



HAL
open science

Conception d'une architecture de BIST analogique et mixte programmable en technologie CMOS très submicronique

Guillaume Prenat

► To cite this version:

Guillaume Prenat. Conception d'une architecture de BIST analogique et mixte programmable en technologie CMOS très submicronique. Micro et nanotechnologies/Microélectronique. Institut National Polytechnique de Grenoble - INPG, 2005. Français. NNT : . tel-00011327

HAL Id: tel-00011327

<https://theses.hal.science/tel-00011327>

Submitted on 9 Jan 2006

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.

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

N° attribué par la bibliothèque

--	--	--	--	--	--	--	--	--	--

T H E S E

pour obtenir le grade de

DOCTEUR DE L'INPG

Spécialité : Micro et Nano Electronique

préparée au laboratoire TIMA

dans le cadre de **L'Ecole Doctorale ELECTRONIQUE, ELECTROTECHNIQUE, AUTOMATIQUE,
TELECOMMUNICATION, SIGNAL**

présentée et soutenue publiquement

par

Guillaume PRENAT

le 18 novembre 2005

Titre :

**Conception d'une architecture de BIST analogique et mixte programmable en
technologie CMOS très submicronique**

Directeur de thèse :

Bernard COURTOIS

JURY

Mme. Nadine GUILLEMOT
M. Michel RENOVELL
M. José MACHADO DA SILVA
M. Bernard COURTOIS
M. Salvador MIR
M. Laroussi BOUZAI DA
M. Diego VÁZQUEZ

,Président
,Rapporteur
,Rapporteur
,Directeur de Thèse
,Co-encadrant
,Examineur
,Invité

Remerciements

Je tiens tout d'abord à remercier Monsieur Bernard COURTOIS, directeur du laboratoire TIMA, pour m'avoir accepté au sein de son laboratoire, pour m'avoir fait l'honneur de diriger ma thèse et pour m'avoir fourni tous les moyens nécessaires à son bon déroulement.

Je remercie ensuite tout particulièrement Monsieur Salvador MIR pour la qualité de l'encadrement qu'il m'a offert pendant ces trois années, pour sa gentillesse, sa disponibilité et son aide permanente. Grâce à lui, j'ai participé à de nombreuses conférences faisant ainsi connaissance avec la communauté scientifique de mon domaine de recherche.

Ces travaux de thèse ont été réalisés dans le cadre du projet européen EUREKA MEDEA + T101 Technodat. Je remercie ST-Microelectronics, en particulier Messieurs Laroussi BOUZAÏDA et Philippe GARCIN pour leur invitation à participer à ce projet. Je remercie le Ministère de l'Economie, des Finances et de l'Industrie pour leur aide financière dans ce projet.

Je remercie également tous les membres de mon jury pour avoir accepté de s'intéresser à mon travail, d'avoir passé du temps à l'évaluer.

Je remercie sincèrement tous ceux qui m'ont aidé dans mon travail : Luis ROLÍNDEZ, tout d'abord, pour sa contribution essentielle à mon travail, notamment pour les parties analogiques de la technique, dont il est le concepteur, ainsi qu'Ahcène BOUNCEUR pour la partie logicielle. Diego VÁZQUEZ ensuite, pour son apport technique crucial, José MACHADO da SILVA et Lorena ANGHEL pour leur aide concernant les standards de test. Alexandre CHAGOYA, administrateur du CIME, pour son secours fréquent et son support permanent, sans lesquels les petits soucis logiciels auraient probablement handicapé mon travail, Jean-François PAILLOTIN et Kholdoun TORKI pour leur aide précieuse et sans laquelle je n'aurais jamais réussi à envoyer la moindre puce à fabriquer dans les temps, Benoît CHARLOT et Skandar BASROUR pour m'avoir donné le goût de la recherche pendant mon stage de DEA qu'ils ont encadré.

Je remercie ensuite tous les membres du groupe RMS pour leur proximité professionnelle et personnelle, en particulier Libor, Emmanuel, Luis, Ahcène, mais sans oublier tous les autres!

Je remercie aussi toute l'équipe de TIMA pour m'avoir aidé à surmonter toutes les difficultés administratives ou pratiques au cours de ma thèse : Anne-Laure ITIE, Isabelle AMIELH, Sébastien FILLION, Frédéric CHEVROT, Joëlle PARRAU, Isabelle SCHAUNER, Chantal BENIS, Corinne DURAND-VEIL, Lucie TORELLA et tous les autres.

Je remercie bien sûr toute ma famille pour m'avoir toujours encouragé dans mes études en me laissant toute liberté quant à mes choix d'orientation et en m'apportant tout leur support chaque fois que nécessaire.

Je remercie également tous ceux qui ont contribué par leur amitié à faire de ces trois années une période extrêmement agréable, tant sur le plan professionnel que personnel : tous les énergumènes qui se sont succédés dans le tristement célèbre bureau 120 (Benoît, Fabien, Christian, Nicolas, Zein, Sergio, Amel, Marcin, Yasser, Antonio et Paulo), toute la bande du Restau U (Alexandre, Hubert, Benoît, Salvador, Libor, Marcin, Karine, Antonio, Paulo et bien d'autres), mais aussi Lorena, Livier, João, Karine, sans oublier les amis extérieurs à TIMA (Alex, Christophe, Alice, Hoang...).

Et je remercie enfin toutes les personnes qui m'ont aidé à organiser la soutenance de thèse et le pot qui a suivi : mes parents, ma soeur Caroline, ma tante Yvette, Alex, Séverine, Livier et mes collègues de bureau!

Table des matières

Glossaire	15
1 Introduction	17
1.1 Le test de circuits intégrés	17
1.2 Le test de circuits numériques	18
1.3 Le test de circuits analogiques et mixtes	19
1.4 Objectifs du travail de thèse	19
1.5 Structure du mémoire de thèse	20
2 Etat de l'art	21
2.1 Introduction	21
2.2 Modélisation de fautes et métrique de test	21
2.2.1 Types de fautes et de défauts	22
2.2.2 Métrique pour évaluation du test structurel	23
2.2.3 Métrique pour évaluation du test fonctionnel	24
2.3 Techniques générales pour le BIST analogique	25
2.3.1 Le test industriel de cellules analogiques	25
2.3.2 Stratégies complètes de BIST analogique	26
HBIST	26
Technique basée sur l'utilisation d'un DSP	26
HABIST et test de convertisseurs	26
OBIST	27
Test pseudo-aléatoire	28
Test transitoire	29
Utilisation de la redondance matérielle	29
Autres méthodes	30
2.3.3 Génération sur puce de signaux de test analogiques	30
Méthodes basées sur un oscillateur	30
Méthodes basées sur un train binaire stocké en mémoire	32
Autres méthodes	33
2.3.4 Analyse sur puce de la réponse analogique	33
Comparaison de la réponse avec une réponse nominale	33
Utilisation d'un modulateur $\Sigma\Delta$	34
Analyse harmonique de la réponse	35
Autres méthodes	35
2.4 Produits commercialisés au niveau industriel	37
2.4.1 BISTMaxx de Fluence	37
HABIST	37

	VCOBIST	39
	DYNBIST	40
	APBIST	40
	DACBIST	41
	ADCBIST	41
2.4.2	TimeBIST de Vector ¹²	41
2.4.3	Les outils de LogicVision	44
	Outil de test de SerDes	44
	Outil de test de PLL	44
2.5	Conclusion	45
3	Génération des signaux de test analogiques, basée sur la modulation $\Sigma\Delta$	47
3.1	Introduction	47
3.2	Théorie	47
3.2.1	Principe général	47
3.2.2	Génération du train binaire	48
	Principe du modulateur $\Sigma\Delta$	48
	Logiciel de CAO et optimisation	51
3.2.3	Obtention du signal analogique	52
	Répétition du train binaire	52
	Filtrage analogique à capacités commutées	53
3.3	Principe du filtrage analogique	55
	Intégrateur	56
	Filtres de Rauch	56
3.4	Résultats de simulation	59
3.4.1	Génération du train binaire	59
3.4.2	Obtention d'un signal analogique non-différentiel	59
3.4.3	Obtention d'un signal analogique différentiel	60
3.5	Conclusion	61
4	Analyse de la signature basée sur la réponse harmonique	63
4.1	Introduction	63
4.2	Théorie	64
4.3	Considérations pratiques	66
4.3.1	Comptage	66
4.3.2	Modulation par les signaux carrés et calcul de la signature	67
4.4	Optimisation de la signature	69
4.4.1	Paramètres permettant d'optimiser la signature	69
4.4.2	Résultats de simulation	70
4.4.3	Cas linéaire	72
4.4.4	Cas non-linéaire	76
4.5	Conclusion	78
5	Prototypes réalisés	81
5.1	Introduction	81
5.2	Premier démonstrateur : circuit de génération	81
5.2.1	Schéma et résultats de simulation	81
	Partie numérique	81

Partie analogique	83
5.2.2 Layout et finition	85
5.2.3 Préparation du test du démonstrateur	88
5.2.4 Test du banc de filtres	90
5.2.5 Test du circuit complet	92
5.3 Second démonstrateur : architecture de BIST complète	92
5.3.1 Schéma et résultats de simulation	92
Partie numérique	92
Partie analogique	94
5.3.2 Layout et finition	95
5.3.3 Préparation du test du démonstrateur	97
5.3.4 Test de la partie numérique	97
5.3.5 Réparation au FIB	98
5.3.6 Test du circuit de génération	98
5.3.7 Test du circuit d'analyse de signature	100
5.3.8 Test de l'architecture complète	101
5.4 Conclusion	101
6 Compatibilité de l'architecture avec les normes de test	103
6.1 Introduction	103
6.2 Les standards de test numérique (IEEE 1149.1)	104
6.2.1 Le contrôleur TAP	105
6.2.2 Le registre d'instructions	106
6.2.3 Les registres de données	107
Le registre Boundary	107
Le registre Bypass	109
Le registre d'identification	109
Les registres définis par l'utilisateur	109
6.2.4 Les différents modes de fonctionnement de l'architecture	109
Les modes non-invasifs	109
Les modes invasifs	112
6.3 Les standards de test analogique et mixte (IEEE 1149.4)	112
6.4 Compatibilité de l'architecture avec les différents standards	115
6.4.1 Compatibilité avec la norme 1149.1	115
Utilisation dans une architecture IEEE 1149.1 dédiée	115
Utilisation dans une architecture IEEE 1149.1 existante	116
6.4.2 Utilisation avec la norme 1149.4	117
6.5 Conclusion	117
7 Conclusions et travaux futurs	119
A Schémas de certaines parties des démonstrateurs	121
A.1 Circuits analogiques	121
A.1.1 Premier amplificateur	121
A.1.2 Second amplificateur	121
A.2 Circuits numériques	123
A.2.1 Compteur	123
A.2.2 Diviseur programmable	124

B	Layouts de certaines parties des démonstrateurs	127
B.1	BIST complet	127
B.2	Amplificateur différentiel	128
B.3	Filtres à capacités commutées et isolation des parties numériques	128
C	Réparation de la puce par la technique de FIB	131
C.1	Description de la panne	131
C.2	Images de la réparation	131
	Index	135
	Bibliographie	137

Table des figures

1.1	<i>Architecture de la technique de BIST retenue</i>	20
2.1	<i>Schéma générique de test en milieu industriel</i>	25
2.2	<i>Structure de test HBIST</i>	26
2.3	<i>Structure de BIST basée sur l'utilisation d'un DSP</i>	26
2.4	<i>Structure de test HABIST</i>	26
2.5	<i>Exemples d'histogrammes : (b) correspond à la sinusoïde (a) et (d) à la différence entre l'histogramme de (c) et celui de (a)</i>	27
2.6	<i>Structure de test OBIST par Huertas et al.</i>	28
2.7	<i>Structure de test pseudo-aléatoire</i>	28
2.8	<i>Structure de test transitoire</i>	29
2.9	<i>Oscillateur basé sur un résonateur numérique du second ordre</i>	31
2.10	<i>Schéma d'un oscillateur utilisant un modulateur $\Sigma\Delta$ passe-bas</i>	31
2.11	<i>Génération d'un signal de test par stockage d'un train binaire : (a) partie logicielle et (b) partie sur la puce</i>	32
2.12	<i>Modulateur $\Sigma\Delta$ à plusieurs bits</i>	33
2.13	<i>Méthode d'analyse basée sur la comparaison de la réponse analogique avec une réponse de référence</i>	34
2.14	<i>Méthode d'analyse basée sur la création d'une signature numérique à partir de la réponse analogique</i>	34
2.15	<i>Méthode d'analyse basée sur l'utilisation d'une signature obtenue par modulation $\Sigma\Delta$</i>	35
2.16	<i>Méthode d'analyse par comparaison avec un bruit blanc</i>	36
2.17	<i>Architecture du produit HABIST de Fluence</i>	39
2.18	<i>Architecture du produit VCOBIST de Fluence</i>	39
2.19	<i>Architecture du produit DYNBIST de Fluence</i>	40
2.20	<i>Architecture du produit APBIST de Fluence</i>	40
2.21	<i>Architecture du produit DACBIST de Fluence</i>	41
2.22	<i>Architecture du produit timeBIST</i>	43
2.23	<i>Paramètres évalués lors du test par EST</i>	44
2.24	<i>Circuiterie de test avec le module EPT</i>	45
3.1	<i>Vue globale de la technique de génération de signaux de test analogiques : (a) partie logicielle et (b) partie sur puce</i>	48
3.2	<i>Les deux types d'intégrateurs utilisés pour un modulateur $\Sigma\Delta$: (a) un intégrateur simple et (b) un intégrateur avec délai</i>	49
3.3	<i>Schéma d'un modulateur $\Sigma\Delta$ d'ordre 1</i>	49
3.4	<i>Modélisation du bruit de quantification dans un modulateur $\Sigma\Delta$ d'ordre 1</i>	49

3.5	<i>Spectre du bruit de quantification ajouté par un modulateur $\Sigma\Delta$, pour une fréquence d'échantillonnage f_s de 1GHz</i>	50
3.6	<i>Schéma d'un modulateur $\Sigma\Delta$ d'ordre 2</i>	50
3.7	<i>Valeur du THD en fonction de la phase du signal analogique envoyé au modulateur $\Sigma\Delta$</i>	51
3.8	<i>Filtrage d'un train binaire obtenu par modulation $\Sigma\Delta$, par un filtre de fréquence de coupure proche du fondamental</i>	52
3.9	<i>Captures d'écran du logiciel de CAO utilisé pour générer les signaux de test analogiques</i>	53
3.10	<i>Schéma électrique d'une capacité commutée</i>	54
3.11	<i>Horloges non recouvrantes pour capacité commutées</i>	54
3.12	<i>Schéma électrique d'un intégrateur</i>	56
3.13	<i>Schéma électrique d'une structure de filtre de Rauch</i>	57
3.14	<i>Structure de capacités commutées utilisée pour remplacer les résistances dans les filtres</i>	57
3.15	<i>SFDR et THD de trains binaires encodés avec un modulateur $\Sigma\Delta$ du second ordre en fonction de la longueur du registre programmable</i>	59
3.16	<i>Représentations temporelle (a) et spectrale (b) d'un signal généré</i>	60
3.17	<i>Représentations temporelle (a) et spectrale (b) d'un signal généré avec une partie analogique différentielle</i>	61
4.1	<i>Architecture de la technique d'analyse de la réponse analogique du circuit sous test</i>	64
4.2	<i>Fonction de transfert d'un compteur</i>	66
4.3	<i>Comparaison du spectre du signal modulé avant et après le $\Sigma\Delta$: (a) signal avant le compteur et (b) signal après le compteur</i>	68
4.4	<i>Les paramètres permettant d'évaluer la qualité de la signature</i>	69
4.5	<i>Architecture de la technique de BIST avec le filtre passe-bas comme circuit sous test</i>	70
4.6	<i>Espace des paramètres</i>	71
4.7	<i>Espace des performances</i>	71
4.8	<i>Espace des signatures</i>	71
4.9	<i>Répartitions des signatures correspondant: (a) à la valeur DC et (b) au premier harmonique</i>	74
4.10	<i>Signature correspondant aux deux premiers harmoniques</i>	74
4.11	<i>Répartitions des signatures correspondant: (a) à la valeur A2, (b) aux valeurs de DC et A2, (c) aux valeurs de A1 et A2 et (d) aux trois premiers harmoniques</i>	75
4.12	<i>Filtre avec terme non-linéaire</i>	76
4.13	<i>Répartitions des signatures correspondant: (a) à la valeur DC, (b) au premier harmonique, (c) au second harmonique, (d) à la valeur DC et au premier harmonique, (e) à la valeur DC et au second harmonique et (f) aux premier et second harmoniques</i>	77
4.14	<i>Signature correspondant aux trois premiers harmoniques</i>	78
5.1	<i>Schéma du circuit de génération de signaux de test analogiques</i>	82
5.2	<i>Schéma du circuit de génération de phases</i>	83
5.3	<i>Les différentes phases de fonctionnement du circuit de génération de signaux de test analogiques</i>	83
5.4	<i>Schéma de description d'une capacité MIM5</i>	85
5.5	<i>Description d'un interrupteur analogique: (a) connexion typique des substrats des transistors et (b) la connexion que nous avons utilisée</i>	86
5.6	<i>Utilisation d'un anneau n d'isolation pour avoir un substrat connecté à la masse analogique</i>	86
5.7	<i>Isolation des parties analogiques et numériques du démonstrateur</i>	87

5.8	<i>Schéma illustrant la séparation entre les parties analogiques et numériques dans le banc de filtres</i>	88
5.9	<i>Layouts et photos des différentes parties du premier démonstrateur : (a) et (b) montrent respectivement les layouts des deux versions du circuit, (c) et (d) les photos correspondantes et (e) montre une photo de l'ensemble de la puce</i>	89
5.10	<i>Environnement de test du démonstrateur au CIME : (a) le banc de test et (b) un exemple de carte de test disponible</i>	90
5.11	<i>Carte conçue pour faire l'interface entre la carte de test disponible et la puce</i>	91
5.12	<i>Diagrammes de Bode des filtres d'ordre 2 à 6 du premier démonstrateur</i>	91
5.13	<i>Test du premier démonstrateur : (a) signal analogique généré, (b) spectre de ce signal</i>	92
5.14	<i>Schéma du circuit de BIST complet</i>	93
5.15	<i>Layouts et photos des différentes parties du second démonstrateur : (a) et (b) montrent respectivement les layouts des deux versions du circuit, (c) et (d) les photos correspondantes et (e) montre une photo de l'ensemble de la puce</i>	96
5.16	<i>Les cartes conçues pour faire l'interface entre les cartes de test disponibles et la puce pour le second démonstrateur : (a) la version pour les tests préliminaires, (b) la version pour les tests plus précis et (c) cette dernière version ajustée sur la carte du testeur</i>	97
5.17	<i>Comparaison entre le spectre d'une sortie simple et celui du même signal obtenu par simulation</i>	99
5.18	<i>Comparaison entre les deux sorties du filtre prises indépendamment et la sortie différentielle, dans le second démonstrateur</i>	99
5.19	<i>Test du second démonstrateur : (a) signal analogique généré, (b) spectre de ce signal et (c) et (d) les résultats de simulation correspondants</i>	100
5.20	<i>SFDR en fonction de la longueur du registre : résultats du test pour le second démonstrateur</i>	101
6.1	<i>Architecture générique d'un circuit intégré compatible IEEE 1149.1</i>	104
6.2	<i>Diagramme de transition du contrôleur TAP</i>	105
6.3	<i>Architecture d'un registre boundary-scan standard avec une partie capture et une partie maintien : (a) le registre et (b) une cellule du registre avec un élément de capture et un élément maintien</i>	107
6.4	<i>Dessin possible d'un cellule du registre d'instruction</i>	107
6.5	<i>Dessin possible d'une cellule boundary-scan</i>	108
6.6	<i>Schéma détaillé d'une architecture boundary-scan</i>	111
6.7	<i>Architecture générique d'une architecture IEEE 1149.1 dans le cas d'un circuit mixte</i>	113
6.8	<i>Architecture générique d'un circuit intégré compatible IEEE 1149.4</i>	114
6.9	<i>Intégration du BIST dans une architecture IEEE 1149.1 dédiée</i>	115
6.10	<i>Intégration du BIST dans une architecture IEEE 1149.1 existante</i>	116
6.11	<i>Intégration du BIST dans une architecture IEEE 1149.4 : (a) architecture dédiée et (b) architecture existante</i>	118
A.1	<i>Schéma de l'amplificateur utilisé pour le premier démonstrateur</i>	122
A.2	<i>Les différents étages de l'amplificateur du second démonstrateur</i>	122
A.3	<i>Schéma de la structure repliée</i>	123
A.4	<i>Schéma détaillé du compteur programmable</i>	124
A.5	<i>Schéma du diviseur de fréquence programmable</i>	125
A.6	<i>Obtention d'un train d'impulsions de fréquence double de celle de l'horloge</i>	125
B.1	<i>Layout du circuit de BIST complet</i>	127

B.2	<i>Layout de l'amplificateur différentiel utilisé: (a) l'amplificateur et (b) la paire différentielle en peignes interdigités</i>	128
B.3	<i>Layout de l'intégrateur à capacités commutées: (a) l'intégrateur et (b) les transistors de commutation isolés par les anneaux d'isolation</i>	129
C.1	<i>Images expliquant la réparation à effectuer</i>	132
C.2	<i>Images illustrant les différentes étapes de la réparation au FIB</i>	133

Liste des tableaux

2.1	<i>Les différents types de défauts et les fautes en découplant</i>	22
2.2	<i>Les différentes versions de produits BISTMaxx et leurs applications</i>	38
2.3	<i>Les différentes versions de produits de Vector¹²</i>	43
3.1	<i>Exemple de fréquences générables en divisant la fréquence d'horloge par un puissance de deux et en choisissant la taille du registre</i>	54
3.2	<i>Coefficients d'un filtre de Butterworth pour les ordres 1 à 6</i>	56
3.3	<i>SFDR minimums obtenus pour les différents ordres de filtres passe-bas</i>	60
4.1	<i>Les paramètres de qualité du test dans le cas d'un circuit purement linéaire</i>	73
4.2	<i>Les paramètres de qualité du test dans le cas d'un circuit non-linéaire</i>	78
5.1	<i>Les blocs utilisés et leurs valeurs de capacités pour obtenir les différents filtres d'ordres 2 à 6</i>	84
5.2	<i>Surfaces occupées sur le démonstrateur par les principaux blocs</i>	96
6.1	<i>Les différents modes de fonctionnement des cellules boundary-scan</i>	108
6.2	<i>Séquence de scanning</i>	110

Glossaire

ABM	Analog Boundary Module
ABS	ABSolute value
ADCBIST	Analog to Digital Converter Built-In-Self-TEST
APBIST	Analog Probe Built-In-Self-Test
ATAP	Analog Test Access Port
ATE	Automated Test Equipement
ATPG	Automatic Test Pattern Generation
BERT	Bit Error Rate Testing
BIST	Built-In-Self-Test
CAN	Convertisseur Analogique Numérique
CAO	Conception Assistée par Ordinateur
CIME	Centre Inter-universitaire de Micro Electronique
CM	Common MODE
CMFB	Common-Mode FeedBack
CMP	Circuits Multi Projets
CNA	Convertisseur Numérique Analogique
CUT	Circuit Under Test
DACBIST	Digital to Analog Converter Built-In-Self-Test
DBM	Digital Boundary Module
DDFST	Direct Digital Frequency Synthesis Techniques
DFT	Discrete Fourier Transform
DfT	Design for Testability
DNL	Differential Non Linearity
DR	Dynamic Range
DRC	Design Rules Checker
DSP	Digital Signal Processor
DYNBIST	DYNamic Built-In-Self-Test
ENSERG	Ecole Nationale Supérieure d'Electronique et de Radioélectricité de Grenoble
EPT	Embedded PLL Test
EST	Embedded SerDes Test

FFT	Fast Fourier Transform
FIB	Focused Ion Beam
HABIST	Histogram-based Built-In-Self-Test
HBIST	Hybrid Built-In-Self-Test
INL	Integral Non Linearity
IP	Intellectual Property
LFSR	Linear Feedback Shift Register
LIT	Linéaire, Invariant dans le Temps
LVS	Layout Versus Schematic
MADBIST	Mixed Analog-Digital Built-In-Self-Test
MISR	Multiple Input Signature Register
OBIST	Oscillation-based Built-In-Self-Test
PLL	Phase-Locked Loop
RTL	Register Transfer Language
SerDes	Serializer-Deserializer
SFDR	Spurious Free Dynamic Range
SNR	Signal to Noise Ratio
SoC	System-on-Chip
sw-opamp	Switchable opérationnal amplifier
TAP	Test Access Port
TBIC	Test Bus Access Interface Circuit
TDM	Time Division Multiplexing
THD	Total Harmonic Distorsion
VCOBIST	VCO Built-In-Self-TEST
WARGA	Weighted Average Ranking Genetic Algorithm

Chapitre 1

Introduction

1.1 Le test de circuits intégrés

Le test des circuits intégrés est un sujet fondamental aujourd'hui : en effet, le marché de la microélectronique est en perpétuelle évolution et concerne maintenant des applications grand public. Les volumes de production sont de plus en plus importants et il est impératif d'assurer une fiabilité satisfaisante pour un prix abordable. Cela nécessite de développer des techniques de test efficaces et le moins coûteuses possible pour les circuits en microélectronique.

Le but du test est d'éviter les circuits défectueux, c'est-à-dire les circuits qui ne se comportent pas comme ils devraient. Nous appellerons circuits fonctionnels les circuits qui fonctionnent correctement. Les causes de défaillance sont les défauts dans le circuit. Ces défauts physiques vont pouvoir être modélisés par une faute, modèle comportemental du défaut, qui va permettre un traitement dans le domaine du test. Ces fautes vont entraîner une erreur, c'est-à-dire une manifestation locale de la faute. Cette erreur va ensuite entraîner ou non une défaillance du circuit. Le but du test est donc de prévoir, trouver et éliminer les fautes. La première phase va consister à détecter les fautes, la seconde va consister à en localiser les causes. D'une manière générale, le test consiste à envoyer un signal au circuit à tester et à recueillir sa réponse de manière à l'analyser, typiquement à la comparer à la réponse d'un circuit fonctionnant correctement. Pour ce faire, l'idéal serait bien entendu de pouvoir accéder à tous les points du circuit pour pouvoir trouver la faute. Ceci est évidemment impossible dans la pratique, tant la complexité des circuits intégrés est importante. De plus, le nombre limité de plots ne permet un accès physique qu'à une petite partie des points d'un circuit, limitant la testabilité de celui-ci, c'est-à-dire sa capacité à être testé. Pour pallier à ce problème, les concepteurs prennent maintenant en compte les problèmes de test au moment de concevoir le circuit : on parle de DfT.

Les enjeux du test de circuits intégrés sont les suivants :

- Améliorer la qualité du test. On parle de couverture de fautes dans le cas de circuits numériques. Le but est alors de diminuer les erreurs de test, c'est-à-dire le nombre de circuits défectueux passant le test et le nombre de circuits rejetés alors qu'ils sont fonctionnels.
- Améliorer le temps de test.

Ces améliorations ont pour finalité de réduire les coûts du test. En effet, la "loi de 10" indique qu'à chaque étape de la vie d'un produit, le coût résultant de la détection d'une défaillance est multiplié par 10. Ainsi, une erreur détectée une fois la puce emballée coûtera dix fois plus cher qu'au niveau plaquette. Au niveau carte, le prix sera encore multiplié par 10 et à nouveau au niveau système. Si, enfin, la faute est détectée au niveau de l'utilisateur final, le prix sera encore 10 fois

plus élevé, soit 10000 fois plus qu’au niveau plaquette. Il est donc particulièrement important de bénéficier d’un test efficace dès la phase de production du circuit intégré. Il existe plusieurs types de test suivant les étapes de fabrication d’un produit :

- Si l’on teste au niveau prototype, il faut une caractérisation complète du circuit afin de permettre un éventuel retour en conception pour améliorer le produit. Ce type de test est très exhaustif, demande du temps, des ingénieurs de test qualifiés, mais ne concerne qu’un nombre limité de circuits. La vitesse du test n’est pas alors un critère essentiel.
- Si l’on teste au niveau production, l’on peut se contenter d’un test fonctionnel (test consistant à vérifier seulement si la fonctionnalité du circuit est respectée) dans le cas où les processus de fabrication ne sont pas encore matures ou d’un test structurel (test consistant à vérifier l’intégrité “physique” du circuit) dans le cas où la technologie est mature. Ces tests sont donc moins approfondis que la caractérisation, mais concernent un très grand nombre de circuits. Il va donc être intéressant d’automatiser et d’accélérer le test par des techniques de DfT, car le coût devient alors quasiment proportionnel au temps de test.

1.2 Le test de circuits numériques

Dans le domaine numérique, les signaux de test à générer sont binaires, on parle de vecteurs de test. L’analyse de la réponse est simple, car elle consiste simplement à comparer la sortie obtenue à la sortie de référence d’un circuit. Si la sortie obtenue est différente, le circuit présente une défaillance.

Le test de circuits numériques pose cependant des problèmes qui le rendent de plus en plus complexe :

- L’intégration devenant de plus en plus fine, la complexité et le temps de test augmentent de façon exponentielle et le coût du test également. Il pourrait dépasser le prix du silicium dans un proche avenir.
- L’augmentation des fréquences d’horloges permet à des puces de fonctionner à des vitesses que les testeurs conventionnels ne peuvent atteindre. On ne peut pas alors tester ces circuits à vitesse nominale. Cela est d’autant plus gênant qu’à ces vitesses, des parasites peuvent intervenir à cause des connexions entre la puce et les équipements de test.
- Enfin, les délais deviennent suffisamment importants par rapport aux cycles d’horloge pour engendrer des défaillances à haute vitesse. Le test à vitesse nominale est donc aujourd’hui indispensable.

Jusqu’à une période assez récente, les circuits intégrés étaient essentiellement numériques et le test de ces circuits est donc étudié depuis longtemps, les techniques ont aujourd’hui atteint un bon niveau de maturité. La génération de vecteurs de test est faite de manière automatique grâce à de nombreux outils d’ATPG. Concernant l’aspect DfT, deux grands concepts permettent de faciliter énormément le test : les techniques de scan-path et de BIST. Le scan-path permet d’accéder aux noeuds internes du circuit tout en limitant le nombre de plots : on améliore donc l’observabilité du circuit. Des bascules internes de mémorisation sont utilisées pour appliquer les signaux aux différentes entrées. En mode test, ces bascules sont connectées en registre à décalage et le vecteur de test est chargé en série dans celui-ci. Ensuite, on passe au fonctionnement en mode normal, où le vecteur est directement appliqué aux entrées concernées et les sorties stockées à leur tour dans des bascules. On repasse ensuite en mode test, au cours duquel les sorties sont extraites, toujours en mode série. Une norme a été créée en 1990 pour décrire des règles de conception d’une technique de scan-path : la norme IEEE 1149.1, ou boundary-scan. Le BIST consiste à intégrer des outils de

test directement dans le circuit à tester. Typiquement, le BIST évite d’avoir recours à des testeurs externes pour injecter les vecteurs de test. Dans la mesure du possible, l’analyse de la réponse doit se faire aussi sur la puce, évitant tout équipement extérieur. Ceci évite notamment les parasites dus aux connexions entre la puce et les équipements de test. De plus, les cellules de test étant dans la même technologie que le circuit à tester, le test peut se faire à vitesse nominale.

1.3 Le test de circuits analogiques et mixtes

Depuis quelques années, on assiste au développement des SoC, qui font cohabiter sur une même puce des parties numériques, mais aussi des parties analogiques voir des parties RF ou microsystèmes. Bien que les parties analogiques ne représentent qu’une petite partie d’un SoC en terme de surface, la complexité est très importante et le coût du test également. En effet, le test analogique nécessite des testeurs analogiques capables de générer des signaux analogiques avec une très grande précision et sont donc extrêmement coûteux. De plus, l’analyse de la réponse est beaucoup plus complexe dans le domaine analogique que dans le domaine numérique. En effet, dans le cas numérique, la signature est bonne ou fautive, et il suffit de la comparer à la signature théorique pour voir s’il y a ou non une défaillance. Dans le cas d’un circuit analogique, la sortie est toujours sujette à des fluctuations. Le circuit va être refusé si ses caractéristiques s’éloignent trop des caractéristiques voulues. A la notion de fautes (implicitement discrètes), il faut ajouter une notion de variation de paramètres, à celle de détection une notion de tolérance. La tendance actuelle est donc d’essayer de s’inspirer des techniques de test numérique pour le test de circuits analogiques et mixtes. Beaucoup de recherches sont menées dans ce domaine, mais les solutions industrielles sont encore rares et celles qui existent sont très spécifiques (comme nous le verrons dans l’état de l’art). Pourtant, un standard de test analogique existe déjà : il s’agit de la norme IEEE 1149.4 (inspirée de la norme IEEE 1149.1).

1.4 Objectifs du travail de thèse

Le but de ce travail de thèse est de présenter une technique de BIST analogique et mixte totalement intégrée, capable de réaliser sur la puce elle-même la génération de signaux de test analogiques et l’analyse de la réponse analogique du circuit sous test. Cette technique doit être la plus générique possible (dans le sens qu’elle ne soit pas limitée à une classe spécifique de circuits mixtes) et son interface doit être purement numérique afin d’assurer des techniques de test à bas coût. Enfin, cette architecture devra être entièrement compatible avec les normes de test existantes. Un démonstrateur devra être fabriqué pour valider la technique.

Nous nous sommes orientés vers un test fréquentiel basé sur la réponse harmonique du circuit sous test, permettant notamment des mesures de linéarité (figure 1.1). Ce choix est du au fait que l’on peut par cette méthode mesurer le THD qui est un paramètre présentant une très bonne couverture de fautes. Dans cette technique, la génération des signaux de test analogiques et l’analyse de la réponse sont intégrées en utilisant des techniques basées sur l’utilisation de modulateurs $\Sigma\Delta$. La structure peut être divisée en deux parties : la partie de génération de signaux de test et la partie d’analyse de la réponse. Le reste de l’architecture est dédiée au contrôle de l’architecture.

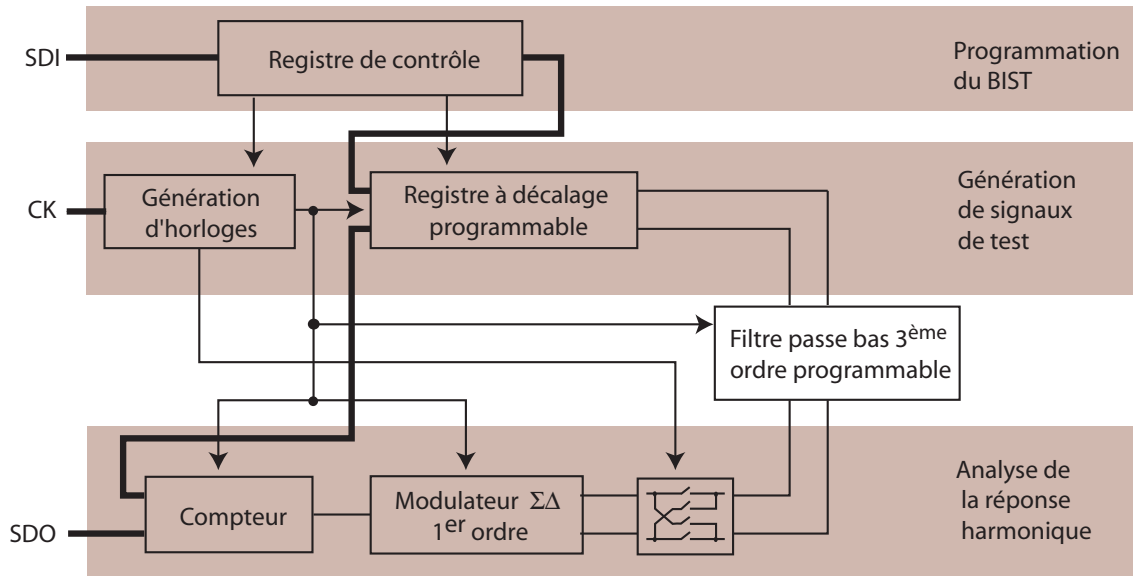


FIG. 1.1 – Architecture de la technique de BIST retenue

1.5 Structure du mémoire de thèse

Le mémoire de cette thèse sera structuré de la manière suivante : tout d'abord, nous ferons un état de l'art des techniques existantes dans le domaine du test intégré analogique. Puis nous étudierons les deux aspects de la technique proposée ici, à savoir la génération de signaux de test analogiques et l'analyse de la réponse analogique du circuit sous test. Ensuite, nous verrons les résultats de test des deux démonstrateurs fabriqués en technologie CMOS 0,18 μm de chez ST-Microelectronics. Enfin, nous verrons comment nous pouvons rendre cette architecture compatible avec les différents standards de test, avant de conclure et d'exposer les travaux futurs à effectuer dans ce domaine. Afin d'en faciliter la lecture, ce document comprend une annexe et les termes techniques sont définis dans un glossaire en début d'ouvrage. Le lecteur se reportera notamment à ce glossaire pour trouver la signification de tous les sigles utilisés.

Chapitre 2

Etat de l'art

2.1 Introduction

Avant de décrire le travail effectué au cours de cette thèse, nous allons faire le point dans ce chapitre sur les techniques de BIST analogiques existantes. Ce chapitre se divisera en trois parties :

- Tout d'abord, nous verrons comment nous pouvons introduire la notion de fautes dans le domaine analogique et les différentes façons d'évaluer la qualité du test dans ce domaine.
- Ensuite, nous exposerons de manière générale les techniques de BIST analogiques ayant été proposées dans la littérature. Cette partie ayant déjà été traitée dans un rapport interne ([93]), nous en reprendrons les grandes lignes en commençant par des considérations générales sur le test industriel des cellules analogiques. Puis nous exposerons les grandes stratégies de BIST analogique en distinguant les stratégies complètes, les stratégies spécifiques à la génération de signaux de test analogiques et les stratégies spécifiques à l'analyse de la réponse du circuit sous test.
- Enfin, la troisième partie décrira les produits de BIST analogique commercialisés.

2.2 Modélisation de fautes et métrique de test

Nous avons signalé en introduction que dans le cas de circuits analogiques, des variations continues de paramètres interdisent d'utiliser la notion de faute telle qu'on l'utilise dans le domaine numérique. En fait, on va parler de fautes catastrophiques, qui sont directement responsables d'une défaillance du circuit (comme les courts-circuits et les circuits ouverts par exemple) et de fautes paramétriques, qui vont causer des variations dans les performances du circuit (comme des variations dans les procédés de fabrication par exemple). Un paramètre important pour évaluer la qualité d'un test est la couverture de fautes, c'est-à-dire le pourcentage de fautes possibles qui vont être détectées par le test. Cette définition est valable dans le cas numérique où les fautes sont dénombrables. La première définition qui a été considérée dans le domaine analogique est le pourcentage du nombre de courts-circuits et de circuits ouverts (c'est-à-dire de fautes catastrophiques) qui peuvent être détectés par le test. Cette définition ne tient pas compte des fautes paramétriques qui sont pourtant la source principale de non-fonctionnement dans le domaine analogique et qui posent le plus de problèmes à la détection. [100] propose donc un ensemble de définitions pour mesurer la qualité d'un test en tenant compte de tous les types de fautes, qu'elles soient catastrophiques ou paramétriques.

2.2.1 Types de fautes et de défauts

[100] raisonne en présentant les différentes combinaisons de défauts et de fautes résultantes, comme présenté dans le tableau 2.1.

Défaut (cause)	Faute (effet)		
	Toutes les performances dans les limites des spécifications	Fautes paramétriques	Fautes catastrophiques
Paramètres de procédé dans les limites de spécifications	Pas de détection et pas de faute	A2	A3
Paramètres de procédé hors des limites de spécifications (localement)	B1	B2	B3
Courts-circuits et circuits ouverts	C1	C2	C3

TAB. 2.1 – *Les différents types de défauts et les fautes en découplant*

Voyons brièvement à quels cas ces différentes combinaisons peuvent correspondre dans la pratique :

- A2 : en théorie, cette situation ne doit pas arriver : si les spécifications des paramètres sont respectées, le circuit doit fonctionner correctement. En réalité, cela peut se produire, car le concepteur ne peut pas tester toutes les combinaisons possibles de paramètres dans toutes les conditions possibles.
- A3 : en théorie, cette situation ne doit pas se produire, encore moins que la situation A2. En effet, même si une combinaison de paramètres corrects peut mener à des performances incorrectes, il est très peu probable d'aboutir à un non-fonctionnement du circuit. Cette situation peut cependant se produire à titre exceptionnel si la conception n'est pas assez robuste (notamment si les marges ne sont pas suffisamment importantes) et les conditions de fonctionnement extrêmes.
- B1 : cette situation est dangereuse, car même si le circuit se trouve fonctionner correctement avec des paramètres incorrects, un problème de fiabilité est à craindre.
- B2 : cette catégorie inclut les fautes paramétriques classiques.
- B3 : cette situation peut se présenter par exemple lorsqu'une variation de paramètre cause une variation dans la tension de seuil d'un transistor qui ne commute alors pas.
- C1 : cette situation peut se présenter dans le cas d'un circuit présentant des redondances matérielles. Une faute catastrophique peut alors être transparente au niveau du circuit mais poser des problèmes de fiabilité.
- C2 : cette situation peut se présenter par exemple dans une chaîne de transistors, si l'un d'entre eux est court-circuité : il peut en résulter un courant d'alimentation excessif.
- C3 : cette catégorie correspond aux fautes catastrophiques classiques.

Pour définir la métrique du test, les auteurs ne tiennent pas compte des cas B1 et C1 car les combinaisons possibles de paramètres sont infinies. De même, l'effet des courts-circuits et circuits ouverts est étudié depuis longtemps. Les auteurs s'intéressent donc aux combinaisons restantes du tableau, correspondant aux défauts paramétriques.

Afin de quantifier les fautes paramétriques, les auteurs utilisent le modèle suivant : chaque paramètre du circuit est considéré comme ayant une distribution donnée, typiquement gaussienne. Un paramètre est alors considéré, tous les autres gardant leur valeur nominale. Ce paramètre varie suivant sa distribution et on considère qu'il y a une faute lorsqu'une performance du circuit n'est plus respectée, à cause de ce paramètre. Autrement dit, on considère qu'il y a une faute sur un paramètre lorsque celui-ci s'écarte suffisamment de sa valeur nominale pour que les performances du circuit ne soient plus respectées, tous les paramètres étant fixés par ailleurs à leur valeur nominale. La probabilité d'occurrence de la faute est la probabilité que le paramètre s'éloigne suffisamment de sa valeur nominale pour qu'il y ait faute. Dans la pratique, les autres paramètres deviennent également. Pour avoir un résultat plus réaliste, on peut avoir recours à des simulations de type Monte Carlo. Ces simulations consistent à générer un grand nombre d'instances en faisant varier de façon aléatoire tous ces paramètres. Si le nombre d'instances est suffisamment important, on peut ainsi considérer que l'on couvre toute la plage de combinaisons possibles. Cette méthode est sans doute la plus précise, mais elle est extrêmement coûteuse en calculs. Une autre approche consiste à effectuer un certain nombre d'opérations à un des noeuds sélectionnés pour déterminer les relations mathématiques entre les performances et les paramètres. Cette approche est moins coûteuse en calculs, mais également moins précise. Enfin, la dernière approche possible est de créer une sorte de base de données des variations de performances mesurées dans un procédé de fabrication et faire une analyse statistique pour déterminer les valeurs moyennes et les variations standards.

Il est très important de noter ici que dans un tel modèle, il est essentiel de connaître les paramètres de la technologie utilisée pour pouvoir évaluer la qualité du test. Ceci n'est pas réellement indispensable dans le cas du test numérique où il n'y a pas de déviations de paramètres.

2.2.2 Métrique pour évaluation du test structurel

La qualité d'un test est mesurée par sa couverture de fautes et le niveau de défaut. Nous avons vu que la définition initiale de la couverture de fautes dans le domaine analogique (ne considérant que les fautes catastrophiques) n'était pas satisfaisante. Les fautes paramétriques n'étant pas dénombrables, [100] propose d'utiliser les modèles de fautes paramétriques précédemment présentés pour redéfinir une métrique de test adaptée aux circuits analogiques. Commençons par donner quelques définitions :

Le rendement (yield en anglais) est la probabilité qu'un circuit soit fonctionnel :

$$Y = \prod_{i=1}^n (1 - p_i^{spec}) \quad (2.1)$$

avec n le nombre de fautes potentielles et p_i^{spec} la probabilité pour la faute i d'apparaître et de provoquer le non respect d'une spécification.

Le rendement de test est la probabilité qu'un circuit passe le test :

$$Y_T = \prod_{j=1}^m (1 - p_j^{test}) \quad (2.2)$$

avec m le nombre de fautes potentielles détectées par le test et p_j^{test} la probabilité pour la faute j d'apparaître et d'être détectée.

Enfin, la probabilité pour un circuit fonctionnel de passer le test :

$$G_P = \prod_{i=1}^n (1 - \max(p_i^{spec}, p_j^{test})) \quad (2.3)$$

avec n le nombre de fautes potentielles.

La couverture de fautes (yield coverage) et les pertes de rendement (yield loss) sont alors définies par :

$$Y_C = \frac{G_P}{Y} \quad (2.4)$$

$$Y_L = 1 - Y_C \quad (2.5)$$

Le niveau de défaut (defect level) est la proportion de circuits défailants dans ceux qui passent le test.

$$\begin{aligned} D &= 1 - \frac{G_P}{Y_T} \\ &= 1 - \frac{Y G_P}{Y_T} \\ D &= 1 - \frac{Y Y_C}{Y_T} \end{aligned} \quad (2.6)$$

Dans [100], ces différentes définitions sont alors comparées aux définitions déjà existantes de ces paramètres en fonction des conditions et des exemples viennent illustrer ces différences et montrer la pertinence de cette approche.

2.2.3 Métrique pour évaluation du test fonctionnel

L'approche présentée précédemment est valable pour le test structurel, c'est à dire où l'on tient compte justement des paramètres physiques du circuit. Dans une approche fonctionnelle, on ne regarde que si le comportement du circuit correspond au comportement prévu, sans tenir compte de ce qui se passe à l'intérieur. Ceci mène à une approche différente, plus empirique. Des paramètres comme la fausse acceptation (circuits défailants ayant passé le test) ou le faux rejet (circuits fonctionnels ayant été rejetés par le test) sont évalués par simulation mais sans tenir compte de la notion de faute. Ceci est l'approche utilisée dans la technique présentée dans cette thèse, comme nous le verrons dans les chapitres suivants : effectivement, nous utilisons un test fréquentiel et nous faisons une analyse harmonique de la réponse ce qui est intrinsèquement une approche fonctionnelle. Afin d'évaluer la qualité du test, nous utilisons donc des simulations de Monte Carlo, comme nous le verrons dans le chapitre 4.

2.3 Techniques générales pour le BIST analogique

Nous allons maintenant présenter ici les principales approches de BIST ayant été proposées dans la littérature ([93]). Comme nous l'avons déjà expliqué en introduction, les difficultés du test de cellules analogiques et mixtes sont dues à plusieurs facteurs ([46]) : la grande variété de circuits existants, le grand nombre de spécifications à tester et la nature même des circuits analogiques dont le comportement dévie de façon continue. Le test devra donc permettre d'évaluer ces déviations pour choisir si un circuit est fonctionnel ou défaillant. Nous avons également signalé en introduction que pour le test de production qui concerne un grand nombre de circuits à tester très rapidement, le BIST est une solution très intéressante. Nous assistons donc depuis quelques années au développement de recherches sur le BIST analogique.

2.3.1 Le test industriel de cellules analogiques

Tous les types de test employés au niveau industriel consistent en l'application d'un signal de test au circuit sous test (CUT) et une analyse de sa réponse, comme le montre la figure 2.1.

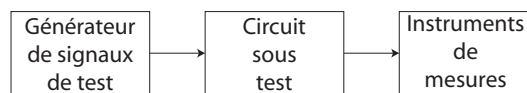


FIG. 2.1 – Schéma générique de test en milieu industriel

On peut classer les méthodes de test en quatre catégories selon le type de stimulus appliqué ([86]) :

- Le test DC consiste à appliquer un signal de test de valeur constante, afin de déterminer notamment l'offset, le gain, les impédances d'entrée-sortie et le biais du circuit. Ce type de test est souvent utilisé en début de conception pour déterminer les variations de biais dues aux déviations dans les procédés de fabrication. Un test de courant I_{DDQ} permet par exemple de détecter les courts-circuits et les circuits ouverts.
- Le test AC utilise des signaux sinusoïdaux pour obtenir une réponse fréquentielle du circuit (gain et phase) et permet de mesurer la distorsion, le rapport signal sur bruit ou l'impédance complexe d'entrée-sortie. Ce test concerne principalement les circuits linéaires, qui donnent une sinusoïde en sortie quand ils sont excités par une sinusoïde.
- Le test transitoire permet, en utilisant un signal de test de type carré ou rampe par exemple, de tester des paramètres tels que les délais, la distorsion. Il peut notamment être utilisé pour les circuits non-linéaires.
- Enfin, on peut réaliser un test de bruit, en connectant l'entrée à la masse pour extraire le bruit interne du circuit.

Ces différents tests utilisent des générateurs de signaux ou des testeurs analogiques pour générer les signaux de test et des appareils tels que des multimètres, oscilloscopes, analyseurs de spectre et autres systèmes d'acquisition de données pour analyser la réponse du circuit. La précision nécessaire pour ces instruments est de 3 bits supérieure à la résolution du circuit sous test ([80]).

2.3.2 Stratégies complètes de BIST analogique

HBIST

Une des premières approches appelée HBIST a été proposée dans [75] (figure 2.2). Elle consiste à générer un signal pseudo-aléatoire par un LFSR. Ce signal est traité puis converti en signal analogique envoyé au circuit sous test. La réponse de celui-ci est convertie en un signal numérique puis compressée en une signature numérique grâce à un MISR, qui va permettre de classer le circuit comme fonctionnel ou défaillant. Cette technique a l'inconvénient de nécessiter un CNA et un CAN qui sont difficiles à implémenter et qui doivent être eux-mêmes testés. Les variations aléatoires présentes dans le circuit analogique ne sont pas prises en compte ([117]).

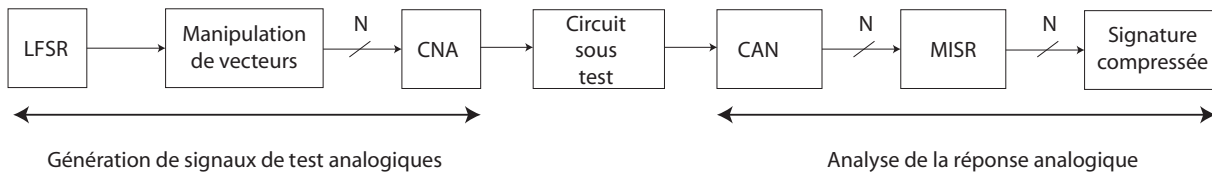


FIG. 2.2 – Structure de test HBIST

Technique basée sur l'utilisation d'un DSP

Une méthode similaire a été proposée, qui utilise un DSP pour générer les signaux de test et analyser la réponse (figure 2.3). Les convertisseurs sont encore présents et doivent être testés, ainsi que le DSP. Une technique appelée MADBIST a été proposée dans [87] dans cette optique.

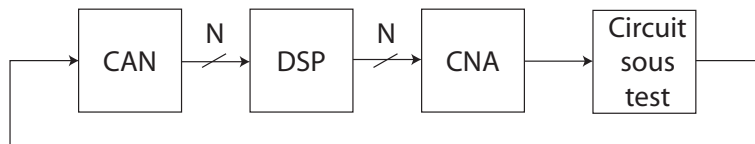


FIG. 2.3 – Structure de BIST basée sur l'utilisation d'un DSP

HABIST et test de convertisseurs

Une des rares techniques à avoir été commercialisées au niveau industriel (chapitre 2.4.1) est la technique HABIST ([28], [106] et [105]) (figure 2.4) ou BIST par histogramme. Appliquée en particulier aux convertisseurs, elle est particulièrement adaptée à des tests nécessitant peu de programmation et une décision de type "fonctionnel/défaillant". Dans cette technique, la réponse du circuit à analyser est convertie grâce à un CAN à n bits (qui présente donc 2^n valeurs de sortie possibles). En comptant le nombre de fois où chaque valeur de sortie apparaît sur un certain nombre de périodes, on obtient un histogramme du circuit dont la différence avec un histogramme de référence (figure 2.5) va permettre de choisir si le circuit est fonctionnel ou défaillant.

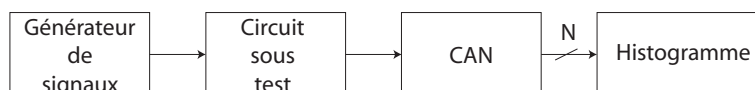


FIG. 2.4 – Structure de test HABIST

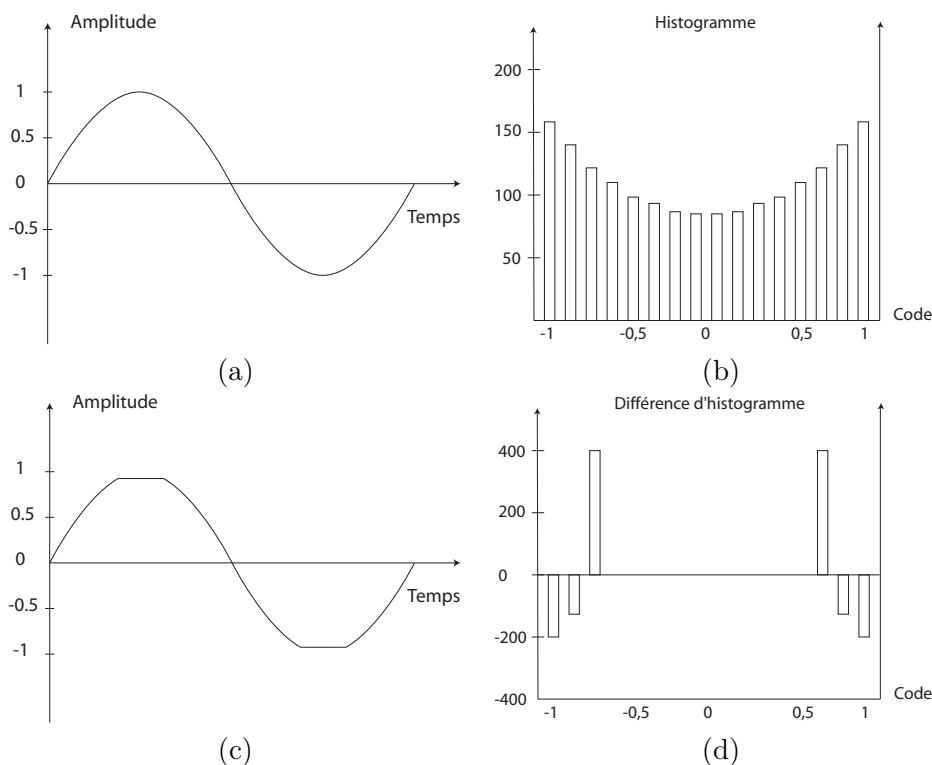


FIG. 2.5 – Exemples d’histogrammes : (b) correspond à la sinusoïde (a) et (d) à la différence entre l’histogramme de (c) et celui de (a)

La technique HABIST permet la mesure de paramètres statiques de convertisseurs, tels que INL ou DNL. D’autres techniques de BIST pour les convertisseurs CNA et CAN ont été présentées dans la littérature, en particulier pour le test dynamique. Par exemple, [64] présente une technique de mesure d’harmoniques dans le cas d’un CAN. Nous ne rentrerons pas ici dans les détails car le test de convertisseurs est le sujet d’une autre thèse dans l’équipe de travail ([95]).

OBIST

Une autre stratégie, appelée OBIST ([2], [3], [4] et [5]) consiste à forcer le circuit sous test à osciller en le modifiant. La fréquence d’oscillation est alors liée aux paramètres du circuit. Une méthode standardisée pour garantir des oscillations entretenues et robustes en utilisant un comparateur dans la boucle de contre-réaction est proposée dans [45] et [46] (figure 2.6).

L’ajout de la mesure de l’amplitude à celle de la fréquence augmente la couverture de fautes. La particularité de cette technique est qu’elle ne nécessite pas de générer des signaux de test. Elle nécessite en revanche des moyens de mesurer l’amplitude et la fréquence des oscillations avec une grande précision ([50], [110] et [109]). Une solution, basée sur le TDM, a été proposée dans [90] pour évaluer la sortie du OBIST. Elle utilise un comparateur dont la tension de référence est modifiable, et balayée périodiquement. Le principal inconvénient de cette technique est le fait que les variations de température, par exemple, ou encore les modifications des paramètres des procédés de fabrication peuvent influencer sur la fréquence d’oscillation et entraîner une mauvaise classification des circuits.

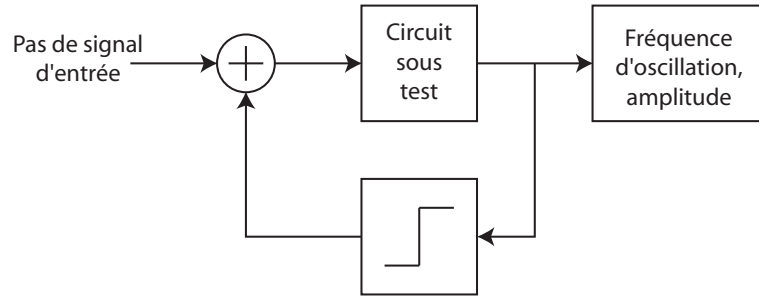


FIG. 2.6 – Structure de test OBIST par Huertas et al.

Test pseudo-aléatoire

Une autre approche consiste à évaluer la réponse impulsionnelle du circuit sous test : si l'on excite un circuit Linéaire et Invariant dans le Temps (LIT) par une impulsion, la corrélation croisée entre son entrée et sa sortie donne un résultat proportionnel à sa réponse impulsionnelle. En effet, la sortie du circuit y sous test est donnée par la corrélation entre son entrée x et sa réponse impulsionnelle h : $y(k) = x(k) * h(k)$. Si l'on calcule la corrélation croisée entre la sortie et l'entrée, on obtient les équations suivantes :

$$R_{xy}(k) = y(k) * x(-k) \quad (2.7)$$

$$= (h(k) * x(k)) * x(-k) \quad (2.8)$$

$$= h(k) * R_{xx}(k) \quad (2.9)$$

où R_{xy} est la corrélation croisée entre l'entrée et la sortie et R_{xx} l'autocorrélation de l'entrée (voir figure 2.7). Dans le cas où l'entrée est une impulsion, R_{xx} est un Dirac et l'on obtient bien un résultat proportionnel à la réponse impulsionnelle. Dans le cadre d'une technique de BIST, un signal numérique pseudo-aléatoire (dont le spectre est proche de celui d'un bruit blanc) obtenu par un LFSR peut être utilisé comme signal de test ([77], [78] et [103]). La réponse du circuit est convertie en une signature numérique qui est constituée des échantillons de la réponse impulsionnelle (figure 2.7).

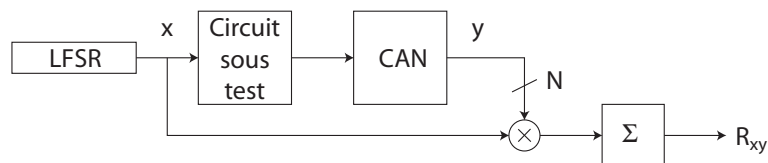


FIG. 2.7 – Structure de test pseudo-aléatoire

Dans le cas de signaux binaires, la corrélation croisée se fait très facilement, car on ne manipule que des "1" et des "-1" et les produits scalaires se résument à une suite d'additions et de soustractions. La réponse est comparée à une sortie théorique stockée dans un dictionnaire de circuits fonctionnels ([63]), ce qui permet de classer le circuit comme fonctionnel ou défaillant. Cette technique a été utilisée pour le test de microsystèmes dans [98] et [20]. Elle peut également être utilisée pour régler sur puce des filtres continus ([18]).

Test transitoire

Les articles [108] et [107] proposent une technique de test transitoire, dans laquelle le circuit est stimulé par une suite d'impulsions générées par un LFSR, à la fréquence f_s . Chaque échantillon de la réponse est comparé à une tension de référence et l'on obtient ainsi une signature numérique (figure 2.8). Le choix de cette tension et de la fréquence d'échantillonnage permet de vérifier au mieux le fonctionnement du circuit. Cette technique n'utilise que des composants numériques, mais elle ne permet que de réaliser des tests transitoires.

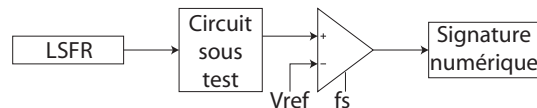


FIG. 2.8 – Structure de test transitoire

Utilisation de la redondance matérielle

Plusieurs techniques utilisant la redondance matérielle sont décrites dans [59]. Le principe le plus simple est de dupliquer le circuit à tester et d'envoyer le même stimulus aux deux circuits pour comparer leurs réponses. La grande surface que nécessite cette technique la rend peu intéressante en pratique. Elle peut cependant s'appliquer avantageusement à certains types de circuits présentant naturellement une redondance matérielle :

- Supposons un circuit contenant N composants identiques. On ajoute un $N + 1^{\text{ième}}$ composant, puis on peut effectuer un test des N modules, dont on compare la réponse au $N + 1^{\text{ième}}$. Ces opérations pouvant être réalisées pour n'importe quel circuit considéré comme circuit "supplémentaire", on a $N + 1$ combinaisons. La performance du test est la même quel que soit le module non-utilisé ([33]).
- On peut aussi tester par cette approche des filtres à capacités commutées et les filtres "à échelle" ([48], [47], [114] et [115]) qui sont constitués d'une succession d'étages doubles. On utilise un bloc biquad programmable permettant de simuler le comportement de chacun des étages qui sont ajoutés à l'architecture. On a ainsi la possibilité de comparer la sortie de chacun des étages au bloc programmable.

Dans ces techniques, la comparaison entre les sorties analogiques peut être réalisée par l'intermédiaire d'un comparateur à fenêtres ([52] et [115]). L'accès aux noeuds internes peut se faire grâce à des "sw-opamp" qui sont des amplificateurs opérationnels qui peuvent fonctionner selon un mode normal et un mode "buffer" pour éviter un composant ([11] et [111]).

D'autres méthodes profitent de la redondance intrinsèque dans les circuits différentiels ([51], [118], [67] et [66]). Dans un circuit différentiel idéal, l'équation suivante doit être respectée dans chaque noeud : $V^+ + V^- = 2V_{bias}$ où V_{bias} est la tension de mode commun. Grâce à un dispositif de vérification analogique, on décide si cette équation est respectée. Cette technique est difficile à mettre en oeuvre, car V^+ et V^- peuvent prendre une grande gamme de valeurs. Dans [58], une solution est proposée pour n'avoir à visualiser que les entrées des amplificateurs opérationnels : si le circuit est fonctionnel, on doit avoir : $V^+ = V^- = V_{bias}$. Cette propriété simplifie le dispositif de vérification analogique. [69] a montré qu'en observant les entrées et les sorties des amplificateurs opérationnels, on peut non-seulement détecter la présence de fautes, mais aussi les diagnostiquer. Ces techniques ne sont évidemment utilisables que pour les circuits différentiels et elles nécessitent un dispositif de vérification analogique. De plus, les fautes symétriques ne peuvent pas être détectées.

Autres méthodes

Une technique basée sur les courbes de Lissajous a été proposée dans [12]. Le principe est de tracer la sortie d'un circuit excité par un signal périodique en fonction de son entrée. On obtient ainsi la courbe de Lissajous correspondante, qui va changer en fonction des variations des paramètres du circuit. On trace également une ligne dite "ligne de contrôle". Le nombre d'intersections entre cette ligne et la courbe va permettre de classer le circuit comme fonctionnel ou défaillant. Cette technique est très efficace en termes de couverture de fautes mais nécessite des composants analogiques délicats à implémenter sur puce.

Une autre méthode consiste à tester des filtres en étudiant leur fonction de transfert ([113]) : supposons que l'on connaisse par avance la fonction de transfert d'un filtre. L'on va pouvoir reconfigurer ce filtre pendant la phase de test pour annuler ses pôles. La sortie va donc être constante et l'erreur sur cette constante va permettre de choisir si le filtre est fonctionnel ou défaillant. La nécessité de connaître parfaitement la fonction de transfert du filtre et de modifier celui-ci est le principal défaut de cette approche. Une approche similaire est proposée dans [14], mais en utilisant un algorithme adaptatif.

Enfin, citons la méthode de [16] qui utilise le calcul des variables d'état du système. Elle utilise des calculs suffisamment gourmands en surface pour la rendre peu intéressante à implémenter.

2.3.3 Génération sur puce de signaux de test analogiques

Hormis la technique OBIST précédemment présentée, toutes les techniques de BIST nécessitent de pouvoir générer un signal de test analogique. L'idéal étant d'éviter les circuits analogiques sensibles, on cherche généralement à utiliser une technique numérique pour générer ces signaux. On peut alors penser à utiliser un DSP pour coder le signal à générer par des techniques de génération numériques de fréquence (DDFST), puis le convertir par un CNA. Cependant, cette approche est coûteuse en surface et donc peu intéressante dans une technique de BIST. Plusieurs autres approches ont donc été présentées, utilisant notamment le modulateur $\Sigma\Delta$ qui permet d'économiser de la surface et surtout de travailler sur un train binaire à un bit de précision, plus facile à traiter qu'un signal numérique classique. Nous allons voir que deux grandes approches ont été présentées :

Méthodes basées sur un oscillateur

La première approche propose d'utiliser un oscillateur numérique pour générer un signal numérique, qui pourra ensuite être converti par un CNA. La première technique d'oscillateur analogique utilisant un modulateur suréchantillonné a été proposée en [56] et [57] (figure 2.9). Cet oscillateur consiste en un résonateur numérique du second ordre, constitué de deux intégrateurs à temps discret. On obtient un signal sinusoïdal de pulsation ω_0 , de phase ϕ et d'amplitude A , ces trois paramètres pouvant être réglés indépendamment, selon les équations :

$$\omega_0 = f_s \cos^{-1} \left(1 - \frac{k_1 k_2}{2} \right) \quad (2.10)$$

$$\phi = \tan^{-1} \left(\frac{x_1(0) \sin \left(\frac{\omega_0}{f_s} \right)}{\left(1 - k_1 k_2 - \cos \left(\frac{\omega_0}{f_s} \right) \right) x_1(0) + k_1 x_2(0)} \right) \quad (2.11)$$

$$A = \frac{(1 - k_1 k_2 x_1(0)) x_1(0) + k_1 x_2(0)}{\sin \left(\frac{\omega_0}{f_s} + \phi \right)} \quad (2.12)$$

K_1 est une puissance de 2, ce qui simplifie l'implémentation car la multiplication par ce coefficient consiste alors seulement en un décalage. Cependant, le coût en surface reste important, à cause de la multiplication $N \times N$ et du convertisseur en sortie.

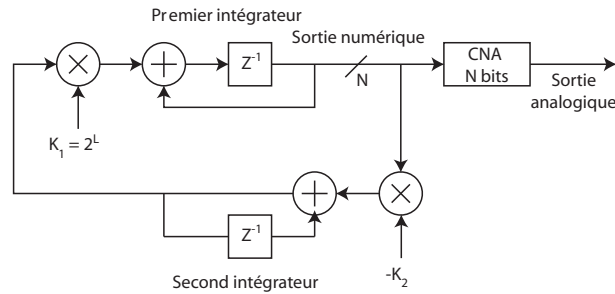


FIG. 2.9 – Oscillateur basé sur un résonateur numérique du second ordre

Une solution pour pallier à ce problème consiste à insérer un modulateur $\Sigma\Delta$ dans la boucle (figure 2.10). Comme la sortie est binaire, la multiplication peut être maintenant facilement réalisée avec un multiplexeur à deux entrées et la conversion en signal analogique peut être réalisée grâce à un convertisseur à un seul bit, qui est beaucoup plus facile à concevoir. L'inconvénient de cette configuration est qu'elle nécessite un filtre passe-bas à la sortie du CNA pour supprimer le bruit de quantification introduit par le modulateur aux hautes fréquences. Des signaux monofréquences de basse fréquence avec un SNR (rapport signal sur bruit) de 70 dB ont été générés par cette technique pour tester un filtre passe-bas de Butterworth d'ordre 6 et un CNA de 1 bit ([57]).

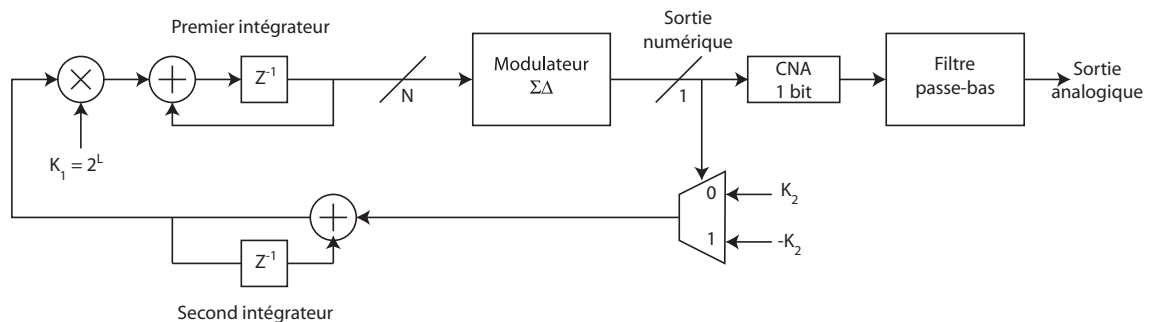


FIG. 2.10 – Schéma d'un oscillateur utilisant un modulateur $\Sigma\Delta$ passe-bas

On peut améliorer le rapport signal sur bruit en augmentant l'ordre du modulateur. Cependant, des problèmes de stabilité apparaissent alors. Pour éviter ces problèmes, une structure plus générale permettant de choisir indépendamment les fonctions de transfert de bruit et du signal est présentée dans [35] et [36]. Un signal basse fréquence avec un SNR de 100 dB a été obtenu en utilisant un modulateur d'ordre 4, mais la distorsion introduite par le CNA de 1 bit et le filtre analogique n'est pas prise en compte ([36]).

Dans [116] une autre configuration permettant de générer des sinusoïdes hautes fréquences est proposée. Des signaux de 250 kHz avec un SNR de 51 dB dans la bande passante ont été obtenus avec un oscillateur passe-bande d'ordre 4 en utilisant un convertisseur numérique-analogique de un bit extérieur à la puce.

On peut obtenir également des signaux multifréquences ([54] et [55]), en additionnant deux sorties monofréquences.

Méthodes basées sur un train binaire stocké en mémoire

Une autre stratégie consiste à stocker dans un registre à décalage un train binaire de longueur N , qui code le signal analogique à générer ([37], [22] et [23]). Ce train binaire peut être obtenu par simulation, en envoyant le signal analogique que l'on cherche à obtenir à un modulateur $\Sigma\Delta$. Le train binaire obtenu à la sortie est celui qui sera envoyé sur la puce. Ce train binaire est ensuite répété pour obtenir un signal périodique. Ce signal peut ensuite être filtré par un filtre passe-bas pour récupérer le signal analogique désiré. L'inconvénient de cette approche est que l'on utilise un train binaire de durée limitée dans le temps en entrée, au lieu d'un signal infini. La résolution de codage de l'amplitude est d'autant meilleure que la longueur du registre à décalage est plus importante. Cette technique sera développée en détails dans le chapitre 3. Le schéma de principe est donné figure 2.11.

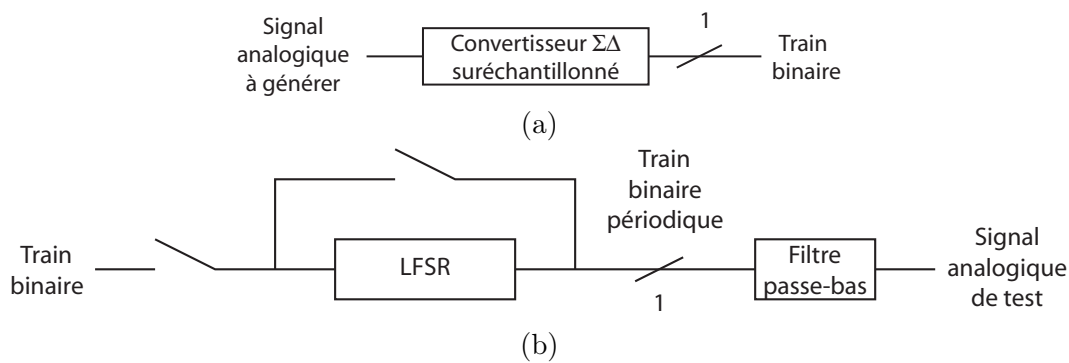


FIG. 2.11 – Génération d'un signal de test par stockage d'un train binaire : (a) partie logicielle et (b) partie sur la puce

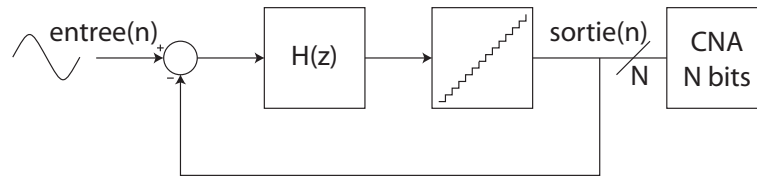
Plusieurs propositions ont été faites pour améliorer cette méthode. Le choix de la phase du signal d'entrée permet d'optimiser le signal de sortie. Dans [38], un signal sinusoïdal de 10 kHz avec 60 dB de SNR a été obtenu avec $N=392$ bits. Dans [24], un SFDR de 70 dB est obtenu avec $N=1024$ bits.

Une amélioration de cette technique est exposée dans cette thèse et décrite dans [65], [70] et [96]. Elle utilise la possibilité de programmer la longueur du registre et la fréquence d'échantillonnage. Cette amélioration permet, avec une longueur de registre moins importante, d'avoir une grande résolution dans le choix de la fréquence du signal analogique. En revanche, la résolution dans le choix de l'amplitude est faible. Nous verrons dans les chapitres 3 et 4 que ceci peut ne pas être un handicap si la technique d'analyse de signature ne nécessite pas une grande précision dans le choix de l'amplitude.

Dans [41], des modulateurs d'ordre très haut sont utilisés pour augmenter la bande passante du signal. Un CNA basé sur un interrupteur de courant haute vitesse est proposé pour supprimer la distorsion dans la reconstitution du signal analogique.

Une autre possibilité, toujours en utilisant le même genre de technique, est de faire appel à un modulateur $\Sigma\Delta$ avec un quantificateur à plusieurs bits (figure 2.12). Théoriquement, chaque bit supplémentaire ajoute 6 dB au SNR. Mais les non-linéarités du CNA multi-bits ont plutôt un effet néfaste sur la qualité du signal. Dans [23], [25] et [43], une technique de calibrage a été proposée pour pallier à ces problèmes, mais ce calibrage est difficile à réaliser sur la puce.

Cette stratégie de génération de signaux de test a été utilisée dans de nombreuses techniques de

FIG. 2.12 – Modulateur $\Sigma\Delta$ à plusieurs bits

BIST ([30], [29], [43], [40] et [81]).

Autres méthodes

D'autres méthodes font appel à un oscillateur harmonique intégré. Dans [34], un oscillateur CMOS à temps continu permet la génération de signaux triangulaires, carrés et sinusoïdaux. Un oscillateur basé sur un trigger de Schmidt a été proposé dans [60] et [19]. Des impulsions et des signaux en dents de scie peuvent être obtenus moyennant l'ajout de composants ([61]). D'autres méthodes ont été proposées, notamment dans [68] ou [50], mais les résultats permettant de caractériser la qualité du signal en termes de SFDR, SNR ou THD manquent.

Un autre champ d'investigation est la génération de rampes très linéaires pour des applications de BIST, et notamment pour les stratégies de type histogramme. Un générateur de rampes calibré a été présenté dans [82], aboutissant à des signaux de neuf bits de précision. Cette technique a été revue par la suite pour obtenir onze bits de précision ([83]). La précision atteint quinze bits dans [6], mais aucun résultat expérimental n'a été présenté. Dans [42], les résultats expérimentaux donnent des rampes avec une linéarité de douze bits. De la même manière, des signaux triangulaires ont été générés pour le test histogramme de CAN dans [8]. La précision est supérieure à douze bits, mais aucun résultat expérimental n'est présenté.

2.3.4 Analyse sur puce de la réponse analogique

Comme nous l'avons dit précédemment, le problème pour analyser une réponse analogique est qu'elle n'est pas déterministe par nature : jamais une réponse ne sera exactement la réponse attendue. Il faut donc trouver un moyen de comparer deux réponses au sens "estimer l'écart entre ces deux réponses" avec une certaine tolérance.

Comparaison de la réponse avec une réponse nominale

Une première idée pour évaluer la valeur de la réponse est de calculer la sortie nominale, de la stocker de manière numérique puis de la générer sur la puce grâce à un CNA (figure 2.13). Cette sortie idéale est ensuite comparée à la sortie réelle grâce à un comparateur analogique. Cette technique a été proposée dans [34]. Un CNA et un comparateur analogique sont nécessaires, rendant cette méthode difficile à implémenter.

L'autre approche consiste à convertir la réponse analogique à analyser en une réponse numérique. Cette réponse peut alors être traitée numériquement, ou comparée numériquement à une réponse de référence. Un convertisseur est toujours nécessaire, mais la comparaison se fait numériquement.

Dans [71] et [72], par exemple, le signal analogique est converti en numérique grâce à un CAN. Les échantillons sont intégrés sur un certain nombre de périodes, donnant ainsi une signature proportionnelle à l'amplitude du signal sinusoïdal. Cette signature est comparée numériquement à une signature correcte (figure 2.14). Le principal inconvénient de cette technique est qu'elle postule que

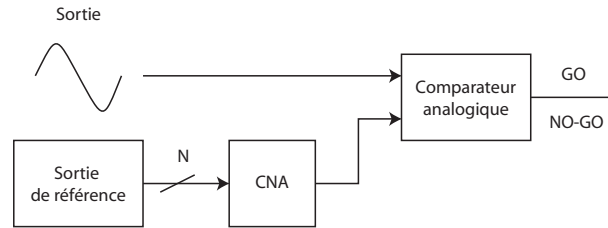


FIG. 2.13 – Méthode d'analyse basée sur la comparaison de la réponse analogique avec une réponse de référence

le circuit sous test est linéaire (on se contente d'évaluer la valeur du fondamental). Mais si le circuit ne fonctionne pas correctement, il se peut qu'une distorsion apparaisse, qui se traduira par l'apparition de nouveaux harmoniques, mais éventuellement sans changer notablement l'amplitude du fondamental. Certaines erreurs dans le circuit peuvent donc ne pas être détectées.

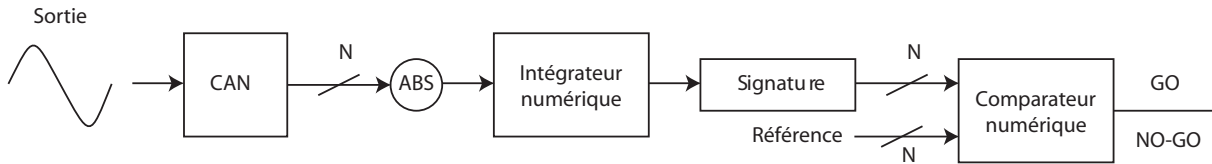


FIG. 2.14 – Méthode d'analyse basée sur la création d'une signature numérique à partir de la réponse analogique

Pour pallier à ce problème, une solution a été proposée dans [89] et [91] : le signal est décomposé en différentes bandes de fréquences, et le même traitement que précédemment est appliqué à chaque bande. On obtient ainsi un certain nombre de signatures couvrant une plus grande gamme de fréquences et la distorsion est détectée. Cette technique ne nécessite que des composants numériques supplémentaires et le principal problème reste la présence du CAN.

Utilisation d'un modulateur $\Sigma\Delta$

Le modulateur $\Sigma\Delta$ est un dispositif suréchantillonné qui génère un train binaire à partir d'un signal analogique. La sortie n'a qu'un seul bit de précision et c'est la fréquence instantanée de "1" et de "0" qui va coder la valeur du signal analogique à cet instant. Dans un tel dispositif, le bruit de quantification est repoussé vers les hautes fréquences. Nous verrons plus en détails le fonctionnement du modulateur $\Sigma\Delta$ dans le chapitre 4. Un tel dispositif peut donc être utilisé pour convertir la réponse analogique car le train binaire est plus facile à analyser qu'une réponse numérique classique à plusieurs bits de résolution. En revanche, cette technique n'est pas adaptée aux dispositifs fonctionnant à très haute fréquence, à cause du suréchantillonnage.

Dans [39] et [40], deux modulateurs $\Sigma\Delta$ du premier et du second ordre sont utilisés. Ils fournissent respectivement 65 et 87 dB de DR dans la bande de fréquence 0 à 20 kHz du modulateur. Le bruit de quantification introduit par le modulateur étant négligeable par rapport au bruit interne introduit par le circuit sous test, il est possible de mesurer le bruit de substrat. Dans cette méthode, l'analyse de la signature se fait en dehors de la puce. Dans [43], l'analyse se fait sur la puce grâce à une transformée de Fourier discrète, ce qui nécessite encore une grande surface de circuit.

La modulation $\Sigma\Delta$ est aussi utilisée dans [15] (figure 2.15). Les parties analogiques sont reconfigurées pour effectuer elles-mêmes la modulation. Le circuit est excité par une impulsion, la

sortie du modulateur ainsi créé est filtrée par un filtre anti-recouvrement et les valeurs absolues des cinq premiers échantillons sont additionnées pour former la signature. Plusieurs tests sont effectués avec différentes fréquences d'échantillonnage pour éviter les problèmes de recouvrement. L'introduction de modifications dans le chemin du signal, ainsi que la mauvaise précision obtenue, sont les principales limites de cette technique.



FIG. 2.15 – Méthode d'analyse basée sur l'utilisation d'une signature obtenue par modulation $\Sigma\Delta$

Analyse harmonique de la réponse

Une technique d'analyse harmonique de la réponse a été proposée dans [112]. Elle consiste à effectuer une sorte de transformée de Fourier du signal de sortie qui a été préalablement transformé en train binaire par un modulateur $\Sigma\Delta$. Pour éviter d'avoir à générer sur la puce les signaux sinusoïdaux nécessaires à la transformée de Fourier, on utilise des signaux carrés. Cette technique permet d'évaluer numériquement les différents harmoniques du signal à analyser. Cependant, le fait d'utiliser des signaux carrés plutôt que des signaux sinusoïdaux pour la modulation introduit des harmoniques impairs dont il faut tenir compte dans l'évaluation des harmoniques. Des résultats de simulation ont donné un DR jusqu'à 80 dB. Cette méthode est illustrée sur la figure 4.1 du chapitre 4 et sera expliquée beaucoup plus en détails dans ce chapitre.

Autres méthodes

Plusieurs méthodes tirent parti de la nature répétitive des signaux analogiques. Dans [53], la méthode consiste à obtenir des échantillons équivalents à ceux obtenus avec une très haute fréquence d'échantillonnage, en utilisant une succession de convertisseurs à approximation, et en contrôlant la période d'échantillonnage et la tension de seuil du comparateur. On peut ainsi, en tenant compte de la périodicité du signal analogique, obtenir des captures de différentes périodes du signal et en tirer toute l'information ([62]).

Une approche similaire, appelée conversion analogique-numérique multi-pass, a été présentée dans [32] et [31]. Un échantillonneur-bloqueur capture N échantillons de chaque période du signal. Un comparateur, avec 2^M tensions de référence programmables, détecte les correspondances entre le signal et la tension de référence. Cette tension est modifiée pour chaque période du signal, 2^M périodes sont donc nécessaires pour collecter toute l'information du signal. Cette technique est très intéressante en ce qui concerne la surface de silicium, mais elle présente plusieurs inconvénients :

- La fréquence d'échantillonnage est limitée par celle de l'échantillonneur-bloqueur
- La résolution en amplitude est limitée par la tension d'offset du comparateur
- Il faut générer sur la puce 2^M tensions de référence

Dans [30] et [29], des tensions de référence sont générées sur la puce grâce à un train binaire créé par un modulateur $\Sigma\Delta$, la valeur DC étant représentée par la moyenne de la séquence périodique.

Une autre stratégie est présentée dans [74] et [73] (figure 2.16). La sortie est comparée à un bruit blanc par un comparateur analogique pour obtenir un train binaire qui peut être analysé de façon numérique. En effet, il a été montré que :

$$R_y[k] = \frac{2}{\pi} \arcsin\left(\frac{R_x[k]}{R_x[0]}\right) \quad (2.13)$$

$$R_x[k] = R_{x,signal}[k] + R_{x,bruit_blanc}[k] \quad (2.14)$$

$$R_{x,bruit_blanc}[k] = \frac{\alpha}{2} \delta[k] \quad (2.15)$$

où R_y et R_x sont respectivement les auto-corrélations de la sortie et de l'entrée de l'amplificateur. $R_{x,signal}$ et $R_{x,bruit_blanc}$ sont les deux composantes de l'auto-corrélation de l'entrée x , correspondant aux deux entrées de l'amplificateur. Comme la fonction arcsin peut être approchée par une fonction linéaire au voisinage de zéro, on voit que R_y est proportionnelle à la somme de $R_{x,signal}$ et d'un bruit blanc :

$$R_y[k] \simeq \frac{2}{\pi} \frac{1}{R_x[0]} \left(R_{x,signal}[k] + \frac{\alpha}{2} \delta[k] \right) \quad (2.16)$$

Le spectre pouvant être calculé à partir de l'auto-corrélation, on peut extraire le spectre de la sortie sans avoir à utiliser un CAN. Cette technique nécessite un générateur de bruit blanc sur la puce, et un DSP pour analyser la réponse analogique. De plus, l'implémentation du comparateur analogique, qui doit fonctionner à très haute fréquence, peut nettement détériorer la qualité du BIST.

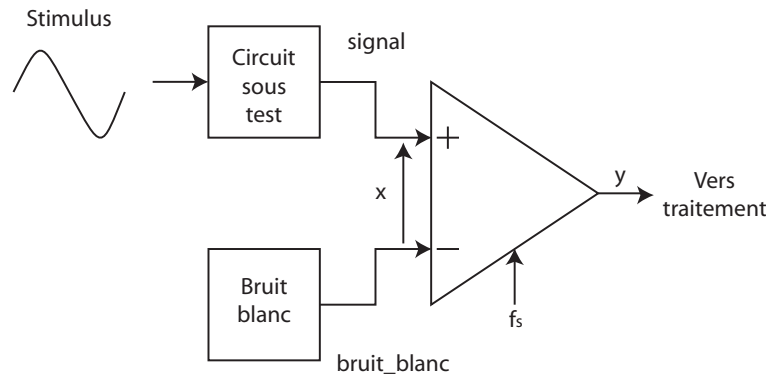


FIG. 2.16 – Méthode d'analyse par comparaison avec un bruit blanc

D'autres solutions ont été proposées, utilisant des composants analogiques supplémentaires. Dans [84] et [85] le circuit est stimulé par une impulsion et sa réponse est compressée par un simple intégrateur. Dans [60], [84] et [19] on utilise une double intégration pour compresser la réponse en signal sinusoïdal. Cette technique a été améliorée dans [13] par l'utilisation d'un calibrage en température du double intégrateur pour s'affranchir de toute influence de la température. Dans [99], des paramètres comme le gain ou la phase sont codés en une tension constante par une circuiterie analogique. Cette tension peut ensuite être utilisée comme signature. Il n'y a pas besoin de convertisseurs dans ces techniques. Elles utilisent cependant des composants analogiques supplémentaires, dont les imprécisions affectent la qualité du BIST.

2.4 Produits commercialisés au niveau industriel

2.4.1 BISTMaxx de Fluence

Fluence propose un ensemble d'outils, appelé BISTMaxx, permettant d'effectuer différents types de tests sur des composants analogiques, mixtes ou numériques (www.fluence.com, [26]). Ces outils peuvent se présenter sous quatre différentes formes :

- Des éléments sous forme d'IP, qui nécessitent une part de personnalisation de la part de l'utilisateur. Il sont liés de près au circuit à tester.
- Des blocs micro-instruments de fabrication de BIST, à usage générique et qui requièrent peu ou pas du tout de personnalisation.
- Ces mêmes micro-instruments disponibles auprès du fondeur sous forme de cellules standards.
- Enfin, des solutions distribuées, qui peuvent être implémentées sur un FPGA ou sur une carte d'ATE. Si la qualité du test est moins bonne que pour les solutions embarquées sur la puce, elles permettent des modifications, réparations et adaptations plus faciles.

Ces outils sont fournis avec un Design Kit, qui contient :

- Le coeur numérique, qui peut être synthétisé en VHDL ou Verilog. Il contient toutes les parties numériques de l'architecture, y compris le registre de contrôle dont nous reparlerons plus tard.
- Un manuel de conception, qui aide à la conception de l'architecture de BIST. Il propose notamment des exemples de conception pour les blocs analogiques.
- Un logiciel de test pour piloter l'architecture.
- Un logiciel de diagnostic (en option) pour aider à faire un diagnostic précis de la panne.

Les avantages procurés par BISTMaxx sont les suivants :

- Une stratégie de test distribuée, complète et adaptée aux besoins du concepteur
- Un faible coût en surface de silicium
- Des temps de test courts
- La compatibilité avec les standards de test existants, comme IEEE 1149.1
- Des coûts de test réduits

Le tableau 2.2 montre les symboles logiques des différents produits, ainsi que leurs domaines d'application. Nous allons maintenant décrire rapidement ces différents produits :

HABIST

Cette technique de test par histogramme a déjà été décrite de manière générale dans la première partie de ce chapitre. La figure 2.17 montre la version de cette technique proposée par Fluence. Un échantillonneur bloqueur et un CAN assurent la conversion du signal analogique en signal numérique et une matrice de compteurs permet de compter combien de fois chaque valeur de sortie apparaît pour créer l'histogramme du circuit sous test.

Produit	Schéma logique	Fonctions
HABIST		<ul style="list-style-type: none"> - Test analogique complet - Résultat compressé en une signature numérique - Mesures des erreurs de gain, d'offset, de la distorsion des niveaux logiques et des fronts
VCOBIST		<ul style="list-style-type: none"> - Mesure RMS de la gigue - Mesure pic à pic de la gigue - Mesures de délais - Applications aux VCO's, PLL's, télécommunications, réseaux haute vitesse - En option : délai de phase, et phase de verrouillage
DYNBIST		<ul style="list-style-type: none"> - Compteur de fréquence - Mesures sur VCO's PLL's, multiplieurs d'horloge, test oscillatoire numérique
APBIST		<ul style="list-style-type: none"> - Multiplexeur hautes performances 4,8 ou 16 entrées - Jusqu'à deux sorties analogiques haut-courant - Impédances d'entrées très faibles - Impédance de sortie réglable entre 25 et 100 ohms - Sert de tampon entre les signaux internes et les équipements externes - En option : circuit de correction d'offset
DACBIST		<ul style="list-style-type: none"> - Caractérisation complète du CNA - Mesures INL, DNL, linéarité, monotonie, bits effectifs, rapport signal sur bruit, erreurs de gain et d'offset
ADCBIST		<ul style="list-style-type: none"> - Caractérisation complète du CAN - Mesures INL, DNL, linéarité

TAB. 2.2 – Les différentes versions de produits BISTMaxx et leurs applications

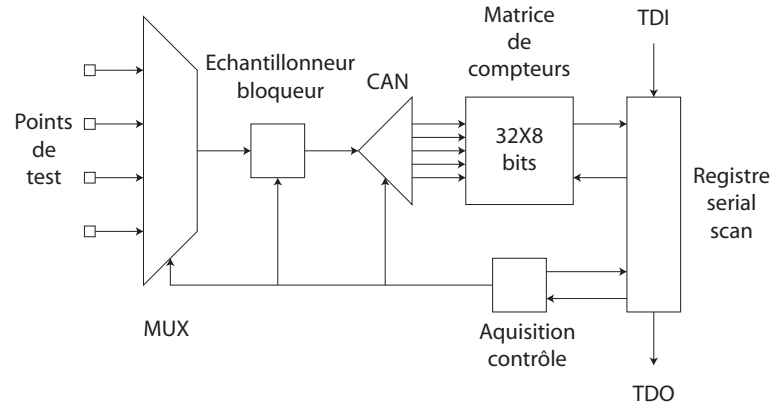


FIG. 2.17 – Architecture du produit HABIST de Fluence

VCOBIST

Cette technique permet des mesures de gigue et de délais. Elle permet de mesurer la durée entre deux fronts d'horloge successifs, et donc la période d'un signal. En effectuant plusieurs mesures au cours du temps, on peut voir si cette période varie et si oui de combien (mesure de gigue). On peut de la même façon mesurer le temps entre deux fronts de deux signaux différents pour estimer le délai entre les deux. Le principe (figure 2.18) est de fabriquer deux chaînes d'oscillateurs fonctionnant quasiment à la même fréquence. On déclenche le premier oscillateur au temps zéro, puis le suivant est déclenché sur le front suivant du premier. Les sorties des deux oscillateurs sont connectées à des compteurs, qui comptent le nombre de fronts par unité de temps et donc mesurent la fréquence. La mesure s'arrête lorsque les deux oscillateurs sont à nouveau en phase. L'avantage d'utiliser les deux chaînes d'oscillateurs et de mesurer la différence entre les deux est que beaucoup des sources d'erreurs en mode commun sont éliminées. Avec cette méthode, la précision de mesure de la gigue peut atteindre cent picosecondes sur FPGA et dix picosecondes sur puce. Cette technique peut être utilisée avec des éléments de HABIST pour enregistrer les résultats de test dans la mémoire consacrée à l'histogramme. Ceci permet d'améliorer grandement le temps de test.

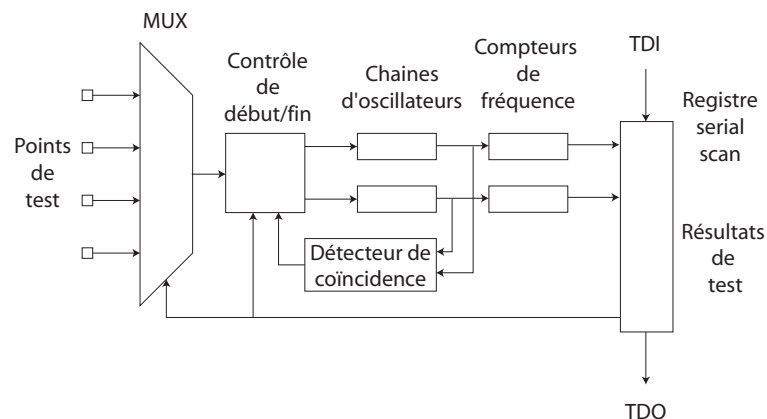


FIG. 2.18 – Architecture du produit VCOBIST de Fluence

DYNBIST

Cette technique utilise un simple compteur pour mesurer la fréquence, en comptant le nombre de fronts pendant une période donnée. Le schéma de principe est donné sur la figure 2.19. L'utilisation la plus intéressante est le test numérique oscillatoire : une boucle de rétroaction connecte la sortie d'un bloc combinatoire à une des entrées. En mode test, ce bloc peut être configuré pour former un oscillateur dont la fréquence d'oscillation est une mesure du temps total de propagation à travers le bloc.

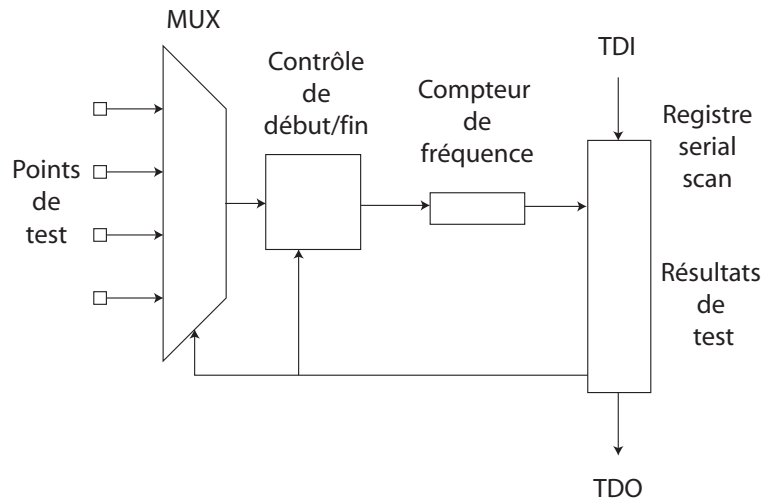


FIG. 2.19 – Architecture du produit DYNBIST de Fluence

APBIST

C'est une système de sondes qui permettent d'extraire des signaux de la puce pour les communiquer au monde extérieur. L'impédance de ces sondes est programmable pour être adaptée à la sortie observée et le circuit permet de compenser en temps réel et en continu les déviations dues aux variations de température et de procédé. L'IP APBIST est un multiplexeur analogique de haute performance. Il permet de choisir quel signal sera envoyé à sa sortie pour être observé. Le schéma est donné figure 2.20.

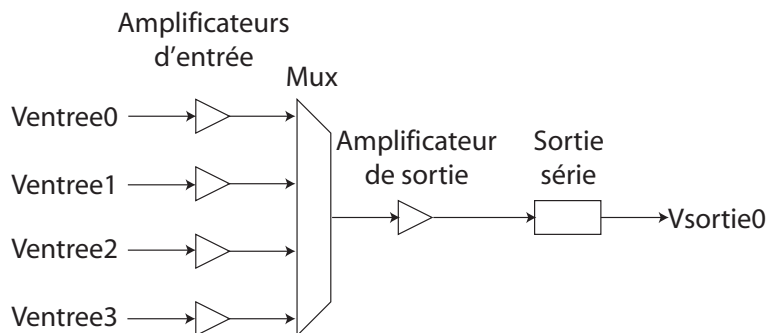


FIG. 2.20 – Architecture du produit APBIST de Fluence

DACBIST

Cette technique permet de mesurer les paramètres donnés dans le tableau 2.2 et de tracer la fonction de transfert du CNA sous test. Son fonctionnement est le suivant (figure 2.21) : à la sortie du convertisseur à tester, un amplificateur opérationnel et un comparateur forment un convertisseur analogique numérique à 1 bit. En entrée du convertisseur à tester, l'on peut entrer deux codes numériques, encodant l'un un signal de valeur supérieure à la masse analogique, l'autre une valeur inférieure. Quand le signal codant la valeur supérieure est chargé, la sortie de l'amplificateur augmente pour atteindre la valeur de seuil du comparateur, qui bascule alors, chargeant le signal codant une valeur inférieure dans le CNA. On obtient ainsi un signal oscillant à la sortie du CNA sous test, et la mesure de la fréquence des ces oscillations permet de caractériser ce dernier. Ce protocole est répété pour différentes valeurs des codes, et on peut ainsi tester le CNA sur ses différentes plages de fonctionnement.

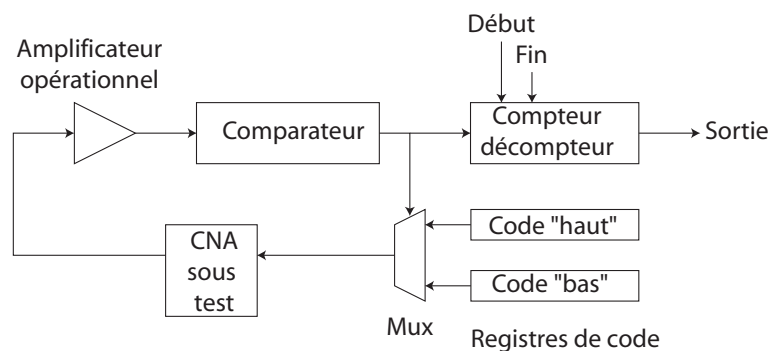


FIG. 2.21 – Architecture du produit DACBIST de Fluence

ADCBIST

L'IP ADCBIST contient une version toute numérique de HABIST pour collecter les données numériques issues du CAN. Ceci permet de simplifier énormément la collecte de données. Une comparaison entre le signal analogique pilotant le CAN et ses différentes représentations numériques en sortie permet de caractériser entièrement le convertisseur sous test.

Tous ces produits BISTMaxx sont contrôlés par le même registre de contrôle et une interface à cinq connexions, qui permet de communiquer facilement avec un TAP compatible IEEE 1149.1. Le registre de contrôle, comme toute la partie numérique, est fourni sous forme de VHDL ou Verilog synthétisable, avec le produit. Pour plus d'informations sur le fonctionnement de ce registre de contrôle, se référer au chapitre 6.

2.4.2 TimeBIST de Vector¹²

Vector¹² propose des solutions de test transitoire embarquées (www.vector12.com). En effet, les ATE actuels ne sont pas adaptés au test de circuits intégrés fonctionnant à haute vitesse. Les tests transitoires haute vitesse sont responsables de la plus grande part du prix des ATE et du coût total de test. La tendance actuelle est donc de s'orienter vers des testeurs à bas coût et d'utiliser des techniques de DfT pour diminuer les coûts et ainsi accélérer le temps de mise sur le marché.

Partant de ces constatations, Vector¹² propose TimeBIST, une solution de test transitoire embarquée, qui permet des mesures de paramètres temporels, dont la gigue. Les avantages présentés par ce produit sont les suivants :

- Réduction de la dépendance vis-à-vis des testeurs traditionnels, en embarquant les composants à haute vitesse du test sur la puce, permettant aux fabricants d'utiliser leurs propres équipements de test.
- Réduction des coûts de test par l'utilisation de testeurs moins onéreux et suppression des temps de transmission entre la puce et les testeurs.
- Amélioration de la qualité du test (donc des rendements), par élimination des parasites dus à l'environnement de test.
- Amélioration de l'observabilité des noeuds internes grâce à l'utilisation de DfT.

Plusieurs versions du produit existent selon le type de composant à tester (tableau 2.3). Le prix de la licence dépend de la vitesse du composant à tester, des fonctions requises, et du nombre de mesures à réaliser sur une puce. Le produit est fourni sous forme d'IP et peut être accompagné de différents éléments tels que le guide de référence, un environnement de simulation, des scripts de synthèse ou de placement-routage. Un support est assuré pour la conception, la génération de vecteurs de test et les programmes de test.

La figure 2.22 montre l'architecture du produit timeBIST : il est constitué d'une partie circuit sur la puce et d'une partie logicielle hors puce. La partie embarquée mesure, échantillonne et code de manière numérique les paramètres temporels mesurés qui sont extraits de la puce par un protocole compatible avec le standard IEEE 1149.1. Elle est elle-même composée de deux parties : des sondes satellites pour capturer les signaux et une unité centrale pour les traiter.

- Les sondes satellites sont conçues pour diminuer la charge du circuit et offrir une résolution dix fois supérieure à celles des techniques traditionnelles. Ces sondes communiquent avec l'unité centrale par un bus de 8 bits.
- L'unité centrale est fournie sous forme de code RTL. Elle permet de piloter le test et de traiter les signaux issus des sondes et d'extraire les résultats selon un protocole compatible IEEE 1149.1.

TimeBIST n'influe pas sur le fonctionnement du circuit pendant le test. Il peut supporter jusqu'à 128 sondes avec une seule unité centrale. Le temps de test varie entre huit et cent millisecondes. Le coût équivalent en surface du circuit embarqué est compris entre quatre et dix mille portes selon les fonctions implémentées. La surface équivalente des sondes satellites est entre six cents et mille portes logiques.

Type de produit	Spécifications testées			
Test transitoire générique	Gigue RMS	Période de la gigue	Gigue cycle à cycle	Gigue pic à pic
	Fréquence	Inclinaison	Gigue partiellement dépendante du vecteur de test	
Test haute vitesse	Temps d'établissement et de maintien	Temps de montée et de descente	Gigue relative	Gigue à long terme
Test de composants de communication et réseau	Gigue totalement dépendante du vecteur de test		Bert embarqué	
Test de PLL/DLL	Temps de verrouillage	Plage de verrouillage	Réponse indicielle	
	Tolérance de gigue	Transfert de gigue	Bande passante de la boucle de gain	

TAB. 2.3 – Les différentes versions de produits de Vector¹²

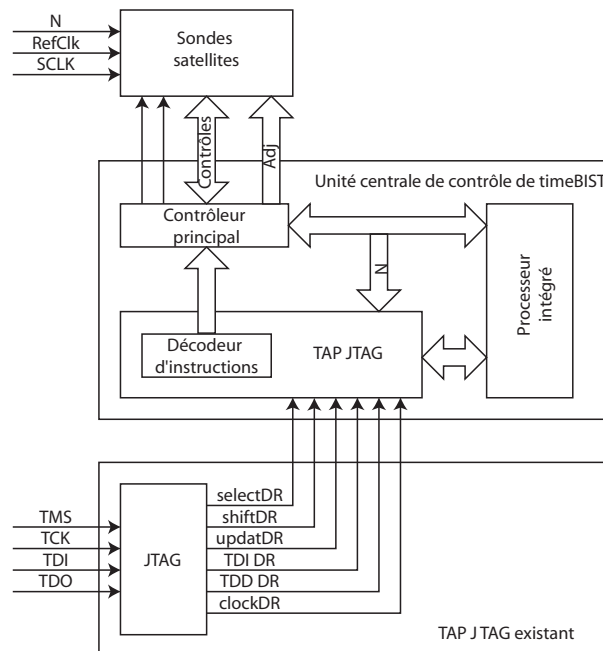


FIG. 2.22 – Architecture du produit timeBIST

2.4.3 Les outils de LogicVision

LogicVision (www.logicvision.com) propose plusieurs outils de test de circuits intégrés et notamment de test de circuits analogiques spécifiques :

Outil de test de SerDes

Cet outil, appelé EST (Embedded SerDes Test), est utilisé pour tester les composants SerDes, que l'on trouve dans beaucoup d'interfaces, telles que PCI express, Gbit Ethernet, Serial ATA, RapidIO, Fiber Channel ou encore Infiniband. Le test de ces composants est donc devenu un défi important dans le test de circuits intégrés. Il a été traité notamment dans [102]. Traditionnellement, ce type de composants est testé par la méthode BERT, qui utilise des équipements externes et qui est donc chère et complexe. La méthode EST proposée par LogicVision permet d'évaluer les paramètres temporels présentés dans la figure 2.23 (diagramme de l'oeil). Les bénéfices de cette technique sont :

- l'utilisation d'une technologie indépendante,
- l'absence d'impact sur le fonctionnement du SerDes,
- la possibilité de tester à la vitesse nominale,
- une précision des mesures inférieure à la picoseconde
- la possibilité de réutiliser l'IP pour plusieurs circuits,
- des temps de test très courts
- et une configuration ATE minimale.

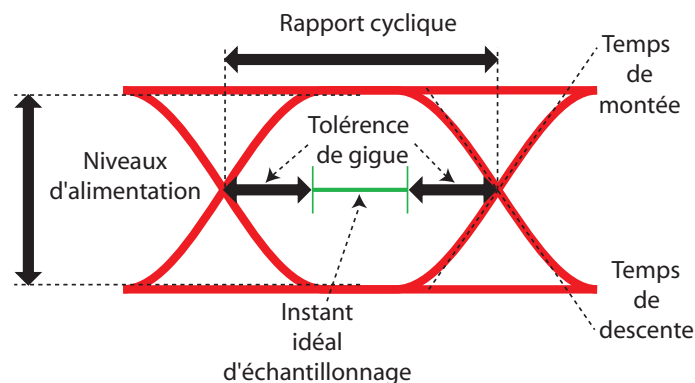


FIG. 2.23 – Paramètres évalués lors du test par EST

Outil de test de PLL

Le test de PLL a notamment été traité dans [101]. L'outil proposé par LogicVision, appelé EPT (Embedded PLL Test), est utilisé pour tester les PLL de manière numérique. Ce produit n'interfère pas avec le fonctionnement de la PLL proprement dite : il nécessite seulement d'avoir accès à l'entrée et à la sortie de celle-ci. La figure 2.24 montre l'ensemble de la circuiterie, avec la PLL à tester et le module de test EPT. L'entrée de la PLL est utilisée comme référence par EPT qui utilise l'horloge de sortie et éventuellement un indicateur de verrouillage pour calculer la gigue ou les limites supérieure et inférieure en fréquence de la plage de verrouillage ainsi que sa durée. Les valeurs de ces paramètres sont ensuite comparées numériquement à des limites choisies

pour classifier la PLL comme fonctionnelle ou défaillante. Les données d'entrée et de sortie sont respectivement chargées et extraites en série et l'architecture est compatible avec le standard de test IEEE 1149.1.

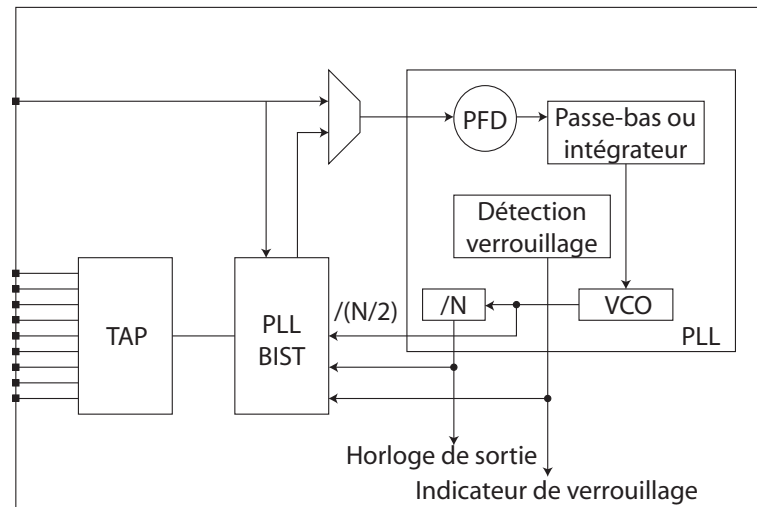


FIG. 2.24 – Circuiterie de test avec le module EPT

2.5 Conclusion

Comme nous l'avons vu tout au long de ce chapitre, le BIST analogique est en pleine évolution et n'a pas encore atteint son stade de maturité. Les techniques véritablement opérationnelles sont encore rares, surtout au niveau industriel et la plupart des produits utilisables nécessitent une part d'équipements extérieurs à la puce ou ont un usage assez spécifique.

Plusieurs points de cet état de l'art sont à retenir :

- Certaines techniques ont une vocation générale, mais beaucoup sont spécifiques. Ceci est particulièrement vrai pour les solutions industrielles que nous avons présentées :
 - BISTMaxx de Flunce présente différentes solutions spécifiques adaptées à différents types de circuits,
 - TimeBIST de Vector¹² ne permet de faire que du test transitoire,
 - les outils de LogicVision ne concernent que des composants particuliers (SerDes et PLL).
- La nécessité d'avoir une technique ajoutant le moins de surface possible élimine les solutions qui utilisent des traitements complexes à base de DSP par exemple. On essaye également d'éviter autant que possible d'avoir recours à des CNA ou des CAN.
- L'utilisation du modulateur $\Sigma\Delta$ se généralise de plus en plus :
 - elle permet de simplifier beaucoup les techniques de génération par exemple, en offrant la possibilité de travailler sur un train binaire plutôt que sur un signal purement analogique,
 - elle permet de réaliser des convertisseurs analogique-numérique de très bonne qualité,
 - l'implémentation du modulateur est assez simple et peu coûteuse en surface.

Durant cette thèse, nous avons cherché à proposer une technique complète de BIST analogique et mixte intégrée à bas coût qui soit la plus générique possible dans le sens qu'elle ne soit pas limitée à une classe spécifique de circuits mixtes. Nous avons décidé de réaliser un test fréquentiel, qui

permet de tester les circuits linéaires. Nous verrons que sous certaines conditions, cette technique permet de faire une analyse harmonique de la réponse du circuit sous test, même si celui-ci n'est pas linéaire. Nous avons travaillé sur les deux aspects du BIST : génération de signaux de test et analyse de la réponse. Nous avons utilisé des techniques basées sur l'utilisation d'un modulateur $\Sigma\Delta$ pour ces deux aspects : pour la génération, nous avons opté pour une technique utilisant un train binaire (généré par le modulateur) stocké en mémoire, avec une structure programmable. La programmabilité n'est pas fréquente dans les techniques existantes. Elle permet ici d'optimiser la signature, de réduire la surface de circuiterie et le temps de test. En ce qui concerne l'analyse de la réponse, nous avons utilisé une technique d'analyse harmonique de celle-ci, basée sur la création d'un train binaire à partir du signal analogique. Nous allons voir par la suite que ces deux techniques se complètent parfaitement pour former une architecture simple et complète de test fréquentiel complètement intégrée.

Chapitre 3

Génération des signaux de test analogiques, basée sur la modulation $\Sigma\Delta$

3.1 Introduction

Nous avons vu dans l'état de l'art qu'il existait plusieurs types de tests analogiques, le test AC (ou fréquentiel) étant l'un des plus répandus. Nous nous sommes donc intéressés dans cette thèse à cette approche, pour présenter un BIST analogique permettant de tester un dispositif linéaire, ou, de manière plus générale, d'évaluer les harmoniques de la réponse d'un dispositif, pour vérifier sa linéarité par exemple. Nous avons donc besoin d'une technique permettant de générer sur puce des signaux sinusoïdaux avec une résolution suffisante. Les travaux concernant la génération de signaux de test analogiques ont été démarrés avant le début de la thèse au cours des stages de fin d'études. On se référera donc à [97], [21] et [92] pour plus de détails. La méthode retenue est basée sur l'utilisation d'un train binaire stocké en mémoire, encodant le signal analogique à générer ([37], [22] et [23]). Ce train binaire est ensuite filtré sur la puce pour obtenir le signal analogique. L'ajout de la possibilité de programmer l'architecture permet de diminuer notablement le surcoût en surface de la circuiterie de test ([65], [70] et [96]). Cette technique permet de générer des signaux mono ou multifréquences. Par simplicité, nous ne nous intéresserons ici qu'aux signaux monofréquences. La génération et l'optimisation de signaux multifréquences a été traitée dans [10].

3.2 Théorie

3.2.1 Principe général

Le principe général de la technique présentée ici a été proposé dans [24]. La génération du signal analogique est composée de deux étapes :

- une étape logicielle qui va générer le train binaire encodant le signal à générer par modulation $\Sigma\Delta$,
- une étape sur puce qui va créer le signal analogique en question par filtrage à partir du train binaire stocké en mémoire.

Dans cette technique, un train binaire de longueur 1024 bits est utilisé. Nous avons voulu diminuer le coût en surface de la technique. Pour cela, nous avons ajouté la possibilité de programmer la longueur du train binaire et la fréquence d'échantillonnage, ce qui nous permet de choisir la fréquence du signal généré avec une grande précision, avec une longueur du train binaire plus faible

(de 100 à 200 bits) ([96]). La contrepartie est une faible résolution dans le codage de l'amplitude du signal, car le nombre de bits est plus petit. Il faut tenir compte de cette limitation dans l'hypothèse d'une intégration de cette technique dans un BIST. La figure 3.1 présente une vue d'ensemble de la chaîne de génération du signal analogique.

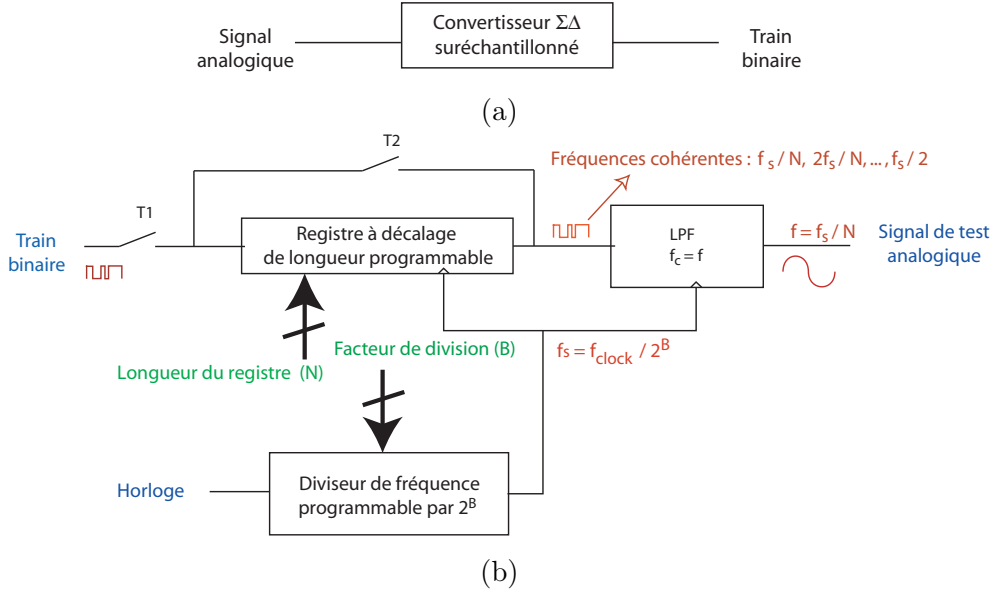


FIG. 3.1 – Vue globale de la technique de génération de signaux de test analogiques: (a) partie logicielle et (b) partie sur puce

3.2.2 Génération du train binaire

Le train binaire est obtenu de manière logicielle: le signal analogique que l'on veut générer est envoyé en entrée d'un modulateur $\Sigma\Delta$ qui va générer le train binaire encodant ce signal. Un logiciel de CAO permet d'effectuer cette opération, en ajoutant en plus la possibilité d'optimiser le train binaire comme nous allons le voir par la suite.

Principe du modulateur $\Sigma\Delta$

Le modulateur $\Sigma\Delta$ est un dispositif suréchantillonné, permettant de réaliser une conversion analogique-numérique avec un seul bit de résolution: il consiste à coder les variations du signal d'entrée par un train binaire. Il est souvent utilisé avec un filtre numérique pour former un convertisseur $\Sigma\Delta$. Avant d'étudier le fonctionnement du modulateur, rappelons les deux types d'intégrateurs échantillonnés que l'on peut rencontrer. La figure 3.2 rappelle les deux possibilités. Le schéma (a) représente un intégrateur simple de fonction de transfert $H(z) = \frac{z}{z-1}$, le schéma (b) un intégrateur avec délai de fonction de transfert $H(z) = \frac{1}{z-1}$.

La figure 3.3 montre le schéma d'un modulateur $\Sigma\Delta$ d'ordre 1. C'est un dispositif avec boucle de retour, qui intègre la différence entre le signal d'entrée et le train binaire de sortie. Comme on n'a qu'un bit de résolution en sortie, le train binaire ne va bien sûr pas coder directement la valeur du signal analogique. C'est la fréquence instantanée de "1" et de "0" du train binaire qui va coder la valeur du signal à cet instant. On comprend ainsi la nécessité du suréchantillonnage dans ce type

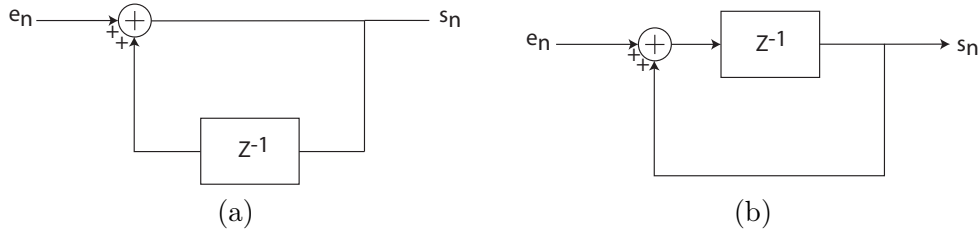


FIG. 3.2 – Les deux types d'intégrateurs utilisés pour un modulateur $\Sigma\Delta$: (a) un intégrateur simple et (b) un intégrateur avec délai

de dispositifs. C'est d'ailleurs le taux de suréchantillonnage qui va donner la précision du codage, le bruit de quantification diminuant lorsque ce taux augmente.

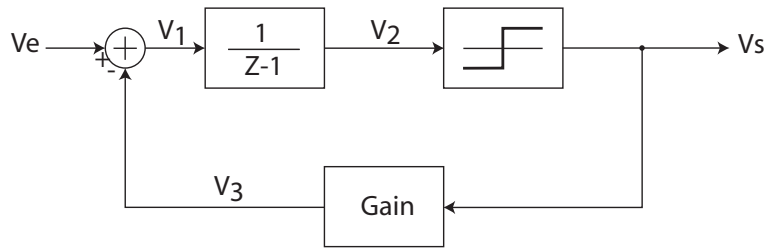


FIG. 3.3 – Schéma d'un modulateur $\Sigma\Delta$ d'ordre 1

Comme tout convertisseur analogique numérique, le modulateur $\Sigma\Delta$ introduit un bruit de quantification. Ce bruit est introduit par le comparateur. On peut donc modéliser le modulateur comme indiqué sur la figure 3.4.

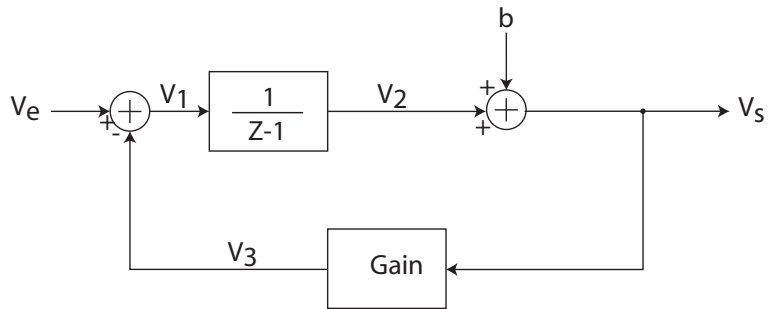


FIG. 3.4 – Modélisation du bruit de quantification dans un modulateur $\Sigma\Delta$ d'ordre 1

Le bruit de quantification est modélisé par la variable b . On va pouvoir, grâce à cette représentation, écrire :

$$V_s(z) = FTS(z)V_e(z) + FTB(z)b(z) \quad (3.1)$$

avec FTS la fonction de transfert du signal, obtenue en considérant un bruit b nul et FTB la fonction de transfert du bruit, obtenue en considérant une tension d'entrée V_e nulle. Le calcul donne les fonctions de transfert des équations 3.2.

$$\begin{aligned} FTS &= z^{-1} \\ FTB &= 1 - z^{-1} \end{aligned} \quad (3.2)$$

En remplaçant z par $e^{j2\pi n f T_s}$ où T_s est la période d'échantillonnage, f la fréquence, on trouve :

$$FTB(f) = 2 \sin\left(\frac{2\pi f T_s}{2}\right) e^{j\frac{\pi - n 2\pi f T_s}{2}}$$

$$|FTB(f)| = |2 \sin(\pi f T_s)| \tag{3.3}$$

$$\arg(FTB) = \frac{\pi}{2} - \pi f n T_s \tag{3.4}$$

Si l'on trace maintenant le module de la fonction de transfert de bruit en fonction de la fréquence, on obtient la figure 3.5. Le signal étant réel, on peut tracer cette fonction de transfert sur l'intervalle $[0, f_s]$ sachant qu'elle sera symétrique par rapport à $\frac{f_s}{2}$. On note que le bruit de quantification est rejeté au centre du spectre, vers les fréquences élevées. Lorsque le modulateur est utilisé dans un convertisseur $\Sigma\Delta$, il est suivi d'un filtre numérique et d'un filtre anti-recouvrement. Le rejet du bruit de quantification en dehors de la bande passante du filtre va alors permettre d'avoir un filtrage plus efficace. Dans notre cas, le train binaire obtenu en sortie du modulateur va être filtré par un filtre passe-bas pour obtenir un signal sinusoïdal, donc le rejet du bruit va être intéressant également car la qualité du filtrage va s'en trouver améliorée.

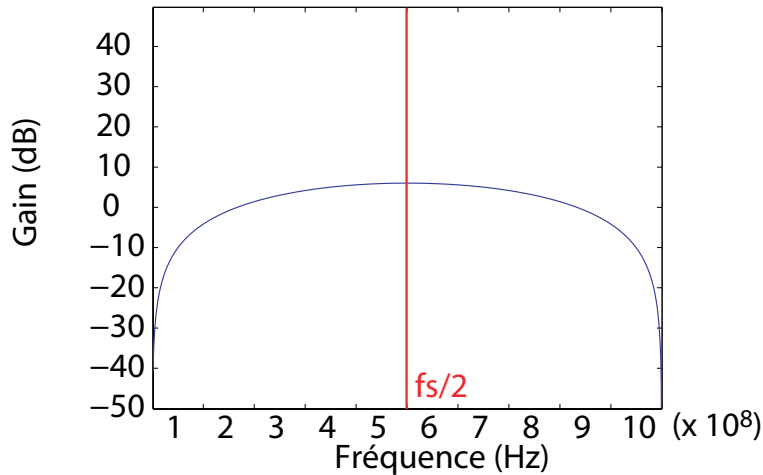


FIG. 3.5 – Spectre du bruit de quantification ajouté par un modulateur $\Sigma\Delta$, pour une fréquence d'échantillonnage f_s de 1GHz

Le principe du modulateur $\Sigma\Delta$ d'ordre 2 est le même, mais avec deux intégrateurs (figure 3.6) : un intégrateur simple et un intégrateur à retard. D'une manière générale, lorsque l'on augmente l'ordre du modulateur, le bruit de quantification est rejeté de plus en plus loin vers les fréquences élevées, ce qui rend un filtrage ultérieur encore plus efficace. En revanche, des problèmes de stabilité peuvent apparaître.

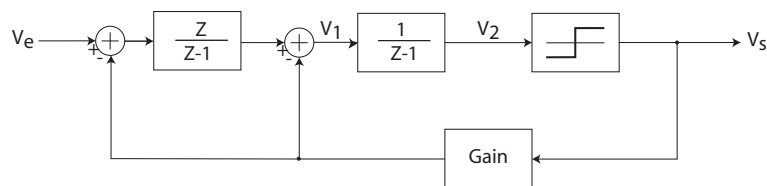


FIG. 3.6 – Schéma d'un modulateur $\Sigma\Delta$ d'ordre 2

Logiciel de CAO et optimisation

Un logiciel a été conçu dans le groupe pour la génération du train binaire ([10] et [9]). Ce logiciel, écrit en langage SKILL, est entièrement intégré à l'environnement de conception Cadence. Il prend en entrée les caractéristiques du signal analogique à générer, et fournit en sortie le train binaire sous forme d'un fichier compatible avec un testeur numérique¹. Ce logiciel permet non-seulement de simuler le comportement du $\Sigma\Delta$ pour générer le train binaire, mais aussi d'optimiser ce train binaire. Afin d'expliquer cette opération d'optimisation, commençons par définir les paramètres que nous allons utiliser pour mesurer la qualité de celui-ci :

$$\begin{aligned} SFDR &= 20 \log \left(\frac{A_1}{\max(A_2, A_3, \dots, A_k)} \right) \\ THD &= \frac{\sqrt{\sum_{i=2}^n A_i^2}}{A_1} \end{aligned} \tag{3.5}$$

où A_i représente l'amplitude de la $i^{\text{ième}}$ composante du signal

Dans la pratique, on peut montrer que, pour un train binaire, ces paramètres dépendent fortement de la phase de chaque composante fréquentielle du signal analogique envoyé au modulateur $\Sigma\Delta$. La figure 3.7 montre le THD du train binaire généré en fonction de la phase du signal analogique envoyé au modulateur. On peut constater les énormes différences de qualité de signal qui apparaissent à cause de petites variations de phase. L'optimisation va donc se faire sur le choix de cette phase afin d'optimiser les différents paramètres définis ci-dessus. On parlera d'optimisation multi-objectifs. Cette optimisation est réalisée grâce à un algorithme WARGA (Weighted Average Ranking Genetic Algorithm). Cette méthode est décrite dans [17] et [27].

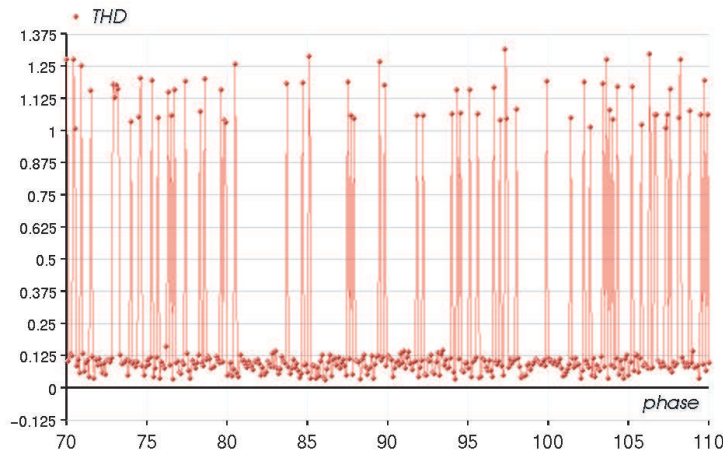


FIG. 3.7 – Valeur du THD en fonction de la phase du signal analogique envoyé au modulateur $\Sigma\Delta$

Si l'optimisation est efficace, les harmoniques du train binaire les plus proches du fondamental seront faibles. Il reste cependant le bruit de quantification du $\Sigma\Delta$, mais celui-ci est repoussé vers le

1. Testeur numérique IMS du CIME, utilisé pour le test du démonstrateur

centre du spectre comme expliqué précédemment. Dans ces conditions, en utilisant un filtre passe-bas dont la fréquence de coupure est proche du fondamental, on obtiendra un filtrage très efficace du train binaire et donc un signal analogique de bonne qualité (figure 3.8). On voit sur cette figure la fonction de transfert du filtre sur la gauche et le spectre du train binaire sur la droite, avec le bruit de quantification rejeté au centre du spectre. On constate qu’avec une fréquence de coupure proche du fondamental, le filtrage est très efficace : en effet, les premières valeurs significatives du spectre sont déjà très atténuées grâce à l’optimisation du train binaire et correspondent à une fréquence où l’atténuation du filtre est importante. Ceci explique la bonne qualité du signal dans cette technique.

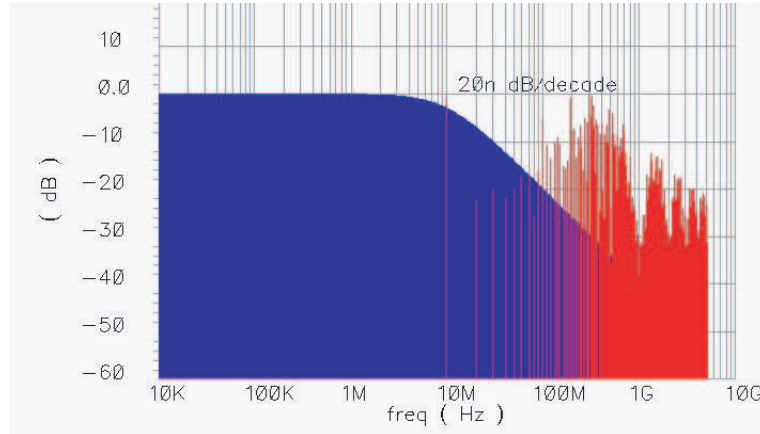


FIG. 3.8 – Filtrage d’un train binaire obtenu par modulation $\Sigma\Delta$, par un filtre de fréquence de coupure proche du fondamental

La figure 3.9 montre des captures d’écran du logiciel : (a) représente la fenêtre dans laquelle l’utilisateur peut rentrer les paramètres du signal à générer, et (b) représente le fichier généré en sortie. Ce fichier contient le train binaire encodant le signal analogique, ainsi que les bits nécessaires à la programmation de l’architecture. Ce fichier peut être directement envoyé au testeur numérique pour effectuer le test.

3.2.3 Obtention du signal analogique

Répétition du train binaire

Le train binaire contient une période du signal analogique à générer. Ce train binaire est ensuite répété périodiquement, de manière à obtenir un train binaire périodique. On montre que ce train binaire contient les fréquences :

$$f = f_s \frac{M}{N} \quad (3.6)$$

avec f_s la fréquence d’échantillage du modulateur $\Sigma\Delta$, N la longueur du train binaire et $M = 1, 2, 3, \dots, \frac{N}{2}$. Ce train binaire va ensuite pouvoir être filtré pour ne garder qu’une partie de ces fréquences. Par exemple, si l’on cherche à générer un signal monofréquence, on va garder le fondamental (correspondant à $M=1$). La fréquence de la sinusoïde générée est alors $f = \frac{f_s}{N}$. Nous voyons bien ici qu’il est possible de choisir la fréquence obtenue en programmant la longueur N du train binaire et la fréquence f_s d’échantillonnage. Nous avons choisi pour l’implémentation de la technique de faire varier la longueur du registre (donc du train binaire) entre 100 et 200 bits. Nous avons aussi choisi de programmer la fréquence d’échantillonnage à partir d’une fréquence d’horloge de 1 GHz

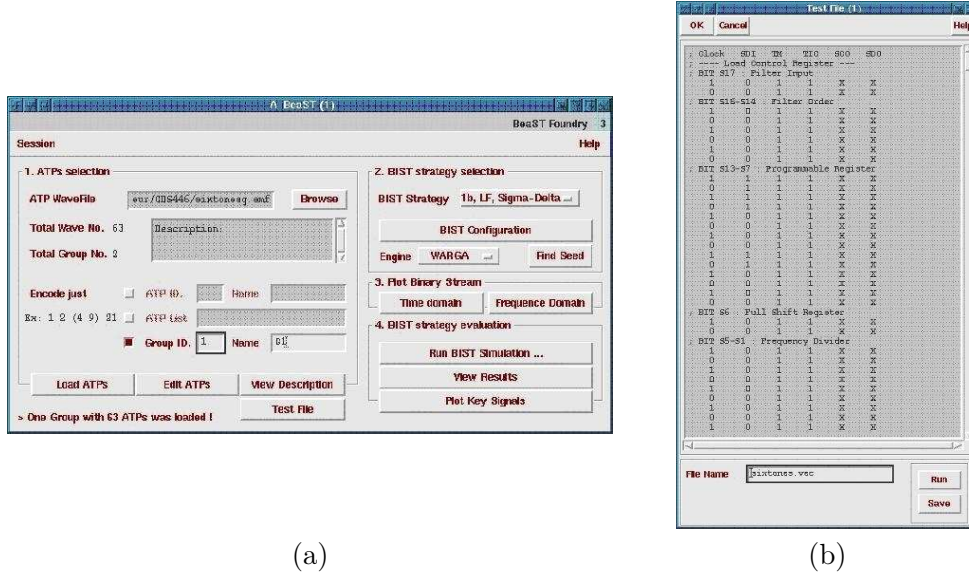


FIG. 3.9 – Captures d'écran du logiciel de CAO utilisé pour générer les signaux de test analogiques

par une division de fréquence par une puissance de deux ($f_s = \frac{f_c}{2^B}$, avec f_c la fréquence d'horloge générale de la puce et 2^B le facteur de division, B variant entre 1 et 20). Ainsi les fréquences que l'on peut générer sont données dans le tableau 3.1. En fait, chaque valeur de B couvre une octave de fréquence et une résolution de 100 fréquences par octave est possible en changeant la longueur du registre N . Par exemple, pour une horloge à 1 GHz, on constate que la résolution du signal analogique sera meilleure que 1%, avec 1600 fréquences possibles dans une gamme de 10 Hz à 10 MHz.

Il faudra ensuite filtrer ce train binaire par un filtre passe-bas dont la fréquence de coupure soit proche de $\frac{f_s}{N}$ pour récupérer un signal proche du signal sinusoïdal de départ, encodé par le train binaire.

Filtrage analogique à capacités commutées

Le filtre utilisé pour retrouver le signal analogique encodé à partir du train binaire doit être programmable, afin que sa fréquence de coupure corresponde à la fréquence du signal analogique. Cette considération nous a amené à faire le choix d'une structure à capacités commutées pour le filtre.

Rappelons tout d'abord le principe des capacités commutées. La figure 3.10 montre le schéma de principe d'une capacité commutée. Les deux interrupteurs T_1 et T_2 sont commandés par les horloges représentées sur la figure 3.11. Pendant la phase 1 (T_1 fermé et T_2 ouvert), la charge de la capacité C vaut : $Q_1 = CV_1$. Pendant la phase 2, la charge de la capacité vaut $Q_2 = CV_2$. Pendant une période d'horloge T , la quantité de charge dQ écoulée de V_1 vers V_2 est $Q_1 - Q_2 = C(V_1 - V_2)$, donc le courant moyen entre V_1 et V_2 est :

$$I_{moy} = \frac{dQ}{T} = \frac{C(V_1 - V_2)}{T}$$

La capacité commutée est donc équivalente à une résistance de valeur $R_{eq} = \frac{T}{C}$. Notons qu'il est extrêmement important que les horloges commandant les deux interrupteurs ne se recouvrent pas,

B	2^B	f_s	N					
			100	101	...	199	200	
1	1	$1,00 \cdot 10^9$	$1,00 \cdot 10^7$	$9,90 \cdot 10^6$			$5,03 \cdot 10^6$	$5,00 \cdot 10^6$
2	2	$5,00 \cdot 10^8$	$5,00 \cdot 10^6$	$4,95 \cdot 10^6$			$2,51 \cdot 10^6$	$2,50 \cdot 10^6$
3	4	$2,50 \cdot 10^8$	$2,50 \cdot 10^6$	$2,48 \cdot 10^6$			$1,26 \cdot 10^6$	$1,25 \cdot 10^6$
4	8	$1,25 \cdot 10^8$	$1,25 \cdot 10^6$	$1,24 \cdot 10^6$			$6,28 \cdot 10^5$	$6,25 \cdot 10^5$
5	16	$6,25 \cdot 10^7$	$6,25 \cdot 10^5$	$6,19 \cdot 10^5$			$3,14 \cdot 10^5$	$3,13 \cdot 10^5$
6	32	$3,13 \cdot 10^7$	$3,13 \cdot 10^5$	$3,09 \cdot 10^5$			$1,57 \cdot 10^5$	$1,56 \cdot 10^5$
7	64	$1,56 \cdot 10^7$	$1,56 \cdot 10^5$	$1,55 \cdot 10^5$			$7,85 \cdot 10^4$	$7,81 \cdot 10^4$
8	128	$7,81 \cdot 10^6$	$7,81 \cdot 10^4$	$7,74 \cdot 10^4$			$3,93 \cdot 10^4$	$3,91 \cdot 10^4$
9	256	$3,91 \cdot 10^6$	$3,91 \cdot 10^4$	$3,87 \cdot 10^4$			$1,96 \cdot 10^4$	$1,95 \cdot 10^4$
10	512	$1,95 \cdot 10^6$	$1,95 \cdot 10^4$	$1,93 \cdot 10^4$			$9,81 \cdot 10^3$	$9,77 \cdot 10^3$
11	1024	$9,77 \cdot 10^5$	$9,77 \cdot 10^3$	$9,67 \cdot 10^3$			$4,91 \cdot 10^3$	$4,88 \cdot 10^3$
12	2048	$4,88 \cdot 10^5$	$4,88 \cdot 10^3$	$4,83 \cdot 10^3$			$2,45 \cdot 10^3$	$2,44 \cdot 10^3$
13	4096	$2,44 \cdot 10^5$	$2,44 \cdot 10^3$	$2,42 \cdot 10^3$			$1,23 \cdot 10^3$	$1,22 \cdot 10^3$
14	8192	$1,22 \cdot 10^5$	$1,22 \cdot 10^3$	$1,21 \cdot 10^3$			$6,13 \cdot 10^2$	$6,10 \cdot 10^2$
15	16384	$6,10 \cdot 10^3$	$6,10 \cdot 10^2$	$6,04 \cdot 10^2$			$3,07 \cdot 10^2$	$3,05 \cdot 10^2$
16	32768	$3,05 \cdot 10^4$	$3,05 \cdot 10^2$	$3,05 \cdot 10^2$			$1,53 \cdot 10^2$	$1,53 \cdot 10^2$
17	65536	$1,53 \cdot 10^4$	$1,53 \cdot 10^2$	$1,51 \cdot 10^2$			$7,67 \cdot 10^1$	$7,63 \cdot 10^1$
18	131072	$7,63 \cdot 10^3$	$7,63 \cdot 10^1$	$7,55 \cdot 10^1$			$3,83 \cdot 10^1$	$3,81 \cdot 10^1$
19	262144	$3,81 \cdot 10^3$	$3,81 \cdot 10^1$	$3,78 \cdot 10^1$			$1,92 \cdot 10^1$	$1,91 \cdot 10^1$
20	524288	$1,91 \cdot 10^3$	$1,91 \cdot 10^1$	$1,89 \cdot 10^1$			$9,58 \cdot 10^0$	$9,54 \cdot 10^0$

TAB. 3.1 – Exemple de fréquences générables en divisant la fréquence d'horloge par un puissance de deux et en choisissant la taille du registre

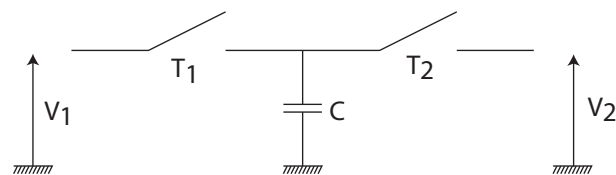


FIG. 3.10 – Schéma électrique d'une capacité commutée

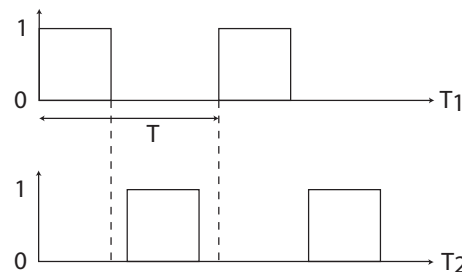


FIG. 3.11 – Horloges non recouvrantes pour capacité commutées

c'est-à-dire qu'elles ne doivent jamais être à "1" en même temps. Sinon, les deux interrupteurs sont fermés, et il y a un phénomène d'injection de charges qui nuit aux performances.

L'avantage des capacités commutées est qu'il est possible de réaliser des résistances de grande valeur. Le rapport de capacités est également beaucoup plus précis que le rapport de résistances. Contrairement à une résistance, il n'y a pas de pertes par effet Joule. De plus, nous voyons dans notre cas que l'on va pouvoir régler la valeur de la résistance grâce à la valeur de la fréquence des horloges de contrôle. Ainsi, si le filtre est réalisé entièrement à base de capacités commutées en remplacement des résistances, on va pouvoir programmer sa fréquence de coupure grâce à la fréquence de l'horloge, de la même façon que, pour une longueur de registre donnée, la fréquence du signal analogique dépend de celle de l'horloge.

L'idéal serait d'avoir une fréquence de coupure exactement égale à la fréquence du fondamental du signal à filtrer. Mais ceci est très difficile à réaliser dans la pratique. Dans notre cas, nous avons choisi, pour plus de facilité, d'avoir une fréquence de coupure égale à $\frac{f_s}{100}$. Il ne s'agit donc pas exactement de la fréquence du fondamental et le filtrage sera dans l'absolu plus efficace quand la longueur du registre sera proche de 100, comme l'illustre la figure 3.8. Cependant, le logiciel procure un train binaire optimisé, c'est-à-dire que les harmoniques suivant le fondamental doivent être assez faibles. De plus, le bruit de quantification est rejeté vers le centre du spectre, et ainsi le filtrage doit rester efficace pour toutes les longueurs du registre. En revanche, la qualité du signal s'améliore quand le taux de suréchantillonnage augmente comme nous l'avons expliqué dans la section traitant du fonctionnement d'un modulateur $\Sigma\Delta$. Les résultats de simulation vont permettre de comparer l'effet de ces deux facteurs combinés et d'évaluer la qualité du signal dans toutes les conditions.

3.3 Principe du filtrage analogique

Afin d'effectuer le filtrage analogique, nous avons opté pour un filtre de type Butterworth ([21]). Les filtres de Butterworth ont la particularité d'avoir une fonction de transfert très plate autour de l'origine des fréquences. La forme générale du module de la fonction de transfert d'un filtre de Butterworth d'ordre n est donnée dans l'équation 3.7.

$$|H(j\omega)| = \frac{1}{\sqrt{1 + \omega^{2n}}} \quad (3.7)$$

Voyons maintenant comment obtenir un tel filtre. Pour cela, considérons la fonction de transfert générale d'un filtre passe-bas d'ordre n :

$$H(p) = \frac{1}{p^n + b_{n-1}p^{n-1} + \dots + b_0} \quad (3.8)$$

En remplaçant p par $j\omega$ et en calculant le module de la fonction obtenue, on peut par identification calculer les coefficients pour obtenir la forme de l'équation 3.7. Le tableau 3.2 donne les coefficients que l'on obtient pour les filtres d'ordre 1 à 6.

Dans la pratique, nous voulons pouvoir essayer des filtres d'ordre 2 à 6. Pour réaliser ces filtres de Butterworth, nous allons utiliser un enchaînement de filtres d'ordre 1 et 2. Les filtres d'ordre 1 seront de simples intégrateurs, et les filtres d'ordre 2 des filtres de Rauch.

n	b_0	b_1	b_2	b_3	b_4	b_5
1	1	-	-	-	-	-
2	1	1,41421	-	-	-	-
3	1	2	2	-	-	-
4	1	2,61313	3,41421	2,61313	-	-
5	1	3,23607	5,23607	5,23607	3,23607	-
6	1	3,86370	7,46410	9,14162	7,46410	3,86370

TAB. 3.2 – Coefficients d'un filtre de Butterworth pour les ordres 1 à 6

Intégrateur

Le schéma d'un intégrateur est donné sur la figure 3.12. Sa fonction de transfert est donnée dans l'équation 3.9.

$$\frac{V_s}{V_e} = \frac{1}{RCp + 1} \quad (3.9)$$

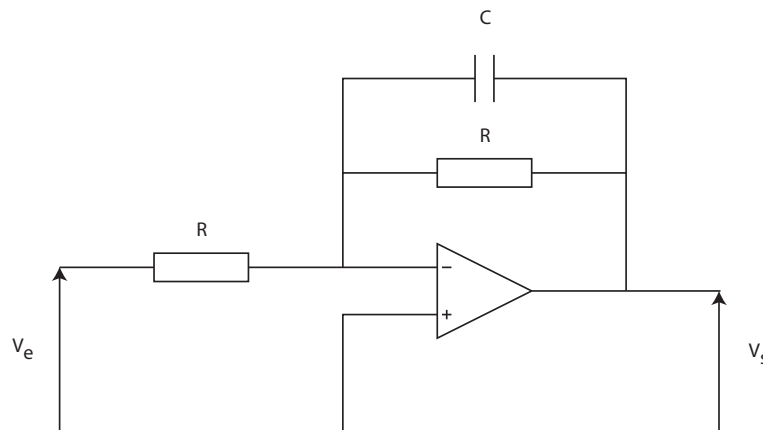


FIG. 3.12 – Schéma électrique d'un intégrateur

Filtres de Rauch

Le schéma d'un filtre de Rauch est donné sur la figure 3.13. Sa fonction de transfert est :

$$\frac{V_s}{V_e} = \frac{-1}{R^2 C_1 C_2 p^2 + 3RC_2 p + 1} \quad (3.10)$$

Comme nous l'avons annoncé précédemment, les résistances R vont être remplacées par des capacités commutées, selon la figure 3.14.

Dans ce cas, on a $R = \frac{1}{C_0} f_s$ où f_s est la fréquence d'échantillonnage. Nous choisissons la valeur de la capacité C_0 à 500 fF. Nous allons alors pouvoir calculer les valeurs des capacités C_1 et C_2 afin d'avoir une fréquence de coupure égale au centième de la fréquence d'échantillonnage. Prenons l'exemple du filtre d'ordre 2 (constitué uniquement du filtre de Rauch) :

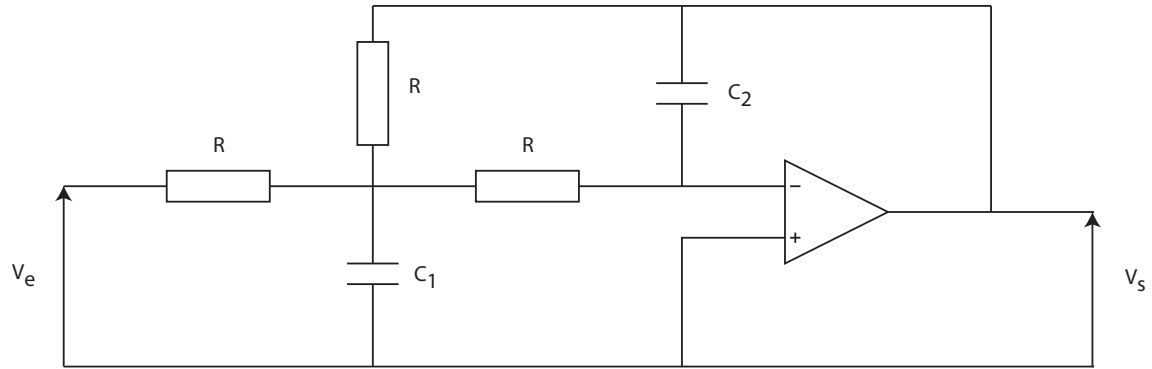


FIG. 3.13 – Schéma électrique d'une structure de filtre de Rauch

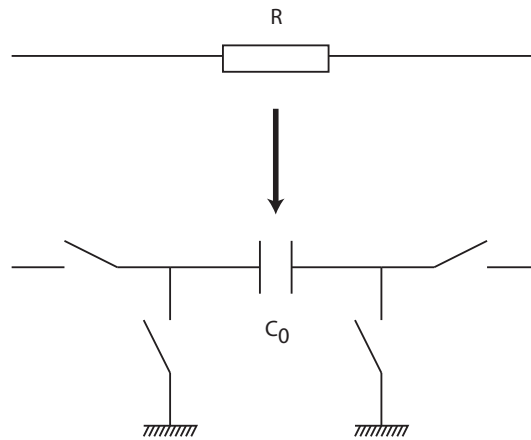


FIG. 3.14 – Structure de capacités commutées utilisée pour remplacer les résistances dans les filtres

Nous pouvons écrire sa fonction de transfert, avec les coefficients déterminés dans le tableau 3.2 :

$$H(s) = \frac{1}{s^2 + 1.41421s + 1} \quad (3.11)$$

En effectuant le changement de variable : $s = \frac{p}{\omega_c}$ on obtient l'équation 3.12.

$$H(p) = \frac{1}{\left(\frac{p}{\omega_c}\right)^2 + 1.41421\frac{p}{\omega_c} + 1} \quad (3.12)$$

En procédant par identification des coefficients à partir de l'équation 3.10, on trouve :

$$\frac{1,4142}{\omega_c} = 3RC_2 \quad (3.13)$$

$$R^2C_1C_2 = \frac{1}{\omega_c^2} \quad (3.14)$$

et finalement :

$$C_0 = 500fF$$

$$C_1 = 16,88pF$$

$$C_2 = 3,75pF$$

Considérons maintenant un filtre d'ordre 3: sa fonction de transfert peut être décomposée en deux éléments simples, c'est à dire le produit de fonctions de transfert de filtres d'ordre 1 et 2, comme le montre l'équation 3.15.

$$H(s) = \frac{1}{s^3 + 2s^2 + 2s + 1} = \left(\frac{1}{s + 1} \right) \left(\frac{1}{s^2 + s + 1} \right) = \left(\frac{1}{\frac{p}{\omega_c} + 1} \right) \left(\frac{1}{\left(\frac{p}{\omega_c} \right)^2 + \frac{p}{\omega_c} + 1} \right) \quad (3.15)$$

Nous voyons que ce filtre peut être réalisé à partir d'un filtre d'ordre 1 (simple intégrateur) et un filtre de Rauch d'ordre 2. Par identification des coefficients comme précédemment, on trouve les valeurs suivantes des capacités pour ces deux composants pour le filtre de Rauch (A) et l'intégrateur (B) :

$$\begin{aligned} C_{0A} &= 500fF & C_{0B} &= 500fF \\ C_{1A} &= 7,96pF & C_{1B} &= 23,87pF \\ C_{2A} &= 2,65pF & & \end{aligned} \quad (3.16)$$

Afin de déterminer le meilleur filtre pour notre application, nous avons décidé de tester des filtres jusqu'à l'ordre 6. Voici les valeurs des composants que l'on obtient :

- Pour le filtre d'ordre 4, deux filtres de Rauch d'ordre 2, avec les valeurs de capacités :

$$\begin{aligned} C_{0A} &= 500fF & C_{0B} &= 500fF \\ C_{1A} &= 31,19pF & C_{1B} &= 12,92pF \\ C_{2A} &= 2,03pF & C_{2B} &= 4,90pF \end{aligned}$$

- Pour le filtre d'ordre 5, deux filtres de Rauch d'ordre 2 et un intégrateur, avec les valeurs de capacités :

$$\begin{aligned} C_{0A} &= 500fF & C_{0B} &= 500fF & C_{0C} &= 500fF \\ C_{1A} &= 14,754pF & C_{1B} &= 38,63pF & C_{1C} &= 7,96pF \\ C_{2A} &= 4,29pF & C_{2B} &= 1.64pF & & \end{aligned}$$

- Pour le filtre d'ordre 6, trois filtres de Rauch d'ordre 2, avec les valeurs de capacités :

$$\begin{aligned} C_{0A} &= 500fF & C_{0B} &= 500fF & C_{0C} &= 500fF \\ C_{1A} &= 46,12pF & C_{1B} &= 16,88pF & C_{1C} &= 12,36pF \\ C_{2A} &= 1,37pF & C_{2B} &= 3,75pF & C_{2C} &= 5,12pF \end{aligned}$$

Les schémas d'intégrateur et de filtre de Rauch donnés ici correspondent à une structure non-différentielle. Les calculs sont les mêmes pour une structure différentielle. L'implémentation physique de ces filtres sera décrite dans le chapitre 5.

3.4 Résultats de simulation

3.4.1 Génération du train binaire

La figure 3.15 donne respectivement le SFDR et le THD des signaux générés en fonction de la longueur N du registre à décalage. On peut remarquer que la qualité des signaux augmente en moyenne avec la longueur du registre à décalage, pour une octave quelconque du tableau 3.1. Cependant, pour les longueurs de registre impaires, la qualité du signal est beaucoup moins bonne que pour les valeurs paires. Ceci est dû au fait que le second harmonique devrait être codé par un registre dont la longueur est la moitié de celle du fondamental. Lorsque la longueur du registre est impaire, le second harmonique ne peut être codé aussi correctement que si la longueur est paire. Nous voyons ainsi apparaître une distorsion qui est liée à un problème de cohérence des harmoniques. D'une manière plus générale, si N n'est pas divisible par i , c'est le $i^{\text{ième}}$ harmonique qui ne va pas être codé correctement (mais le plus sensible est le second harmonique). Ainsi, les meilleures valeurs de N sont celles qui possèdent un grand nombre de diviseurs, telles que 120 par exemple.

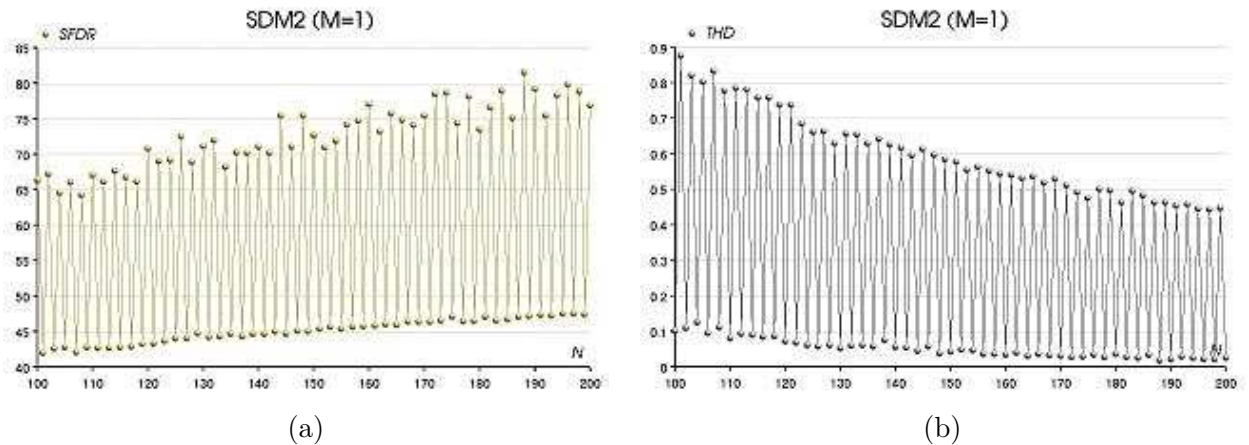


FIG. 3.15 – SFDR et THD de trains binaires encodés avec un modulateur $\Sigma\Delta$ du second ordre en fonction de la longueur du registre programmable

3.4.2 Obtention d'un signal analogique non-différentiel

Comme nous allons le voir par la suite, deux démonstrateurs ont été fabriqués pour valider cette technique. Dans le premier cas, nous avons conçu un banc de filtres programmables d'ordre 2 à 6 pour pouvoir comparer leurs performances. A ce stade, la structure de ces filtres n'est pas différentielle. Nous avons testé ces différents filtres et obtenu des résultats satisfaisants pour tous les filtres et pour une gamme de fréquence d'horloge située entre 1,91 kHz et 31,3 MHz (ce qui correspond à des fréquences de signaux situées entre 9,5 Hz et 313 kHz, cette gamme définissant la bande passante des filtres). Les résultats obtenus sont les mêmes pour toutes les fréquences. Nous avons utilisé à chaque fois un train binaire encodant un signal d'amplitude 360 mV. Pour les cinq filtres d'ordre 2 à 6, nous obtenons les SFDR minimums présentés dans le tableau 3.3 .

Nous avons noté au cours de ces différentes simulations que les performances d'un filtre d'ordre 3 sont satisfaisantes avec un SFDR supérieur à 50 dB dans la quasi-totalité des cas. Les filtres d'ordre supérieur sont plus difficiles et plus lourds à implémenter. Nous nous sommes donc limités à cet

Ordre du filtre	SFDR min (dB)
Ordre 2	40,52
Ordre 3	48,54
Ordre 4	53,32
Ordre 5	55,88
Ordre 6	58,09

TAB. 3.3 – SFDR minimums obtenus pour les différents ordres de filtres passe-bas

ordre de filtre pour la suite. Comme l'ordre du filtre doit au moins être supérieur d'une unité à celui du modulateur $\Sigma\Delta$, cela nous permet d'utiliser un modulateur d'ordre 2 qui reste également assez facile à implémenter. Des modulateurs d'ordre supérieur posent des problèmes de stabilité. Nous verrons dans le chapitre 5 que nous avons cependant fabriqué un démonstrateur contenant tous les ordres de filtres afin de vérifier leur fonctionnement sur silicium et confirmer ces résultats. La figure 3.16 montre les représentations temporelle et fréquentielle d'un signal de 360 mV à la fréquence 300,5 kHz obtenu avec un filtre d'ordre 3. La longueur du train binaire est de 104 et le modulateur $\Sigma\Delta$ utilisé est d'ordre 2. On voit sur cette figure que le signal est échantillonné à cause des capacités commutées. Le SFDR obtenu dans ce cas vaut 65,97 dB.

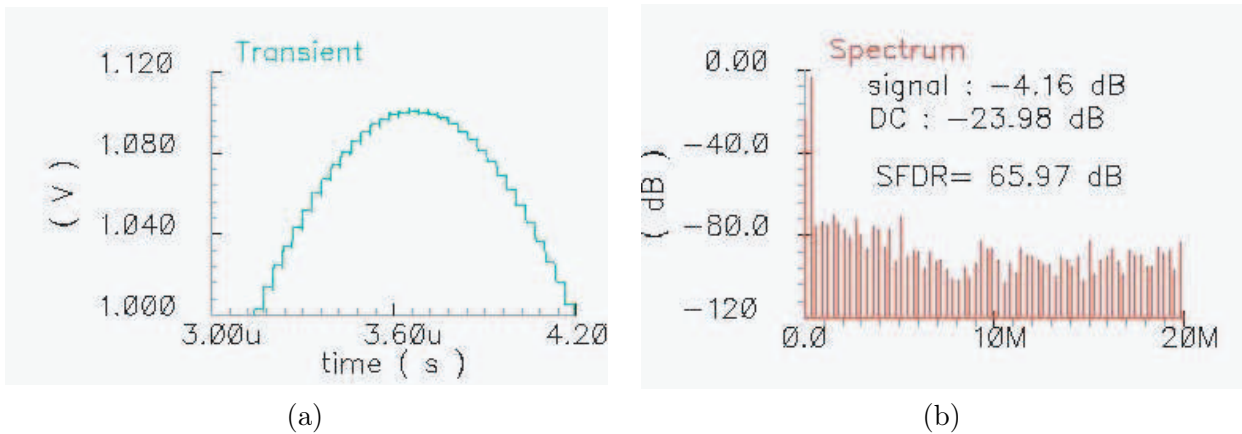


FIG. 3.16 – Représentations temporelle (a) et spectrale (b) d'un signal généré

3.4.3 Obtention d'un signal analogique différentiel

Les résultats expérimentaux obtenus avec le premier démonstrateur ont montré des limites dans la partie analogique, notamment un bruit important à basse fréquence et des distorsions qui n'existaient pas lors des simulations. Nous avons donc conçu une nouvelle architecture analogique, mais entièrement différentielle pour diminuer les effets d'injection de charges et les non-linéarités. La conception de l'amplificateur a également été revue afin de diminuer la consommation qui était excessive dans la première version. Tous ces détails seront présentés dans le chapitre 5. La figure 3.17 montre les représentations temporelle et spectrale d'un signal à 833 Hz, pour une longueur de registre de 120. L'amplitude du signal encodé est 360 mV, ce qui donne une amplitude double (720 mV) puisque la structure est différentielle. Cette amplitude correspond à celle du signal encodé donc avant le filtrage. Si la longueur du train binaire était 100, on se situerait à la fréquence de coupure

et on devrait avoir une atténuation de 3 dB qui correspondrait à une amplitude de 510 mV. Ici, la longueur du train binaire est plus importante (120), donc le fondamental est un peu moins atténué, ce que l'on peut constater sur la figure. Le filtre utilisé ici est à nouveau un filtre d'ordre 3. Nous voyons que la qualité du signal obtenu est tout-à-fait convenable, avec un SFDR d'environ 65 dB.

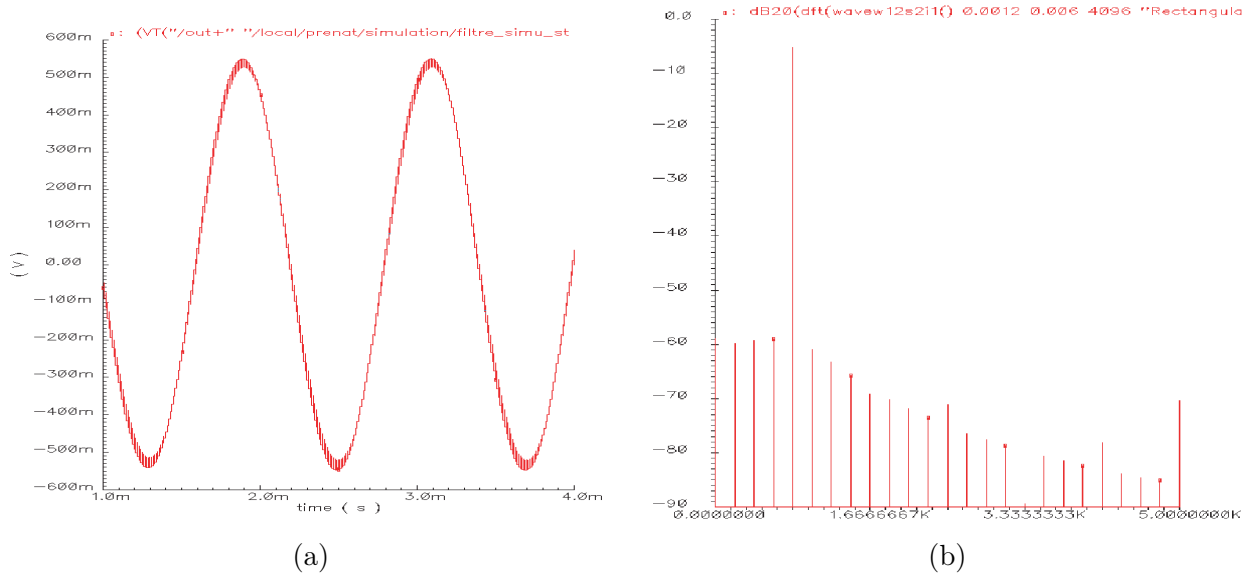


FIG. 3.17 – Représentations temporelle (a) et spectrale (b) d'un signal généré avec une partie analogique différentielle

3.5 Conclusion

Nous avons présenté dans cette section une technique de génération de signaux de test analogiques basée sur le filtrage d'un train binaire optimisé encodant le signal à générer. Cette technique a été validée par simulation et permet de générer des signaux mono ou multifréquences de bonne qualité : les résultats de simulation ont donné des SFDR de plus de 60 dB dans la quasi-totalité des cas. Dans cette technique, le choix de la fréquence se fait en programmant la longueur du train binaire et la fréquence d'horloge. La résolution en fréquence est meilleure que 1% dans ces conditions. La structure adoptée est très simple à implémenter. Le signal analogique étant généré à partir d'un train binaire, un simple filtre passe-bas est utilisé pour obtenir le signal analogique, ce qui est beaucoup plus avantageux au niveau surface qu'un convertisseur numérique-analogique classique. De plus, ce filtre peut éventuellement exister dans le reste du circuit et être réutilisé.

Cette technique souffre de deux principales limitations :

- Comme on utilise un registre de petite longueur, le choix de l'amplitude du signal est codé avec peu de précision.
- Le filtrage est analogique et agit sur un train binaire qui peut être vu comme numérique en ce sens qu'il ne peut prendre que deux valeurs, mais qui reste fondamentalement analogique. Ainsi, les variations dans les tensions d'alimentation peuvent se retrouver en sortie du filtre.

Cette technique de génération doit donc être utilisée dans une technique de BIST dans laquelle ces limitations ne sont pas handicapantes, comme nous allons le voir par la suite.

Chapitre 4

Analyse de la signature basée sur la réponse harmonique

4.1 Introduction

L'objectif du travail étant de produire une technique de test complète et entièrement numérique, nous avons cherché un moyen de créer une signature numérique représentative de la réponse analogique. La technique de génération de signaux de test délivrant des signaux sinusoïdaux, nous avons cherché à faire une analyse de type harmonique de la réponse : le but va être d'évaluer les harmoniques de celle-ci. Pour ceci, l'on pourrait effectuer une transformée de Fourier analogique de la réponse, mais cette technique nécessite de pouvoir générer sur puce des signaux sinusoïdaux à la fréquence des différents harmoniques. Une autre solution qui pourrait venir à l'esprit consisterait à numériser la réponse et à effectuer une DFT afin de mesurer les harmoniques. Cette méthode nécessite un CAN. De plus, la DFT est une opération complexe qui nécessite l'utilisation d'un DSP. Cette technique est donc trop complexe et nécessite trop de circuiterie supplémentaire pour être utilisée dans le cadre d'un BIST. Nous avons donc choisi d'utiliser une technique dérivée de celle proposée dans [112] pour évaluer la valeur des harmoniques d'un signal analogique. Cette technique est simple à implémenter et nous allons voir qu'elle se marie parfaitement avec la technique de génération de signaux pour former une technique complète et simple de BIST intégré.

Le principe de cette technique est de moduler le signal analogique par deux signaux carrés à la fréquence de l'harmonique à mesurer, décalés de $\frac{\pi}{2}$. Cette opération est similaire à la modulation par un sinus et un cosinus dans une transformée de Fourier pour obtenir ses deux composantes. Le résultat de la modulation est converti en train binaire par un modulateur $\Sigma\Delta$ du premier ordre. On peut ensuite effectuer la somme des échantillons pour extraire la valeur de la composante. Cette opération est similaire à l'intégration dans le cas d'une transformée de Fourier. Le problème qui semble se poser est qu'en utilisant des signaux carrés pour la modulation, on introduit les harmoniques impairs des signaux carrés dans la modulation. Nous allons voir qu'il ne s'agit pas d'un problème en réalité car les signatures obtenues sont utilisées ici de manière relative.

4.2 Théorie

Soit un signal s périodique, de période T (fréquence f). Sa décomposition en série de Fourier peut s'écrire :

$$s(t) = a_0 + \sum_{n=1}^{+\infty} a_n \cos(2\pi nft) + b_n \sin(2\pi nft) \quad (4.1)$$

Cette expression peut aussi s'écrire :

$$s(t) = A_0 + \sum_{n=1}^{+\infty} A_n \cos(2\pi nft + \phi_n) \quad (4.2)$$

avec $A_0 = a_0$, $a_n = A_n \cos(\phi_n)$, $b_n = -A_n \sin(\phi_n)$ et $A_n = \sqrt{a_n^2 + b_n^2}$

Les coefficients a_n et b_n sont donnés par :

$$a_n = \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} s(t) \cos(2\pi nft) dt \quad (4.3)$$

$$b_n = \frac{2}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} s(t) \sin(2\pi nft) dt \quad (4.4)$$

L'expression 4.1 fait apparaître les deux composantes (sinus et cosinus) de la transformée de Fourier. L'expression 4.2 fait apparaître le signal sous forme d'une somme d'harmoniques, chacun étant caractérisé par son amplitude A_n et sa phase ϕ_n . Ces expressions nous montrent que pour évaluer l'amplitude d'un harmonique donné d'un signal analogique, il suffit de le moduler par un sinus et un cosinus à la fréquence de l'harmonique considéré (ce qui correspond à une translation de cet harmonique vers l'origine des abscisses), puis d'intégrer le résultat pour en extraire l'amplitude. Evidemment, cette méthode n'est pas adaptée à une architecture de BIST, car elle nécessite de pouvoir générer sur la puce des signaux sinusoïdaux de très bonne qualité. C'est pourquoi nous avons opté pour une autre méthode pour extraire l'amplitude des harmoniques, proposée en [112]. La figure 4.1 montre l'architecture de cette technique.

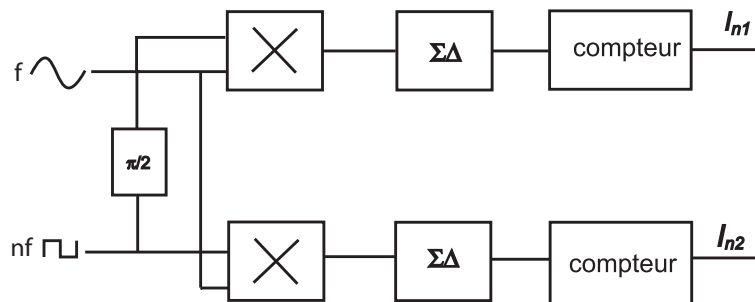


FIG. 4.1 – Architecture de la technique d'analyse de la réponse analogique du circuit sous test

Dans cette technique, les signaux sinusoïdaux utilisés pour la modulation sont remplacés par des signaux carrés, de phase 0 et $\frac{\pi}{2}$, à la fréquence de l'harmonique à évaluer. Le résultat de la modulation est encodé en un train binaire par un modulateur $\Sigma\Delta$ d'ordre 1. L'intégrateur va ainsi pouvoir être remplacé par un simple compteur. On obtient ainsi en sortie du compteur deux

nombres que nous noterons I_{n1} et I_{n2} et que nous appellerons les deux composantes de la signature I_n correspondant au $n^{\text{ième}}$ harmonique. La signature du circuit sera alors composée des signatures correspondantes à un ou plusieurs harmoniques. Bien sûr, le fait que nous utilisions des signaux carrés plutôt que sinusoïdaux pour la modulation introduit les harmoniques impairs du signal carré dans la modulation. Prenons par exemple le cas du fondamental : la modulation du signal analogique par les deux signaux carrés puis l'intégration du résultat nous amène à écrire :

$$I_{11} = \int_{-\frac{T}{2}}^0 s(t)dt - \int_0^{\frac{T}{2}} s(t)dt \quad (4.5)$$

$$I_{12} = \int_{-\frac{T}{2}}^{-\frac{T}{4}} s(t)dt - \int_{-\frac{T}{4}}^{\frac{T}{4}} s(t)dt + \int_{\frac{T}{4}}^{\frac{T}{2}} s(t)dt \quad (4.6)$$

En remplaçant $s(t)$ par son expression dans 4.2, on trouve :

$$I_{11} = - \sum_{k=0}^{+\infty} \frac{2}{\pi} A_{2k+1} \frac{\sin(\phi_{2k+1})}{2k+1} \quad (4.7)$$

$$I_{12} = \sum_{k=0}^{+\infty} \frac{2}{\pi} A_{2k+1} \frac{\cos(\phi_{2k+1})}{2k+1} \quad (4.8)$$

On voit dans ces expressions (4.8), que tous les harmoniques impairs ont une contribution dans l'expression de I_1 , mais que toutes ces contributions sont parfaitement déterminées et décroissent proportionnellement à l'ordre de l'harmonique. Par exemple, si l'on considère que $\forall k, A_{2k+1} \gg A_1$, alors on obtient :

$$I_{11} \approx - \frac{2A_1}{\pi} \sin(\phi_1) \quad (4.9)$$

$$I_{12} \approx \frac{2A_1}{\pi} \cos(\phi_1) \quad (4.10)$$

Dans ce cas, la signature peut être vue comme une approximation de l'harmonique. Cela est notamment valable dans le cas des circuits linéaires. Cela n'est plus valable dans le cas de circuits non-linéaires. En fait, nous ne cherchons pas ici à mesurer les harmoniques à proprement parler : la signature va être représentative de la valeur de l'harmonique en ce sens que les variations de la signature vont essentiellement dépendre des variations de l'harmonique (puisque la contribution des autres harmoniques décroît rapidement). Nous allons donc comparer une signature du circuit (composée, rappelons le, de signatures représentant un ou plusieurs harmoniques) à une signature nominale, qui intègre elle-même les harmoniques impairs nominaux dus à la modulation. Ces harmoniques ne vont donc pas avoir d'influence, seules leurs variations vont en avoir une, on procède de manière relative. On va ensuite choisir une limite de tolérance sur cette signature pour déterminer si le circuit est fonctionnel. De plus, pour caractériser un circuit donné dont on connaît globalement les caractéristiques et le comportement, on va pouvoir éventuellement utiliser plusieurs signatures pour apporter plus d'information et augmenter le pouvoir de discrimination de l'analyse entre circuits fonctionnels et circuits défectueux. Le calcul des composantes de I_2 et I_3 en utilisant la même méthode donne les résultats suivants. Pour I_2 ,

$$I_{21} = - \sum_{k=0}^{+\infty} \frac{2}{\pi} A_{4k+2} \frac{\sin(\phi_{4k+2})}{4k+2} \quad (4.11)$$

$$I_{22} = \sum_{k=0}^{+\infty} \frac{2}{\pi} A_{4k+2} \frac{\cos(\phi_{4k+2})}{4k+2} \quad (4.12)$$

La contribution principale est bien A_2 , la suivante est la sixième, avec une atténuation très importante. Pour I_3 ,

$$I_{31} = -\sum_{k=0}^{+\infty} \frac{2}{\pi} A_{6k+3} \frac{\sin(\phi_{6k+3})}{6k+3} \quad (4.13)$$

$$I_{32} = \sum_{k=0}^{+\infty} \frac{2}{\pi} A_{6k+3} \frac{\cos(\phi_{6k+3})}{6k+3} \quad (4.14)$$

La contribution principale est bien A_3 , la suivante est la neuvième.

La signature d'un circuit va être définie par les signatures d'un ou plusieurs harmoniques. Le choix de la signature va être effectué en fonction du type de circuit, la qualité de la signature ayant été au préalable évaluée au moyen de simulations de Monte Carlo, comme nous allons l'expliquer plus tard.

4.3 Considérations pratiques

4.3.1 Comptage

La fonction de transfert d'un compteur est donnée par :

$$H(z) = \frac{Y(z)}{X(z)} = 1 + z^{-1} + \dots + z^{-(n-1)} = \sum_{k=0}^{n-1} z^{-k} = \frac{1 - z^{-n}}{1 - z^{-1}} \quad (4.15)$$

où n est le nombre d'échantillons comptés. En remplaçant z par $e^{j2\pi fT_s}$, on obtient :

$$H(f) = \frac{Y(f)}{X(f)} = e^{j2\pi(n-1)fT_s} \frac{\sin(\pi n f T_s)}{\sin(\pi f T_s)} \quad (4.16)$$

La figure 4.2 montre le tracé de cette fonction de transfert. Nous voyons que cette fonction s'annule pour $f = \frac{f_s}{n}$ et ses multiples. Sa valeur est n pour $f = 0$. En pratique, la longueur du compteur n doit être un multiple de la longueur du registre programmable N . Ainsi, l'erreur de mesure n'est influencée par aucun harmonique introduit dans la modulation, puisque la réponse du compteur va être nulle à ces fréquences.

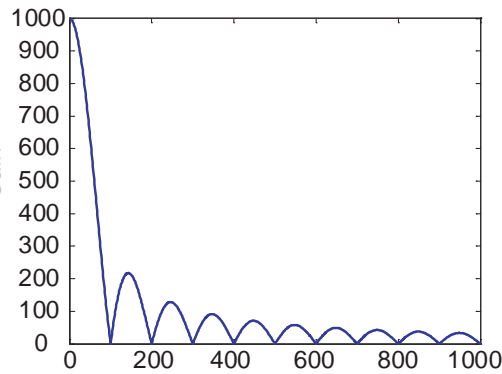


FIG. 4.2 – Fonction de transfert d'un compteur

Cependant, l'erreur de quantification due à la conversion $\Sigma\Delta$ doit être moyennée en utilisant plusieurs périodes de comptage. En effet, si nous considérons le modulateur $\Sigma\Delta$ du premier ordre présenté sur la figure 3.4 du chapitre 3, nous pouvons écrire :

$$\begin{aligned}
 V_s(n) &= V_2(n) + b(n) \\
 &= \frac{V_e(n) - \text{gain} \times V_s(n)}{z - 1} + b(n) \\
 V_s(1 + \frac{\text{gain}}{z - 1}) &= \frac{V_e(n)}{z - 1} + b(n) \\
 V_s &= \frac{V_e(n)}{z - 1 + \text{gain}} + b(n)(z - 1)
 \end{aligned} \tag{4.17}$$

Si l'on considère que le gain vaut 1, on obtient :

$$\begin{aligned}
 V_s &= \frac{V_e(n)}{z} + b(n) - b(n - 1) \\
 V_s(n) &= V_e(n - 1) + b(n) - b(n - 1)
 \end{aligned} \tag{4.18}$$

Si l'on somme maintenant P échantillons, il vient :

$$\begin{aligned}
 \sum_{n=1}^P V_s(n) &= \sum_{n=1}^P (V_e(n - 1) + b(n) - b(n - 1)) \\
 \sum_{n=1}^P V_s(n) &= \sum_{n=0}^{P-1} V_e(n) + b(P) - b(0)
 \end{aligned} \tag{4.19}$$

$b(P) - b(0)$ vaut 2 au maximum et il reste toujours le premier terme et le dernier quelque soit le nombre N d'échantillons sommés (les termes de bruit s'annulent deux à deux lors de la sommation). L'erreur de quantification n'augmente donc pas avec le nombre d'échantillons et l'erreur de quantification moyenne diminue quand le nombre d'échantillons augmente. La résolution du compteur est donc d'autant meilleure que l'on compte sur un plus grand nombre de périodes.

4.3.2 Modulation par les signaux carrés et calcul de la signature

La modulation du signal analogique par un signal carré centré en zéro est équivalente à une inversion ou non du signal analogique. Nous nous sommes demandé s'il n'était pas possible d'effectuer cette inversion sur le train binaire après le modulateur $\Sigma\Delta$ afin que l'inversion soit numérique et donc plus facile à implémenter et plus robuste. Cependant, nous nous sommes aperçus que si l'on s'y prend de cette manière, le bruit de quantification introduit par le modulateur est également modulé (donc ramené vers l'origine des fréquences) et intégré par le compteur, entraînant une erreur importante sur la mesure. La figure 4.3 montre le spectre (obtenu par Matlab/Simulink) d'un signal mono-fréquence, avant et après le comptage, en comparant les cas où la modulation a lieu avant ou après le modulateur $\Sigma\Delta$.

On voit en (a) le bruit de quantification rejeté à l'origine des fréquences juste avant le comptage. En (b), on voit le résultat du comptage avec une erreur due à ce bruit de quantification. Ce résultat est prévisible si l'on reprend les équations du modulateur $\Sigma\Delta$ de la section A.4. Supposons que l'on ajoute le dispositif de modulation après la sortie du modulateur. Appelons V_S la tension de sortie du modulateur (par le signal carré). La tension de sortie du $\Sigma\Delta$ s'appelle toujours V_s . Supposons également qu'on s'intéresse au premier harmonique (on effectue donc une modulation par un signal

carré de la même période que le train binaire). Considérons alors que l'on somme les Q échantillons d'une seule période. On peut alors écrire (en supposant pour simplifier que Q est pair) :

$$\begin{aligned}
 \sum_{n=1}^Q V_S &= \sum_{n=\frac{Q}{2}+1}^Q V_s - \sum_{n=1}^{\frac{Q}{2}} V_s \\
 &= \sum_{n=\frac{Q}{2}+1}^Q (V_e(n-1) + b(n) - b(n-1)) - \sum_{n=1}^{\frac{Q}{2}} (V_e(n-1) + b(n) - b(n-1)) \\
 &= \sum_{n=\frac{Q}{2}+1}^Q V_e(n-1) - \sum_{n=1}^{\frac{Q}{2}} V_e(n-1) + b(Q) - b\left(\frac{Q}{2} + 1\right) - b\left(\frac{Q}{2}\right) + b(0) \quad (4.20)
 \end{aligned}$$

On voit que, dans ce cas, deux termes de bruit supplémentaires apparaissent à cause du changement de signe du signal carré de modulation. En fait, à chaque changement de signe, deux termes de bruit ne seront pas éliminés et ceci tout le long de l'intégration. L'erreur de bruit va donc augmenter à chaque période et le fait de compter sur plusieurs périodes ne va plus améliorer la précision de comptage. Ceci est bien sûr valable pour tous les harmoniques. Nous avons essayé plusieurs solutions pour résoudre ce problème, notamment d'introduire la modulation dans le modulateur $\Sigma\Delta$ ou encore d'introduire un bruit blanc pour améliorer la linéarité. Aucune de ces solutions n'a cependant offert les mêmes performances que la modulation (par le signal carré) avant le modulateur $\Sigma\Delta$, comme montré sur la figure 4.1. Nous avons donc décidé d'en rester à une modulation du signal analogique. Le signal étant différentiel, cette modulation est très facile à effectuer : il suffit d'invertir les sorties du filtre analogique. Pour cela, nous utilisons des interrupteurs analogiques pilotés par le signal carré de modulation.

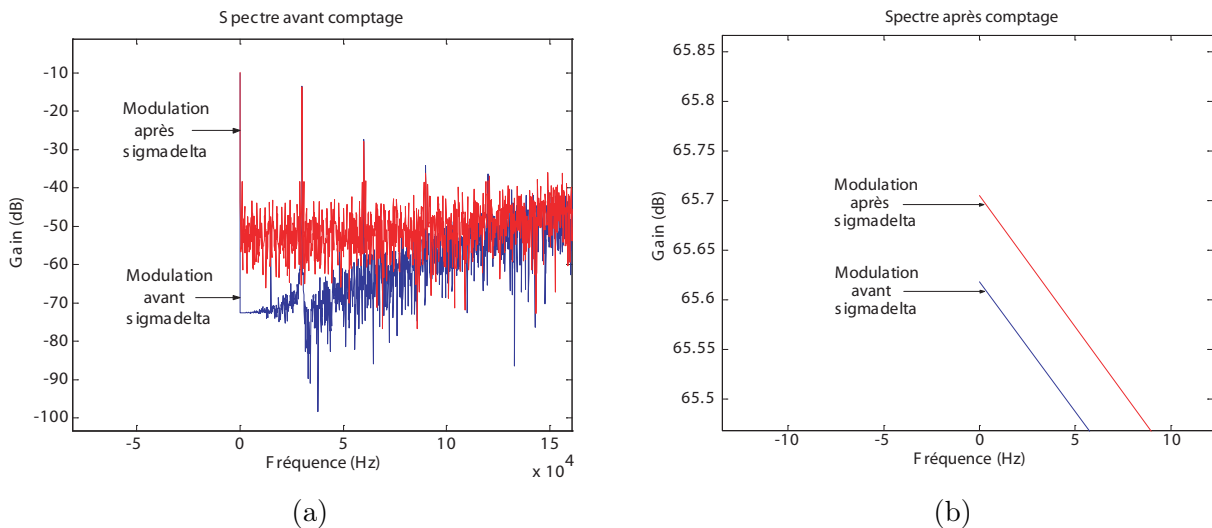


FIG. 4.3 – Comparaison du spectre du signal modulé avant et après le $\Sigma\Delta$: (a) signal avant le compteur et (b) signal après le compteur

4.4 Optimisation de la signature

4.4.1 Paramètres permettant d'optimiser la signature

Comme nous avons signalé dans l'état de l'art, il y a deux approches dans l'évaluation d'une technique de test :

- Si l'on cherche à effectuer un test structurel, il faut introduire une notion de faute dans le domaine analogique comme nous l'avons expliqué dans la section 2.2.1.
- Mais quand on effectue un test fonctionnel comme c'est le cas ici, avec une analyse harmonique de la réponse, alors ces notions ne sont plus nécessaires.

Dans l'approche qui va être décrite maintenant, nous allons simuler le test d'un grand nombre de circuits dont les paramètres varient. Cependant, ces paramètres vont varier d'une façon relativement irréaliste dans le sens où ces variations ne seront pas celles d'une technologie réelle. Or, nous avons vu dans l'introduction de la notion de faute dans le test analogique au début de l'état de l'art qu'on ne peut pas vraiment évaluer la qualité d'un test sans connaître les paramètres technologiques associés. Le but de notre démarche est plutôt d'expliquer sur des exemples simplifiés et généraux comment on peut choisir une signature et évaluer son efficacité. La démarche est donc plus "mathématique" que technologique et c'est pourquoi nous préférons parler d'optimisation de la signature plutôt que d'évaluation de la qualité du test. La figure 4.4 présente les paramètres que nous allons chercher à mesurer pour optimiser la signature.

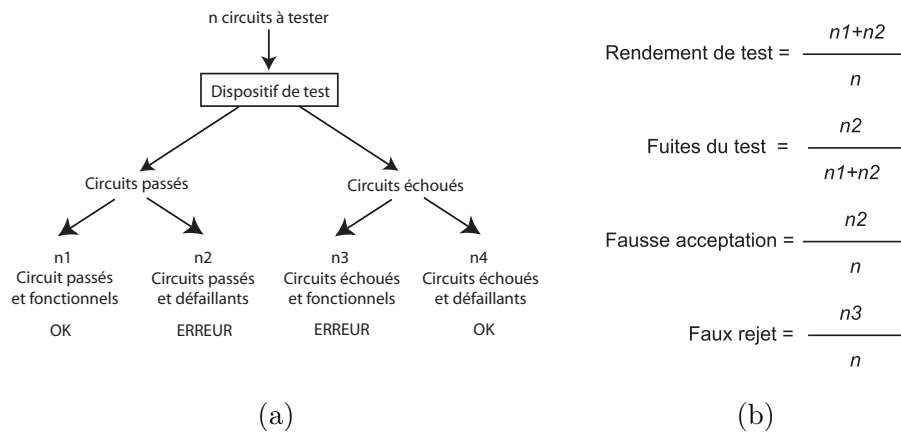


FIG. 4.4 – Les paramètres permettant d'évaluer la qualité de la signature

Notons que le rendement de test, qui est défini par le rapport entre le nombre de circuits passant le test et le nombre total de circuits n'est pas réellement un paramètre permettant d'évaluer la qualité de la signature : en effet, il ne tient pas compte de la proportion de circuits fonctionnels ou défaillants parmi ceux qui passent le test. Un mauvais rendement peut donc être dû à un mauvais processus de fabrication (un grand nombre de circuits sont défaillants, par exemple à cause de marges de conception trop petites) ou à un trop fort rejet du test (trop de circuits fonctionnels sont rejetés). Notons aussi que le taux de fuites du test (défini par le rapport entre le nombre de circuits défaillants ayant passé le test et le nombre de circuits ayant passé le test) et le taux de fausse acceptation (rapport entre le nombre de circuits défaillants ayant passé le test et le nombre total de circuits) ont des significations très similaires. Nous ne considérerons donc pour l'optimisation de la signature que les taux de faux rejet et de fausse acceptation (défini par le rapport entre le nombre de circuits fonctionnels ayant échoué au test et le nombre total de circuits).

4.4.2 Résultats de simulation

Dans l'absolu, pour évaluer les paramètres précédemment définis, il faudrait fabriquer un très grand nombre de circuits, les tester par la méthode proposée et vérifier par une méthode classique que les performances sont bien les performances voulues. Les circuits qui passeraient le test seraient alors considérés comme "passés", les autres "échoués". La vérification des performances permettrait de vérifier véritablement si le circuit fonctionne correctement ou non et donc de le classer comme fonctionnel ou défaillant. On aurait ainsi toutes les valeurs nécessaires pour calculer les paramètres d'évaluation de la signature. Cependant, le nombre de circuits à fabriquer et le temps de test qui seraient nécessaires interdisent d'avoir recours à une telle méthode. Nous allons donc procéder par simulations de Monte Carlo en utilisant le logiciel Matlab. Pour cela, nous allons considérer le filtre passe-bas du troisième ordre comme circuit sous test (figure 4.5).

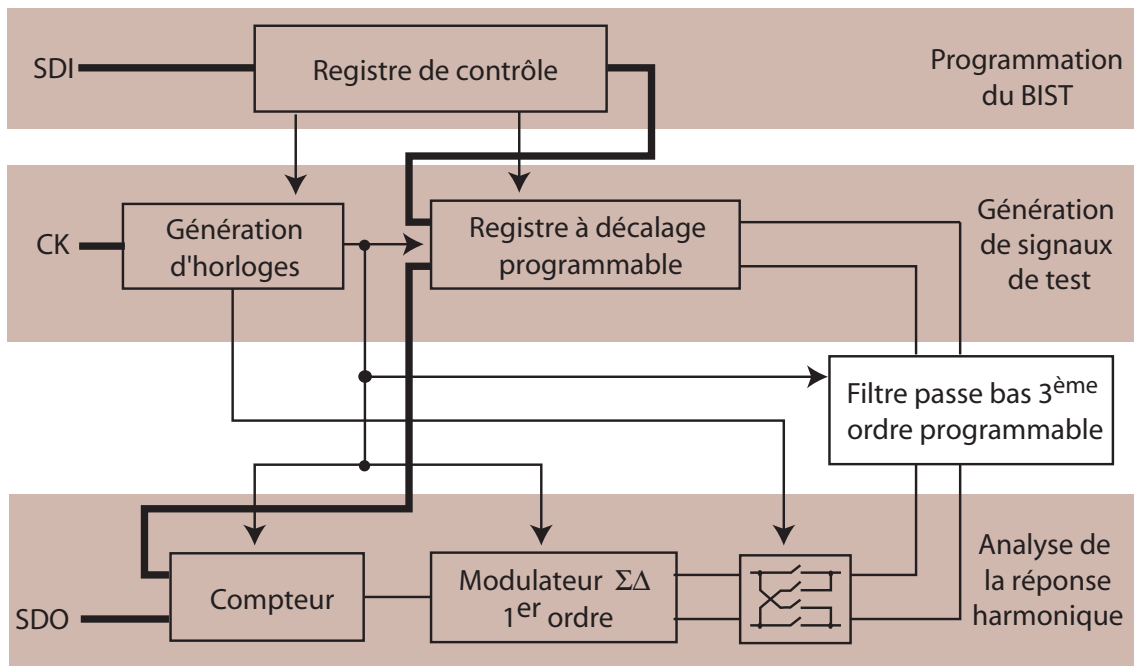


FIG. 4.5 – Architecture de la technique de BIST avec le filtre passe-bas comme circuit sous test

La fonction de transfert de ce filtre peut s'écrire sous la forme suivante (équation 4.21) :

$$H(p) = \frac{a_0}{a_1 + a_2s + a_3s^2 + a_4s^3} \quad (4.21)$$

Les simulations de Monte Carlo vont consister à faire dévier les paramètres du filtre (ici ses coefficients), selon une distribution gaussienne. On parlera, pour être clair, d'espace des paramètres pour désigner l'ensemble des valeurs prises par les coefficients du filtre. Afin de tester le cas non-linéaire, l'on peut aussi ajouter un terme non-linéaire au filtre. La figure 4.6 montre les paramètres de ces déviations, avec un écart-type σ .

On génère ainsi un certain nombre d'instances. Chacune d'entre elles va pouvoir être simulée, afin de pouvoir vérifier ses performances. Ici, les performances sont le gain, la fréquence de coupure et éventuellement le SFDR (si l'on se place dans le cas non-linéaire). On parlera d'espace des performances pour désigner l'ensemble des valeurs de performances prises par ces instances. Cet ensemble est illustré sur la figure 4.7. Les tolérances choisies vont permettre de classer les instances

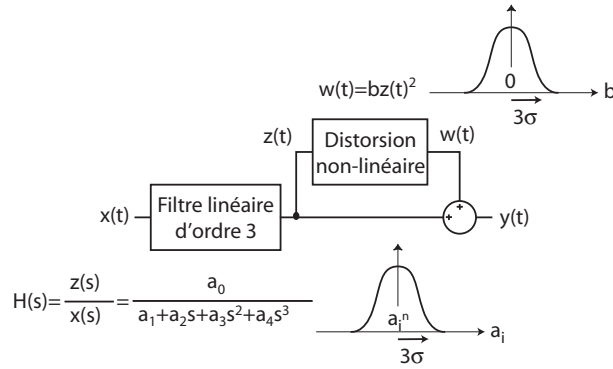


FIG. 4.6 – Espace des paramètres

comme fonctionnelles ou défailantes selon qu'elles respectent ou non ces spécifications. Ces limites reviennent donc à choisir une limite dans l'espace des performances.

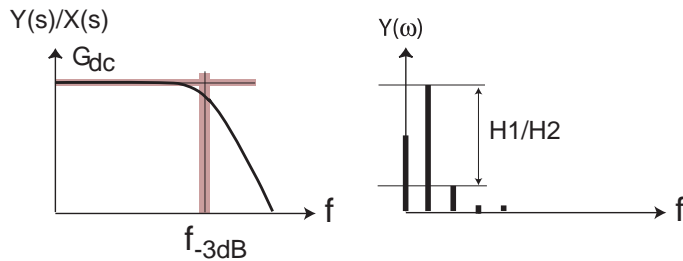


FIG. 4.7 – Espace des performances

Enfin, pour chacune de ces instances nous allons pouvoir calculer une signature et obtenir ainsi un dernier espace, que nous appellerons espace des signatures. Il faudra ensuite choisir une limite dans cet espace des signatures, qui va permettre de classer les instances comme ayant passé ou échoué au test. La figure 4.8 illustre cette opération.

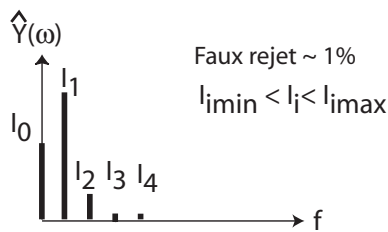


FIG. 4.8 – Espace des signatures

A la fin de ces opérations, on va avoir un certain nombre d'instances générées (n). Un certain nombre d'entre elles (n_1) ont passé le test tout en étant fonctionnelles. D'autres, en nombre n_4 ont échoué au test en étant défailantes. Par contre un certain nombre d'entre elles (respectivement n_2 et n_3) ont soit passé le test alors qu'elles étaient défailantes, soit échoué au test alors qu'elles étaient fonctionnelles. Ce sont des erreurs du test. On va donc pouvoir évaluer les paramètres définis sur la figure 4.4.

4.4.3 Cas linéaire

Dans un premier temps, afin d'illustrer le fonctionnement de la technique d'analyse, nous avons considéré le filtre parfaitement linéaire (le coefficient de non-linéarité a été choisi nul). Nous avons généré 1000 circuits sous Matlab/Simulink en choisissant des déviations telles que 3σ vaille 5% pour les coefficients. Le nombre de périodes de comptage est 1000 également. Le train binaire d'entrée a été introduit depuis un fichier issu du logiciel de CAO : ce train binaire encode un signal mono-fréquence de valeur DC égale à 0,0167 V et de valeur de fondamental égale à 0,369 V. La fréquence du signal a été fixée à 15 kHz et la longueur du registre est $N=120$ (la fréquence d'échantillonnage est donc 1,8 Mhz). Chacun des circuits a été simulé afin de déterminer ses performances. Pour cela, nous avons réalisé une FFT de l'entrée et de la sortie du filtre pour mesurer leurs harmoniques avec précision. Le gain est alors calculé par le rapport (en dB) entre les valeurs DC de sortie et d'entrée. Le SFDR est calculé comme le rapport (toujours en dB) entre la valeur du fondamental et la valeur de l'harmonique le plus important. En ce qui concerne la fréquence de coupure, la mesure est plus difficile à effectuer : en effet, par la FFT nous n'avons accès qu'à la valeur des harmoniques de l'entrée et de la sortie. Or, pour mesurer la fréquence de coupure à -3 dB, il faudrait connaître la répartition spectrale des signaux avec une bonne résolution afin de trouver la fréquence correspondant à une atténuation de 3 dB. Ceci demande une simulation complète et donc longue du circuit. Comme nous avons besoin de générer et de simuler beaucoup d'instances pour avoir un échantillon représentatif, les temps de simulation deviennent rapidement inacceptables. Dans le cas linéaire, nous pourrions évaluer cette fréquence par calcul à partir des coefficients de chaque circuit. Mais dans le cas non-linéaire, les notions de fonction de transfert, de fréquence de coupure n'ont plus réellement de sens. Nous avons donc préféré simplifier l'analyse aussi bien dans le cas linéaire que dans le cas non linéaire en évaluant l'atténuation du fondamental, en calculant le rapport en dB entre le fondamental du signal d'entrée et celui du signal de sortie. Cette performance n'est pas exactement équivalente à la fréquence de coupure, mais elle est similaire en ce sens que les variations de la fréquence de coupure vont essentiellement se retrouver dans l'atténuation du fondamental. Nous disposons alors de trois performances qui vont permettre de classer chaque circuit comme fonctionnel ou défaillant selon qu'il s'éloigne plus ou moins des valeurs nominales de ces performances avec les limites suivantes : le gain en dB doit être de zéro (gain nominal) plus ou moins 0,40 dB et l'atténuation doit être de 3,0123 dB (atténuation nominale) plus ou moins 0,60 dB (notons que l'atténuation nominale n'est pas exactement 3 dB, car la longueur du registre est 120 et non pas 100, voir chapitre 3). Ces valeurs sont choisies de manière empirique de manière à avoir un nombre de circuits fonctionnels et un nombre de circuits défaillants suffisamment important pour que ces populations soient valables du point de vue statistique. De plus, nous nous sommes arrangés pour que le pouvoir discriminant de ces deux performances soit à peu près le même (par exemple, si l'on prend une tolérance trop importante sur l'atténuation du fondamental, cela ne changera rien sur la classification des circuits par rapport à la seule limite sur le gain). Le SFDR n'est pas considéré, le circuit étant linéaire.

Parallèlement à la simulation du circuit, nous avons calculé les signatures normalisées (c'est-à-dire le rapport entre la signature et la signature nominale) pour chacun de ces circuits et choisi une frontière dans l'espace des signatures de manière à avoir un taux de faux rejet aux alentours de 1%. Ce choix des frontières se fait de la manière suivante : on définit une distance entre chaque instance et l'instance nominale comme le maximum des écarts entre les composantes de sa signature et celles de l'instance nominale (1,1,1) :

$$d(instance_i, instance_{nom}) = \max \left(\left| \frac{I_0}{I_{0nom}} - 1 \right|, \left| \frac{I_1}{I_{1nom}} - 1 \right|, \left| \frac{I_2}{I_{2nom}} - 1 \right| \right) \quad (4.22)$$

On va ensuite classer les circuits ayant passé le test selon leur distance à l'instance nominale. Le pourcent d'instances les plus éloignées va être éliminé, ce qui revient à créer 1% de faux rejet. Dans notre cas, par exemple, nous avons 1000 instances. On va donc les classer par distance croissante et choisir comme limite de distance la valeur de distance correspondant à la 989^{ième} instance. Les dix plus éloignées seront donc éliminées et on aura créé une frontière permettant d'avoir un faux rejet de 1%. Il est ensuite facile de déterminer les frontières correspondantes sur chaque composante de la signature. Grâce à ces frontières, on détermine si chaque instance passe le test ou non, et nous pouvons calculer les paramètres qui sont présentés dans le tableau 4.1.

1000 périodes		Espace des signatures						
		I_0	I_1	I_2	$I_0 \& I_1$	$I_0 \& I_2$	$I_1 \& I_2$	$I_0 \& I_1 \& I_2$
Paramètres du test	Rendement	94.00	95.70	98.00	91.20	93.90	98.30	91.20
	Fuites	2.979	4.702	7.245	0	2.875	7.223	0
	FA	2.800	4.500	7.100	0	2.700	7.100	0
	FR	1.000	1.000	1.300	1.000	1.000	1.000	1.000

TAB. 4.1 – Les paramètres de qualité du test dans le cas d'un circuit purement linéaire

On voit que les signatures correspondant à la valeur DC ou à la valeur A1 ne sont pas efficaces prises indépendamment. Ceci est tout-à-fait normal : en effet, nous avons choisi des limites sur le gain et la fréquence de coupure, et il faut donc évaluer les valeurs des deux premiers harmoniques pour pouvoir estimer si les instances restent dans les spécifications ou non. De plus, le circuit est linéaire, donc il est normal que le second harmonique ne soit pas du tout représentatif. En revanche, lorsque nous considérons les signatures IO et I1 correspondant à la valeur DC et au premier harmonique, le taux de fausse acceptation devient nul. Afin d'avoir un résultat plus visuel, nous avons tracé la répartition des différentes signatures pour les différences instances. La figure 4.9 représente les signatures IO et I1 prises indépendamment, la figure 4.10 les mêmes signatures prises ensemble. Afin d'avoir un résultat facilement visualisable, nous avons représenté les valeurs normalisées de ces signatures. C'est pourquoi ces signatures sont toujours centrées autour de la valeur 1. Ceci restera valable pour toutes les représentations graphiques des signatures à venir. On voit également en pointillés les frontières choisies comme expliqué précédemment pour chaque composante.

Sur ces figures, les circuits fonctionnels apparaissent sous forme de croix bleues, et les circuits défaillants sous forme de ronds rouges. On peut ainsi constater que dans le cas de la signature constituée de la valeur DC et du premier harmonique, les deux ensembles sont très bien séparés, donc la signature est efficace. En revanche, les signatures correspondant à la valeur DC et au premier harmonique pris indépendamment ne sont pas efficaces ici. Pour information, nous donnons sur la figure 4.11 les représentations graphiques des répartitions des signatures A2, DC et A2, A1 et A2 et enfin DC, A1 et A2. On peut remarquer sur ces figures que dans le cas du second harmonique, les signatures prennent des valeurs discrètes, rendant les résultats inexploitable. Ceci est dû au fait que l'on atteint la limite de résolution du compteur (voir section A.4). Ceci est également responsable de la valeur de faux rejet qui ne vaut pas rigoureusement 1% : en effet, le regroupement des valeurs des signatures en valeurs discrètes fait qu'il est difficile d'exclure un nombre précis de circuits puisque plusieurs vont se trouver à la même distance de la valeur nominale. On peut repousser cette limite en augmentant le nombre de périodes de test mais au détriment de la vitesse. Ceci illustre le compromis à trouver entre durée de test et résolution lors de l'utilisation de cette technique pour la mesure de toutes petites valeurs de tension, qui nécessite alors une grande résolution.

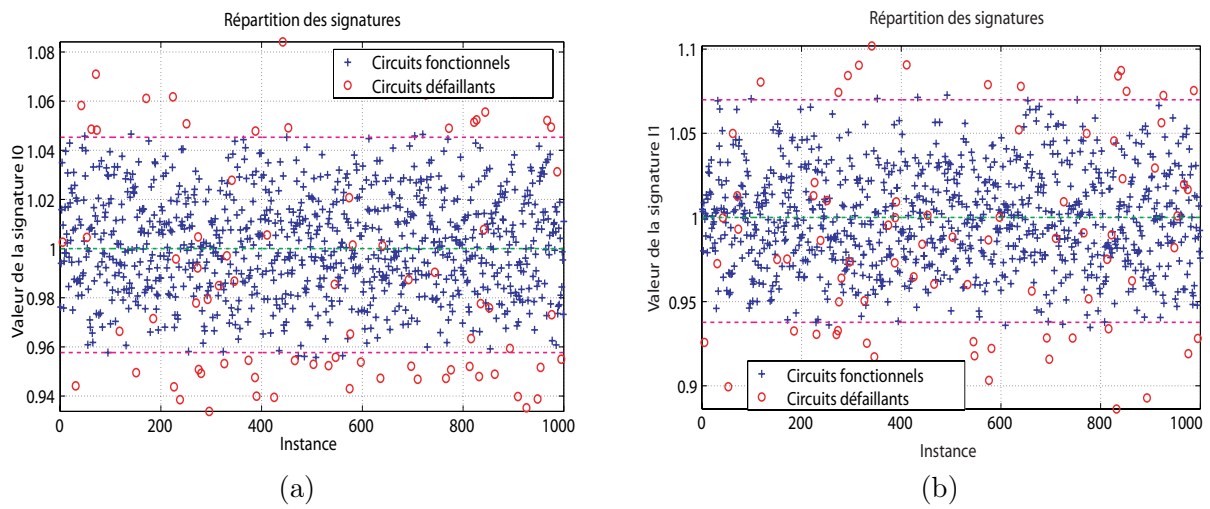


FIG. 4.9 – Répartitions des signatures correspondant : (a) à la valeur DC et (b) au premier harmonique

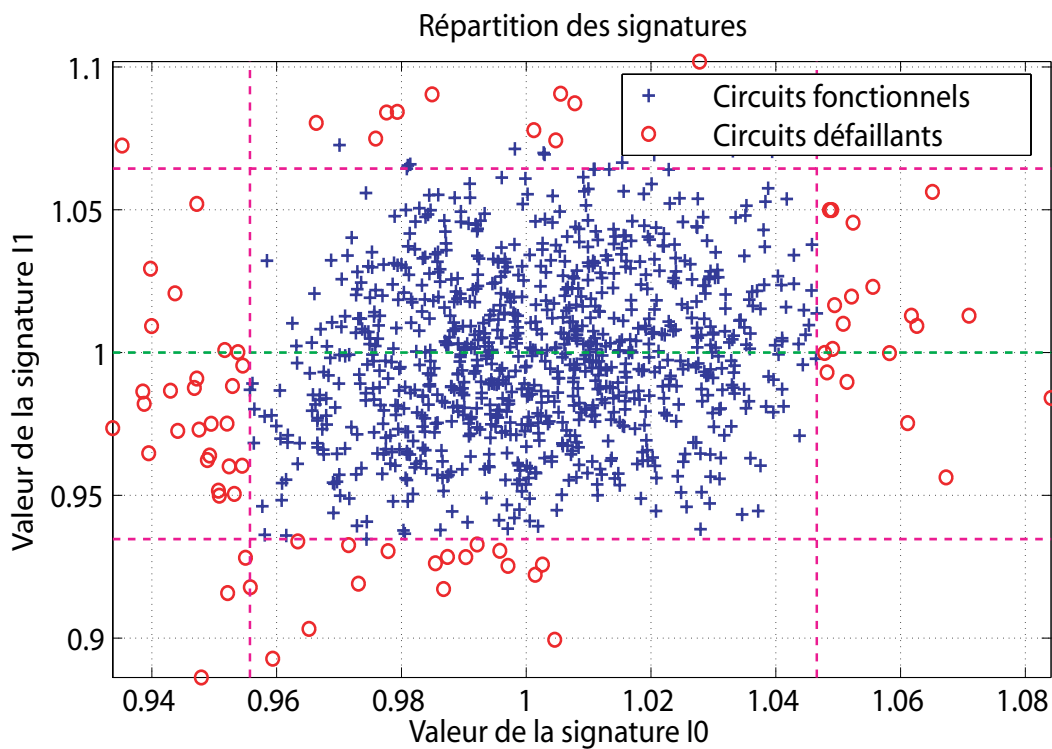


FIG. 4.10 – Signature correspondant aux deux premiers harmoniques

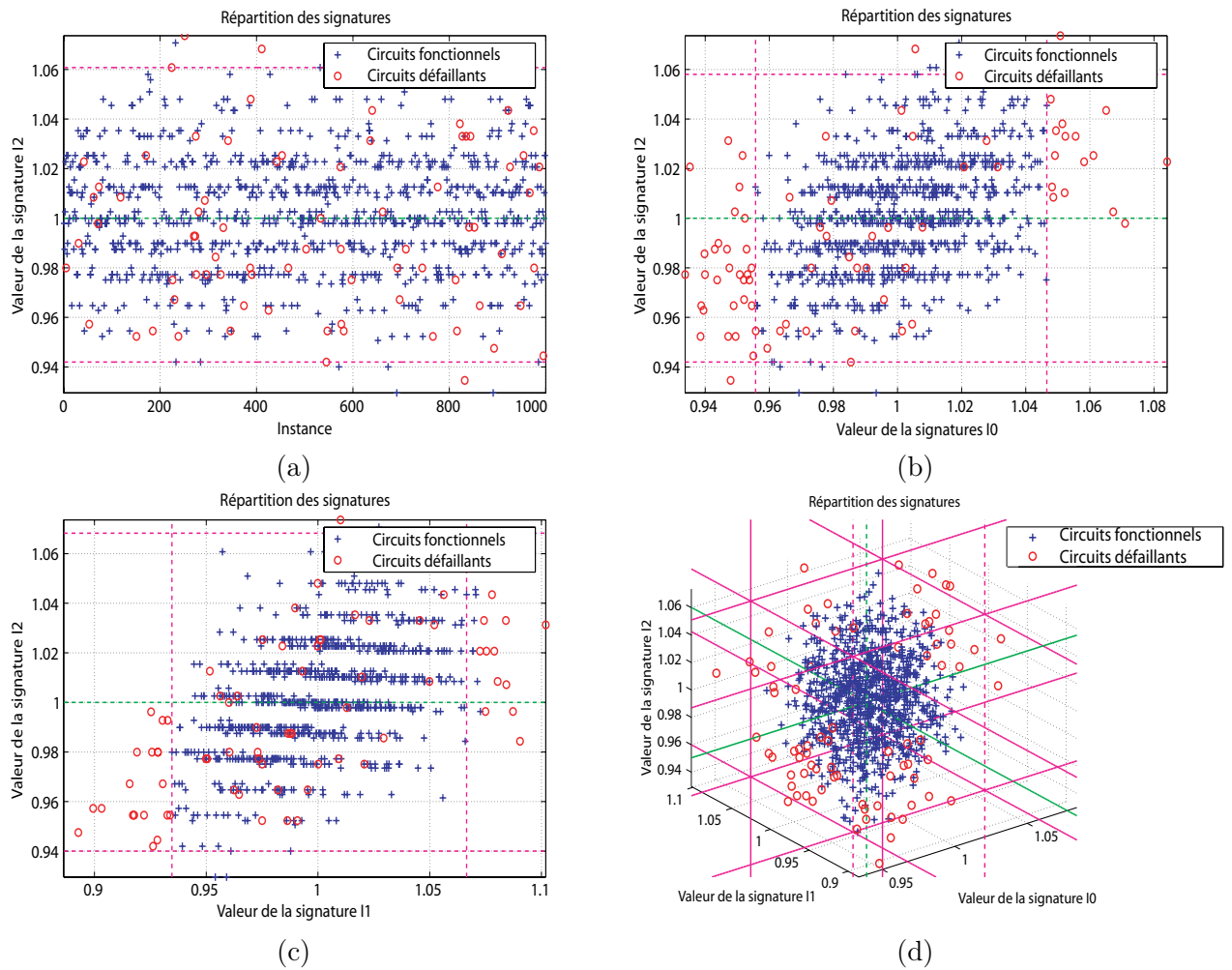


FIG. 4.11 – Répartitions des signatures correspondant : (a) à la valeur A_2 , (b) aux valeurs de DC et A_2 , (c) aux valeurs de A_1 et A_2 et (d) aux trois premiers harmoniques

4.4.4 Cas non-linéaire

Considérons maintenant le même filtre, avec le terme non-linéaire. Cette non-linéarité sera modélisée par un terme au carré. Si l'on regarde la figure 4.12, on a alors :

$$y(t) = w(t) + z(t) = z(t) + gz(t)^2 \quad (4.23)$$

Si l'on considère un signal parfaitement sinusoïdal en sortie du filtre, avec une composante continue, on va avoir $z(t) = Z_0 + Z_1 \cos \omega t$. Le signal analogique obtenu va être donné par :

$$y(t) = A_0 + g \left(A_0^2 + \frac{A_1^2}{2} \right) + A_1 (2gA_0 + 1) \cos \omega t + g \frac{A_1^2}{2} \cos 2\omega t \quad (4.24)$$

Les variations de g ont donc une influence sur les valeurs des trois harmoniques, et nous allons voir si la méthode d'analyse permet de détecter efficacement ces variations.

Pour cela, nous reprenons la même technique que précédemment, mais avec le terme non-linéaire, comme indiqué sur la figure 4.12.

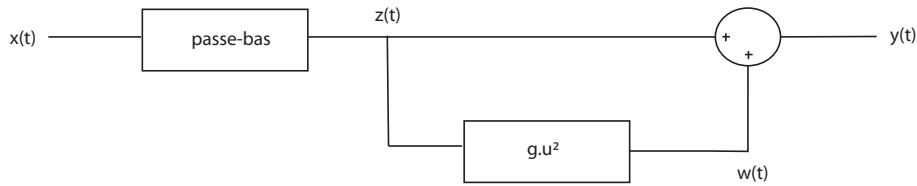


FIG. 4.12 – *Filtre avec terme non-linéaire*

Nous pouvons alors relancer les simulations de Monte Carlo : les coefficients du filtre dévient de la même façon que précédemment. Le coefficient g est centré en zéro et subit une déviation d'écart-type 0.1. Nous générons à nouveau 1000 instances et nous comptons pendant 1000 périodes. Les caractéristiques du signal d'entrée et la longueur du registre sont les mêmes que dans l'analyse dans le cas linéaire. Les limites sur les performances ont cependant été modifiées, toujours pour avoir une population de circuits représentative : le gain doit être de zéro plus ou moins 0,45 dB. L'atténuation est toujours de 3,0123 plus ou moins 0,60 dB. Nous ajoutons ici une limite sur le SFDR qui doit être supérieur à 47 dB.

Nous relançons les simulations de Monte Carlo et nous obtenons les résultats donnés sur les figures 4.13 et 4.14 ainsi que dans le tableau 4.2. On voit ici qu'il est nécessaire de mesurer les trois premiers harmoniques pour avoir une signature efficace, ce qui est logique puisque tous les harmoniques sont ici touchés par les variations de paramètres. Dans ces conditions, en gardant un taux de faux rejet de 1%, on obtient des taux de fausse acceptation de moins de 1%. Nous avons aussi donné sur les figures 4.13 et 4.14 les représentations graphiques des répartitions des différentes signatures. Nous voyons que la seule signature discriminant correctement les circuits est celle constituée des trois composantes.

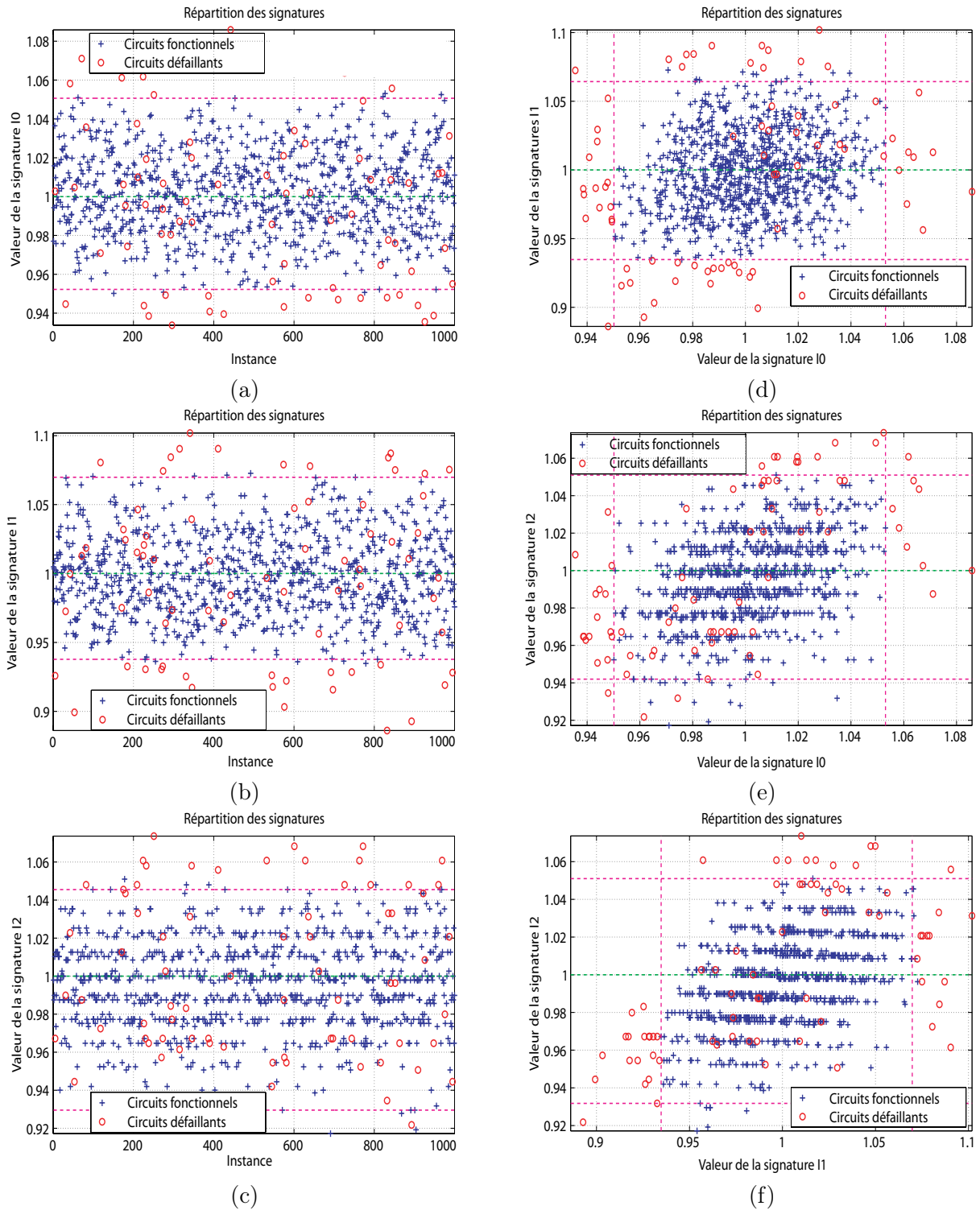


FIG. 4.13 – Répartitions des signatures correspondant : (a) à la valeur DC, (b) au premier harmonique, (c) au second harmonique, (d) à la valeur DC et au premier harmonique, (e) à la valeur DC et au second harmonique et (f) aux premier et second harmoniques

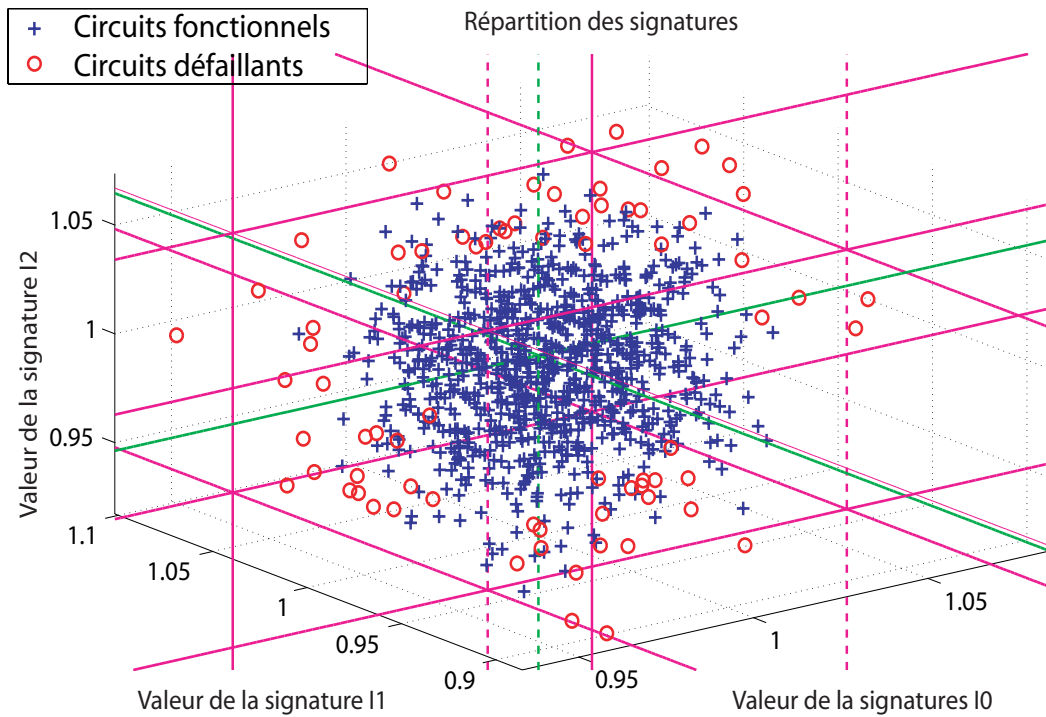


FIG. 4.14 – Signature correspondant aux trois premiers harmoniques

1000 périodes		Espace des signatures						
		I_0	I_1	I_2	$I_0 \& I_1$	$I_0 \& I_2$	$I_1 \& I_2$	$I_0 \& I_1 \& I_2$
Paramètres du test	yield	96.30	95.70	96.40	93.30	94.80	97.10	92.40
	Fuites	4.880	4.284	5.705	1.822	3.692	5.664	0.8658
	FA	4.700	4.100	5.500	1.700	3.500	5.500	0.8000
	FR	1.000	1.000	1.700	1.000	1.300	1.000	1.000

TAB. 4.2 – Les paramètres de qualité du test dans le cas d'un circuit non-linéaire

4.5 Conclusion

Nous avons présenté une technique d'analyse de signature basée sur l'évaluation des harmoniques de la réponse analogique. Cette méthode ne permet pas réellement de mesurer la valeur des harmoniques dans le cas général à cause des harmoniques introduits par l'utilisation d'un signal carré. Cependant, l'erreur introduite par ces harmoniques est parfaitement déterministe. Comme on procède par comparaison d'une signature avec une signature idéale, ces harmoniques sont déjà présents dans cette signature de référence et ne sont pas gênants, seules leurs variations ont une influence. Il semble donc que la seule limitation en résolution de la technique vienne de la résolution du compteur qui dépend essentiellement du nombre de périodes de comptage.

En nous fixant un taux de rejet de 1%, nous arrivons à obtenir un taux de fausse acceptation en dessous de 1% dans tous les cas, ce qui est satisfaisant. Il est probable qu'en augmentant le nombre de périodes de comptage de manière significative, le résultat puisse être amélioré aux dépens de la vitesse du test. Cependant, les simulations de Monte Carlo deviennent très longues et nous

n'avons pas vraiment réussi à vérifier cette hypothèse. Comme nous l'avons précisé, ces résultats sont à prendre avec précaution, puisqu'ils ne correspondent pas à la réalité d'une technologie. Par exemple, pour avoir un nombre de circuits défaillants statistiquement valable, nous avons été obligés de diminuer le rendement de notre production virtuelle. Dans la pratique, le rapport entre le nombre de circuits défaillants et le nombre total de circuits est beaucoup plus faible et les résultats de fausse acceptation et faux rejet devraient être plus bas.

Cette technique est très intéressante dans le sens où elle n'est pas limitée à un type de circuits particulier et elle peut notamment être utilisée dans le cas de circuits non-linéaires. Elle est particulièrement efficace associée à la technique de génération de signaux de test présentée :

- Le signal analogique ayant une période multiple de la fréquence d'horloge, la génération de signaux carrés pour la modulation est très simple et consiste seulement en une division de fréquence, comme nous l'expliquerons plus en détails dans le chapitre 5.
- Les parties analogiques de l'architecture étant entièrement différentielles, la modulation par ces signaux carrés consiste seulement à intervertir les sorties différentielles du filtre analogique.

Nous pouvons aussi noter que les limitations possibles de la partie génération décrite au chapitre précédent ne sont pas gênantes ici :

- Comme on procède par comparaison de la signature d'un circuit avec la signature nominale, le manque de résolution dans le codage de l'amplitude n'est pas un problème. Ce qui est important, c'est que cette amplitude soit déterministe, c'est-à-dire que pour un train binaire donné, l'amplitude soit toujours la même, ce qui est le cas au moins du point de vue théorique.
- Les fluctuations de niveau du train binaire à l'entrée du filtre qui peuvent affecter la qualité du signal analogique généré vont être moyennées pendant le comptage et leur influence devrait être faible. Il faudrait cependant vérifier cette hypothèse par simulation. Mais l'introduction d'un bruit alourdit beaucoup les simulations et nous n'avons pas eu le temps de les mener à leur terme.

La génération de signaux de test permettant également de générer des signaux multi-fréquence, il serait intéressant d'étudier une méthode d'analyse utilisant ce type de signaux et d'estimer l'intérêt que pourrait avoir la technique d'analyse présentée dans ce chapitre pour une telle méthode.

Chapitre 5

Prototypes réalisés

5.1 Introduction

Afin de valider cette technique, nous avons réalisé deux démonstrateurs sur silicium, en technologie CMOS $0,18\mu m$ de ST-Microelectronics. Ces démonstrateurs ont été fabriqués grâce au service CMP¹. Cette réalisation a permis notamment d'aborder en totalité la partie pratique de la conception analogique et mixte, du schéma jusqu'au layout final avec les plots de connexion, les simulations et vérifications nécessaires à toutes les étapes.

5.2 Premier démonstrateur : circuit de génération

5.2.1 Schéma et résultats de simulation

Le premier prototype, réalisé en première année de thèse, contient le circuit de génération de signaux analogiques (figure 5.1). Pour plus de détails sur la réalisation de celui-ci, se référer à [92]. Deux circuits ont été fabriqués, l'un contenant un banc de filtres d'ordres 2 à 6 (voir chapitre 3) et un registre à décalage de 1024 bits dans un but de comparaison, l'autre, moins gourmand en surface, contenant uniquement le filtre d'ordre 3 et le registre de longueur programmable entre 100 et 200 bits. Décrivons le fonctionnement de cette architecture.

Partie numérique

Le signal SDI (Scan Data In) est un signal binaire, qui contient le train binaire encodant le signal analogique à générer (voir chapitre 3), et les bits de programmation de l'architecture. Ces bits de programmation sont au nombre de 17 et sont chargés en série dans le registre de contrôle. Ils sont ensuite redistribués à l'ensemble du circuit.

Les bits S_1 à S_5 servent à choisir le facteur de division pour le diviseur de fréquence par une puissance de 2. Si nous appelons clk l'horloge arrivant sur la puce, et $clock$ l'horloge cadencant le circuit, leurs fréquences respectives sont liées par la relation $fclock = \frac{fclk}{2^B}$, où B varie entre 1 et 20. Ce diviseur de fréquence est un simple enchaînement de bascules D, chacune des sorties étant reliée à l'horloge de la bascule suivante. Chaque bascule divise la fréquence par deux. Le choix de la sortie est réalisé par un multiplexeur.

1. Circuits Multi Projets : service du CNRS permettant de fabriquer des circuits en partageant une plaquette entre plusieurs concepteurs

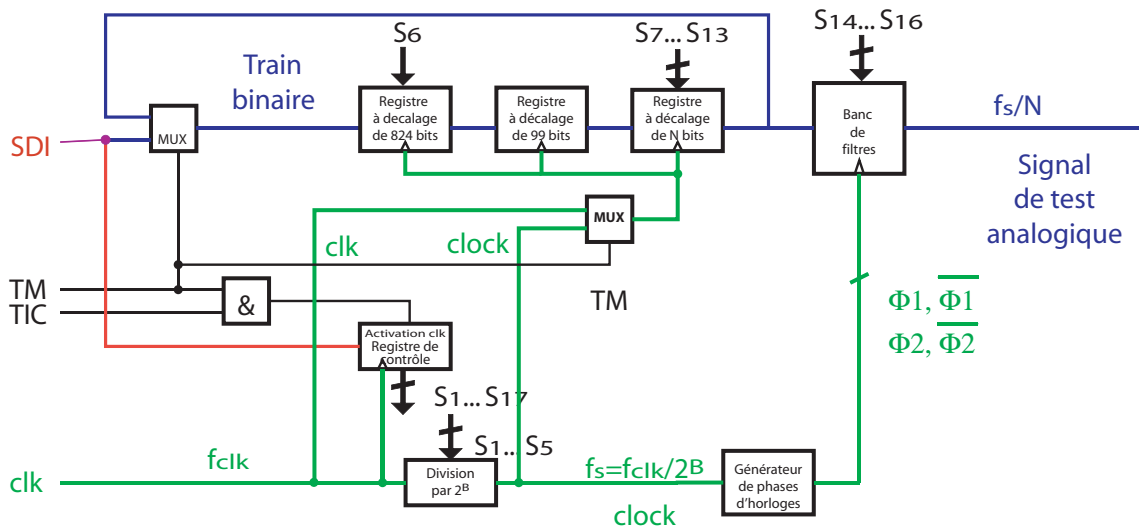


FIG. 5.1 – Schéma du circuit de génération de signaux de test analogiques

Les bits S_7 à S_{13} permettent de choisir la longueur du registre à décalage. En fait, afin de pouvoir comparer les performances de la technique avec celles utilisant un registre beaucoup plus long, nous avons choisi d'avoir la possibilité de disposer d'un registre à décalage de 1024 bits. Le registre est donc constitué de trois parties :

- un registre de 824 bits qui peut être actif ou non (bit S_6),
- un registre de 99 bits,
- un registre dont la taille peut varier entre 1 et 101 bits (S_7 à S_{13}), le choix de la sortie étant réalisé par un multiplexeur.

On a donc la possibilité d'avoir un registre de 1024 bits, ou un registre dont la longueur varie entre 100 et 200 bits.

Les bits S_{14} à S_{16} permettent de choisir l'ordre du filtre. Comme nous l'avons vu dans le chapitre 3, nous voulons avoir la possibilité de choisir entre des filtres de Butterworth d'ordres 2 à 6, afin d'en comparer les performances. Ces filtres sont à capacités commutées, il faut donc générer des horloges pour les interrupteurs de ces capacités. Ceci est réalisé par le générateur de phases, qui génère les horloges non recouvrantes ϕ_1 , ϕ_2 , $\overline{\phi_1}$ et $\overline{\phi_2}$. Le schéma du générateur de phases est donné sur la figure 5.2. Il est constitué d'une logique à base de portes NAND qui génère deux signaux ϕ_1 et ϕ_2 en opposition de phase à partir de l'horloge d'entrée. Les éléments de retard sont constitués d'une suite d'inverseurs et permettent d'obtenir les délais pour éviter le recouvrement. Pour les interrupteurs des capacités commutées, il faut aussi générer les signaux $\overline{\phi_1}$ et $\overline{\phi_2}$. Ceci est réalisé grâce aux inverseurs INV et INV_bis en fin de chaîne. Il faut que les inverseurs soient conçus afin que le temps de propagation de INV soit le double de celui de INV_bis, afin que les signaux et leurs complémentaires ne soient pas décalés.

Les étapes de fonctionnement du circuit sont gérées par les bits de contrôle TM (Test Mode) et TIC (Test Input Control) (figure 5.3). Au début du fonctionnement, TIC et TM passent à la valeur "1" logique. Pendant cette phase, les premiers bits de l'entrée SDI (bits de programmation) sont chargés dans le registre de contrôle. Ensuite, TIC passe à zéro. Le registre de contrôle n'est plus cadencé et les bits de programmation restent inchangés pendant la suite du fonctionnement. Le train binaire encodant le signal à générer est alors chargé, via l'entrée SDI, dans le registre à

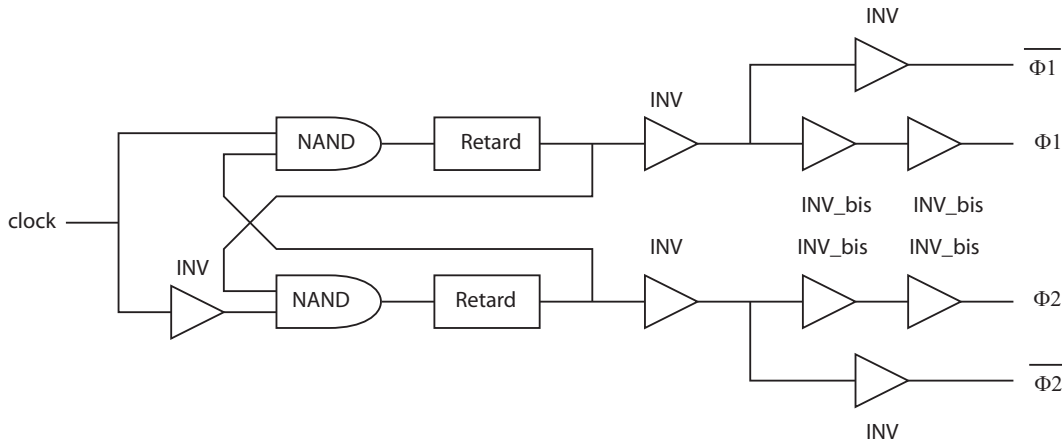


FIG. 5.2 – Schéma du circuit de génération de phases

décalage. Enfin, TM passe à zéro et on entre dans la phase de test proprement dite, où le registre à décalage est rebouclé pour obtenir en sortie le train binaire périodique. Celui-ci est filtré par le banc de filtres pour obtenir le signal analogique voulu. A la fin du test, TM et TIC repassent à “1” et un nouveau vecteur de test peut être chargé.

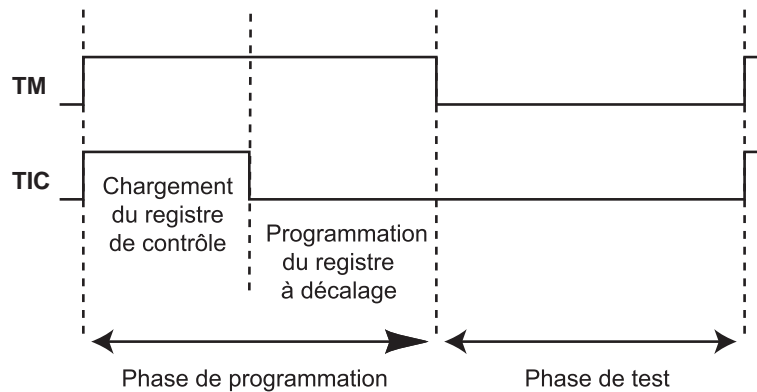


FIG. 5.3 – Les différentes phases de fonctionnement du circuit de génération de signaux de test analogiques

Partie analogique

En ce qui concerne le filtrage, nous avons décidé, pour ce premier démonstrateur, de fabriquer une version du circuit avec un banc de filtres d'ordres 2 à 6. Nous avons expliqué la structure de ces différents filtres dans le chapitre 3. Nous avons vu que ces filtres nécessitent entre un et trois filtres de Rauch et éventuellement un intégrateur, avec différentes valeurs de capacités. Dans la pratique, le démonstrateur comprendra donc un intégrateur (que nous nommerons bloc α) et trois filtres de Rauch du second ordre (blocs β , γ et ω). Parmi ces blocs, deux ont des valeurs de capacités programmables (β et ω). C'est la connexion entre ces différents blocs et la programmation de ces capacités qui va permettre de choisir l'ordre du filtre (tableau 5.1) comme nous l'avons montré dans le chapitre 3, section 3.3. Bien entendu, le choix de la fréquence d'échantillonnage va aussi permettre de choisir la fréquence de coupure de ce filtre comme expliqué dans le chapitre 3.

Bloc	α	β		ω		γ	
Capacité	C_{int}	C_1	C_2	C_1	C_2	C_1	C_2
Ordre 2	-	-	-	-	-	16,88	3,75
Ordre 3	7,95	23,87	2,65	-	-	-	-
Ordre 4	-	31,19	2,03	12,92	4,90	-	-
Ordre 5	7,95	38,63	1,64	14,75	4,29	-	-
Ordre 6	-	46,12	1,37	12,36	5,12	16,88	3,75

TAB. 5.1 – Les blocs utilisés et leurs valeurs de capacités pour obtenir les différents filtres d’ordres 2 à 6

Ces différents blocs contiennent tous le même amplificateur opérationnel, qui est un amplificateur à deux étages classique (une paire différentielle et un étage de gain en source commune, stabilisé par une capacité de Miller). Pour plus de détails sur cet amplificateur, se reporter à l’annexe A. Les performances obtenues sont les suivantes :

- Gain : 36,1 dB
- Marge de phase : 63°
- Fréquence de coupure : 22,4 Mhz
- Fréquence de gain unité : 2,3 GHz

Cet amplificateur s’est révélé consommer un très gros courant dans la structure utilisée et les rails d’alimentation sont très gros et sur quatre niveaux de métal pour pouvoir supporter un tel courant. La conception de ce filtre a été revue pour le second démonstrateur, que nous présenterons plus tard. Le second circuit de ce premier démonstrateur contient uniquement le filtre d’ordre 3 avec les blocs β et α connectés en série. Dans cette version, il n’y a donc pas de programmation du filtre.

Nous avons cherché à simuler l’ensemble de l’architecture avant l’envoi en fabrication, mais nous nous sommes heurtés à un problème de temps de simulation : effectivement, nous cherchons à faire de la simulation mixte. Le problème est donc le suréchantillonnage important utilisé : si la longueur du registre est de 100, chaque période du signal analogique représente 100 périodes d’horloge. Pour pouvoir simuler le fonctionnement du filtre sur un nombre de périodes suffisant, il faudrait donc faire une simulation électrique sur des centaines de périodes d’horloge, ce qui s’est avéré impossible dans la pratique. Nous avons donc simulé la partie analogique en y injectant directement le train binaire obtenu par CAO et obtenu les résultats donnés dans le chapitre 3. En ce qui concerne la partie numérique, nous avons testé ses différents blocs indépendamment et sur une durée limitée. Nous n’avons pas pu simuler l’ensemble de la partie numérique sur un cycle de fonctionnement complet. Afin de pouvoir tout de même tester le filtre au cas où la partie numérique ne fonctionnerait pas, nous avons prévu la possibilité d’injecter un signal directement dans celui-ci, grâce à un multiplexeur analogique. Ceci permettra éventuellement d’injecter un signal analogique ou directement le train binaire.

5.2.2 Layout et finition

Comme nous l'avons déjà signalé, nous avons fait fabriquer sur une puce deux versions du circuit :

- Le circuit complet tel qu'il a été présenté dans les parties précédentes. Ce circuit va permettre de tester tous les ordres de filtre et également de comparer la qualité de la sortie à celle que l'on obtient en utilisant un registre de 1024 bits ([24]).
- Un circuit ne comprenant qu'un filtre d'ordre 3 (celui-ci ayant donné les meilleurs résultats en simulation) et ne contenant pas le registre de 824 bits.

Ces circuits ont été réalisés grâce au CMP, dans une technologie CMOS 0,18 μm de chez ST-Microelectronics. Le layout des parties numériques n'a pas posé de problème particulier. La logique étant relativement simple, elle a été conçue manuellement à partir des portes standard du kit de conception de ST.

Le layout de l'amplificateur utilise de longs transistors qui doivent conduire beaucoup de courant. Nous avons choisi pour ces transistors des structures en peignes interdigités (voir annexe B). Pour les filtres à capacités commutées, nous avons utilisé un réseau de capacités plutôt que d'utiliser de grosses capacités. Un point fondamental est la linéarité du filtre. Nous avons simulé le comportement des capacités disponibles et nous avons opté pour des capacités appelées MIM5 (figure 5.4) plutôt que pour des capacités polysilicium pourtant moins coûteuses en surface. Ces capacités sont constituées de deux électrodes, l'une en niveau de métal 5 et l'autre dans un niveau de métal spécifique qualifié de niveau mim 5. Ces capacités présentent une grande linéarité, mais également une plus grande surface pour une capacité donnée. De plus, elles sont très fragiles et demandent des précautions particulières de conception pour éviter leur claquage : le nombre total de contacts sur le niveau de métal 5 et sur le niveau mim 5 devront être les mêmes dans le cas d'un réseau de capacités. Si cette condition n'est pas respectée, les charges accumulées sur les contacts lors de la fabrication ne seront pas équilibrées et les capacités seront détruites. Ceci est dû à la très faible distance entre les deux niveaux de métal 5. De plus, la fragilité de ces capacités rend les règles de dessin très strictes pour éviter les problèmes d'antennes, également critiques pour ces capacités.

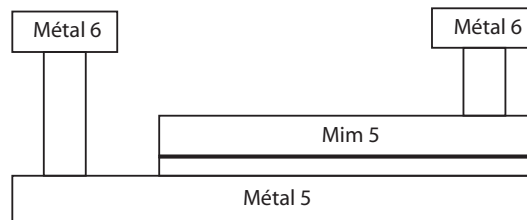


FIG. 5.4 – Schéma de description d'une capacité MIM5

Pour améliorer encore la qualité des filtres, nous avons souhaité connecter les substrats des transistors utilisés dans les interrupteurs de programmation à la masse analogique (à 0,9 V) plutôt qu'à leur tension d'alimentation (ceci afin de réduire les effets body). La figure 5.5 montre la structure envisagée pour réaliser les interrupteurs analogiques : le signal de commande Φ est connecté à la grille d'un transistor NMOS et son complémentaire $\bar{\Phi}$ à celle d'un transistor PMOS. Les drains et les sources de ces deux transistors sont reliés ensemble pour former l'entrée et la sortie de l'interrupteur. Dans cette configuration, on observe un effet body qui change la valeur de la tension de transition si les substrats de tous les transistors ne sont pas connectés à leur source. Or, cette connexion est impossible, puisque les transistors PMOS sont dans un caisson n qui doit être polarisé. On obtient alors la configuration de la figure 5.5(a) qui donne de mauvais résultats (le signal analogique

est dégradé en sortie de l'interrupteur). Nous avons donc choisi la configuration de (b) où les deux substrats des transistors sont reliés ensemble à la masse analogique `gnd_ana`. De même, les transistors utilisés dans les interrupteurs de capacités commutées (considérés comme numériques en raison de leur haute fréquence de commutation) ont leur substrat connecté à la tension intermédiaire numérique (`vint_num`), les simulations ayant donné de meilleurs résultats dans ces conditions. Ceci nécessite de pouvoir avoir un substrat différent pour la puce et pour ces transistors. Ceci a été rendu possible par la possibilité d'isoler une partie du substrat de la puce par un anneau n+ dans les trois directions, comme le montre la figure 5.6. L'isolation verticale du caisson p se fait par l'utilisation d'un niveau dopé n, appelé "niso" dans la technologie employée.

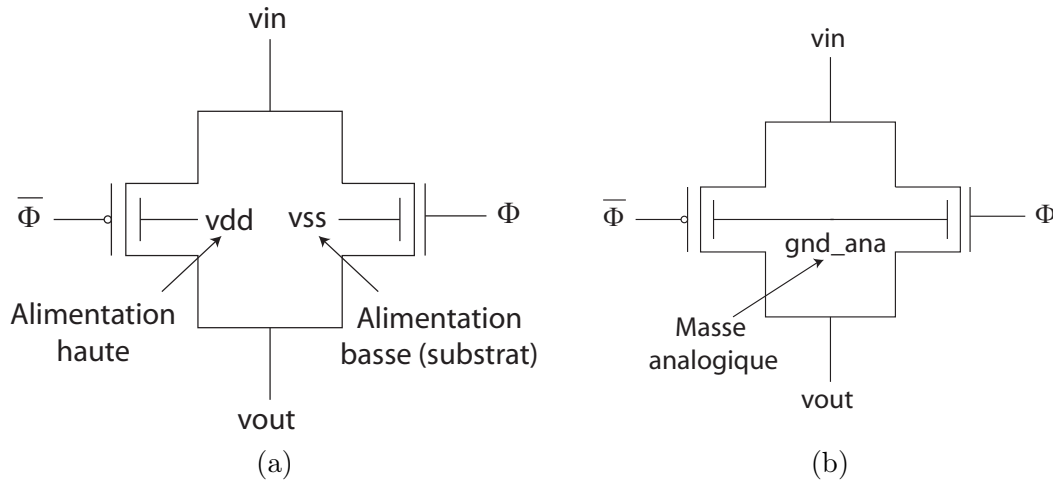


FIG. 5.5 – Description d'un interrupteur analogique : (a) connexion typique des substrats des transistors et (b) la connexion que nous avons utilisée

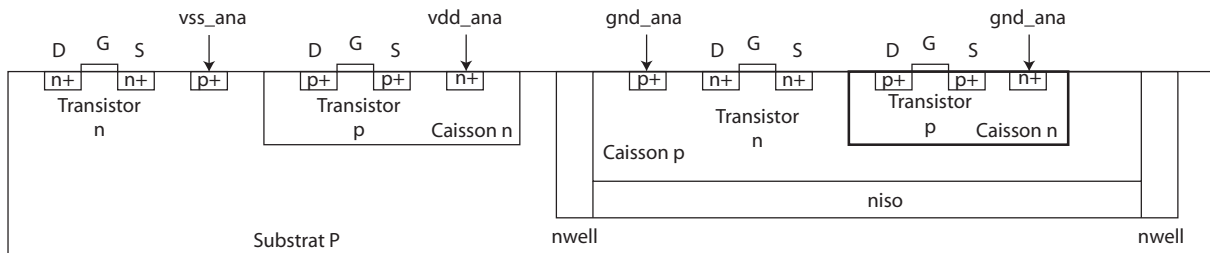


FIG. 5.6 – Utilisation d'un anneau n d'isolation pour avoir un substrat connecté à la masse analogique

Le layout de ce type de circuits mixtes pose un problème : comme nous faisons cohabiter sur le même substrat des parties numériques et des parties analogiques, il faut à tout prix éviter que les parasites de la partie numérique (fonctionnant à haute fréquence) viennent perturber la partie analogique. La première précaution à prendre est d'utiliser des alimentations différentes pour les parties analogiques et numériques. Nous appellerons `vdd_ana` et `vss_ana` les tensions supérieure et inférieure de la partie analogique (respectivement à 1,8 et 0 V dans la technologie que nous utilisons) et `gnd_ana` la masse analogique à 0,9 V. Pour la partie numérique, `vdd_num` et `gnd_num` représentent respectivement les tensions d'alimentation haute et basse numériques. Nous utilisons aussi une tension intermédiaire numérique que nous appellerons `vint_num` (à 0,9 V) dont nous

avons déjà parlé pour les interrupteurs.

Cela n'est cependant pas suffisant : en effet, du bruit peut se propager par le substrat. Il faut donc séparer physiquement les parties analogiques et numériques. Pour cela, nous avons utilisé un anneau d'isolation dopé n, relié par des prises substrat à l'alimentation vdd_num et entourant la partie numérique du circuit. Ceci permet de créer une résistance au passage des perturbations venant des parties numériques et qui sont alors évacuées vers l'alimentation vdd numérique. Pour améliorer encore l'isolation entre les parties analogiques et numériques, nous avons ajouté un anneau de prises substrat p entre la partie numérique et l'anneau n. La figure 5.7 illustre ce procédé et des illustrations sont données dans l'annexe B.

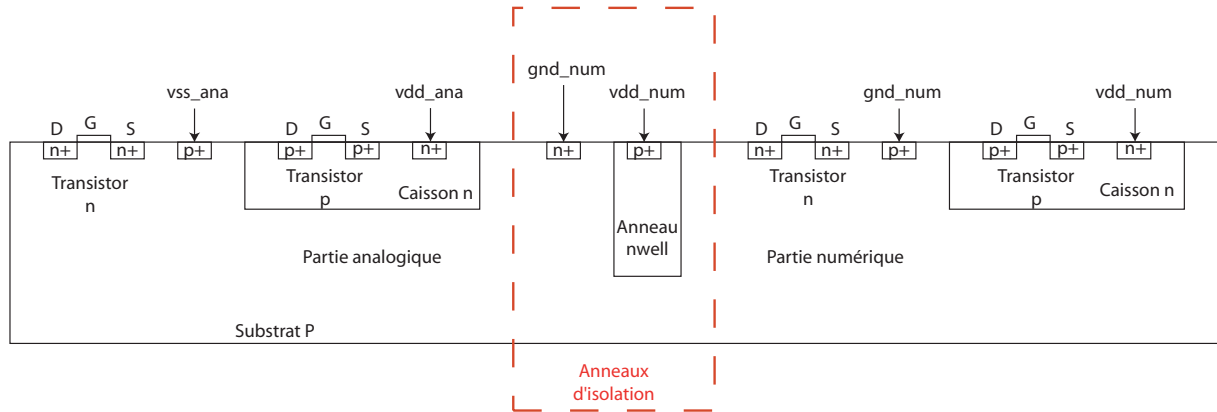


FIG. 5.7 – Isolation des parties analogiques et numériques du démonstrateur

La distinction entre parties analogiques et numériques est ambiguë en ce qui concerne les filtres eux-mêmes : en effet, ces filtres sont à capacités commutées, ils ont donc un aspect échantillonné. Les interrupteurs qui pilotent les capacités fonctionnent à haute vitesse et seront donc considérés comme numériques. Les transistors de programmation (permettant de programmer l'ordre du filtre) et les capacités elles-mêmes seront vus comme analogiques. Ainsi, les filtres seront décomposés en une partie numérique et une partie analogique, comme le montre la figure 5.8. On remarque sur cette figure que les rails d'alimentation sont placés entre les parties sensibles et les parties polluantes afin d'améliorer encore leur isolation.

Une fois le layout terminé et vérifié par des opérations de DRC et de LVS, la dernière étape a consisté à choisir les plots de connexion. Le kit de conception ST-Microelectronics comprend des plots numériques d'entrée et de sortie de plusieurs sortes, adaptés à des situations de fonctionnement différentes (fréquence de fonctionnement, capacité de charge par exemple). Nous nous sommes servis des plots de base pour nos entrées-sorties numériques ("IBUF" pour les entrées numériques et "B2CR" pour les sorties), les signaux que nous utilisons ne présentant pas de contraintes particulières. Les plots analogiques (nommés "ANA") sont bidirectionnels. Enfin, les plots d'alimentation étaient séparés en plots de masse (VSSCO) et plots d'alimentation positive (VDDCO). Dans notre cas, nous avons aussi deux alimentations intermédiaires à 0,9 V. Pour ces alimentations, nous avons choisi des plots de type VDDCO. Il faut aussi noter que la couronne de plots a également besoin d'être alimentée. Ainsi, les plots d'alimentations nommées VDDIOCO et VSSIOCO permettent d'alimenter la couronne aussi bien que le coeur du circuit (les plots VDDCO et VSSCO n'alimentent que le coeur du circuit). Toujours dans le but d'éviter les perturbations dues aux fréquences de fonctionnement de la partie numérique, nous avons utilisé des plots VDDIOCO et VSSIOCO uniquement pour alimenter la couronne de plots. Nous avons utilisé deux plots de chaque type afin de nous assurer que

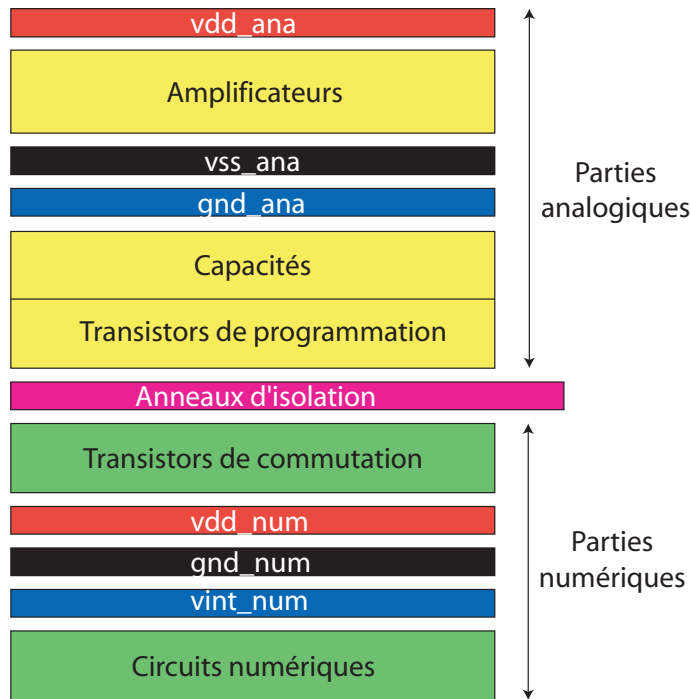


FIG. 5.8 – Schéma illustrant la séparation entre les parties analogiques et numériques dans le banc de filtres

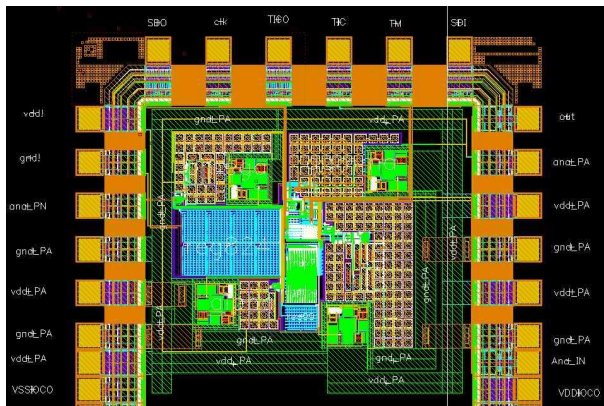
l'alimentation de la couronne entière sera assurée, sans perte de courant. Pour alimenter le coeur des circuits, nous avons utilisé des plots VDDCO et VSSCO en nombre suffisant pour alimenter tout le circuit. Nous utilisons évidemment des plots différents pour les alimentations analogiques et numériques. Le choix de l'emplacement des plots est également très important : nous avons pris garde notamment à éloigner les plots analogiques le plus possible des plots numériques. Les plots VSSCO et VDDCO sont éloignés pour éviter les capacités parasites.

La figure 5.9 présente les layouts des deux circuits du démonstrateur, les photos des parties de la puce fabriquée correspondantes, ainsi qu'une image de l'ensemble de la puce.

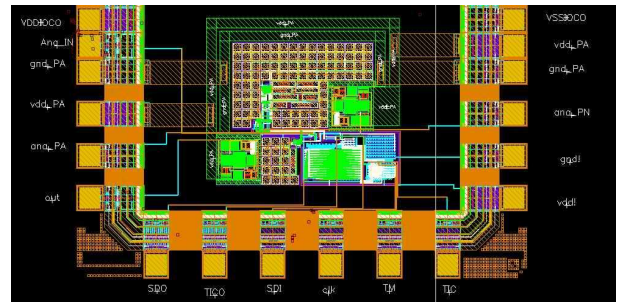
5.2.3 Préparation du test du démonstrateur

Le test du démonstrateur a été effectué au CIME sur un testeur numérique IMS. La photo de la figure 5.10(a) montre le banc de test utilisé pour tester la puce. Pour effectuer ce test nous disposons de divers équipements : un testeur IMS et une station de travail UNIX pilotant ce dernier, un générateur de signaux et un oscilloscope numérique. Les infrastructures de test du CIME nous permettent également de disposer d'autres appareils tels que des analyseurs de spectre et différents types d'oscilloscopes et appareils de mesure.

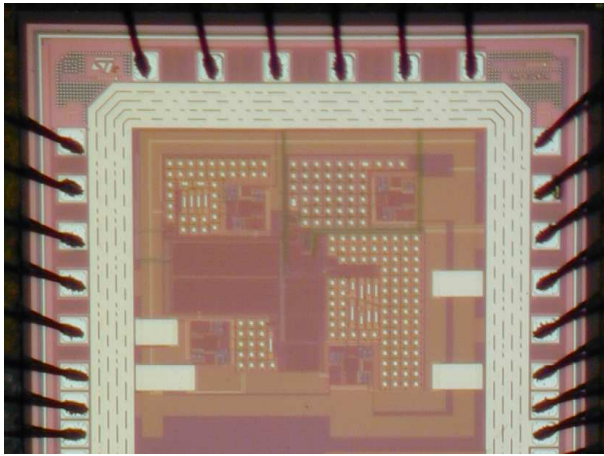
La première étape du test a consisté à connecter la puce sur le testeur. Le testeur utilise des cartes de test comme celle présentée sur la figure 5.10(b), qui permettent de faire l'interface entre le testeur et la puce. Cependant, le type de boîtier utilisé pour ce démonstrateur (PGA100) n'était prévu sur aucune des cartes disponibles. Nous avons donc dû concevoir une carte servant d'adaptateur entre la carte de test et la puce (figure 5.11). Une fois ces questions pratiques résolues, nous avons pu passer au test proprement dit.



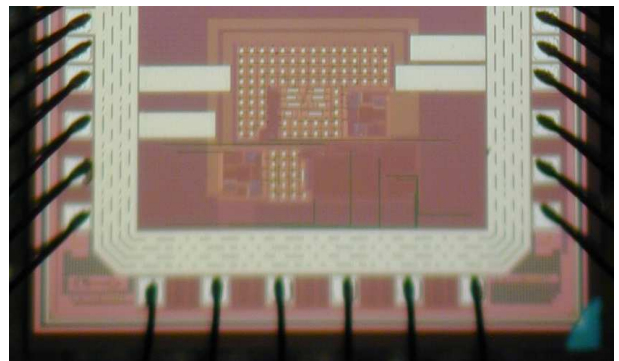
(a)



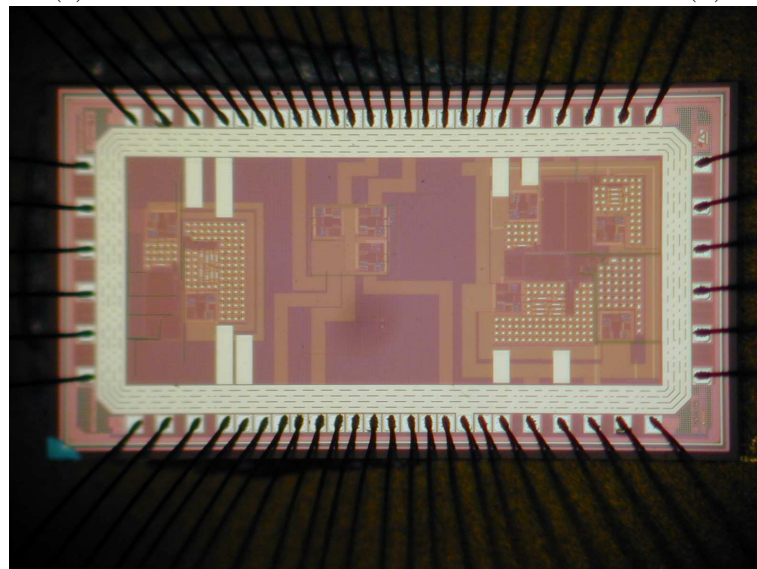
(b)



(c)



(d)



(e)

FIG. 5.9 – Layouts et photos des différentes parties du premier démonstrateur : (a) et (b) montrent respectivement les layouts des deux versions du circuit, (c) et (d) les photos correspondantes et (e) montre une photo de l'ensemble de la puce



FIG. 5.10 – Environnement de test du démonstrateur au CIME : (a) le banc de test et (b) un exemple de carte de test disponible

5.2.4 Test du banc de filtres

La première étape du test a consisté à vérifier le fonctionnement du banc de filtres analogiques. Pour cela, nous avons directement envoyé un signal analogique à chacun des filtres des différents ordres et nous avons pu constater qu'ils marchaient correctement. La figure 5.12 montre les diagrammes de Bode que nous avons tracés pour les 5 ordres de filtres disponibles. Ces filtres sont programmés afin d'avoir une fréquence de coupure à 1 kHz. On voit que tous les filtres se comportent parfaitement.

Nous avons ensuite envoyé le train binaire généré par l'outil de CAO et les résultats se sont révélés très proches. La figure 5.13 montre un signal de fréquence 2,02 kHz généré avec un train binaire de longueur 198 bits. Ce signal est généré avec le filtre d'ordre 3. Les résultats semblent similaires quelques soient les fréquences dans toute la bande passante et pour toutes les longueurs du filtre. On peut faire plusieurs remarques :

- On remarque un bruit aux basses fréquences. En zoomant suffisamment, nous nous sommes aperçus que les fréquences correspondantes étaient des multiples de la fréquence du secteur.
- Le seuil du bruit se trouve toujours environ à 60 dB en-dessous de la valeur du fondamental. Le fait que cette valeur ne change pas en fonction des différentes conditions de test peut signifier qu'il s'agit du bruit interne au testeur qui, rappelons le, est un testeur numérique. Ce bruit peut notamment se retrouver dans les niveaux du train binaire avant filtrage et détériorer la qualité du signal de sortie, comme expliqué dans le chapitre 3.
- Le deuxième harmonique est visible dans le spectre du signal. Il n'est pas tout à fait noyé dans le bruit. Le SFDR est moins bon que les résultats obtenus en simulation, mais l'allure du spectre est très proche.

Nous constatons ainsi la difficulté à tester ce circuit : le but d'une technique de BIST étant en partie d'éviter l'utilisation de matériel externe pour le test afin d'éviter tous les parasites dus aux appareils ou à leurs connexions avec la puce, on ne peut pas réellement tester ses performances en faisant appel à des méthodes de test externes, car on réintroduit justement leurs limitations que l'on essayait d'éviter. Cependant, on pourrait imaginer plusieurs solutions pour avoir un test plus précis, comme par exemple utiliser un testeur analogique ou encore générer les signaux de contrôle avec un FPGA programmé par un PC et qui produirait probablement un bruit inférieur à celui du

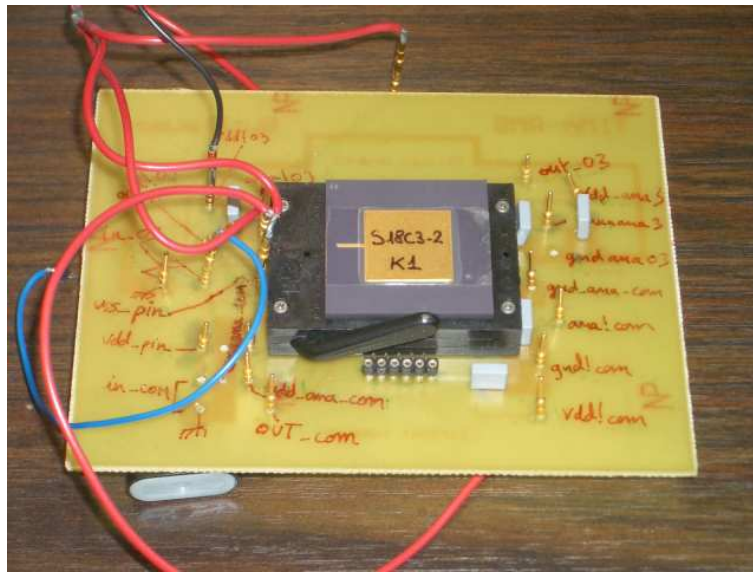


FIG. 5.11 – Carte conçue pour faire l'interface entre la carte de test disponible et la puce

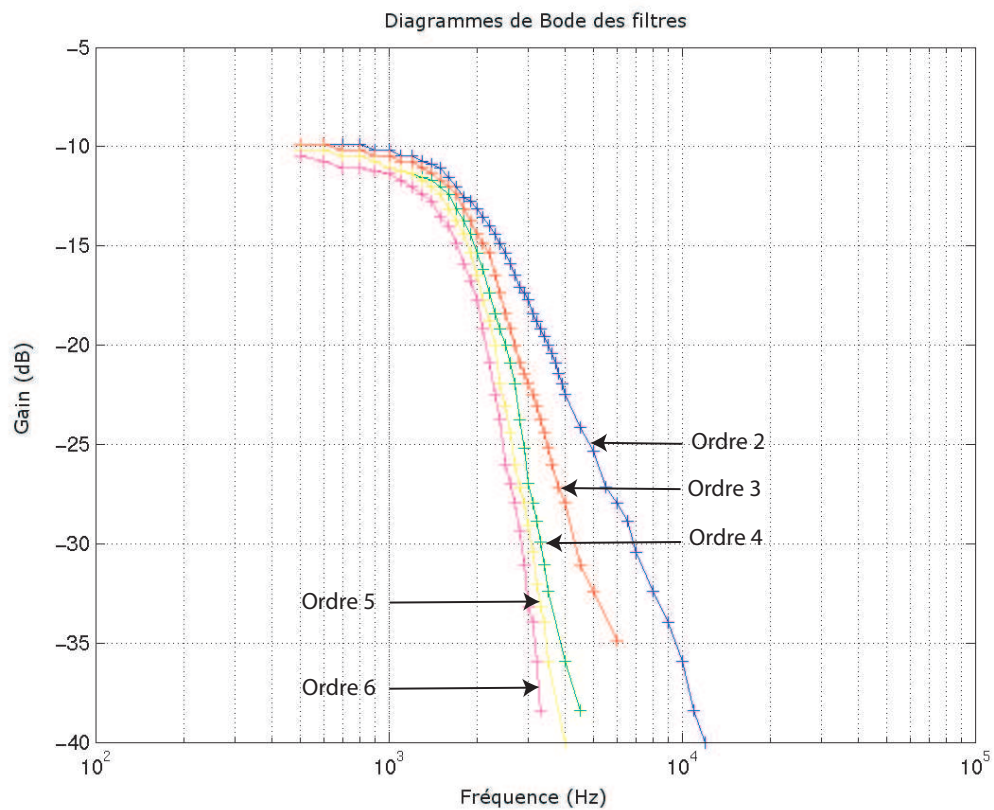


FIG. 5.12 – Diagrammes de Bode des filtres d'ordre 2 à 6 du premier démonstrateur

testeur. Nous n'avons cependant pas eu le temps d'explorer ces solutions.

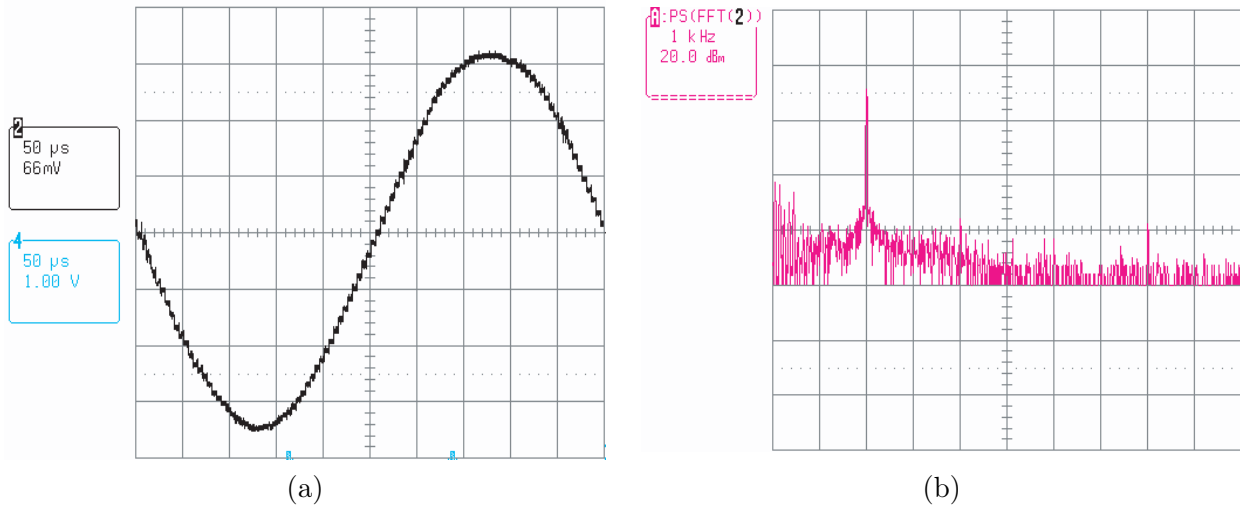


FIG. 5.13 – Test du premier démonstrateur : (a) signal analogique généré, (b) spectre de ce signal

5.2.5 Test du circuit complet

Lorsque nous avons tenté de simuler l'ensemble du circuit, nous avons constaté que la partie numérique ne fonctionne pas entièrement correctement : en effet, le long registre à décalage (200 bits) est cadencé par la même horloge. La logique fournissant cette horloge ne délivre pas assez de courant pour cadencer 100 bascules. Nous aurions dû concevoir un arbre d'horloges à base de buffers suffisamment puissants pour supporter la charge en sortie. Heureusement, tout le reste de la partie numérique fonctionne parfaitement, notamment la partie contrôle qui permet de programmer l'architecture et de faire fonctionner la partie analogique. Nous avons donc pu entrer un signal analogique depuis un générateur de signaux, mais également directement le train binaire. Cette dernière opération a demandé un effort supplémentaire, car le fichier fourni par le logiciel de CAO fournit les bits de programmation ainsi qu'une seule période du train binaire pour l'entrée SDI. Le train binaire est ensuite rebouclé sur la puce. Il a donc fallu modifier les fichiers de test afin qu'ils fournissent directement le train binaire rebouclé. Ceci a été fait grâce à un simple programme en langage C conçu pour automatiser ces opérations. Ce programme a ensuite été développé pour effectuer diverses modifications sur le fichier de test en fonction des situations rencontrées.

5.3 Second démonstrateur : architecture de BIST complète

5.3.1 Schéma et résultats de simulation

Au cours de la seconde année, nous avons conçu et réalisé l'architecture complète de la technique. La figure 5.14 montre cette architecture.

Partie numérique

En haut de cette figure, on retrouve la partie de génération de signal que nous avons déjà décrite précédemment et qui fonctionne exactement de la même façon. Nous avons cependant ajouté un

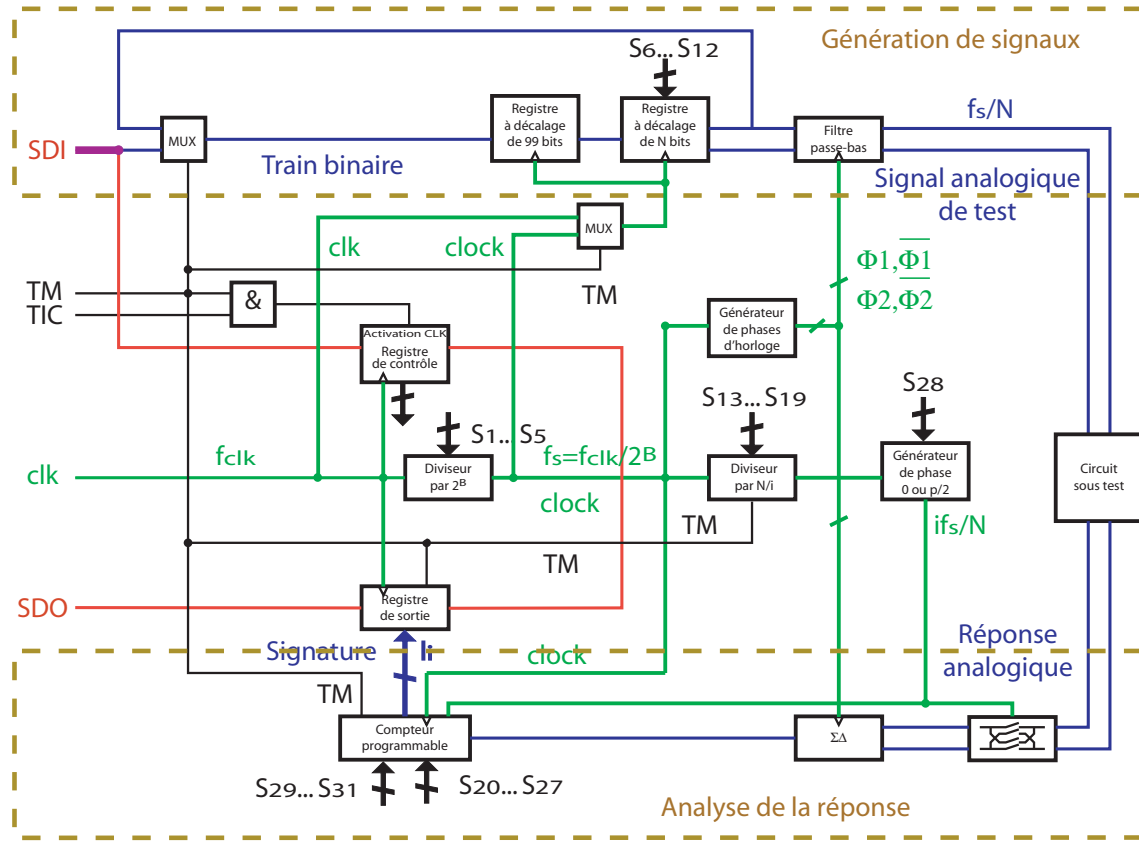


FIG. 5.14 – Schéma du circuit de BIST complet

arbre d'horloge pour s'assurer de pouvoir supporter la charge de toutes les bascules. Dans cette version, la partie d'analyse de la signature est présente (représentée en bas de la figure 5.14). Le circuit sous test est constitué du filtre passe-bas d'ordre 3. Il est également possible de sortir le signal de test pour l'envoyer vers un circuit sous test externe et d'en réinjecter la réponse vers le circuit d'analyse de signature. Cette partie d'analyse contient le multiplexeur analogique permettant de réaliser la modulation par le signal carré, le modulateur $\Sigma\Delta$ d'ordre 1 et le compteur. La partie de génération d'horloges a également été modifiée, de manière à pouvoir générer les signaux carrés de modulation. Outre le diviseur de fréquence par une puissance de 2, utilisé pour obtenir la fréquence d'échantillonnage à partir de la fréquence d'horloge, on retrouve ainsi un diviseur de fréquence programmable qui va permettre de générer les signaux carrés de modulation par division de la fréquence f_s d'échantillonnage (voir annexe A) et un générateur de phases qui va permettre de créer les deux signaux carrés déphasés d'une phase $\frac{\pi}{2}$.

Dans cette architecture, les bits S_1 à S_5 permettent toujours de programmer le facteur de division 2^B , les bits S_6 à S_{12} la longueur N du registre à décalage. Le fonctionnement de la partie génération de signaux fonctionne de la même façon que précédemment. Les bits S_{13} à S_{19} programment le facteur de division correspondant aux signaux carrés de modulation, les bits S_{20} à S_{27} le nombre de périodes de comptage, le bit S_{28} la phase du signal carré (0 ou $\frac{\pi}{2}$) et les bits S_{29} à S_{31} l'harmonique à mesurer.

Attardons nous sur le fonctionnement du compteur : ce compteur est programmable, dans le sens où l'on peut programmer le nombre de périodes de comptage. Pour pouvoir compter le nombre de périodes du signal qui ont été traitées, on utilise un signal carré à la même fréquence que celle

du train binaire (généré à partir des signaux carrés de modulation) : pour compter le nombre de périodes du signal analogique, on compte le nombre de périodes de ce signal carré. Le composant que nous appelons “compteur” sur la figure 5.14 est donc composé en réalité de deux compteurs :

- Le compteur proprement dit, qui va compter le nombre de “1” du train binaire. Ce compteur est incrémenté sur un front descendant de l’horloge *clock* si la valeur de SDI vaut “1”.
- Le compteur de périodes, qui est en fait un décompteur. La valeur initiale (le nombre de périodes à compter) est chargée dans le décompteur et cette valeur va être décrétementée à chaque période du train binaire détectée à l’aide du signal carré.

Une fois le nombre de périodes atteint, le compteur émet un signal annonçant la fin du comptage, qui va permettre de rendre le résultat disponible. Lorsque TM repasse à “1” à la fin du test, le résultat du comptage, c’est-à-dire la signature, est extrait en série depuis le registre de sortie par SDO. Pour plus de détails sur la conception de ce compteur, se reporter à l’annexe A.

Partie analogique

Dans cette version, nous avons choisi une structure entièrement différentielle pour la partie analogique. Ce choix a deux avantages :

- D’abord, la structure différentielle pour le filtre passe-bas permet de limiter les effets d’injections de charges et de supprimer les harmoniques pairs de la sortie. Le bruit à basses fréquences doit également être éliminé.
- De plus, la structure différentielle de la chaîne analogique permet de rendre la modulation par le signal carré beaucoup plus facile.

L’amplificateur a également été redessiné pour diminuer la grande consommation de courant du démonstrateur précédent. La nouvelle structure du filtre est une structure de type cascode “rebouclée” afin d’assurer la stabilité du filtre quelle que soit la capacité de charge en sortie. Le contrôle de tension en mode commun permet d’assurer une linéarité en sortie sur la plus grande bande passante possible. L’amplificateur ainsi conçu a les performances suivantes :

- Gain : 80 dB
- Bande passante : 150 MHz
- Gamme d’amplitude en sortie : -600 mV à 600 mV
- Consommation de courant : 1,5 mA

Pour plus d’informations sur la conception de cet amplificateur, se reporter à [49], [7], [94] et à l’annexe A.

La conception du filtre d’ordre 3 est la même que celle utilisée dans le démonstrateur précédent, mais avec le nouvel amplificateur et une structure différentielle. Le modulateur $\Sigma\Delta$ est également conçu suivant le même genre de principe avec des capacités commutées. Nous ne rentrerons pas ici dans les détails de la conception.

Cette fois encore, nous nous sommes heurtés à un problème de temps de simulation en essayant de simuler l’ensemble du circuit de manière électrique. La mémoire des machines utilisées n’était même pas suffisante pour mener à terme la simulation. Cependant, nous avons fait un effort pour avoir des simulations le plus complètes possible :

- La partie numérique étant cette fois beaucoup plus complexe, nous avons simulé entièrement son comportement en Verilog, avec l’outil VerilogXL intégré sous Cadence en utilisant les modèles Verilog des portes de base fournis dans le kit de conception ST. Ceci permet de vérifier la logique uniquement.

- La partie analogique a été simulée avec succès comme précédemment en envoyant un train binaire en entrée du filtre analogique et en observant la sortie du modulateur $\Sigma\Delta$.
- Nous avons essayé d’effectuer des simulations mixtes (en simulant la partie numérique de manière comportementale et la partie analogique au niveau électrique). Cependant, nous nous sommes aperçus qu’il était difficile de séparer ces deux parties dans notre circuit, car les filtres étant échantillonnés et programmés, ils contiennent des portes logiques considérées comme numériques, mais traitées comme analogiques par nous-mêmes (comme nous l’avons déjà expliqué dans la section traitant de la séparation entre parties analogiques et numériques dans le premier démonstrateur). Nous n’avons donc pas plus insisté dans cette voie, d’autant plus que cette simulation mixte n’aurait pas apporté beaucoup d’informations supplémentaires : en effet, la partie numérique est toujours simulée au niveau comportemental et le résultat est le même que si nous avons envoyé le train binaire généré par la simulation Verilog de la partie numérique à la partie analogique. Cette simulation aurait été presque redondante par rapport aux deux précédentes.
- Par contre, pour éviter les problèmes de type charge de sortie que nous avons rencontrés pour le premier démonstrateur, nous avons décidé d’essayer de simuler l’ensemble de la partie numérique de manière électrique au moins sur la phase de programmation et sur une période du train binaire pour vérifier si cette phase d’initialisation se passe bien. Pour cela, nous avons demandé à utiliser la station de travail du CMP (beaucoup plus puissante) et nous avons réussi à terminer cette simulation (qui a duré environ deux jours). Les résultats étaient tout-à-fait ceux attendus, mais nous n’avons volontairement pas ajouté la partie analogique pour ne pas rallonger les simulations.
- Enfin, l’idéal aurait été, après l’extraction du layout, de pouvoir faire une simulation mixte, mais en ajoutant les délais de propagation après placement-routage pour la partie numérique. Mais nous n’avons pas eu le temps d’aboutir avant la date d’envoi en fabrication. Cela est dommage, car les simulations auraient alors été les plus réalistes que nous aurions pu obtenir.

5.3.2 Layout et finition

Pour ce second démonstrateur, nous avons à nouveau fait fabriquer deux circuits : l’un contenant uniquement la partie numérique du BIST afin de pouvoir la tester plus facilement, l’autre l’ensemble de l’architecture. La technique de layout pour ce nouveau circuit est sensiblement la même que pour le premier démonstrateur, notamment en ce qui concerne l’isolation entre les parties analogiques et numériques. Nous ne détaillerons donc pas à nouveau le travail effectué. La structure des parties analogiques étant cette fois différentielle, nous avons pris garde à avoir la structure la plus symétrique possible. Les plots utilisés sont les mêmes. La figure 5.15 montre ce démonstrateur : (a) représente le layout du circuit complet, (b) le layout de la partie purement numérique, (c) et (d) les photos de ces deux circuits sur silicium et (e) montre l’ensemble de la puce.

Nous pouvons tout de suite donner les surfaces des différentes parties composant ce démonstrateur (tableau 5.2). Parmi elles, les blocs analogiques peuvent éventuellement être présents dans le circuit sous test et être réutilisés pour le test (nous rappelons que tous les composants du BIST doivent être testés de toute façon). La surface réellement dédiée au BIST est donc celle qui correspond aux parties numériques, particulièrement le registre à décalage et le multiplexeur correspondant.

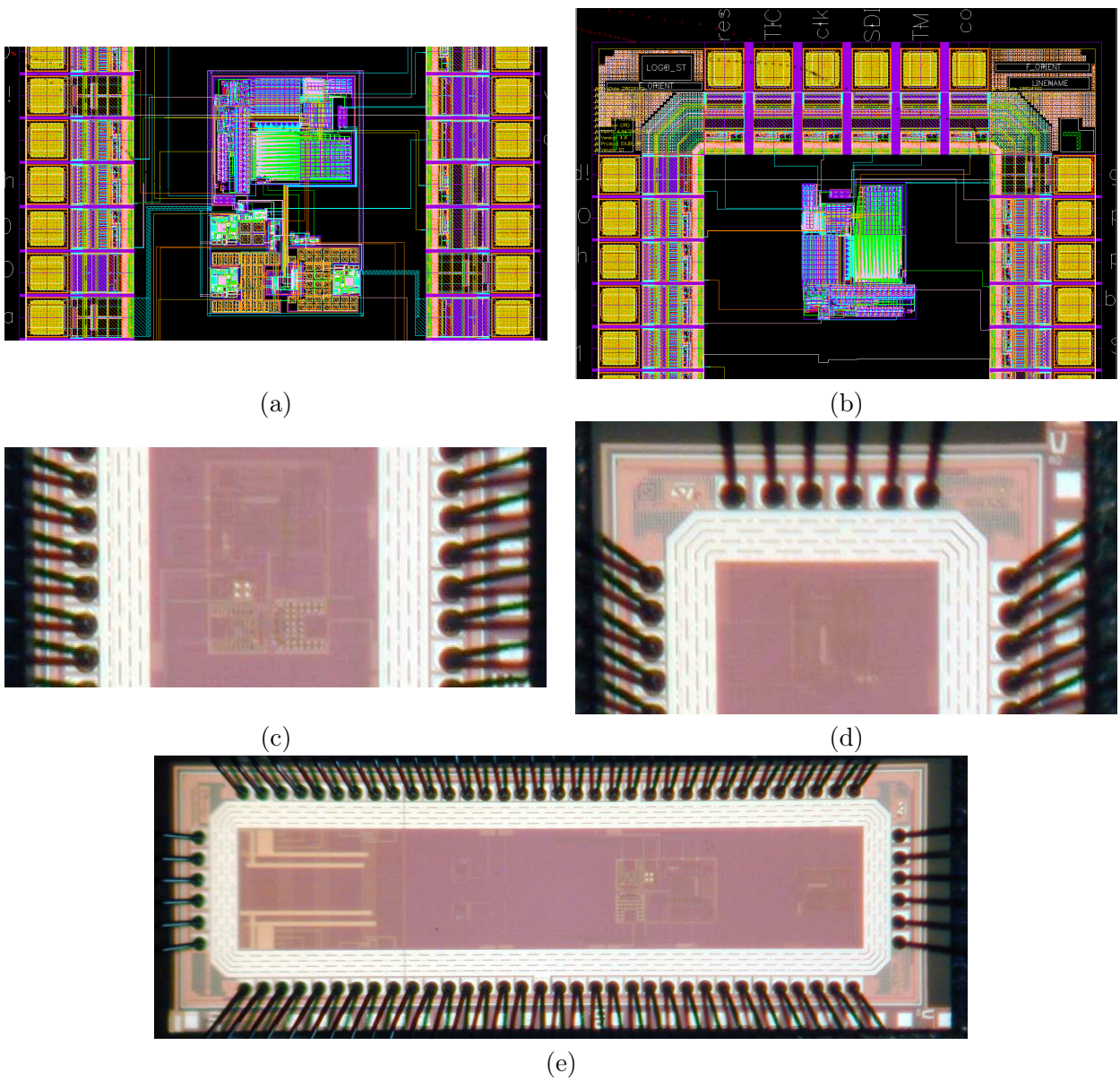


FIG. 5.15 – Layouts et photos des différentes parties du second démonstrateur : (a) et (b) montrent respectivement les layouts des deux versions du circuit, (c) et (d) les photos correspondantes et (e) montre une photo de l'ensemble de la puce

	Composant	Surface (μm^2)
Partie numérique	Registre et multiplexeur	26000
	Compteur	16100
	Autres composants	20000
Partie analogique	Filtre analogique	42000
	Modulateur $\Sigma\Delta$	9800

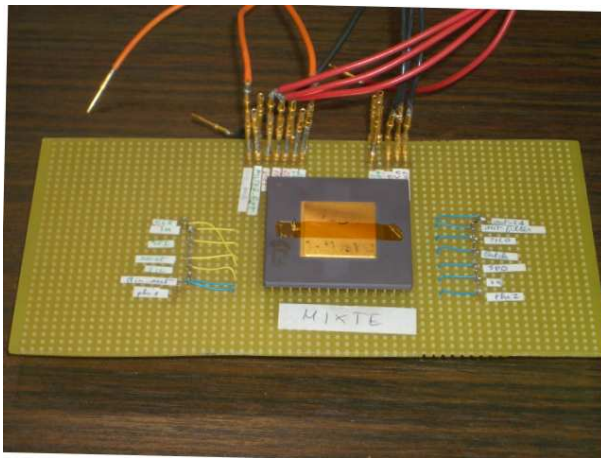
TAB. 5.2 – Surfaces occupées sur le démonstrateur par les principaux blocs

5.3.3 Préparation du test du démonstrateur

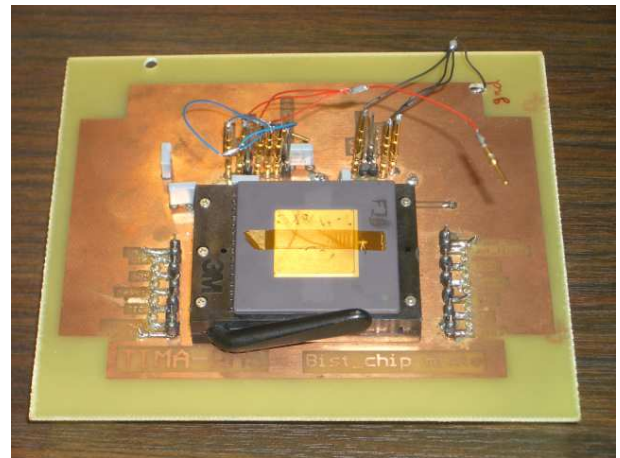
Afin de pouvoir tester ce nouveau circuit, nous avons dû à nouveau concevoir une carte supplémentaire destinée à adapter le boîtier utilisé (PGA144) à la carte de test. Nous avons tout d'abord fabriqué une carte de qualité grossière (figure 5.16(a)) afin de vérifier si le circuit semblait fonctionner correctement et s'il était exploitable. Une fois les premiers tests réalisés, nous avons conçu un PCB à l'ENSERG (figure 5.16(b)).

5.3.4 Test de la partie numérique

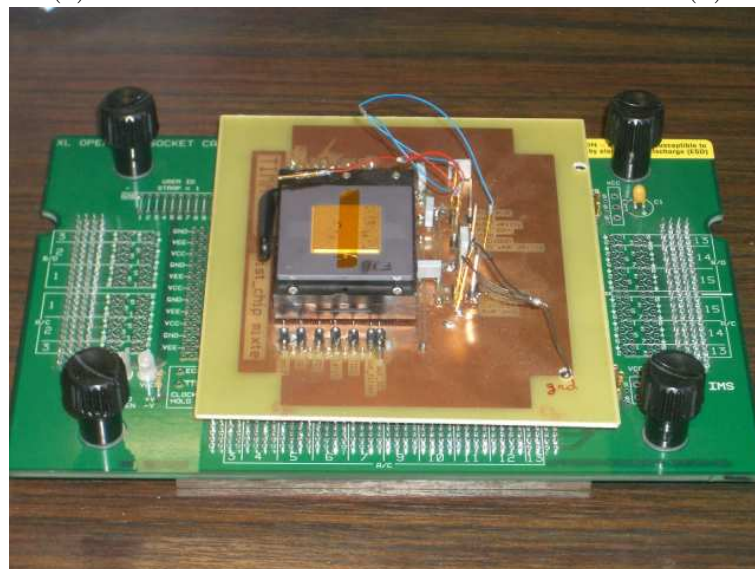
Comme nous l'avons déjà vu, la puce contenait une version du circuit constituée uniquement des éléments numériques. Dans ce circuit, la sortie du registre à décalage est directement connectée à la



(a)



(b)



(c)

FIG. 5.16 – Les cartes conçues pour faire l'interface entre les cartes de test disponibles et la puce pour le second démonstrateur : (a) la version pour les tests préliminaires, (b) la version pour les tests plus précis et (c) cette dernière version ajustée sur la carte du testeur

partie compteur de la partie d'analyse. Ce circuit contient également les éléments de génération des signaux carrés de modulation. Le test de cette partie a révélé un fonctionnement parfait : les bits de contrôle et le train binaire se chargent parfaitement, le train binaire se reboucle grâce à l'ajout d'un arbre d'horloge adapté pour supporter la charge de toutes les bascules du registre à décalage. Afin de vérifier le fonctionnement de cette partie numérique, nous avons entré un train binaire simple constitué d'un certain nombre (connu) de "1" et nous avons vérifié que le compteur donnait le bon nombre de "1". Bien sûr, nous avons fait un grand nombre de tests dans des conditions différentes (nombre de "1", nombre de périodes de comptage, longueur du registre) définies par la partie contrôle du circuit elle-même, vérifiant ainsi son fonctionnement. A chaque test, nous avons aussi choisi quel harmonique d'un éventuel signal analogique nous voulions analyser et nous avons vérifié que les signaux carrés de modulation étaient corrects. Tous ces tests se sont révélés concluants.

5.3.5 Réparation au FIB

Le premier constat que nous avons fait en commençant les tests de la partie analogique est qu'une erreur de conception avait été commise : il manquait un plot sur une des sorties différentielles du filtre, interdisant ainsi de vérifier la qualité du signal différentiel. Heureusement, nous avons remarqué que la configuration du circuit rendait une réparation possible : d'abord, un plot analogique était disponible, correspondant à une sortie du modulateur $\Sigma\Delta$. De plus, la sortie non-observable du filtre ainsi que l'accès au plot en question étaient câblés dans un haut niveau de métal, rendant leur accès possible par une méthode de FIB (Focused Ion BEAM) qui permet, en envoyant un faisceau d'ions à très forte énergie, de graver l'oxyde et les métaux pour atteindre des niveaux plus profonds d'une puce. Cette technique permet donc de couper des connexions, mais également d'en créer : il est en effet possible de creuser des vias jusqu'au niveau de métal des points que l'on veut connecter, puis de faire un dépôt de tungstène au dessus de la couche de passivation de la puce pour faire la connexion. Cette opération est similaire à celle qui consiste à créer les connexions d'un niveau de métal de la technologie. Nous avons donc décidé d'utiliser cette technique² pour déconnecter, sur une puce, une sortie du modulateur $\Sigma\Delta$ d'un plot analogique et de connecter la sortie du filtre à ce plot. L'annexe C.1 donne des précisions sur cette intervention.

5.3.6 Test du circuit de génération

En attendant la réparation de la puce, nous avons observé la sortie simple du filtre pour la comparer aux résultats de simulation. La figure 5.17 montre la comparaison entre le signal généré, de fréquence 833 Hz, avec un registre de 120 bits et le signal obtenu par simulation dans les mêmes conditions. On constate la similitude, avec une forte distorsion correspondant au second harmonique.

Une fois la réparation effectuée (avec succès), nous avons observé la sortie différentielle du filtre. Nous avons constaté un bon fonctionnement de celui-ci. La figure 5.18 donne la représentation d'un signal sinusoïdal. On voit la comparaison entre les deux sorties (qui sont très proches, prouvant la qualité de la réparation) et le signal différentiel. La figure 5.19 montre la comparaison entre le signal généré et le signal obtenu en simulation. La première observation que nous pouvons faire est que le bruit à basse fréquence a entièrement disparu. Les harmoniques sont également très atténués, déjà noyés dans le bruit, ce qui n'était pas le cas dans le premier démonstrateur (revoir la figure 5.13). En revanche, le seuil de bruit est toujours le même (environ 60 dB en dessous de la valeur du fondamental), ce qui tendrait à confirmer qu'il vient de l'environnement de test et qu'il sera très difficile d'évaluer la qualité du signal avec plus de précision dans ces conditions.

2. Réparation effectuée par l'entreprise "SERMA Technologies"

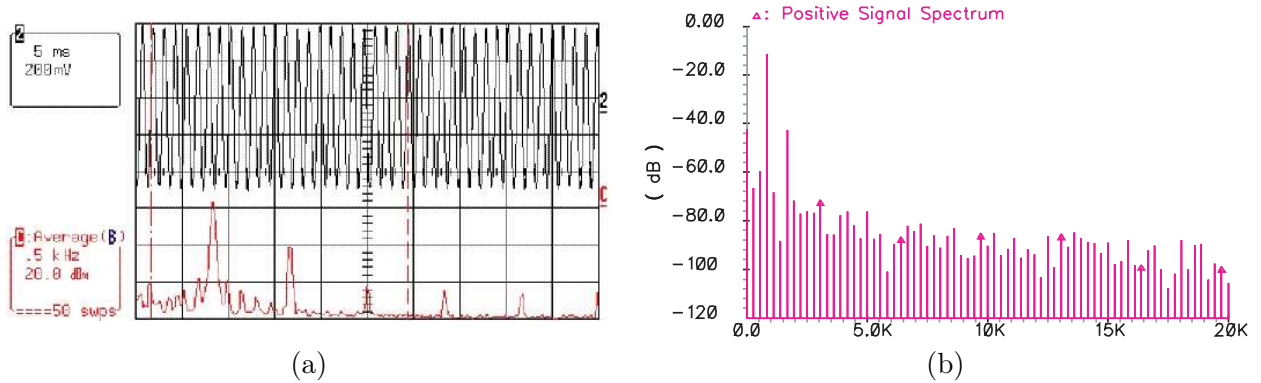


FIG. 5.17 – Comparaison entre le spectre d'une sortie simple et celui du même signal obtenu par simulation

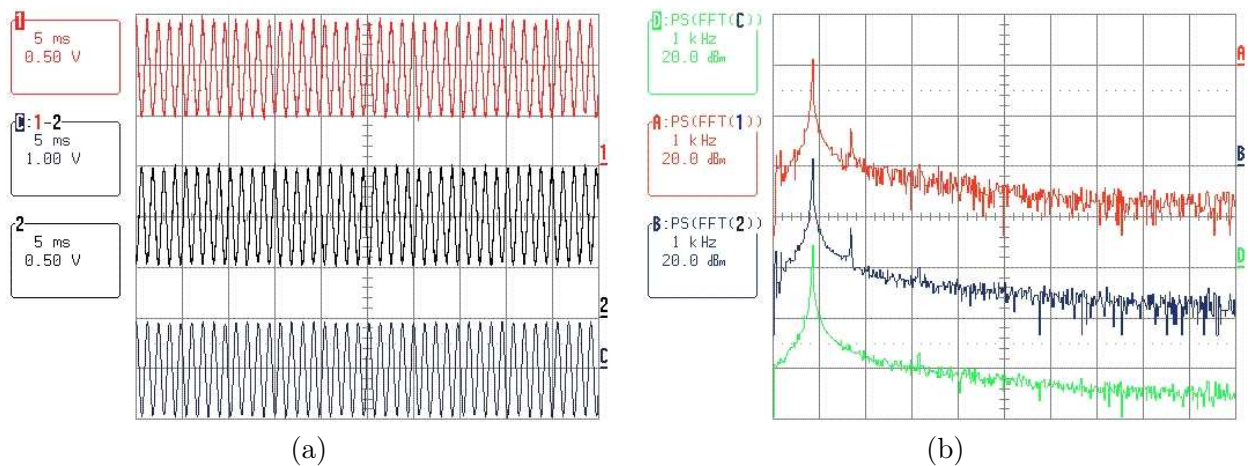


FIG. 5.18 – Comparaison entre les deux sorties du filtre prises indépendamment et la sortie différentielle, dans le second démonstrateur

Afin de vérifier les résultats obtenus en simulation, nous avons tracé l'évolution du SFDR en fonction de la longueur du registre, pour les valeurs paires et impaires de celui-ci (figure 5.20). Pour des question de temps de test, nous n'avons pas pu tracer cette courbe avec un grand nombre de points. On peut cependant faire deux remarques :

- L'allure des courbes est globalement bonne, avec un SFDR qui augmente avec la longueur du registre et des valeurs sensiblement moins bonnes pour les valeurs impaires que pour les valeurs paires.
- Les résultats sont globalement moins bons que ceux obtenus en simulation. Ceci a déjà été remarqué dans le spectre du signal généré et peut être dû aux bruits de l'environnement de test.

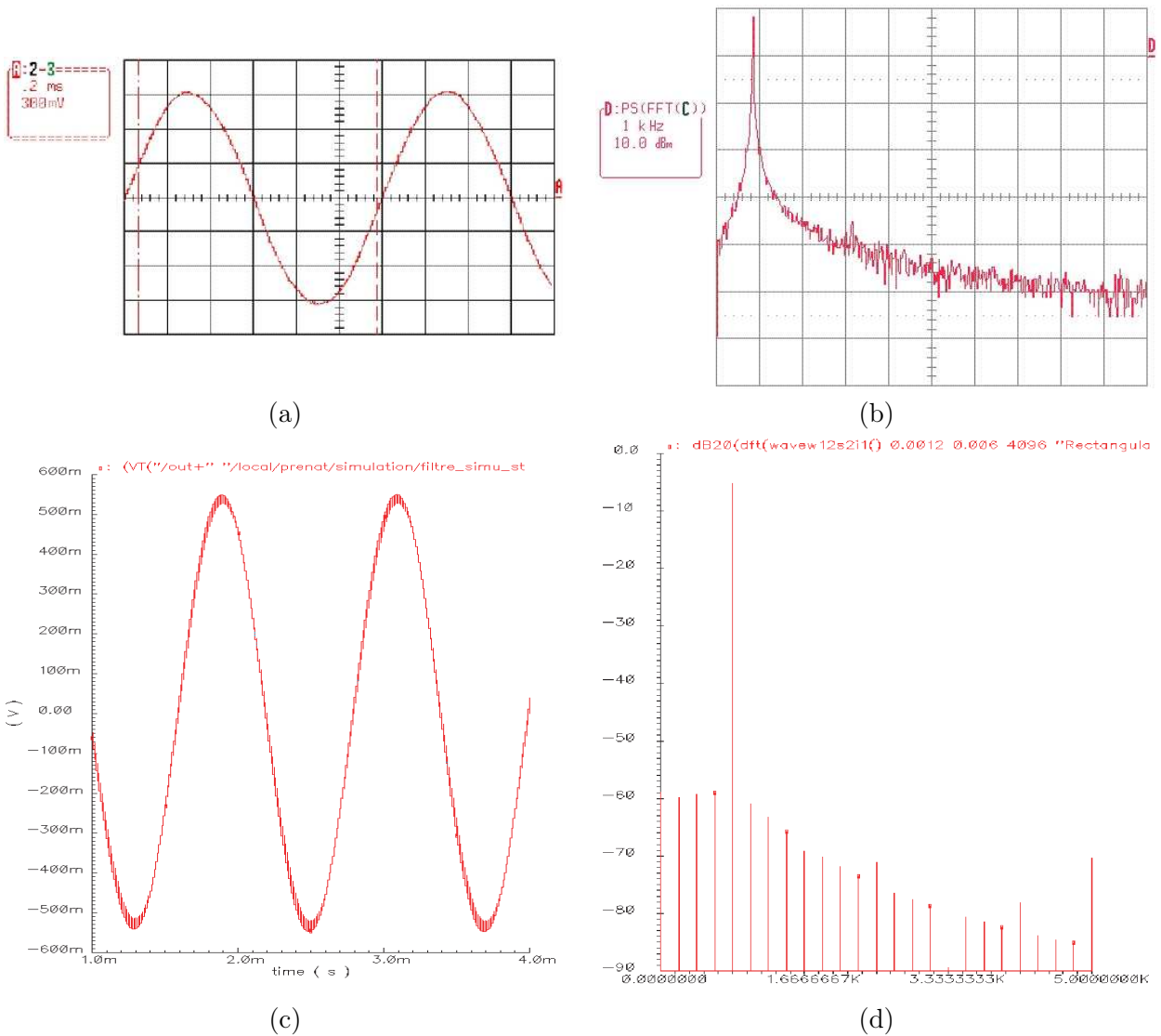


FIG. 5.19 – Test du second démonstrateur: (a) signal analogique généré, (b) spectre de ce signal et (c) et (d) les résultats de simulation correspondants

5.3.7 Test du circuit d'analyse de signature

La partie numérique du circuit d'analyse marche parfaitement, nous l'avons déjà vérifié lors du test de la version entièrement numérique du circuit. Malheureusement, le modulateur $\Sigma\Delta$ ne fonctionne pas bien: il fonctionne de manière très approximative et pour une faible gamme de fréquences d'horloge. Il ne fonctionne pas du tout pour les hautes fréquences, mais également pour les faibles fréquences, ce qui est plus surprenant. A ce jour, nous n'avons pas été capables de comprendre les raisons de ce non-fonctionnement qui n'avait pas été détecté en simulation. Nous avons cependant remarqué que les phases pour les capacités sont à la limite du recouvrement (sans doute n'avons nous pas pris assez de marge dans les délais). Ceci peut être une raison de ce mauvais fonctionnement.

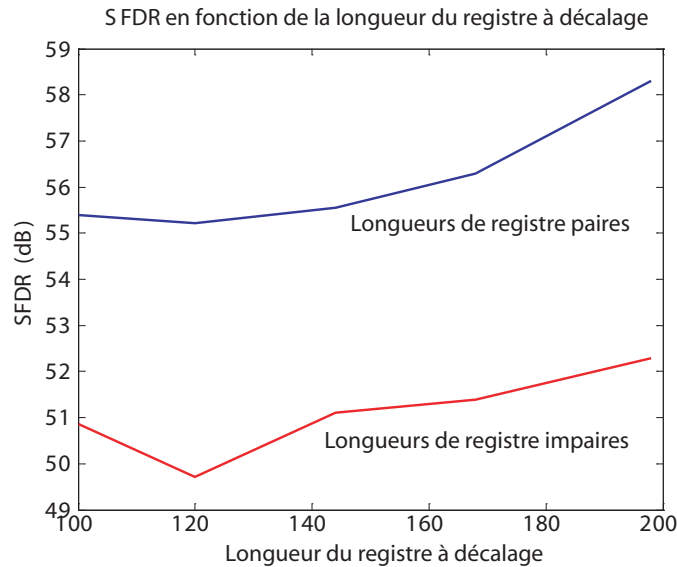


FIG. 5.20 – *SFDR en fonction de la longueur du registre : résultats du test pour le second démonstrateur*

5.3.8 Test de l'architecture complète

La partie analyse étant handicapée par le mauvais fonctionnement du modulateur $\Sigma\Delta$, nous n'avons malheureusement pas pu vérifier le fonctionnement de l'ensemble de l'architecture.

5.4 Conclusion

Deux démonstrateurs ont été fabriqués pour valider cette technique. Plusieurs remarques peuvent être faites :

- Le principe de génération a été validé. La technique fonctionne et la qualité du signal est satisfaisante, surtout dans le cas d'une structure différentielle. La différence de qualité entre le signal réel et les résultats de la simulation sont probablement imputables aux conditions de test :
 - Le testeur numérique est très bruyant.
 - Le fait d'utiliser un testeur externe pour vérifier la qualité des signaux est contraire au principe même du BIST qui présente l'intérêt d'éviter le bruit dû aux connexions entre la puce et les appareils de test.

Afin d'améliorer la qualité des signaux générés, on pourrait envisager d'avoir recours à un CAN 1 bit avant le filtre passe-bas afin de générer avec précision les niveaux du train binaire malgré le bruit de l'environnement.

- La partie d'analyse de signature n'a pu être testée correctement à cause d'un mauvais fonctionnement du modulateur $\Sigma\Delta$ qui n'a pas pu être détecté en simulation. Ce résultat manque, d'autant plus que toute la logique de l'analyse semble fonctionner parfaitement. On peut cependant supposer que les problèmes de bruit auraient également limité la précision des mesures et il aurait été difficile de connaître la part de l'environnement dans cette imprécision. Il aurait

pourtant été intéressant de faire des mesures même approximatives d'harmoniques de signaux connus pour au moins valider le principe de la méthode. Enfin, pour mesurer les paramètres définis dans le chapitre 4, il aurait fallu fabriquer un très grand nombre d'instances, ce qui est très difficile.

- En ce qui concerne la surface de silicium ajoutée par le BIST, on peut remarquer que la plus grande partie est composée :
 - des filtres et du modulateur $\Sigma\Delta$ pour la partie analogique,
 - du long registre à décalage et du multiplexeur correspondant pour la partie numérique.

Dans un SoC, on peut espérer réutiliser les ressources analogiques déjà existantes pour le BIST. Il ne reste donc que la surface du registre et du multiplexeur, qui est raisonnable par rapport à une technique de BIST plus complexe où il faudrait faire un traitement numérique de la réponse.

Plusieurs travaux seraient donc envisageables pour améliorer l'appréciation de la technique. La première serait de comprendre la cause du non fonctionnement du modulateur $\Sigma\Delta$ et éventuellement de refaire un démonstrateur, en essayant de mener à bien des simulations plus réalistes. Ce démonstrateur pourrait ensuite être testé avec un testeur analogique, ou au moins un FPGA pour générer les signaux des test en évitant le bruit du testeur.

Chapitre 6

Compatibilité de l'architecture avec les normes de test

6.1 Introduction

Comme nous l'avons déjà signalé en introduction, l'augmentation de la complexité des circuits intégrés limite l'accès aux différents noeuds de ceux-ci, rendant le test beaucoup plus difficile. Les techniques de scan-path ont pour objet de faciliter l'accès au plus grand nombre possible de noeuds du circuit, en nécessitant un nombre limité de plots. Le principe est d'entrer toutes les données de test en série dans un registre à décalage (cette opération ne nécessitant qu'un seul plot d'entrée) et une fois le registre chargé, de distribuer les données en parallèle dans tout le circuit aux noeuds concernés. La réponse est alors chargée en parallèle dans ce même registre et extraite en série de la puce. Cette technique ne nécessite donc que deux plots d'entrée-sortie (en plus des plots servant aux signaux de contrôle de l'architecture de test) au lieu de nécessiter un plot par noeud.

Très vite, la création d'un standard pour normaliser ces techniques s'est avérée nécessaire. Une norme est apparue en 1990 sous le nom officiel IEEE 1149.1. Elle est également connue sous les noms de "boundary-scan" ou "JTAG"¹. Cette norme, qui concerne le test de circuits numériques est une extension des techniques de scan-path au niveau carte. Cette norme permet d'uniformiser des techniques de test de composants au niveau carte : chaque puce est compatible boundary-scan. Le standard permet alors d'effectuer des tests d'interconnexion entre les composants, des tests des composants eux-mêmes ou encore de piloter un BIST inclus dans un composant. Cette norme a été plusieurs fois mise à jour. Une version dédiée au test de circuits analogiques et mixtes est apparue plus récemment sous l'appellation IEEE 1149.4, basée sur le même principe.

Nous allons maintenant décrire brièvement ces deux normes, avant de voir comment rendre notre architecture compatible avec celles-ci. La description détaillée des standards est un gros travail qui ne sera bien sûr pas réalisé ici. Pour plus d'informations, on pourra se référer à plusieurs ouvrages dont [79] ou [1] et consulter le site internet des groupes de travail correspondant à ces standards <http://grouper.ieee.org/groups/1149/>.

1. Joint Test Action Group

6.2 Les standards de test numérique (IEEE 1149.1)

L'architecture générique d'un circuit intégré compatible avec le standard IEEE 1149.1 est montrée sur la figure 6.1.

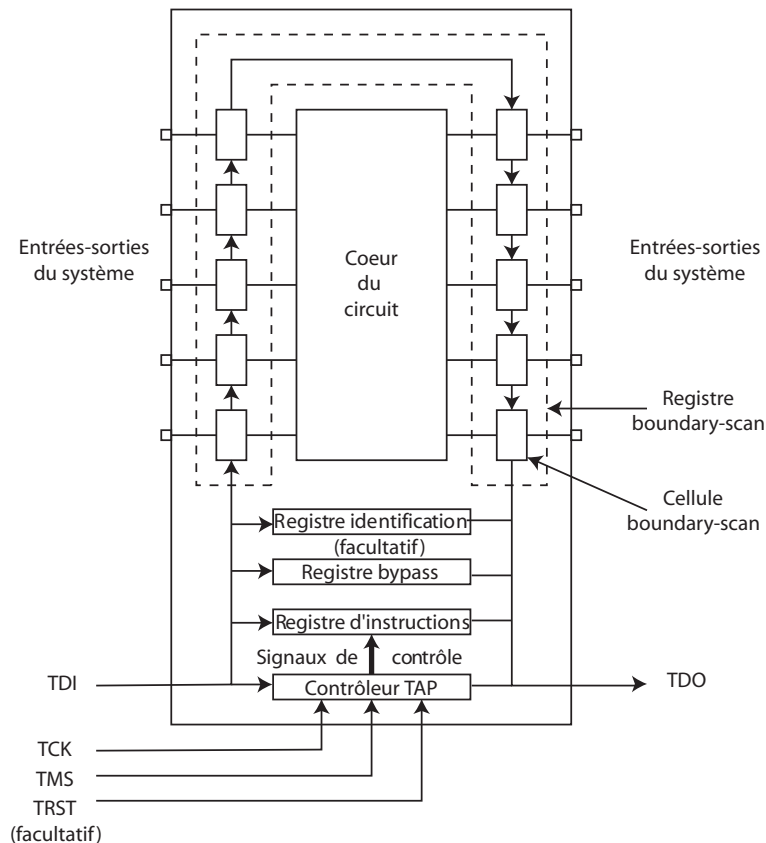


FIG. 6.1 – Architecture générique d'un circuit intégré compatible IEEE 1149.1

Sur ce schéma, nous voyons que l'architecture est constituée de plusieurs éléments :

- Un contrôleur TAP qui est une machine à seize états dont le rôle est de contrôler l'ensemble de l'architecture. Ce contrôleur est piloté par les signaux TMS (Test Mode Select) et TCK, une horloge qui lui est dédiée, indépendante du reste du circuit.
- Un ensemble de registres, qui partagent les mêmes entrées-sorties, respectivement appelées TDI (Test Data In) et TDO (Test Data Out) :
 - Des registres de données. Les deux seuls registres de données obligatoires sont le registre Boundary et le registre Bypass. D'autres peuvent exister mais sont optionnels, comme le registre d'identification. Ces registres ont un mode de fonctionnement différent selon le mode de fonctionnement de l'architecture boundary-scan.
 - Un registre d'instruction, qui va définir le mode d'opération des registres de données.
 - Le standard permet de définir des registres personnalisés en plus des registres obligatoires.

Nous allons maintenant voir le rôle de chacun de ces éléments.

6.2.1 Le contrôleur TAP

Le contrôleur TAP est une machine à seize états, pilotée par la valeur de TMS. Les transitions entre les différents états ont lieu sur les fronts montants de l'horloge TCK. Le diagramme de transition de la machine à état TAP est donné sur la figure 6.2.

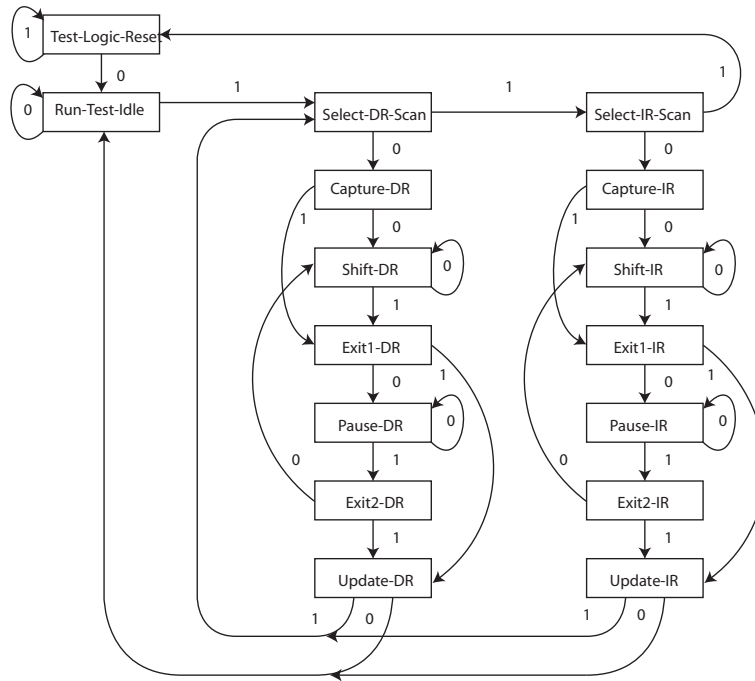


FIG. 6.2 – Diagramme de transition du contrôleur TAP

On peut remarquer deux colonnes sur ce diagramme : la première correspond aux registres de données, la seconde au registre d'instructions. Les états correspondants aux deux types de registres ont des noms similaires, car ils fonctionnent de manière à peu près identique. Décrivons brièvement chacun de ces états :

- Test-Logic-Reset est l'état reset. Il correspond à une désactivation du dispositif de test de sorte que le circuit sous test soit dans son état de fonctionnement normal. Le registre d'instruction est initialisé avec l'instruction IDCODE si le registre d'identification du circuit est présent et l'instruction BYPASS sinon. La mise sous tension du dispositif doit mettre le système dans cet état et seul un passage de TMS à la valeur "0" logique fait entrer le TAP dans l'état Run-Test-Idle. Si TMS reste haut durant cinq fronts montants de TCK, le système revient automatiquement à l'état Test-Logic-Reset, quel que soit son état.
- Run-Test-Idle est un état d'attente, permettant à certaines instructions particulières de s'exécuter, comme par exemple l'instruction RUNBIST dont nous reparlerons par la suite. Un passage de TMS à la valeur "1" logique permet de quitter cet état pour aller dans l'état Select-DR-Scan.
- Select-DR-Scan est un état de décision : si TMS passe à "0", on entre dans la colonne d'états correspondant au registre de données concerné. Sinon, on passe à l'état de décision Select-IR-Scan.
- Select-IR-Scan est un état de décision permettant d'entrer dans la colonne d'états correspondant au registre d'instructions si TMS passe à zéro. Sinon, on retourne à l'état Test-Logic-

Reset.

Une fois rentré dans une des colonnes, une succession d'états est possible pour chaque type de registre :

- Capture-IR (respectivement Capture-DR) : dans cet état, la partie capture du registre d'instructions (respectivement de données) (voir figure 6.3 pour la description de ces registres) est chargée en parallèle sur un front montant de TCK. Dans le cas du registre d'instructions, les deux bits de poids faible ont obligatoirement la valeur "01", les autres une valeur quelconque (qui ne code pas obligatoirement une instruction).
- Shift-IR (respectivement Shift-DR) : dans cet état, le registre d'instructions (respectivement de données) est connecté en série entre TDI et TDO. Le train binaire présent dans le registre est translaté de TDI vers TDO à chaque front montant de TCK. Dans cet état, un train binaire peut être évacué vers TDO tandis qu'un nouveau train binaire est chargé en série par TDI.
- Exit1-IR (respectivement Exit1-DR) est un état temporaire dans l'attente de la décision entre entrer dans l'état Pause-IR (respectivement Pause-DR) si TMS vaut "0" et Update-IR (respectivement Update-DR) si TMS vaut "1".
- Pause-IR (respectivement Pause-DR) : cet état permet d'interrompre provisoirement le décalage du registre d'instructions (respectivement de données). Ceci peut être utile par exemple lorsqu'un système ATE recharge sa mémoire.
- Exit2-IR (respectivement Exit2-DR) est un état temporaire dans l'attente de la décision entre entrer dans l'état Update-IR (respectivement Update-DR) si TMS vaut "1" et Shift-IR (respectivement Shift-DR) si TMS vaut "0".
- Update-IR (respectivement Update-DR) : dans cet état, les valeurs de bits précédemment chargés en série dans la partie capture du registre d'instructions (respectivement du registre de données) sont chargées dans la partie maintien de ce même registre. Ceci a lieu sur un front descendant de CLK. Voir la figure 6.3 pour la description des registres d'instructions et de données.

En fait, le TAP va générer les signaux `shift_IR`, `capture_IR`, `update_IR`, `shift_DR`, `capture_DR` et `update_DR`, qui vont permettre de choisir les modes de fonctionnement des registres d'instruction et de données (voir description de ces registres).

6.2.2 Le registre d'instructions

Le registre d'instructions est composé suivant le principe d'un registre boundary-scan typique. Un registre Boundary est composé de deux parties, comme le montre la figure 6.3 : une partie capture de données et une partie maintien des données. Ce principe permet de pouvoir maintenir des données en parallèle en sortie du registre, tout en chargeant de nouvelles données en série dans la partie capture. Un exemple de dessin d'une cellule du registre d'instructions est donné figure 6.4.

La fonction de ce registre est de définir le mode sur lequel les registres de données vont fonctionner. Il contient une instruction qui va être décodée par un circuit de décodage pour fournir des commandes "mode" correspondant à chaque cellule du registre de données, dont nous allons voir le fonctionnement par la suite.

On peut voir sur la figure 6.4 qu'une cellule du registre d'instructions est pilotée par les signaux `shift_IR`, `clock_IR` et `update_IR`. `clock_IR` est un signal dérivé du signal CLK qui sert à capturer ou à décaler les données. `update_IR` est dérivé du signal CLK complémenté et sert à mettre à jour la partie maintien du registre sur un front montant (donc sur un front descendant de CLK, pour

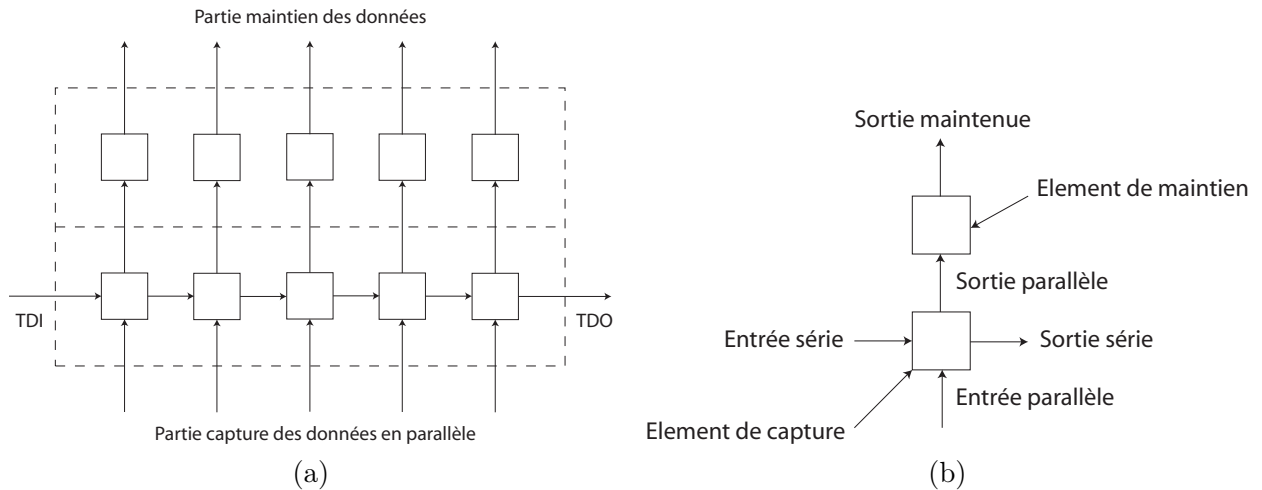


FIG. 6.3 – Architecture d'un registre boundary-scan standard avec une partie capture et une partie maintien : (a) le registre et (b) une cellule du registre avec un élément de capture et un élément maintien

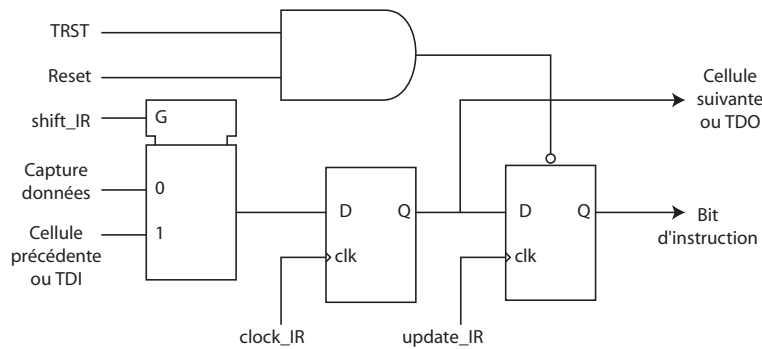


FIG. 6.4 – Dessin possible d'un cellule du registre d'instruction

rendre disponible une instruction. Ces signaux sont générés par le contrôleur TAP et déterminent l'état de l'architecture IEEE 1149.1.

6.2.3 Les registres de données

Le registre Boundary

Le registre Boundary est le registre de données le plus important : c'est lui qui permet de faire l'interface entre les plots du circuit et le circuit à tester. Il est construit sur un principe semblable à celui du registre d'instructions, constitué de cellules boundary-scan, dont un exemple de conception est donné sur la figure 6.5.

On voit sur cette figure que que l'on retrouve une architecture similaire à celle du registre d'instructions, avec les commandes `shift_DR`, `clock_DR` et `update_DR` également générées par le contrôleur TAP et qui déterminent l'état correspondant du système. On peut également noter la présence du signal "mode". Ce signal est généré par le registre d'instructions et le décodeur d'instructions et détermine le mode de fonctionnement de la cellule. Ces différents modes sont décrits dans le tableau 6.1.

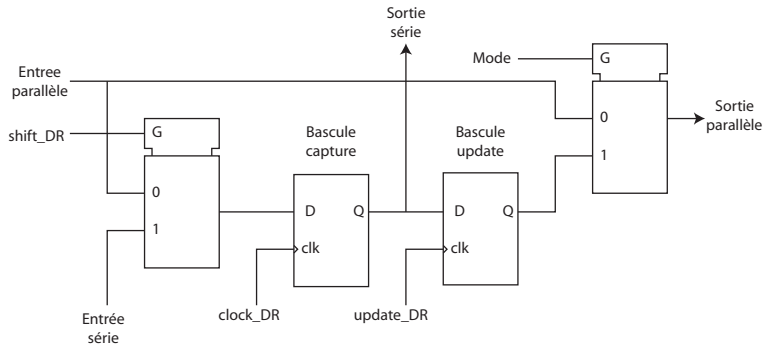


FIG. 6.5 – Dessin possible d'une cellule boundary-scan

Schéma	Mode	Valeurs des signaux de contrôle	Commentaires
	Normal	$mode = 0$	Cellule transparente : mode normal de fonctionnement de la cellule sous test
	Scan	$shift = 1$	Décalage des données dans la partie capture du registre Boundary
	Capture	$shift = 0$	Données chargées en parallèle dans la partie capture du registre Boundary
	Update	$\nearrow update$	Données de la partie capture rendues accessibles dans la partie maintien des données, sur un front montant de $update$

TAB. 6.1 – Les différents modes de fonctionnement des cellules boundary-scan

Le registre Bypass

Le registre bypass consiste en une seule cellule et ne permet pas de maintien des données en parallèle. Ce registre est sélectionné par l'instruction BYPASS. Elle sert à "court-circuiter" un composant en le réduisant à une seule cellule. Ceci est utile par exemple pour diminuer le temps de test lorsqu'on veut tester d'autres composants sur une carte.

Le registre d'identification

Ce registre, facultatif, doit, s'il existe, avoir une longueur de 32 bits. Il contient un code identifiant le composant.

Les registres définis par l'utilisateur

Ces registres sont facultatifs et peuvent s'avérer très utiles pour effectuer des tests par une technique de scan-path par exemple. Comme nous l'avons déjà expliqué, cette technique consiste à introduire un vecteur de test en série dans un registre à décalage et à distribuer ses valeurs en parallèle à différents points du circuit. Cette technique permet de diminuer le nombre de plots. Pour utiliser ce type de technique, le concepteur peut décider de ne pas utiliser le registre Boundary et de concevoir un registre supplémentaire pour accéder à d'autres noeuds du circuit. Cette méthode peut en particulier être utilisée avec des instructions spécifiques dans une technique de BIST comme nous le verrons par la suite.

6.2.4 Les différents modes de fonctionnement de l'architecture

Nous pouvons maintenant donner un schéma plus détaillé de l'architecture 1149.1 (figure 6.6), montrant les différents signaux générés par le TAP et le registre d'instructions.

Le principe de fonctionnement de l'architecture pour effectuer un test est le suivant :

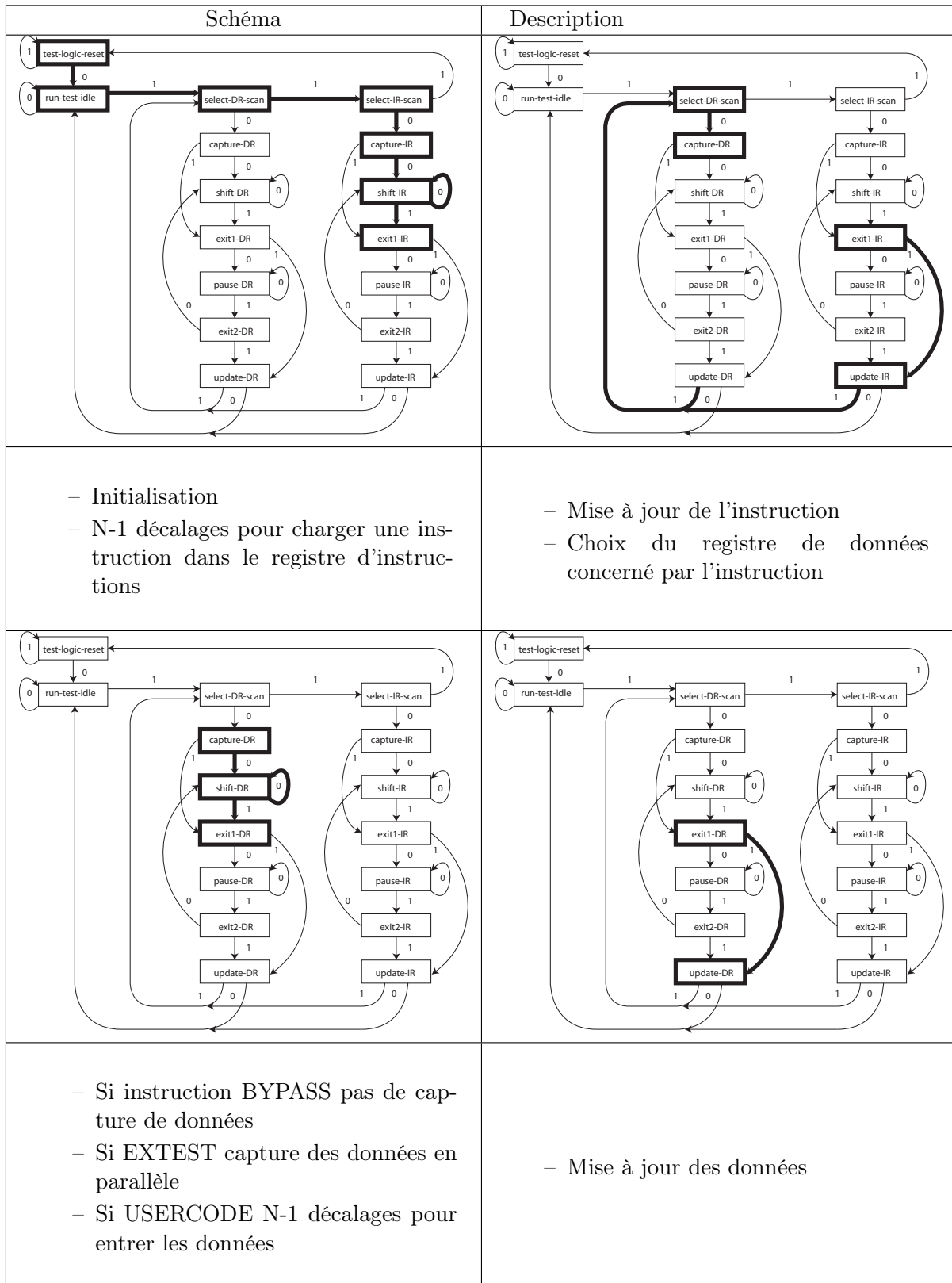
- Initialisation du TAP et des registres
- Chargement d'une instruction en mode série dans le registre d'instructions par TDI
- Configuration par le TAP des circuits concernés pour effectuer cette instruction
- Chargement des stimuli dans les registres de données
- Exécution de l'instruction et récupération des données en série par TDO
- Réinitialisation du TAP et des registres

Afin d'illustrer plus précisément cette séquence, reportons nous au diagramme d'états du TAP (tableau 6.2).

Les modes non-invasifs

Le type de test à effectuer est piloté par l'instruction du registre d'instructions et donc par le mode de fonctionnement des registres de données. Nous allons donc passer en revue les différents modes d'opération de l'architecture. Nous commençons par les modes dits "non-invasifs", c'est-à-dire qui ne déconnectent pas le circuit sous test de ses pins. Autrement dit, si l'on se réfère à la figure 6.5, le signal "*mode*" vaut "0" pour tous ces modes.

Le mode BYPASS place le registre Bypass entre TDI et TDO pour éviter le test du composant concerné dans le cas du test d'autres composants d'une carte par exemple. Ceci permet de gagner en temps de test.



TAB. 6.2 – Séquence de scanning

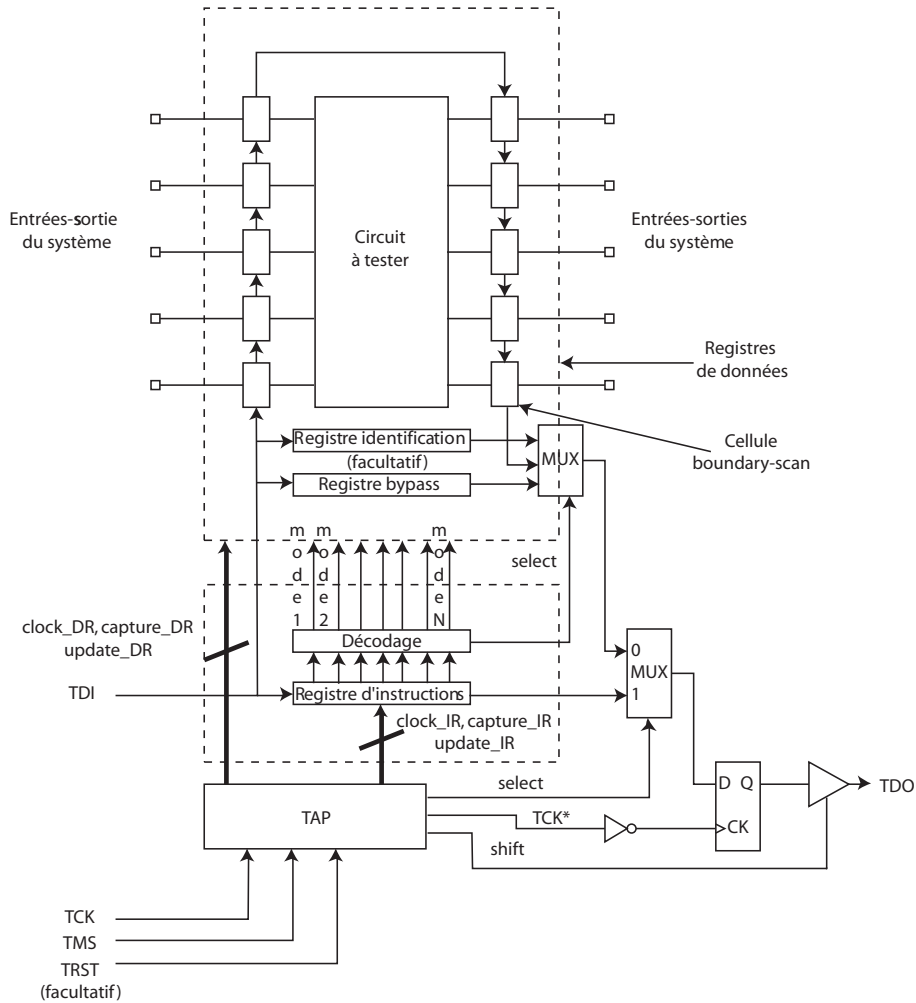


FIG. 6.6 – Schéma détaillé d'une architecture boundary-scan

Le mode IDCODE place le registre d'identification (s'il est présent) entre TDI et TDO pour identifier le composant. Ce registre est alors chargé en parallèle par le code identifiant le composant.

Le mode USERCODE place à nouveau le registre d'identification entre TDI et TDO, mais cette fois, ce dernier est chargé par une instruction définie par l'utilisateur. Ceci permet d'identifier le composant dans le cas par exemple où celui-ci est programmable.

Le mode SAMPLE (encore appelé mode d'échantillonnage) place le registre Boundary entre TDI et TDO. Dans ce mode, les cellules d'entrée et les cellules de sortie du circuit sont en mode "capture" (voir tableau 6.1). Ce mode permet d'échantillonner les entrées et les sorties du circuit (c'est-à-dire les introduire dans la chaîne boundary-scan) pendant son fonctionnement normal. Ceci permet par exemple de tester les entrées et certaines valeurs logiques du système.

Le mode PRELOAD place le registre Boundary entre TDI et TDO sans toutefois déconnecter le circuit de ses pins (mode non-invasif). Ce mode est utilisé pour initialiser les cellules du registre Boundary.

Les modes invasifs

Voyons maintenant les modes qui utilisent les pins du circuit, en ce sens qu'ils déconnectent le coeur du circuit de ces pins (le signal "mode" vaut "1") :

Le mode EXTEST (encore appelé test d'interconnexions) place le registre Boundary entre TDI et TDO et déconnecte les pins du coeur du circuit : dans ce mode, les cellules d'entrée du circuit sont en mode "*capture*", tandis que ses cellules de sortie sont en mode "*update*". Les données des pins sont chargées en série dans les cellules de sortie des différents composants. Ces données sont ensuite translatées vers la partie maintien de ces cellules et sont donc introduites en parallèle dans les cellules d'entrée du composant suivant. Après l'opération de test, les données du registre Boundary sont extraites en série afin de vérifier l'intégrité du train binaire, pour confirmer que les connexions entre les puces se font correctement.

Le mode INTEST est un mode optionnel, qui permet de tester le composant à travers la chaîne boundary-scan. Dans un premier temps, le vecteur de test est chargé dans le registre Boundary en série. Puis, ce vecteur est envoyé au coeur du circuit par une opération de "*update*" sur les cellules d'entrée. Les bascules de sorties entrent alors en mode "*capture*" afin que les valeurs de sortie soient introduites dans le registre Boundary. Les valeurs de sortie peuvent alors être extraites en série pour vérification de la réponse du circuit.

Le mode RUNBIST est un mode optionnel. Il place un registre quelconque entre TDI et TDO. Ce registre est souvent un registre défini par l'utilisateur, mais ce peut être un registre existant, comme le registre Bypass ou Boundary. Typiquement, ce registre va être utilisé pour distribuer les bits d'entrée lors d'un BIST et pour récupérer la réponse du circuit. Ces résultats devront donc être récupérables par décalage de ce registre. C'est cette instruction que nous allons utiliser pour rendre notre technique compatible avec les standards.

6.3 Les standards de test analogique et mixte (IEEE 1149.4)

Le standard 1149.1 est adapté aux circuits numériques. Beaucoup de systèmes aujourd'hui embarquent des parties analogiques et des convertisseurs entre les parties numériques et analogiques. Dans ce cas, des cellules boundary-scan sont insérées entre la partie numérique et les convertisseurs comme le montre la figure 6.7 (ces cellules font partie alors de la chaîne boundary-scan). Ces cellules sont toujours connectées au circuit numérique d'un côté, mais l'autre est connecté aux convertisseurs plutôt qu'aux plots. Cette technique n'est pas satisfaisante, pour plusieurs raisons : elle ne permet pas de tenir compte des plots analogiques et les circuits même numériques présentent de plus en plus de caractéristiques analogiques (délais, tension de biais, de référence par exemple). Un groupe de travail a été créé en 1992 pour mettre au point une norme analogique (IEEE 1149.4). Cette norme utilise des éléments analogiques et introduit la notion d'interconnexions "étendues", c'est-à-dire ayant des caractéristiques analogiques dont il faut tenir compte. L'architecture générique d'un circuit intégré compatible avec le standard IEEE 1149.4 est montrée sur la figure 6.8.

Le principe est globalement le même que pour le standard 1149.1. Il y a cependant la possibilité supplémentaire de charger ou d'extraire des signaux analogiques. Des éléments analogiques sont ajoutés pour la manipulation de ces signaux : le ATAP (TAP Analogique), le bus interne analogique,

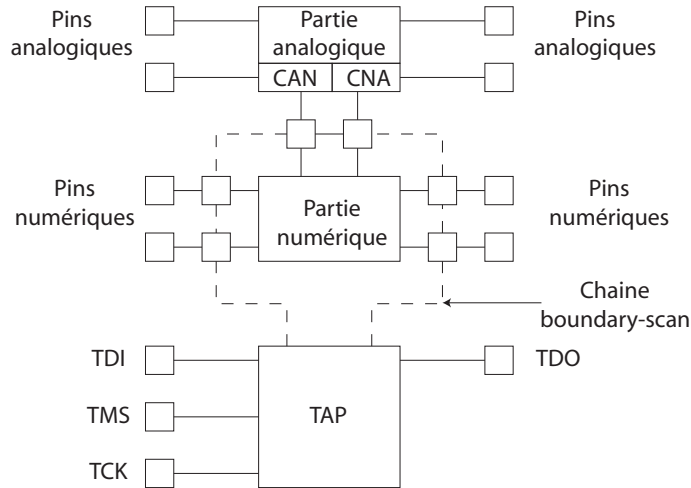


FIG. 6.7 – Architecture générique d'une architecture IEEE 1149.1 dans le cas d'un circuit mixte

le TBIC (circuit d'interface pour l'accès au bus interne) et les ABM (modules boundary-scan analogiques). Les cellules de boundary-scan numériques de la norme 1149.1 sont renommées en DBM (modules boundary-scan digitaux). Les autres éléments, notamment le TAP, sont les mêmes que dans un standard 1149.1.

Le ATAP est constitué de deux pins au moins :

- AT1 qui sert généralement à introduire un signal analogique de test dans l'architecture,
- AT2 qui sert à extraire de la puce une réponse analogique en vue par exemple de faire des mesures.

Dans le cas de signaux différentiels, deux autres pins peuvent être présents (AT1N et AT2N) correspondant respectivement à AT1 et AT2. Le bus interne permet éventuellement de transmettre ces signaux aux ABM. Il est composé de deux connexions AB1 et AB2 correspondant respectivement aux pins AT1 et AT2. Enfin, le TBIC est un ensemble d'interrupteurs analogiques commandés par le TAP, qui permettent de contrôler la connexion entre le bus interne et les pins AT1 et AT2. Le TBIC est équipé d'une logique de contrôle qui permet, en fonction des signaux issus du TAP et du registre de contrôle, de connecter en particulier AT1 et AT2 à AB1 et AB2, ou aux tensions d'alimentation V_H et V_L .

Les ABM sont des cellules analogiques comparables aux cellules boundary-scan numériques : elle sont connectées en série pour former une partie du registre Boundary. Ces cellules offrent des possibilités d'opérations de contrôle et d'observation, permettent le test d'interconnexions mais aussi certains types d'opérations de test paramétrique. Comme le TBIC, les ABM possèdent un dispositif à base d'interrupteurs contrôlés par une logique de contrôle. Ces interrupteurs permettent :

- de connecter ou non les pins au coeur du circuit,
- de connecter le pin à AB1 dans le but d'injecter un signal de test,
- de connecter le pin à AB2 dans le but d'extraire et de visualiser une réponse,
- de connecter le pin à V_H ou V_L pour des tests d'interconnexion,
- de fournir une sortie à un bit représentative de la tension du pin (par comparaison à la valeur V_{TH}),
- de connecter le pin à une tension de référence V_G pour effectuer un test paramétrique notamment.

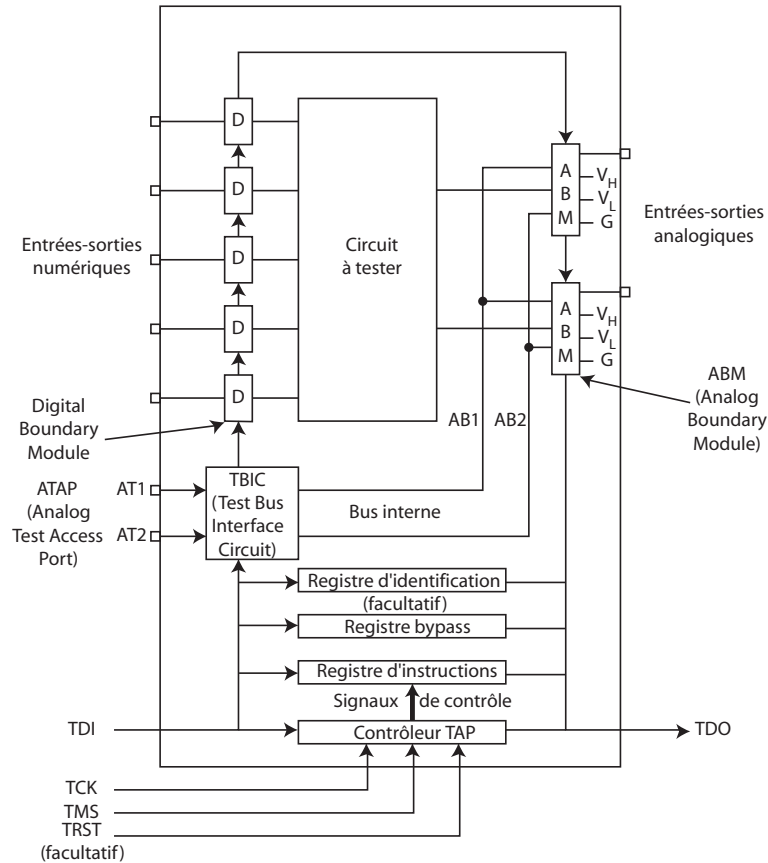


FIG. 6.8 – Architecture générale d'un circuit intégré compatible IEEE 1149.4

Le principe de fonctionnement est sensiblement le même que pour le standard numérique et les instructions sont semblables : selon le mode de fonctionnement, le coeur peut être connecté aux pins par les ABM ou non. On peut aussi connecter les pins à un signal analogique de test, aux tensions d'alimentation ou à un signal interne de référence (toujours par les ABM). Une nouvelle instruction obligatoire a cependant été introduite : l'instruction PROBE qui est sensiblement la même que l'instruction SAMPLE mais dans le domaine analogique, avec la restriction que l'on ne peut échantillonner qu'un plot analogique à la fois, car un seul bus AB est disponible pour l'échantillonnage. Les instructions BYPASS, PRELOAD, IDCODE et USERCODE sont exactement les mêmes que dans le standard numérique en ce sens qu'elles sont non-invasives et n'ont pas d'effet sur les pins analogiques. L'instruction EXTEST est basée sur le même principe que son homologue dans le domaine numérique, mais en introduisant des possibilités de mesures analogiques. Les autres instructions sont également comparables entre les deux standards avec des considérations analogiques. Nous ne développerons pas plus ces instructions ici.

Nous allons maintenant montrer comment la conception de l'architecture de BIST présentée dans les chapitres précédents la rend facilement compatible avec ces standards.

6.4 Compatibilité de l'architecture avec les différents standards

6.4.1 Compatibilité avec la norme 1149.1

Comme nous l'avons vu, l'architecture de BIST a une interface entièrement numérique: les signaux d'entrée sont numériques et la réponse est un train binaire qui est analysé de manière numérique. Il est donc possible d'intégrer cette technique dans une architecture 1149.1, dans le cas où l'on ne désire pas avoir accès aux noeuds analogiques. Dans ce cas, deux solutions sont possibles, selon que l'on veut incorporer le BIST dans une architecture dédiée (on a alors la possibilité de définir de nouvelles instructions) ou une architecture existante (il est alors impossible de définir de nouvelles instructions).

Utilisation dans une architecture IEEE 1149.1 dédiée

Une architecture est proposée sur la figure 6.9. Dans cette architecture, un registre utilisateur est ajouté à l'architecture, constitué du registre à décalage contenant le train binaire, du registre de contrôle contenant les signaux servant à programmer l'architecture et du registre qui va contenir la signature. Afin de ne pas surcharger la figure, nous n'avons pas représenté ici le registre Boundary classique, qui est bien entendu présent, faisant l'interface avec les pins pour les tests classiques d'interconnexions par exemple. Comme nous l'avons signalé, l'ajout d'un registre est une méthode classique pour lancer un BIST en utilisant l'instruction RUNBIST.

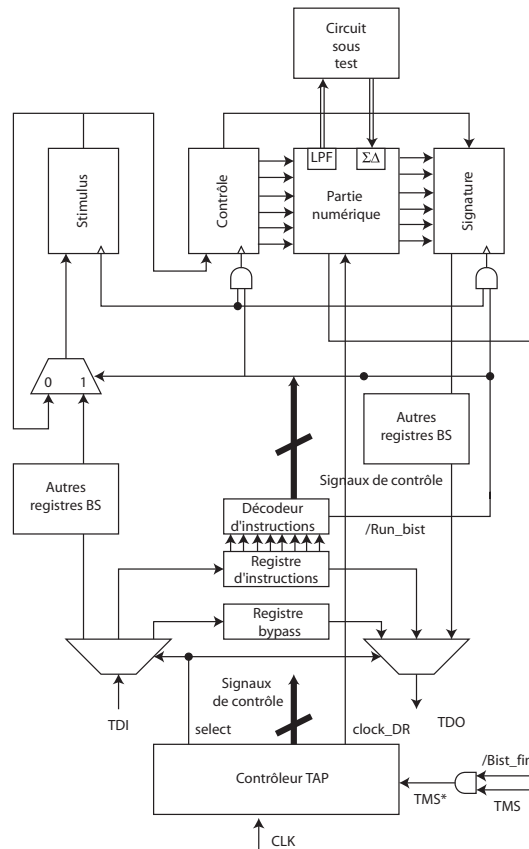


FIG. 6.9 – Intégration du BIST dans une architecture IEEE 1149.1 dédiée

L'enchaînement des opérations pour mener à bien le test est le suivant :

- Charger les registres de contrôle et de stimulus en série en utilisant le même train binaire, durant un état `shift_DR` du diagramme d'états
- Charger le registre d'instructions avec une instruction `RUNBIST`
- Tant que l'instruction est active, `TCK` permet de décaler les bits du registre de stimuli mais sans décaler les registres de contrôle et de signature (l'horloge est désactivée pour ces registres). Ces registres devront donc avoir une structure de type `boundary-scan`. Durant le même temps, le signal `BIST_fin` reste à sa valeur basse pour désactiver le signal `TMS` local (`TMS*`)
- Quand le test est terminé, `BIST_fin` passe au niveau haut et la signature peut être extraite en série avec une opération `shift_DR`.

Utilisation dans une architecture IEEE 1149.1 existante

Dans le cas où le BIST est intégré dans une architecture existante, il n'est pas a priori possible de définir de nouvelles instructions. Nous allons donc avoir besoin de rajouter une logique supplémentaire comme indiqué sur la figure 6.10. Nous voyons sur cette figure qu'un registre d'instruction de travail (`WIR`, Working Instruction Register) associé à une logique de contrôle est ajouté en série avant les autres registres `boundary-scan`.

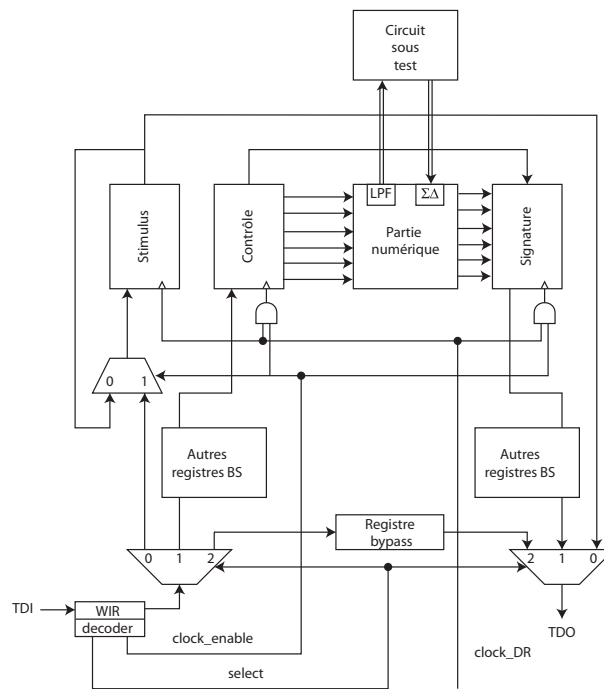


FIG. 6.10 – Intégration du BIST dans une architecture IEEE 1149.1 existante

L'enchaînement des opérations devient :

- Charger le `WIR` avec l'instruction `W_LoadStimulus` dans un mode `shift_DR`
- Charger le registre d'instructions avec une instruction `PRELOAD`. Les multiplexeurs sont ainsi en position "0"
- Charger le registre de stimulus avec le train binaire et le `WIR` avec l'instruction `W_LoadControl` en utilisant un mode `shift_DR`

- Charger l'instruction globale PRELOAD dans le registre d'instructions, les multiplexeurs étant alors en position "1"
- Charger le registre de contrôle avec les bits de contrôle et le WIR avec l'instruction W_RunBIST
- Charger l'instruction PRELOAD les multiplexeurs sont alors en position "1" et TMS* est désactivé jusqu'à ce que la signature soit prête
- La signature est alors extraite et une nouvelle instruction peut être chargée dans le WIR

6.4.2 Utilisation avec la norme 1149.4

Cependant, il est très intéressant d'utiliser une norme IEEE 1149.4. En effet, comme nous l'avons vu, une architecture boundary-scan permet de faire des tests d'interconnexions entre les composants d'une carte. Si la partie analogique communique avec l'extérieur du circuit par des plots analogiques, on ne peut tester ces interconnexions étendues que par un standard 1149.4. De plus, si l'on désire pouvoir également envoyer directement des signaux de test analogiques au circuit analogique, une architecture 1149.4 s'avère indispensable. La figure 6.11 présente des possibilités d'intégration du BIST dans un standard de test analogique, selon, comme précédemment, que l'on s'insère dans une architecture dédiée ou existante. Nous voyons que cette figure est très similaires aux figures présentées dans le cas d'une intégration dans une architecture numérique : un registre utilisateur a été également ajouté, mais celui-ci contient des cellules ABM entre les parties analogique et numérique, afin de pouvoir éventuellement injecter un signal analogique directement dans la partie analogique pour effectuer un test traditionnel. De toute façon, le registre Boundary classique est là encore présent (bien que non représenté sur les figures) pour l'accès aux pins numériques et analogiques. L'enchaînement des instructions est exactement le même que dans une architecture 1149.1.

6.5 Conclusion

L'architecture de BIST proposée utilise le concept de scan-path afin de rentrer les vecteurs de test en série et d'extraire la signature de la même façon. Globalement, un effort de conception a été fait pour que l'architecture soit facilement compatible avec les normes de test. Les vecteurs de test étant numériques ainsi que la signature et son traitement, le BIST en lui-même est entièrement compatible avec le standard 1149.1. Cependant, si l'on considère l'ensemble du circuit mixte ainsi que la circuiterie de test et que l'on veut pouvoir tenir compte d'éventuels plots du circuit analogique, il faut alors intégrer l'ensemble dans une architecture compatible 1149.4, ce qui est également facilement réalisable. Les entrées-sorties du BIST consistant uniquement en deux signaux analogiques différentiels, le BIST pourra être fourni sous forme d'IP facilement intégrable dans un système.

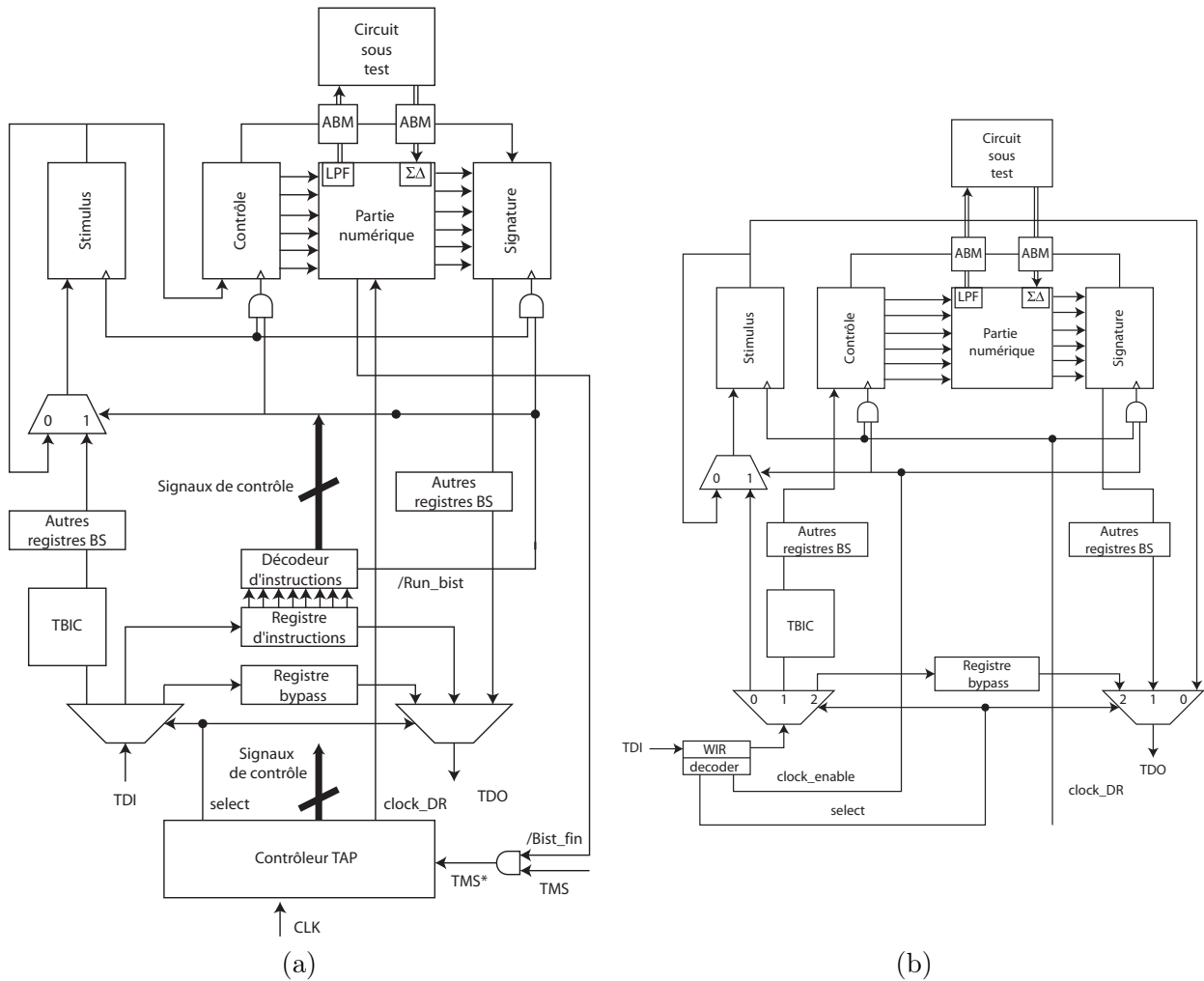


FIG. 6.11 – Intégration du BIST dans une architecture IEEE 1149.4: (a) architecture dédiée et (b) architecture existante

Chapitre 7

Conclusions et travaux futurs

Les recherches sur les techniques de BIST analogiques et mixtes sont en pleine évolution. Cependant, à l'heure actuelle, peu de solutions industrielles existent et la plupart sont assez spécifiques ou font intervenir du matériel externe à la puce. Le but de cette thèse était de présenter une technique de BIST qui soit totalement intégrée, dans le sens où aucun élément extérieur ne participe au test : la génération de signaux de test devait se faire sur la puce ainsi que l'analyse de la réponse, de telle sorte que la puce produise elle-même un signal l'identifiant comme fonctionnelle ou défaillante. Cette technique devait être la plus générique possible, c'est-à-dire qu'elle ne soit pas limitée à une catégorie particulière de circuits. Enfin, l'interface de la technique devait être totalement numérique et compatible avec les standards de test existants. L'architecture à laquelle nous avons abouti s'appuie sur une technique de génération de signaux de test sinusoïdaux par utilisation d'un train binaire codant ce signal et sur une technique d'analyse harmonique de la réponse qu'elle marie avantageusement afin d'accomplir un test fréquentiel à bas coût, à une vitesse intéressante et avec une surface de silicium supplémentaire raisonnable.

La technique de génération permet de générer des signaux périodiques mono ou multifréquences de bonne qualité (au moins 60 dB de SFDR) mais avec une faible résolution dans le choix de l'amplitude. Ceci limite l'utilisation de cette technique de génération à des techniques de test qui ne nécessitent pas de générer un signal d'amplitude donnée (ce qui est le cas ici). Une autre limitation de cette méthode vient du principe qui consiste à filtrer un train binaire issu d'un modulateur $\Sigma\Delta$: la nature intrinsèquement analogique de ce train binaire fait que des variations des tensions d'alimentation peuvent altérer la qualité du signal à un moment donné. Dans notre cas, l'extraction des composantes harmoniques se fait par moyennage sur un grand nombre de périodes du signal modulé, donc cette limitation ne doit pas poser de problème non plus. Il aurait été intéressant de vérifier cette assertion par simulation en ajoutant un bruit à la sortie du modulateur $\Sigma\Delta$, mais ceci aurait obligé à voir le train binaire de manière analogique et donc d'augmenter fortement le pas de calcul des simulations de Monte Carlo, ce qui s'est avéré impossible dans la pratique.

La technique d'analyse est basée sur la modulation du signal par un signal carré puis sur une intégration du résultat pour évaluer un harmonique. La limitation de cette technique vient du fait que les signaux carrés introduisent des harmoniques impairs dans la modulation. Ceci a pour effet de rendre l'évaluation de la valeur d'un harmonique dépendante de la valeur des suivants. L'utilisation de cette méthode pour mesurer les harmoniques ne va donc être possible que si ces harmoniques suivants sont négligeables (ce que l'on ne peut pas affirmer a priori). Ici, cette technique a donc été utilisée de manière relative : des signatures sont définies, qui correspondent à chacun des harmoniques sans en être réellement une mesure. La signature du circuit va être constituée des signatures

correspondant à différents harmoniques et comparée à la signature nominale correspondant à un circuit parfait, obtenue par simulation et suivant le même principe. Cette signature subissant elle aussi l'effet de la modulation carrée qui est parfaitement déterministe, ces variations ne vont pas troubler le test et ce sont bien les variations des harmoniques, donc les variations de performances qui vont être détectées.

Les deux aspects de génération et d'analyse ainsi que l'architecture complète ont été validés en simulation. Deux démonstrateurs ont été réalisés, l'un contenant l'architecture complète, et donnent des résultats encourageants :

- La partie génération fonctionne correctement avec une qualité de signaux satisfaisante. Nous avons cependant noté un retrait des performances par rapport à la simulation, mais il est très difficile de déterminer quelle est la part de dégradation due aux conditions de test.
- La partie d'analyse n'a pas pu être testée à cause d'un mauvais fonctionnement du modulateur $\Sigma\Delta$, interdisant malheureusement de valider son fonctionnement et donc celui de l'architecture complète. Ce résultat est le principal manque dans le travail effectué au cours de cette thèse. L'idéal serait de comprendre pourquoi le modulateur ne fonctionne pas correctement et de refabriquer un démonstrateur afin de pouvoir tester l'ensemble de la technique dans les conditions réelles.

Le coût supplémentaire en surface est acceptable et on peut espérer réutiliser certains éléments analogiques présents dans le circuit pour le test, afin d'économiser en surface. La programmabilité de l'architecture est également un moyen intéressant de permettre l'optimisation de la signature, d'économiser de la surface de circuit et de diminuer le temps de test.

Cette technique est assez générique. Elle peut s'appliquer :

- à des circuits fonctionnant à une fréquence modérée (à cause du suréchantillonnage),
- à des circuits linéaires ou non linéaires,
- à des tests ne nécessitant pas de connaître l'amplitude du signal généré avec précision.

Pour le test de convertisseurs, le principe de la technique peut être encore valable. Mais il faut une résolution en amplitude beaucoup plus importante pour la partie génération. [95] a démontré la validité du principe de génération pour des signaux ayant une précision supérieure à 20 bits. Par contre, la validité du principe d'analyse de signature pour le test de convertisseurs reste encore à étudier.

Enfin, nous avons fait un effort de conception avec un chargement des données de type scan-path afin que l'architecture soit facilement compatible avec les standards de test. Nous avons montré dans le dernier chapitre que cette compatibilité pouvait très facilement être assurée.

Les résultats de cette thèse sont donc très encourageants, mais beaucoup de travaux restent à accomplir autour de cette technique :

- Il serait intéressant d'évaluer l'utilisation de signaux multifréquences dans le cadre de cette technique.
- L'utilisation de cette technique pourrait être évaluée dans le cadre du test de convertisseurs qui est le sujet d'une thèse en cours dans le groupe de travail.
- Il serait envisageable d'améliorer la qualité du signal analogique généré en utilisant un CAN de 1 bit pour générer les niveaux du train binaire avec une bonne précision.
- Un nouveau démonstrateur devrait être fabriqué afin de confirmer les résultats, avec un effort de conception portant sur la compatibilité de l'architecture avec les normes de test, l'utilisation d'une technologie plus récente et la réalisation d'un circuit plus abouti et plus robuste permettant des mesures précises des performances.

Annexe A

Schémas de certaines parties des démonstrateurs

A.1 Circuits analogiques

A.1.1 Premier amplificateur

L'amplificateur utilisé pour le premier démonstrateur est d'une structure classique à deux étages (figure A.1). Dans cette figure, seule la largeur des transistors est indiquée, toutes les longueurs étant prises à $0,18 \mu\text{m}$, le minimum de la technologie. La configuration classique d'un amplificateur contient généralement un troisième étage qui est un buffer de sortie, dans le cas où l'on a une charge résistive. Ici, nous n'avons qu'une charge capacitive en sortie, nous nous contentons donc des deux premiers étages :

- un premier étage constitué d'une paire différentielle (T_1 et T_2) et d'un miroir de courant (T_3 et T_4) pour charger celle-ci en courant,
- un étage de gain constitué du transistor T_5 en source commune avec T_6 .

La polarisation est assurée par les transistors T_6 , T_7 et T_8 montés en miroir de courant pour que les intensités les traversant soient les mêmes que celle traversant R .

Nous utilisons une compensation Miller pour le second étage : celle-ci est d'habitude assurée par une capacité dite de Miller C en série avec une résistance. Dans notre cas, la résistance est réalisée par un transistor (T_9) en zone résistive.

A.1.2 Second amplificateur

Nous avons constaté en testant le premier prototype des limitations dues au premier amplificateur, notamment une très grosse consommation de courant et un bruit à basse fréquence. La conception de l'amplificateur a donc été revue par Luis Rolíndez pour le second démonstrateur, avec une structure différentielle.

La conception d'amplificateurs différentiels nécessite un circuit de contre-réaction en mode commun (CMFB) pour contrôler la tension de sortie en mode commun. La conception d'un tel circuit est assez délicate :

- La boucle de mode commun (CM) doit avoir un gain A_{CM} aussi élevé que possible et une bande passante (GBW_{CM}) comparable à celle de la boucle différentielle.
- Contrairement à l'étage d'entrée, l'étage de sortie travaille avec des signaux d'entrée de grande amplitude et doit fonctionner avec une grande dynamique.

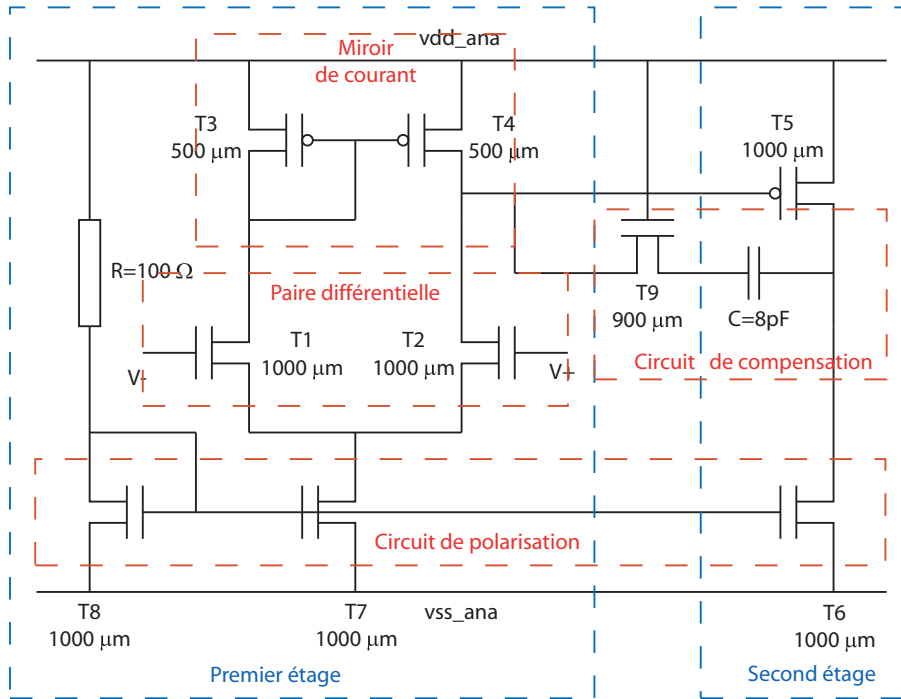


FIG. A.1 – Schéma de l'amplificateur utilisé pour le premier démonstrateur

L'étage principal de l'amplificateur est constitué d'une structure cascode rebouclée. Le CMFB utilisé en entrée les tensions V_{out+} et V_{out-} et fournit la tension V_{CM} pour compenser le niveau DC de sortie de l'amplificateur. Le circuit de polarisation fournit les tension V_{bias_p} , V_{bias_n} , V_{cas_p} et V_{cas_n} qui sont nécessaires aux étages suivants. Le dernier étage est un circuit de déclenchement.

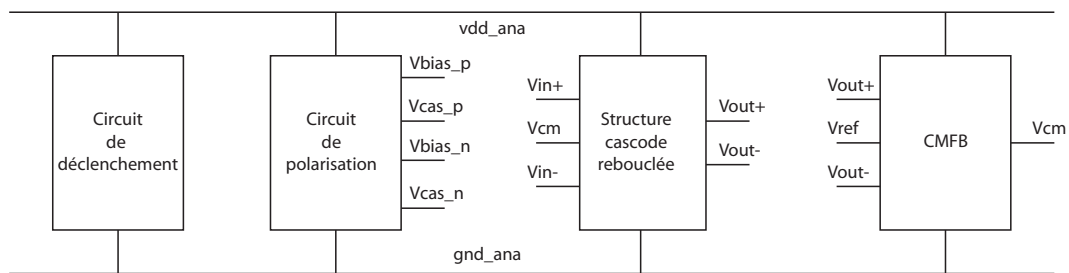


FIG. A.2 – Les différents étages de l'amplificateur du second démonstrateur

La structure cascode rebouclée (figure A.3) assure la stabilité de l'amplificateur pour toute charge capacitive.

Le circuit de contrôle de tension en mode commun est décrit dans [94].

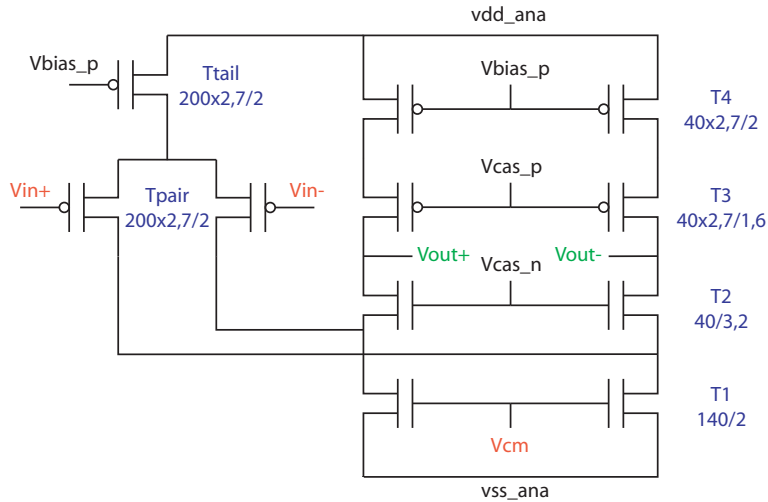


FIG. A.3 – Schéma de la structure repliée

A.2 Circuits numériques

A.2.1 Compteur

Le schéma détaillé du compteur programmable est donné figure A.4. Il est constitué essentiellement d'un compteur et d'un décompteur programmable.

Le but de ce compteur est de compter le nombre de "1" du train binaire sur un certain nombre de périodes de celui-ci. Pour cela, nous allons utiliser un signal carré de fréquence égale à celle du train binaire. Ce signal est facile à générer, toujours par division de la fréquence d'horloge : nous rappelons que pour évaluer la $i^{\text{ième}}$ harmonique, nous devons générer des signaux carrés (SQ) de fréquence $\frac{f_{clock}}{N}$, c'est-à-dire $\frac{f_{clock} \times i}{N}$. Pour obtenir un signal à la fréquence du train binaire, il suffit donc de diviser à nouveau la fréquence de SQ par i pour obtenir le signal SQ' de fréquence $\frac{f_{clock}}{N}$, c'est-à-dire celle du train binaire. Ces divisions de fréquence sont assurées par un diviseur de fréquence programmable qui sera décrit dans la section suivante. Ici, c'est la valeur de i (facteur de division) qui est codée par les bits S_6 , S_7 et S_8 .

Le nombre de périodes de comptage est chargé dans le décompteur pendant que TM vaut "1". Ce compteur va être décrémenté à chaque période du train binaire, c'est-à-dire du signal SQ' , jusqu'à ce que sa valeur de sortie soit nulle. Le signal dec passe alors à "1", mettant fin au comptage en agissant sur l'entrée $clear$ de la bascule D4. Nous désirons pouvoir compter jusqu'à 256 périodes, le décompteur est donc sur 8 bits. La longueur du registre est au maximum 200 nous voulons donc pouvoir compter jusqu'à 51200, le compteur doit donc être sur 16 bits.

Nous voulons également attendre deux périodes du train binaire avant de commencer à compter pour s'assurer que la stabilité est bien établie. Pour cela, nous utilisons le registre constitué des bascules D1 et D2 : la valeur "1" est chargée à l'entrée de la première bascule et SQ' cadence ces bascules, si bien que la sortie du registre passe à "1" au bout de deux périodes pour autoriser le comptage. La décrémentatation du décompteur commence en même temps et a lieu sur front montant de $\overline{SQ'}$ c'est-à-dire sur front descendant de SQ' (on compte le nombre de périodes écoulées du train binaire).

Nous cherchons à compter le nombre de "1" du train binaire. La porte AND2 permet d'incrémenter le compteur sur front descendant de l'horloge (pour éviter des problèmes d'aléa) si le train

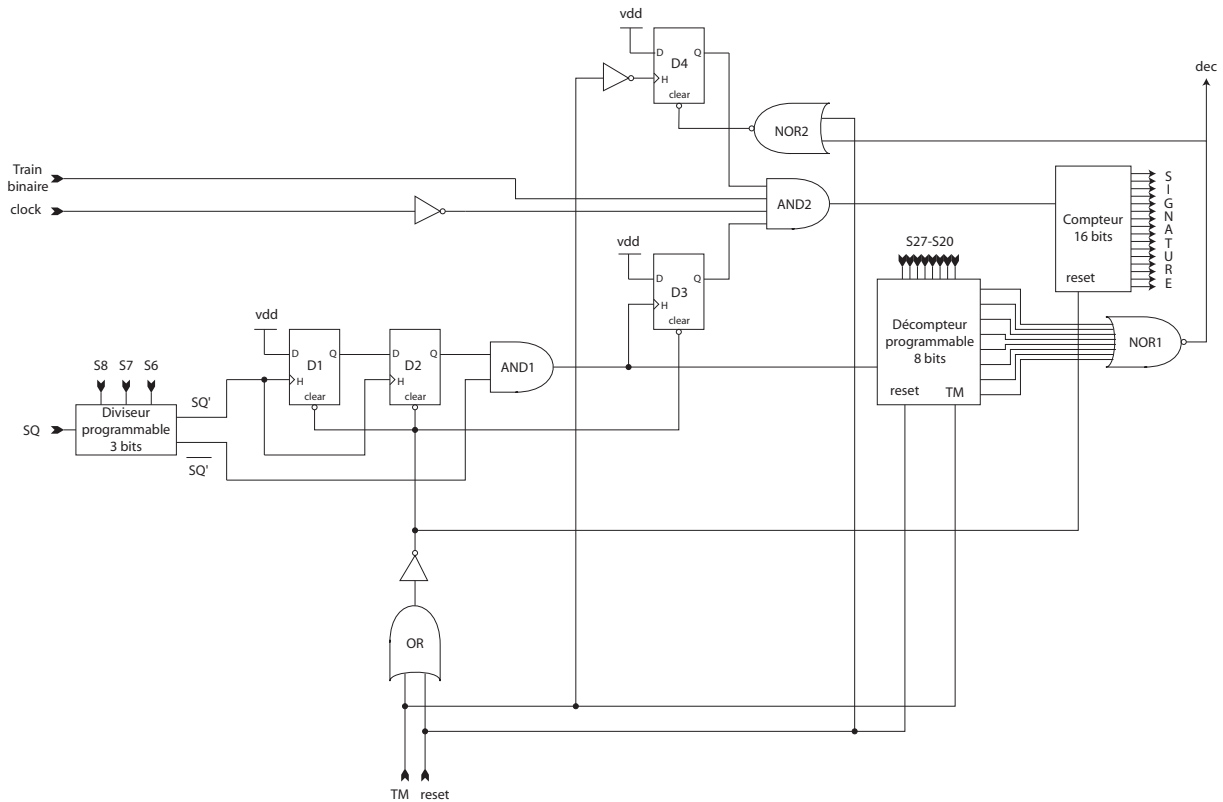


FIG. A.4 – Schéma détaillé du compteur programmable

binaires valent “1” et si les signaux servant à autoriser le comptage valent “1” également.

L'ensemble des composants sont remis à “0” si *reset* vaut “1”, le comptage ne peut être actif que si *TM* vaut “0” et la valeur initiale du compteur est chargée si *TM* vaut “1”.

A.2.2 Diviseur programmable

Le diviseur de fréquence programmable est décrit figure A.5. Les bits S_{20} à S_{27} codent la valeur du facteur de division, que nous appellerons *div*. Le principe est de compter le nombre de périodes de l'horloge (grâce au compteur à bascules en bas de la figure) et de le comparer (par les portes “OU exclusif” inversées) à la valeur du facteur de division. Quand ce nombre est atteint, la sortie du diviseur bascule grâce à la bascule T de sortie et le compteur est remis à zéro. L'élément de retard (constitué d'une série d'inverseurs) permet de s'assurer que les temps de maintien sont bien respectés avant la remise à zéro.

Si nous nous contentons de procéder ainsi, nous n'effectuons pas une division par *div*, mais par $2 \times div$: effectivement, la bascule T va basculer une première fois au bout de *div* périodes d'horloge, puis une seconde fois au bout de *div* périodes. La période du signal de sortie sera donc $2 \times div$ fois celle de l'horloge. Pour éviter cela, nous allons compter le nombre de demi-périodes d'horloge en créant un train d'impulsions de fréquence double de celle de l'horloge, en effectuant une opération de “OU exclusif” entre l'horloge et cette même horloge suffisamment décalée, comme l'illustre la figure A.6.

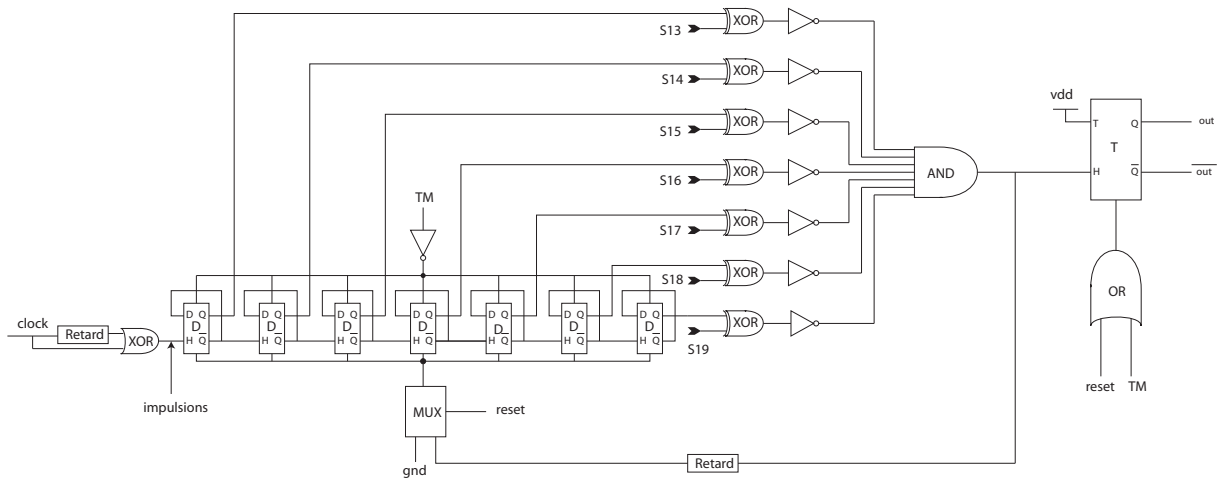


FIG. A.5 – Schéma du diviseur de fréquence programmable

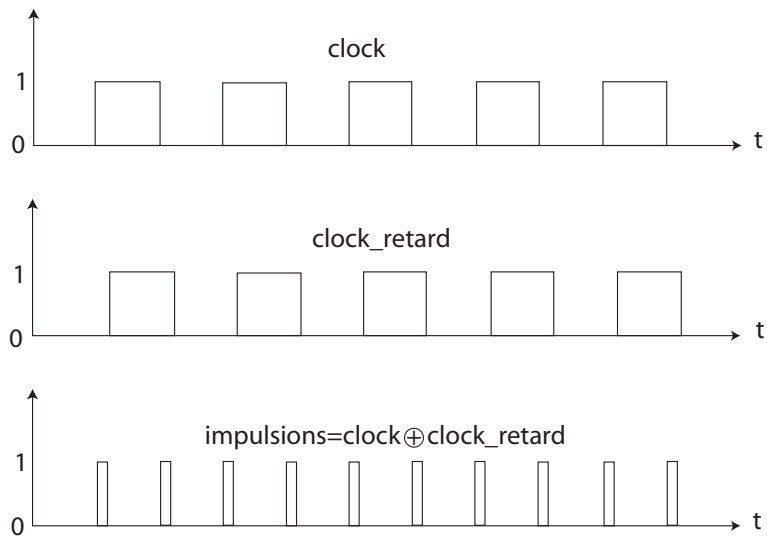


FIG. A.6 – Obtention d'un train d'impulsions de fréquence double de celle de l'horloge

Annexe B

Layouts de certaines parties des démonstrateurs

B.1 BIST complet

La figure B.1 montre le layout du circuit de BIST complet. On peut distinguer les parties numérique et analogique, les blocs constituant ces deux parties et avoir une idée de leur surface.

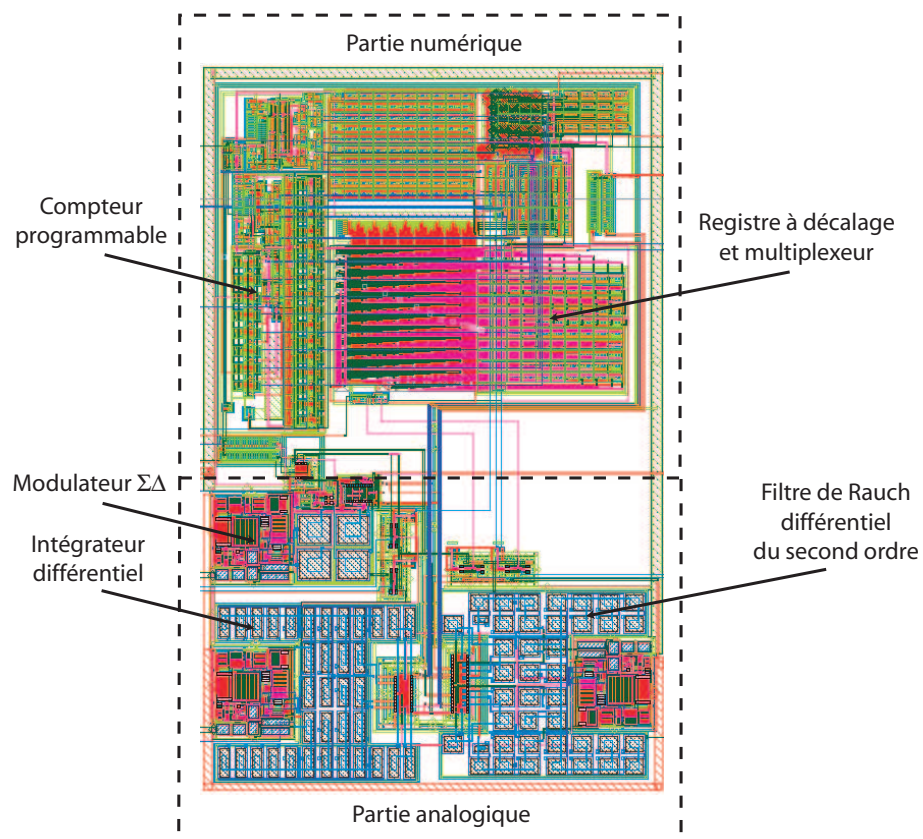


FIG. B.1 – *Layout du circuit de BIST complet*

B.2 Amplificateur différentiel

La figure B.2 montre le layout de l'amplificateur différentiel utilisé avec un agrandissement de la paire différentielle constituée de longs transistors en peignes interdigités.

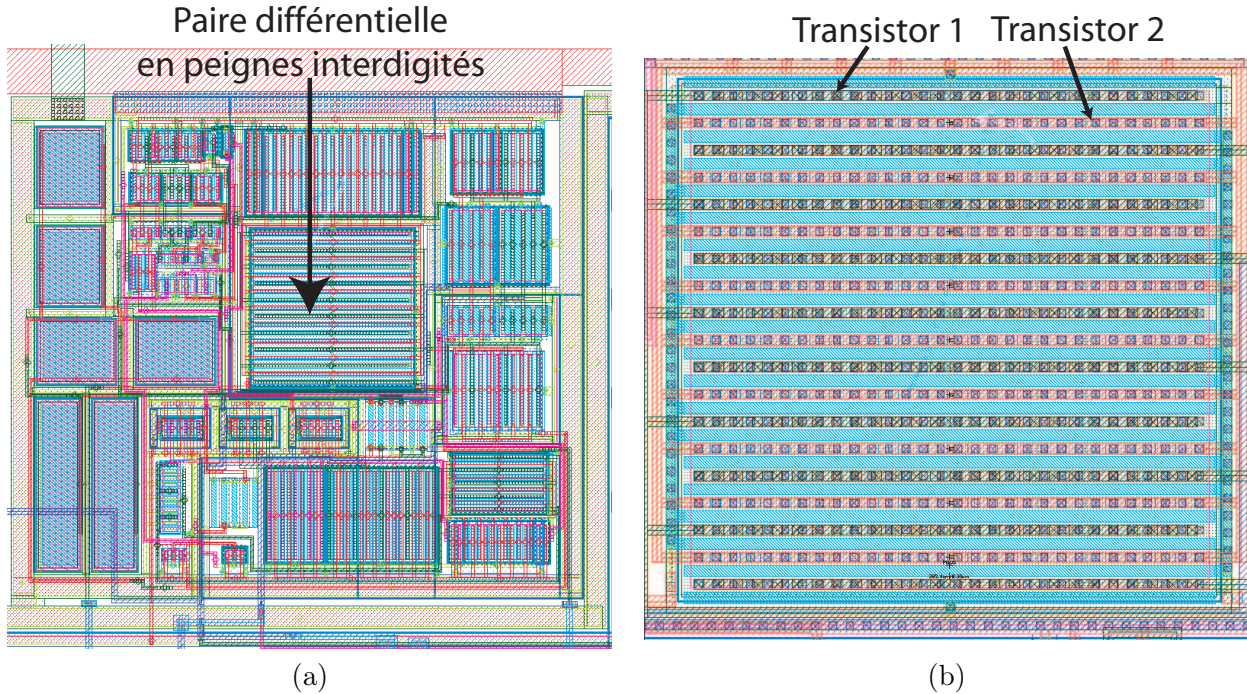


FIG. B.2 – *Layout de l'amplificateur différentiel utilisé : (a) l'amplificateur et (b) la paire différentielle en peignes interdigités*

B.3 Filtres à capacités commutées et isolation des parties numériques

La figure B.3 montre le layout de l'intégrateur utilisé pour le filtre différentiel. L'agrandissement montre les transistors utilisés pour les interrupteurs des capacités commutées, le caisson utilisé (polarisé à la tension intermédiaire numérique `vint_num` à 0,9 V) et les deux anneaux d'isolation entre ces transistors et les parties purement analogiques du filtre et du circuit en général.

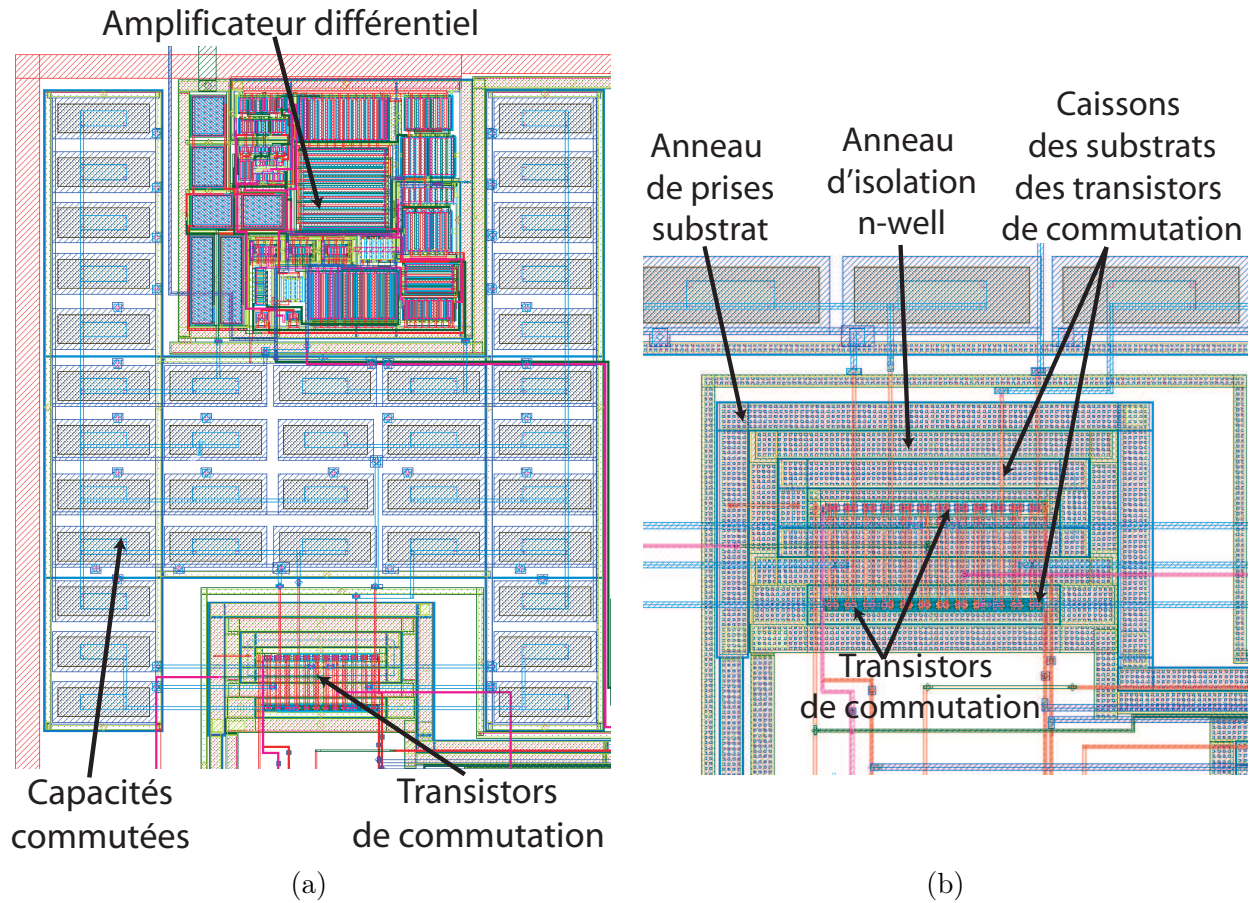


FIG. B.3 – Layout de l'intégrateur à capacités commutées : (a) l'intégrateur et (b) les transistors de commutation isolés par les anneaux d'isolation

Annexe C

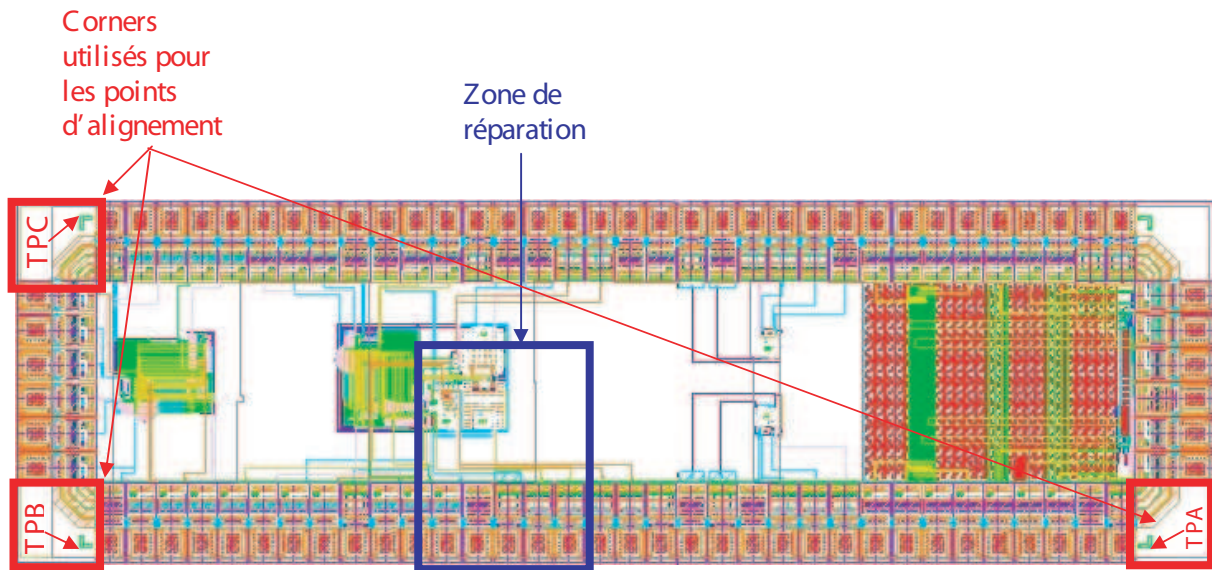
Réparation de la puce par la technique de FIB

C.1 Description de la panne

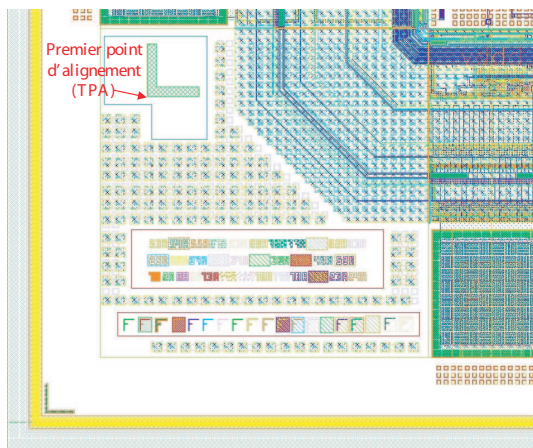
La première étape de la réparation a consisté à fournir à la société SERMA Technologies un dossier décrivant la réparation à effectuer. La figure C.1 montre certaines des figures utilisées dans ce dossier : (a) représente l'ensemble de la puce, avec la zone concernée par la réparation mise en évidence. (b) indique un des trois points d'alignement utilisés pour indiquer les coordonnées des points clés de la réparation. Les trois points d'alignement choisis correspondent à trois des quatre "corners" de la puce. (c) est un agrandissement de la zone de réparation, où l'on peut voir le filtre analogique et le plot que l'on cherche à connecter à sa sortie. (d) et (e) représentent respectivement la sortie du filtre et la connexion d'accès au plot. Il faudra donc couper (cut) cette connexion et tirer un strap entre celle-ci et la sortie du filtre.

C.2 Images de la réparation

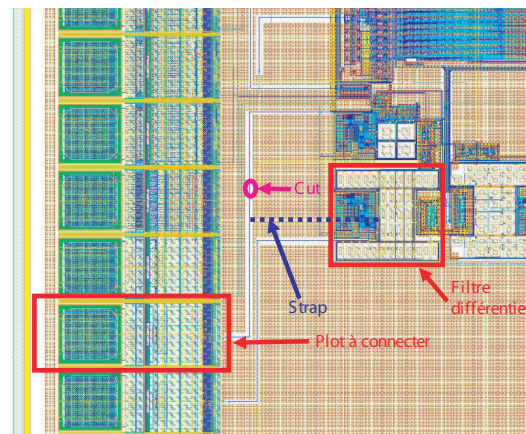
Des images décrivant la réparation sont données sur la figure C.2. (a) représente la zone de la connexion au plot qui va être coupée. Ici, la connexion a déjà été dénudée et on peut voir le niveau de métal correspondant. (b) montre la sortie du filtre analogique. (c) et (d) montrent l'accès au niveau de métal correspondant à cette sortie et le début du dépôt de tungstène. Enfin, (e) et (f) montrent respectivement la zone du strap (où sera déposé le tungstène) et le strap une fois terminé. On peut constater ici que la réparation est assez facile, car les points à connecter sont réalisés en métal 6 facilement accessible. La réparation aurait été très difficilement réalisable et en tout cas beaucoup plus chère si les niveaux de métaux utilisés avaient été plus profonds.



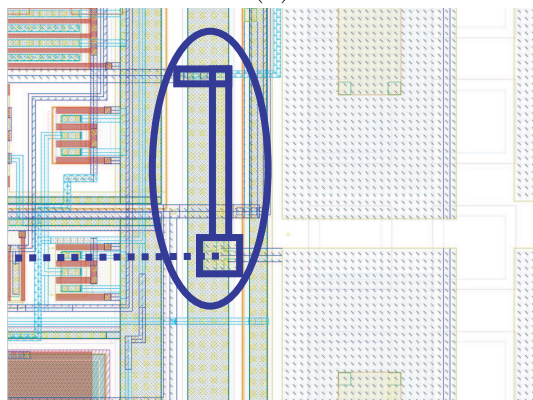
(a)



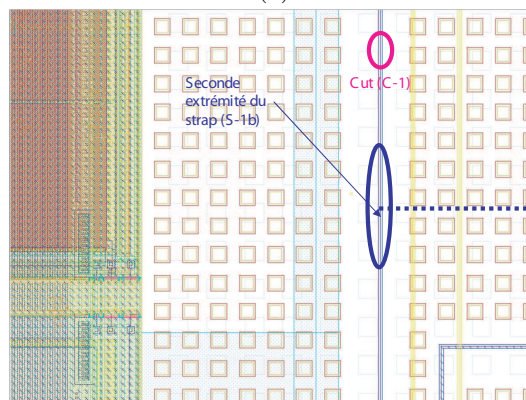
(b)



(c)

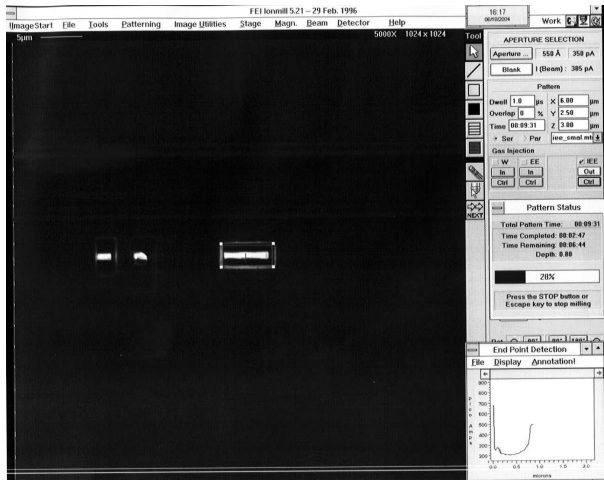


(d)

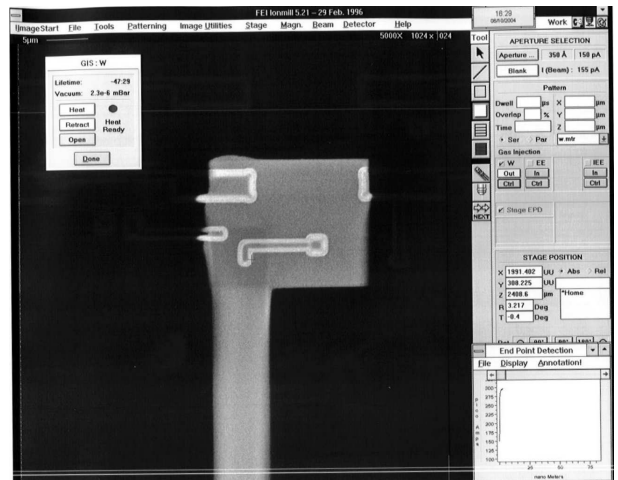


(e)

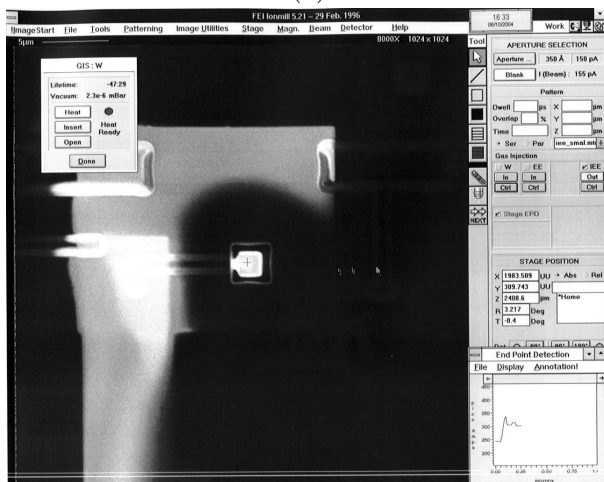
FIG. C.1 – Images expliquant la réparation à effectuer



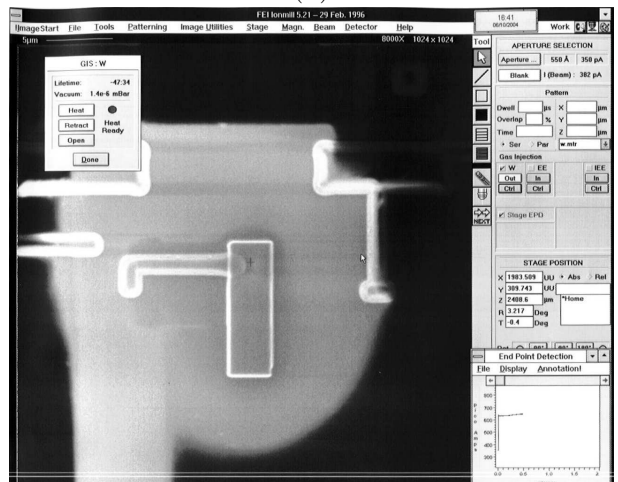
(a)



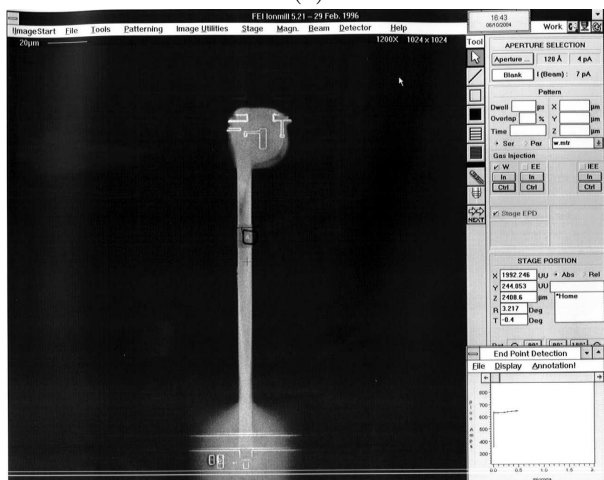
(b)



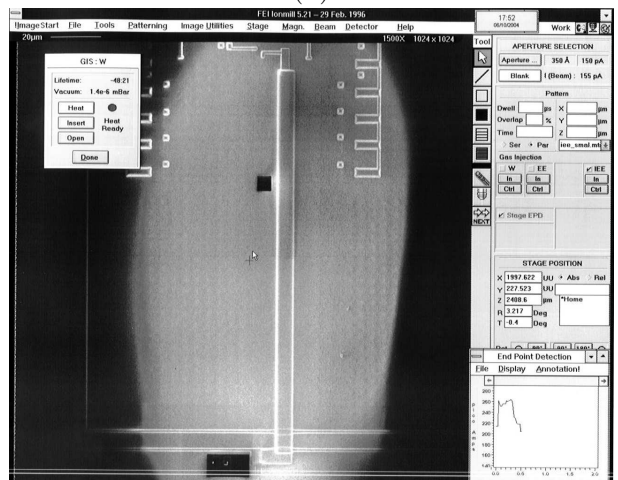
(c)



(d)



(e)



(f)

FIG. C.2 – Images illustrant les différentes étapes de la réparation au FIB

Index

- $\Sigma\Delta$, 31, 32, 34, 47, 48, 63
- ABM, 113, 114
- ADCBIST, 38, 41
- Algorithme adaptatif, 30
- APBIST, 38, 40
- ATAP, 112, 113
- ATE, 37, 41, 44, 106
- ATPG, 18

- BERT, 44
- BIST, 18, 21, 25, 27, 28, 30, 33, 36, 45–48, 61, 63, 64, 95, 101, 109, 112, 114–117, 119
- BISTMaxx, 37
- Boundary-scan, 18, 103
- Butterworth, 55
- BYPASS, 109

- CAN, 26, 27, 33, 34, 36–38, 41, 45, 63, 101, 120
- CAO, 48, 51, 72, 84, 90, 92
- Capacité de Miller, 121
- Capacités commutées, 53
- Caractérisation, 18
- CIME, 88
- CM, 121
- CMFB, 121, 122
- CMP, 81
- CNA, 26, 27, 30–33, 38, 41, 45
- Courbes de Lissajous, 30
- Couverture de fautes, 17, 24
- CUT, 25

- Défaillance, 17
- Défaut, 17
- DACBIST, 38, 41
- DBM, 113
- DDFST, 30
- DFT, 63
- DfT, 17, 18, 41, 42
- DNL, 27
- DR, 34, 35
- DRC, 87
- DSP, 26, 30, 36, 63
- DYNBIST, 38, 40

- ENSERG, 97
- EPT, 44
- Espace de paramètres, 70
- Espace des performances, 70
- Espace des signatures, 71
- EST, 44
- EXTEST, 112

- Faute, 17
- Fautes catastrophiques, 21, 22
- Fautes paramétriques, 21, 22
- FFT, 72
- FIB, 98, 131
- FPGA, 37, 90, 102

- Gigue, 39

- HABIST, 26, 37–39, 41
- HBIST, 26
- Histogramme, 26

- IDCODE, 111
- IEEE 1149.1, 18, 103, 104
- IEEE 1149.4, 19, 103, 112
- IMS, 88
- INL, 27
- INTEST, 112
- IP, 37, 41, 42, 44

- JTAG, 103

- LFSR, 26, 28, 29
- LIT, 28
- LVS, 87

- MADBIST, 26

MIM5, 85
MISR, 26
Monte Carlo, 23, 70

Niso, 86
Niveau de défaut, 24

OBIST, 27

PLL, 44, 45
PRELOAD, 111
PROBE, 114

Réponse impulsionnelle, 28
Rauch, 56
Registre d'instructions, 104, 106
Registre de données, 104, 107
Rendement, 23
Rendement de test, 23, 69
RF, 19
RTL, 42
RUNBIST, 112

SAMPLE, 111
Scan-path, 18, 103
SerDes, 44
SFDR, 32, 33, 51, 59, 61, 72, 76, 90, 119
SNR, 31–33
SoC, 19, 102
Sw-opamp, 29

TAP, 41, 104–107, 109, 112, 113
Taux de fausse acceptation, 69, 73, 76
Taux de faux rejet, 69, 73, 76
Taux de fuites, 69
TBIC, 113
TDM, 27
Test fonctionnel, 18, 24
Test pseudo-aléatoire, 28
Test structurel, 18, 24
THD, 19, 33, 51
TimeBIST, 41

USERCODE, 111

VCOBIST, 38, 39
Verilog, 37
VHDL, 37

WARGA, 51

Bibliographie

- [1] L. ANGHEL. *Test et Conception en vue du Test de Circuits Intégrés*. 2004. Copie de transparents de cours.
- [2] K. ARABI and B. KAMINSKA. Oscillation Test Strategy for Analogue and Mixed-Signal Integrated Circuits. In *IEEE VLSI Test Symposium*, pages 476–482, Princeton, USA, Apr.-May 1996.
- [3] K. ARABI and B. KAMINSKA. Testing Analog and Mixed-Signal Integrated Circuits Using Oscillation-Test Method. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 16(7):745–753, Jul. 1997.
- [4] K. ARABI and B. KAMINSKA. Design for Testability of Embedded Integrated Operational Amplifiers. *IEEE Journal of Solid-State Circuits*, 33(4):573–581, Apr. 1998.
- [5] K. ARABI and B. KAMINSKA. Oscillation-Test Methodology for Low-Cost Testing of Active Analog Filters. *IEEE Transactions on Instrumentation and Measurements*, 48(4):798–806, Aug. 1999.
- [6] F. AZAIS, S. BERNARD, Y. BERTRAND, X. MICHEL, and M. RENOVELL. A Low-Cost Adaptive Generator for Analog BIST applications. In *IEEE VLSI Test Symposium*, pages 266–271, Marina de Rey, USA, Mar.-Apr. 2001.
- [7] B. BABLOCK, P. ALLEN, and G. RICON-MORA. Designing 1 V op amps using standard digital CMOS technology. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, 45(7):769–780, July 1998.
- [8] S. BERNARD, F. AZAIS, Y. BERTRAND, and M. RENOVELL. On-Chip Generation of Ramp and Triangle-Wave Stimuli for ADC BIST. *Journal of Electronic Testing: Theory and Applications*, 19(4):469–479, Aug. 2003.
- [9] A. BOUNCEUR and S. MIR. Génération et optimisation de vecteurs de test pour des composants analogiques et mixtes. In *7emes Journées Nationales du Réseau Doctoral de Microélectronique*, pages 198–200, Marseille, France, Mai 2004.
- [10] A. BOUNCEUR, S. MIR, and E. SIMEU. Optimisation of digitally coded test vectors for mixed-signal components. In *19th Conference on Design of Circuits and Integrated Systems*, pages 895–900, Bordeaux, France, November 2004.
- [11] A. BRATT, A. RICHARDSON, R. HARVEY, and A. DOREY. A Design-For-Test Structure for Optimising Analogue and Mixed Signal IC Test. In *European Design and Test Conference*, pages 24–33, Paris, France, Mar. 1995.
- [12] A. BROSA and J. FIGUERAS. Digital Signature Proposal for Mixed-Signal Circuits. *Journal of Electronic Testing: Theory and Applications*, 17(5):385–393, Oct. 2001.
- [13] L. CARRO, E. COTA, M. LUBSZEWSKI, Y. BERTRAND, F. AZAIS, and M. RENOVELL. TI-BIST: a Temperature Independent Analog BIST for Switched-Capacitor Filters. In *IEEE Asian Test Symposium*, pages 78–83, Taipei, Taiwan, Dec. 2000.

- [14] L. CARRO and M. NEGREIROS. Efficient Analog Test Methodology Based on Adaptive Algorithms. In *ACM/IEEE Design Automation Conference*, pages 32–37, San Francisco, USA, Jun. 1998.
- [15] L. CASSOL, O. BETAT, and M. LUBASZEWSKI. The $\Sigma\Delta$ -BIST Method Applied to Analog Filters. *Journal of Electronic Testing: Theory and Applications*, 19(1):13–20, Feb. 2003.
- [16] A. CHATTERJEE. Concurrent Error Detection and Fault-Tolerance in Linear Analogue Circuits Using Continuous Checksums. *IEEE Transactions on VLSI Systems*, 1(2):138–150, Jun. 1993.
- [17] Y. COLLETTE and P. SIARRYS. *Optimisation Multiobjectif*. Edition Eyrolles, 2002.
- [18] F. CORSI, C. MARZOCCA, G. MATARRESE, A. BASCHIROTTI, and S. D’AMICO. Pseudo-random Sequence Based Tuning System for Continuous-time Filters. In *IEEE Design, Automation and Test in Europe Conference*, pages 94–99, Paris, France, Feb 2004.
- [19] E. COTA, L. CARRO, M. RENOVELL, M. LUBASZEWSKI, F. AZAIS, and Y. BERTRAND. Reuse of Existing Ressources for Analog BIST of a Switch Capacitor Filter. In *Design, Automation and Test in Europe Conference*, pages 226–230, Paris, France, Mar. 2000.
- [20] A. DHAYNI, S. MIR, and L. RUFER. MEMS Built-In-Self-Test using MLS. In *IEEE European Test Symposium*, pages 66–71, May 2004.
- [21] C. DIEDRICH. Conception de générateurs de tests pour l’autocontrôle de circuits analogiques et mixtes. Final year project report, Laboratoire TIMA-RMS, Sep. 2002.
- [22] B. DUFORT and G. ROBERTS. Optimized Periodic Sigma-Delta Bitstreams for Analogue Signal Generation. In *Midwest Symposium on Circuits and Systems*, pages 289–292, Sacramento, USA, Aug. 1997.
- [23] B. DUFORT and G. ROBERTS. Signal Generation Using Periodic Single and Multi-bit Sigma-Delta Modulated Streams. In *IEEE International Test Conference*, pages 396–405, Washington DC, USA, Nov. 1997.
- [24] B. DUFORT and G. ROBERTS. On-chip analog signal generation for mixed-signal Built-In-Self-Test. *IEEE Journal of Solid-State Circuits*, 34(3):318–330, 1999.
- [25] B. DUFORT and G. ROBERTS. Increasing the Performance of Arbitrary Waveform Generators Using Periodic Sigma-Delta Modulated Streams. *IEEE Transactions on Instrumentation and Measurements*, 49(1):188–199, Feb. 2000.
- [26] FLUENCE. *BISTMaxx Product Catalog*. 2002.
- [27] M. FONSESCA and P. FLEMING. Genetic Algorithms for Multiobjective Optimization : Formulation, Discussion and Generalization. In *Fifth International Conference on Genetic Algorithms*, pages 416–423, San Mateo, California, Janvier 1993.
- [28] A. FRISH and T. ALMY. HABIST : Histogram-based Analog Built-In-Self-Test. In *International Test Conference*, pages 760–767, Nov. 1997.
- [29] M. HAFED, N. ABASKHAROUN, and G. ROBERTS. A Stand-Alone Integrated Test Core for Time and Frequency Domain Measurements. In *IEEE International Test Conference*, pages 1031–1040, Atlantic City, USA, Oct. 2000.
- [30] M. HAFED and G. ROBERTS. A Stand-Alone Integrated Excitation/Extraction System for Analog BIST Applications. In *Custom Integrated Circuits Conference*, pages 83–86, Orlando, USA, May 2000.
- [31] A. HAJJAR and G. ROBERTS. A High Speed and Area Efficient On-Chip Analog Waveform Extractor. In *IEEE International Test Conference*, pages 688–697, Washington DC, USA, Oct. 1998.

-
- [32] A. HAJJAR and G. ROBERTS. A Multi-Pass A/D Conversion Technique for Extracting On-Chip Analog Signals. In *IEEE International Symposium on Circuits and Systems*, volume 2, pages 280–283, May-Jun. 1998.
- [33] R. HARJANI and B. VINNAKOTA. Analog Circuit Observer Blocks. In *IEEE VLSI Test Symposium*, pages 258–263, Princeton, USA, Apr. 1994.
- [34] A. HATZOPOULUS, S. SISKOS, and J. KONTOLEON. A Complete Scheme of Built-In-Self-Test Structure for Fault Diagnosis in Analog Circuits and Systems. *IEEE Transactions on Instrumentation and Measurements*, 42(3):689–694, Jun. 1993.
- [35] X. HAURIE and G. ROBERTS. Arbitrary-Precision Signal Generation for Bandlimited Mixed-Signal Testing. In *IEEE International Test Conference*, pages 78–86, Washington D.C, USA, Oct. 1995.
- [36] X. HAURIE and G. ROBERTS. Arbitrary-Precision Signal Generation for Mixed-Signal Built-In-Self Test. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, 45(11):1425–1432, Nov. 1998.
- [37] E. HAWRYSH and G. ROBERTS. An Integration of Memory-Based Analog Signal Generation into Current DFT Architectures. In *IEEE International Test Conference*, pages 528–537, Washington DC, USA, Oct. 1996.
- [38] E. HAWRYSH and G. ROBERTS. An Integration of Memory-Based Analog Signal Generation into Current DFT Architectures. *IEEE Transactions on Instrumentation and Measurements*, 47(3):748–759, Jun. 1998.
- [39] H. HONG, J. HUANG, K. CHENG, and C. WU. On-Chip Analog Response Extraction with 1-Bit $\Sigma\Delta$ Modulator. In *IEEE Asian Test Symposium*, pages 49–54, Guam, USA, Nov. 2002.
- [40] H. HONG, J. HUANG, K. CHENG, C. WU, and D. KWAI. Practical Considerations in Applying $\Sigma\Delta$ Modulation-Based Analog BIST to Sampled-Data Systems. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, 50(9):553–565, Sep. 2003.
- [41] F. HUANG, X. SHEN, X. ZOU, and C. CHEN. Analog Signal Generator for BIST of Wideband IF Signals Bandpass Sigma-Delta modulator. *Micro Electronics Journal*, 34:85–91, 2003.
- [42] J. HUANG and K. CHENG. A BIST Scheme for On-Chip ADC and DAC Testing. In *Design, Automation and Test in Europe Conference*, pages 216–220, Paris, France, Mar. 2000.
- [43] J. HUANG and K. CHENG. A Sigma-Delta Modulation Based BIST Scheme for Mixed-Signal Circuits. In *Asia and South Pacific Design Automation Conference*, pages 605–610, Yokohama, Japan, Jan. 2000.
- [44] G. HUERTAS, D. VÁZQUEZ, E. PERALÍAS, A. RUEDA, and J. HUERTAS. Practical Oscillation-Based Test of Integrated Filters. In *IEEE Design and Test of Computers*, pages 64–72, Nov.-Dec. 2002.
- [45] G. HUERTAS, D. VÁZQUEZ, A. RUEDA, and J. HUERTAS. Effective Oscillation-Based Test for application to a DFTM Filter Bank. In *IEEE VLSI Test Symposium*, pages 549–555, Piscataway, USA, Sep. 1999.
- [46] G. HUERTAS, D. VÁZQUEZ, A. RUEDA, and J. HUERTAS. Practical Oscillation-Based Test of Integrated Filters. In *IEEE Design and Test of Computers*, pages 64–72, Nov.-Dec. 2002.
- [47] J. HUERTAS, A. RUEDA, and D. VÁZQUEZ. Testable Switched-Capacitor Filters. *IEEE Journal of Solid-State Circuits*, 28(7):719–724, July 1993.
- [48] J. HUERTAS, D. VÁZQUEZ, and A. RUEDA. On-line Testing of Switched-Capacitor Filters. In *IEEE VLSI Test Symposium*, pages 102–106, New Jersey, USA, Apr. 1992.

- [49] D. JOHNS and K. MARTINS. *Analog Integrated Circuit Design*, pages 259–260 280–291. John Wiley & Sons Inc, 1996.
- [50] S. KHALED, B. KAMINSKA, B. COURTOIS, and M. LUBASZEWSKI. Frequency-Based BIST for analog circuit testing. In *IEEE VLSI Test Symposium*, pages 54–59, Princeton, USA, Apr.-May 1995.
- [51] V. KOLARIK, M. LUBASZEWSKI, and B. COURTOIS. Towards Self-Checking Mixed-Signal Integrated Circuits. In *European Solid-State Circuits Conference*, pages 202–205, Séville, Espagne, Sep. 1993.
- [52] V. KOLARIK, S. MIR, M. LUBASZEWSKI, and B. COURTOIS. Analog Checkers with Absolute and Relative Tolerances. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14(5):607–612, May 1995.
- [53] K. LOFSTROM. Early Capture for Boundary Scan Timing Measurements. In *IEEE International Test Conference*, pages 417–422, Washington DC, USA, Oct. 1996.
- [54] A. LU and G. ROBERTS. An Analog Multitone Signal Generator for Built-In-Self-Test Applications. In *IEEE International Test Conference*, pages 650–659, Washington D.C, USA, Oct. 1994.
- [55] A. LU and G. ROBERTS. An Oversampling-Based Analog Multitone Signal Generator. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, 45(3):391–394, Mar. 1998.
- [56] A. LU, G. ROBERTS, and D. JOHNS. A High-Quality Analog Oscillator Using Oversampling D/A Conversion Techniques. *IEEE International Symposium on Circuits and Systems*, 2:1298–1301, May. 1993.
- [57] A. LU, G. ROBERTS, and D. JOHNS. A High-Quality Analog Oscillator Using Oversampling D/A Conversion Techniques. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, 41(7):437–444, Jul. 1994.
- [58] M. LUBASZEWSKI, S. MIR, V. KOLARIK, C.NIELSEN, and B. COURTOIS. Design of Self-Checking Fully Differential Circuits and Boards. *IEEE Transactions on VLSI Systems*, 8(2):113–128, Apr. 2000.
- [59] M. LUBASZEWSKI, S. MIR, A. RUEDA, and J. HUERTAS. Concurrent Error Detection in Analog and Mixed-Signal Integrated Circuits. *IEEE Midwest Symposium on Circuits and Systems*, 2(3):1151–1156, Aug. 1995.
- [60] M. LUBASZEWSKI, S. MIR, A. RUEDA, and J. HUERTAS. ABILBO: Analog Built-in Block Observer. In *IEEE International Conference on Computer-Aided Design*, pages 600–603, San Jose, USA, Nov. 1996.
- [61] M. LUBASZEWSKI, M. RENOVELL, S. MIR, F. AZAIS, and Y. BERTRAND. A Built-In Multi-Mode Stimuli Generator for Analogue and Mixed-Signal Testing. In *IEEE Brazilian Symposium on Integrated Circuit Design*, pages 175–178, Buzios, Brazil, Sep.-Oct. 1998.
- [62] M. MAHONEY. Tutorial: DSP-based Testing of Analog and Mixed-Signal Circuits. In *IEEE Computer Society Press*, Washington DC, USA, 1987.
- [63] C. MARZOCCA and F. CORSI. Mixed-Signal Circuit Classification in a Pseudo-Random Testing Scheme. *Journal of Electronic Testing: Theorie and Applications*, 18(3):333–342, Jun. 2002.
- [64] H. MENDONÇA, J. MACHADO DA SILVA, and J. MATOS. An algorithm for ADC’s harmonic estimation suitable for preliminary BIST operations. In *IEEE European Test Workshop*, pages 211–212, Maastricht, The Netherlands, May 2003.

- [65] S. MIR, C. DIEDRICH, C. ROMAN, and C. DOMINGUES. On-chip test signal generation for acoustic and ultrasound microelectronic interfaces. In *8th IEEE International Mixed Signal Testing Workshop*, pages 137–144, Montreux, Switzerland, June 2002.
- [66] S. MIR, M. LUBASZEWSKI, and B. COURTOIS. Unified Built-In-Self-Test for Fully Differential Analog Circuits. *Journal of Electronic Testing: Theory and Applications*, 9(1-2):135–151, Aug.-Oct. 1996.
- [67] S. MIR, M. LUBASZEWSKI, V. KOLARIK, and B. COURTOIS. Analogue On-line/Off-line Test Unification for Fully Differential Circuits. In *IEEE International Mixed-Signal Testing Workshop*, pages 56–61, Grenoble, France, Jun. 1995.
- [68] S. MIR, M. LUBASZEWSKI, V. KOLARIK, and B. COURTOIS. Programmable self-Checking Analogue Oscillators. In *IEEE International On-Line Testing Workshop*, pages 30–33, Nice, France, Jul. 1995.
- [69] S. MIR, M. LUBASZEWSKI, V. KOLARIK, and B. COURTOIS. Fault-Based Testing and Diagnosis of Balanced Filters. *Analog Integrated Circuits and Signal Processing*, 11(1):5–19, Sep. 1996.
- [70] S. MIR, L. ROLÍNDEZ, C. DOMINGUES, and L. RUFER. An implementation of memory-based on-chip analogue test signal generation. In *Asia South Pacific Design Automation Conference*, pages 663–668, January 2003.
- [71] N. NAGI, A. CHATTERJEE, and J. ABRAHAM. A Signature Analyser for Analog and Mixed-Signal Circuits. In *International Conference on Computer Design: VLSI in Computer and Processors*, pages 284–287, Cambridge, USA, Oct. 1994.
- [72] N. NAGI, A. CHATTERJEE, H. YOON, and J. ABRAHAM. Signature Analysis for Analog and Mixed-Signal Circuits TEST Response Compaction. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 17(6):540–546, Jun. 1998.
- [73] M. NEGREIROS, L. CARRO, and A. SUSIN. A Statistical Sampler for a New On-Line Analog Test Method. *Journal of Electronic Testing: Theory and Applications*, 19(5):585–595, Oct. 2003.
- [74] M. NEGREIROS, L. CARRO, and A. SUSIN. Ultra Low-Cost Analog BIST Using Spectral Analysis. In *IEEE VLSI Test Symposium*, pages 77–82, Napa Valley, USA, Apr.-May 2003.
- [75] M. OHLETZ. Hybrid Built-In-Self-Test (HBIST) for mixed analog/digital integrated circuits. In *2nd European Test Conference IEEE*, pages 307–316, 1991.
- [76] C.-K. ONG, K.-T. CHENG, and L.-C. WANG. Delta-sigma modulator based mixed-signal BIST architecture for SoC. In *Asia South Pacific Design Automation Conference*, pages 669–674, January 2003.
- [77] C. PAN and K. CHENG. Implicit Functional Testing for Analog Circuits. In *IEEE VLSI Test Symposium*, pages 489–494, Apr.-May 1996.
- [78] C. PAN and K. CHENG. Pseudorandom Testing for Mixed-Signal Circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 16(10):1173–1185, Oct. 1997.
- [79] K. PARKER. *The Boundary-Scan Handbook*. Kluwer Academic Publishers, 2003.
- [80] K. PARTHASARATHY, T. KUYEL, D. PRICE, L. JIN, D. CHEN, and R. GEIGER. BIST and Production Testing of ADC's Using Imprecise Stimulus. *ACM Transactions on Design Automation of Electronic Systems*, 8(4):522–545, Oct. 2003.
- [81] G. PRENAT, S. MIR, D. VÁZQUEZ, and L. ROLÍNDEZ. A Low-Cost Digital Frequency Testing Approach for Mixed-Signal Devices using $\Sigma\Delta$ Modulation. In *International Mixed Signal Testing Workshop*, pages 107–115, Portland, USA, Jun. 2004.

- [82] B. PROVOST and E. SANCHEZ-SINENCIO. Auto-Calibrating Analog Timer for On-Chip Testing. In *IEEE International Test Conference*, pages 541–548, Atlantic City, USA, Sep. 1999.
- [83] B. PROVOST and E. SANCHEZ-SINENCIO. On-Chip Ramp Generators for Mixed-Signal BIST and ADC Self-Test. *IEEE Journal of Solid-State Circuits*, 38(2):263–273, Feb. 2003.
- [84] M. RENOVELL, F. AZAÏS, and Y. BERTRAND. On-Chip Analog Output Response Compaction. In *European Design and Test Conference*, pages 568–572, Paris, France, Mar. 1997.
- [85] M. RENOVELL, M. LUBSZEWSKI, F. AZAÏS, and Y. BERTRAND. A Multi-mode Signature Analyser for Analog and Mixed Circuits. In *IFIP International Conference on Very Large Scale Integration*, pages 65–76, Gramado, Brazil, Aug. 1997.
- [86] G. ROBERTS. Metrics, Technics and Recent Developpements in Mixed-Signal Testing. In *IEEE/ACM International Conference on Computer Aided Design*, pages 514–521, Nov. 1996.
- [87] G. ROBERTS. DFT Technics for Mixed-Signal Integrated Circuits. In *Circuits and Systems In The Information Age*, pages 251–271. IEEE Press, Jun. 1997.
- [88] G. ROBERTS and B. DUFORT. Making Complex Mixed-Signal Telecommunications Integrated Circuits Testable. In *IEEE Communication Magazine*, pages 90–96, Jun. 1999.
- [89] J. ROH and J. ABRAHAM. Subband Filtering Scheme for Analog and Mixed-Signal Circuit Testing. In *IEEE International Test Conference*, pages 221–229, Atlantic City, USA, Sep. 1999.
- [90] J. ROH and J. ABRAHAM. A Comprehensive TDM Comparator Scheme for Effective Analysis of Oscillation-Based Test. In *IEEE VLSI Test Symposium*, pages 143–148, Montreal, Canada, Apr.-May 2000.
- [91] J. ROH and J. ABRAHAM. Subband Filtering for Time and Frequency Analysis of Mixed-Signal Circuit Testing. *IEEE Transactions on Instrumentation and Measurements*, 53(2):602–611, Apr. 2004.
- [92] L. ROLÍNDEZ. Conception de circuits mixtes analogique-numérique avec éléments de test intégrés dans la puce, en technologie CMOS submicronique. Final year project report, Laboratoire TIMA et Centro Politécnico Superior de Zaragoza, Apr. 2002.
- [93] L. ROLÍNDEZ. State-Of-The-Art : Analog and Mixed-Signal BIST. Rapport interne TIMA-STMicroelectronics, Oct. 2004.
- [94] L. ROLÍNDEZ and S. MIR. *TIMA - ANNUAL REPORT 2004*, chapter Bulk-driven CMFB circuit for a fully differential amplifier, page 55.
- [95] L. ROLÍNDEZ, S. MIR, and A. BOUNCEUR. A digital BIST for a 16-bit audio Sigma Delta analogue-to-digital converter. In *11th IEEE International Mixed-Signals Testing Workshop*, pages 45–52, Cannes, France, May 2005.
- [96] L. ROLÍNDEZ, S. MIR, G. PRENAT, and A. BOUNCEUR. A 0.18 μ m CMOS implementation of on-chip analogue test signal generation from digital test patterns. In *Design and Test Conference in Europe*, pages 704–705, February 2004.
- [97] C. ROMAN. Building CAT tools for System-on-Chip. Final year project report, Laboratoire TIMA-RMS, June 2002.
- [98] L. RUFER, S. MIR, E. SIMIEU, and C. DOMINGUEZ. On-chip Testing of MEMS using pseudo-random test sequences. In *IEEE Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS*, pages 50–55, Cannes, France, May 2003.
- [99] M. SLAMANI and B. KAMINSKA. T-BIST: A Built-In-Self-Test for Analog Circuits based on Parameter Translation. In *IEEE Asian Test Symposium*, pages 172–177, Beijing, China, Nov. 1993.

-
- [100] S. SUNTER and N. NAGI. Test Metrics for Analog Parametric Faults. In *IEEE VLSI Test Symposium*, page 226, 1999.
- [101] S. SUNTER and A. ROY. BIST for Phase-Locked Loops in Digital Applications. In *IEEE International Test Conference*, pages 532–540, Atlantic City, USA, Sep. 1999.
- [102] S. SUNTER, A. ROY, and J. CÔTÉ. An Automated, Complete, Structural Test Solution for SerDes. In *IEEE International Test Conference*, pages 95–104, Charlotte, North Carolina, USA, Oct. 2004.
- [103] J. TOFTE, C. ONG, J. HUANG, and K. CHENG. Characterization of a Pseudo-Random Testing Technique for Analogue and Mixed-signal Built-In-Self-Test. In *IEEE VLSI Test Symposium*, pages 237–246, May 2000.
- [104] M. TONER and G. ROBERTS. A frequency response, harmonic distortion and intermodulation distortion test for BIST of a Sigma-Delta ADC. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, 43(8):613–613, August 1996.
- [105] J. TURINO. Built-In-Self-Test Streamlines Testing of Mixed-Signal SoCs. *Electronic Design*, pages 67–74, July 9 2001.
- [106] J. TURINO. Reducing Mixed-Signal SoC Test Costs Using BIST. In *IEEE International Mixed-Signal Test Workshop*, pages 103–107, June 2001.
- [107] P. VARIYAM and A. CHATERJEE. Digital-Compatible BIST for Analog Circuits Using Transient Response Sampling. *IEEE Design and Test of Computers*, 17(3):106–115, Jul.-Sep 2000.
- [108] P. VARIYAM, A. CHATERJEE, and N. NAGI. Low-Cost and Efficient Digital-Compatible BIST for Analog Circuits Using Pulse Response Sampling. In *IEEE VLSI Test Symposium*, pages 261–266, Monterey, USA, Apr.-May 1997.
- [109] D. VÁZQUEZ, G. HUERTAS, G. LEGER, E. PERALÍAS, A. RUEDA, and J. HUERTAS. On-Chip Evaluation of Oscillation-Based-Test Output Signals for Switched-Capacitor Circuits. *Analog Integrated Circuits and Signal Processing*, 33(2):201–211, Nov. 2002.
- [110] D. VÁZQUEZ, G. HUERTAS, G. LEGER, A. RUEDA, and J. HUERTAS. Practical Solution for the Application of the Oscillation-Based-Test: Start-Up and On-Chip Evaluation. In *IEEE VLSI Test Symposium*, pages 433–438, Monterey, USA, Apr.-May 2002.
- [111] D. VÁZQUEZ, J. HUERTAS, and A. RUEDA. Reducing the Impact of DFT on the performance of Analog Integrated Circuits: Improved SW-OPAMP Design. In *IEEE VLSI Test Symposium*, pages 42–47, Princeton, USA, May 1996.
- [112] D. VÁZQUEZ, G. LEGER, G. HUERTAS, A. RUEDA, and J. HUERTAS. A Method for Parameter Extraction of Analog Sine-wave Signals for Mixed-Signal Built-In-Self-Test. In *IEEE Design And Test Conference in Europe*, pages 298–303, Paris, France, February 2004.
- [113] D. VÁZQUEZ, A. RUEDA, and J. HUERTAS. A New Strategie for Testing Analog Filters. In *IEEE VLSI Test Symposium*, pages 36–41, Cherry Hill, USA, Apr. 1994.
- [114] D. VÁZQUEZ, A. RUEDA, and J. HUERTAS. A Solution for On-line Test of Analog Ladder Filters. In *IEEE VLSI Test Symposium*, pages 48–53, Princeton, USA, Apr.-May 1995.
- [115] D. VÁZQUEZ, A. RUEDA, J. HUERTAS, and E. PERALÍAS. A High-Q Bandpass Fully Differential SC Filter with Enhanced Testability. *IEEE Journal of Solid-State Circuits*, 33(7):976–986, July 1998.
- [116] B. VEILLETTE and G. ROBERTS. High Frequency Sinusoidal Generation Using Sigma-Delta Modulation Techniques. In *IEEE International Symposium on Circuits and Systems*, pages 637–640, May 1995.

- [117] B. VEILLETTE and G. ROBERTS. Spectral-Based Built-In-Self-Test Methods for Mixed-Signal integrated Circuits. In *Analog and Mixed Signal Test*, pages 153–184. B. Vinnakota, Prentice-Hall Inc., 1998.
- [118] B. VINNAKOTA and R. HARJANI. The Design of Analog Self-Checking Circuits. In *IEEE International Conference on VLSI Design*, pages 67–70, Calcutta, India, Jan. 1994.
- [119] R. WILLIAMS and C. HAWKINS. Errors in testing. In *IEEE International Test Conference*, pages 1018–1027, January 1990.

TITRE

CONCEPTION D'UNE ARCHITECTURE DE BIST ANALOGIQUE ET MIXTE
PROGRAMMABLE EN TECHNOLOGIE CMOS TRÈS SUBMICRONIQUE

RESUME

Ce mémoire présente une technique de BIST dont l'interface est totalement numérique, pour le test fréquentiel de circuits analogiques et mixtes. L'objectif de cette approche est de faciliter les techniques de test à bas coût des Systèmes sur Puce, rendant le test des blocs mixtes compatibles avec l'utilisation de testeurs numériques. La génération de signal de test analogique est réalisée sur la puce elle-même par un filtrage passe-bas d'un train binaire encodé par un modulateur $\Sigma\Delta$. L'analyse harmonique de la réponse analogique est également réalisée sur la puce en utilisant une modulation par un signal carré et une modulation $\Sigma\Delta$. La génération de signal analogique et l'analyse de la réponse de test étant programmables numériquement sur la puce, la compatibilité avec un testeur numérique à faible coût est assurée. L'optimisation des signatures de test est discutée en détail pour trouver un compromis entre temps et qualité du test.

Mots clés : BIST, test bas coût, conception analogique, CMOS, technologie très submicronique

TITLE

DESIGN OF A PROGRAMMABLE ANALOG AND MIXED-SIGNAL BIST ARCHITECTURE
IN DEEP SUBMICRON TECHNOLOGY

ABSTRACT

This report presents a BIST technique for harmonic testing of Analogue and Mixed-Signal (AMS) circuits. The interface of the BIST is fully digital. This approach is aimed at facilitating low-cost test techniques for System-on-Chip (SoC) devices, rendering the test of mixed-signal cores compatible with the use of a low-cost digital tester. Analogue test signal generation is performed on-chip by low pass filtering a $\Sigma\Delta$ encoded bit-stream. Analogue harmonic test response analysis is also performed on-chip using square wave modulation and $\Sigma\Delta$ modulation. Since both analogue signal generation and test response analysis are digitally programmable on-chip, compatibility with a low-cost digital tester is ensured. Optimisation of test signatures is discussed in detail as a trade-off between test time and test quality.

Keywords : BIST, low-cost testing, analog design, CMOS, deep submicron technology

Thèse préparée au laboratoire TIMA (Technique de l'Informatique et de la Microélectronique pour l'Architecture des ordinateurs), INPG, 46 avenue Félix Viallet, 38031 Grenoble, France.

ISBN (Paperback) : 2-84813-075-X

ISBNE (Electronic Format) : 2-84813-075-X