HAL
open science

Outils d'aide à la conception de systèmes mixtes analogiques/numériques dédiés à la radio logicielle

Ludovic Barrandon, Samuel Crand, Dominique Houzet

To cite this version:

Ludovic Barrandon, Samuel Crand, Dominique Houzet. Outils d'aide à la conception de systèmes mixtes analogiques/numériques dédiés à la radio logicielle. MajecSTIC 2005 : Manifestation des Jeunes Chercheurs francophones dans les domaines des STIC, IRISA - IETR - LTSI, Nov 2005, Rennes, France. pp.372-376. inria-00000761

HAL Id: inria-00000761

https://inria.hal.science/inria-00000761v1

Submitted on 17 Nov 2005

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Outils d'aide à la conception de systèmes mixtes analogiques/numériques dédiés à la radio logicielle

L. Barrandon, S. Crand, D. Houzet
IETR, Université de Rennes 1, 263 Avenue du Général Leclerc, 35042 RENNES Cedex, France
ludovic.barrandon@univ-rennes1.fr

Résumé : L'émergence de la radio logicielle restreinte (Software Defined Radio - SDR) conduit à développer de nouvelles méthodologies de conception et de prototypage rapide de systèmes de télécommunications mixtes (analogiques/numériques) tout en imposant la reconfigurabilité et la généricité des solutions proposées.
Cette étude consiste en une exploration architecturale focalisée sur deux fonctions principales : la conversion analogique numérique (CAN) et les traitements numériques haute fréquence associés (Tête Numérique ou FEN pour Front-End Numérique) pour un récepteur mobile.
Mots Clés : Radio logicielle, front-end mixte, conception haut niveau, VHDL, CAN.

1 INTRODUCTION

1.1 Contexte

L'évolution des télécommunications mobiles conduit à imaginer des systèmes flexibles tels que la radio logicielle pour répondre aux contraintes de standards distincts [Kountouris 2001]. Afin de favoriser la reconfigurabilité, les convertisseurs analogique numérique (CAN) et numérique analogique (CNA) doivent être placés au plus près de l'antenne. Ainsi, le traitement du signal est principalement effectué en numérique ce qui a pour résultat de fournir un système plus flexible à l'aide de composants numériques reconfigurables. En ce qui concerne la réception, la radio logicielle implique de convertir directement le signal RF en numérique, les CAN devant alors avoir des performances qui n'existent pas à ce jour. Ainsi, le compromis utilisé consiste à transposer le signal RF en fréquence intermédiaire (FI) avant le CAN. Cette solution alternative est appelée radio logicielle restreinte (software defined radio - SDR).
Dans ce contexte, ce travail est axé sur la conversion et les traitements numériques associés appelés front-end numérique (FEN). Le FEN est composé de blocs peu complexes, reconfigurables et travaillant à des fréquences élevées. Il constitue un étage intermédiaire entre la conversion et les traitements numériques en bande de base. La CAN et le FEN, qui représentent un goulet d'étranglement de la SDR, sont étudiés en détail dans ce papier du point de vue réception.

1.2 Méthodologie descendante

L'exploration architecturale est menée sur les CAN et le front-end numérique, ces deux parties composant le front-end mixte (FEM).
Le concept SoC-AMS conduit à rechercher des méthodologies de conception descendantes semblables à la conception numérique et appliquées au monde analogique et mixte. Ce travail consiste en une optimisation haut niveau des CAN pipeline se basant sur l'utilisation d'une fonction de mérite (FDM).
La construction du FEN s'appuie sur un partitionnement modulaire original des différentes fonctions synthétisables en VHDL (filtrage, conversion de fréquences d'échantillonnage). Un outil de synthèse spécifique complète cette étude et permet de faire le lien entre les spécifications et l'implémentation sur FPGA. La mutualisation des résultats concernant la CAN et le FEN est ensuite présentée avant de conclure sur les perspectives de cette étude.

2 EXPLORATION : CAN PIPELINE

2.1 Démarche

L'objectif final n'est pas de fabriquer un CAN mais de mettre au point une méthodologie pour aider à son dimensionnement. L'exploration architecturale est effectuée sans se préoccuper de la technologie de fabrication, ce qui revient à suivre les techniques de conception descendante utilisées dans le domaine numérique. Elle consiste à comparer les différentes solutions possibles de façon relative et exhaustive avant de considérer l'implémentation matérielle. Elle s'appuie sur l'utilisation d'une FDM présentée dans le paragraphe suivant. L'architecture CAN pipeline est celle retenue pour le front-end mixte [Sumanen 2002] : c'est un convertisseur ayant un bon compromis fréquence / résolution et une architecture modulaire (figure 1). Elle se compose d'étages de conversion numérisant leur signal d'entrée et fournissant à l'étage qui les succède un signal d'erreur, c'est-à-dire le reste de l'opération de conversion. Afin de conserver une dynamique satisfaisante le long de la chaîne de conversion, ce signal est amplifié à la sortie de chaque étage. Cette amplification introduit des imperfections (erreur de gain, non linéarités) dégradant les performances du CAN global. Ce point fait l'objet du §2.3.

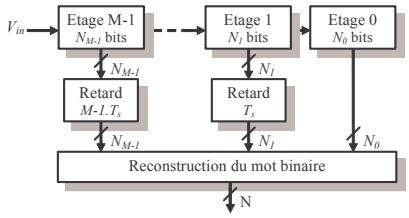


Figure 1 : architecture générale d'un CAN pipeline

2.2 Fonction de mérite

Lors de la conception d'un système complexe, il devient vite évident que le nombre de paramètres entrant en jeu rend difficile la sélection intuitive d'une solution [Vogels 2003]. Deux concepts permettent de franchir cette difficulté : le choix d'un niveau d'abstraction adéquat et l'utilisation d'une fonction de mérite. Il y a donc nécessité de raisonner au niveau structurel auquel l'architecture pipeline se prête bien. L'exploration concernant les CAN pipeline effectuée au niveau « étage de conversion » suit la démarche illustrée figure 2 :

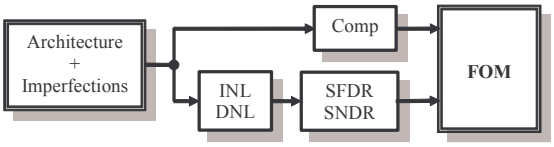


Figure 2 : Méthodologie d'évaluation des performances

La FDM proposée est un exemple mis au point pour la radio logicielle restreinte pour laquelle les performances spectrales (SNDR – rapport signal à bruit et SFDR - dynamique) sont prépondérantes [Loumeau, 2002]. Le nombre *Comp* de comparateurs représente une estimation relative de la surface de silicium à utiliser. L'INL et le DNL correspondent respectivement à l'erreur sur l'intervalle des paliers de conversion d'un CAN et à l'erreur sur la position de ces paliers. D'autres paramètres tels que la surface relative des amplificateurs du signal d'erreur ou l'évaluation de la consommation pourraient éventuellement être considérés :

$$FOM = \delta.\left[\alpha.\left(\frac{SNDR}{SNDR_{\max}}\right)+\beta.\left(\frac{SFDR}{SFDR_{\max}}\right)+\gamma.\left(\frac{Comp_{\min}}{Comp}\right)\right]$$

$\alpha = SNDR_{\lim}/SNDR_{\max}$ $\qquad$ $\gamma = Comp_{\min}/Comp_{\lim}$

$\beta = SFDR_{\lim}/SFDR_{\max}$ $\qquad$ $\delta = 1/[3.(\alpha+\beta+\gamma)]$

δ est calculé pour obtenir $0 < FOM < 1$

Les indices *max* et *min* indiquent les valeurs optimales (cas idéal) et *lim* représente les contraintes limites que le CAN doit respecter pour une application donnée, ce sont des valeurs imposées par le concepteur.

2.3 Calcul

Erreur de gain : La fonction de transfert TF_ε d'un étage de conversion pipeline, prenant en compte l'erreur relative de gain ε_{gain}, est définie à partir de la fonction de transfert idéale *TF* :

$TF_\varepsilon = (1+\varepsilon_{gain})\ .\ TF$

Non linéarité : La non–linéarité de cette même fonction de transfert est modélisée à l'aide d'une fonction tangente hyperbolique inverse [Nuzzo 2003] :

$$TF_{NL} = \frac{1}{\alpha_{NL}}\tanh^{-1}(\tanh(\alpha_{NL}).TF)$$

où α_{NL} est un coefficient associé à l'amplitude de la non linéarité. $1/\alpha_{NL}$ et $\tanh(\alpha_{NL})$ sont des facteurs de normalisation.

Calcul de l'INL : L'INL est la différence entre TF et la fonction de transfert réelle (figure 3).

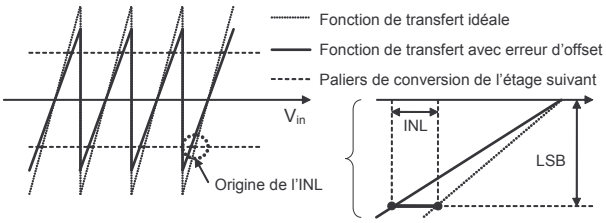


Figure 3 : Influence des erreurs de gain et de non linéarité sur l'INL d'un étage isolé

L'INL d'un étage pipeline non idéal peut s'écrire :

$$INL(i) = \left((1-\varepsilon_{gain})\left(\frac{1}{\alpha_{NL}}\tanh^{-1}(\tanh(\alpha_{NL}).x_i)\right)\right) - x_i$$

où x_i est la position idéale du palier de conversion *i* et *INL(i)* est l'INL correspondant à x_i. Ainsi, l'INL globale d'un CAN pipeline est la somme des contributions de chaque étage pris séparément à cette INL.

Calcul du SNDR à partir de l'INL

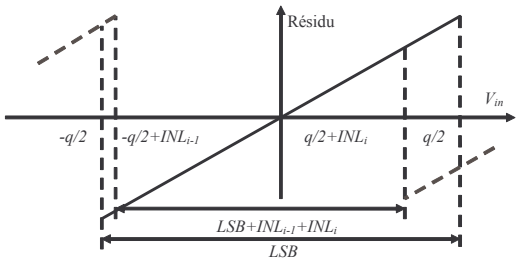


Figure 4 : Erreur de conversion sur le i^{ème} code avec et sans INL

Le SNDR est calculé à partir de l'erreur quadratique moyenne d'une sinusoïde non idéalement numérisée, c'est-à-dire en ajoutant l'INL aux paliers de conversion idéaux (figure 3) :

$$\overline{\varepsilon_i^2} = \frac{1}{q}\int_{-q/2+INL_{i-1}}^{q/2+INL_i} V_{in}^2\, dV_{in} = \frac{1}{3q}\left[\left(\frac{q}{2}+INL_i\right)^3+\left(\frac{q}{2}-INL_{i-1}\right)^3\right]$$

$$= \frac{q^2}{12}+\frac{q}{4}(INL_i-INL_{i-1})+\frac{1}{2}(INL_i^2+INL_{i-1}^2)+\frac{1}{3q}(INL_i^3-INL_{i-1}^3)$$

où $q^2/12$ est le terme classique, égal au bruit de conversion d'un convertisseur idéal. Le SNDR d'un CAN N bits devient alors :

$$SNDR = 20.\log\left(\frac{V_{ref}^2/2}{\overline{\varepsilon^2}/2^N}\right) \quad avec \quad \overline{\varepsilon^2} = \sum_i \overline{\varepsilon_i^2}$$

Calcul du SFDR à partir de l'INL

Dans [Pan 1999], les niveaux des harmoniques d'une sinusoïde numérisée idéalement sont calculés à partir d'un développement en séries de Fourier :

$$a_k = \frac{2}{\pi k}\sum_{i=1}^{2^n} y_i\left[\sin\left(k.\cos^{-1}(x_i)\right) - \sin\left(k.\cos^{-1}(x_{i+1})\right)\right]$$

où les x_i sont les paliers et où $y_i=(x_i+x_{i+1})/2$. L'originalité de ce travail consiste à ajouter le DNL aux paliers de conversion idéaux. Le SFDR est alors l'harmonique la plus élevée issue de l'expression :

$$HD_k \equiv 20.\log\left(\left|a_k/a_1\right|\right) \quad dB_c$$

2.4 Résultats

Dans cette étude, les CAN pipeline utilisent la méthode du bit de redondance (Redundant Sign Digit - RSD) pour permettre une correction automatique [Sumanen 2002]. L'utilisation d'étages 1,5 bit (caractéristique économisant des ressources par rapport aux étages 2 bits pour une même efficacité) est permise, sauf pour le premier étage (pour lequel la correction numérique est inopérante). Le tableau 1 présente les extrema obtenus pour une exploration architecturale exhaustive de CAN pipeline 10 bits (1596 combinaisons) avec les contraintes suivantes (fixées par le cahier des charges ou par le concepteur) :

ε_{gain} = -1.5% ; α_{NL} = 0.2

$Comp_{lim}$ = 60 = 12% de $Comp_{max}$

$SFDR_{lim}$ = 75dB = 88% de $SFDR_{max}$

$SNDR_{lim}$ = 56dB = 90% de $SNDR_{max}$

	Comp	SNR(dB)	ENOB(Bits)	SFDR(dB)	FOM
2/2/2/7	136	49,7	7,97	81,7	**0,77**
2/9	**514**	53,0	8,51	84,5	0,79
2/2/2/2/2/2/2/2	27	**49,4**	**7,91**	79,7	0,85
9/2	**514**	**62,0**	**10,00**	84,6	0,85
2/1.5/2/2/2/2/2/1.5/2	25	51,0	8,17	**79,1**	0,87
2/1.5/1.5/1.5/1.5/1.5/1.5/1.5/2	**20**	52,9	8,49	84,9	0,94
4/1.5/1.5/2/1.5/1.5/2	29	60,0	9,68	**86,7**	0,95
3/1.5/1.5/1.5/1.5/1.5/1.5/2	22	57,5	9,26	85,9	**0,96**

α= 0.909; β= 0.885; γ= 0.333; δ= 0.470

Tableau 1 : Extrema obtenus pour des CAN 10 bits

La configuration 9/2 est celle qui se rapproche le plus de l'architecture flash; ses performances sont donc les meilleures pour le SNDR et le SFDR, mais le nombre de comparateurs est aussi le plus élevé. L'architecture uniquement composée d'étages 2 bits est la moins performante. Néanmoins, grâce à un nombre de comparateurs plutôt faible, cette solution ne conduit pas à la FDM la moins favorable. La meilleure architecture, au regard de la FDM, est un CAN composé d'un premier étage sur N bits avec N un entier dimensionné par rapport à ε_{gain} et à α_{NL} (i.e. N=3 dans ce cas), suivi d'étages 1,5 bit et se terminant sur un étage 2 bits. Cette analyse est confirmée par d'autres valeurs de ε_{gain} et α_{NL}. Ces observations conduisent à construire un modèle VHDL-AMS de CAN pipeline utilisé pour la simulation du FEM complet (cf. paragraphe 4).

3 EXPLORATION : FEN

3.1 Fonctions

Le FEN comprend 3 modules [Hentshel 1999] : démodulation, sélection de canal et conversion de fréquence d'échantillonnage (sample rate conversion-SRC).

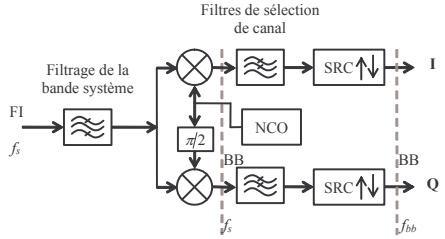


Figure 5 : Architecture générale d'un FEN

La SRC consiste à calculer de nouveaux échantillons $y_a(t_l)$ où $y_a(t)$ approxime le signal d'entrée continu $x_a(t)$.

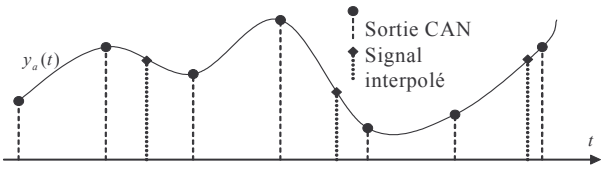


Figure 6 : Illustration de l'opération de SRC

3.2 Filtres utilisés

Les filtres utilisés sont répertoriés dans le tableau 2 [Hentshel 2000]. Ils permettent non seulement d'effectuer du filtrage mais prennent aussi en charge la SRC. La conception de systèmes effectuant une SRC de facteur entier ou rationnel est relativement simple puisqu'elle ne nécessite que l'utilisation de filtres interpolateurs et décimateurs tels que des FIR (Finite Impulse Response) ou des CIC (Cascaded Integrator Comb) [Hogenauer 1981]. Les filtres d'interpolation peuvent assurer une SRC quelconque [Farrow 1988].

3.3 Flot de conception

L'automatisation du flot est illustrée sur la figure 7. A partir du gabarit de filtres à utiliser, du facteur de SRC et du débit des données en entrée, l'objectif et d'obtenir un code VHDL synthétisable du FEN dans son ensemble. Il faut donc déterminer l'architecture la plus compacte (nécessitant le moins de coefficients et donc le moins de multiplieurs) tout en respectant les contraintes de départ.

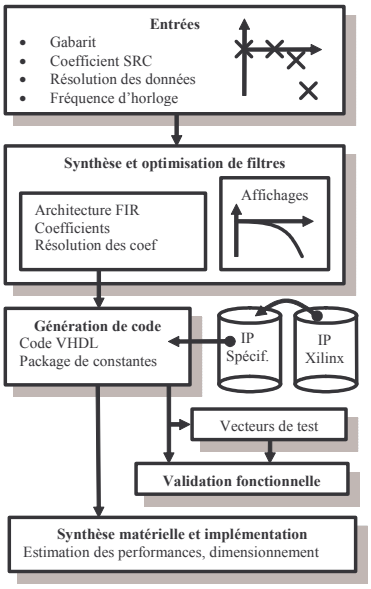


Figure 7 : Flot de synthèse de filtres

Structure	Coefficient SRC	Compacité (multiplieurs)	Conformation de la fonction de transfert	Utilisation
CIC	Entier	++ (pas de coef)	-- (pas de coef)	Difficile à utiliser seul, FT figée
FIR interpolateur / décimateurs	Entier	-	++	Facteurs entiers seulement
Filtres d'interpolation	Quelconque	--	++	Aucune restriction

Tableau 2 : Comparatif des différents types de FIR permettant d'effectuer une SRC

4 SIMULATION DU FEM

4.1 Méthode

La simulation bas niveau (ou niveau transistor) du FEM dans son ensemble est à proscrire du fait du temps de calcul prohibitif pour ce type de travail. L'idée consiste donc à baser l'exploration sur des langages dits de haut niveau. Les temps de simulations sont alors réduits et il est possible de simuler puis de comparer efficacement différentes architectures. Le VHDL et son extension aux signaux analogiques VHDL-AMS sont utilisés pour simuler des modèles comportementaux dans l'espace de conception mixte. De plus, le VHDL est un langage très répandu pour spécifier et implémenter des systèmes numériques dans des FPGA.

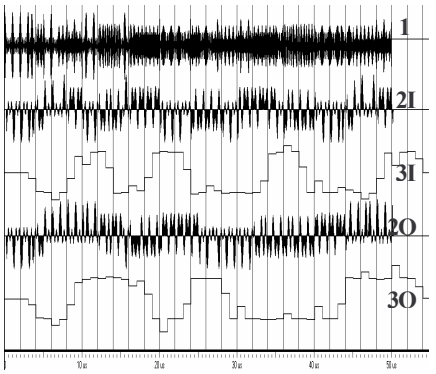
4.2 Exemple

Le FEM est conçu à partir d'une horloge de 100 MHz et d'un CAN 12 bits. Pour la transposition de fréquence, le démodulateur est très simplifié lorsque la fréquence d'échantillonnage est le quadruple de la fréquence intermédiaire du signal converti. Ainsi, dans ce cas particulier la liste des échantillons du NCO est réduite à la séquence: {0,1,0,-1}. Une modulation QPSK avec une porteuse à 25 MHz avec un débit de 5 µs par mot ce qui revient à une largeur de bande de 200 kHz.

Le premier étage est un filtre utilisé pour isoler la bande des fréquences parasites et de l'aliasing. C'est un FIR de 12 étages. Les coefficients sont calculés pour avoir une largeur de bande de 500-kHz autour de la porteuse.

Après transposition, l'opération de SRC est effectuée par des filtres CIC décimateur de facteur 100 à 2 étages.

L'entrée du banc de test et un signal analogique qui est la somme de 5 porteuses à 23, 24, 25, 26 et 27 MHz, avec une modulation QPSK idéale. La simulation globale du FEM est effectuée à l'aide de ADVanceMS [Mentor 2005].



1 : entrée analogique
2 : signal après démodulation
3 : signal après filtrage bande de base et décimation

Figure 8 : Résultats de simulation

5 CONCLUSIONS ET PERSPECTIVES

Le développement de méthodologies de conception d'un front-end mixte appliquée à la radio logicielle restreinte a été présenté. L'exploitation des outils mis au point permettent la simulation et la validation conjointe analogique / numérique, et conduisent au développement de méthodologies systématiques et d'outils de synthèse utilisant une démarche de conception descendante.

BIBLIOGRAPHIE

[Farrow, 1988] Farrow C. W., "A continuously variable digital delay element". Proc. IEEE Int. Symp. Circ. & Syst., Juin 1988, p. 2641-2645, Espoo, Finland.

[Hentshel, 1999] Hentshel T., Henker M., Fettweis G., "The digital front-end of software radio terminals". IEEE Personal Communications, août 1999.

[Hentshel, 2000] Hentshel T., Fettweis G., "SRC for software radio". IEEE Comm. magazine, août 2000.

[Hogenauer, 1981] Hogenauer E. B., "An economical class of digital filters for decimation and interpolation". IEEE Trans. on Acoustics Speech & Signal Proc., 1981.

[Kountouris, 2001] Kountouris A. A., Moy C., Rambaud L., Le Corre P., "A reconfigurable radio case study: a software-based multi-standard transceiver for UMTS, GSM, EDGE and BlueTooth". IEEE Vehicular Technology Conference, Oct 2001, Atlantic City, USA.

[Loumeau, 2000] Loumeau P., Naviner J.F., Petit H., Naviner L., Desgreys P., "Analog to digital conversion : technical aspects" Annales des Télécommunications, Tome 57, No5-6 Mai-Juin 2002.

[Mentor, 2005] www.mentor.com

[Nuñez, 1998] Nuñez-Aldana A., Doboli A., Vemuri R., "A top-down synthesis methodology for behavioral mixed-signal systems specified in VHDL-AMS". Workshop on Design of Mixed-Mode Integrated Circuits and Applications, Guanajuato, 1998.

[Nuzzo 2003] P. Nuzzo, "Architectural Space Characterization of a Transconductance Amplifier », Università di Pisa, 2003".

[Pan, 1999] Pan H., "A 3.3-V, 12-bit, 50-MS/s A/D Converter in 0.6-µm CMOS", Los Angeles, PhD thesis, 1999, University of California.

[Sumanen, 2002] Sumanen L., "Pipeline analog-to-digital converters for wideband wireless communications". Electronic Circuit Design, Report 35, 2002, Helsinki University of Technology.

[Vogels, 2003] Vogels M., Gielen G., "Figure of merit based selection of A/D converters", DATE 2003, p.1190, Munich.