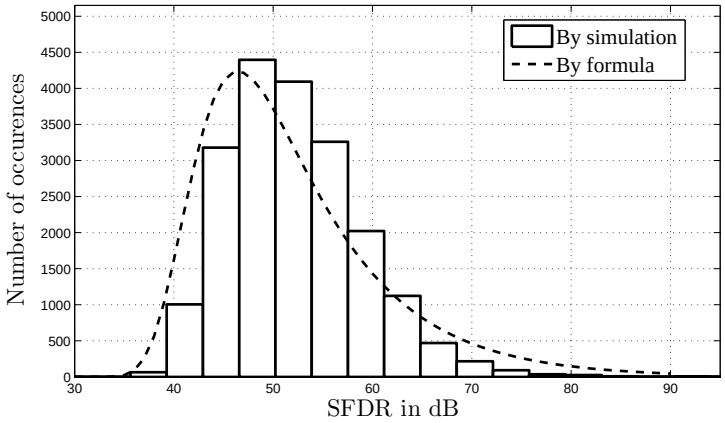


Figure 4.12: PDF of a two-channel TI-ADCs with a bandwidth mismatch of 1 %, a sampling frequency of 320 MHz and with a input sinusoid of amplitude 1 V, a frequency of 137 MHz and a cutoff frequency of 160 MHz

4.6.3 Cumulative Density Function of SFDR

As the SFDR is function of the random variables $S_a(k)$, it is also a random variable whose dispersion depends on standard deviation of mismatches σ_a. Therefore using an analogy to the usual *yield*, we introduce the robustness criterion $SFDR(\eta)$, that states the SFDR remains higher than this threshold value with a probability $1-\eta$. For M odd :

$$\begin{aligned}\eta &= P\Big(SFDR < SFDR(\eta)\Big) \\ &= 1 - \prod_{k=1}^{\frac{M-1}{2}} P\left(\frac{2}{M} S_a(k) < \frac{2M}{\sigma_a^2\, C_a\, SFDR(\eta)}\right) \\ \eta &= 1 - \left[F_2\left(\frac{2M}{\sigma_a^2\, C_a\, SFDR(\eta)}\right)\right]^{\frac{M-1}{2}}\end{aligned} \tag{4.19}$$

$F_2(s) = 1 - e^{-\frac{s}{2}}$ is the Cumulative Density Function of the two degrees of freedom Chi-square variables $\frac{2}{M} S_a(k)$, which are independent each other for $k \in \{1, ..., \lfloor \frac{M}{2} \rfloor\}$ [57].

Inverting (4.19) with respect to $SFDR(\eta)$ gives :

$$SFDR_{dB}(\eta) = -10 \log_{10} \frac{C_a \sigma_a^2}{M} - 10 \log_{10} K(\eta, M) \tag{4.20}$$

With $K(\eta, M) \simeq \ln\left(\frac{M-1}{2\eta}\right)$. (4.20) still holds numerically for M even. It states the SFDR remains higher than the threshold value $SFDR_{dB}(\eta)$ with the probability $1-\eta$ allowing to control the reliability of any mismatch calibration process which is not possible when only mean values are considered [56] [57] [58].

Fig. 4.13 shows the SFDR law as a function of M, the mismatch type and its standard deviation. Intrinsic channel error dispersions introduced by device mismatches make the TI-ADC architecture not compatible with applications requiring high SFDR in the order of 90 dB.

Indeed, based on [62], the bandwidth mismatch is about 35 000 ppm in 0.13 µm process. Assuming $f_s = 320\,\text{MHz}$, $f_0 = 146.29\,\text{MHz}$, $f_c = 4\,\text{GHz}$ and $M = 4$, results in $\frac{C_a^{1/2}\sigma_a}{\sqrt{M}} = 640$ ppm. Substituting this value in (4.20) and considering $\eta = 1\text{e}{-3}$ gives a low SFDR of 55 dB .

If we take again the time-skew mismatch of 0.15 % i.e 1 500 ppm obtained in section 4.6.1 with a clock frequency of 50 MHz in 65 nm. And if we assume an interleaving factor of 4 and a signal frequency of 10 MHz, we obtain a low SFDR of 50 dB.

These two examples emphasize the necessity of calibrating TI-ADC when high linearity is needed. Targetting 90 dB SFDR, the initial mismatches should be reduced at least to two orders of magnitude. Tab. 4.3 summarizes the matching requirements for the four type of mismatches.

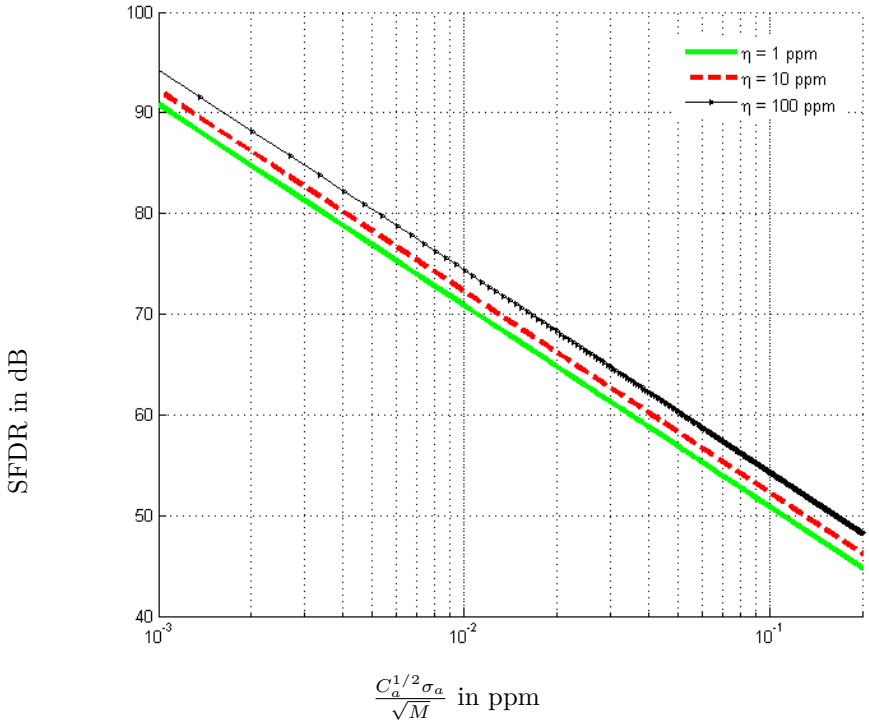


Figure 4.13: SFDR distortion law

-	Gain	Time-skew	Bandwidth	Offset
Accuracy in ppm	20	7	540	120

Table 4.3: Matching needed to reach the desired SFDR of 90 dB in 99.9 % of cases with a 4 TI-ADCs, an input signal $x(t) = 1.5\sin(2\pi f_0 t)$, $f_0 = 146.29$ MHz and $f_s = 320$ MHz

4.7 Integral and Differential Non-Linearities

The DNL of code m of a M TI-ADCs is the average of DNL of different sub-ADCs:

$$DNL(m) = \frac{1}{M} \sum_{i=0}^{N-1} DNL_i(m) \tag{4.21}$$

Where $DNL_i(m)$ is the DNL of code m of the i^{th} ADC. As seen in formula (2.38), the INL/DNL of each sub ADC can be written as a sum of capacitor mismatch which can be modeled by random variable normally distributed. Therefore the DNL of each sub-ADC can be considered also as gaussian variable and the variance of the DNL of the overall TI-ADC is:

$$\sigma^2(DNL) = \frac{1}{M^2} \sum_{i=0}^{M-1} \sigma^2(DNL_i) \tag{4.22}$$

(4.22) shows that DNL errors of the overall TI-ADCs is lower than the DNL errors of each of the M sub-ADCs. The same result can be obtained with INL.

4.8 Conclusion

In this chapter we have derived a deterministic model describing simultaneously the gain, time-skew, bandwidth and offset mismatches in a TI-ADC. A probabilistic law linking the SFDR to mismatch requirements has been provided. Numerical results show that initial mismatches must be reduced significantly by two orders of magnitude to obtain substantial SFDR in the order of 90 dB. It leads to consider a fully digital calibration mechanism with two purposes: mismatch estimation and compensation for which the following problems are anticipated.

On the one hand, the complexity level of the estimation phase depends on prior knowledge on the input signal, whether these are known sinusoids or unknown band-limited signals with known power spectral density. Since it works sporadically and requires flexibility, this block is preferably implemented in software.

On the other hand, the compensation phase is obvious for gain and offset, whereas the frequency dependency of time-skew and bandwidth makes the associated filtering problem challenging. This blocks can be implemented in hardware as it works at the sampling rate and is settled once and for all.

The next chapter will focus on the digital calibration algorithm we propose to cancel the effects of channel mismatch errors.

Chapter 5

Proposed Digital Calibration Scheme

5.1 Introduction and State of art

In the previous chapters, a lot of effort has been made to model a ADCs. In chapter 2, the differential equations governing the dynamics of non ideal S/H circuits were derived and then solved. This substantial effort of modeling has allowed us to identify precisely the analog variables which were responsible for distortion. Chapter 3 has completed this analysis with the statistical description of noise in ADCs. Then in chapter 4, this model of a single ADC has been extended to the case of time interleaved architecture in function of gain, time-skew and bandwidth mismatches all together. We demonstrated that these discrepancies between sub ADCs lead to the degradation of dynamic performances such as SFDR and should be mitigated.

There are two possible ways to deal with channel mismatches. The first is to complexify the analog circuit design of the ADC in order to reduce the magnitude of the original mismatches at the cost of more power consumption and area. But this increases the time-to-market of the ADC. The second solution is to alleviate the design and to correct the errors with a calibration technique as we are going to do in this chapter. Depending of the nature of the input signal, calibration techniques can be classified into foreground and background.

Foreground techniques interrupt the normal ADC operation by injecting a known signal to perform calibration and they are not suitable for applications such as communications where the ADC should be always running. In background calibration techniques, the ADC operation continues when the calibration is being performed. The mismatches are continuously estimated and corrected. These techniques can be subdivided into semiblind and blind. Semiblind background calibration techniques combine the input signal with a test signal that will be used for the calibration [5]. In blind background calibration techniques, no test signal is used [6]. Blind background calibration techniques are the most difficult to design because they should track and adjust to the changing operation conditions of ADCs in demanding environments with rapidly

changing temperatures. In addition, they should work with no informations or with little a priori informations on the input signal such as statistics.

When the calibration uses a feedback to the analog front-end of the ADC, it is a mixed signal calibration [7][8]. When it is done entirely in the digital domain, the calibration is said to be fully digital [9][10][5]. Mixed signal calibration techniques are popular in current TI-ADCs chips but fully digital calibrations are more and more desired because they require no custom redesign of the analog front-end of the ADC [8].

Several works have been done on correcting the gain, time-skew and offset mismatches in TI-ADCs [10] [7] [13] [6] [14] [15] [16], but little work has been done on bandwidth mismatches. In [5] [17] [18], some bandwidth mismatch calibrations are proposed for two channels ADCs but they dont take into account the time-skew and gain mismatch. Indeed bandwidth mismatch is frequency dependent and is likely to combine constructively or destructively with time-skew and gain mismatch as demonstrated in chapter 4. Therefore these three mismatches should be treated jointly for more optimality. In [19], a calibration method for gain, time-skew and bandwidth mismatches using a feedforward equalizer is proposed, but the algorithm takes a long time to converge. In [20], a calibration of frequency response mismatch is proposed by modeling transfer function as polynomial with variable order differentiators and coefficients. This was done for only two channels and the decomposition in differentiator filters is more accurate for time-skew mismatch correction than for bandwidth mismatch.

To overcome these limitations of the state of art, we propose in this chapter a fully digital adaptive and blind background calibration of channel mismatches in TI-ADCs. The contribution of this chapter is:

- a simultaneously and blind estimation of gain, time-skew and bandwidth mismatches. The estimation of all mismatches is achieved with less than 10-K samples with an accuracy of 98%, 94% and 88% for gain, time-skew and bandwidth.
- a fully digital correction of gain, time-skew, bandwidth and offset mismatches. The linearity of the TI-ADCs can be improved by almost 40 dB.
- a calibration that can be applied to any interleaved factor.

5.2 Estimation of channel mismatches

Before correcting channel mismatch errors, the mismatches should be estimated. To do it, we choose channel 0 as reference channel and then apply a low-pass and a fractional delay filter to this reference channel to obtain some samples that will be compared to the output of non reference channels allowing the identification of channel mismatches. This is done blindly i.e without a test signal so that the ADC can keep running while estimation is being processed. We only assume that the input signal is bandlimited and Wide Sense Stationary (WSS) which is relevant for a communication system.

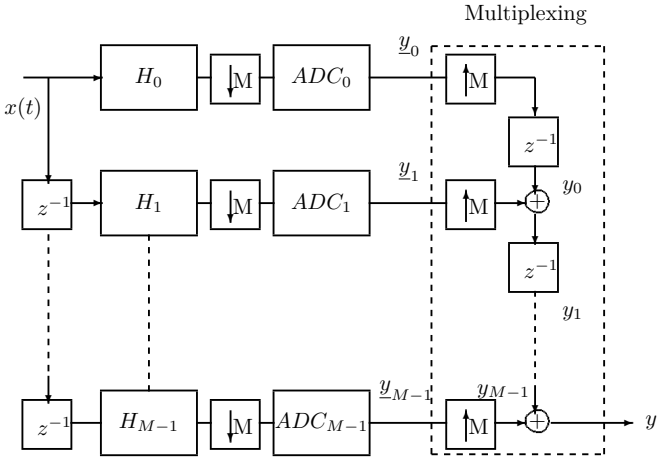


Figure 5.1: Time-Interleaved ADCs

Let's consider the M TI-ADC of Fig. 5.1 and an unknown input signal $x(t)$ bandlimited and WSS. Let's call $\underline{y}_m$ the signal just after the ADC of channel m as represented on Fig. 5.1:

$$\begin{aligned}\underline{y}_m[k] &= (h_m \star x)(kMT_s - mT_s) + w_m[k] \\ &= x_m[k] + w_m[k]\end{aligned} \tag{5.1}$$

w_m is the total noise on channel m which is assumed to be white and gaussian [63]. The DTFT $X_m(f)$ of the signal x_m in (5.1) consists of replicas of the fundamental signal at shifted frequencies $f - \frac{kf_s}{M}$ as shown on Fig. 5.2:

$$X_m(f) = \sum_k^{\infty} \exp(-j2\pi(.)mT_s)\Big[1 + \delta H_m(.)\Big] H_0(.)X(.)\Big|_{f-\frac{kf_s}{M}} \tag{5.2}$$

If we apply a lowpass filter h_{LP} to $\underline{y}_m$ we can obtain a signal z_m which is alias free as shown on Fig. 5.2:

$$z_m[k] = \underbrace{(h_{LP} \star x_m)(kT_s)}_{\underline{x}_m[k]} + \underbrace{(h_{LP} \star w_m)(kT_s)}_{b_m[k]} \tag{5.3}$$

The noise b_m in (5.3) is gaussian and bandlimited and the DTFT $\underline{X}_m$ of the signal $\underline{x}_m$ in (5.3) has no alias (Fig. 5.2):

$$\underline{X}_m(f) = \exp(-j2\pi f m T_s)\Big[1 + \delta H_m(f)\Big] H_0(f)X(f) \tag{5.4}$$

Let's call h_{dl}^m, the delay filter whose transfer function is $H_{dl}^m(f) = \exp(-j2\pi f m T_s)$. h_{dl}^m has a rational fractional delay $\frac{m}{M}$ because on each channel the sampling period is MT_s and $m < M$. Using (5.3) and (5.4), let's define the signal Δz_m by:

$$\begin{aligned}\Delta z_m[k] &= z_m[k] - (h_{dl}^m \star z_0)[k] \\ &= \Big(h_{LP} \star h_{dl}^m \star \delta h_m \star h_0 \star x\Big)[k] \\ &\quad + b_m[k] - (h_{dl}^m \star b_0)[k]\end{aligned} \tag{5.5}$$

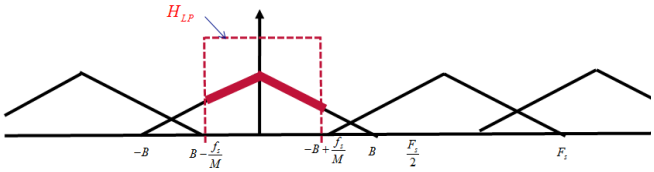


Figure 5.2: Spectrum of the m^{th} ADC output

Equations (5.6) and (5.7) remind the expression of the transfer function mismatch in function of the gain, time-skew and bandwidth mismatches and also the expressions of filters Q_0, Q_1 and Q_2.

$$\delta H_m(f) = \delta g_m Q_0(f) + \delta t_m Q_1(f) + \delta f_{c_m} Q_2(f) \tag{5.6}$$

$$\text{where} \begin{cases} Q_0(f) = 1 \\ Q_1(f) = j2\pi f T_s \\ Q_2(f) = \frac{j\frac{f}{f_c}}{1+j\frac{f}{f_c}} \end{cases} \tag{5.7}$$

Using (5.5) and (5.6) we can estimate adaptively the gain, the time-skew and the bandwidth mismatches of each channel m with the structure of Fig. 5.3.
Let's denote $\boldsymbol{\theta}_m = [\delta g_m \ \delta\tau_m \ \delta f_{c_m}]^t$ the mismatch vector of channel m, $\hat{\boldsymbol{\theta}}$ the estimated value of mismatches, $\Delta\hat{z}_m = (\delta\hat{h}_m \star z_0)$ with $\delta\hat{h}_m$ the estimated transfer function mismatch calculated through (5.6). Let's $e(\boldsymbol{\theta}, k)$ be the error function used to adapt $\hat{\boldsymbol{\theta}}$ as shown Fig. 5.3:

$$\begin{aligned} e(\boldsymbol{\theta}, k) &= \Delta z_m[k] - \Delta\hat{z_m}[k] \\ &= \Big(h_{LP} \star h_{dl}^m \star \{\delta h_m - \delta\hat{h}_m\} \star h_0 \star x\Big)[k] \\ &\quad - b_m[k] - (h_{dl}^m \star b_0)[k] \end{aligned} \tag{5.8}$$

Applying the gradient descent algorithm, we find the iteration to use:

$$\hat{\boldsymbol{\theta}}_m(k+1) = \hat{\boldsymbol{\theta}}_m(k+1) - \boldsymbol{\mu} e(\hat{\boldsymbol{\theta}}_m, k) \bigtriangledown_{\hat{\boldsymbol{\theta}}_m} e(\hat{\boldsymbol{\theta}}_m, k) \tag{5.9}$$

Where $\boldsymbol{\mu}$ is the iteration step vector and $\bigtriangledown_{\hat{\boldsymbol{\theta}}_m} e(\hat{\boldsymbol{\theta}}_m, k)$ is the gradient of the error function given by:

$$\bigtriangledown_{\hat{\boldsymbol{\theta}}_m} e(\hat{\boldsymbol{\theta}}_m, k) = \Big[(Q_0 \star z_0)[k] \quad (Q_1 \star z_0)[k] \quad (Q_2 \star z_0)[k]\Big]^t \tag{5.10}$$

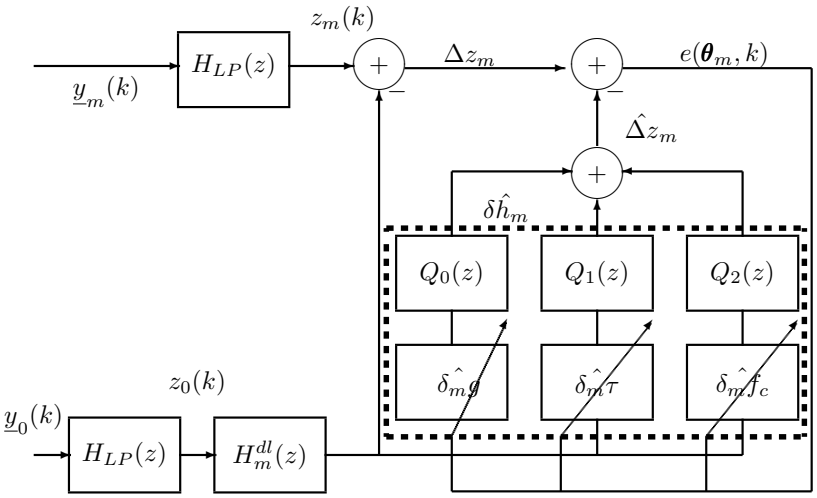


Figure 5.3: Adaptive filtering structure

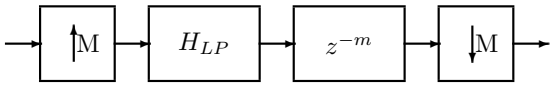


Figure 5.4: Multirate fractional rational delay m/M

5.3 Compensation of channel mismatch errors

Once mismatches have been estimated, they should be corrected. To do it, we use a matrix approach first introduced in [5] for the compensation of only bandwidth mismatch in the specific case of a 2-channels TI-ADCs that we extend for gain, time-skew and bandwidth mismatches and for any interleaved factor M. Let's use the following notations:

- $\mathbf{Y} = [Y_0(f)...Y_{M-1}(f)]^t$ with $Y_m(f)$ the output of channel m of the TI-ADCs of Fig. 5.1 in the frequency domain.
- $\mathbf{V}$ the MxM unitary matrix given by $\boldsymbol{V}^{mk} = \zeta^{-mk}$
- $\boldsymbol{\Delta}$ the MxM matrix given by $\boldsymbol{\Delta}^{mk} = \zeta^{-mk}\delta H_m(.)\Big|_{f-\frac{kf_s}{M}}$
- $\mathbf{D}$ the Mx1 vector given by: $\mathbf{D}_k = H_0(.)X(.)\Big|_{f-\frac{kf_s}{M}}$.

The DTFT Y_m of the y_m of channel m can be written as:

$$Y_m(f) = \frac{1}{M}\Lambda(f)\sum_{k=0}^{M-1}\zeta^{-mk}\Big[1+\delta H_m(.)\Big]H_0(.)X(.)\Big|_{f-\frac{kf_s}{M}} \tag{5.11}$$

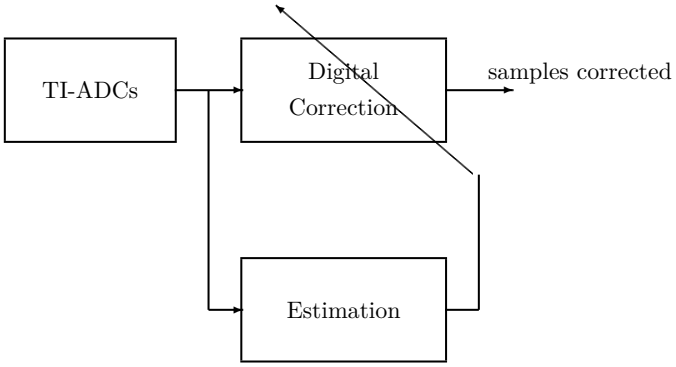


Figure 5.5: Principle of the fully digital calibration.

Based on (5.11), the output vector of the TI-ADCs is:

$$\mathbf{Y} = \frac{\Lambda}{M}[\mathbf{V} + \mathbf{\Delta}]\mathbf{D} = \frac{\Lambda}{M}\Big[\mathbf{I}_M + \frac{1}{M}\mathbf{\Delta}\mathbf{V}^\dagger\Big]\mathbf{V}\mathbf{D} \tag{5.12}$$

In (5.12), $\mathbf{I}_M$ is the MxM identity matrix and $\mathbf{V}^\dagger$ is the hermitian transpose of $\mathbf{V}$. In the ideal case, there are no mismatches, $\mathbf{\Delta} = \mathbf{0}$, and the ideal output is:

$$\mathbf{Y}_{ideal} = \frac{\Lambda}{M}\mathbf{V}\mathbf{D} \tag{5.13}$$

In practice, device mismatches are inherent to any manufacturing process and result in channel mismatch errors ($\mathbf{\Delta} \neq \mathbf{0}$). The actual output can be written as:

$$\mathbf{Y} = \Big[\mathbf{I}_M + \frac{1}{M}\mathbf{\Delta}\mathbf{V}^\dagger\Big]\mathbf{Y}_{ideal} \tag{5.14}$$

To calibrate the TI-ADCs, the effect of mismatches should be canceled in (5.14) such that the corrected output $\mathbf{Y}_c$ becomes almost equal to $\mathbf{Y}_{ideal}$:

$$\mathbf{Y}_c = \Big[\mathbf{I}_M + \frac{1}{M}\mathbf{\Delta}\mathbf{V}^\dagger\Big]^{-1}\mathbf{Y} \simeq \Big[\mathbf{I}_M - \frac{1}{M}\mathbf{\Delta}\mathbf{V}^\dagger\Big]\mathbf{Y} = \mathbf{F}\mathbf{Y} \tag{5.15}$$

In (5.15), $\mathbf{F}$ contains the set of filters that should be applied to the output of the TI-ADCs for calibration. The approximation of (5.15) is obtained using the Taylor expansion for matrix since the mismatches in $\mathbf{\Delta}$ are small.

5.3.1 Particular case M=2

For the particular case of two channels, the matrix $\mathbf{F}$ of compensation filters obtained from (5.15) are given in (5.16). The associated compensation structure is shown on Fig. 5.6. One should notice that only channel 1 is calibrated since channel 0 is the reference channel.

$$\begin{cases} \mathbf{F}_{00}(f) = 1 \qquad \mathbf{F}_{01}(f) = 0 \\ \mathbf{F}_{10}(f) = -\frac{1}{2}\Big[\delta H_1(f) - \delta H_1(f - \frac{f_s}{2})\Big] \\ \mathbf{F}_{11}(f) = 1 - \frac{1}{2}\Big[\delta H_1(f) + \delta H_1(f - \frac{f_s}{2})\Big] \end{cases} \tag{5.16}$$

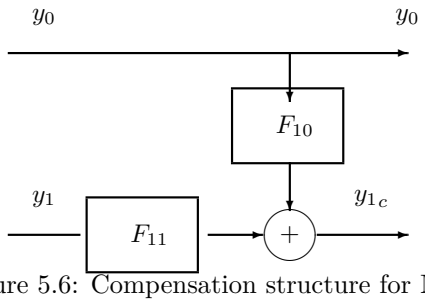


Figure 5.6: Compensation structure for M=2

With this particular case of two channels, we are going to analyze analytically and by simulation the performance of this technique by evaluating the spur power before and after calibration. We will consider an input signal consisting of two unit sinusoids at frequencies 40 MHz and 80 MHz sampled at $f_s = 320$ MHz. The spectrum will be observed in the interval $[0\ f_s]$.

5.3.2 Before calibration

The DTFT Y_0 and Y_1 of the two outputs y_0 and y_1 are given by:

$$\begin{aligned} Y_0(f) &= \frac{1}{2}\Big[H_0(f)X(f) + H_0(f')X(f')\Big] \\ Y_1(f) &= \frac{1}{2}\Big[\Big\{1+\delta H(f)\Big\}H_0(f)X(f) - \{1+\delta H(f')\}\,H_0\,(f')\,X\,(f')\Big] \end{aligned} \tag{5.17}$$

To simplify notations in (5.17), $f' = f - \frac{f_s}{2}$. The output of the TI-ADCs is obtained by summing the two outputs:

$$\begin{aligned} Y(f) &= Y_0(f) + Y_1(f) \\ &= \underbrace{\Big[1+\frac{1}{2}\delta H(f)\Big]H_0(f)X(f)}_{Fundamental} - \underbrace{\frac{1}{2}\delta H(f')H_0(f')X(f')}_{spurious\ at\ f'=f-\frac{f_s}{2}} \end{aligned} \tag{5.18}$$

From (5.18), we see that channel mismatches create a disturbing component which occurs at shifted frequency $f - \frac{f_s}{2}$. When there no mismatches, $\delta H(f) = 0$ and the ideal output is:

$$Y(f) = H_0(f)X(f) \tag{5.19}$$

5.3.3 After calibration

To reduce the magnitude of the disturbing component in (5.18), Y_0 will be considered as the reference channel and Y_1 will be calibrated. Let's apply the filters F_{10} and F_{11} respectively to Y_0 and Y_1 in order to obtain the calibrated signal Y_{1cal}:

$$Y_{1cal}(f) = F_{10}(f)Y_0(f) + F_{11}Y_1(f) \tag{5.20}$$

With F_{10} and F_{11} defined in (5.16). Now the serial output Y can be calculated again as:

$$\begin{aligned} Y(f) &= Y_0(f) + Y_{1cal}(f) \\ &= \overbrace{\Big[1 - \frac{1}{4}\delta H(f)\Big(\delta H(f) + \delta H(f')\Big)\Big] H_0(f)X(f)}^{Fundamental} \\ &+ \underbrace{\frac{1}{4}\Big[\delta H(f')\Big(\delta H(f) + \delta H(f')\Big)\Big] H_0(f')X(f')}_{Spurious\ at\ f'=f-\frac{f_s}{2}} \end{aligned} \tag{5.21}$$

From (5.18) and (5.21), we see that the mismatches have been reduced from the first order to the second order. Now the actual output in (5.21) is close to the ideal output obtained in (5.19).

5.4 Simulation results with a two-channel ADCs

We consider WiFI IEEE 802.11.ac with a bandwidth up to 160 MHz implemented with an IQ architecture. Each base-band signal has 80 MHz bandwidth oversampled at 340 MHz with a 12-bits 2-TI-ADCs. The gain, the time-skew and the cutoff frequency of the first ADC is $G_0 = 1$, $\tau_0 = 1$ ps and $f_{c0} = \frac{f_s}{2}$. The gain, the time-skew and the bandwidth mismatch are respectively $\delta g_1 = 1\,\%$, $\delta\tau_1 = 1\,\%$ and $\delta f_{c1} = 2\,\%$. The estimation uses the adaptive filtering structure of Fig. 5.3 and the compensation uses the structure of Fig. 5.6.

Filters H_{LP}, H_m^{dl}, Q_0, Q_1, Q_2, F_{10} and F_{11} are implemented with FIRs. The impulse response of H_{LP} is obtained by applying a Kaiser window to its inverse DTFT. The fractional delay filter H_m^{dl} is obtained from H_{LP} with the multirate structure of Fig. 5.4 since it has a rational fractional delay [64]. Filters Q_0, Q_1 and Q_2 are designed with the sampling frequency technique since they should have a predefined magnitude and phase response specified by (5.7). F_{10} and F_{11} of (5.16) can be obtained from Q_0, Q_1 and Q_2 with (5.6). The iteration step vector is set to $\boldsymbol{\mu} = [1e{-}5 \;\; 3e{-}5 \;\; 5e{-}6]$ in order to have a good precision with a suitable convergence.

Tab. 5.1 presents simulation results before and after calibration with 2 unit sinusoids at frequencies 40 MHz and 80 MHz. For only gain mismatches, the SFDR is improved by 41 dB. For only time-skew mismatch the SFDR is improved by 40 dB and for bandwidth mismatch the SFDR is improved by 32 dB. Considering all mismatches, the SFDR can be improved by 40 dB and Fig. 5.8 shows the spectrum before and after calibration. Fig. 5.7 presents the convergence of the mismatch estimation with the number of samples. Globally we need 10K samples to estimate all mismatches. The estimation accuracy is 98%, 94% and 99% respectively for gain, time-skew and bandwidth mismatches.

Mismatch	Before calibration	After calibration	Improvement
Only gain	46	87	41
Only time-skew	43	83	40
Only bandwidth	48	80	32
All mismatches	38	80	42

Table 5.1: SFDR in dB for a 2-channel at 40 MHz and 80 MHz

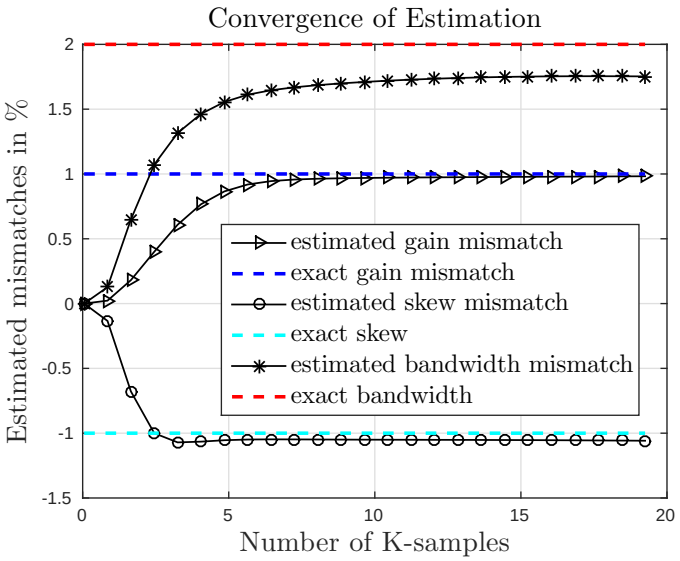


Figure 5.7: Simulation of the Convergence of mismatch estimation

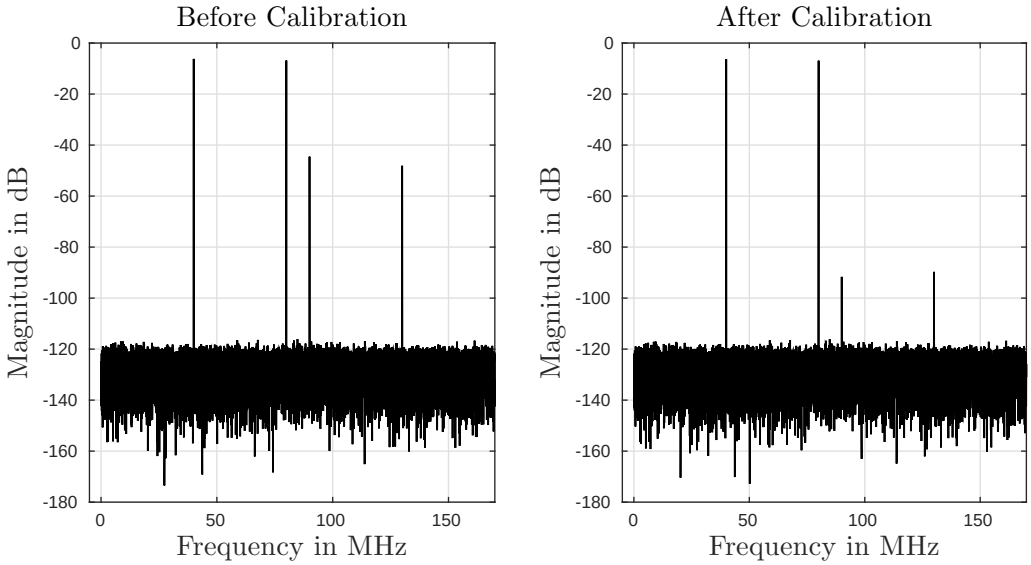


Figure 5.8: Simulation of the output spectrum before and after calibration

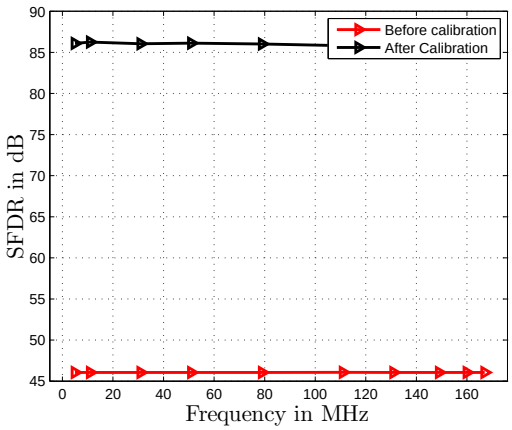


Figure 5.9: SFDR before and after correction with a two-channel TI-ADCs with 1% gain mismatch. The sampling frequency is 340 MHz.

5.5 Impact of the signal bandwidth

Now we can analyze the performances of the algorithm in function of the signal bandwidth. For this purpose we are going to do several simulations for different signal bandwidths ($[0\ \frac{f_s}{2}]$). The results are shown on Fig. 5.9, 5.10, 5.11 and 5.12 respectively for gain (1%), time-skew(-1%), bandwidth(2%) and all mismatches. We notice that for gain mismatches, the correction is constant with the signal bandwidth. That is normal because gain mismatch is independent of the frequency. Time-skew and bandwidth mismatches are frequency dependent and we see that the performances of the calibration decrease with the signal bandwidth and around the nyquist frequency an important loss appears. When all the mismatches are present the same effect is visible in function of the signal bandwidth. This phenomena has two explanations. One one hand it is due to the fact that when the bandwidth of the signal increases, the bandwidth where there are no alias decreases and so the estimation is done on a signal which is very slow. Because the signal is slow the errors due to timing mismatches (time-skew and bandwidth) are small and it becomes difficult to identify them. On the other hand the transfer function mismatch is a high pass filter as we can see on Fig. 5.13. When the input signal bandwidth increases, the transfer function mismatch incrases and the approximation of (5.15) becomes less accurate.

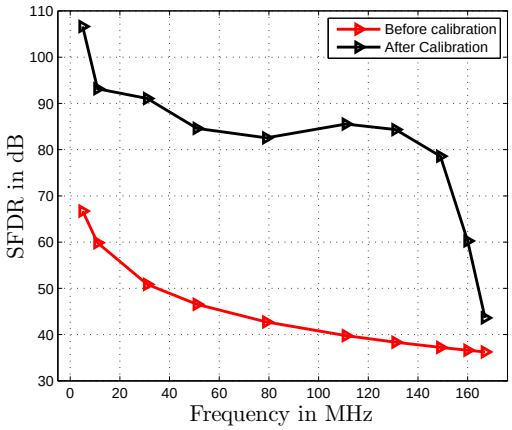


Figure 5.10: SFDR before and after correction with a two-channel TI-ADCs with -1% time-skew mismatch . The sampling frequency is 340 MHz.

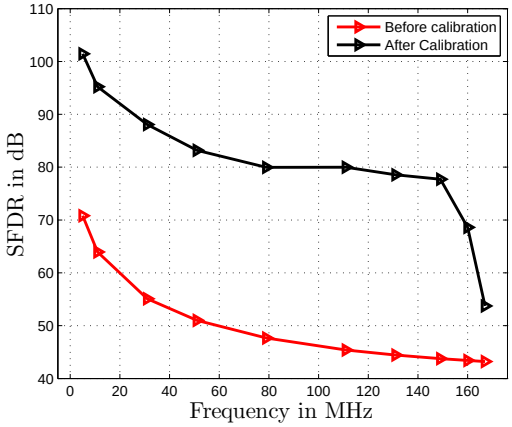


Figure 5.11: SFDR before and after correction with a two-channel TI-ADCs with 2% bandwidth mismatch. The sampling frequency is 340 MHz.

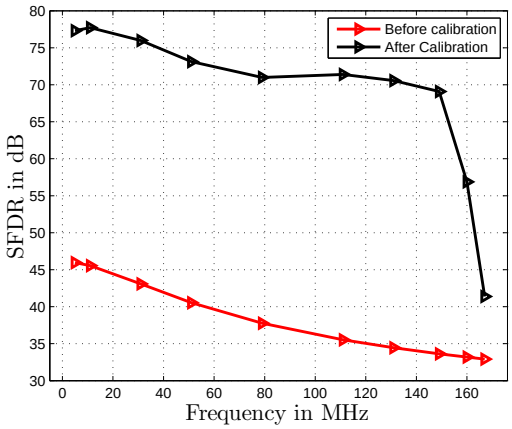


Figure 5.12: SFDR before and after correction with a two-channel TI-ADCs with 1% gain mismatch, -1% time-skew mismatch and 2% bandwidth mismatch. The sampling frequency is 340 MHz.

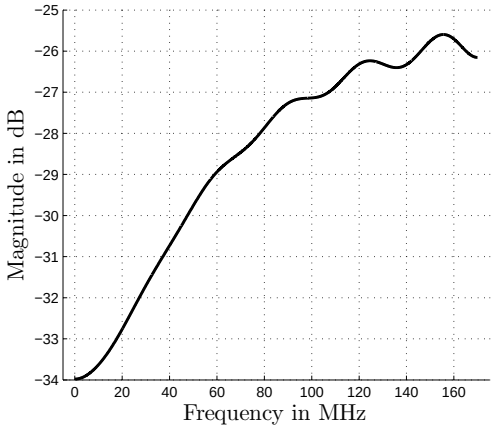


Figure 5.13: Transfer function mismatch with 1% gain mismatch, -1% time-skew mismatch and 2% bandwidth mismatch. The sampling frequency is 340 MHz.

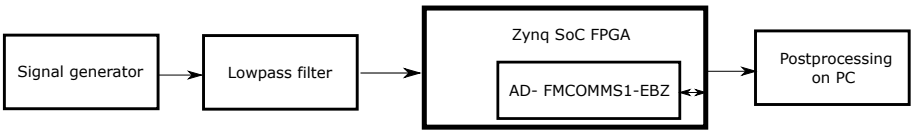


Figure 5.14: Test bench

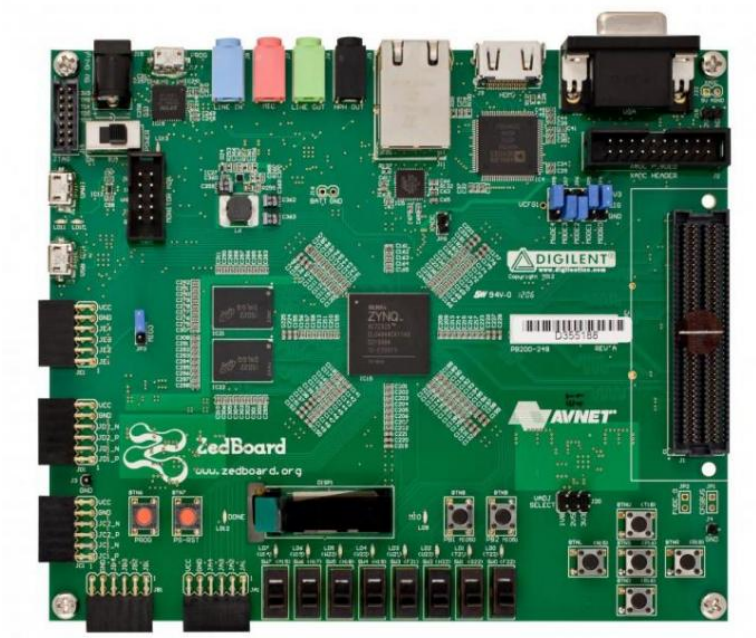


Figure 5.15: Xilinx Zynq 7000 All Programmable SoC zc702 Evaluation Kit

5.6 Measurement results on two-channel ADC board

Our technique was tested on a 14-bits two-channel ADCs AD9643 from Analog Devices. Fig. 5.14 shows the testbench used. The AD-FMCOMMS1-EBZ is an analog front end hardware platform that contains the two-channel ADCs AD9643 and it is connected to the Xilinx Zynq 7000 All Programmable SoC zc702 Evaluation Kit FPGA Platform. The output samples are collected on a computer where the spectrum is displayed. The FPGA platform Xilinx Zynq 7000 All Programmable SoC zc702 Evaluation Kit is shown on Fig. **??** and the AD-FMCOMMS1-EBZ is shown on Fig. 5.16 and 5.17.

Figure 5.16: AD-FMCOMMS1-EBZ

The input signal consists of a sinusoid of amplitude 0.35 V at frequency $f_0 = 31$ MHz. The estimation uses the adaptive filtering structure of Fig. 5.3 and the compensation uses the structure of Fig. 5.6.

Fig. 5.18(a) shows the output spectrum before calibration with a sampling frequency $f_s = 140$ MHz. Gain, time-skew and bandwidth mismatch errors create a spurious which occurs at frequency $f_{spur} = -f_0 + \frac{f_s}{2} = 39$ MHz and its magnitude is -64 dB. Offset mismatch creates a spurious of magnitude -64 dB at the DC component and a spurious of magnitude -57 dB at the nyquist frequency. The other spurious visible on the spectrum are inherent to the individual sub-ADCs we have used and are not due to interleaving.

A compensation of offset mismatch errors has been used in order to remove the DC component and the spurious at the nyquist frequency. For this purpose we have subtracted the mean amplitude from each sample. Fig. 5.18(b) shows the output spectrum after calibration with 51 taps compensation FIRs. The digital calibration method we propose is able to reduce the magnitude of the spurious due to gain, time-skew and bandwidth mismatches to -102 dB i.e an improvement of 38 dB. Fig. 5.19 shows the speed of the algorithm for the estimation of the gain, time-skew and bandwidth mismatches. We need practically 10-K samples for the estimation of the three mismatches which is faster than [5] and approximately the same speed obtained in [18].

Fig. 5.20 shows that the correction of channel mismatch errors increases

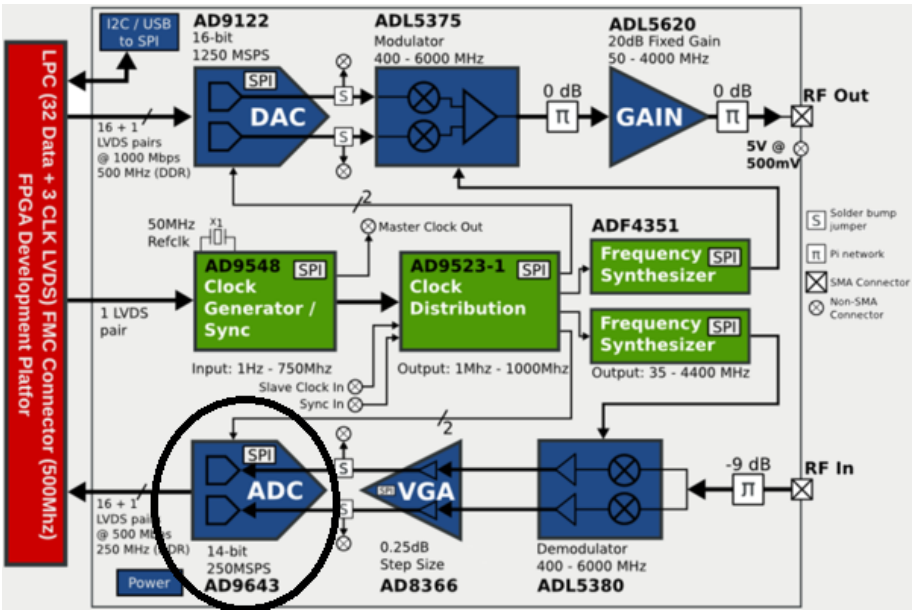


Figure 5.17: AD9643 on the AD-FMCOMMS1-EBZ functional blocks with the number of taps and the achievable limit is 38 dB obtained with 51 taps FIRs. However the dynamic performances are limited by the presence of some spurious which are due to the internal imperfections of the individual sub-ADCs we have used for this work.

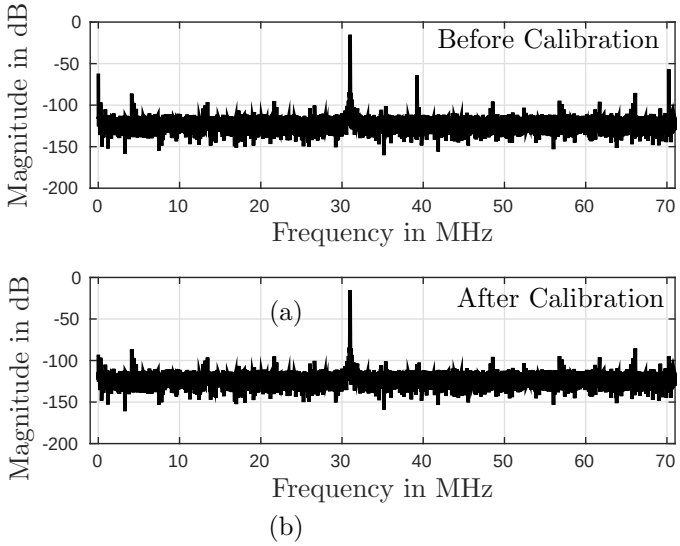


Figure 5.18: Measurement results of the output spectrum before and after calibration

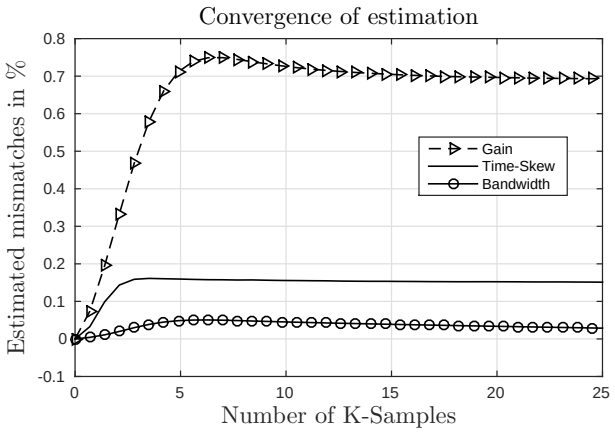


Figure 5.19: Measurement results of the mismatch estimation convergence

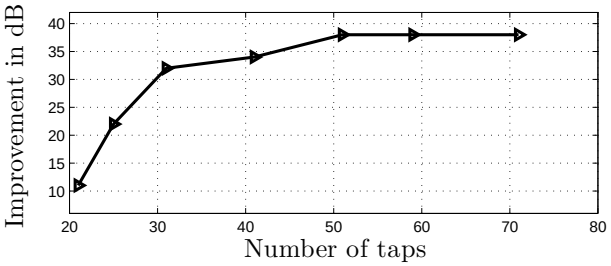


Figure 5.20: Reduction of spurious magnitude with the number of taps of correction filters

5.7 ASIC synthesis

The algorithm we presented in the previous section was described with a high level description tool such as Matlab/Simulink. In addition some simulations and measurements were carried out to verify the correctness of the algorithm. Now we are going to evaluate the area and the power consumption in the 65 nm CMOS process.

Fig. 5.21 shows the flow diagram of the ASIC synthesis. If the calibration algorithm presented above is designed in floating point data type, it will finally be implemented in a fixed-point architecture. In the previous section, when considering all mismatches for a two-channel TI-ADCs, the highest linearity was obtained with two 51 taps FIR filters as shown on Fig. 5.20 and Fig. 5.6. Therefore, for the ASIC synthesis we are going to consider two 51 taps FIR filters. However a lower number of taps can be chosen if less linearity is desired. In order to find the optimal word length data, we compute the SNDR of the compensated signal in function of the word length as shown on Fig. 5.22. From this analysis, we can see that the highest SNDR is 82 dB and is obtained with 15 bits. As a consequence, we will consider a word length of 15 bits. If less linearity is desired, a lower value can be chosen.

After that, the behavioral algorithm should be described in a Hardware Description Language such as Verilog for our case. Then some cosimulations must be done to verify that the algorithm works well in this HDL language. The last step consist in doing the logic synthesis in a process technology. In the 65 nm process, a power consumption of 10 mW and an area of 0.035 mm^2 were found.

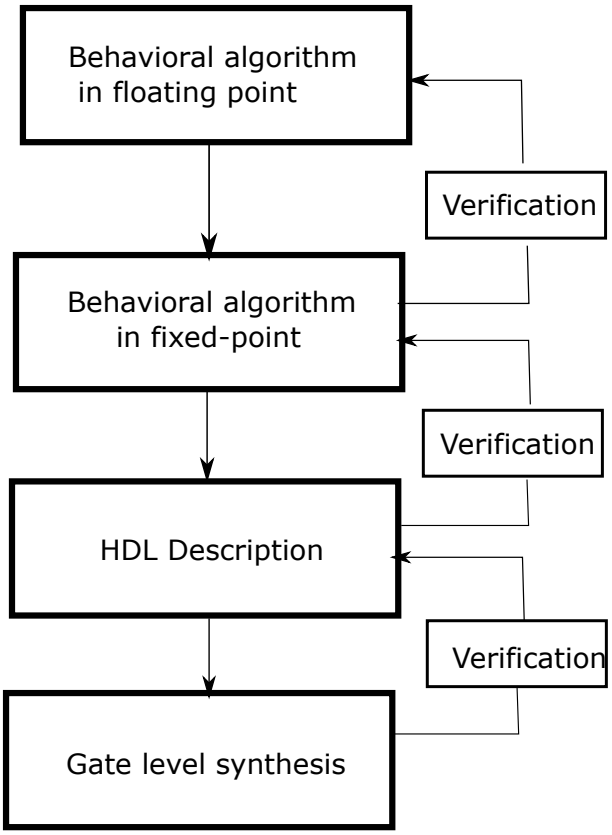


Figure 5.21: ASIC synthesis flow

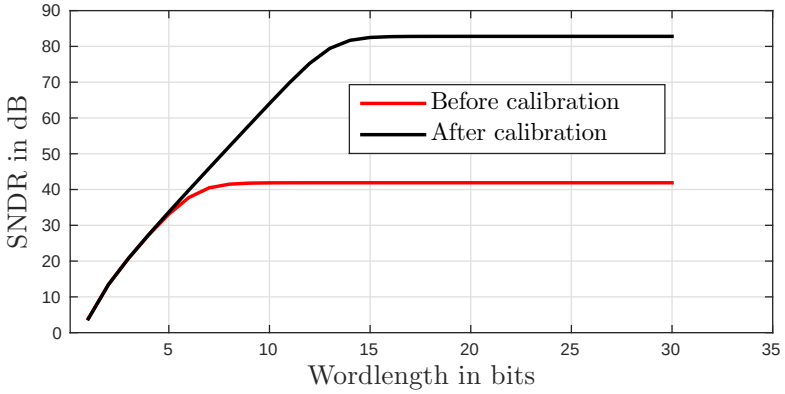


Figure 5.22: SNDR in function of the word length

5.8 Conclusion

In this chapter, we have proposed a fully digital calibration technique of gain, time-skew and bandwidth mismatches in TI-ADCs. The calibration is divided into two independent steps: estimation and compensation. The estimation uses a rational fractional delay filter combined to a lowpass filter to estimate adaptively the different mismatches. It is done blindly so that the ADC keeps running while estimation is being processed. The compensation is based on the development of a matrix approach to cancel the effects of channel mismatch errors. This technique can be applied to any interleaved factor with a high flexibility.

Measurements were carried out on a two-channel ADC board AD9643 from Analog Devices and results show the with only 10-K samples, gain, time-skew and bandwidth mismatches can be estimated. The linearity can be improved by almost 40 dB. With this algorithm we can expect a power consumption of 10 mW and a chip area of $0.035\,\mathrm{mm}^2$ for two 51 taps FIR filters and with 15 bits.

Chapter 6

Conclusion and Perspectives

In this work we designed a digital blind calibration algorithm which corrects all channel mismatches in Time-Interleaved ADCs. First a considerable modeling effort of a single ADC was made to derive the differential equations of the circuit and to find the analog variables responsible for distortion. For example in chapter 2, the gain, time-skew, bandwidth and offset of a single ADC were studied. We also made a deterministic and statistical analysis of harmonic distortion in differential bootstrapped S/H circuits. We saw that mismatches between the p-channel and n-channel must be lower than 1 % to have substantial second harmonic distortion in the order of 100 dB. In chapter 3, this model of a single ADC was completed with the statistical description of noises in term of probability density function and power spectral density.

Then later in chapter 4, this model of a single ADC was used to derive closed form equations for TI-ADCs. A general deterministic mismatch model including the bandwidth was provided. The transfer function mismatch of a given channel was defined and related to its gain, time-skew and bandwidth mismatches. In addition to this deterministic model, when considering random character of manufacturing process, the resulting mismatches and spurious become also random variables and their description involves necessarily a statistical modeling. That's why we also provided a statistical description of the SFDR in term of probability density function and the probability to be lower than a critical value. Therefore, for a level of performance determined by a minimum SFDR and its probability of achievement we can specify the required mismatch dispersion. This practical information becomes of relevant importance to establish robust design with safe margins.

Based on the TI-ADC model developed in chapter 4, a fully digital calibration algorithm was derived. The calibration consisted into mismatch estimation and compensation. The estimation used a rational fractional delay filter combined to a lowpass filter to estimate adaptively the different mismatches. The estimation was done blindly so that the ADC can keep running while estimation is being processed. Numerical results show that all the mismatches can

be estimated with less than 10-K samples which is fast compared to the state of art. In addition, the accuracy of the mismatch estimation is 98%, 94% and 88% for gain, time-skew and bandwidth mismatches. The compensation was based on the development of a matrix approach to cancel the effects of channel mismatch errors. The method was tested with a two-channel ADC board from Analog Devices and the results show that the linearity can be improved by almost 40 dB. The ASIC synthesis of the calibration algorithm in 65 nm GPLVT process shows that we can expect a power consumption in the order of 10 mW and an area consumption in the order of 0.035 mm^2 with two 51 taps compensation filters and with signals on 15 bits.

The calibration algorithm we developed assumes that the on-resistance is constant i.e independent of the input signal. However as we saw in chapter 2, this is not the case and the distortion created can be reduced by modifying the analog front end of the ADC with a bootstrapped circuit. As an perspective we can foresee a fully digital calibration which remove the nonlinearity created by the signal dependency of the on-resistance of the S/H and which also corrects channel mismatch errors. We can also apply the same method than that of used for the nonlinearity of the bootstrap to analyze other errors like charge injection and clock feedthrough.

Bibliography

[1] Boris Murmann. Adc performance survey 1997-2014 available at http://www.stanford.edu/ murmann/adcsurvey.html.

[2] Kurosawa Naoki, Kobayashi Haruo, Maruyama Kaoru, Hidetake Sugawara, and Kensuke Kobayashi. Explicit analysis of channel mismatch effects in time-interleaved adc systems. *IEEE Transactions on Circuits and Systems*, 48:261–271, 2001.

[3] Stéphane Paquelet, Gaël Kamdem De Teyou, and Yann Le Guillou. Ti-adcs sfdr requirement analysis. *IEEE International New Circuits and Systems Conference*, 2013.

[4] Shafiq M. Jamal et all. Calibration of sample-time error in a two-channel time-interleaved analog-to-digital converter. *IEEE Trans. on Circuits and Syst I.*, 51:130–139, 2004.

[5] P. Satarzadeh, B. C. Levy, and P. J. Hurst. Adaptive semiblind calibration of bandwidth mismatch for two-channel time-interleaved adcs. *IEEE Trans. on Circuits and Syst.*, 56:2075–2088, 2009.

[6] Steven Huang and Bernard C. Levy. Adaptive blind calibration of timing offset and gain mismatch for two-channel time-interleaved adcs. *IEEE Trans. on Circuits and Syst.*, 53:1278–1288, 2006.

[7] David Camarero. *Mixed-signal clock-skew calibration in time-interleaved analog-to-digital converters.* PhD thesis, Tlcom ParisTech, 46 rue barrault, 75013, Paris, France, 2007.

[8] Philippe Benabes, Caroline Lelandais-Perrault, and Nicolas Le Dortz. Mismatch calibration methods for high-speed time-interleaved adcs. *Proceedings of the 12th IEEE International New Circuits and Systems Conference*, 2014.

[9] Gaël Kamdem De Teyou, Hervé Petit, and Patrick Loumeau. Adaptive and digital blind calibration of transfer function mismatch in time-interleaved adcs. *IEEE International New Circuits and Systems Conference*, 2015.

[10] Han Le Duc et al. A fully digital background calibration of timing skew in undersampling ti-adc. *IEEE International New Circuits and Systems Conference*, 2014.

[11] F. Rivet, A. Mariano, D. Dallet, and J-B. Begueret. A mixed-signal built-in self-calibrated time-interleaved adc in 65 nm cmos technology. *IEEE New Conference on Circuits and Systems*, 2010.

[12] N. Le Dortz et. al. A 1.6gs/s time-interleaved sar adc with fully digital background mismatch calibration achieving interleaving spurs below 70 dbfs. *IEEE Solid-State Circuits Conference*, 2014.

[13] Jonas Elbornsson. *Analysis, Estimation and Compensation of Mismatch Effects in A/D Converters*. PhD thesis, Department of Electrical Engineering, University of Linkopings, Sweden, 2003.

[14] Dusan Stepanovic. *Calibration Techniques for Time-Interleaved SAR A/D Converters*. PhD thesis, University of California Berkeley, USA, 2012.

[15] Vijay Divi and Gregory W. Wornell. Blind calibration of timing skew in time-interleaved analog-to-digital converters. *IEEE Journal Of Selected Topics In Signal Processing.*, 3:509–522, 2009.

[16] Raouf Khalil, Marie-Minerve Louerat, Roger Petigny, and Hugo Gicquel. Background time skew calibration for time-interleaved adc using phase detection method. *IEEE New Circuits and Systems Conference.*, 2012.

[17] Tsung-Heng Tsai, Paul J. Hurst, and Stephen H. Lewis. Bandwidth mismatch and its correction in time-interleaved analog-to-digital converters. *IEEE Trans. on Circuits and Syst II.*, 53:1133–1137, 2006.

[18] Fatima Ghanem. *Bandwidth Mismatch Calibration in Time-Interleaved Analog-to-Digital Converters*. PhD thesis, Telecom ParisTech, France, 2012.

[19] Paul J. Hurst Tsung-Heng Tsai and Stephen H. Lewis. Correction of mismatches in a time-interleaved analog-to-digital converter in an adaptively equalized digital communication receiver. *IEEE Trans. on Circuits and Syst I.*, 56:307–319, 2009.

[20] Shahzad Saleem and Christian Vogel. Adaptive blind background calibration of polynomial-represented frequency response mismatches in a two-channel time-interleaved adc. *IEEE Trans. on Circuits and Syst I.*, 58:1300–1310, 2011.

[21] A. Bonnetat, J-M Hodé, D. Dallet, and G. Ferré. A new fully digital frequency response mismatch compensation algorithm for time-interleaved analog-to-digital converters. *IEEE International Radar Conference*, 2014.

[22] Texas Instruments. Understanding data converters. Technical report, Texas Instruments, 1995.

[23] Walt Kester. Understand sinad, enob, snr, thd, thd + n, and sfdr so you don't get lost in the noise floor. Technical report, Analog Device.

[24] Boris Murmann. Thermal noise in track-and-hold circuits, analysis and simulation techniques. *IEEE Solid-State Circuits Magazine*, pages 46–54, 2012.

[25] P. Hurst, S. Lewis, P. Gray, and R. Meyer. *Analysis and Design of Analog Integrated Circuits*. New York: Wiley, 2001.

[26] Behzad Razavi. *Principles of Data Conversion System Design.* IEEE Press, NY, 1995.

[27] A. M. Abo and P. R. Gray. A 1.5-v, 10-bit, 14.3-ms/s cmos pipeline analog-to-digital converter. *IEEE Journal Of Solid-State Circuits*, 34:599–606, 1999.

[28] H. Pan et al. A 3.3-v 12-b 50-ms/s a/d converter in 0.6 um cmos with over 80 db sfdr. *IEEE Journal Of Solid-State Circuits*, 35:599–606, 2000.

[29] S. Gupta, M. Choi, M. Inerfield, and J. Wang. A 1 gs/s 11b time-interleaved adc in 0.13 um cmos. *IEEE International Of Solid-State Circuits Conference*, 2006.

[30] P. R. Gray et al. *Analysis and Design and Analog Integrated Circuits*. New York Wiley, 2001.

[31] P. Satarzadeh, B. C. Levy, and P. J. Hurst. Digital calibration of a nonlinear s/h. *IEEE Journal Of Selected Topics In Signal Processing*, 3:454–471, 2009.

[32] Liang Dai and Ramesh Harjani. Cmos switched-op-amp-based sample-and-hold circuit. *IEEE Journal Of Solid-State Circuits*, 35:109–113, 2000.

[33] M. Waltari. *Circuit Techniques for Low-Voltage and High-Speed A/D Converters*. PhD thesis, Helsinki University of Technology, 2002.

[34] H.K. Data converters, 2005.

[35] M. WALTARI and K. HALONEN. A 220-msample/s cmos sample-and-hold circuit using double-sampling. *Analog Integrated Circuits and Signal Processing*, 18:21–31, 1999.

[36] M. WALTARI and K. HALONEN. Timing skew insensitive switching for double-sampled circuits. *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems*, 2:61–64, 1999.

[37] Anand Mohan. *A Reconfigurable High Speed Analog to Digital Converter Architecture for Ultra Wideband Devices*. PhD thesis, Faculty of Health, Engineering and Science of Victoria University, Australia, 2010.

[38] Y. Creten, P. Merken, W. Sansen, R.P. Mertens, , and C. Van Hoof. An 8-bit flash analog-to-digital converter in standard cmos technology functional from 4.2 k to 300 k. *IEEE Journal of Solid-State Circuits*, 44:2019–2025, 2009.

[39] M.O. Shaker & S. Gosh & M.A. Bayoumi. A 1-gs/s 6-bit flash adc in 90 nm cmos. *IEEE International Midwest Symposium on Circuits and Systems, MWSCAS*, pages 144–147, 2009.

[40] Walt Kester. Adc architectures ii: Successive approximation adcs available at http://www.analog.com/static/imported-files/tutorials/mt-021.pdf. Technical report, Analog Device, 2008.

[41] Xuan-Lun Huang, Ping-Ying Kang, Hsiu-Ming Chang, and Jiun-Lang Huang. A self-testing and calibration method for embedded successive approximation register adc. *IEEE Design Automation Conference (ASP-DAC), Asia and South Pacific*, pages 713–718, 2011.

[42] M. Casubolo & M. Grassi & A. Lombardi & F. Maloberti & P. Malcovati. A two-bit-percycle successive-approximation adc with background offset calibration. *IEEE International Conference on Electronics, Circuits and Systems*, pages 650–653, 2008.

[43] Bonnie Bakerr. How delta-sigma adcs work, part 1 available at http://www.ti.com/lit/an/slyt423/slyt423.pdf. Technical report, Texas Instruments Incorporated, 2011.

[44] Erkan Alpman. *A 7 bit 2.5 GS/sec Time-Interleaved C-2C SAR ADC for 60 GHz Multi-Band OFDM-Based Receivers.* PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University, 2009.

[45] Tibi Galambos. Adc performance metrics, measurement and calibration techniques. Technical report, PMC-s=SIERRA, 2009.

[46] A. B. Sripad and D. L. Snyder. A necessary and a sufficient condition for quantization errors to be uniform and white. *IEEE Transactions On Acoustics, Speech and Signal Processing.*, 25:442–448, 1977.

[47] Russell K. Hobbie and Bradley John Roth. *Intermediate physics for medicine and biology.* Springer, 2007.

[48] X. Gao, E. A. M. Klumperink, P. F. J. Geraedts, and B. Nauta. Jitter analysis and a benchmarking figure-of-merit for phase-locked loops. *IEEE Transactions On Circuits ans systems II.*, 56:117–121, 2009.

[49] Thomas Neu. Clock jitter analyzed in the time domain part 1 available at http://www.ti.com/lit/an/slyt379/slyt379.pdf. Technical report, Texas Instrument Incorporated, 2010.

[50] Andrew Fogg. Baseband / rf digital interface specification. logical, electrical and timing characteristics, egprs version available at http://mid.mipi.org/docs/digrf_standard_v112.pdf. Technical report, Digital Interface Working Group, 2004.

[51] M. Lohning and G. Fettweis. The effects of aperture jitter and clock jitter in wideband adcs. *Computer Standards and Interfaces*, 2007.

[52] Ali Hajimiri, Sotirios Limotyrakis, and Thomas H. Lee. Jitter and phase noise in ring oscillators. *IEEE Journal of Solid-State Circuits*, 34:790–804, 1999.

[53] B. Razavi. Noise, lecture, fall 11 ee university of california la available at http://www.ee.ucla.edu/ brweb/teaching/215a_f2011/noise.pdf.

[54] Yann Le Guillou. *Contribution l'tude de la conversion analogique-numrique sigma-delta intgre dans une chaine de rception radiofrquence pour les applications cellulaires.* PhD thesis, University of Caen, France, 2005.

[55] Jennifer Eve Hoffman. *A Search for Alternative Electronic Order in the High Temperature Superconductor Bi2 Sr_2 $CaCu_2$ $O_{8+?}$ by Scanning Tunneling Microscopy.* PhD thesis, University of California Berkeley, USA, 2003.

[56] M. E. Chammas and B. Murmann. General analysis on the impact of phase-skew in time-interleaved adcs. *IEEE Trans. on Circuits and Syst. I: Regular Papers*, 56:902–910, 2004.

[57] Gildas Leger et. al. Impact of random channel mismatch on the snr and sfdr of time-interleaved adcs. *IEEE Trans. on Circuits and Syst.*, 51:140–150, 2009.

[58] Christian Vogel. The impact of combined channel mismatch effects in time-interleaved adcs. *IEEE Trans. on Instrumentation and Measurement*, 54:415–427, 2005.

[59] K. Poulton et all. A 20 gs 8b adc with a 1 mb memory in 0.18um cmos. *Proceedings of IEEE International Solid-State Circuits Conference.*, 55:318–496, 2003.

[60] B. Nauta X. Gao and E. Klumperink. Advantages of shift registers over dlls for flexible low jitter multiphase clock generation. *IEEE Transactions On Circuits and Systems II: Express Briefs.*, 55:244–248, 2008.

[61] H. Mahmoodi, S. Mukhopadhyay, and K. Roy. Estimation of delay variations due to random-dopant fluctuations in nanoscale cmos circuits. *IEEE Journal of Solid-State Circuits*, 40:1787–1796, 2005.

[62] S. M. Louwsma, A. J. M van Tuijl, M. Vertregt, and B. Nauta. A 1.35 gs/s, 10 b, 175 mw time-interleaved ad converter in 0.13 μm cmos. *IEEE Journal of Solid-State Circuits*, pages 778–786, 2008.

[63] Gaël Kamdem De Teyou, Hervé Petit, Patrick Loumeau, Hussein Fakhoury, Yann Le Guillou, and Stéphane Paquelet. Statistical analysis of noise in broadband and high resolution adcs. *IEEE International Conference on Circuits and Systems*, 2014.

[64] Javier Diaz-Carmona* Gordana Jovanovic-Dolecek. One method for fir fractional delay filter design. *Proceedings of the Fourth IEEE International Caracas Conference on Devices Circuits and Systems*, 2002.

[65] Y. Chiu. Sample and hold basics. University Lecture, 2012.

[66] Parastoo Nikaeen. *Digital Compensation Of Dynamic Acquisition Errors At The Front-End Of ADCs.* PhD thesis, University of Stanford, USA, 2008.

[67] M. Al-Shyoukh, D. Aksin, and F. Maloberti. Switch bootstrapping for precise sampling beyond supply voltage. *IEEE Journal of Solid-State Circuits*, 41:1938–1943, 2006.

[68] M. Al-Shyoukh, D. Aksin, and F. Maloberti. Switch bootstrapping for precise sampling beyond supply voltage. *IEEE Journal Of Solid-State Circuits*, 2006.

[69] D. Bormann et al. A multiband multistandard notch filter lna for lte, wcdma and gsm for saw-less frontends. *Proceedings of the Asia-Pacific Microwave Conference*, pages 498–501.

[70] T. ltakura. Effects of the sampling pulse width on the frequency characteristics of a sample-and-hold circuit. *IEEE Procedings, Circuits, Devices and Systems*, 141:328–336, 1994.

[71] Bob Verbruggen, Masao Iriguchi, and Jan Craninckx. A 1.7mw 11b 250mss 2x interleaved fully dynamic pipelined sar adc in 40nm digital cmos. *IEEE International Solid-State Circuits Conference*, 2012.

[72] Jonas Elbornsson. *Analysis, Estimation and Compensation of Mismatch Effects in AD Converters*. PhD thesis, Linkpings universitet, SE 581 83 Linkping, Sweden, 2003.

[73] Paul Horowitz and Hill Winfield. *The Art of Electronics*. Cambridge University Press, 1989.

[74] Leo P. Mulcahy. Statistical an alysis of digital fixed-point multiplication errors and quantization errors. *National Technical Information Service*, pages 7–8, 1971.

[75] Brian Black. Analog-to-digital converter architectures and choices for system design. *Analog Dialogue*, 33-8, 1999.

[76] Inc Analog Device. *Linear Circuit Design Handbook*. Hank Zumbahlen, 2008.

[77] Gupta Sachin and Phatak Akshay. Adc guide, part 3: Offset errors. Technical report, Cypress Semiconductor Corp.

[78] E . B . Loewenstein. *The Measurement, Instrumentation and Sensors Handbook on CD-ROM*. Crc Press Llc, 1999.

[79] KOBAYASHI Haruo, MOFUMURA Masanao, KOBAYASHI Kensuke, and ONAYA Yoshitaka. Aperture jitter effects in wideband adc systems. *IEEE/SICE International Symposium on System Integration*, 1999.

[80] Michael Lohning and Gerhard Fettweis. The effects of aperture jitter and clock jitter in wideband adcs. *Computer Standards and Interfacese*, 29:11–18, 2007.

[81] Walt Kester. Aperture time, aperture jitter, aperture delay time— removing the confusion. Technical report, Analog Devices, 2009.

[82] Mike Tyler. *he Mechatronics Handbook - 2 Volume Set*. Robert H . Bishop CRC Press 2002, 2002.

[83] Christopher Taillefer. *Analog-to-Digital Conversion via Time-Mode Signal Processing*. PhD thesis, McGill University, Montral, 2007.

[84] MAXIM IC. Adc and dac glossary, Jul 2002.

[85] ANEKAL B. SRIPAD and DONALD L. SNYDER. A necessary and sufficient condition for quantization errors to be uniform and white. *IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING*, 5:443–448, 1977.

[86] Walt Kester. Understand sinad, enob, snr, thd, thd + n, and sfdr so you don't get lost in the noise floor. Technical report, Analog Devices, 2008.

[87] Claude Shannon. Communication in the presence of noise. *Institute of Radio Engineers*, 37:10–21, 1949.

[88] Haruo Kobayashi, Kensuke Kobayashi, Masanao Morimura, Yoshitaka Onaya, Yuuich Takahashi, Kouhei Enomoto, and Hideyuki Kogure. The effects of aperture jitter and clock jitter in wideband adcs. *IEICE Transactions Fundamentals*, 85:335, 2002.

[89] E.J. Peralias, A. Rueda, J.L. Huertas, and G. Leger. Impact of random channel mismatch on the snr and sfdr of time-interleaved adcs. *IEEE Transactions on Circuits and Systems*, 51:140–150, 2004.

[90] C. Jun, R. Feng, and X. Mei-hua. Ic design of 2ms/s 10-bit sar adc with low power. *International Symposium on High Density packaging and Microsystem Integration*, pages 1–3, 2007.

List of Figures

List of Tables

Appendix A

Appendix A: CMOS Bootstrapped and Sample and Hold Circuit

A.1 On-resistance in function of the input signal in a single ended CMOS Bootstrap circuit

Fig. A.1 shows a the logical structure of a CMOS bootstrapped S/H circuit. Ideally the gate-to-source voltage v_{gs} is now independent of the the input signal. But due to parasitic capacitances at node N_1, mobility degradation and back gate effet, this is not the case. For example, let's call C_p the parasitic capacitance at note N_1. The voltage v_{gs} can be written as:

$$v_{gs} = \frac{C_3}{C_3 + C_p} v_{dd} - \frac{C_p}{C_3 + C_p} x \tag{A.1}$$

In (A.1), the gate-to-source voltage still depends of the input signal through C_p. By replacing this value of v_{gs} in the expression of the on-resistance we obtain:

$$\begin{aligned} R_{on} &= \left[\mu C_{ox} \frac{W}{L} \Big(v_{gs} - v_{th}\Big)\right]^{-1} \\ &\simeq b_{on} + a_{on} x \end{aligned} \tag{A.2}$$

In (A.2), the parasitic capacitance has been considered small compared to the bootstrap capacitance C_3. a_{on} and b_{on} are given by:

$$\begin{aligned} a_{on} &= \frac{1}{\mu C_{ox} \frac{W}{L} \left[\frac{C_3}{C_3 + C_p} v_{dd} - v_{th}\right]} \\ b_{on} &= \frac{\frac{C_p}{C_3 + C_p}}{\mu C_{ox} \frac{W}{L} \left[\frac{C_3}{C_3 + C_p} v_{dd} - v_{th}\right]} \end{aligned} \tag{A.3}$$

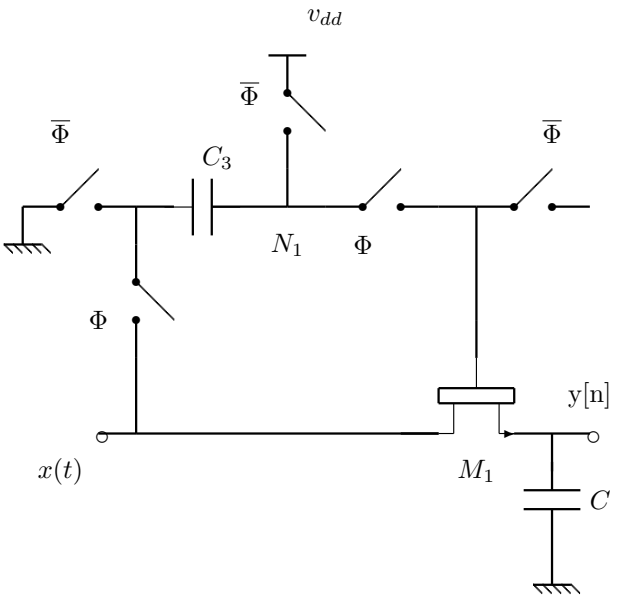


Figure A.1: Logical structure of the bootstrap circuit.

From this we see that, the on-resistance still depends of the the input signal, but in a linear manner. We have not considered the back gate effect and mobility degradation. However, this linear dependency remains when considering these nonidealities as simulation results of a CMOS bootstrapped S/H in 65 nm process show it on Fig. 2.7.

A.2 Output signal of a single ended bootstrap S/H circuit

Fig. A.2, shows the clock diagram of a S/H.

A.2.1 Sampling mode

During the sampling mode, the transistor-switch is ON and the input signal charges or discharges the hold capacitor so that the voltage $y(t)$ across the capacitor is practically proportional to the the input voltage $x(t)$. This stage goes from $nT_s - \beta T_s$ to nT_s. n is the index of a given sample at S/H output and βT_s is the acquisition time which is taken as a fraction of the sampling period T_s. A typical value of β is $\frac{1}{2}$. In this stage, The circuit is governed by an inhomogeneous linear Ordinary Differential Equation (ODE) of first order:

$$y(t) + \tau(x)\frac{dy(t)}{dt} = x(t) \tag{A.4}$$

We can write (A.4) as:

$$y'(t) + A'(t)y = x(t)A'(t) \tag{A.5}$$

In (A.5), $y'(t)$ is the derivate of y and the derivate $A'(t)$ is given by:

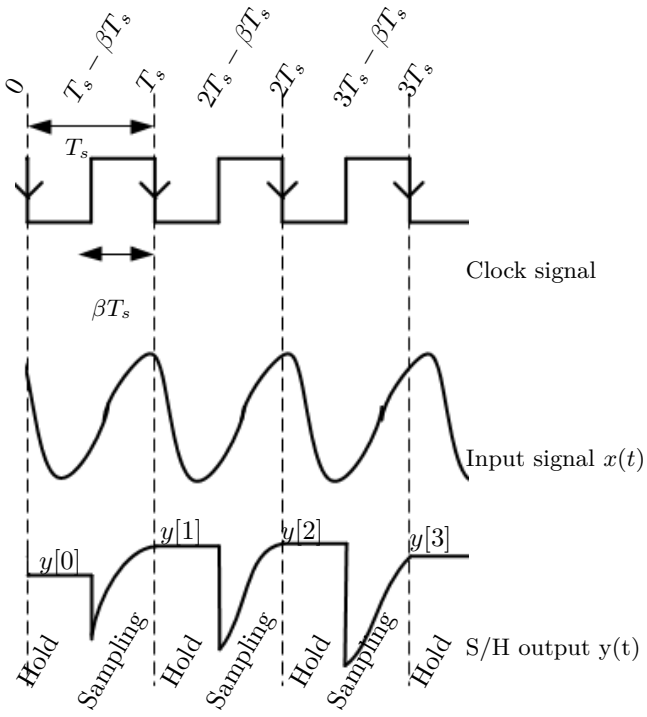


Figure A.2: Clock diagram of a bootstrapped S/H

$$\begin{aligned} A'(t) &= \frac{1}{\tau(x)} \\ &= \frac{1}{b + ax(t)} \\ &\simeq \frac{1}{b}\Big[1 - \frac{a}{b}x(t)\Big] \end{aligned} \tag{A.6}$$

In (A.6), the time constant $\tau(x)$ has been replaced by its linear expression of section 2.2.3:

A.2.2 Hold mode

During this hold mode, the switch is OFF and the input signal is disconnected from the capacitor. The voltage accross the capacitor is stored as the sampled value $y[n]$. The equation governing the circuit is: voltage accross the capacitor is stored as the sampled value $y[n]$. The equation governing the circuit is:

$$\begin{aligned} &y(t) = y[n] = y(nT_s) \\ &nT_s \leq t \leq (n+1)T_s - \beta T_s \end{aligned} \tag{A.7}$$

We are going to solve the inhomogeneous linear ODE of(A.5) with the variation of parameters method. First we will find the solutions of the homegenoeus equation associated to (A.5) and then we will find a particular solution.

A.2.3 Homogeneous ODE

The homogeneous ODE associated to (A.5) is:

$$y'(t) + A'(t)y = 0 \tag{A.8}$$

The general solution y_0 of the homogeneous equation (A.8) is:

$$y_0(t) = C_0 e^{-A(t)} \tag{A.9}$$

With C_0 a constant parameter.

A.3 Solution of the inhomogeneous equation

The general solution $y(t)$ can be written as:

$$y(t) = C(t)e^{-A(t)} \tag{A.10}$$

y must satisfy (A.5). So by derivating y and inserting this derivate in (A.5), we find that:

$$C'(t) = A'(t)x(t)e^{A(t)} \tag{A.11}$$

We obtain C by integrating (A.11):

$$C(t) = \int_{nT_s-\beta T_s}^{t} A'(\theta)x(\theta)e^{A(\theta)}d\theta + C_1 \tag{A.12}$$

In (A.12), C_1 is a constant parameter that will be determined with the intitial conditions and $A(t)$ is given by:

$$\begin{aligned} A(t) &= \frac{1}{b}\int_{nT_s-\beta T_s}^{t}\Big[1 - \frac{a}{b}x(u)\Big]du \\ &= \frac{t - nT_s - \beta T_s}{b} - \frac{a}{b^2}\int_{nT_s-\beta T_s}^{t} x(u) \end{aligned} \tag{A.13}$$

And using the fact that $ax \ll b$, we have:

$$e^{-A(t)} \simeq e^{-\frac{t-nT_s-\beta T_s}{b}}\Big[1 + \frac{a}{b^2}\int_{nT_s-\beta T_s}^{t} x(u)du\Big] \tag{A.14}$$

Inserting (A.12) and (A.14) in (A.10), we have:

$$
\begin{aligned}
y(t) &= \Big[\int_{nT_s-\beta T_s}^{t} A'(\theta)x(\theta)e^{A(\theta)} + C_1\Big]e^{-A(t)} \\
&\simeq \Big[\int_{nT_s-\beta T_s}^{t} \frac{1}{b}\Big(1-\frac{a}{b}x(\theta)\Big)x(\theta)e^{\frac{\theta-nT_s-\beta T_s}{b}}\Big(1-\frac{b}{a^2}\int_{nT_s-\beta T_s}^{\theta}x(u)du\Big)d\theta\Big] \\
&.e^{-\frac{t-nT_s-\beta T_s}{b}}\Big[1+\frac{a}{b^2}\int_{nT_s-\beta T_s}^{t}x(v)dv\Big] \\
&= \underbrace{\frac{1}{b}\int_{nT_s-\beta T_s}^{t}x(\theta)e^{-\frac{t-\theta}{b}}}_{y_1(t)} - \underbrace{\frac{a}{b^2}\int_{nT_s-\beta T_s}^{t}x^2(\theta)e^{-\frac{t-\theta}{b}}}_{y_2(t)} \\
&\underbrace{-\frac{a}{b^3}\int_{nT_s-\beta T_s}^{t}\int_{nT_s-\beta T_s}^{\theta}x(u)x(\theta)e^{-\frac{t-nT_s-\beta T_s}{b}}e^{-\frac{\theta-nT_s-\beta T_s}{b}}dud\theta}_{y_3(t)} \\
&\underbrace{+\frac{a}{b^3}\int_{nT_s-\beta T_s}^{t}\int_{nT_s-\beta T_s}^{\theta}x(v)x(\theta)e^{-\frac{t-nT_s-\beta T_s}{b}}e^{-\frac{\theta-nT_s-\beta T_s}{b}}dvd\theta}_{y_4(t)} + \underbrace{C_1e^{-\frac{t-nT_s-\beta T_s}{b}}\Big[1+\frac{a}{b^3}\int_{nT_s-\beta T_s}^{t}x(u)du\Big]}_{y_5(t)}
\end{aligned}
\tag{A.15}
$$

Initial Conditions

At $t = nT_s - \beta T_s$, we have:

$$
y(nT_s - \beta T_s) = y[n-1] = C_1 \tag{A.16}
$$

Expressions of y_1 ... y_5

We can find the different expressions of $y_i[n]$ as:

$$
\begin{aligned}
y_1[n] &= \frac{1}{b}\int_{nT_s-\beta T_s}^{nT_s}x(\theta)e^{-\frac{nT_s-\theta}{b}}d\theta \\
&= (h\star x)(nT_s) - \alpha(h\star x)(nT_s-\beta T_s) \\
&\simeq (h\star x)(nT_s)
\end{aligned}
\tag{A.17}
$$

With $\alpha = e^{-\frac{\beta T_s}{b}} \simeq 0$ because the sampling duration is several time bigger than the time constant. Nevertheless α can be taken into account if cases arises.

$$
\begin{aligned}
y_2[n] &= -\frac{a}{b}\int_{nT_s-\beta T_s}^{nT_s}x^2(\theta)e^{-\frac{nT_s-\theta}{b}}d\theta \\
&= (h\star x^2)(nT_s) - \alpha(h\star x^2)(nT_s-\beta T_s) \\
&\simeq (h\star x^2)(nT_s)
\end{aligned}
\tag{A.18}
$$

$$
\begin{aligned}
y_3[n] + y_4[n] &= \frac{a}{b^3}\int_{nT_s-\beta T_s}^{nT_s}e^{-\frac{nT_s-\theta}{b}}\int_{\theta}^{nT_s}e^{-\frac{\theta-v}{b}}dvd\theta \\
&\simeq \frac{a}{b}\Big(h\star\Big[x.(h\star x)\Big]\Big)(nT_s)
\end{aligned}
\tag{A.19}
$$

And y_5 is given by:

$$\begin{aligned} y_5[n] &= y[n-1]\alpha\Big[1+\frac{a}{b^2}\Big((u\star x)(nT_s)-(u\star x)(nT_s-\beta T_s)\Big)\Big] \\ &\simeq 0 \end{aligned} \tag{A.20}$$

Finally we have:

$$\begin{aligned} y[n] &= y_1[n]+y_2[n]+y_3[n]+y_4[n]+y_5[n] \\ &= (h\star x)(nT_s)-\frac{a}{b}\Big\{(h\star x^2)(nT_s)-\Big(h\star\Big[x.(h\star x)\Big]\Big)(nT_s)\Big\} \end{aligned} \tag{A.21}$$

A.4 Output of the S/H without nonlinearities

Considering that, the S/H is enough linear in (A.21), the output of the S/H can be written as:

$$y[n] = (h\star x)(nT_s) \tag{A.22}$$

If we consider also, the time-skew t_0, the gain G, the transfer function of the S/H can be written as:

$$H(f) = \frac{G}{1+j\frac{f}{f_c}}e^{-j2\pi f t_0} \tag{A.23}$$

Appendix B

Spectrum of the TI-ADCs

B.1 Spectrum of the DC Component

The DTFT of the DC component at the TI-ADCs output is given:

$$\begin{aligned} Y_{DC}(f) &= \sum_{n=-\infty}^{+\infty} O_{n[M]} \exp(-j2\pi f n T_s) \\ &= \sum_{m=0}^{M-1} O_m \exp(-j2\pi f m T_s) \sum_{k=-\infty}^{+\infty} \exp(-j2\pi f k M T_s) \end{aligned} \tag{B.1}$$

Poisson theorem states that:

$$\sum_{k=-\infty}^{+\infty} \exp(-j2\pi k f M T_s) = \frac{f_s}{M} \sum_{k=-\infty}^{+\infty} \delta\left(f - k\frac{f_s}{M}\right) \tag{B.2}$$

Using (B.2), Y_{DC} in (B.1) becomes:

$$\begin{aligned} Y_{DC} &= \frac{f_s}{M} \sum_{k=-\infty}^{+\infty} \sum_{m=0}^{M-1} O_m \exp\left(-j2\pi k \frac{f_s}{M} m T_s\right) \delta\left(f - k\frac{f_s}{M}\right) \\ &= \frac{f_s}{M} \sum_{k=-\infty}^{+\infty} \Big[\sum_{m=0}^{M-1} O_m \zeta^{-mk}\Big] \delta\left(f - k\frac{f_s}{M}\right) \\ &= \frac{f_s}{M} \sum_{k=-\infty}^{+\infty} \Big[\sum_{m=0}^{M-1} \Big(1 + \delta O_m\Big) O_0 \zeta^{-mk}\Big] \delta\left(f - k\frac{f_s}{M}\right) \end{aligned} \tag{B.3}$$

Finally by separating harmonic frequencies kf_s from non-harmonic frequencies $k\frac{f_s}{M}$ we obtain:

$$\frac{Y_{DC}(f)}{f_s} = \sum_{k=-\infty}^{+\infty} \overbrace{\Big[1 + \frac{1}{M}\sum_{m=0}^{M-1} \delta O_m(.)\Big] O_0 \delta(f - k f_s)}^{Regular\ part} + \sum_{\infty}^{+\infty} \frac{1}{M} \underbrace{\left[\sum_{m=0}^{M-1} \zeta^{-mk} \delta_m O\right] O_0 \delta(f - k\frac{f_s}{M})}_{Spurious\ part} \quad \text{(B.4)}$$

B.2 Spectrum of the AC component

The DTFT of the AC component at the TI-ADCs output is given by:

$$\begin{aligned}
Y_{AC}(f) &= \sum_{n=-\infty}^{+\infty} y_{n[M]} \exp(-j2\pi f n T_s) \\
&= \sum_{m=0}^{M-1} \exp(-j2\pi f m T_s) \sum_{k=-\infty}^{+\infty} (h_m \star x)(kMT_s + mT_s) \exp(-j2\pi f k M T_s) \\
&= \sum_{m=0}^{M-1} \exp(-j2\pi f m T_s) \frac{f_s}{M} \sum_{k=-\infty}^{+\infty} \exp\Big(j2\pi(.) m T_s\Big) H_m(.) X(.)\Big|_{f - k\frac{f_s}{M}} \\
&= \frac{f_s}{M} \sum_{k=-\infty}^{+\infty} \Big[\sum_{m=0}^{M-1} \zeta^{-mk} H_m(.)\Big] X(.)\Big|_{f - k\frac{f_s}{M}} \\
&= \frac{f_s}{M} \sum_{k=-\infty}^{+\infty} \Big[\sum_{m=0}^{M-1} \zeta^{-mk} \Big(1 + \delta H_m(.)\Big)\Big] H_0(.) X(.)\Big|_{f - k\frac{f_s}{M}}
\end{aligned} \quad \text{(B.5)}$$

Finally by separating harmonic replicas at kf_s from non-harmonic replicas at $k\frac{f_s}{M}$ we obtain:

$$\frac{Y_{AC}(f)}{f_s} = \sum_{\substack{k \neq 0[M] \\ -\infty}}^{+\infty} \frac{1}{M} \underbrace{\left[\sum_{m=0}^{M-1} \zeta^{-mk} \delta H_m(.)\right] H_0(.) X(.)\Big|_{f - k\frac{f_s}{M}}}_{Spurious\ part} + \sum_{k=-\infty}^{+\infty} \overbrace{\Big[1 + \frac{1}{M}\sum_{m=0}^{M-1} \delta H_m(.)\Big] H_0(.) X(.)\Big|_{f - k f_s}}^{Regular\ part} \quad \text{(B.6)}$$

Appendix C

Statistical Analysis of TI-ADCs

C.1 Probability Density Function of $S_a(k)$

We have:

$$\begin{aligned} S_a(k) &= \Big| \sum_{m=1}^{M-1} \zeta^{-mk} \xi_m^a \Big|^2 \\ &= \Big[\sum_{m=1}^{M-1} \cos\left(\frac{2\pi mk}{M}\right) \xi_m^a \Big]^2 + \Big[\sum_{m=1}^{M-1} \sin\left(\frac{2\pi mk}{M}\right) \xi_m^a \Big]^2 \\ &= X_k^2 + Y_k^2 \end{aligned} \tag{C.1}$$

X_k and Y_k are two gaussian random variables as they are the sum of independent gaussian random variables. In addition we have: $\mathbb{E}[X_k Y_k] = \mathbb{E}[X_k]\mathbb{E}[Y_k]$. As a consequence X_k and Y_k are two gaussian independent variables. Their PDF are given by:

$$\begin{cases} X_k = \Big[\sum_{m=1}^{M-1} \cos\left(\frac{2\pi mk}{M}\right) \xi_m^a \Big] \sim & \mathcal{N}\Big(0, \sqrt{\sum_{m=1}^{M-1} \cos^2\left(\frac{2\pi mk}{M}\right)}\Big) \\ Y_k = \Big[\sum_{m=1}^{M-1} \sin\left(\frac{2\pi mk}{M}\right) \xi_m^a \Big] \sim & \mathcal{N}\Big(0, \sqrt{\sum_{m=1}^{M-1} \sin^2\left(\frac{2\pi mk}{M}\right)}\Big) \end{cases} \tag{C.2}$$

After the calculation of their variances we can see that:

$$\begin{cases} X_k,\ \ Y_k \sim \mathcal{N}\left(0, \frac{M}{2} - 1\right) & k \neq \frac{M}{2} \\ X_k \sim \mathcal{N}\left(0, M - 1\right) & k = \frac{M}{2} \\ Y_k \sim \mathcal{N}\left(0, 0\right) & k = \frac{M}{2} \end{cases} \tag{C.3}$$

We can make the following analysis:

- If $k \neq \frac{M}{2}$, $S_a(k)$ is the sum of two independent squared gaussian random variables. As a consequence its PDF is a Chi-square law with two degree freedom χ_2^2.

- If $k = \frac{M}{2}$, $S_a(k)$ is the square of a gaussian random variable. As a consequence its PDF is a Chi-square with one degree freedom χ_1^2.

C.2 PDF of S_{max}

Let's consider the case where M is odd. For M is even, the methodology is the same. The random variables $S_a(k)$ follows a Chi-square distribution χ_2^2 with two degrees of freedom. The demonstration is shown in Appendix C. C.1. Therefore all the random variable $S_a(k)$ have the same PDF which is:

$$f_0(s) = \frac{1}{M'} e^{\frac{-s}{M'}} \tag{C.4}$$

With $M' = M-2$. The Cumulative Density Function F_0 of the random variables $S_a(k)$ is given by:

$$F_0(s) = 1 - e^{-\frac{s}{M'}} \qquad s > 0 \tag{C.5}$$

The proof of (C.5) is detailed in Appendix C. C.3. For M odd there are $(M-1)/2$ spurs from $k = 1$ to $k = (M-1)/2$. The CDF F_{max} of S_{max} is the product of the CDF of the different $S_a(k)_{k=1\ldots(M-1)/2}$ as the different $S_a(k)$ are independent [57]. As a consequence, we have:

$$F_{max}(s) = (F_0(s))^{\frac{M-1}{2}} = (1 - e^{-\frac{s}{M}})^{\frac{M-1}{2}} \tag{C.6}$$

The PDF f_{max} of S_{max} is obtained by differentiating its PDF:

$$f_{max}(s) = \frac{M-1}{2M} e^{-\frac{s}{M}} \left(1 - e^{-\frac{s}{M}}\right)^{\frac{M-3}{2}} \tag{C.7}$$

From (4.16) and (C.7), we can find the PDF of the SFDR by a change of variables:

C.3 Cumulative Density Function of $S_a(k)$

The Cumulative Density Function (CDF) is obtained by integrating the PDF. For $k \neq \frac{M}{2}$ odd, the CDF F_0 is given by:

$$\begin{aligned} F_0(s) &= \int^s f_0(s')ds' \\ &= \int^s \frac{1}{M} e^{-\frac{s}{M}} \\ &= 1 - e^{-\frac{s}{M}} \end{aligned} \tag{C.8}$$

The constant 1 in (C.8) by considering the fact that $s > 0$ and $F_0(0) = 0$.

Appendix D

Appendix E: Thermal Noise

D.0.1 Total power

Let's denote $h(t) = \frac{1}{R_{on}C} e^{-\frac{t}{R_{on}C}}$, $t > 0$, the impulse response of the RC S/H of Fig. 2.4. The noise source is filtered trough h and the total noise power at S/H output is obtaining by integrating output noise in all frequencies :

$$\begin{aligned} \overline{v_{thermal}^2} &= \int_{-\infty}^{+\infty} 2k_B T R_{on} \Big| \frac{1}{1 + j2\pi R_{on} C f} \Big|^2 df \\ &= \frac{k_B T}{C} \end{aligned} \tag{D.1}$$

D.0.2 Power spectral density of thermal noise

$R_{in}(\tau) = 2k_B T R_{on} \delta(\tau)$ denotes the autocorrelation function of thermal noise across R_{on}. The autocorrelation of thermal noise at the S/H output is :

$$\begin{aligned} R_{th}(\tau) &= R_{in}(\tau) \star h(\tau) \star h(-\tau) \\ &= \frac{k_B T}{C} e^{-\frac{|\tau|}{R_{on}C}} \end{aligned} \tag{D.2}$$

In discrete mode (D.2) becomes :

$$R_{th}[n] = \frac{k_B T}{C} e^{-\frac{|n|T_s}{R_{on}C}} \tag{D.3}$$

With T_s the sampling period. The PSD $S_{thermal}$ of thermal noise at S/H output is obtained by taking the Discrete Time Fourier Transform (DTFT) of R_{th} :

$$\begin{aligned} S_{thermal}(f) &= \frac{1}{f_s} \sum_{n=-\infty}^{+\infty} R_{th}[n] \exp(-j2\pi f n T_s) \\ &= \frac{1}{f_s} \sum_{n=-\infty}^{+\infty} \frac{k_B T}{C} \exp\left(-\frac{|n|T_s}{R_{on}C}\right) \exp(-j2\pi f n T_s) \end{aligned} \tag{D.4}$$

Using the serie [**?**] :

$$\sum_{n=-\infty}^{+\infty} \exp(-|na|)\exp(-jnb) = \frac{\sinh(a)}{\cosh(a)-\cos(b)} \tag{D.5}$$

We deduce that :

$$\begin{aligned} S_{thermal}(f) &= \frac{1}{f_s}\frac{k_B T}{C}\frac{1-\exp(-\frac{2T_s}{R_{on}C})}{1-2\exp(-\frac{T_s}{R_{on}C})\cos(2\pi\frac{f}{f_s})+\exp(-\frac{2T_s}{R_{on}C})} \\ &\simeq \frac{1}{f_s}\frac{k_B T}{C} \end{aligned} \tag{D.6}$$

The above approximation is valid because for ADCs of concern $R_{on} \sim 10\Omega$, $C \sim 10$pF and $f_s \sim 300$ MHz $\rightarrow \exp(-\frac{T_s}{R_{on}C}) \ll 1$.

Appendix E

Power Spectral Density of Jitter

E.1 Useful property of WSS signal

$$\mathbb{E}\Big[\overline{X(f_1)}X(f_2)\Big] = \delta(f_2 - f_1)\mathbb{E}\Big[|X(f_2)|^2\Big] \tag{E.1}$$

This property comes from the fact that the autocorrelation function of a signal can be written as :

$$\begin{aligned} R_{xx}(\tau) &= \mathbb{E}\Big[\int_{-\infty}^{+\infty} \overline{X(f_1)}\exp(-j2\pi f_1 t) \\ &\int_{-\infty}^{+\infty} X(f_2)\exp\Big(j2\pi f_2(t+\tau)\Big)\,\mathrm{d}f_1\,\mathrm{d}f_2\Big] \\ &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \mathbb{E}\Big[\overline{X(f_1)}X(f_2)\Big]\exp\Big(j2\pi(f_2-f_1)t\Big)\exp(j2\pi f_2\tau)\,\mathrm{d}f_1\,\mathrm{d}f_2 \end{aligned}$$

If the signal is stationnary, its autocorrelation function only doesn't depend of t.

E.2 Autocorrelation function of jitter noise

Before evaluating the PSD of jitter noise, we should first evaluate the autocorrelation of jitter noise. We find that, it is given by :

$$\begin{aligned} R_{ee}(k) &= \mathbb{E}\Big\{\overline{e[n]}e[n+k]\Big\} \\ &= \mathbb{E}\Big[\int_{f_1} \overline{X(f_1)}e^{-j2\pi f_1 nT_s}\Big(e^{-j2\pi f_1\epsilon[n]} - e^{-j2\pi f_1\epsilon[0]}\Big) \\ &\int_{f_2} X(f_2)e^{j2\pi f_2 nT_s}e^{j2\pi f_2 kT_s}\Big(e^{j2\pi f_1\epsilon[n+k]} - e^{-j2\pi f_1\epsilon[0]}\Big)\,\mathrm{d}f_1\,\mathrm{d}f_2\Big] \\ &= \int_{f_1} S_{xx}(f_1)\exp(j2\pi f_1 kT_s) \\ &\Big\{1 + \mathbb{E}\Big[e^{j2\pi f_1(\xi[n+k]-\xi[n])}\Big] - \mathbb{E}\Big[e^{j2\pi f_1\xi[n+k]}\Big] - \mathbb{E}\Big[e^{j2\pi f_1\xi[n]}\Big]\Big\}\,\mathrm{d}f_1 \end{aligned}$$

E.3 PSD of aperture jitter

As the aperture jitter is a gaussian white process, we have the following properties :

- $\mathbb{E}\left[e^{j2\pi f_1 \xi[n]}\right] = e^{-2\pi^2 f_1^2 \sigma_{apt}^2}$
- $\mathbb{E}\left[e^{j2\pi f_1 (\xi[n+k]-\xi[n])}\right] = e^{-4\pi^2 f_1^2 \sigma_{apt}^2} + \delta(k)(1 - e^{-4\pi^2 f_1^2 \sigma_{apt}^2})$

So, the autocorrelation function of jitter noise becomes :

$$R_{ee}(k) = \int_{-\infty}^{+\infty} S_{xx}(f_1) \exp(j2\pi f_1 k T_s) \left[(1 - e^{-2\pi^2 f_1^2 \sigma_{apt}^2})^2 + \delta(k)(1 - e^{-4\pi^2 f_1^2 \sigma_{apt}^2})\right] \mathrm{d}f_1$$

And we deduce the power spectral density of jitter noise by taking the DTFT of the autocorrelation function :

$$S_{jitter\_apt}(f) = \overbrace{S_{xx}(f)(1 - e^{-2\pi^2 f^2 \sigma_{apt}^2})^2}^{Frequency\ dependent\ component} + \overbrace{\frac{1}{f_s} \int_{-\infty}^{+\infty} S_{xx}(f_1)(1 - e^{-4\pi^2 f_1^2 \sigma_{apt}^2})\, \mathrm{d}f_1}^{White\ component} \tag{E.2}$$

E.4 PSD of sampling noise due to clock jitter of a free-running oscillator

For a free-running oscillator, there is an accumulation of clock jitter with time. So we have the following properties :

- $\xi[n] \sim \mathcal{N}(0, n\sigma_{clk}^2)$
- $\mathbb{E}\left[e^{j2\pi f_1 \xi[n]}\right] = e^{-2\pi^2 f_1^2 n \sigma_{clk}^2}$
- $\mathbb{E}\left[e^{j2\pi f (\xi[n+k]-\xi[n])}\right] = e^{-2\pi^2 f^2 |k| \sigma_{clk}^2}$

When $n \to +\infty$, $e^{-2\pi^2 f_1^2 n \sigma_{clk}^2} \to +0$, the jitter noise becomes stationnary and its autocorrelation is :

$$R_{ee}(k) \simeq \int_{-\infty}^{+\infty} S_{xx}(f_1) \exp(j2\pi f_1 k T_s) \left[(1 + e^{-2\pi^2 f_1^2 |k| \sigma_{clk}^2}\right] \mathrm{d}f_1$$

We deduce the power spectral density of clock jitter for a free-running oscillator :

$$S_{jitter\_clk}(f) = S_{xx}(f) + \overbrace{\int_{-\infty}^{+\infty} S_{xx}(f_1) \frac{(f_1 \sigma_{clk})^2 f_s}{\pi^2 (f_1 \sigma_{clk})^4 f_s^2 + (f - f_1)^2}\, \mathrm{d}f_1}^{Lorentzian\ spectrum} \tag{E.3}$$