



HAL
open science

Méthode de modélisation et de raffinement pour les systèmes hétérogènes. Illustration avec le langage System C-AMS

Franck Paugnat

► **To cite this version:**

Franck Paugnat. Méthode de modélisation et de raffinement pour les systèmes hétérogènes. Illustration avec le langage System C-AMS. Autre. Université de Grenoble, 2012. Français. NNT : 2012GRENT041 . tel-00800204

HAL Id: tel-00800204

<https://theses.hal.science/tel-00800204v1>

Submitted on 13 Mar 2013

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.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Nanoélectronique et nanotechnologies**

Arrêté ministériel : 7 aout 2006

Présentée par :

Franck PAUGNAT

Thèse dirigée par **Laurent FESQUET** et
codirigée par **Katell MORIN-ALLORY** et **Emmanuel SIMEU**

préparée au sein du **Laboratoire TIMA – Techniques de l'Informatique
et de la Microélectronique pour l'Architecture des systèmes intégrés**
dans l'**École Doctorale EEATS – Électronique, Électrotechnique,
Automatique et Traitement du Signal**

Méthode de modélisation et de raffinement pour les systèmes hétérogènes

Illustration avec le langage SystemC-AMS

Thèse soutenue publiquement le **25 octobre 2012**,
devant le jury composé de :

Bruno ROUZEYRE

Professeur des Universités, Université de Montpellier 2, Président

Nathalie JULIEN

Professeur des Universités, Université de Bretagne-Sud, Rapportrice

Ian O'CONNOR

Professeur des Universités, École Centrale de Lyon, Rapporteur

Laurent FESQUET

Maitre de conférences, Institut Polytechnique de Grenoble, Directeur de thèse

Katell MORIN-ALLORY

Maitre de conférences, Institut Polytechnique de Grenoble, Co-encadrante

Emmanuel SIMEU

Maitre de conférences, Université Joseph Fourier, Co-encadrant

Hela BOUSSETTA

Ingénieure en modélisation, Docea Power, Invité





Franck Paugnat
Laboratoire TIMA
franck.paugnat@imag.fr

Thèse de doctorat
Spécialité : nanoélectronique & nanotechnologies

Méthode de modélisation et de raffinement pour les systèmes hétérogènes

Illustration avec le langage SystemC-AMS

Date de révision du manuscrit : 12 novembre 2012

A propos de l'orthographe

Le texte de ce mémoire a été rédigé en appliquant la réforme de l'orthographe de 1990 qui simplifie l'usage de l'accent circonflexe en autorisant sa suppression sur le 'u' et le 'i' tant qu'il n'y a pas de risque de confusion (cas de « sur » et de « sûr » ou de « mur » et de « mûr »).

Exemple : « Connaitre les couts de développement pour mieux les maitriser »

La réforme de l'orthographe de 1990 simplifie également l'usage du tiret et est favorable à la soudure pour les néologismes tant qu'il n'y a pas d'hésitation sur la prononciation, d'où : « semiconducteur », « semiformelle », « nanoélectronique », « cosimulation », etc. mais « co-encadrant »

Remerciements

J'adresse mes plus sincères remerciements à tous les membres du jury pour le temps consacré à la lecture de ce mémoire et pour s'être déplacés afin d'en apprécier la soutenance. Je tiens tout particulièrement à saluer Nathalie Julien et Ian O'Connor pour avoir accepté la tâche de rapporteur.

Quand on soutient une thèse à 43 ans c'est que son cursus universitaire n'a pas été direct. C'est une carrière de technicien supérieur qui ne progressait plus qui m'a motivé à reprendre les études afin de devenir ingénieur en électronique, c'était il y dix ans déjà. Mes premières pensées vont à mes camarades du CNAM et à cette établissement aujourd'hui bien malmené à Grenoble. Sans le CNAM, je ne serais pas devenu ingénieur et je ne soutiendrais pas aujourd'hui cette thèse. En effet, ce sont les cours de spécialité du cycle d'Ingénieur en électronique qui m'ont amenés à l'ENSERG, devenue Phelma depuis. Parmi ces cours, il y avait celui dispensé par Laurent Fesquet et c'est grâce à lui que je suis arrivé au Laboratoire TIMA, d'abord comme ingénieur stagiaire dans l'équipe CIS, dirigée alors par Marc Renaudin, puis comme Ingénieur de Recherche. Ensuite, répondant à une proposition de Laurent Fesquet et d'Emmanuel Simeu, je suis devenu doctorant et Laurent est devenu mon directeur de thèse. À ce titre, il m'a guidé et conseillé tout au long de ma thèse. J'adresse donc à un grand merci à Laurent pour ces années de collaboration qui se poursuivront peut-être encore après ma thèse.

Aux côtés de Laurent Fesquet, ma thèse a été encadrée par Katell-Morin Allory et Emmanuel Simeu. Katell m'a beaucoup aidé à structurer ma démarche en me conseillant et me corrigeant jusqu'à obtenir des constructions correctes et à des démonstrations rigoureuses, à l'enchaînement parfaitement logique. Merci donc Katell pour ton aide précieuse tant sur le plan de la démarche que sur la rédaction du mémoire ou des articles. Si mes échanges avec Emmanuel ont été plus ponctuels, je n'en ai pas moins apprécié ses encouragements ainsi que ses conseils quant aux précautions à prendre lors de la conception d'un modèle informatique d'un circuit analogique.

Le modèle d'amplificateur opérationnel simplifié que je présente dans cette thèse s'est révélé être un travail considérable et je n'aurai pu obtenir un résultat simple et efficace sans les éclairages que m'ont donnés Gilles Sicard et Olivier Rossetto quant à la conception des circuits analogiques et aux structures internes des amplificateurs ainsi qu'aux phénomènes les plus essentiels à répercuter aux niveaux d'abstraction supérieurs. Je les remercie pour le temps qu'ils ont accepté de consacrer à mes questions.

Je tiens à exprimer ma considération à Laurent Bousquet avec qui j'ai réussi ma toute première publication. Merci pour cette collaboration et pour toute les réflexions techniques que nous avons eues par la suite, nos sujets de recherche étant assez proches, mais également pour toutes ces discussions informelles des pauses café.

C'est aussi grâce à la bonne ambiance du laboratoire que je suis parvenu à mener à bien cette thèse. Les moments de doutes, les interrogations et les coups de déprime ont bien souvent trouvé des réponses grâce aux échanges avec les uns ou les autres, qu'il s'agisse d'obtenir une information technique ou de refaire le monde. En premier lieu, chercher à expliquer son problème permet dans bien des cas de mieux le cerner et ainsi trouver les failles de raisonnement. Ensuite, parler d'autre chose et lever le nez de son sujet de recherche, ne serait-ce que cinq minutes, est un bon moyen de ne pas verser dans l'obsession et d'éviter de se couper complètement du monde. C'est pourquoi j'adresse mes meilleures salutations tout d'abord à mes collègues du groupe de recherche CIS, chercheurs ou ingénieurs : Alex – aux blagues potaches dont je n'apprécie pas toujours le goût mais qui empêchent de s'encrouter dans la routine, Tugdual, Hawraa, Ahmed, Taha, Oussama, Hakim, Florent, Jérémie, Greg et les autres. Alice, j'espère que tu me pardonneras d'avoir eu à supporter certains de mes éclats lorsque j'étais confronté à quelque dysfonctionnement du matériel ou des logiciels alors que tu n'y étais pour rien, simplement parce que ton bureau était en face du mien. Sache que j'ai beaucoup apprécié nos discussions de cinéphiles ou alors à propos de notre société, notre époque, de l'humanité toute entière, discussions parfois partagées avec Tugdual, Nicolas ou quelques autres. Ces bonnes relations ont aussi existé avec les membres des autres groupes de recherche. Parmi ceux-ci, je tiens à exprimer ma gratitude envers Luca qui m'a beaucoup aidé à progresser dans la maîtrise de C++. Je n'oublie pas Laurence Pierre pour son apport sur ces questions de langage informatique et pour ses conseils quant à la façon de présenter une démonstration mathématique. Merci aussi à Adrien pour ses éclairages au sujet des circuits numériques de grande complexité. La bonne ambiance d'un laboratoire passe aussi par la bonne marche de ses services, c'est pourquoi je veux rendre ici hommage au personnel administratif, tout spécialement Sophie, Youness, Laurence et Anne-Laure, et à celui du service informatique, Nicolas, Frédéric et Ahmed, pour leur compétence et leur disponibilité.

Les excellentes relations avec le CIME Nanotech ont également comptées dans le bon déroulement de cette thèse. Je souhaite souligner tout spécialement la grande disponibilité d'Alexandre Chagoya et de Robin Roland.

Le cours d'une thèse est ponctué de rencontres notamment dans les conférences où se croisent des chercheurs du monde entier. La rencontre de Torsten Mähne a été particulièrement marquante pour moi. Torsten m'a beaucoup appris en ce qui concerne la modélisation système et la connaissance approfondie de SystemC-AMS. Ma maîtrise de ce langage en a été grandement améliorée. J'ai aussi beaucoup apprécié les randonnées pédestres que nous avons faites ensemble ou celles qu'il m'a conseillé de faire. J'espère que nous continuerons à nous voir régulièrement.

Il me reste encore à remercier ceux que je n'ai pas cités mais qui à un moment ou un autre m'ont manifesté leur soutien et leur amitié : Hubert, Christian et Katinka, Ben et Véro, Cécile et Nicolas, Philippe et Bénédicte, Dom et Sophie ... et tous les autres !

Sommaire

Index des figures.....	iii
Index des tableaux.....	v
Chapitre I)	
Introduction.....	1
Chapitre II)	
État de l'art.....	7
II.1) SystemC-AMS en bref.....	7
II.2) SystemC-AMS vis-à-vis des autres langages et environnements de modélisation analogique.....	10
II.3) Comparatifs entre les langages de modélisation analogique.....	20
II.4) Ce qui ressort de l'état de l'art.....	23
Chapitre III)	
Une méthodologie pour la conception des systèmes mixtes et analogiques.....	25
III.1) Cahier des charges et spécifications.....	28
III.2) Exploration d'architecture.....	28
III.2.1) Niveau système.....	28
III.2.2) Unités fonctionnelles.....	29
III.2.3) Modules.....	30
III.3) Implémentation.....	32
III.3.1) Transfert entre registres et structure interne des modules analogiques.....	32
III.3.2) Porte logique et bloc analogique.....	34
III.3.3) Structure interne des portes logiques et des blocs analogiques.....	34
III.4) Dimensionnement et tracé	35
Chapitre IV)	
Usage de SystemC-AMS dans le processus de raffinement analogique.....	37
IV.1) Styles de modélisation et niveaux d'abstraction.....	38
IV.2) Styles de modélisation de SystemC-AMS et vitesse de simulation.....	40
IV.2.1) Les quatre styles de modélisation de SystemC-AMS.....	40
IV.2.2) Répercussion du style de modélisation lors du passage à l'échelle.....	41
IV.2.3) Répercussion du style de modélisation sur un système complexe.....	46
IV.2.4) Ce qui ressort des cas d'étude.....	53
Chapitre V)	
Modèle d'amplificateur opérationnel simplifié.....	55
V.1) Pourquoi écrire à nouveau un modèle d'amplificateur opérationnel ?.....	55
V.1.1) Insuffisances du modèle de l'amplificateur idéal.....	56
V.1.2) Cas du nullor	59
V.1.3) Raisons pour lesquelles les modèles existants d'amplificateurs opérationnels à haut niveau d'abstraction ne conviennent pas non-plus.....	61
V.2) Paramètres requis au niveau transistor à spécifier au niveau bloc analogique.....	62
V.3) Modèle de l'amplificateur opérationnel simplifié.....	63
V.3.1) Hypothèses.....	63
V.3.2) Conventions et notations.....	64
V.3.3) Définition du modèle de l'amplificateur opérationnel simplifié.....	65
V.3.4) Étude du modèle sans saturation, ni limitation de la vitesse de variation.....	67
V.3.5) Étude du modèle avec saturations et vitesse de balayage finie.....	72
V.3.6) Modèle complet de l'amplificateur opérationnel simplifié	76
V.3.7) Résultats obtenus avec l'implémentation en SystemC-AMS.....	77
V.3.8) Contraintes d'utilisation du modèle d'amplificateur opérationnel simplifié.....	83

Chapitre VI)	
Perspectives.....	<u>89</u>
VI.1) Introduire ou mesurer des effets analogiques dans les modèles de circuits logiques.....	<u>89</u>
VI.1.1) Éléments généraux et définitions à propos des temps de propagation.....	<u>89</u>
VI.1.2) Modélisation du temps de propagation par des signaux rectangulaires retardés.....	<u>91</u>
VI.1.3) Transition en rampe, temps de propagation fonction des temps de transition.....	<u>92</u>
VI.1.4) Transition en exponentielle, temps de transition dépendant de la capacité de charge.....	<u>97</u>
VI.2) Améliorer SystemC-AMS.....	<u>99</u>
VI.3) Vers la génération assistée par ordinateur de systèmes hétérogènes.....	<u>101</u>
Chapitre VII)	
Conclusion.....	<u>103</u>
Annexes.....	<u>105</u>
Annexe 1)	
Transformée de Laplace : rappels succincts.....	<u>107</u>
A 1.1) Définitions.....	<u>107</u>
A 1.2) Propriétés utilisées dans l'exposé.....	<u>107</u>
A 1.3) Transformées de Laplace remarquables utilisées dans l'exposé.....	<u>108</u>
Annexe 2)	
Stabilité d'un système linéaire : rappels.....	<u>109</u>
Annexe 3)	
Amplificateur idéal : détail du calcul du gain en tension.....	<u>111</u>
Annexe 4)	
Amplificateur opérationnel simplifié.....	<u>113</u>
A 4.1) Amplificateur non-bouclé : détails des calculs.....	<u>113</u>
A 4.1.1) Équation temporelle.....	<u>114</u>
A 4.1.2) Réponse impulsionnelle.....	<u>114</u>
A 4.1.3) Réponse indicielle.....	<u>115</u>
A 4.2) Montage en contre réaction : détail des calculs.....	<u>116</u>
A 4.3) Montage en rétroaction positive : détail des calculs.....	<u>120</u>
A 4.4) Démonstration des propriétés vérifiées par le modèle de l'amplificateur opérationnel simplifié.....	<u>123</u>
A 4.5) Paramètres du modèle de l'amplificateur opérationnel simplifié.....	<u>126</u>
A 4.6) Algorithme de la tension de sortie de l'amplificateur opérationnel simplifié.....	<u>127</u>
Annexe 5)	
Fonction inverse translatée	<u>131</u>
Glossaire.....	<u>139</u>
Bibliographie.....	<u>143</u>
Publications dans le cadre de la thèse.....	<u>151</u>

Index des figures

Fig. 1 : Positionnement du projet BDREAMS par rapport au flot de conception d'un système à signaux mixtes et analogiques.....	2
Fig. 2 : Organisation du projet BDREAMS.....	3
Fig. 3 : Accroissement des couts consacrés au logiciel et à la vérification en fonction du nœud technologique.....	13
Fig. 4 : Principes clés de la modélisation système.....	14
Fig. 5 : SystemC-AMS et son principe d'interfaçage avec SystemC.....	18
Fig. 6 : Vitesse de simulation des langages de modélisation analogique les plus courants.....	20
Fig. 7 : Précision des modèles analogiques en fonction de la vitesse de simulation pour différents langages de modélisation.....	21
Fig. 8 : SystemC-A et autres langages de modélisation à différents niveaux d'abstraction.....	21
Fig. 9 : Raffinement structuré pour les systèmes mixtes.....	27
Fig. 10 : Le dictaphone au niveau système.....	29
Fig. 11 : Le dictaphone au niveau unités fonctionnelles.....	30
Fig. 12 : Modules constituant l'unité de capture du son.....	30
Fig. 13 : Modules constituant le générateur d'horloge.....	31
Fig. 14 : Principe de la décomposition des filtres analogiques.....	33
Fig. 15 : Exemple de structures pour un filtre passe bas du 3e ordre.....	33
Fig. 16 : VCO à base d'un oscillateur en anneau dont la tension d'alimentation est ajustée.....	33
Fig. 17 : Exemple de structure interne d'un amplificateur opérationnel.....	35
Fig. 18 : Filtre d'ordre n modélisé en TDF.....	41
Fig. 19 : Filtre d'ordre n modélisé en LSF.....	42
Fig. 20 : Filtre d'ordre n modélisé en LSF encapsulé.....	42
Fig. 21 : Filtre d'ordre n modélisé en ELN.....	43
Fig. 22 : Progression du nombre d'équations en fonction de l'ordre du filtre pour chaque style de modélisation.....	43
Fig. 23 : Temps de simulation avec élaboration en fonction de l'ordre du filtre.....	45
Fig. 24 : Temps de simulation avec élaboration en fonction de l'ordre du filtre (zoom).....	45
Fig. 25 : Temps de simulation seule en fonction de l'ordre du filtre.....	45
Fig. 26 : Four électrique avec régulateur PI.....	46
Fig. 27 : Four électrique modélisé en TDF.....	47
Fig. 28 : Four électrique modélisé en LSF.....	48
Fig. 29 : Four électrique modélisé en LSF encapsulé.....	48
Fig. 30 : Four électrique modélisé en ELN.....	49
Fig. 31 : Four électrique – modèle OvenPIDTDF sous Simulink.....	51
Fig. 32 : Four électrique – modèle OvenLTFTDF sous Simulink.....	51
Fig. 33 : Four électrique – modèle OvenLSF sous Simulink.....	51
Fig. 34 : Comparaison des temps de simulation du four.....	52
Fig. 35 : Nécessité de spécifier les alimentations d'un amplificateur opérationnel.....	57
Fig. 36 : Configuration en contre réaction dans le cas général.....	58
Fig. 37 : Configuration en rétroaction positive dans le cas général.....	58
Fig. 38 : Comparateur sans hystérésis.....	60
Fig. 39 : OTA cascadé et replié à sortie différentielle.....	60
Fig. 40 : Gabarit du gain en fréquence d'un système du premier ordre.....	68
Fig. 41 : Saturation par écrêtage.....	73
Fig. 42 : Transitions entre modes de saturation.....	74
Fig. 43 : Vitesse de variation limitée à la vitesse de balayage.....	74
Fig. 44 : Transitions entre modes de limitation de la vitesse de variation.....	75
Fig. 45 : Transitions entre tous les modes du modèle complet de l'amplificateur opérationnel simplifié.....	77
Fig. 46 : Circuit de test sans boucle de rétroaction (MoC ELN).....	77
Fig. 47 : Amplification différentielle et réjection du mode commun.....	78
Fig. 48 : Atténuation en fréquence.....	78
Fig. 49 : Gain en fréquence en absence de boucle de rétroaction.....	79
Fig. 50 : Déphasage en fréquence en absence de boucle de rétroaction.....	79
Fig. 51 : Saturation.....	79
Fig. 52 : Limitation de la vitesse de variation de la sortie à la vitesse de balayage de l'amplificateur.....	80

Fig. 53 : Circuit de test pour l'évaluation du courant.....	80
Fig. 54 : Courant à travers l'amplificateur.....	81
Fig. 55 : Circuit de test avec boucles de rétroaction (MoC ELN).....	81
Fig. 56 : Différenciation entre rétroactions positive et négative.....	82
Fig. 57 : Dérive exponentielle due à la rétroaction positive.....	82
Fig. 58 : Tension de sortie de l'amplificateur en contre réaction égale à la tension e_2 (entrée "+") quand les signaux d'entrée sont identiques.....	82
Fig. 59 : Gain et déphasage en fréquence avec et sans rétroaction.....	83
Fig. 60 : Reconstruction d'une sinusoïde en fonction du nombre de segments par période.....	83
Fig. 61 : Temporalisation des modèles numériques par ajout d'un bloc retardateur.....	90
Fig. 62 : Temps de propagation à la descente et à la montée.....	90
Fig. 63 : Temps de propagation modélisé par un retard.....	91
Fig. 64 : Comportement avec un temps de propagation modélisé par un retard.....	91
Fig. 65 : Transitions normales en rampe	92
Fig. 66 : Décomposition du temps de propagation.....	93
Fig. 67 : Description réaliste grâce au temps mort avant le temps de réponse à l'échelon.....	94
Fig. 68 : Transitions avortées.....	95
Fig. 69 : Impulsions brèves en entrée.....	95
Fig. 70 : Transitions critiques plus brèves que le temps mort.....	96
Fig. 71 : Transitions critiques plus longues que le temps mort.....	96
Fig. 72 : Comportement avec des transitions en rampe.....	97
Fig. 73 : Modélisation des interconnexions par un RC – transitions en arcs d'exponentielle.....	98
Fig. 74 : Schéma détaillé d'un convertisseur DE vers ELN pour une sortie numérique.....	99
Fig. 75 : Théorème des accroissements finis.....	99
Fig. 76 : Gabarit du gain en fréquence d'un système du premier ordre.....	113
Fig. 77 : Coefficients de translation.....	131
Fig. 78 : Facteur d'échelle.....	131

Index des tableaux

Tableau 1 : Tâches des sous-projets BDREAMS et répartition des partenaires	4
Tableau 2 : Les modèles de calcul de SystemC-AMS.....	8
Tableau 3 : Répartition, avant le projet BDREAMS, de l'usage des langages de modélisation dans différents domaines physiques à différents niveaux d'abstraction.....	22
Tableau 4 : Répartition actualisée de l'usage des langages de modélisation dans différents domaines physiques à différents niveaux d'abstraction.....	23
Tableau 5 : Styles de modélisations et niveaux d'abstraction.....	38
Tableau 6 : Nombre d'équations et de solveurs pour chaque style de modélisation d'un filtre d'ordre n.....	43
Tableau 7 : Paramétrage de la simulation du filtre d'ordre n.....	44
Tableau 8 : Nombre d'équations et de solveurs pour chaque style de modélisation du four.....	49
Tableau 9 : Paramétrage des simulations du four.....	52
Tableau 10 : Lien entre produit gain-bande, nombre de points calculés et temps de simulation.....	86
Tableau 11 : Réduction du temps de simulation par décimation du nombre de points enregistrés.....	87

Chapitre I)

Introduction

Depuis le début du vingt-et-unième siècle, les systèmes embarqués sur puce ou en boîtier se sont répandus au point que leur usage est désormais quotidien même si nous n'en n'avons pas toujours conscience. Les avancées technologiques les ont transformés d'unités de calcul numérique autonomes en systèmes communicants et capables d'interagir avec leur environnement ou de s'adapter à leur condition d'utilisation. Les progrès de la miniaturisation leur ont donné la possibilité d'être intégrés dans la plus part des objets usuels, rendant ceux-ci plus autonomes, plus automatiques ou plus adaptatifs. C'est par exemple le cas d'un lave-linge qui optimise la quantité d'eau au volume de linge à laver, d'un projecteur qui fait la mise au point tout seul, d'un système d'injection qui optimise la consommation de carburant ou encore d'une suspension active qui réagit aux chaos de la route. Pour les appareils nomades comme un appareil photo ou un téléphone mobile, le système embarqué se confond maintenant avec l'appareil lui-même.

Les capacités d'adaptation ou d'autocorrection sont obtenues grâce au déroulement d'un logiciel enfoui (*firmware*) qui contrôle le système embarqué. À partir des informations issues de capteurs et à l'aide de méthodes heuristiques, ce logiciel agit sur les actionneurs et règle la marche du système.

Les interactions avec le monde extérieur sont par nature analogiques. Elles nécessitent donc des interfaces qui soient analogiques. Plusieurs domaines de la physique entrent souvent en jeu, soit du fait de l'emploi de différents capteurs ou actionneurs, soit parce que les composantes du capteur ou de l'actionneur relèvent de domaines différents de la physique. Par exemple, un capteur de vision contient des photodiodes et au moins une lentille optique. Les systèmes embarqués sur puce ou en boîtier sont aujourd'hui hétérogènes.

L'hétérogénéité et l'augmentation du nombre de fonctions intégrées ont rendu la conception et la vérification des systèmes sur puce ou en boîtier de plus en plus complexes et difficiles. Dans le même temps, le succès de ces systèmes a accru la compétition entre les compagnies qui les produisent. La pression sur les coûts et les temps de mise sur le marché a rendu plus contraints les coûts et les temps de développement. De nouvelles méthodes de conception sont devenues indispensables pour appréhender la complexité et raccourcir les temps de développement. Elles impliquent d'élever le niveau d'abstraction afin de pouvoir décrire et simuler un système dans son ensemble avec toutes ses parties interconnectées.

Les méthodologies éprouvées qui ont été développées pour la conception de l'électronique numérique au niveau système emploient la technique du prototype virtuel. Un prototype virtuel est un modèle informatique abstrait qui décrit l'architecture matérielle mais avec lequel il est possible d'exécuter le logiciel enfoui. Le prototype virtuel est un moyen de fournir rapidement aux concepteurs une plateforme de développement souple avec laquelle ils peuvent à la fois entreprendre l'exploration d'architecture et l'élaboration du logiciel enfoui grâce à la simulation conjointe. Dans une telle plateforme, les interactions entre le logiciel et le matériel sont restituées. Cependant, les prototypes virtuels destinés aux systèmes purement numériques ne sont pas en mesure de saisir les comportements des interfaces analogiques. Les interactions avec l'environnement sont donc incorrectement reproduites et même souvent complètement ignorées durant le développement des composantes numériques tant matérielles que logicielles. Cette inconvénient est d'autant plus critiques que dans un système hétérogène dont l'architecture est convenablement optimisée, le logiciel enfoui interagit directement et de façon significative avec les différents capteurs et actionneurs analogiques. Il est impossible de pouvoir garantir la conformité du comportement, et la robustesse de l'architecture du système, avec une étude séparée et découplée des parties analogiques et des parties numériques. La validation des systèmes hétérogènes et le contrôle de leur conformité aux spécifications exigent des cosimulations, c'est-à-dire des simulations conjointes, qui intègrent le logiciel enfoui et toutes les composantes matérielles numériques et analogiques. Les prototypes virtuels doivent donc évoluer pour prendre en compte les éléments analogiques du systèmes. La plus grande difficulté est d'introduire les descriptions analogiques sans

dégrader la vitesse de simulation du prototype. C’est effectivement le problème rencontré dans le cas des cosimulations avec un simulateur électrique beaucoup trop gourmand en calcul et fournissant un niveau de détail trop important car inexploité. Même les cosimulations utilisant les langages de description matérielle, qui déjà élèvent le niveau d’abstraction, ne sont pas assez rapides pour convenir à une plateforme destinée au développement du logiciel enfoui. Pour rendre possible la réalisation d’un prototype virtuel mixte réunissant parties numériques et analogiques, les composantes analogiques doivent donc être décrites au niveau système ce qui signifie une élévation du niveau d’abstraction par rapport à ce qui se faisait jusqu’alors avec les langages de description matérielle analogique. Répondre à ce besoin d’élévation des descriptions analogiques a été le point de départ de la création de SystemC-AMS, l’extension de SystemC pour la description et la simulation des systèmes hétérogènes mixtes et analogiques. [1 ; 2]

La mise en œuvre des prototypes virtuels mixtes, numériques et analogiques, requiert de nouvelles méthodes pour modéliser les parties analogiques avec le bon niveau d’abstraction mais surtout pour convenablement structurer le passage depuis les niveaux les plus abstraits, utilisés dans le prototype virtuel, vers les niveaux les plus détaillés, nécessaires à la mise au point de l’électronique du système réel. Le projet BDREAMS a été monté afin de faire des propositions en ce sens et aboutir à une méthodologie spécifique.

Le projet BDREAMS (*Beyond Design Refinement of Embedded Analogue and Mixed-signal Systems*) est un projet européen qui s’inscrit dans le cadre du programme Eurêka et qui est financé par le groupement CATRENE (*Cluster for Application and Technology Research in Europe on NanoElectronics*). Il s’est déroulé d’octobre 2008 à octobre 2011. La ligne directrice de ce programme est l’amélioration de la compétitivité des industries européennes du domaine de la conception des systèmes embarqués à signaux mixtes et analogiques (*Embedded Analog Mixed-Signal systems – EAMS systems*) afin de leur préserver une position dominante dans ce secteur. L’amélioration de la compétitivité exige une réduction du temps de mise sur le marché des produits dans une parfaite maîtrise de leur cout de développement. Le projet doit conduire à la modernisation de la conception des systèmes embarqués, sur puce ou en boîtier, à signaux mixtes et analogiques. Pour y parvenir, il doit établir un cadre pour la modélisation et la simulation au moyen de prototypes virtuels mixtes et fournir une méthodologie pour le raffinement des modèles. Il doit aussi apporter une méthodologie pour la bonne définition des blocs matériels propriétaires analogiques (*analog Intellectual Property – aIP*) afin de rendre possible leur réemploi direct sans devoir les redéfinir ou les modifier. Les outils et les méthodes élaborés au cours de ce projet doivent autoriser une modélisation et une validation des parties analogiques de façon intégrées avec les parties numériques. La figure 1 montre les couches manquantes dans le flot de conception d’un système embarqué à signaux mixtes et analogiques (gauche de la figure) et le positionnement du projet BDREAMS par rapport à ce flot (droite de la figure). Il était prévu dès le début du projet que les preuves de concepts et les démonstrateurs soient réalisés en utilisant SystemC et SystemC-AMS. Il était également prévu que l’encapsulation des blocs matériels propriétaires analogiques soit obtenue à l’aide d’une version étendue du langage IP-XACT. Un autre objectif du projet BDREAMS était de faire des propositions pour la normalisation SystemC-AMS par le consortium OSCI (*Open SystemC Initiative*) et pour la normalisation par le consortium Accellera de la version de IP-XACT étendue à la description des blocs matériels propriétaires analogiques.

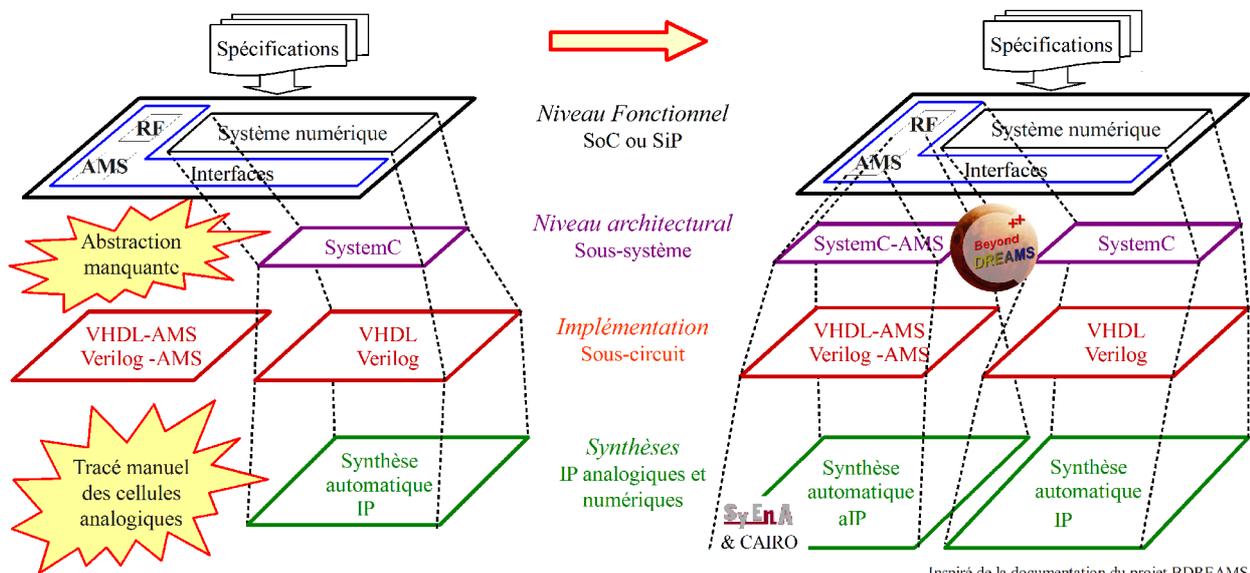


Fig. 1 : Positionnement du projet BDREAMS par rapport au flot de conception d’un système à signaux mixtes et analogiques

Il est à noter que les consortiums OSCI et Accellera se sont rapprochés pour fusionner fin 2011 et ainsi former l'organisation Accellera Systems Initiative.

L'équipe du projet BDREAMS a intégré des membres de deux projets focalisés sur la modélisation analogique bas niveau et la synthèse analogique avec des blocs matériels propriétaires analogiques (génération et réemploi). Il s'agit du projet SYENA (Ministère fédéral allemand de l'Éducation et de la Recherche) et du projet CAIRO du Laboratoire d'Informatique de Paris (LIP6)

Le projet BDREAMS a impliqué des acteurs européens majeurs dans le domaine de la conception des systèmes embarqués à signaux mixtes et analogiques :

- quatre compagnies de semiconducteurs, spécialistes des systèmes sur puce et en boîtier
 - Infineon (IFX) – Allemagne
 - NXP Semiconductors (NXP) – Pays-Bas
 - STEricsson (STE) – France
 - STMicroelectronics (STM) – France
- un équipementier automobile spécialiste de l'électronique pour l'automobile
 - Robert Bosch Automotive Electronics (RB) – Allemagne
- trois PME spécialisées dans les outils de conception assistée par ordinateur (CAO) avec une grande expertise en SystemC, SystemC-AMS ou IP-XACT
 - Magillem Design Services (MDS) – France
 - Dizain-Sync (D-Sync) – Pays-Bas
 - Twente Institute for Wireless and Mobile Communications (WMC) – Pays-Bas
- trois centres européens de recherche et de transfert technologique
 - Fraunhofer IIS/EAS (FhG) – Allemagne
 - CEA-LETI – France
 - IMEC-NL – Pays-Bas
- quatre institutions académiques
 - Université Technologique de Vienne (TUV) – Autriche
 - Institut Polytechnique de Grenoble, Laboratoire TIMA (TIMA) – France
 - Université Pierre et Marie Curie, Laboratoire LIP6 (UPMC) – France
 - Université Technologique de Delft (TUD) – Pays-Bas

La figure 2 donne un aperçu de l'organisation du projet BDREAMS tandis que le tableau 1 liste les sous-projets et les tâches affectées à ceux-ci. Le tableau 1 montre également la répartition des partenaires entre les différents sous-projets.

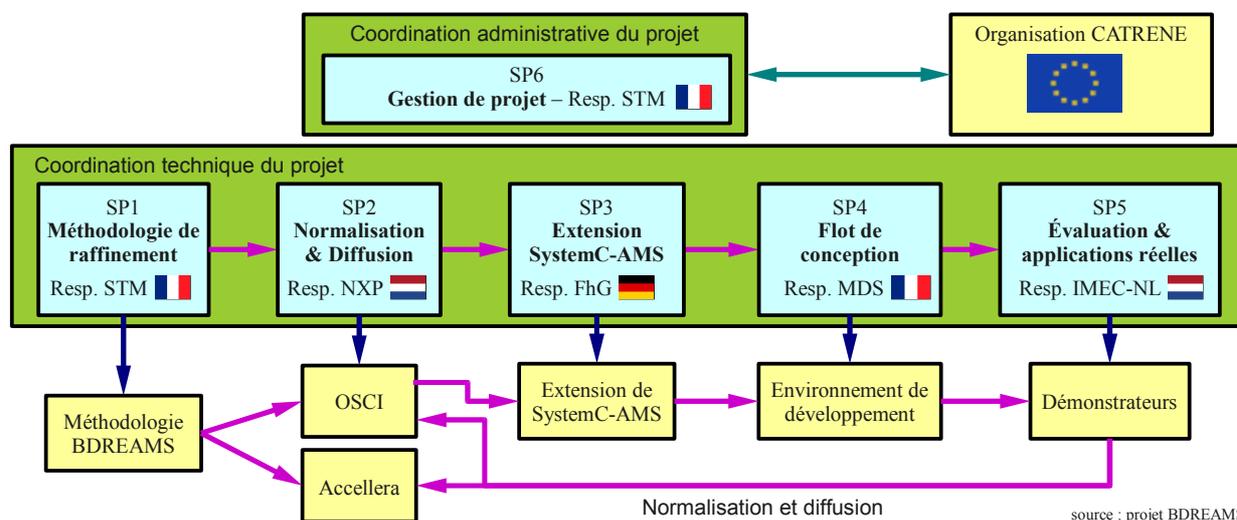


Fig. 2 : Organisation du projet BDREAMS

Sous-projet	Dénomination du sous-projet et tâches	Partenaires
SP1	Spécification de la méthodologie de raffinement <ul style="list-style-type: none"> Analyse des besoins pour les systèmes EAMS Modèles de calcul AMS et interaction avec SystemC TLM Méthodologie de raffinement Spécification des outils et des bibliothèques 	TUV, NXP, STM, FhG, TIMA, UPMC, CEA-LETI, D-Sync, RB, IMEC-NL
SP2	Standardisation et déploiement <ul style="list-style-type: none"> Standard AMS pour SystemC (OSCI) Standard AMS pour IP-XACT (Accellera) Déploiement et diffusion Relation avec les vendeurs d'outils de CAO 	TUV, NXP, STM, IFX, FhG, MDS, D-Sync, RB
SP3	Extensions SystemC/SystemC-AMS <ul style="list-style-type: none"> Établissement du flot logiciel de conception et de vérification Mise en œuvre de la preuve de concept de la bibliothèque d'extension de SystemC pour la description et la simulation des systèmes EAMS Mise en œuvre de la preuve de concept de la bibliothèque TLM AT pour les systèmes EAMS complexes Bibliothèques d'application spécifique Mise en œuvre des concepts de prise en charge des interfaces normalisées IP-XACT 	TUV, NXP, STM, FhG, TIMA, UPMC, MDS, CEA-LETI
SP4	Environnement et flot de conception <ul style="list-style-type: none"> Mise en œuvre des métamodèles IP-XACT Flot de conception AMS et définition du concept de « spécifications simulables » Mise en œuvre de la conception mixte et analogique au niveau système 	NXP, STM, MDS, RB, TUD
SP5	Évaluation à l'aide d'applications réelles <ul style="list-style-type: none"> Conception de prototypes virtuels Évaluation de la méthodologie 	NXP, STM, STE, IFX, FhG, TIMA, UPMC, MDS, CEA-LETI, WMC, RB, TUD, IMEC-NL
SP6	Gestion du projet	STM, NXP, UPMC, MDS, WMC

Tableau 1 : Tâches des sous-projets BDREAMS et répartition des partenaires

Cette thèse s'inscrit dans le cadre du projet BDREAMS. Elle est une partie intégrante du sous-projet n°1 qui est consacré à l'élaboration de la méthodologie de raffinement. Le travail de thèse porte sur l'analyse de méthodes de conception des systèmes mixtes à haut niveau d'abstraction afin de définir les niveaux d'abstractions manquants pour combler le fossé existant entre les descriptions algorithmiques au niveau système et les descriptions de plus bas niveaux telles celles utilisées par les simulateurs électriques. Le chapitre III répond à cet objectif par la définition de niveaux d'abstraction pour la modélisation des parties analogiques et par leur structuration de façon à les mettre en correspondance avec les niveaux d'abstraction déjà définis pour la modélisation des parties numériques. Un processus de raffinement méthodique des modèles analogiques est introduit de façon à abaisser progressivement le niveau d'abstraction de ces modèles dans le contexte d'une conception descendante globale et coordonnée du système. Le travail de thèse a trait aussi à la sémantique du langage SystemC-AMS à utiliser pour modéliser les systèmes embarqués hétérogènes de façon efficace. Le chapitre IV présente une étude sur les différents modèles de calcul de SystemC-AMS. L'objectif de cette étude est d'aider les concepteurs à choisir le modèle de calcul le mieux adapté à chaque niveau d'abstraction de façon à atteindre le meilleur compromis entre vitesse de simulation et précision. Pour obtenir une méthode de conception fiable et efficace de ces systèmes hétérogènes, il est important de garantir que les contraintes du modèle de plus haut niveau seront propagées aux niveaux d'abstraction inférieurs et d'autre part que les limitations et les non-linéarités, qui apparaîtront au fur et à mesure que le raffinement s'approche des niveaux les plus détaillés, n'altèrent pas le comportement du système au point de le faire s'écarter des spécifications. Pour atteindre une telle fiabilité, la progressivité doit être maintenue tout au long du processus de raffinement. Or, ce n'est pas toujours le cas avec les méthodes classiques de

conception qui ont tendance à négliger les niveaux intermédiaires pour passer au plus vite aux modèles les plus précis. Le chapitre V discutera à travers le cas d'étude d'un modèle d'amplificateur opérationnel simplifié de ce que devrait être un modèle de niveau intermédiaire dans le cadre du processus de raffinement méthodique introduit au chapitre III. Le propos est continué au chapitre VI par une réflexion, en perspective de futurs travaux, sur l'utilisation de SystemC-AMS pour enrichir les modèles numériques, décrits en SystemC, avec des informations concernant les phénomènes analogiques ayant potentiellement des conséquences sur le comportement du circuit numérique. D'autres perspectives sont évoquées pour l'amélioration de SystemC-AMS ou pour continuer l'évolution des méthodes de conception des systèmes hétérogène vers une génération automatique. Afin de mieux situer le contexte de travail, une brève présentation technique de SystemC-AMS est faite au début du chapitre II. Elle est suivie d'un rappel historique qui retrace les circonstances qui ont amenées à la création de ce langage. Lui succède un comparatif entre les langages les plus employés à ce jour pour la modélisation et la simulation des circuits analogiques.

Chapitre II)

État de l'art

Le travail consigné dans cette thèse a été élaboré en se servant de SystemC-AMS pour bâtir des exemples illustratifs ou démontrer la faisabilité de certains concepts. Afin de préciser le contexte du travail, une description technique succincte introduit le langage SystemC-AMS. L'évolution des concepts d'abstraction, de la modélisation et des descriptions des composants, circuits puis systèmes électroniques (ou à forte dominante électronique) est ensuite retracée depuis la conception sur carte de test jusqu'à l'apparition de SystemC-AMS. Les langages de modélisation et de simulation les plus marquants sont évoqués au cours de cet exposé historique. Ce dernier est suivi d'une présentation de comparatifs entre les langages qui sont les plus employés, au moment où est écrit ce mémoire, pour la modélisation et la simulation des circuits et des systèmes analogiques.

II.1) SystemC-AMS en bref

SystemC-AMS est une extension de SystemC. SystemC est une bibliothèque C++ de classes spécialisées qui est destinée à la modélisation et à la simulation de systèmes numériques fonctionnant avec un logiciel enfoui. SystemC est tout particulièrement conçu pour étudier le partitionnement logiciel/matériel et simuler les interactions entre le matériel et le logiciel. La modélisation transactionnelle (*Transaction Level Modeling – TLM*) qui abstrait les communications le rend particulièrement efficace pour le développement du logiciel enfoui. Ses différents modèles de calcul donnent la possibilité aux concepteurs de raffiner leur modèle depuis le niveau système jusqu'au niveau de transfert entre registres (*Register Transfer Level – RTL*) et ainsi de mener à bien l'exploration d'architecture. Mais SystemC, qui a été construit pour la simulation événementielle de systèmes manipulant des valeurs numériques, n'est pas en mesure de simuler des grandeurs à temps continu et à amplitude continue. Il n'est adapté ni pour créer ni pour raffiner des modèles de systèmes analogiques. Une extension de SystemC pour la description et la simulation des systèmes hétérogènes mixtes et analogiques a donc été créée afin de répondre aux demandes des industriels des télécommunications, de l'automobile et des semiconducteurs : SystemC-AMS. Ces industriels sont en effet demandeurs de méthodes uniformisées et de plateformes de développement normalisées de façon à favoriser le réemploi et l'échange de blocs matériels propriétaires à signaux mixtes et analogiques (*analog Intellectual Property – aIP*). Leur souhait est de pouvoir développer le programme informatique enfoui en interaction avec les parties analogiques du système sans devoir dépendre de la fabrication d'un prototype matériel. Pour cela, ces parties analogiques doivent être modélisées et les modèles doivent s'exécuter suffisamment rapidement pour ne pas entraver le travail des ingénieurs logiciels. Ceci nécessite des vitesses de simulations plus élevées que toutes les solutions à base de cosimulation de plusieurs outils de conception assistée par ordinateur séparés, chacun spécialisé dans un domaine. Il est à noter que la mise en place d'une cosimulation n'est pas toujours aisée, les outils CAO étant rarement prévus pour ce besoin. [1 ; 3]

L'extension SystemC-AMS apporte les avantages suivants :

- SystemC et SystemC-AMS forment un langage spécialement conçu pour la modélisation et la simulation des systèmes embarqués hétérogènes. Il est capable de rendre compte des interactions entre matériel et logiciel ainsi qu'entre parties numériques et analogiques. Le système peut donc être élaboré globalement avec toutes ses éléments constitutifs qu'ils soient logiciels ou matériels, numériques ou analogiques, ou qu'ils appartiennent à des domaines physiques différents.

- SystemC et SystemC-AMS forment un langage qui est adapté aux descriptions à haut niveau d’abstraction tout en permettant l’exploration d’architecture.
- SystemC-AMS fournit tout le nécessaire pour s’interfacer et se synchroniser avec SystemC. Développer en SystemC et SystemC-AMS offrent la possibilité de décrire des systèmes mixtes sans avoir les soucis de la mise en œuvre de la cosimulation rencontrés avec l’utilisation d’outils ou de langages spécialisés différents (ex : Matlab + Saber ; SystemC + Spice).
- L’association de SystemC avec SystemC-AMS fournit un compromis performant entre vitesse de simulation et précision.
- SystemC-AMS est conçu pour être évolutif avec la possibilité d’ajouter des solveurs spécifiques à un besoin.
- SystemC et SystemC-AMS sont normalisés (IEEE 1666-2011 pour SystemC et par Accellera Systems Initiative pour l’extension SystemC-AMS). Cette normalisation doit garantir l’interopérabilité entre les outils de développement conformes à cette norme quelle que soit la société qui les développe.
- L’esprit impulsé par les consortiums de normalisation est que les implémentations de SystemC et de SystemC-AMS doivent être à code ouvert de façon à ce que le langage reste adaptable et évolutif.
- La preuve de concept (*Proof of Concept – PoC*) de SystemC-AMS réalisée par l’institut Fraunhofer constitue une plateforme de développement gratuite et accessible à tous puisqu’elle est mise à disposition par celui-ci sans contrepartie pécuniaire.
- SystemC et SystemC-AMS forment un langage avec de bonnes perspectives de pérennité. SystemC est déjà adopté par les industriels et SystemC-AMS a retenu l’attention de plusieurs d’entre eux.

SystemC-AMS dans sa version normalisée 1.0 fournit trois modèles de calcul : le flot de données temporelisé (*Timed Data Flow – TDF*), le flot de signal linéaire (*Linear Signal Flow – LSF*) et le réseau électrique linéaire (*Electrical Linear Network – ELN*). Dans le contexte de ce langage, un modèle de calcul (*model of computation – MoC*) est un formalisme de modélisation utilisé pour composer un modèle exécutable. Ce formalisme constitue la sémantique du modèle par la réunion des éléments syntaxiques du langage de modélisation, des règles syntaxiques d’assemblage des éléments syntaxiques et des règles de composition des règles syntaxiques. Un modèle exécutable désigne un modèle, écrit à l’aide d’un langage informatique, qui permet le déroulement de simulations sur ordinateur. Les caractéristiques principales de ces modèles de calcul sont résumées dans le tableau 2, les détails peuvent être trouvés dans le guide l’utilisateur [4] et dans le manuel de référence du langage de SystemC-AMS [5].

Modèle de calcul	TDF <i>Timed Data Flow</i>	LSF <i>Linear Signal Flow</i>	ELN <i>Electrical Linear Network</i>
Représentation temporelle	Temps discret	Temps continu	Temps continu
Modèle de données	Signaux abstraits échantillonnés	Signaux abstraits continus	Signaux électriques
Interconnexions entre modules	Flot de données à travers des ports orientés	Flot de signal à travers des ports orientés	Terminaux reliés par des nœuds électriques
Description comportementale	Non restreinte, modélisation possible des non-linéarités	Opérations mathématiques linéaires sur des signaux	Lois de Kirchhoff. Comportement des composants restreint à leur zone linéaire
Type de traitement informatique	Algorithmes	Systèmes d’équations différentielles algébriques	Systèmes d’équations différentielles algébriques

Tableau 2 : Les modèles de calcul de SystemC-AMS

Tout comme SystemC, SystemC-AMS fonctionne par association de modules. Les modèles de calcul TDF et LSF manipulent des modules et des signaux abstraits alors que le modèle de calcul ELN est destiné

à des représentations électriques. Le modèle de calcul ELN est le seul qui prenne en charge la notion de composants électriques. Un assemblage de modules appartenant à un même modèle de calcul forme une grappe (*cluster*).

Le comportement d'un module TDF n'est pas prédéfini. Il est dépeint à l'aide d'un algorithme créé par l'utilisateur. La description algorithmique permet de modéliser les non-linéarités relativement facilement sans pénaliser la vitesse de simulation. Il n'y a quasiment aucune limitation. Des classes¹ spéciales sont fournies pour réaliser des descriptions comportementales au moyen de fonctions de transfert ou de représentations d'états.

Les modèles de calcul LSF et ELN sont fournis avec une bibliothèque de primitives que l'utilisateur assemble pour créer sa description. Une primitive est un module au comportement prédéfini bien que paramétré. Dans la version 1.0 de la norme SystemC-AMS, l'utilisateur n'est pas autorisé à modifier une primitive et il n'existe pas de mécanisme pour lui laisser la liberté d'en créer d'autres.

Chaque primitive LSF correspond à une opération ou une fonction mathématique, y compris une fonction de transfert ou une représentation d'états. L'association de ces primitives crée un traitement mathématique sur les signaux du circuit. Les traitements dans le modèle de calcul LSF reposent sur la résolution d'équations différentielles algébriques. Par conséquent, seules les fonctions mathématiques linéaires sont disponibles. Il n'y a pas, par exemple, de fonction exponentielle. Des sources sinusoïdales sont certes mises à disposition mais elles sont dépendantes du temps, elles ne peuvent donc pas être appliquées à des signaux du système. Le modèle de calcul LSF est particulièrement adapté pour la modélisation comportementale des systèmes asservis linéaires. Il a été conçu dans cet objectif.

Chaque primitive ELN correspond à un composant électrique idéal. L'association de ces primitives crée une description de type réseau électrique. Les traitements dans le modèle de calcul ELN reposent aussi sur la résolution d'équations différentielles algébriques. Par conséquent, seul des composants au comportement linéaire ou restreint au domaine linéaire de leur réponse sont inclus dans la liste des primitives. Par exemple, il n'y a pas de transistors. Les transistors doivent être remplacés par des sources contrôlées ou par des interrupteurs. Des sources sinusoïdales sont présentes dans la liste de primitives mais, comme dans les modèles de calcul LSF, ce sont forcément des fonctions du temps, elle ne peuvent pas être appliquées à des signaux internes au circuit.

Les modules TDF sont connectés entre-eux via des ports qui sont reliés par des signaux. Les connexions entre modules LSF sont effectuées avec un système similaire. Les modules ELN sont reliés entre-eux par des terminaux qui sont connectés au moyen de nœuds électriques.

Dans le modèle de calcul TDF, les données traitées correspondent à des signaux échantillonnés (temps discret et amplitude continue). Les modèles de calcul LSF et ELN, eux, traitent de signaux à temps continu. Le modèle de calcul ELN est dit conservatif* car il se conforme aux lois de Kirchhoff. Il doit par conséquent calculer deux valeurs pour chaque nœud, l'une pour la tension, l'autre pour le courant. Le modèle de calcul LSF est non conservatif et les signaux sont traités selon un flot orienté. Ce modèle de calcul ne peut pas traiter des effets mutuels entre une sortie et une entrée reliées entre-elles. En contrepartie, il n'a qu'une valeur à traiter par signal. Bien que basé sur le même principe de traitement que le modèle ELN (résoudre des équations différentielles algébriques), le modèle de calcul LSF sera donc plus rapide que celui-ci car il a moins d'équations à gérer pour un modèle de taille comparable.

Les échanges entre le domaine analogique (description en SystemC-AMS) et le domaine numérique (description en SystemC) se font grâce à des ports spécialisés qui convertissent les données et surtout garantissent la bonne synchronisation entre les deux domaines. Chaque modèle de calcul de SystemC-AMS possède ses propres convertisseurs vers le domaine numérique. Certaines des primitives LSF et ELN incluent un tel convertisseur dans leur constitution. Elles peuvent être directement connectées à un signal numérique SystemC. D'autres ports spécialisés servent de convertisseurs entre les modèles de calcul LSF et TDF et entre ELN et TDF. Il n'existe, par contre, pas de convertisseurs qui permettent une liaison directe entre une primitive LSF et une primitive ELN.

SystemC-AMS tire sa vitesse de simulation essentiellement de son ordonnanceur statique. Chaque simulation est donc nécessairement précédée d'une phase d'élaboration qui, outre la vérification des bonnes connections entre les modules, détermine dans quel ordre, et combien de fois, les fonctions de traitement de chacun des modules doivent être appelées. Une fois établie, cette séquence d'appels de fonction est déroulée pendant la simulation.

¹ Classe doit être compris ici au sens de C++

* Voir glossaire

II.2) SystemC-AMS vis-à-vis des autres langages et environnements de modélisation analogique

Chaque saut technologique en électronique offre de nouvelles possibilités selon deux axes : l'augmentation de la densité, le nombre de transistors par millimètre carré augmente, ou l'intégration, des circuits jusqu'alors discrets deviennent intégrés, rassemblés en une seule puce ou au moins dans un même boîtier. À cause de la réduction des dimensions et de l'accroissement du nombre de composants à gérer dans un même circuit, chaque saut technologique s'accompagne de nouvelles difficultés qui ne peuvent être résolues qu'avec l'introduction de nouveaux outils et de nouvelles méthodes.

L'apparition des circuits intégrés s'est accompagnée de nouvelles problématiques pour le concepteur de circuits électroniques. Dans le cas d'un circuit à composants discrets, la mise au point d'un nouveau circuit est effectuée en assemblant des composants sur une plaquette de test. Le prototype obtenu est ici très proche du circuit final et il peut servir à évaluer les performances du circuit. Le circuit définitif correspond à une mise au propre du prototype. Le passage à l'échelle microscopique des circuits intégrés rend cette démarche inapplicable, d'une part à cause des dimensions physiques inaccessibles à un opérateur humain (et donc aux appareils de mesure classiques comme l'oscilloscope) et, d'autre part parce que la mise au point sur un circuit discret équivalent ne permet pas une évaluation correcte des performances. En effet, les couplages capacitifs et autres effets parasites ne sont pas du même ordre de grandeur entre un circuit intégré et un circuit discret. À cela s'ajoute le coût de fabrication d'un circuit intégré qui interdit une mise au point purement par essais et échecs jusqu'à l'obtention du résultat souhaité. Pourvoir s'assurer du comportement requis et surtout pourvoir évaluer les performances du circuit complet avant la fabrication du circuit intégré s'est révélé indispensable dès la fin des années 1960. Ce besoin a conduit à la création de **simulateurs électriques**. Ce sont des simulateurs analogiques qui mettent en œuvre des systèmes d'équations algébriques et différentielles, linéaires ou non, pour décrire les circuits. Les systèmes d'équations sont calculés sur ordinateur à l'aide des techniques de résolution numérique. La simulation est de type continue : le simulateur calcule en permanence des valeurs y compris lorsque les entrées du circuit ne changent pas, les sorties pouvant, elles, continuer à évoluer. Le plus célèbre des simulateurs électriques est incontestablement SPICE de l'Université de Berkeley. C'est aussi le premier du genre. Sa première version a été achevée en 1972. Dans SPICE, les circuits sont décrits comme des assemblages de composants élémentaires : la résistance, le condensateur, l'inductance indépendante, l'inductance couplée, la source indépendante de courant, la source indépendante de tension, la source de courant commandée en tension, la diode, le transistor bipolaire, le JFET* et le MOS-FET*. La source de courant commandée en tension existe en deux versions, l'une linéaire et l'autre non-linéaire. Elle est décrite avec un polynôme dont l'ordre est à fixé par le concepteur. Les semiconducteurs sont chacun modélisés à l'aide de deux schémas électriques équivalents : un schéma non-linéaire grand signal pour l'analyse statique et un schéma linéaire petit signal pour l'analyse dynamique. Les équations décrivant le circuit sont formulées à l'aide de l'analyse nodale ou de l'analyse nodale modifiée. L'analyse nodale modifiée est plus efficace en présence de sources de tension flottante, c'est-à-dire non reliée à la masse. Quand toutes les sources de tension sont reliées à la masse, l'analyse nodale pure est la plus performante en termes de vitesse de calcul et de ressources mobilisées. [6]

Au fil des années, les modèles des composants de base et tout particulièrement ceux des transistors ont été améliorés afin de refléter le plus fidèlement possible les comportements observés sur silicium. La précision atteinte est telle que les simulateurs électriques sont toujours appréciés aujourd'hui et les simulations obtenues sont à la fois une référence et un passage obligé avant toute fabrication de circuit intégré.

Toutefois, décrire entièrement un circuit sous la forme d'une liste de transistors et de composants passifs est très vite fastidieux et source de nombreuses erreurs. Cela devient même insurmontable pour un humain quand le nombre de composants dépasse la centaine. La représentation des fonctions numériques par compositions de portes logiques a constitué un premier niveau d'abstraction qui a simplifié l'élaboration des circuits numériques. En effet, il n'était conservé des signaux électriques que les aspects pertinents à une représentation en logique binaire : des signaux à deux niveaux logiques, '0' et '1' dont la valeur de l'amplitude est masquée, et avec des fronts verticaux c'est-à-dire avec des transitions instantanées. Mais cela s'est révélé rapidement insuffisant, car, même si une porte logique regroupe plusieurs transistors, le nombre de portes logiques à intégrer est lui-même devenu trop grand pour qu'un humain puisse y faire face.

Les **langages de description matérielle** (*Hardware Description Languages – HDL*) ont vu le jour au cours des années 1980 pour faire face à l'accroissement de complexité et de taille des circuits qu'a engendré le succès des microprocesseurs. Ces langages élèvent le niveau d'abstraction en dissociant la description

* Voir glossaire

comportementale du schéma électronique. Même s'il est toujours possible de représenter le circuit comme un assemblage de portes logiques, son comportement peut désormais être décrit au moyen d'équations logiques (algèbre booléenne) et de machines à états finis. Des syntaxes ont été inventées pour traduire le parallélisme entre les différentes parties du circuit (concurrence des processus). Avec les langages de description matérielle, le concepteur se concentre davantage sur ce que doit faire le composant plutôt que comment il doit le faire. La conception descendante qui part des spécifications devient possible. Ces langages sont prévus pour la description modulaire et hiérarchique. Par la séparation de l'interface d'un module de son architecture, ils rendent possible différentes représentations d'un même circuit avec éventuellement différents niveaux d'abstraction. Il est également faisable de diffuser une description en ne publiant que l'interface, l'architecture étant livrée cachée dans une bibliothèque de composants. Les langages de description matérielle sont donc bien adaptés à la création de blocs matériels propriétaires (*IP – Intellectual Property*) qui autorisent le réemploi de descriptions créées au cours d'un précédent projet ou provenant d'une société tierce. Ces langages étant conçus pour la modélisation des circuits numériques, la simulation est événementielle. Contrairement à la simulation continue, celle employée dans les simulateurs analogiques et les simulateurs électriques, la simulation événementielle ne progresse que lorsqu'un événement se produit. Un événement est typiquement un changement sur au moins une des entrées auxquelles le circuits simulé est déclaré sensible. Il est postulé que l'état d'un circuit ne peut changer qu'à la suite d'un événement et par conséquent, une fois l'événement traité, le circuit reste inchangé entre deux événements quelle que soit la durée qui les sépare. Cette approche allège grandement la charge de calcul et ainsi accélère fortement la vitesse de simulation par rapport à une simulation électrique. Description modulaire purement comportementale, vitesse de simulation élevée et blocs matériels propriétaires ont fait le succès des langages de description matérielle. Les langages de description matérielle qui ont perduré et qui sont toujours utilisés à ce jour par les industriels sont le VHDL et le Verilog.

VHDL a été normalisé une première fois en 1987 par l'IEEE (IEEE Std. 1076). La communauté des utilisateurs de VHDL a rapidement identifié le besoin de bibliothèques de modèles de composants prédéfinis et de règles normalisées d'écriture. Les règles normalisées d'écriture servent à rendre les modèles indépendants et interchangeable vis-à-vis des outils de développement et des styles d'écriture des différents concepteurs. Une première étape fut réalisée en ce sens avec la révision du standard de 1993. C'est grâce à cette révision que VHDL fut largement adopté. Les utilisateurs de VHDL avaient également exprimé le désir d'une synthèse automatique des circuits à partir des descriptions VHDL, le VHDL ayant été d'abord conçu pour la simulation. Il fallait donc que les modélisations soient temporellement précises et tiennent notamment compte des temps de propagation. Les règles d'écriture et la structure du code devinrent plus strictes. Cette démarche pour réaliser des ASIC* à partir de description VHDL se concrétisa en 1995 par la normalisation des bibliothèques VITAL (*VHDL Initiative Toward ASIC Libraries – IEEE Std. 1076.4*).

Verilog était à l'origine un langage propriétaire. Face à la concurrence de VHDL déjà ouvert et normalisé et au risque de perdre des parts de marché au profit de celui-ci, la société Cadence, alors propriétaire des droits sur Verilog, décide en 1991 d'ouvrir le code de Verilog. Il s'en suit un travail de clarification et de structuration du langage de façon à le rendre le plus indépendant possible des implémentations des vendeurs d'outils de conception assistée par ordinateur (CAO). Ce travail déboucha sur la normalisation de Verilog en 1995 (IEEE Std 1364). Verilog était conçu à l'origine pour vérifier la logique. Il est toujours apprécié pour cette capacité de vérification par simulation des circuits numériques. Verilog a été ensuite augmenté pour offrir la possibilité de synthétiser automatiquement un ASIC à partir d'une description Verilog.

L'amélioration des technologies a ouvert, au cours des années 1990, la voie des appareils portables autonomes et des connexions sans fil. Des circuits hybrides virent le jour. Réunissant dans un même boîtier les composants analogiques et numériques nécessaires, ils rendirent prête à l'emploi des fonctions complexes, comme un émetteur-récepteur ou un convertisseur DC-DC. Des microcontrôleurs avec des convertisseurs analogiques vers numériques intégrés sur la même puce que le cœur numérique existaient déjà depuis la fin des années 1980. Avec ces premiers circuits à signaux mixtes, à la fois numériques et analogiques, se fit sentir le besoin d'avoir des langages de description matérielle analogique. Il ne s'agissait pas de seulement fournir un simulateur analogique capable de résoudre des systèmes d'équations algébriques et différentielles au cours d'une simulation continue, cela était déjà disponible avec les simulateurs électriques. Il manquait la possibilité de décrire les circuits analogiques par leur comportement. Différentes compagnies ont proposées des solutions en ce sens avec des langages purement analogiques tels que FAS (Anacad, 1988), SpectreHDL (Cadence, 1994), and Verilog-A (Open Verilog International, 1997) ou mixtes comme MAST (Analogy, 1986), HDL-A (Mentor Graphics, 1992). Les langages mixtes prennent en charge à la fois les simulations événementielles et continues. Cependant ces solutions étaient propriétaires et non normalisées. Or, les

* Voir glossaire

concepteurs de circuits désiraient obtenir pour les langages de description matérielle analogique ce qui existait déjà pour les langages de description matérielle numérique : description comportementale séparée de la description structurelle, abstraction à différents niveaux, modularité, structuration hiérarchique ainsi que l'indépendance et l'interchangeabilité vis-à-vis des outils de développement. Ils voulaient en plus un langage qui soit uniforme pour les descriptions numériques et analogiques [7 ; 8]. Cette demande conduisit Gary Arnold, président et directeur général de la société Analogy, à déclarer en 1995 que la conception à signaux mixtes et technologie mixte exigeait la puissance réunie de MAST, VHDL et Verilog [9]. Il n'était pas envisageable de repartir de zéro, VHDL et Verilog devaient évoluer et être étendus de façon à autoriser les descriptions mixtes. La deuxième moitié des années 1990 a été aussi la période où il fut possible d'intégrer un émetteur-récepteur complet sur la même puce que le processeur et au cours de laquelle les dispositifs microélectromécaniques (MEMS) ou microfluidiques devinrent une réalité industrielle². L'évolution des langages de description matérielle, initialement prévue pour une description combinée de l'électronique numérique et analogique, a été logiquement élargie de façon à rendre possible la prise en compte de l'hétérogénéité des systèmes. Ceci a abouti à **VHDL-AMS**, normalisé en 1999 (IEEE Std. 1076.1-1999), et à **Verilog-AMS**. Ce dernier n'est pas encore normalisé à ce jour. Cependant l'organisation Accellera Systems Initiative a déjà publié un manuel de référence du langage (LRM) [8]. Verilog-AMS devrait être normalisé très prochainement [10]. L'aptitude de VHDL-AMS et Verilog-AMS à permettre la modélisation de systèmes complexes fortement hétérogènes a été démontré notamment par F. Pecheux, C. Lallement et A. Vachoux dans [11] avec le cas d'étude d'un coussin gonflable de sécurité (*airbag*). Leur modèle était capable de rendre compte des interactions entre domaine électrique et mécanique, électrique et thermique, électrique et optique mais aussi de la réaction chimique utilisée pour le gonflement du coussin. Durant la période au cours de laquelle VHDL-AMS et Verilog-AMS étaient élaborés, les sociétés d'outils de CAO continuèrent le développement de leurs solutions propriétaires. En particulier, MAST fut revu de façon à pouvoir gérer l'hétérogénéité des domaines physiques. Cette version remaniée devint Saber et Analogy fut absorbée par Synopsys. Saber est toujours aujourd'hui un outil de CAO largement utilisé par les avionneurs et les constructeurs automobiles [12].

Au tournant du millénaire les systèmes numériques programmables étaient déjà très répandus et présents dans tous les objets du quotidien allant de la gestion de l'injection des moteurs d'automobile, au lave-linge en passant par l'imprimante ou l'appareil photo. Chaque nœud technologique offre la possibilité d'intégrer davantage de transistors donc de construire des circuits numériques plus gros et plus complexes (multicœurs, regroupement d'un processeur générale avec des processeurs spécialisés différents comme un processeur de signal et un processeur graphique, mémoires de types différents, etc.). Le circuit intégré est devenu un système sur puce (*Single on Chip – SoC*) ou un système en boîtier (*Single in Package – SiP*). Le rôle rempli par le logiciel enfoui n'a cessé de croître et continue de croître à mesure que les gravures sont devenues de plus en plus fines. Les investissements consacrés au développement logiciel et à la vérification des systèmes ont augmenté en proportion comme le montre la figure 3. Cela s'explique notamment par la souplesse qu'apporte un programme en donnant de la flexibilité, impossible à atteindre qu'avec du matériel, à des composants jusque là réalisés uniquement en matériel. L'exemple typique est la radio logicielle. Or, les méthodes de conception conjointe établies avant l'an 2000 se sont révélées dépassées. Dans ces méthodes, le logiciel enfoui était conçu et développé indépendamment du matériel. Quant aux modèles matériels obtenus avec les langages de description matérielle, ils sont vérifiés au niveau transfert entre registres (*Register Transfer Level – RTL*) puis synthétisés sans le logiciel enfoui. Pour vérifier le bon fonctionnement du programme et l'adéquation de l'architecture matérielle, il est donc nécessaire de disposer d'un émulateur ou d'un prototype, au moins sur FPGA, pour y charger le logiciel. Ceci implique des coûts supplémentaires en matériels de développement et nécessite de disposer de la description RTL complète. Le test du logiciel enfoui arrive donc tardivement dans le cycle de développement du système. Il en résulte un risque accru de retard de mise sur le marché en cas de difficultés. En outre, face à la complexité croissante et aux temps de développement et de mise de sur le marché de plus en plus courts, la démarche de conception a évolué vers la réutilisation, dans plusieurs produits et dans différentes applications, d'un même squelette d'architecture constitué des mêmes blocs propriétaires tant matériels que logiciels, d'où le besoin d'une flexibilité accrue. [13 ; 14]

² Je parle ici de la production de masse, les démonstrateurs expérimentaux existaient dès les années 1980

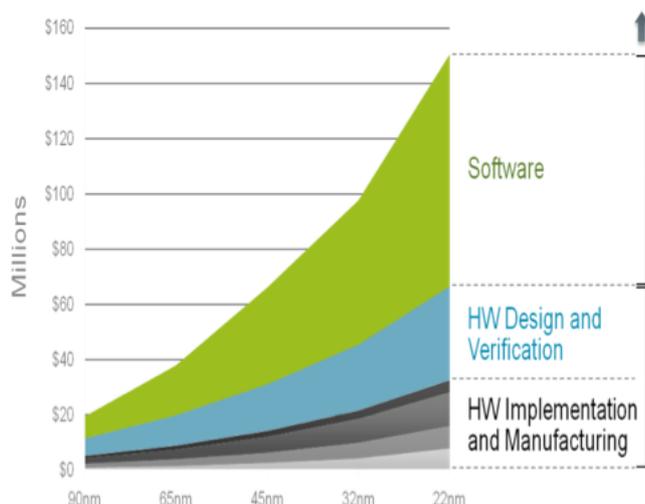


Fig. 3 : Accroissement des coûts consacrés au logiciel et à la vérification en fonction du nœud technologique source [14]

Face à la nécessité d'élever le niveau d'abstraction afin de pouvoir modéliser et vérifier les systèmes et valider les logiciels en tout en améliorant la réemployabilité d'unités fonctionnelles complexes, deux démarches ont été conduites pour exprimer et maîtriser l'interaction logicielle et matérielle : rendre les langages de description matérielle plus abstraits de façon à ce qu'ils puissent donner une vision système tout en étant capable de traiter des séquences d'instructions ou bien, compléter les langages informatiques pour les rendre capables d'appréhender l'architecture matérielle.

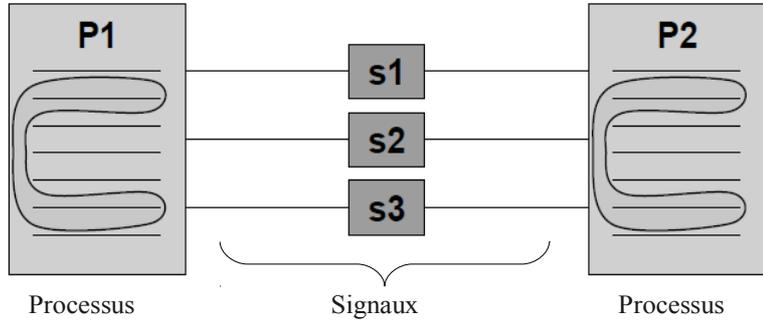
Les langages de description matérielle ont été remaniés de façon à leur adjoindre des caractéristiques typiques de langages informatiques orientés objets : notion de classes et d'objets, d'héritage et de polymorphisme. VHDL a donné naissance au **VHDL orienté objet (OOVHDL)** qui s'articule principalement autour de deux projets qui s'affrontent : Objective VHDL et SUAVE. Le reproche fait à Objective VHDL est, sous prétexte de devoir être synthétisable, d'avoir confondu des questions de sémantique avec des contraintes de réalisation matérielle. L'abstraction n'est pas complète et par conséquent l'utilisation est rendue compliquée par une généralisation insuffisante. Les détracteurs de SUAVE, quant à eux, considèrent que si ce dernier est performant en simulation, il est inadapté pour la synthèse de circuit notamment parce qu'il ne gère pas la concurrence, le parallélisme des processus, de façon convenable. Au-delà de la controverse scientifique, les VHDL orientés objets souffrent du manque criant d'outils de développement et de support technique. Ils ne sont toujours pas utilisés dans l'industrie au moment où sont rédigées ces lignes [15 – 18]. Cependant, grâce à une autre évolution, VHDL peut désormais être utilisé pour vérifier le matériel au moyen de méthodes formelles. En effet, depuis la révision de la norme VHDL, début 2009, le langage de vérification formelle PSL est intégré dans VHDL (IEEE Std. 1076-2008). De son côté, **SystemVerilog**, créé à partir de Verilog, a mieux su répondre à l'attente des concepteurs du matériel d'avoir un langage de description matérielle connu permettant d'appréhender le niveau système. Outre des caractéristiques du langage orienté objet, SystemVerilog a aussi été doté d'une méthode de vérification par assertions. Ce point, conjugué au soutien technique de la part des fournisseurs d'outil de CAO, le rend très attrayant pour mener à bien la vérification du matériel avant l'envoi en fonderie. SystemVerilog est de nos jours largement utilisé dans l'industrie [19 ; 20].

Les efforts pour donner à des langages informatiques la capacité de percevoir l'architecture matérielle se sont surtout portés sur les langages les plus prisés en informatique technique : le C et le C++. Conçues comme des extensions de langages informatiques connus, ces évolutions sont attrayantes pour l'ingénieur système et les concepteurs de logiciels car, d'une part, la prise en main est moins difficile qu'avec un langage totalement inconnu et, d'autre part, parce que les outils de développement existent déjà et dans de nombreux cas ont déjà été acquis par l'entreprise. Handel-C et SpeC ont été construits à partir du C. Handel-C se revendique comme n'étant pas un langage de description matérielle mais plutôt comme un langage capable de traduire des algorithmes de haut niveau directement en structures de portes logiques synthétisables sur FPGA. Il est destiné aux ingénieurs systèmes qui, sans avoir une connaissance approfondie du prototypage sur FPGA, ont besoin d'une plateforme numérique de validation rapide de leurs algorithmes [21 ; 22]. SpeC est un langage de spécification et de conception au niveau système. Il introduit de nouveaux concepts caractéristiques de circuits numériques vus au niveau système : séparation des traitements (calculs)

et des communications, entités de description comportementale interconnectées par des canaux, canaux hiérarchiques et encapsulation des protocoles, description du système avec plusieurs niveaux d'abstraction (voir fig. 4). Les calculs sont effectués dans les entités de description comportementale tandis que les canaux sont chargés de la communication entre ces entités. Les entités peuvent donc être facilement et rapidement remplacées. Plusieurs niveaux d'abstraction peuvent coexister [23].

a) Modèle conventionnel

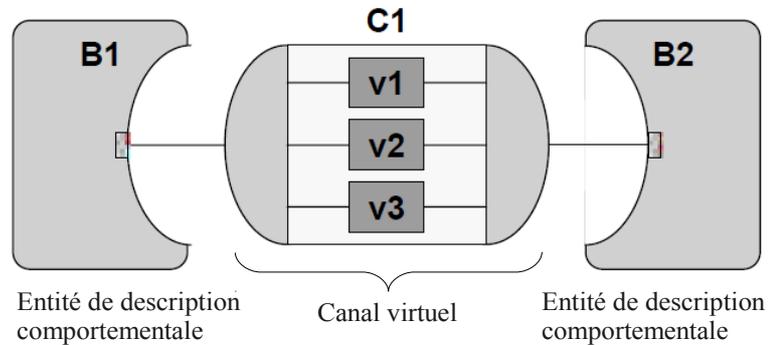
- Processus & signaux
- Mélange des traitements et des communications
- Remplacement automatique impossible à cause de l'imbrication des communications avec le traitement



b) Modélisation système

Modèle pour la spécification et l'exploration d'architecture

- Entités comportementales & canaux
- Séparation des traitements et des communications
- Remplacement des entités ou du canal faisable automatiquement



c) Modélisation système

Modèle pour la réalisation matérielle

- Entités comportementale avec interface de communication
- Séparation maintenue des traitements et des communications
- Disparition du canal et exposition des signaux de communication

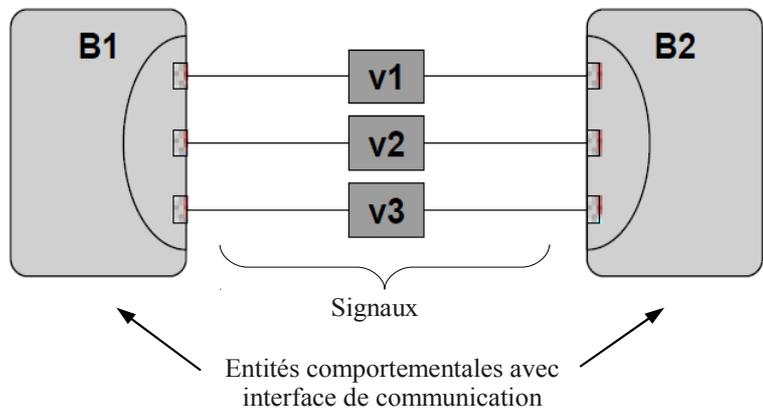


Fig. 4 : Principes clés de la modélisation système d'après [23]

SystemC a été bâti comme une bibliothèque spécialisée de classes C++. Il est donc nativement orienté objet. SystemC intègre lui-aussi tous les concepts de modélisation système déjà évoqués à la figure 4. Le concept de communication par canaux virtuels a été amélioré pour aboutir à la modélisation transactionnelle (*Transaction Level Modeling – TLM*) dans laquelle les communications sont totalement abstraites. Impulsé initialement par deux entreprises majeures de logiciels de CAO, Coware et Synopsys, SystemC a très tôt retenu l'attention de grandes sociétés d'électroniques comme STMicroelectronics et NXP. SystemC a été dès l'origine conçu comme un langage à code ouvert (*open source*) [13]. SystemC a été normalisé par l'IEEE en 2005 (IEEE Std 1666) tandis que l'extension TLM l'a été par le consortium OSCI³ (Open Sys-

³ L'OSCI a fusionnée en 2011 avec l'organisation Accellera pour donner naissance à Accellera Systems Initiative

temC Initiative) en 2009. Avec la version 2.3 de SystemC sortie en 2012, la modélisation transactionnelle est désormais partie intégrante du langage [24]. Ce cadre normalisé garantit l'interopérabilité des outils de développement et l'indépendance des modèles écrits en SystemC vis-à-vis de ceux-ci. Il a certainement contribué à donner une longueur d'avance à SystemC qui connaît un succès grandissant dans l'industrie électronique.

Grâce à la modélisation transactionnelle, l'ingénieur système obtient rapidement un modèle du système en cours de développement. Ce modèle transactionnel est disponible bien avant que la description matérielle du même système soit achevée au niveau transfert entre registres, condition nécessaire pour disposer d'une plateforme matérielle de prototypage sur FPGA. Le modèle transactionnel autorise des simulations suffisamment rapides pour permettre aux architectes logiciels de commencer à construire le logiciel enfoui, à s'assurer que celui-ci est en bonne adéquation avec l'architecture matérielle retenue et qu'il fonctionne correctement. Il sert de prototype virtuel. Quand la plateforme matérielle de prototypage est prête, le modèle transactionnel sert de modèle de référence (*golden model*) pour vérifier l'architecture matérielle. Ce dernier point nécessite l'emploi de vecteurs de test et d'un banc de test capable de les exécuter sur la description RTL. SystemVerilog est apprécié pour cette tâche en particulier parce qu'il est capable de générer des vecteurs de tests aléatoires et de contrôler automatiquement que les résultats obtenus avec ces vecteurs sont corrects. SystemVerilog est aussi bien adapté pour aider les ingénieurs du matériel à avoir une couverture de test suffisante. De nombreuses équipes de développement recourent à la cosimulation SystemC/SystemVerilog pour la vérification conjointe du matériel et du logiciel [13 ; 19 ; 20]. Pour que le modèle transactionnel du système puisse tenir ce rôle central et servir de référence, il doit avoir été lui-même vérifié avec des méthodes formelles. Ce point a été notamment traité par Luca Ferro qui a utilisé une démarche semiformelle (cf. [13]).

Si le tandem SystemC avec SystemVerilog donne satisfaction pour la vérification des systèmes sur puce numériques, il est incapable de prendre en compte l'hétérogénéité, pas même l'interaction qui existe dans les systèmes sur puces à signaux mixtes et analogiques entre les circuits électroniques analogiques et numériques. Or, au cours des années 2000, avec l'avancement des technologies, les systèmes sur puce ou en boîtier qui intègrent des capteurs ou des actionneurs analogiques sont devenus courants. Ce peut être par exemple un téléphone portable avec un appareil photo intégré. Un tel téléphone contient un capteur d'image, l'amplificateur de puissance de l'émetteur radio, un microphone, un amplificateur audio, etc. En outre, l'assistance électronique s'étant généralisée, les études portent sur des systèmes qui ne sont pas nécessairement intégrés mais qui contiennent des éléments appartenant à des domaines physiques différents. Par exemple, un système de suspension active pour automobile contient des éléments mécaniques comme le ressort et l'amortisseur hydraulique passif, un élément électromécanique, la jambe de force qui sert à changer la rigidité de l'ensemble, et un dispositif électronique de pilotage de la jambe de force. Ce dernier est lui-même constitué d'une unité de calcul numérique et de capteurs analogiques pour mesurer les vibrations ou déterminer l'assiette et l'angle de rotation du véhicule. L'unité de calcul exécute un programme enfoui qui évalue les vibrations et asservit la rigidité de la jambe de force. Des outils de CAO spécialisés existent dans chaque discipline, chacun avec une méthodologie particulière à son domaine. La modélisation et la simulation des systèmes hétérogènes a d'abord été abordée par partie, chacune étant élaborée dans son domaine physique et indépendamment des autres. Cette approche ne tient pas compte des interactions entre les différents domaines or celles-ci peuvent avoir un impact fort sur le comportement de l'ensemble du système. Et, elle ne peut plus être négligée dans le cas des systèmes sur puce ou en boîtier comptant plus que quelques unités fonctionnelles différentes [10]. En outre avec la conception par partie, il n'y a aucune vérification de l'interconnexion des différents blocs constitutifs d'un SoC avant l'intégration sur la puce. Les erreurs simples et évitables comme l'interversion des fils dans un bus, des terminaisons laissées non connectées, l'inversion de la polarité d'un signal numérique ou la confusion entre deux plans d'alimentation, sont nombreuses. Il a été estimé qu'elles représentaient 50 % des causes de reprise d'un SoC en 65 nm après retour de fonderie causant un coût supplémentaire de cinq à dix millions de dollars étasuniens et un retard de production de six à huit semaines. Ceci est suffisant pour compromettre le succès d'un SoC [25].

Il est donc devenu indispensable de pouvoir réaliser des simulations conjointes, des cosimulations, en faisant communiquer les différents outils de modélisation et de simulation, ceci dans le cadre cohérent d'une conception descendante. Le simulateur électrique SPICE a été revu de façon à le faire bénéficier des progrès de l'informatique et le rendre plus rapide. Au prix de quelques hypothèses concernant le schéma transistors, celui-ci est divisé en sous parties qui sont exécutées dans des instances différentes du simulateur. Ces instances peuvent être exécutées en parallèle et même réparties sur plusieurs processeurs. Cette révision a donné le jour depuis le début des années 2000 à la génération des simulateurs électriques rapides *Fast Spice*. Selon l'ordinateur utilisé, et notamment en fonction du nombre de cœurs numériques tournant en parallèle,

les simulations sont cinq à cinquante fois plus rapides par rapport au SPICE classique (cf. §II.3), avec une perte de précision inférieure à 5% [26 ; 27]. Cependant, ce gain de vitesse reste faible comparé à vitesse obtenue avec la modélisation numérique au niveau système. Une cosimulation associant un simulateur numériques bâti avec un langage de modélisation au niveau système, comme SystemC ou SystemVerilog, et un simulateur électrique rapide de type *Fast Spice* est pénalisée par la vitesse de simulation insuffisante de ce dernier. La vitesse de cosimulation résultante est trop faible pour autoriser un développement interactif du logiciel enfoui. La simulation sur simulateur électrique reste toutefois toujours une référence du fait de sa grande précision.

Comme cela a déjà été expliqué plus haut, VHDL-AMS et Verilog-AMS sont parfaitement aptes à appréhender l'hétérogénéité. Grâce à la simulation comportementale des systèmes, ils rendent possibles des simulations environ cent fois plus rapides que la simulation électrique classique de type SPICE (voir comparatif au §II.3). Mais, ce gain en vitesse de simulation est encore trop faible pour simuler des systèmes avec une forte dominante logicielle comme dans le cas de liens radio et tout particulièrement lors d'un passage à l'échelle comme dans le cas d'un réseau de capteurs [28]. Il devient nécessaire de pouvoir simuler les parties non-numériques avec un niveau d'abstraction système, de façon à pouvoir simuler un système hétérogène complet avec une vitesse comparable à celle obtenue pour un système purement numérique avec une modélisation transactionnelle.

La modélisation par valeurs réelles (*Real Value Modeling – RVM*) consiste à transcrire les valeurs analogiques au moyen de nombres réels mais à n'utiliser que le simulateur numérique. Les amplitudes des signaux analogiques sont donc continues mais le temps est discrétisé. La simulation traite donc des signaux virtuellement échantillonnés. Puisque seul le simulateur numérique tourne, la simulation est événementielle et la vitesse de simulation des parties analogiques approche celle des parties numériques. Cependant, les valeurs analogiques ne sont mises à jour qu'au moment des événements numériques. La précision est donc très faible. Ce type de modélisation ne peut convenir que dans le cas de simulations peu sensibles à la précision : pour détecter les erreurs d'interconnexions entre les blocs constitutifs du systèmes ou bien pour vérifier le comportement global du système [25 ; 29].

MATLAB/Simulink est un outil de simulation avec un très haut niveau d'abstraction, doté d'un grand nombres de fonctions mathématiques et de méthodes d'analyses mathématiques tout en étant précis. Ses bibliothèques riches et nombreuses en font un outil pluridisciplinaire capable de gérer l'hétérogénéité des systèmes. Les simulations obtenues avec MATLAB/Simulink servent souvent de point de comparaison. Cependant MATLAB/Simulink est un simulateur mathématique. Il n'a pas été conçu pour exécuter des programmes informatiques. Pour l'étude d'un système avec une composante logicielle, il ne peut être employé que dans le cadre d'une simulation conjointe avec un simulateur événementiel numérique comme SystemC ou avec un émulateur. La question de l'échange de données et de la synchronisation entre des outils de natures diverses se pose ici à nouveau. Les mécanismes nécessaires pour répondre à ces questions peuvent se révéler difficile à mettre en œuvre, donc entamer le temps de développement, et de surcroît pas toujours efficaces car consommateurs de temps de calcul. Les vitesses de simulation obtenues avec une plateforme de cosimulation basée sur MATLAB/Simulink sont souvent décevantes. En outre, MATLAB/Simulink ne dispose pas de moyen de représentation matérielle. S'il peut être employé pour commencer l'exploration d'architecture, il est totalement inadapté pour l'implémentation matérielle.

La prise en compte au niveau système des interactions entre les domaines physiques a connu une avancée certaine avec Ptolemy II. La philosophie retenue est que, plutôt que de tenter de créer un nouveau langage englobant toutes les techniques de modélisation, il est plus efficace pour l'ingénieur système de pouvoir utiliser les techniques spécialisées à chaque problématique ou chaque domaine physique. Ceci sera permis grâce à la cosimulation à l'aide des différents modèles de calcul intégrés à Ptolemy II ou par la connexion à des outils spécifiques externes. Par exemple, ADS Ptolemy, la déclinaison de Ptolemy commercialisée par Agilent technologies⁴, est conçu pour s'interfacer avec la suite d'outils Agilent spécialisés pour la radio ou le traitement du signal, mais aussi avec MATLAB. L'approche adoptée est de considérer un système complexe comme une composition hiérarchique de structures de complexité décroissance à mesure que l'on descend dans la hiérarchie. Chacune des parties élémentaires est modélisée à l'aide du modèle de calcul le plus adapté. Une structure fille peut donc être décrite avec un modèle de calcul différent de celui utilisé pour la structure mère qui la contient. Toutefois, si Ptolemy II ou ADS Ptolemy sont bien adaptés pour la création de prototypes virtuels de systèmes hétérogènes, ils ne proposent pas de méthode formalisée pour passer des spécifications simulables sur le prototype virtuel à l'implémentation matérielle, ni pour faire remonter dans les modèles de haut niveau des précisions obtenues lors du raffinement des différentes parties du systèmes. [10 ; 30]

4 A l'origine Ptolemy était un projet de l'Université de Californie à Berkeley

Compte tenu du succès remporté par SystemC notamment dans l'industrie, dès le début des années 2000, il était devenu logique de faire évoluer SystemC. Le premier constat était que malgré l'abondance de solutions spécialisées, il n'existait pas de solution complète et performante pour modéliser et simuler l'hétérogénéité des systèmes. Il fallait combler le fossé entre les descriptions uniquement au niveau système, comme celles obtenues avec Ptolemy II ou MATLAB/Simulink, et les descriptions reflétant la structure matérielle du circuit, celles obtenues avec les langages de description matérielle, tout en incorporant les dépendances entre le logiciel et le matériel. Le tout devait s'exécuter suffisamment rapidement pour permettre le développement du logiciel enfoui. L'objectif était donc de disposer d'un outil de création de prototypes virtuels. La problématique majeure des années 2000 étant la maîtrise des systèmes sur puce communiquant à l'aide d'un lien radio, les différentes propositions d'évolution de SystemC, élaborées en parallèle, ont toutes été structurées initialement autour de la modélisation et de la simulation d'émetteur-récepteurs radio et de communications sans fil entre SoC. [1 ; 2]

L'approche adoptée dans SystemC-WMS (*WMS* pour *Wave Mixed Signal*) présente l'originalité de reposer sur la théorie des ondes et de proposer une méthode pour l'appliquer à tous les circuits à signaux mixtes et analogiques. Les différents modules analogiques communiquent entre eux par échange d'énergie au moyen d'ondes véhiculées le long de canaux. Chaque module reçoit des ondes incidentes, qui font office de signaux d'entrée, et réverbèrent des ondes réfléchies, qui font office de signaux de sortie. A travers chaque canal circulent à la fois l'onde incidente et l'onde réfléchie, comme c'est le cas dans une ligne. Cette approche offre l'avantage de pouvoir résoudre tous les systèmes qui peuvent être décrits, au moins localement, par des équations différentielles ordinaires, même ceux qui impliquent un système d'équations non-linéaires. Puisqu'il s'agit d'échange d'énergie, la méthode peut être employée aussi pour des signaux non-électriques. Cependant, cette approche, bien que très performante et bien adaptée au monde de la radio, est déroutante pour le concepteur qui n'est pas spécialisé en radiocommunication. En effet, le système décrit n'est pas représenté selon le schéma classique d'un assemblage de blocs analogiques reliés directement par des signaux électriques (tensions ou courants). En particulier, puisqu'il faut raisonner en termes d'ondes, la mise en parallèle de plusieurs modules doit être réalisée via des adaptateurs d'onde ou des circulateurs ; la simple dérivation ne convient pas. Toutefois, ce sont les performances de simulation qui ont marqué le glas de cette méthode. En effet, les vitesses de simulation n'ont jamais dépassé celles de MATLAB/Simulink déjà insuffisantes. La raison essentielle de cette lenteur est que le simulateur analogique n'est pas autorisé à calculer de façon indépendante, découplée vis-à-vis du temps simulé, par rapport au simulateur numérique. [10 ; 31]

En 2005, H. Al-Junaid et T. Kazmierski introduisent une extension de SystemC pour les circuits à signaux mixtes et analogiques qu'ils appellent SystemC-A [32]. Des modèles de calcul spécifiques à l'analogique sont ajoutés. Comme leurs homologues numériques, ces nouveaux modèles de calcul autorisent la coexistence de différents niveaux d'abstraction au sein du même modèle. Les différents niveaux d'abstractions possibles s'étendent du niveau système à la liste de composants élémentaires, sous forme de réseau électrique, en passant par la description comportementale analogique. La description de type réseau électrique est obtenue selon l'analyse nodale modifiée (*Modified Nodal Analysis – MNA*) qui utilise le calcul matriciel pour résoudre des systèmes d'équations différentielles. Le solveur analogique de SystemC-A résout les équations différentielles ordinaires ou algébriques. Les interconnexions entre les parties analogiques et numériques sont réalisées à l'aide d'interfaces dédiées. Des composants prédéfinis sont fournis avec le langage, comme la résistance ou le condensateur. SystemC-A contient tout ce qui est nécessaire au concepteur pour créer ses propres composants analogiques (électroniques ou non) à partir d'une classe de base générique. De façon à réduire l'effort de codage, un jeu de méthodes de cette classe de base crée automatiquement les équations différentielles nécessaires à la construction du nouveau composant. Il peut s'agir de composants complexes. Les performances de SystemC-A sont démontrées dans [32 ; 33]. La modélisation d'un papillon de Lorentz, illustrant la théorie du chaos, met en évidence la capacité du langage à la forte abstraction nécessaire à la modélisation système. Les cas d'études d'une alimentation à découpage, d'une boucle à verrouillage de phase (*Phase Locked Loop – PLL*) et d'un oscillateur contrôlé en tension (*Voltage Controlled Oscillator – VCO*) démontrent la possibilité d'obtenir des descriptions comportementales ou sous forme de réseau électrique pour des systèmes électriques à signaux mixtes. La modélisation d'un système de suspension active pour siège de véhicule automobile prouve l'aptitude de SystemC-A à appréhender la complexité d'un système hétérogène relevant de plusieurs domaines de la physique (mécanique, électromécanique, électronique). Pour la PLL, SystemC-A simule deux à quatre fois plus vite que le modèle équivalent en VHDL-AMS. Dans le cas de la suspension active, l'erreur relative du modèle SystemC-A par rapport à son équivalent en VHDL-AMS, est inférieure à 3 % dans le plus mauvais cas. Toutefois, pour obtenir de bonnes performances le couplage du solveur analogique et du solveur numérique a été effectué en modifiant le noyau SystemC. Or, 2005 est l'année au cours de laquelle SystemC a été normalisé. Cette modifica-

tion du noyau signifie que la composant numérique de SystemC-A n'est pas conforme à la norme. Cette non-conformité handicape cette solution [10].

SystemC-AMS est un projet mené par l'OSCI avec la formation, dès 2002, d'un groupe de travail consacré au développement d'une extension de SystemC pour la description et la simulation des systèmes hétérogènes mixtes et analogiques (*OSCI AMS Working Group – AMSWG* [34]) et la publication en 2003 des premières spécifications de cette extension. Initialement, SystemC-AMS ne comportait que le modèle de calcul de type réseau électrique, conservatif et bâti sur l'analyse nodale modifiée, et le modèle de calcul de type flot de données temporalisé qui autorise une construction algorithmique des modèles analogiques. Le modèle de calcul de type flot de signal, non-conservatif, a été ajouté plus tard pour faciliter la description des systèmes asservis contenant des boucles de rétroaction. Dès le début, le projet a été animé par la volonté d'obtenir un langage faisant le consensus des concepteurs et qui soit indépendant des implémentations. La normalisation de SystemC-AMS est une mission de l'AMSWG à partir de 2006. Cependant, face à l'ampleur de la tâche, il est décidé de procéder par étapes en se focalisant sur les demandes considérées comme prioritaires par les industriels. La première norme ne concernera que les systèmes électriques à signaux mixte et analogiques. Les solveurs analogiques fournis avec le langage ne résoudront que les systèmes d'équations linéaires. Les comportements non-linéaires peuvent cependant être modélisés au moyen d'algorithmes dans le cadre du modèle de calcul de type flot de données temporalisé. Pour des raison de simplicité et d'efficacité, la simulation sera à pas de temps fixe [3 ; 35]. La version 1.0 du standard est publiée par l'OSCI en 2010 [5]. Malgré les restrictions énoncées, SystemC-AMS a été conçu et construit pour lui permettre d'évoluer et d'intégrer progressivement les éléments manquants. Il n'est pas directement intégrer à SystemC mais interfacer avec celui-ci à travers une couche intermédiaire dont le rôle premier est de synchroniser le domaine analogique de SystemC-AMS avec le domaine numérique de SystemC. Cette couche de synchronisation (*synchronization layer*) remplit plusieurs rôles (voir fig. 5) :

- elle garantit la séparation de SystemC et SystemC-AMS et donc l'intégrité de SystemC. Elle permet également à ceux-ci d'évoluer indépendamment tant que l'interface est respectée.
- elle autorise les solveurs numériques et analogiques à progresser séparément dans la simulation, ce qui est crucial pour concilier vitesse et précision de simulation, mais elle assure une bonne synchronisation entre eux aux moments opportuns. Les méthodes pour l'échange de données entre les domaines numériques et analogiques sont fournies avec SystemC-AMS.
- elle donne de la souplesse et de l'évolutivité à SystemC-AMS puisque le concepteur peut ajouter l'extension de son choix à SystemC-AMS en la connectant à la couche de synchronisation. Il peut s'agir d'une extension spécialement créée pour ce besoin ou utilisant un solveur spécialisé non fourni par le standard.

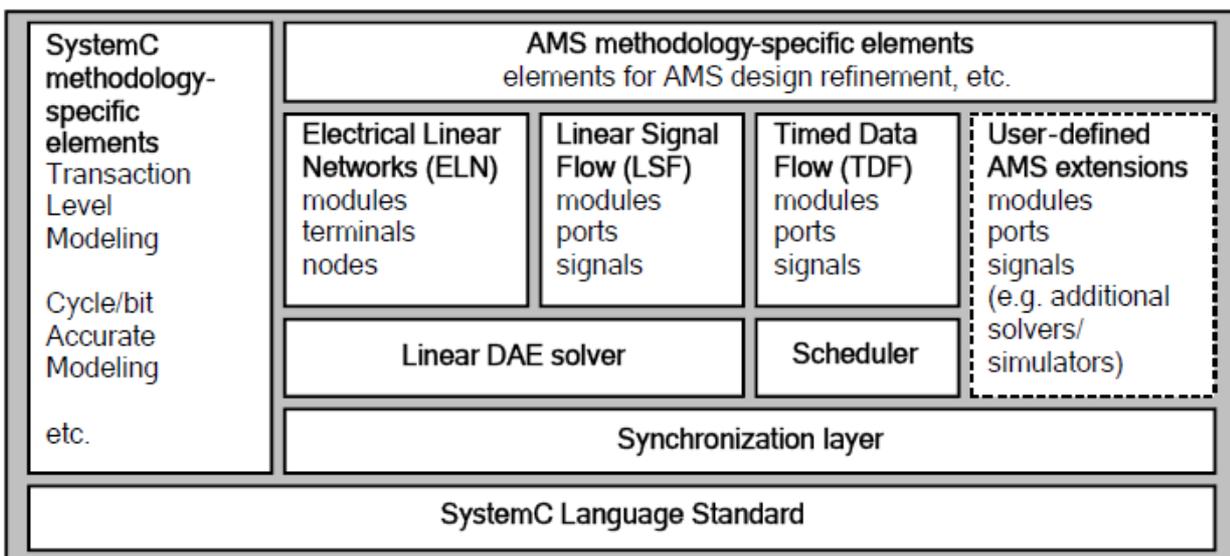


Fig. 5 : SystemC-AMS et son principe d'interfaçage avec SystemC source [1]

L'évolutivité de SystemC-AMS rendue possible grâce à la couche intermédiaire de synchronisation est une différence majeure avec SystemC-A. Toutefois, la version 1.0 du standard SystemC-AMS ne normalise pas la couche de synchronisation ni l'interfaçage avec les solveurs qui restent donc spécifiques à l'implémentation et non documentés. Développer une extension personnalisée est de ce fait une tâche ardue. Elle a cependant déjà été accomplie plusieurs fois, comme le montrent, par exemple, l'extension créée par T. Uhle et K. Einwich [36] ou celle proposée par T. Mähne [10].

L'extension de T. Uhle et K. Einwich est construite autour d'un solveur capable de résoudre des systèmes d'équations différentielles non-linéaires. Le cas d'étude d'un lève-glace électrique montre la capacité de cette extension à maîtriser un système à la croisée de plusieurs domaines de la physique (électronique, mécanique, magnétisme) tout en restituant les effets non-linéaires comme le pic de courant qui apparaît lorsque le moteur est bloqué par la vitre arrivant en butée. [36]

Torsten Mähne a élaboré son extension, qu'il a baptisée SCAX, en s'appuyant sur la théorie des graphes des liens⁵ (*Bond Graph*) au formalisme universel. Grâce à la mise en œuvre de cette théorie qui s'intéresse à la circulation des flux d'énergie entre les éléments du système, il propose d'abstraire les domaines physiques pour se focaliser sur les interactions entre ceux-ci. Il est de la sorte possible de rapidement établir un prototype virtuel d'un système couvrant plusieurs domaines de la physique sans être obligé de solliciter un expert de chacun de ces domaines. [10]

Les deux extensions précédentes ajoutent également un moyen de spécifier la nature des quantités échangées entre les éléments constitutifs du système, de façon comparable à ce qui existe dans VHDL-AMS. Ceci permet dès la compilation de détecter des aberrations dues à des erreurs de connexion. Il est par exemple interdit de connecter une vitesse avec une température, ce qui n'a aucun sens mais qui est possible en SystemC-AMS standardisé s'il s'agit du même type au sens de C++ (typiquement des grandeurs réelles portées par un type `double`). L'extension SCAX contient un mécanisme de contrôle supplémentaire qui vérifie la cohérence de l'unité de mesure employée. Une unité fonctionnelle délivrant une vitesse en kilomètres par heure ne peut pas être connectée directement sans convertisseur à une unité fonctionnelle qui attend en entrée une vitesse en miles par heure. [10 ; 36]

La possibilité de réaliser une modélisation transactionnelle d'un système à signaux mixtes et analogiques dans son ensemble en employant SystemC-AMS a été montrée par M. Damm, C. Grimm, J. Haase et leurs collaborateurs dans [37] et [38]. Le cas d'étude est un système de communication numérique. La partie numérique est modélisée avec le modèle de calcul SystemC-TLM LT (*Transaction Level Modeling Loosely Timed*) tandis que la partie analogique est modélisée avec le modèle de calcul SystemC-AMS TDF (*Timed Date Flow*). Le système est donc entièrement décrit à l'aide des modèles de calculs les plus abstraits disponibles dans SystemC et SystemC-AMS. Le découplage temporel des différentes unités fonctionnelles entre elles, qu'elles soient numériques ou analogiques, permet une utilisation efficace des deux modèles de calcul et donc d'obtenir une simulation performante. Toutefois, le décalage temporel entre les unités fonctionnelles doit être contenu pour éviter un nombre important d'erreurs dans l'ordre d'arrivée des paquets.

Le gain de performance qui peut être obtenu grâce à la modélisation transactionnelle des parties analogiques en employant SystemC-AMS a été mis en évidence dans plusieurs publications dont [28 ; 39 – 41].

Dans le domaine de la transmission par lien radio, le niveau d'abstraction est élevé en considérant que les traitements sont à effectuer en bande de base, ce qui équivaut à faire abstraction de la fréquence de la porteuse et à translater le signal en bande de base. L'efficacité de cette méthode a été établie par M. Vasilevski, F. Pêcheux et leurs collaborateurs, avec la description d'un même émetteur-récepteur modélisé avec MATLAB et avec SystemC-AMS [28]. Un premier modèle a été construit avec le modèle de calcul de type réseau électrique de SystemC-AMS (équivalent du MoC ELN⁶). C'est un modèle conservatif élaboré avec l'approche traditionnelle : la transmission radio est simulée à la fréquence de la porteuse avec un pas de temps permettant d'appréhender une fréquence de 2,4 GHz. Le gain de performance par rapport à MATLAB est de l'ordre de deux. Un autre modèle a été réalisé avec le modèle de calcul de type réseau électrique de SystemC-AMS (équivalent du MoC TDF⁶). Cette fois, la méthode de la translation en bande de base est appliquée. La simulation avec ce modèle est plus de onze cents fois plus rapide qu'avec le modèle conservatif qui n'utilise pas la translation en bande de base (soit plus de deux mille fois plus rapide que MATLAB). Appliquée à un réseau de capteurs sans fil, la méthode de la translation en bande de base permet d'obtenir l'évaluation de l'évanouissement du canal dix fois plus rapidement avec SystemC-AMS qu'avec MATLAB, faisant passer à dix minutes une simulation qui durait auparavant plus d'une heure et demie [40].

5 Selon les auteurs, la traduction de *Bond Graph* est « graphe de liaisons » ou « graphes des liens »

6 La publication date de 2008 et est donc antérieure à la normalisation de SystemC-AMS. Les dénominations employées à l'époque n'étaient pas celles de la norme.

Une démarche d'abstraction d'un principe comparable à la translation en bande de base a été menée par Fabio Cenni pour la modélisation d'un capteur d'image [41]. Un premier modèle conservatif est écrit en SystemC-AMS avec le modèle de calcul ELN selon la démarche habituellement utilisée avec du VHDL-AMS : tous les pixels du capteurs sont modélisés individuellement avec trois photodiodes chacun (une pour le rouge, une pour le vert et une pour le bleu). Le signal en sortie de chaque photodiode est évalué plusieurs fois au cours de l'exposition du capteur à la lumière. Le niveau d'abstraction est élevé en considérant le capteur comme n'étant constitué que d'un seul pixel qui balaye la zone image et en n'évaluant l'intensité lumineuse qu'une seule fois à la fin de l'exposition. Ce modèle non-conservatif plus abstrait est écrit avec modèle de calcul TDF de SystemC-AMS. Il autorise une simulation plus de soixante-dix-mille fois plus rapide que son homologue conservatif en SystemC-AMS ELN et plus de deux millions et demie fois plus rapide que son pendant en VHDL-AMS. Ces gains en vitesse de simulation rendent possible la mise au point du logiciel enfoui de correction d'image en sortie de capteur sur un prototype virtuel qui est disponible dès le début du projet. Il n'est plus obligatoire d'attendre la fabrication d'un prototype matériel de capteur d'image pour commencer le développement dudit logiciel enfoui.

SystemC-AMS suscite un intérêt croissant auprès des industriels comme des chercheurs comme l'a montré la journée SystemC-AMS de 2011 à Dresde [42]. Encouragé par ce succès, l'Accellera Systems Initiative a entrepris de poursuivre l'évolution de SystemC-AMS en fonction de la demande des utilisateurs. L'apport majeur de la prochaine version sera de pouvoir mener des simulations avec un pas de temps variable (contrairement à la version actuelle à pas de temps fixe). Cette proposition d'évolution est déjà concrétisée par la publication en mars 2012 d'une ébauche officielle d'un nouveau manuel de référence de langage [43].

II.3) Comparatifs entre les langages de modélisation analogique

Des comparaisons entre les langages de modélisation analogiques sont régulièrement publiées dans la littérature. Quelques unes d'entre elles sont évoquées ici afin de positionner SystemC-AMS par rapport au langages de modélisation analogique les plus employés au moment où sont écrites ces lignes.

La figure 6 met en avant les gains en vitesse apportés par SystemC-AMS qui autorise des simulations plus rapide qu'avec les autres solutions. L'utilisation de SystemC-AMS rend possible la création de prototypes virtuels analogiques.

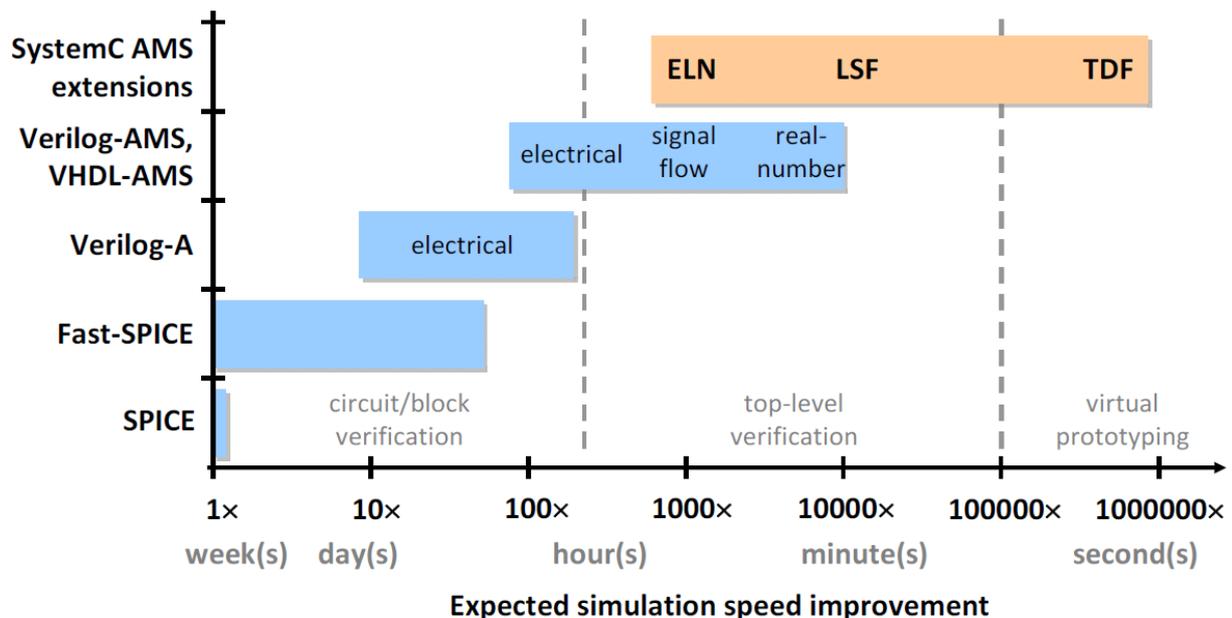


Fig. 6 : Vitesse de simulation des langages de modélisation analogique les plus courants
– en vitesse relative par rapport à SPICE
source [2]

La figure 7 montre le compromis qui peut être atteint par différentes solutions de modélisation entre précision des modèles analogiques et vitesse de simulation. SystemC-AMS se positionne entre les langages de modélisations analogiques conventionnels et les solutions numériques. Le schéma a été construit en prenant SPICE comme référence par conséquent un indices de précision de 100% et un indice de vitesse de simulation de 1 lui sont attribués.

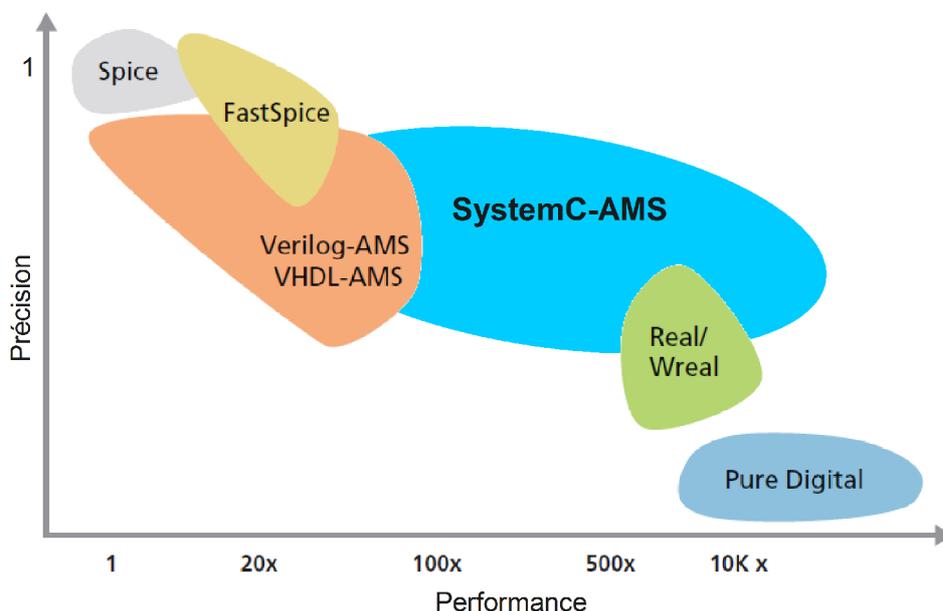


Fig. 7 : Précision des modèles analogiques en fonction de la vitesse de simulation pour différents langages de modélisation – en valeurs relatives par rapport à SPICE, d'après [25]

La figure 8 indique les niveaux d'abstraction accessibles et les méthodes de calcul disponibles pour les langages de modélisation les plus utilisées. Ce graphique a été dessiné en 2005 par les créateurs de SystemC-A [32]. Il montre la portée étendue de ce langage par rapport aux autres. SystemC-AMS a une portée tout à fait comparable à celle de SystemC-A, un graphique comparable pourrait être établi avec SystemC-AMS.

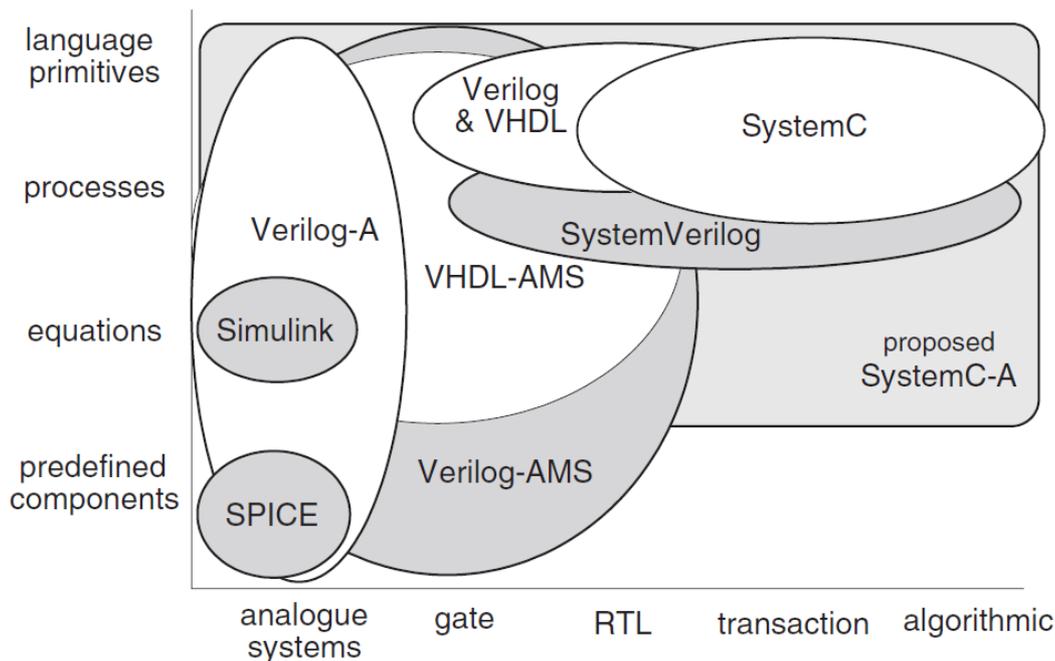


Fig. 8 : SystemC-A et autres langages de modélisation à différents niveaux d'abstraction source [32]

Dans un article de 2007, traitant de l'hétérogénéité dans les systèmes sur puce ou en boîtier, I. O'Connor, B. Courtois, K. Chakrabarty et M. Hampton établissent un bilan des différents langages de modélisation et moteurs de simulation disponibles aux différents niveaux d'abstraction pour les différents domaines physiques utilisés dans ces systèmes [44]. Le tableau 3 reproduit ce bilan qui donne un aperçu des usages avant le projet BDREAMS.

Niveau d'abstraction	Domaine physique							
	Logiciel	Électronique numérique	Électronique analogique	Radiofréquence	Mécanique	Optique	Fluidique	Chimie
Service	CORBA							
Transaction	SystemC / UML		SystemC-AMS					
Macroarchitecture	SystemC		Ptolemy			SystemC		
Microarchitecture	SystemC / VHDL							
Bloc	<i>Sans objet</i>	VHDL	VHDL-AMS	Simulation RF VHDL-AMS	VHDL-AMS		VHDL-AMS	
Circuit	<i>Sans objet</i>	Simulation électrique		Simulation RF				
Physique	<i>Sans objet</i>	Méthodes des éléments finis				Différences finies	FEMLab	Méthodes analytiques

Solution industrielle
 Solution de laboratoire
 Pas de solution connue

Tableau 3 : Répartition, avant le projet BDREAMS, de l'usage des langages de modélisation dans différents domaines physiques à différents niveaux d'abstraction d'après [44]

Les contributions apportées par les différents partenaires du projet BDREAMS ont démontré que SystemC-AMS était désormais une solution industrielle pour l'étude des composantes analogiques et radiofréquence des systèmes embarqués. L'étude consignée dans cette thèse au chapitre V montre qu'il est possible d'étendre l'usage de SystemC-AMS au niveau bloc analogique et ainsi améliorer la progression dans le raffinement des modèles. Des travaux extérieurs au projet BDREAMS, notamment ceux de T. Uhle et K. Einwich dans [36] et ceux de T. Mähne dans [10], étendent SystemC-AMS pour le rendre apte à la modélisation dans tous les domaines de la physique utilisés dans les systèmes hétérogènes. Le tableau 4 est la correction du tableau 3 pour intégrer toutes ces contributions.

Niveau d'abstraction	Domaine physique							
	Logiciel	Électronique numérique	Électronique analogique	Radiofréquence	Mécanique	Optique	Fluidique	Chimie
Service	CORBA							
Transaction	SystemC / UML		SystemC-AMS		SystemC-AMS			
Macroarchitecture	SystemC		SystemC-AMS / Ptolemy		Ptolemy	SystemC		
Microarchitecture	SystemC / VHDL							
Bloc	<i>Sans objet</i>	VHDL	SystemC-AMS	Simulation RF VHDL-AMS	VHDL-AMS		VHDL-AMS	
			VHDL-AMS					
Circuit	<i>Sans objet</i>	Simulation électrique		Simulation RF				
Physique	<i>Sans objet</i>	Méthodes des éléments finis				Différences finies	FEMLab	Méthodes analytiques

Solution industrielle
 Solution de laboratoire
 Pas de solution connue

Tableau 4 : Répartition actualisée de l'usage des langages de modélisation dans différents domaines physiques à différents niveaux d'abstraction d'après [44] et [41]

II.4) Ce qui ressort de l'état de l'art

L'état de l'art a montré l'existence d'un nombre conséquent de possibilités pour la modélisation des systèmes mixtes et analogiques. Avec les modèles de calcul les plus abstraits de SystemC et de SystemC-AMS, il est possible d'obtenir en un temps record un prototype virtuel du système complet avec une modélisation transactionnelle des parties numériques et des parties analogiques. Avec ce prototype virtuel, il est possible d'entreprendre le développement du logiciel enfoui dès le début du projet sans devoir attendre la fabrication d'un prototype matériel des parties analogiques. Cependant, la question de savoir que faire une fois le prototype virtuel créé et une fois la vérification fonctionnelle du système effectuée sur ce prototype, est insuffisamment posée. Il n'émerge pas de démarche structurée et générale de conception des systèmes hétérogènes à signaux mixtes et analogiques qui intègre une méthode de raffinement des modèles dans une démarche descendante et coordonnée entre parties numériques et analogiques. Une telle démarche réduirait les risques d'erreurs et faciliterait la réutilisation, en tant que blocs matériels propriétaires (*IP – Intellectual Property*), de blocs analogiques conçus dans le cadre d'un projet précédent ou issus d'une société tierce. L'objectif de cette thèse est proposer une telle méthode.

Chapitre III)

Une méthodologie pour la conception des systèmes mixtes et analogiques

Les prototypes virtuels au niveau système sont nécessaires pour faire face à la complexité des systèmes intégrés mais ils ne sont pas une fin en soi : le consommateur veut un produit réel pas une simulation. Une méthodologie est indispensable pour organiser le passage des niveaux les plus abstraits du système virtuel aux détails les plus fins de l'électronique du système réel. Elle doit mettre en œuvre un processus de raffinement descendant qui, à partir des spécifications, progresse à travers les trois phases principales de la conception d'un circuit :

- l'exploration d'architecture,
- l'implémentation de l'architecture sélectionnée,
- la réalisation de l'électronique correspondant à l'implémentation élaborée, par le dimensionnement et le tracé des composants élémentaires.

Ces phases se décomposent en une succession d'étapes. A chaque étape correspond un niveau d'abstraction. Plus le niveau d'abstraction est bas et plus la description est détaillée.

Les efforts déployés au cours des dernières décennies pour améliorer le flot de conception des circuits numériques ont abouti à la mise au point d'un processus de raffinement descendant efficace qui répond aux critères énoncés ci-dessus. La conception d'un système numérique dans un temps relativement court est désormais possible malgré sa complexité. Durant la même période, le flot de conception analogique n'a pas connu les mêmes améliorations et la conception des circuits mixtes et analogiques est freinée par le manque de niveaux d'abstraction élevée qui autoriserait un processus de raffinement descendant progressif et structuré. Les analogiciens⁷ passent, en général, directement des spécifications des parties analogiques à une description de chacune de celles-ci sous forme d'une composition de structures élémentaires le plus souvent décrites au niveau transistor. Le risque de discordance entre les spécifications et le circuit créé est élevé. Le risque d'un interfaçage difficile voire impossible avec les parties numériques du système est important.

En s'inspirant des niveaux d'abstraction analogiques identifiés dans [45] & [46], en les réarrangeant et en établissant une correspondance entre niveaux d'abstraction analogiques et niveaux d'abstraction numériques, le présent chapitre introduit une méthodologie pour la conception descendante des circuits mixtes et analogiques. La figure 9 résume cette méthodologie en présentant les étapes du processus de raffinement à dérouler au cours des différentes phases de la conception descendante. La correspondance entre domaines numériques et analogiques est mise en évidence. Le principe clé, repris du raffinement descendant des parties numériques, est qu'après raffinement, un niveau d'abstraction doit être conforme aux spécifications reçues du niveau d'abstraction supérieur et doit servir de spécification au niveau d'abstraction inférieur. Cette démarche structurée est adaptée à la cosimulation des parties analogiques et numériques ainsi qu'à la gestion des interfaces entre ces parties. Les non-linéarités dominantes des parties analogiques ou les effets analogiques dans les parties numériques peuvent être pris en compte à des niveaux d'abstraction relativement élevés. Leurs conséquences sur le comportement du système peuvent donc être évaluées très tôt dans le cycle de conception.

⁷ Analogicien : expert en conception de circuits analogiques

Le concept de signal est essentiel pour effectuer la description des parties analogiques. Les signaux fonctionnels sont reliés aux spécifications fonctionnelles. Ils sont utilisés tout au long du processus de raffinement depuis les niveaux les plus abstraits jusqu'aux niveaux les plus détaillés. Les contraintes non-fonctionnelles sont requises pour obtenir les descriptions les plus détaillées. Leur impact sur le comportement détermine le niveau de raffinement auquel elles doivent être prise en compte. Plus il est important et plus le niveau d'abstraction où elles doivent être considérées est élevé. Ce point est inhabituel par rapport au processus classique de raffinement numérique. Une différence majeure entre conception numérique et analogique est que dans le domaine analogique les signaux sont continus en temps et en amplitude. Dans le domaine numérique, la forme rectangulaire des signaux numériques et leurs deux niveaux logiques, '0' et '1', constituent une abstraction très forte qui masque ce que l'analogue cherche à maîtriser ou à contrôler : l'amplitude réelle, le taux de montée ou de descente, les dépassements, les temps d'établissement, les oscillations résiduelles, etc. Par exemple la saturation de la tension de sortie d'un amplificateur est due à l'impossibilité de cette tension à dépasser la tension d'alimentation de l'amplificateur. Or, la saturation a un impact fort sur le comportement du module dans lequel l'amplificateur est inséré. Par conséquent, la tension d'alimentation de l'amplificateur devra être prise en compte avant même que sa structure interne soit définie.

L'exposé est conduit en s'appuyant sur la figure 9. Un dictaphone, un système portatif d'enregistrement de la parole, servira d'exemple illustratif pour étayer le propos.

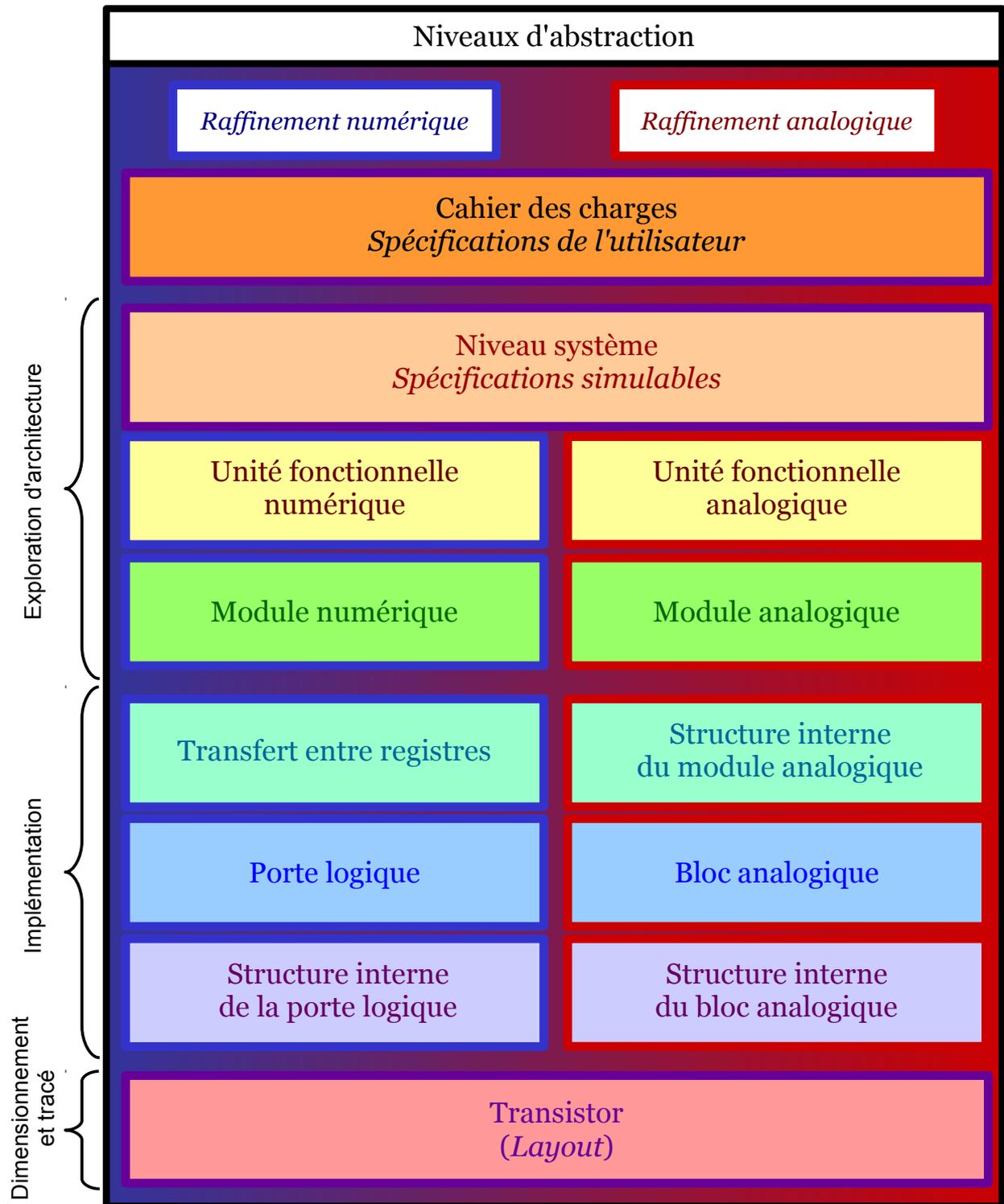


Fig. 9 : Raffinement structuré pour les systèmes mixtes

III.1) Cahier des charges et spécifications

Le cahier des charges indique ce que veut le client. Il sert à exprimer le besoin de l'utilisateur et les usages attendus du futur système. Dans le cas de l'exemple illustratif, celui-ci peut se résumer à : système portatif d'enregistrement de la parole.

Les spécifications décrivent comment doit être le système. Elles peuvent être divisées en deux catégories : les spécifications fonctionnelles et les spécifications non-fonctionnelles. Par les spécifications fonctionnelles, le client explique comment le futur produit doit se comporter. Les spécifications non-fonctionnelles sont des exigences qui ne relèvent pas du comportement mais qui contraignent la conception. Dans l'exemple du dictaphone, parmi les spécifications fonctionnelles, le client peut exiger l'existence de plusieurs boutons, un pour le marche/arrêt, un activer ou suspendre l'enregistrement ou encore, un pour indiquer le début d'un nouvel enregistrement ou encore un pour effacer. Il peut aussi préciser la présence d'un indicateur montrant le taux de remplissage de la mémoire d'enregistrement et d'un voyant signalant que la source d'énergie est faible. Les contraintes des spécifications non-fonctionnelles sont que l'appareil est destiné au grand public, il doit donc être d'un coût réduit et doit fonctionner avec des piles du commerce. L'appareil doit pouvoir être placé dans une poche. Il doit pouvoir être utilisé durant au moins six mois sans changer les piles à raison de deux heures d'enregistrement par jour en moyenne. La fonction première étant l'enregistrement des voix humaines, il est autorisé de ne pas conserver les fréquences supérieures à 4 kHz. Le transfert des enregistrements doit pouvoir se faire sur n'importe quel ordinateur de bureautique par un port *USB*.

III.2) Exploration d'architecture

L'exploration d'architecture consiste à déterminer la structure que devra avoir le système de façon à remplir le cahier des charges en correspondant le plus exactement possible aux spécifications. Cette exploration s'effectue en parcourant trois niveaux d'abstraction : le niveau « système », le niveau « unités fonctionnelles » et le niveau « modules ».

III.2.1) Niveau système

Le niveau système est le niveau d'abstraction le plus élevé. Le système est vu comme un ensemble monolithique qui est caractérisé par des spécifications fonctionnelles. Le processus de raffinement descendant commence par une traduction du cahier des charges et des spécifications fonctionnelles en un modèle exécutable* sur ordinateur qui abstrait fortement le comportement et masque tous les détails d'implémentation. Ce modèle est comme un prototype virtuel qui répond au besoin de spécifications simulables. Il facilite la compréhension entre le client et la société chargée de la conception du système.

Les systèmes purement analogiques sont désormais très rares voire quasi inexistantes. Dans le même temps, les industriels de la microélectronique cherchent à intégrer un nombre de fonctions toujours plus grand. Dans les systèmes modernes, les fonctions par essence analogiques, comme l'acquisition d'image ou d'une mesure de pression ou de température, côtoient des fonctions réalisées numériquement, comme l'amélioration d'une image par traitement numérique ou encore la compression de données. Il en résulte que les spécifications du système ont à la fois des composantes analogiques et numériques. Il est de ce fait cohérent de ne pas faire le partitionnement numérique/analogique au niveau système et par conséquent de travailler avec un modèle simulable qui ne fait pas cette distinction.

Le schéma de la figure 10 donne le principe de fonctionnement du dictaphone de l'exemple. L'interface Homme-Machine (IHM) recouvre les boutons, le témoin piles basses et la jauge indiquant le taux de remplissage de la mémoire. Parmi les spécifications non-fonctionnelles, l'exigence de faible coût amène à introduire un mécanisme de compression de données dans le but de minimiser la taille de la mémoire. La demande d'un remplacement espacé des piles ajoutée à la contrainte d'un faible volume et d'un faible poids (« L'appareil doit pouvoir être placé dans une poche ») conduisent à ajouter un système d'optimisation de la vitesse de traitement afin de réduire la consommation de l'unité de traitement. Avec une consommation réduite, le volume et le poids des piles pourront être diminués.

* Voir glossaire

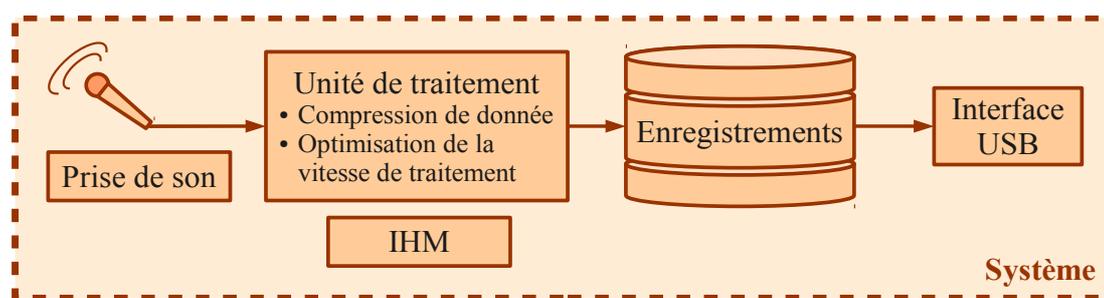


Fig. 10 : Le dictaphone au niveau système

III.2.2) Unités fonctionnelles

La description au niveau système est enrichie par l'apparition des unités fonctionnelles. Une unité fonctionnelle remplit une fonctionnalité système. La modélisation au niveau « unités fonctionnelles » sert à proposer une architecture sommaire pour le système de façon à commencer le partitionnement matériel/logiciel, parties analogiques/numériques et aussi pour évaluer la charge de travail et la répartition entre les équipes. La simulation permet de s'assurer qu'avec cette ébauche d'architecture le système respecte ou non les spécifications et si une nouvelle architecture doit être élaborée.

Afin de répondre au besoin actuel de modularité et de réutilisation, les unités fonctionnelles doivent être conçues de façon à être autonomes et interchangeables. Pour garantir cette propriété, des précautions doivent être prises au niveau des interfaces. Ceci est particulièrement sensible pour les unités fonctionnelles analogiques pour lesquelles des questions d'adaptation d'impédance peuvent se poser. Les interconnexions entre les unités fonctionnelles doivent donc être exprimées mais de façon abstraite. Les communications numériques sont sous forme de transactions où seul le flot de données à travers un canal est considéré. Les échanges entre unités fonctionnelles analogiques sont modélisés par des signaux abstraits. Seuls les signaux fonctionnels, c'est-à-dire les signaux nécessaires pour expliquer le fonctionnement du système sont pris en compte, les autres sont masqués. Dans le cas d'une connexion entre une unité analogique et une unité numérique, une unité de conversion doit être mise en place. Au cours du raffinement, le concepteur devra veiller, d'une part, à préserver l'autonomie de chaque unité fonctionnelle à chaque étape afin de ne pas perdre l'interchangeabilité et, d'autre part, à transmettre les signaux fonctionnels tout au long du processus de raffinement.

Un processeur de signal (*Digital Signal Processor – DSP*) ou une mémoire sont des exemples d'unités fonctionnelles numériques. Une unité d'acquisition d'un signal physique, comme le son, l'image, la température ou la pression est un exemple d'unité fonctionnelle analogique.

Les unités fonctionnelles ne sont décrites qu'en termes de fonctionnalité et de comportement. Par exemple, le concepteur exprimera à ce stade le besoin de mémoriser des données mais ne précisera pas la nature de la mémoire à utiliser. Cela pourrait être aussi bien une mémoire électronique qu'un disque dur. De même, dans le domaine analogique, une unité capable de mesurer une grandeur physique doit apparaître dans la description du système mais sans donner de détails sur ses éléments constitutifs, en particulier sans indiquer ni le nombre, ni le type de capteur à utiliser. Les unités numériques sont modélisées à l'aide d'algorithmes tandis que les unités analogiques sont modélisées à l'aide de fonctions de transfert ou d'équations.

La figure 11 illustre ce que pourrait être le dictaphone au niveau unités fonctionnelles. Pour des questions de flexibilité, la compression de données sera réalisée logiquement. Le son devra donc être numérisé. Une interface analogique-numérique est nécessaire. Les premiers aspects des partitionnements logiciel-matériel et numérique-analogique apparaissent. Afin d'optimiser l'énergie, le cadencement de l'unité de traitement doit être réduit à chaque fois que cela est possible. En effet, la puissance absorbée par un processeur CMOS est fonction de sa fréquence de cadencement et de sa tension d'alimentation [47 ; 48]. Or, un discours ou une conversation comportent toujours des moments de silence. En outre, en fonction de la richesse harmonique du signal, le travail de compression peut s'avérer plus ou moins important. L'unité de traitement logiciel doit donc évaluer à chaque instant l'activité du signal numérisé par l'unité de capture du son ainsi que l'importance du travail de compression à effectuer. À partir de cette évaluation, elle délivre une information de charge de travail à l'unité de gestion du cadencement. Cette dernière transforme l'information de charge de travail en une consigne de fréquence qu'elle envoie au générateur d'horloge. Le générateur d'horloge synthétise alors un signal d'horloge à la fréquence requise. Ce signal cadence le processeur de l'unité de traitement.

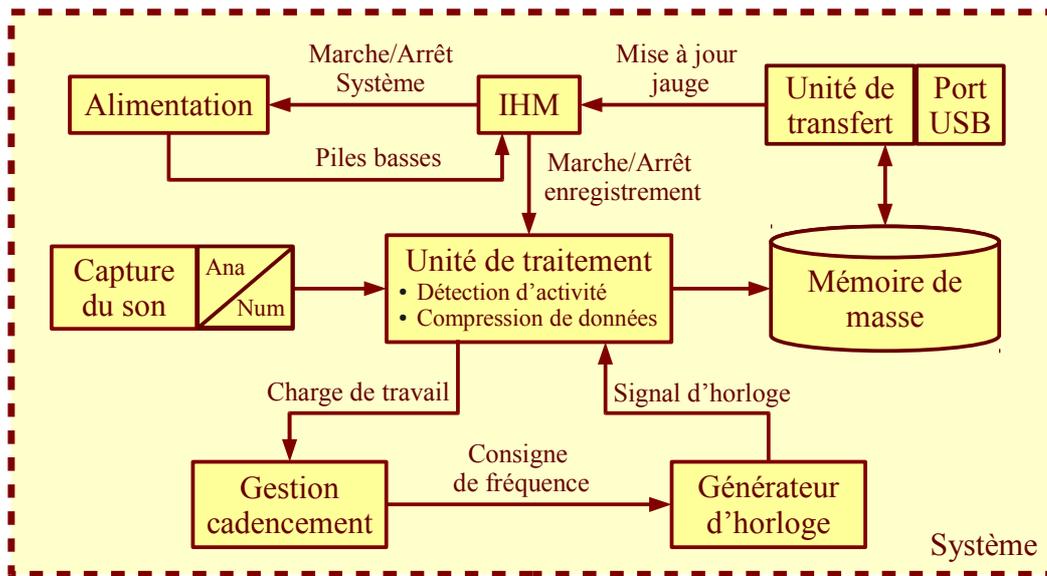


Fig. 11 : Le dictaphone au niveau unités fonctionnelles

III.2.3) Modules

Les modules sont les éléments constitutifs des unités fonctionnelles.

Un capteur, un convertisseur analogique vers numérique (CAN) ou numérique vers analogique (CNA) ou encore un filtre analogique sont des exemples de modules analogiques. Une unité arithmétique et logique (*Arithmetic Logic Unit – ALU*), un multiplieur accumulateur (*Multiplier-Accumulator – MAC*) ou un générateur d'adresses sont des exemples de modules numériques.

Selon le type de modules retenus et leur agencement, un large éventail de topologies est possible pour une unité fonctionnelle. Les contraintes influant sur le comportement sont à prendre en compte. C'est par exemple le temps maximal autorisé pour calculer une transformée de Fourier, le nombre d'échantillons par seconde d'un convertisseur analogique vers numérique ou les fréquences de coupure et de transition d'un filtre. Considérer aussi les spécifications non-fonctionnelles est un moyen de restreindre la gamme des possibilités et d'aboutir à un choix plus pertinent. La séparation totale des spécifications fonctionnelles et des spécifications non-fonctionnelles n'est donc pas possible et, même, n'est pas souhaitable.

Les échanges entre modules restent au même niveau d'abstraction que pour les unités fonctionnelles. Il peut être nécessaire de commencer à formaliser certaines interfaces quand la façon d'échanger des données a une influence sur le comportement. C'est notamment le cas quand un protocole de communication est à mettre au point ou à évaluer.

À des fins de concision, l'illustration par l'exemple du dictaphone ne sera poursuivie que pour le raffinement de les unités fonctionnelles de capture du son et de génération de l'horloge.

L'unité de capture de son nécessite un microphone et un convertisseur analogique vers numérique. Afin d'éliminer le phénomène de repliement lors de l'échantillonnage pour la numérisation, un filtre antirepliement doit être intercalé entre le microphone et le convertisseur comme cela est montré sur la figure 12.

Les spécifications servent à déterminer les caractéristiques des modules. Comme le dictaphone est un système sur pile, de taille réduite et ne nécessitant pas de haute fidélité mais qui doit être bon marché, un microphone à électret standard s'impose.

Concernant le filtre antirepliement, il s'agit par essence d'un filtre analogique passe bas. La fréquence la plus élevée à conserver est 4 kHz. Cependant, le spectre de la voix humaine étant assez pauvre à cette fréquence, il est acceptable de fixer la fréquence de coupure à 4 kHz (donc de consentir une atténuation de 3 dB à cette fréquence). Pour filtrer efficacement les fréquences à 4 kHz, il est exigé une atténuation de 20 dB à l'octave, c'est-à-dire à 8 kHz. Ceci définit le gabarit du filtre. À partir de ce gabarit, un filtre passe

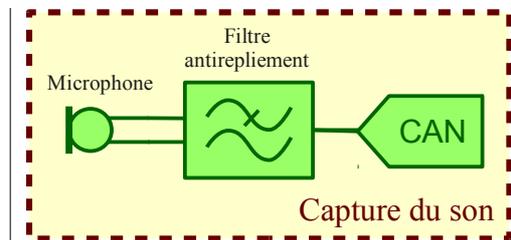


Fig. 12 : Modules constituant l'unité de capture du son

bas prototype est construit par normalisation par rapport à la fréquence de coupure suivie d'une transformation ad hoc. Le comportement du filtre en bande passante et en bande atténuée conditionne la forme de la réponse du filtre et donc le choix de l'approximation qui pourra être faite pour le filtre prototype. S'agissant de capturer la voix, il est attendu de pouvoir reconnaître la personne enregistrée à l'écoute de l'enregistrement. La réponse en bande passante du filtre antirepliement doit donc être plate afin de ne pas déformer la voix enregistrée. Le filtre prototype pourra ici être approximé par un filtre de Butterworth, de Tchebychev II (Tchebychev inverse) ou de Legendre. L'ordre de ce filtre est conditionné par le gabarit à respecter. En général, l'ordre le plus bas est préféré car il conduit à un circuit plus compact. Le choix de l'approximation est donc aussi guidé par l'ordre du filtre. La fonction de transfert* du filtre prototype est déduite de l'approximation retenue et de l'ordre choisi à l'aide du gabarit. La fonction de transfert du filtre réel est extraite par transformation inverse et dénormalisation du filtre prototype. La simulation du filtre antirepliement est réalisée à ce niveau d'abstraction grâce à la fonction de transfert obtenue.

Pour le convertisseur analogique-numérique, la fréquence la plus élevée à conserver étant 4 kHz la fréquence d'échantillonnage doit être au moins de 8 kHz pour respecter le théorème de Shannon. Une fréquence un peu supérieure est à préférer de façon à ce que la reconstruction des fréquences proches de la fréquence la plus élevée ne soit pas trop difficile. Si la reconstruction envisagée ne consiste qu'à une simple lecture des échantillons bruts, ce qui revient à une interpolation d'ordre 0, la fréquence d'échantillonnage devra être bien supérieure de façon à avoir plus de deux échantillons par période à 4 kHz. Des essais doivent être conduits dans ce cas pour déterminer à partir de quelle fréquence d'échantillonnage la qualité sonore est acceptable. Ces essais doivent également permettre d'évaluer le nombre requis de pas de quantification en amplitude pour atteindre ce niveau de qualité. Il en découle des spécifications en débit (nombre d'échantillons par seconde) et en résolution (nombre de pas de quantification) pour le convertisseur. Ces spécifications, complétées par des contraintes supplémentaires en précision, linéarité, vitesse de conversion auxquelles s'ajoutent les spécifications non-fonctionnelles comme le coût et la consommation, doivent permettre de définir quel type de convertisseur analogique-numérique est le plus adapté à l'application.

Le principe de fonctionnement du générateur d'horloge, la deuxième unité fonctionnelle détaillée ici, est montré figure 13. La consigne de fréquence reçue de l'unité de gestion du cadencement est écrite sur un convertisseur numérique vers analogique (CNA). La tension de sortie du CNA pilote un oscillateur contrôlé en tension (*voltage controlled oscillator – VCO*). La plage de fréquences requises pour le cadencement ainsi que le nombre de fréquences différentes dans cette plage sont à déterminer en corrélation avec l'algorithme de compression. Le nombre de fréquences a un impact direct sur la résolution du CNA. L'algorithme de compression spécifie la vitesse à laquelle un changement de fréquence doit être effectué. Cette vitesse de changement conditionne la vitesse de conversion et contraint le temps de stabilisation de l'oscillateur. Un élément clé de cette structure est de définir la plage de tension avec laquelle le convertisseur numérique vers analogique et l'oscillateur contrôlé en tension doivent travailler. Elle est en relation avec la tension d'alimentation du système car elle ne peut être supérieure à celle-ci sauf à ajouter un élévateur de tension ce qui, de façon injustifiée, complexifierait le système et augmenterait son coût.

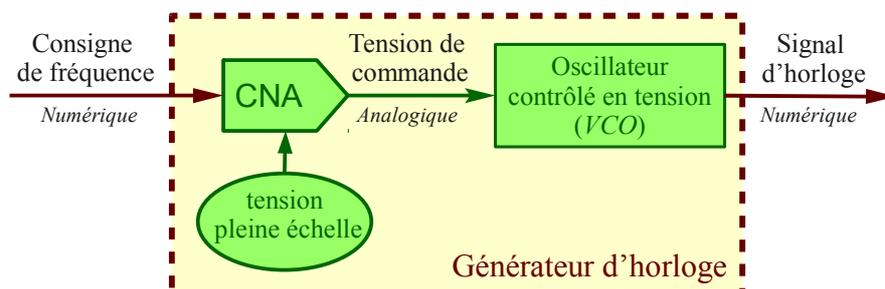


Fig. 13 : Modules constituant le générateur d'horloge

L'exemple du générateur d'horloge illustre le cas d'une partie d'un système qui à haut niveau d'abstraction, ici l'unité fonctionnelle, est vu comme un élément numérique alors que dans sa structure interne il utilise un ou plusieurs éléments analogiques. Ces derniers apparaissent au fur et à mesure du raffinement, ici dès le niveau des modules. Cette exemple montre aussi que dans le raffinement des parties analogiques les tensions d'alimentation doivent dans certains cas être prise en compte à des niveaux d'abstraction élevée, éventuellement, comme ici, avant même le début de l'implémentation. Ce dernier point est inhabituel par

* Voir glossaire

rapport à la conception des parties numériques pour lesquelles les alimentations n'apparaissent que très tardivement dans le flot de conception, au plus tôt au niveau des portes logiques, le plus souvent lors des choix technologiques liés à l'étude de la structure interne des portes (cf. fig. 9).

III.3) Implémentation

Le rôle de l'implémentation est d'approfondir la description des éléments qui ont été définis durant l'exploration d'architecture. Une topologie précise et détaillée doit être établie pour chaque module. Un protocole et une largeur de bus doivent être définis pour les canaux numériques de communication. Les signaux analogiques d'échange entre modules analogiques doivent être enrichis de façon à faire ressortir leur nature physique.

Plusieurs implémentations sont possibles pour une architecture donnée. Toutes les spécifications doivent être prises en compte pour effectuer le choix le plus pertinent. Toutefois, des considérations technologiques peuvent éventuellement apparaître aux niveaux les plus détaillés mais aucunement au début de l'implémentation. L'implémentation est achevée quand toutes les parties du système, quelles soient analogiques ou numériques, sont représentables sous forme d'un schéma de transistors, de diodes et de composants passifs (*transistor netlist*).

Les niveaux d'abstraction de l'implémentation numérique sont le transfert entre registres (*RTL*), le niveau « porte logique » et la structuration interne des portes logiques. Ces niveaux sont détaillés ci-après. Afin d'établir un raffinement analogique comparable et concordant avec le raffinement numérique, les étapes de l'implémentation numériques doivent être revues pour être adaptées au raffinement des parties analogiques. Il s'agit ici de répondre aux questions de quel est le pendant en analogique du transfert entre registres et de quel est le type d'éléments analogiques équivalent à une porte logique.

III.3.1) Transfert entre registres et structure interne des modules analogiques

La structure interne des modules numériques est décrite au niveau transfert entre registres. Les parties combinatoires et les éléments mémorisants sont identifiés. Tous les signaux de contrôles sont exprimés en particulier les horloges, les signaux d'initialisation (*reset*) ou d'activation (*enable*, *High-Z*, ...). En prenant l'exemple d'une unité arithmétique et logique, la liste des fonctions logiques ou des opérations arithmétiques qu'elle doit être capable de traiter dépend du processeur auquel elle est destinée. Pour un additionneur, plusieurs topologies sont possibles selon la façon de propager ou d'anticiper la retenue [49].

Les types et les largeurs de bus sont à expliciter également à ce niveau. Les derniers détails des protocoles d'échange doivent être précisés. Faut-il un port parallèle, un port sériel ou un réseau sur puce ? Dans le cas d'un port sériel, un choix est à faire entre RS232, RS485, USB 1.0, USB 2.0, etc.

L'équivalent analogique consiste à représenter la structure interne des modules analogiques. Ces derniers sont décrits comme un assemblage d'amplificateurs opérationnels, de comparateurs, de gyrateurs, d'interrupteurs analogiques, de diodes et de composants passifs. Plusieurs topologies sont souvent possibles. Par exemple un filtre d'ordre supérieur à deux peut être décomposé de différentes façons en cellules d'ordre un ou d'ordre deux. L'association de ces cellules est différente selon les besoins et elle ne dépend pas uniquement des performances mais aussi des questions d'adaptation d'impédance. Si les puissances mises en jeu sont importantes ou si les fréquences sont très élevées, le filtre sera passif et réalisé avec des condensateurs et des bobines disposés en échelles. L'association des cellules d'ordre un ou d'ordre deux est en série. La méthode de synthèse doit tenir compte de l'interdépendance des impédances des composants à l'échelle du filtre en entier. Quand les fréquences et les puissances sont plus réduites, l'emploi d'amplificateurs opérationnels est envisageable. Le filtre est alors plus facile à réaliser par assemblage de cellules actives d'ordre un ou deux rendues indépendantes grâce au découplage que l'amplificateur opérationnel rend possible. Les cellules actives peuvent être associées en parallèle ou en série (voir fig. 14). Une topologie doit être choisie pour chacune d'elles. Une cellule active du second ordre peut être réalisée avec une structure de Rauch, de Sallen-Key, biquadratique, etc. La figure 15 donne l'exemple de deux structures internes possibles pour un même filtre passe bas d'ordre trois.

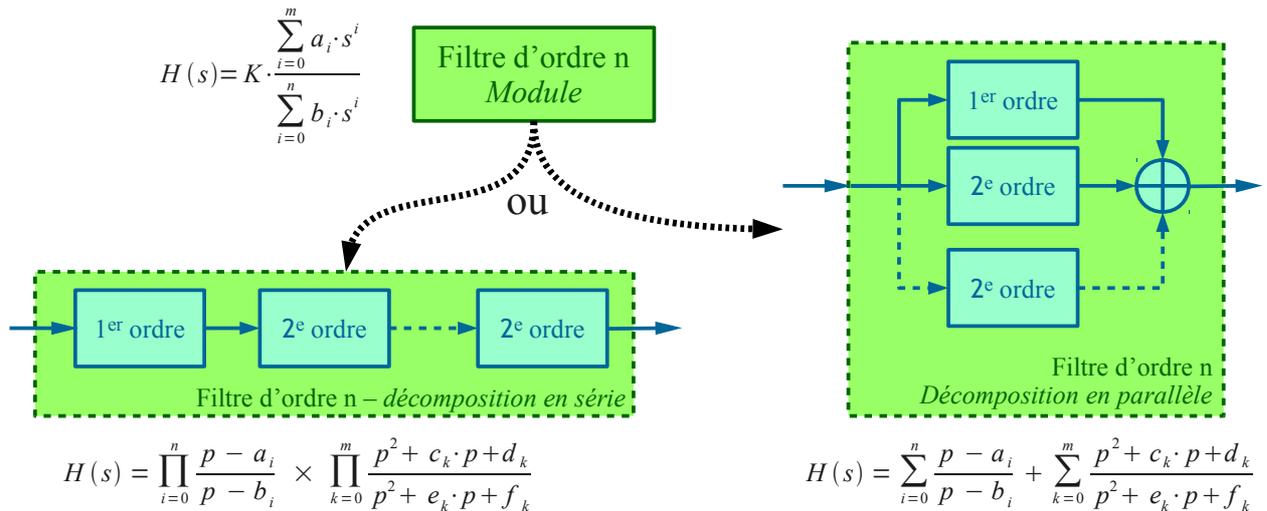


Fig. 14 : Principe de la décomposition des filtres analogiques

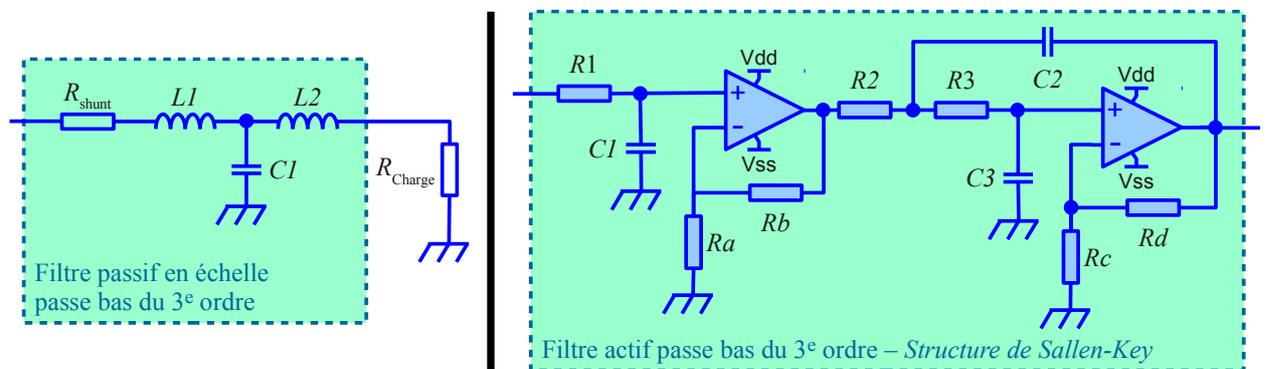


Fig. 15 : Exemple de structures pour un filtre passe bas du 3e ordre

Pour le générateur d'horloge du dictaphone de l'exemple, l'oscillateur contrôlé en tension est le seul module détaillé ici, toujours pour des raisons de concision. Le principe de fonctionnement le plus répandu est de construire un oscillateur à base de condensateurs dont un est remplacé par une diode varicap. Ce composant se comporte comme un condensateur variable dont la capacité est réglée par une tension de contrôle. Puisqu'il s'agit ici de produire un signal carré pour cadencer le processeur, une autre solution consiste à exploiter la propriété des portes logiques CMOS dont le temps de propagation dépend de leur tension d'alimentation. L'oscillateur contrôlé en tension est obtenu en réalisant un oscillateur en anneau à base d'inverseurs dont la tension d'alimentation est ajustée. La tension de commande du module est la tension d'alimentation des inverseurs de l'oscillateur en anneau, comme l'explique la figure 16. La période d'oscillation est deux fois le temps de propagation à travers un inverseur (une fois pour le passage de 0 à 1 et une fois pour le passage de 1 à 0) multiplié par le nombre de portes dans l'anneau (cf. [50]) ; la formule de la fréquence est rappelée sur la figure. Cette exemple illustre l'exploitation d'un phénomène analogique dans un circuit numérique. Le dernier inverseur en sortie de l'oscillateur n'intervient pas dans l'établissement de la fréquence d'oscillation. Son rôle est d'adapter la sortie au circuit attaqué par l'horloge.

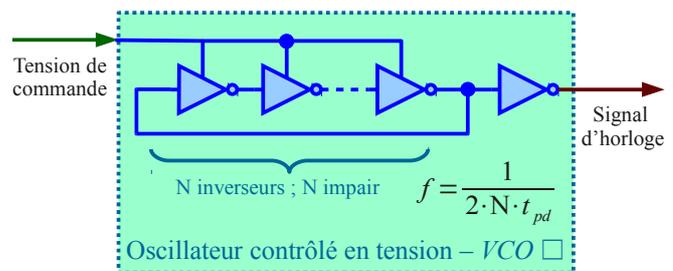


Fig. 16 : VCO à base d'un oscillateur en anneau dont la tension d'alimentation est ajustée

Lors de l'établissement de la structure interne des modules, la représentation des signaux analogiques est à temps continu et à amplitude continue. Tous les signaux électriques doivent être caractérisés y compris les alimentations et les polarisations. A ce niveau, le modèle doit restituer les principales limitations et les

non-linéarités dominantes c'est-à-dire celles qui influencent le plus le comportement du module. Il s'agit, par exemple, de la saturation de la tension de sortie d'un amplificateur opérationnel ou des effets d'une vitesse de balayage finie de l'amplificateur. Les paramètres déterminant les performances tels que le débit, la précision ou le temps de montée sont précisés. Tous les composants requis pour l'amélioration des performances, comme un condensateur de filtrage en relation avec un taux de réjection de l'alimentation, ou de la stabilité sont répertoriés et insérés dans la structure du module.

III.3.2) Porte logique et bloc analogique

Une opération de synthèse transforme la description des modules numériques du niveau « transfert entre registres » en une description du niveau « portes logiques ». Suite à cette opération, les modules numériques sont devenus un assemblage de portes logiques (*gate netlist*). Cette opération s'effectue en deux temps. Dans un premier temps, la synthèse est purement comportementale. Ensuite, les informations sur les alimentations ainsi que sur les temps de propagation, d'établissement et de maintien sont introduites. Ces dernières informations sont fortement liées à la technologie employée mais aussi à la topologie des portes utilisées. Ceci est parfaitement illustré par l'exemple de l'oscillateur commandé en tension avec la structure de la figure 16 qui repose sur un oscillateur en anneau à base d'inverseurs. En effet, la corrélation entre la tension d'alimentation d'une porte et son temps de propagation dépend de la technologie utilisée dans la porte. Ce point met au jour le fait que dans un processus de raffinement descendant, il n'est pas possible de faire un choix avisé à un niveau donné sans un minimum de connaissances des possibilités existant aux niveaux d'abstraction inférieurs. Il démontre aussi comment les conséquences d'un choix technologique peuvent être remontées à des niveaux relativement abstraits (ici le niveau structure du module via l'utilisation des portes logiques pour réaliser le VCO).

Les blocs analogiques complexes utilisés pour définir la structure interne des modules analogiques jouent, dans la description des parties analogiques, un rôle symétrique à celui des portes logiques. Ce sont les amplificateurs opérationnels, les comparateurs, les gyrateurs ou encore les interrupteurs analogiques. Leur description se fait également en deux temps. Dans un premier temps, la description est purement comportementale et ne renferme aucune hypothèse topologique ni technologique. Cependant, elle inclut les non-linéarités et les limitations dominantes déjà évoquées lors de la constitution de la structure interne des modules analogiques. Le chapitre V montrera un modèle d'amplificateur opérationnel qui répond à ces critères de description purement comportementale intégrant les non-linéarités dominantes sans hypothèse topologique ni technologique. Dans un deuxième temps, le modèle est enrichi par l'ajout de toutes les non-linéarités et limitations. Toutes les grandeurs qui servent à les caractériser sont donc précisées. Parmi ces paramètres se trouvent, entre autre, le rapport signal sur bruit (*Signal over Noise Ratio – SNR*), le taux harmonique de distorsion (THD), le taux de réjection du mode commun (*Common-Mode Rejection Ratio – CMRR*), le taux de réjection de l'alimentation (*Power-Supply Rejection Ratio – PSRR*), les erreurs de linéarités ou de gain, etc. A ce stade, les modèles extraits à partir d'un circuit réel ou d'un schéma transistors prennent toute leur pertinence. Le niveau « bloc analogique » est un très bon point de rencontre entre la conception descendante et la conception ascendante des circuits.

III.3.3) Structure interne des portes logiques et des blocs analogiques

Un schéma transistors est associé à chaque porte logique. Plusieurs topologies sont possibles pour chacune d'elles. Par exemple, une porte logique comme un ET ou un OU peut avoir plus de deux entrées, avoir ou non un hystérésis (trigger de Schmitt). Ensuite en fonction de l'optimisation recherchée, en vitesse, en consommation ou en surface minimale, la structure interne sera différente. Elle est également à adapter en fonction de la sortance, c'est-à-dire en fonction du courant maximal que la porte doit pouvoir débiter sans une altération trop importante de ses caractéristiques.

De la même façon, la structure interne des blocs analogiques est construite par association de groupements élémentaires de transistors. Chaque groupement constitue un composant analogique de base tel qu'un miroir de courant, une paire différentielle, une source de courant, un adaptateur de niveau (*level shifter*), etc. Une liste détaillée est donnée dans [51] et [52]. Plusieurs topologies sont possibles pour chaque composant analogique de base. Par exemple, un miroir de courant peut être simple ou à plusieurs branches, cascodé ou non. La topologie est choisie en fonction des performances à atteindre et des spécifications reçues du raffinement des niveaux supérieurs.

La figure 17 donne la structure interne classique de l'amplificateur opérationnel à transconductance (OTA – *Operational Transconductance Amplifier*) simple à deux étages (sortie non-différentielle). Les composants analogiques de base sont mis en évidence.

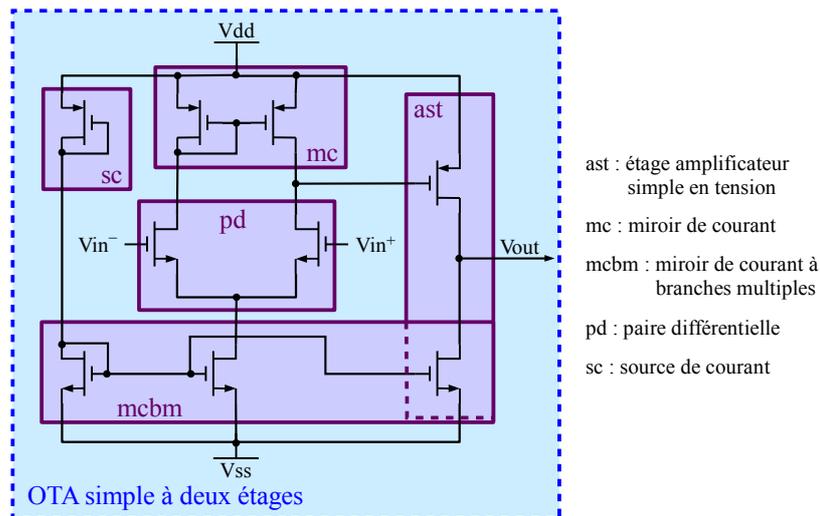


Fig. 17 : Exemple de structure interne d'un amplificateur opérationnel

L'analyse de l'implémentation d'un système mixte menée dans ce sous-chapitre a permis d'identifier que, tandis que les niveaux d'abstraction de l'implémentation numérique sont le transfert entre registres (*RTL*), le niveau « porte logique » et la structure interne des portes logiques, les niveaux d'abstraction de l'implémentation analogique sont la structure interne des modules analogiques, le niveau « bloc analogique » et la structure interne des blocs analogiques.

III.4) Dimensionnement et tracé

Le dimensionnement consiste à calculer les dimensions géométriques des composants : longueur et largeur pour les transistors, surface pour les condensateurs.

Au cours du tracé, un jeu cohérent de patrons de masque est dessiné en respectant les règles de dessin fournies par le fondeur. Ces patrons sont ensuite utilisés par le fondeur pour confectionner les masques indispensables à la fabrication du circuit réel.

Ces deux étapes sont totalement liées à la technologie employée. Il est donc impératif d'avoir au préalable sélectionné une technologie en fonction du compromis à atteindre entre vitesse, consommation, surface et coûts de fabrication.

Chapitre IV)

Usage de SystemC-AMS dans le processus de raffinement analogique

Pour répondre à la densité et à la complexité toujours croissantes des circuits intégrés, de nombreux outils de conception assistée par ordinateur (CAO), ont été conçus et perfectionnés au cours des années. Cependant chacun d'eux est spécialisé dans un domaine (simulateur numérique, simulateur analogique, simulateur radiofréquence, ...). Or, pouvoir mener à bien des simulations du système complet est désormais indispensable pour parvenir à la conception d'un système mixte et analogique, voire hétérogène, dans lequel se côtoient circuits de traitements numériques, capteurs, actionneurs analogiques ou systèmes microélectromécaniques (MEMs) et système d'émission-réception radiofréquence. Ceci implique de cosimuler le système, c'est-à-dire faire exécuter des simulations simultanées et interconnectées sur les différents outils de simulations. Des techniques de cosimulation ont donc été mises au point comme celles proposées ou évoquées dans [53]. Mais ces outils n'ont pas été initialement prévus à cet effet. Les interconnecter et les faire échanger des informations peut s'avérer compliqué et difficile à mettre en œuvre. Comme une grande quantité de ressource se trouve mobilisée pour la synchronisation des différents simulateurs, les performances de simulations sont parfois décevantes. Outre les aspects purement techniques, des considérations de licences entrent en jeu quand les outils ne sont pas libres de droit. Une licence par outil est alors requise ce qui peut rendre la cosimulation rapidement très onéreuse.

SystemC-AMS, extension de SystemC, simplifie la simulation globale en apportant un environnement de simulation unifié. Grâce aux différents niveaux d'abstraction et aux différents modèles de calculs intégrés de SystemC-AMS, la modélisation d'un système dans son ensemble est plus efficace. Elle est également plus facile à réaliser du fait de l'utilisation d'un seul et même langage de programmation pour toutes les composantes du système, qu'elles soient numériques, analogiques, radiofréquences ou microélectromécaniques. Avec le tandem SystemC-AMS et SystemC, le concepteur n'a plus qu'un seul outil de conception et de simulation. Il n'a plus à rechercher comment interconnecter les différents domaines puisque tout le nécessaire est fourni avec le langage. Il doit désormais naviguer entre les différents niveaux d'abstraction et les différents modèles de calcul. Ceci n'est pas forcément évident et requière une bonne connaissance de SystemC-AMS. De plus, un modèle de calcul n'est pas lié à un niveau d'abstraction : un même modèle de calcul peut être utilisé à des niveaux d'abstraction différents et un même niveau d'abstraction peut être modélisé de plusieurs façons. En conséquences, les performances de simulations peuvent être améliorées en précision et en vitesse par un choix avisé du modèle de calcul. De premiers essais de performances, réalisés avec les différents modèles de calcul de SystemC-AMS, ont montré qu'il ne sera pas raisonnable de décrire un système complet en utilisant uniquement le modèle de calcul le plus précis, le modèle ELN (*Electrical Linear Network*), cf. [41]. Le raffinement partiel par unité fonctionnelle est donc incontournable pour parvenir à mener à bien la conception d'un système complet. La méthode de raffinement structuré du chapitre III démontre ici son intérêt. L'emploi de modèles de plus en plus détaillés et cohérents entre eux, rend possible le raffinement poussé d'une partie tout en maintenant le reste de la description du système à un niveau plus abstrait, moins gourmand en calcul. Une fois cette partie étudiée finement, il est alors possible de reprendre son modèle de plus haut niveau afin de passer au raffinement d'une autre partie. Le choix du style de modélisation le mieux adapté à chaque niveau d'abstraction est crucial pour atteindre le meilleur compromis entre vitesse de simulation et précision.

Le présent chapitre expose une étude des différents modèles de calcul de SystemC-AMS dans le but d'aider le concepteur à faire le meilleur choix à chaque niveau d'abstraction. Il ne s'agit pas de faire ici une comparaison de SystemC-AMS avec d'autres simulateurs comme dans [28 ; 54 ; 55] mais d'évaluer les performances des différents modèles de calcul en terme de vitesse de simulation. Les styles de modélisations possibles avec SystemC-AMS et les différents modèles de calcul adaptés à chaque étape de raffinement ont été examinés de façon à comprendre leur impact sur la simulation. Les styles de modélisations les plus adaptés à un niveau d'abstraction du processus de raffinement analogique présenté au chapitre III ont été estimés afin de déterminer le meilleurs compromis possible entre vitesse et précision de simulation. Les différents modèles de calcul de SystemC-AMS ont été catégorisés dans cet objectif. Les niveaux d'abstraction, pour lesquels SystemC et SystemC-AMS peuvent être employés, sont précisés. Un nouveau style de modélisation combinant deux des trois modèles de calcul disponibles dans SystemC-AMS est introduit. Les temps de simulation des différents styles de modélisation de SystemC-AMS sont évalués à travers deux cas d'étude. L'un est destiné à évaluer l'évolution des performances lors du passage à l'échelle d'un système unitairement simple, l'autre à faire l'évaluation dans le cas du raffinement d'un système plus complexe.

IV.1) Styles de modélisation et niveaux d'abstraction

Le tableau 5 présente les styles de modélisation les plus pertinents pour chaque niveau d'abstraction. Il fait clairement apparaître les niveaux pour lesquels SystemC avec son extension SystemC-AMS sont les plus adaptés, les niveaux pour lesquels ils entrent en concurrence avec d'autre langages de modélisation et enfin les niveaux pour lesquels ils ne sont pas adaptés.

Raffinement numérique		Raffinement analogique	
Niveaux d'abstraction	Styles de modélisation		Niveaux d'abstraction
Système	SystemC TLM	SystemC-AMS TDF	Système
Unité fonctionnelle numérique		SystemC-AMS TDF, LSF, ELN	Unité fonctionnelle analogique
Module numérique	SystemC TLM, BCA		Module analogique
Transfert entre registres (RTL)	SystemC BCA, VHDL, Verilog	SystemC-AMS TDF, LSF, ELN VHDL-AMS, Verilog-AMS	Structure interne du module analogique
Porte logique		SystemC-AMS ELN, VHDL-AMS, Verilog-AMS	Bloc analogique
Structure interne de la porte logique	Simulateur électrique (schéma transistors – <i>Netlist</i>)	Simulateur électrique (schéma transistors – <i>Netlist</i>)	Structure interne du bloc analogique
Transistor	Simulateur électrique		Transistor

Tableau 5 : Styles de modélisations et niveaux d'abstraction

SystemC et son extension SystemC-AMS autorisent de fortes abstractions tout en étant très flexibles. Ils sont particulièrement bien adaptés pour l'exploration d'architecture. Leur capacité à modéliser des systèmes complexes et hétérogènes a déjà été démontré comme par exemple dans [28 ; 39 ; 40].

Au **niveau système**, le comportement peut être décrit à l'aide d'algorithmes et de fonctions de transfert. Les modèles de calcul SystemC TLM (*Transaction Level Modelling*) et SystemC-AMS TDF (*Timed Data Flow*) sont particulièrement bien adaptés à ce niveau pour cette raison.

Pour les **unités fonctionnelles numériques**, SystemC TLM est le plus approprié. Pour les **unités fonctionnelles analogiques**, SystemC-AMS TDF est le plus pertinent dans la majorité des cas. Toutefois, le modèle de calcul SystemC-AMS LSF (*Linear Signal Flow*) s'avère plus adapté quand le comportement de l'unité fonctionnelle peut être modélisé par des opérations mathématiques. Il a été tout particulièrement conçu pour la description des boucles de rétroaction. Un système hétérogène comporte des unités fonctionnelles qui ne sont pas purement électriques, il peut y avoir par exemple des unités électromécaniques ou

optoélectroniques. Quand ces unités non-électriques peuvent être modélisées par une représentation d'états, les modèles de calculs TDF et LSF sont toujours les plus adaptés, chacun de ces modèles de calcul possédant des méthodes de calcul dédiées à la représentation d'état. La modélisation des circuits non-électriques peut être aussi faite à l'aide d'un circuit électrique équivalent créé par un expert du domaine. La modélisation par circuit électrique équivalent permet aussi de traiter des caractéristiques non-fonctionnelles à forte contrainte sur un système comme le couplage thermique. Le circuit équivalent est le plus souvent un réseau électrique linéaire utilisant des composants passifs et des sources de courant et de tension. Le modèle de calcul SystemC-AMS ELN (*Linear Electrical Network*) est alors le plus apte à restituer fidèlement le circuit électrique équivalent. Par conséquent, SystemC-AMS ELN peut être ponctuellement employé à des niveaux d'abstraction élevée comme l'unité fonctionnelle.

Pour les **modules numériques**, SystemC TLM reste pertinent tant que les interfaces et les protocoles entre modules peuvent restés abstraits. Dans le cas contraire, tout particulièrement quand un protocole doit être évalué ou élaboré, le modèle de calcul SystemC BCA (*Bit-Cycle Accurate*) est plus approprié. Les trois modèles de calculs de SystemC-AMS conviennent pour la description des **modules analogiques** pour les mêmes raisons que pour les unités fonctionnelles.

Seul le modèle de calcul SystemC BCA a la précision nécessaire pour satisfaire à la description au **niveau transfert de registre**. À ce niveau, SystemC entre en concurrence avec les langages de description matériel comme VHDL et Verilog. SystemC BCA est très proche de VHDL dans sa structure et sa syntaxe. Il ne parvient pas à ce jour à s'imposer face à VHDL et Verilog par manque d'un outil de génération automatique de SystemC BCA à partir de SystemC TLM. Cette transformation devant être effectuée manuellement, les concepteurs numériques préfèrent passer directement de la description en SystemC TLM à une description en VHDL ou du Verilog qui leur permet de poursuivre le raffinement aux niveaux d'abstraction inférieurs.

Lors de l'élaboration de la **structure interne des modules analogiques**, la représentation des signaux analogiques est à temps continu et à amplitude continue. Les modèles de calculs SystemC-AMS LSF et ELN sont donc ceux à utiliser a priori. Toutefois, le modèle de calcul TDF possède des méthodes pour le calcul des transformées de Laplace et des représentations d'états. Il peut continuer à être employé pour modéliser des fonctions de transfert à condition que le pas de simulation soit suffisamment fin pour que les signaux TDF, bien qu'échantillonnés, puissent être assimilés à des signaux continus. Les modèles de calculs TDF et LSF ne sont employés que pendant la recherche de la topologie la plus adaptée pour le module et que ses sous-modules sont encore abstraits par une fonction de transfert (cf. l'exemple du filtre au § III.3.2). Pour l'expression des sous-modules et finalement du module entier sous forme d'un assemblage de blocs analogiques, la description en SystemC-AMS ne peut être faites qu'avec le modèle de calcul ELN parce que c'est le seul modèle de calcul de SystemC-AMS à traduire la notion de composant et à être conforme aux lois de Kirchhoff. A ce niveau, SystemC-AMS entre en concurrence avec VHDL-AMS et Verilog-AMS, les extensions mixtes et analogiques de VHDL et Verilog. Mais les langages de description matérielle se révèlent assez lents à ce niveau d'abstraction à cause du nombre important d'équations et de paramètres qu'ils doivent calculer. SystemC-AMS est donc plus intéressant pour sa vitesse de simulation même s'il est moins précis que les langages de description matérielle.

Pour la description des **portes logiques** et des **blocs analogiques**, les langages de description matérielle sont les plus performants. Ils sont bien plus rapides que les simulateurs électriques, ce qui a fait leur succès depuis de nombreuses années. Ils sont aussi plus précis que SystemC et SystemC-AMS, dans leur version actuelle, car ils sont capables de prendre en compte les effets analogiques ayant le plus d'influence sur les circuits numériques ou encore de restituer les non-linéarités dans la modélisation des circuits analogiques. En VHDL, la bibliothèque VITAL permet la création de modèles normalisés de circuits logiques dans lesquels les aspects temporels tels que les temps de positionnement (*set up time*), de maintien (*hold time*) ou de propagation sont pris en compte. L'extension VHDL-AMS introduit la limitation des temps de montée et de descente (transitions en rampe). Les instructions simultanées conditionnelles et sélectives de VHDL-AMS étendent les structures de choix aux signaux analogiques ce qui autorise nativement les changements d'équations en cours de simulation facilitant ainsi la restitution des comportements non-linéaires. Une utilisation astucieuse de SystemC et de SystemC-AMS pourrait rendre possible l'introduction des effets analogiques dans la description des circuits numériques faites en SystemC. Grâce à la force d'abstraction de SystemC et de SystemC-AMS, ces effets analogiques pourrait être incorporés dans des structures complexes relevant de la description au niveau transfert entre registres et non pas seulement au niveau des portes logiques comme c'est le cas avec VHDL. Il y aurait donc un intérêt à utiliser SystemC lors de la construction de la structure interne des modules numériques afin d'évaluer l'impact de ces effets analogiques sur le

comportement de ces modules numériques. Symétriquement, SystemC-AMS peut atteindre le niveau bloc analogique s'il est étendu avec des modèles de composants complexes. Pour une bonne adéquation, ces modèles doivent enrichir la description d'informations quant aux limitations et non-linéarités les plus importantes (non négligeables) introduites par de tels composants, il ne doit pas s'agir d'un simple changement de vue. L'intérêt d'utiliser SystemC-AMS au niveau bloc analogique apparaît dans le cas où la structure interne des modules analogiques est décrite à l'aide de SystemC-AMS. L'impact sur le système des limitations et non-linéarités des composants complexes, comme les amplificateurs ou les comparateurs, peut être ainsi évalué par remontée de ces informations au cours de l'élaboration de la structure interne des modules. Cette façon de faire réduit le risque de discordance entre la réalisation matérielle et le comportement voulu d'après les spécifications. Ce n'est qu'après avoir mesuré les conséquences des limitations et non-linéarités principales que le choix de la topologie et de la technologie devrait être entrepris. Ces choix sont alors guidés par les résultats de ces mesures. Ceci induit que les modèles de blocs analogiques écrits à l'aide de SystemC-AMS ne doivent pas faire d'hypothèses topologiques ni technologiques. Le chapitre V expliquera la construction sans hypothèse topologique ni technologique d'un modèle de bloc analogique à travers l'exemple d'un amplificateur opérationnel. Ce modèle restitue les limitations et non-linéarités dominantes d'un amplificateur opérationnel que sont la saturation, la vitesse de balayage finie et l'atténuation du gain en fréquence.

Les **structures internes des portes logiques et des blocs analogiques** sont exprimées sous forme de schémas transistors qui nécessitent l'emploi d'un simulateur électrique pour être correctement simulés. Les modèles électriques petits signaux équivalents qui sont utilisés pour dimensionner les transistors sont également exécutés sur les simulateurs électriques. SystemC et SystemC-AMS sont totalement inadaptés, ils n'ont d'ailleurs pas été conçus pour un tel niveau de détails.

Au **niveau transistor**, les opérations de placement, routage et délimitation des zones actives lors du tracé des patrons de masque sont réalisées avec les modules dédiés du simulateur électrique. Les vérifications après le tracé comme le contrôle des règles électriques ou la confrontation du tracé avec le schéma transistors, sont également effectués à l'aide de modules dédiés du simulateur électrique.

IV.2) Styles de modélisation de SystemC-AMS et vitesse de simulation

L'exposé se focalise sur la modélisation des parties analogiques. Le paragraphe IV.1 a établi que les trois modèles de calcul de SystemC-AMS pouvaient être employés pour la modélisation des unités fonctionnelles, des modules analogiques et la description de la structure interne des modules analogiques. Afin d'aider le concepteur à faire un choix pertinent, les différents styles de modélisation que permettent les modèles de calcul de SystemC-AMS sont présentés. La répercussion du choix du style de modélisation sur la vitesse de simulation est ensuite évaluée, à travers deux cas d'étude, dans deux directions : le passage à l'échelle et la complexité. Le lecteur aura à garder à l'esprit que les deux cas d'étude proposés ici n'ont été forgés que dans le seul but de cette évaluation de performance. Les décomptes des équations qui sont faits dans la suite et les temps de simulations présentés se rapportent à la preuve de concept de SystemC-AMS (*Proof of concept – PoC*) réalisée par le Fraunhofer Institut et utilisée pour la mise en œuvre de ces cas d'étude.

IV.2.1) Les quatre styles de modélisation de SystemC-AMS

Les trois modèles de calcul de SystemC-AMS permettent quatre styles de modélisation : la modélisation TDF (*Timed Data Flow*), la modélisation LSF (*Linear Signal Flow*), la modélisation LSF encapsulée, la modélisation ELN (*Electrical Linear Network*).

Dans la **modélisation TDF**, les unités fonctionnelles ou les modules sont décrits avec les modèles de calcul (*Model of Computation – MoC*) TDF sous forme d'algorithmes, de fonctions de transfert ou de représentations d'états.

Dans la **modélisation LSF**, les unités fonctionnelles ou les modules sont décrits comme un assemblage de primitives LSF prédéfinies, chacune d'elles réalisant une fonction mathématique, une fonction de transfert ou une représentation d'états. Les règles de composition de SystemC-AMS imposent que les combinaisons de primitives LSF ne sont autorisées qu'à l'intérieur d'un module SystemC⁸ classique qui sert de

8 Il s'agit ici d'un module au sens de SystemC, à ne pas confondre avec les modules numériques et analogiques décrits au chapitre III lors du raffinement.

conteneur. A l'intérieur du conteneur, les données sont transmises d'une primitive à l'autre par des signaux LSF. Un seul solveur numérique résout toutes les équations de l'assemblage.

Le troisième style de modélisation est une combinaison originale des deux styles précédents que nous proposons d'appeler **LSF encapsulé**. Comme dans la modélisation LSF, les unités fonctionnelles ou les modules sont décrits comme un assemblage de primitives LSF mais ici les signaux entre les primitives LSF sont des signaux appartenant au modèle de calcul TDF. Pour parvenir à cette construction, chaque primitive LSF est encapsulée dans un module SystemC et est interfacée en amont avec un convertisseur TDF vers LSF (primitive `sca_lsf::sca_tdf_source`), et en aval avec un convertisseur LSF vers TDF (primitive `sca_lsf::sca_tdf_sink`). Contrairement à la modélisation LSF où un seul solveur doit résoudre un nombre élevé d'équations, la modélisation LSF encapsulée met en œuvre autant d'instances du solveur qu'il y a de primitives LSF. Ainsi, chaque instance n'a qu'un nombre réduit d'équations à calculer. Cette façon de faire s'apparente à l'approche de réduction de complexité « diviser pour régner » (*Divide and Conquer*).

Dans la **modélisation ELN**, les unités fonctionnelles ou les modules sont décrits comme un assemblage de primitives ELN prédéfinies, chacune d'elles représentant un composant électronique idéalisé. Les règles de composition de SystemC-AMS imposent que les combinaisons de primitives ELN ne sont autorisées qu'à l'intérieur d'un module SystemC classique qui sert de conteneur. A l'intérieur du conteneur, ces primitives sont reliées entre-elles par des nœuds qui appartiennent au modèle de calcul ELN. Un seul solveur numérique résout toutes les équations de l'assemblage.

IV.2.2) Répercussion du style de modélisation lors du passage à l'échelle

Le principe de ce cas d'étude est de mettre en œuvre un circuit simple ne requérant qu'une très faible charge de calcul mais qui est instancié un grand nombre de fois.

Le circuit de base est ici un passe bas du premier ordre. Un filtre passe bas d'ordre n est construit en mettant n fois en cascade le filtre passe bas du premier ordre. Le filtre ainsi constitué est excité par une source de tension sinusoïdale modélisée à l'aide de la primitive ELN `sca_eln::sca_vsouce`.

i) Description du filtre dans les quatre styles de modélisation de SystemC-AMS

Dans la **modélisation TDF**, chaque instance du circuit de base calcule la fonction de transfert du filtre du premier ordre en employant la méthode⁹ TDF `sca_tdf::sca_ltf_nd`, voir fig. 18. La forme typique de la fonction de transfert d'un filtre du premier ordre est une fraction rationnelle avec un numérateur égal à une constante et avec un polynôme du premier degré au dénominateur. La résolution dans le domaine temporelle de l'originale d'une telle fonction de transfert requière une équation différentielle¹⁰. Chaque instance du filtre du premier ordre contribue donc pour une équation. Dans le modèle de calcul TDF, chaque instantiation de la méthode `sca_tdf::sca_ltf_nd` conduit à celle d'un solveur. Il y aura donc un solveur par circuit de base, chacun calculant une seule équation dans ce cas d'étude. Les signaux qui relient les n filtres du premier ordre entre-eux appartiennent au modèle de calcul TDF. Puisque la source sinusoïdale est une primitive ELN, un convertisseur ELN vers TDF est nécessaire (primitive `sca_eln::sca_tdf_vsink`). Chacun contribue pour une équation et un solveur supplémentaire est instancié pour les calculer. Pour simuler le filtre d'ordre n et sa source sinusoïdale, le nombre total d'équation est de $(n+2)$ et le nombre total de solveur de $(n+1)$.

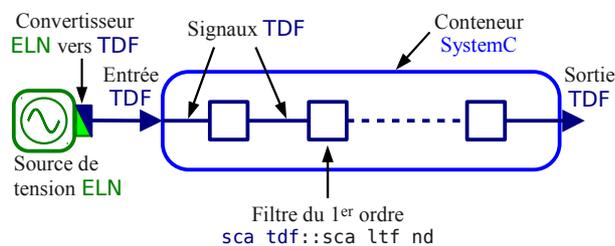
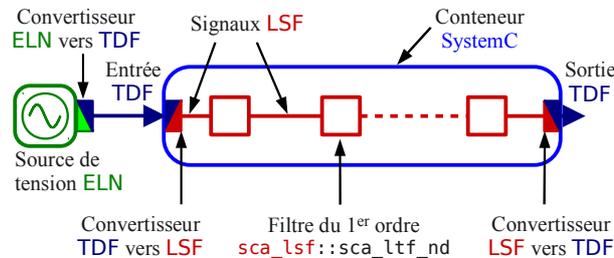


Fig. 18 : Filtre d'ordre n modélisé en TDF

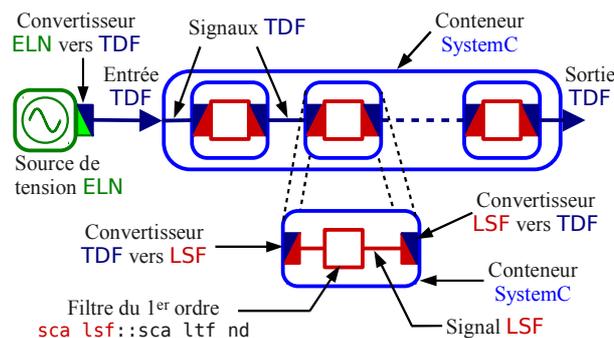
⁹ *méthode* doit être compris ici au sens des langages orientés objet comme C++ dont SystemC-AMS est issu.

¹⁰ Ceci est repris et justifié au V.3 lors de l'étude du modèle de l'amplificateur opérationnel simplifié. Le détail des calculs est donné à l'annexe A 4.1)

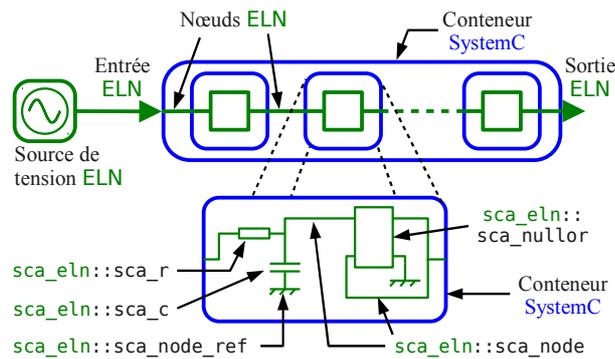
Dans la **modélisation LSF**, chaque instance du circuit de base calcule la fonction de transfert du filtre du premier ordre en employant la méthode LSF `sca_lsf::sca_ltf_nd`, voir fig. 19. Dans la version actuelle de SystemC-AMS, la version 1.0 bêta 2, il n'existe pas de convertisseur ELN vers LSF. La source de tension sinusoïdale est donc reliée au filtre par un signal TDF qui induit l'emploi d'un convertisseur ELN vers TDF et d'un convertisseur TDF vers LSF. Pour des raisons de symétrie et de cohérence, la sortie du filtre d'ordre n est de type TDF. Un convertisseur LSF vers TDF est requis. Le convertisseur TDF vers LSF en entrée du conteneur est relié au premier des filtres du premier ordre par un signal LSF. Puis la sortie de chaque filtre du premier ordre est reliée à l'entrée du suivant par un signal LSF. La sortie du dernier des filtres du premier ordre est relié au convertisseur en sortie du conteneur aussi par un signal LSF. Le modèle de calcul LSF est construit selon l'analyse nodale modifiée qui crée systématiquement une équation pour chaque nœud du réseau électrique modélisé. Dans le style de modélisation LSF, les nœuds électriques sont symbolisés par les signaux LSF. Ainsi chaque signal LSF contribue pour une équation. Une primitive dont le comportement peut être décrit uniquement par une relation directe entre les entrées et les sorties sans calcul intermédiaire n'ajoute pas d'équation supplémentaire. C'est le cas des convertisseurs TDF vers LSF et LSF vers TDF. Par contre chaque primitive `sca_lsf::sca_ltf_nd` ajoute une équation donc chaque instance du filtre du premier ordre contribue pour une équation. Pour simuler le filtre d'ordre n , n équations sont donc générées pour les fonctions de transfert et $(n+1)$ pour les signaux LSF. Ces équations sont calculées par un seul solveur. La source de tension ELN et son convertisseur ELN vers TDF rajoutent chacun une équation. Comme pour la modélisation TDF, ces deux équations sont calculées par un solveur ELN dédié. Au total, le filtre d'ordre n requière $(2n+3)$ équations et 2 solveurs.

Fig. 19 : Filtre d'ordre n modélisé en LSF

Dans la **modélisation LSF encapsulée**, le circuit de base est constitué d'un module SystemC qui encapsule une méthode LSF `sca_lsf::sca_ltf_nd` reliée à l'aide deux signaux LSF à un convertisseur TDF vers LSF en entrée et un convertisseur LSF vers TDF en sortie, voir fig. 20. Comme pour la modélisation LSF, la méthode calcule la fonction de transfert du filtre du premier ordre et contribue pour une équation. De même, chaque signal LSF contribue pour une équation. Chaque instance du circuit de base pèse donc pour trois équations et du fait la construction particulière de celui-ci, elle instancie un solveur qui n'a donc que trois équations à calculer. Les circuits de base sont assemblés à l'aide de signaux TDF dans un module SystemC. Le filtre d'ordre n a donc une entrée et une sortie TDF sans qu'il soit nécessaire de rajouter des convertisseurs. Comme pour la modélisation TDF, l'entrée est excitée par une source de tension sinusoïdale ELN reliée via un convertisseur ELN vers TDF qui contribuent chacun pour une équation, ces équations étant calculées par un solveur ELN dédié. Pour un filtre d'ordre n , le nombre total d'équation est donc de $(3n+2)$ et le nombre total de solveur de $(n+1)$.

Fig. 20 : Filtre d'ordre n modélisé en LSF encapsulé

Dans la **modélisation ELN**, le circuit de base est constitué d'un module SystemC qui encapsule un réseau RC réalisant le filtre du premier ordre, voir fig. 21. La résistance et le condensateur sont simulés respectivement par les primitives ELN `sca_eln::sca_r` et `sca_eln::sca_c`. Afin de faciliter la mise en cascade des circuits de base, une adaptation d'impédance est faite en sortie à l'aide de la primitive ELN `sca_eln::sca_nullor`. Ces différentes primitives sont connectées à l'aide de nœuds ELN `sca_eln::sca_node`. Un nœud de référence qui simule la masse du montage, est défini par la primitive `sca_eln::sca_node_ref`. La source de tension sinusoïdale appartenant au même modèle de calcul que les circuits de base, elle peut être connectée directement à l'entrée du premier d'entre eux, sans convertisseur. La sortie de chaque circuit de base est reliée à l'entrée du suivant par un nœud ELN. Pour des raisons de cohérence, le filtre complet est entièrement décrit avec le modèle de calcul ELN. Sa sortie est donc aussi de type ELN et la sortie du dernier circuit de base est reliée directement à celle-ci sans convertisseur. Dans le modèle de calcul ELN, une résistance dont la valeur est suffisamment grande pour ne pas être assimilée à un court-circuit, est traitée comme une conductance placée en parallèle entre les nœuds auxquels est branchée la résistance. Ajouter une résistance n'ajoute donc aucune équation. Le condensateur est simulé à l'aide d'une équation différentielle du premier ordre. Ajouter un condensateur ajoute donc une équation. L'implémentation du `nullor` instancie une équation. Le modèle de calcul ELN étant lui aussi basé sur l'analyse nodale modifiée, les nœuds ELN contribuent chacun pour une équation. Le nœud de référence, comme les autres nœuds, introduit lui aussi une équation. Comme le montre le détail sur la figure 21, le circuit de base comprend un nœud interne. Chaque instance du circuit de base ajoute donc trois équations. Pour simuler le filtre d'ordre n , $3n$ équations sont générées pour les circuits de base et $(n+1)$ équations pour les nœuds ELN. La source de tension ELN ajoute une équation. Toutes ces équations sont calculées par un seul solveur. Pour un filtre d'ordre n , le nombre total d'équations est donc de $(4n+2)$ et il n'y a qu'un seul solveur.


 Fig. 21 : Filtre d'ordre n modélisé en ELN

Le tableau 6 résume le nombre d'équations et de solveurs pour chaque style de modélisation. La figure 22 illustre la progression du nombre d'équations en fonction de l'ordre du filtre complet.

Style de modélisation	TDF	LSF	LSF encapsulé	ELN
Nombre d'équations	$(n+2)$	$(2n+3)$	$(3n+2)$	$(4n+2)$
Nombre de solveurs	$(n+1)$	2	$(n+1)$	1

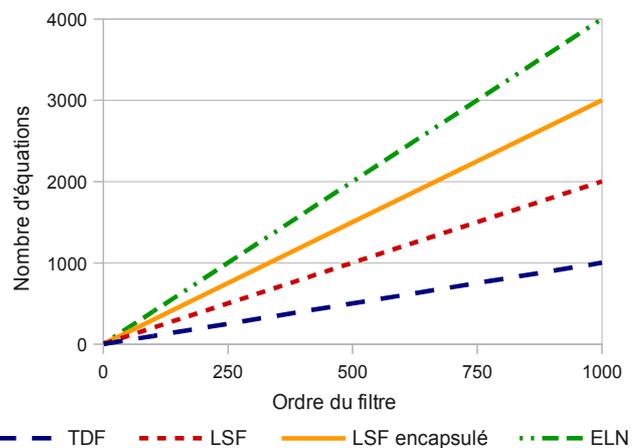
 Tableau 6 : Nombre d'équations et de solveurs pour chaque style de modélisation d'un filtre d'ordre n


Fig. 22 : Progression du nombre d'équations en fonction de l'ordre du filtre pour chaque style de modélisation

ii) Résultats de simulation

Le tableau 7 récapitule les conditions de simulation du filtre d'ordre n . Toutes les instances du filtre du premier ordre ont la même fréquence de coupure. La fréquence de la source de tension sinusoïdale est réglée à cette valeur. Le pas de temps du simulateur est ajusté de façon à produire 1000 points par période de la sinusoïde d'entrée. La résolution temporelle du simulateur est fixée à la valeur du pas de temps afin d'optimiser la vitesse de simulation. La simulation est exécutée pendant 10 périodes du signal d'entrée.

Fréquence de coupure des filtres du premier ordre	100 Hz
Fréquence de la source sinusoïdale	100 Hz
Durée simulée	100 ms
Pas de temps du simulateur	10 μ s
Résolution du simulateur	10 μ s

Tableau 7 : Paramétrage de la simulation du filtre d'ordre n

Afin d'avoir un point de repère, une simulation a aussi été effectuée sous Matlab/Simulink, le modèle étant une transposition de la modélisation TDF. Le solveur sélectionné sous Simulink est ode1 qui utilise la méthode de Euler, celle employée par SystemC-AMS avec un pas de temps fixe. Les déclinaisons bêta 1 et bêta 2 de la version 1.0 de SystemC-AMS ont été utilisées. Pour SystemC, il s'agissait de la version 2.2 tandis que pour Simulink c'était la version 7.10 R2010a.

Toutes les simulations ont été exécutées sur un ordinateur PC sous Linux avec les caractéristiques suivantes : CPU Intel Xeon X5570 (2.93 Ghz, 8 Mo), RAM 24 Go (1066 MHz), HDD 10000 tours/min.

Tous les temps de simulations indiqués dans la suite sont des valeurs moyennées sur dix essais. Pour les modèles SystemC-AMS, ces temps ont été acquis à l'aide la fonction `clock()` de la bibliothèque C pour Linux.

Les temps totaux, c'est-à-dire en incluant la phase d'élaboration, pris par les simulations en fonction de l'ordre du filtre sont présentés fig. 23 pour chaque style de modélisation de SystemC-AMS et pour Simulink. La figure 24 est un zoom de la figure 23 pour les ordres inférieurs ou égaux à 200.

Le premier constat est que dans ce cas d'étude, il n'y a pas de différence majeure entre les déclinaisons bêta 1 et bêta 2 de la version 1.0 de SystemC-AMS en ce qui concerne les temps de simulation.

Quand l'ordre du filtre complet est inférieur à 100, les simulations basées sur les styles de modélisation de SystemC-AMS sont toutes plus rapides que Simulink. Quand le nombre d'instances du filtre du premier ordre dépasse la centaine, le style de modélisation ELN devient plus lent que Simulink. Au delà de deux cents instances, le style de modélisation LSF devient à son tour plus lent que Simulink tandis que les temps de simulation des styles de modélisation TDF et LSF encapsulé sont alors comparables à Simulink. Les simulations utilisant les styles de modélisation LSF et ELN sont très lentes pour un ordre élevé du filtre complet. Ces styles de modélisation n'utilisent qu'un seul solveur pour calculer toutes les équations du filtre (cf. tableau 6 et fig. 22, pour le style LSF le deuxième solveur ne sert qu'à la source de tension sinusoïdale). Par conséquent, le nombre d'équations que le solveur doit résoudre croît avec le nombre d'instances du filtre du premier ordre nécessaire pour atteindre l'ordre du filtre complet. Les temps de simulation supérieurs observés avec le style de modélisation ELN par rapport au style de modélisation LSF s'expliquent par le fait que le modèle de calcul ELN est conservatif, c'est-à-dire qu'il se conforme aux lois de Kirchhoff. Il doit donc résoudre le modèle en tension et en courant, là où le modèle de calcul LSF, non-conservatif, ne traite qu'un signal. Les styles de modélisation TDF et LSF encapsulé utilisent un solveur distinct par instance du filtre du premier ordre. Le nombre d'équations par solveur, outre d'être réduit, reste constant quel que soit le nombre d'instances. La différence de vitesse de simulation observée entre les styles de modélisation TDF et LSF encapsulé provient de la présence des convertisseurs TDF vers LSF et LSF vers TDF dans chaque instance du filtre du premier ordre modélisé avec le style LSF encapsulé. Les signaux requis pour relier ces convertisseurs ajoutent des équations et par conséquent le solveur associé à une instance du filtre du premier ordre a trois équations à calculer avec le style LSF encapsulé là où il n'en a qu'une avec le style TDF. Ces simulations montrent aussi que le style de modélisation TDF n'est pas toujours le plus rapide. Les styles de modélisation LSF et ELN sont en effet plus rapides quand le nombre de primitives instanciées est faible. L'inversion dans l'ordre des temps de simulation se produit pour le style ELN quand le nombre d'instance dépasse la cinquantaine. Pour le style LSF, le point d'inversion se situe entre cent et deux cents instances.

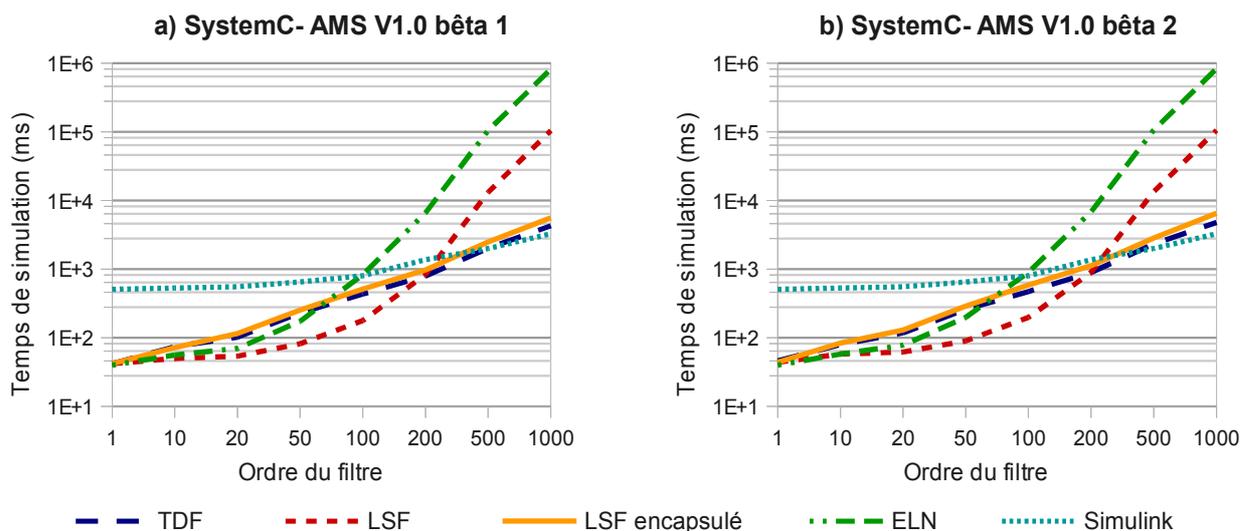


Fig. 23 : Temps de simulation avec élaboration en fonction de l'ordre du filtre

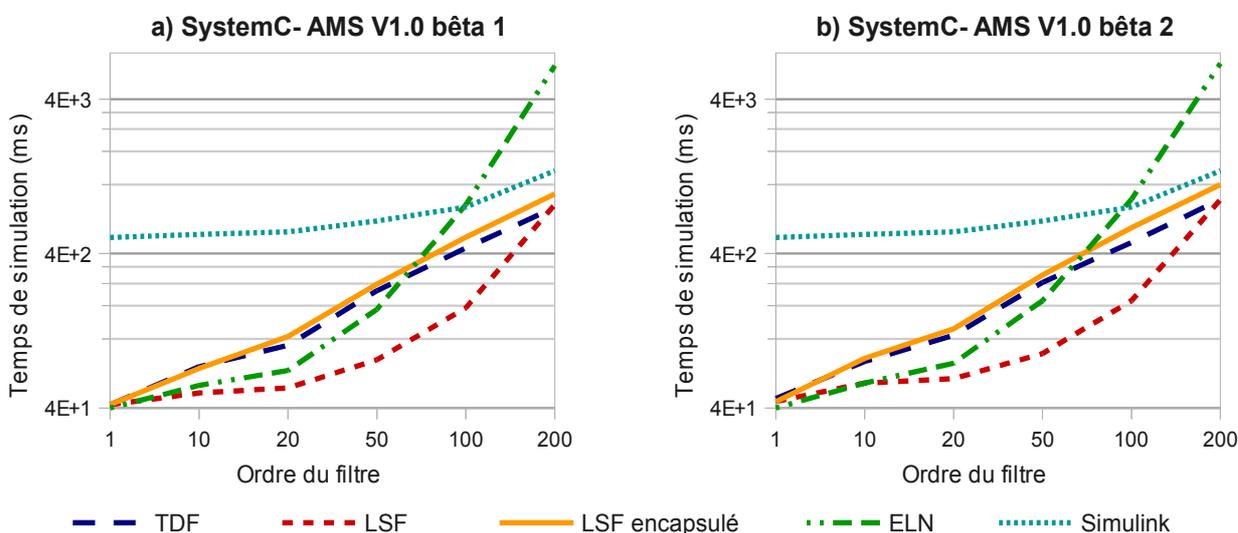


Fig. 24 : Temps de simulation avec élaboration en fonction de l'ordre du filtre (zoom)

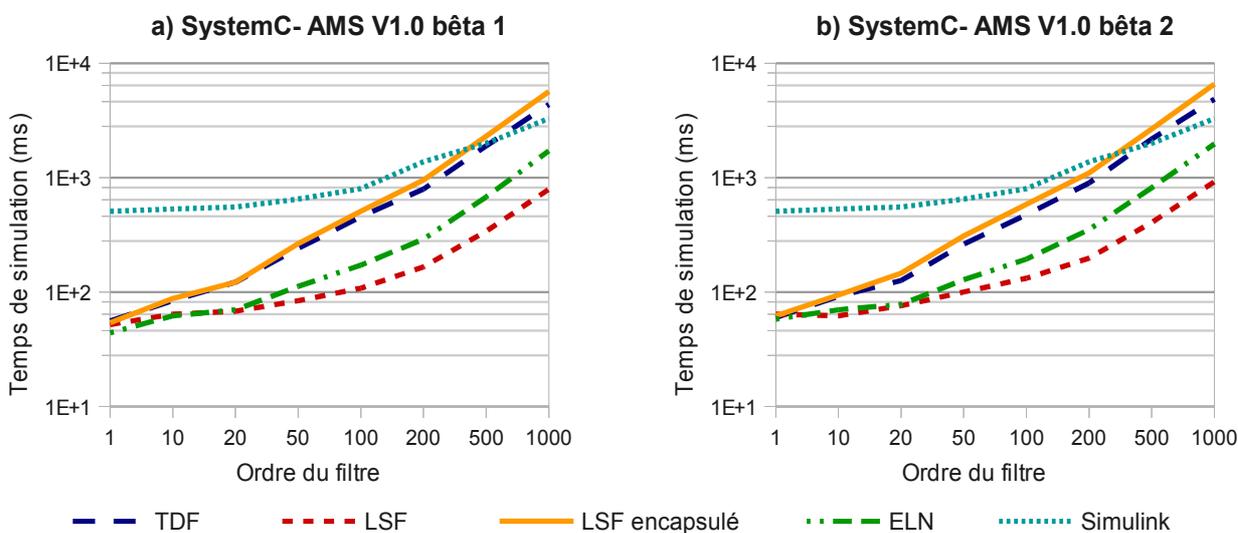


Fig. 25 : Temps de simulation seule en fonction de l'ordre du filtre

Lorsque l'on considère uniquement le temps de simulation sans le temps pris par la phase d'élaboration du simulateur SystemC-AMS, les résultats obtenus diffèrent nettement des précédents. Pour obtenir ces temps, la simulation a été à chaque fois relancée, avec la même durée simulée, en appelant une deuxième fois la fonction `sc_start(...)` à l'intérieur de la fonction principale `sc_main(...)` c'est-à-dire sans quitter l'exécution du simulateur. Ces temps de simulation sans élaboration en fonction de l'ordre du filtre sont reportés sur la figure 25. La courbe obtenue avec Simulink est rappelée.

Les simulations mesurées une fois l'élaboration terminée montrent que les styles de modélisation LSF et ELN se révèlent être les plus rapides dans tous les cas. D'autre part, l'accroissement du temps de simulation des styles LSF et ELN en fonction de l'ordre du filtre suit une progression tout à fait comparable à celui des temps de simulation TDF et LSF encapsulé. Dans ce cas-ci aussi, il n'y a pas de différence notable entre les déclinaisons bêta 1 et bêta 2 de la version 1.0 de SystemC-AMS.

Les modèles de calculs LSF et ELN sont construits selon l'analyse nodale modifiée qui utilise le calcul matriciel pour résoudre les équations servant à la description du système modélisé. Les matrices mise en œuvre comptent une ligne et une colonne par équation. Par conséquent, la taille des matrices est liée au nombre d'équations qui lui-même dépend avant tout du nombre de nœuds dans le système mais aussi du nombre d'éléments qui nécessitent des équations supplémentaires (comme un condensateur ou un source de tension ELN). Or ces matrices s'avèrent souvent très creuses d'où la conduite d'optimisations en vue de calculs plus rapides qu'avec les matrices initiales. Le temps consommé pour mener à bien ces optimisations et au final pour élaborer le simulateur croît rapidement avec la taille des matrices. Les mesures de temps de simulation présentés ci-dessus démontrent que l'essentiel du temps total pris par une simulation LSF ou ELN est consommé par l'élaboration quand le nombre d'éléments devient élevé : dès une cinquantaine de primitives pour le modèle de calcul ELN et à partir d'une centaine pour le modèle de calcul LSF. A l'inverse, quand le nombre de primitives est réduit, typiquement inférieur à la trentaine, le temps d'élaboration est négligeable. Ce dernier point explique pourquoi le style de modélisation LSF encapsulé n'est pas affecté par le temps d'élaboration. Le nombre d'équations par solveur étant au plus de 3 dans le présent cas d'étude, le temps d'élaboration est insignifiant. Le modèle de calcul TDF reposant sur de l'algorithmie ne procède pas à la construction de matrices globales à tout le système. Il ne mets en œuvre le calcul matriciel que pour les primitives qui en ont besoin telles les primitives calculant des fonctions de transfert ou des représentations d'états (primitives `sca_tdf::sca_ltf_nd`, `sca_tdf::sca_ltf_zp`, `sca_tdf::sca_ss`). Sauf à avoir des fonctions de transfert de degré très élevé ou un nombre d'états à représenter très important, ce qui n'est pas le cas du présent cas d'étude, le nombre d'équations par solveur est là aussi très réduit et par conséquent le temps d'élaboration est négligeable.

Dans la version actuelle de SystemC-AMS, il n'est pas possible de conserver le fruit d'une élaboration pour le réemployer ultérieurement. Ceci pénalise lourdement les modèles de calcul LSF et ELN qui pourtant se révèlent les plus rapides en simulation seule. Il y aurait ici matière à améliorer SystemC-AMS.

IV.2.3) Répercussion du style de modélisation sur un système complexe

Le principe de ce cas d'étude est de mettre en œuvre un circuit d'une complexité assez élevée qui n'est instancié qu'une seule fois.

Le système modélisé ici est un four dont la température est contrôlée par un régulateur de température proportionnel-intégral (PI), voir fig. 26. Il est composée de deux unités fonctionnelles : une unité de contrôle de la température et l'unité de cuisson. L'unité de contrôle de la température se décompose en deux

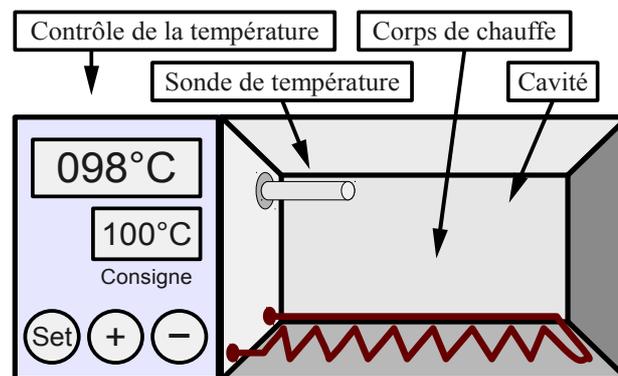


Fig. 26 : Four électrique avec régulateur PI

modules : l'interface homme-machine (IHM) et le régulateur PI. L'unité de cuisson est constituée d'une cavité avec porte renfermant un corps de chauffe et une sonde de température.

La température de la cavité ne peut changer instantanément du fait de l'inertie thermique. Pour refléter cette inertie, la cavité est considérée comme un sous-système dont la fonction de transfert est de type passe bas du premier ordre. La perturbation appliquée au système est une chute momentanée de température comme celle qui peut être observée quand la porte du four est ouverte puis refermée. Ceci est modélisé en modifiant la réponse de la cavité.

i) Description du four dans les quatre styles de modélisation de SystemC-AMS

Dans la **modélisation TDF**, le régulateur PI et la cavité sont modélisés d'une façon globale par une fonction de transfert calculée par la méthode⁹ TDF `sca_tdf::sca_ltf_nd`, voir fig. 27. L'interface homme-machine est traitée algorithmiquement. Dans le cadre de cette étude, elle ne fait que maintenir la consigne fixée au démarrage de la simulation. Il n'y a pas de modification en cours de fonctionnement. La chute de température dans la cavité est modélisée en modifiant en cours de simulation les coefficients de la fonction de transfert de l'unité de cuisson puis en rétablissant leurs valeurs initiales. Chaque module instancie un solveur. Chaque fonction de transfert contribue pour une équation, les dénominateurs des fonctions de transfert étant du premier ordre.

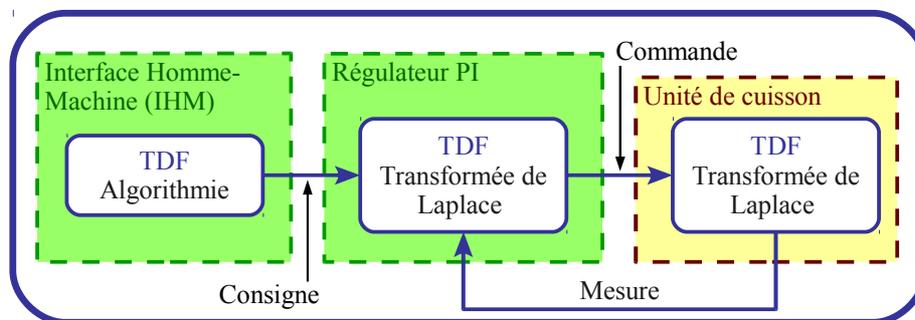


Fig. 27 : Four électrique modélisé en TDF

Dans la **modélisation LSF**, la structure interne du régulateur PI est détaillée. Il s'agit d'une version simplifiée du régulateur proportionnel-intégral-dérivé (PID) donnée dans le guide de l'utilisateur de SystemC-AMS (cf. [4]). La topologie du régulateur émerge sous la forme d'un assemblage de fonctions mathématiques, chacune étant calculée par une primitive LSF, voir fig. 28. Les primitives de la bibliothèque LSF utilisées ici sont : le soustracteur `sca_lsf::sca_sub`, le gain `sca_lsf::sca_gain`, l'intégrateur `sca_lsf::sca_integ` et l'additionneur `sca_lsf::sca_add`. Aucune structure interne n'est précisée pour l'interface homme-machine. Celle-ci est modélisée très simplement par une source LSF `sca_lsf::sca_source` qui délivre le signal de consigne. Ce dernier est maintenu constant tout au long de la simulation. L'unité de cuisson reste au niveau d'abstraction des unités fonctionnelles. Son comportement est modélisée par une équation différentielle du premier ordre dont les coefficients sont modifiés en cours de simulation pour refléter la chute de température. Comme déjà expliqué dans le cas d'étude du filtre, dans le modèle de calcul LSF chaque signal contribue systématiquement pour une équation. Il y a ici six nœuds donc six signaux LSF donc six équations. Les primitives LSF employées ici remplissent leur fonction en employant soit une équation algébrique soit une équation différentielle du premier ordre mettant directement en relation leurs entrées et leurs sorties. Elles n'ajoutent pas d'équations supplémentaires. Ceci est également valable pour l'unité de cuisson qui utilise ici une équation différentielle du premier ordre implémentée directement sans utiliser la primitive `sca_lsf::sca_ltf_nd`. Le nombre total d'équations est donc bien de six et il n'y a qu'un seul solveur.

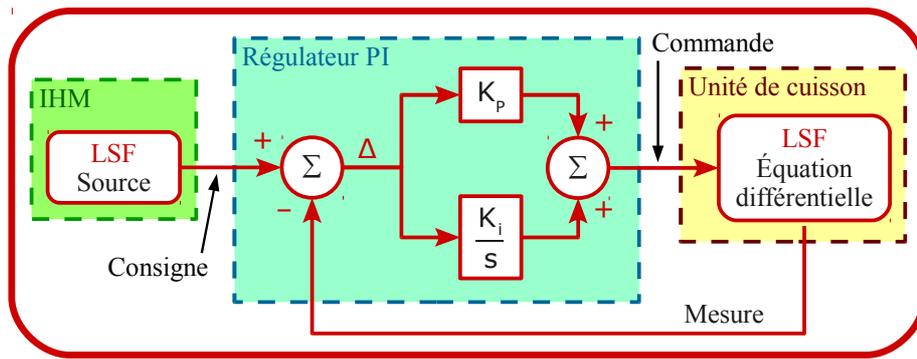


Fig. 28 : Four électrique modélisé en LSF

La **modélisation LSF encapsulée** reprend l'approche de la modélisation LSF avec les mêmes structures et les mêmes niveaux d'abstraction, voir fig. 29. Cependant dans la modélisation LSF encapsulée, chaque primitive LSF est encapsulée dans un module SystemC qui sert de conteneur, chaque entrée de la primitive est reliée via un convertisseur TDF vers LSF et chaque sortie via un convertisseur LSF vers TDF. Les signaux à l'intérieur de chaque conteneur sont des signaux LSF alors que les signaux entre les conteneurs sont remplacés par des signaux TDF. Tout comme le cas du style de modélisation LSF, les primitives LSF employées dans ce cas d'étude n'ajoutent aucune équation. Les convertisseurs TDF vers LSF et LSF vers TDF n'ajoutent pas non plus d'équation. Seuls les signaux LSF conduisent à la création d'équations. Puisqu'il y a six primitives encapsulée, il y a six solveurs LSF. Les équations sont réparties entre ces solveurs LSF de la façon suivante : 1 pour la source, 3 pour le soustracteur, 2 pour le gain, 2 pour l'intégrateur, 3 pour le sommateur et 2 pour l'unité de cuisson, soit treize équations au total. Comme un module TDF supplémentaire est nécessaire pour servir de conteneur à l'ensemble, il a un solveur également de plus soit sept au total (6 LSF + 1 TDF pour synchroniser).

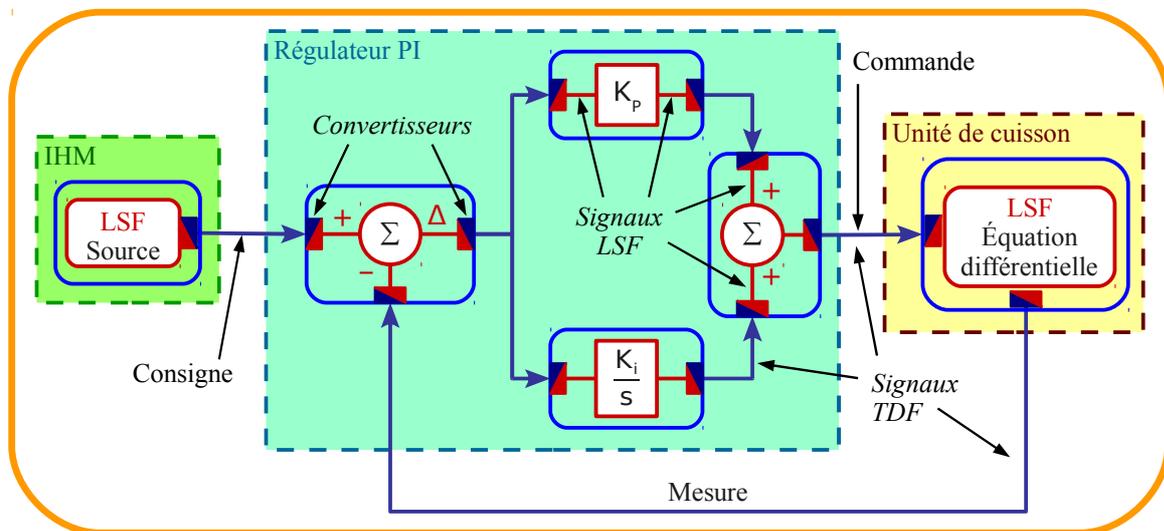


Fig. 29 : Four électrique modélisé en LSF encapsulé

Dans la **modélisation ELN**, le raffinement du régulateur PI est poursuivi. Chaque fonction mathématique apparue dans la modélisation LSF est traduite en un assemblage de blocs analogiques qui sont décrits à l'aide des primitives ELN, voir fig. 30. Les amplificateurs opérationnels utilisés dans le schéma du soustracteur, du gain, de l'intégrateur et de l'additionneur, sont modélisés ici avec la primitive `sca_eln::sca_nullor` câblée en contre-réaction. L'interface homme-machine est modélisée par une source de tension ELN (primitive `sca_eln::sca_vsource`) dont la tension représente la consigne. Cette consigne est fixée au démarrage de la simulation puis est maintenue constante. L'unité de cuisson est gardée au niveau d'abstraction des unités fonctionnelles. Son comportement est modélisée par un filtre du premier ordre de type RC. La perte de chaleur de la cavité est modélisée grâce à une résistance variable placée en

parallèle avec le condensateur. La chute de température consécutive à l'ouverture de la porte du four est simulée en abaissant la valeur de cette résistance au cours de la simulation (augmentation de la perte de chaleur). La fermeture de la porte est simulée en rétablissant la valeur initiale de la résistance variable. Comme déjà expliqué dans le cas d'étude du filtre, dans la modélisation ELN chaque nœud du réseau électrique modélisé introduit une équation. Aucune équation supplémentaire n'est nécessaire pour la modélisation des résistances tandis que la source de tension, les *nullors* et les condensateurs contribuent pour une équation chacun. Il y a dans ce montage, 12 nœuds (dont la masse), 4 *nullors*, 2 condensateurs et 1 source de tension soit 19 équations au total calculées par un seul solveur ELN. Pour des raisons de simplicité, la résistance variable a été réalisée avec la primitive `sca_eln::sca_tdf_r` pilotée par un signal TDF. La gestion de ce signal implique un module TDF et par conséquent l'instanciation d'un solveur TDF. Il y a donc deux solveurs au total.

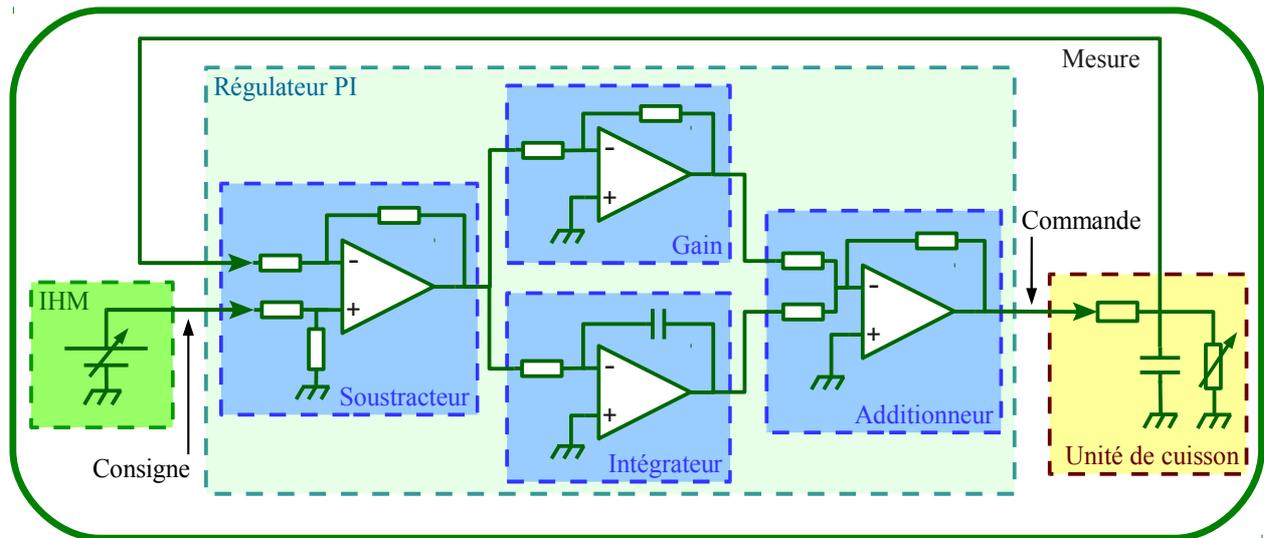


Fig. 30 : Four électrique modélisé en ELN

Le tableau 8 résume le nombre d'équations et de solveurs pour chaque style de modélisation.

Style de modélisation	TDF	LSF	LSF encapsulé	ELN
Nombre d'équations	3	6	13	19
Nombre de solveurs	2	1	7 (6 LSF + 1 TDF)	2 (1 ELN + 1 TDF)

Tableau 8 : Nombre d'équations et de solveurs pour chaque style de modélisation du four

ii) Modélisation du four sous Simulink

De façon à avoir un point de comparaison dans ce cas-ci aussi, trois modèles du four ont été créés sous Matlab/Simulink. Ces modèles ont été nommés OvenPIDTDF, OvenLTFTDF et OvenLSF. Les deux derniers ont été spécialement adaptés pour correspondre au maximum aux modélisations TDF et LSF de SystemC-AMS. Le style de modélisation LSF encapsulé n'a pas de sens sous Simulink. Le style de modélisation ELN ne peut pas être transposés sous Simulink puisqu'il n'existe pas de modèles électriques des composants électroniques comme les amplificateurs opérationnels ou les composants passifs.

Dans le **modèle OvenPIDTDF**, montré fig. 31, le régulateur PI est modélisé en utilisant le bloc PID intégré fourni avec Simulink. Les coefficients sont ceux qui ont été utilisés pour le modèle SystemC-AMS selon le style LSF : K_p (proportionnel) = 1.3, K_i (intégral) = 5, K_d (dérivé) = 0. Le comportement de l'unité de cuisson, dénoté « OvenResponse » sur la figure, est modélisé avec une fonction de transfert d'un passe bas du premier ordre. Les coefficients sont les mêmes que ceux utilisés pour le modèle SystemC-AMS selon

le style TDF. Comme pour le modèle TDF, l'interface homme-machine est résumée à une valeur de consigne constante tout au long de la simulation.

Dans le **modèle OvenLTFTDF**, montré fig. 32, le régulateur PI est modélisé avec une fonction de transfert paramétrée de la même façon que la fonction de transfert utilisée dans le modèle SystemC-AMS selon le style TDF. L'interface homme-machine et l'unité de cuisson sont modélisées de la même façon que dans le modèle OvenPIDTDF.

Le **modèle OvenLSF**, montré fig. 33, reflète le modèle SystemC-AMS selon le style LSF. Le régulateur PI est constitué du même assemblage de fonctions mathématiques paramétrées de la même façon : un soustracteur, un bloc de gain, un intégrateur et un additionneur.

Dans les trois modèles Simulink, l'unité de cuisson est modélisée à l'aide d'une fonction de transfert (cf. le bloc « OvenResponse »). L'ouverture et la fermeture de la porte du four sont simulées en changeant les coefficients de la fonction de transfert à des instants prédéterminés. Pour se faire, le temps de simulation est suivi à l'aide du bloc de type Clock appartenant à la bibliothèque de Simulink. Le bloc Clock fournit l'information sur le temps simulé à un bloc Simulink de type Level2 M-File S-Function. Ce dernier est personnalisable à souhait par l'utilisateur à l'aide d'un fichier externe rédigé dans le langage Matlab (fichier M-file). Ce fichier externe contient des commandes qui sont automatiquement envoyées au modèle Simulink sans devoir suspendre la simulation. Pour les tests réalisés dans cette étude, le programme personnalisé examine la valeur courante du temps simulé, valeur reçue du bloc Clock, et la compare avec deux paramètres enregistrés. Le premier est l'instant où la porte du four est ouverte et le second est l'instant où elle est refermée. Quand le moment de l'ouverture de la porte est atteint, les coefficients de la fonction de transfert qui modélise la réponse du four, sont modifiés puis ils sont rétablis à leur valeur initiale au moment prévu pour la fermeture de la porte.

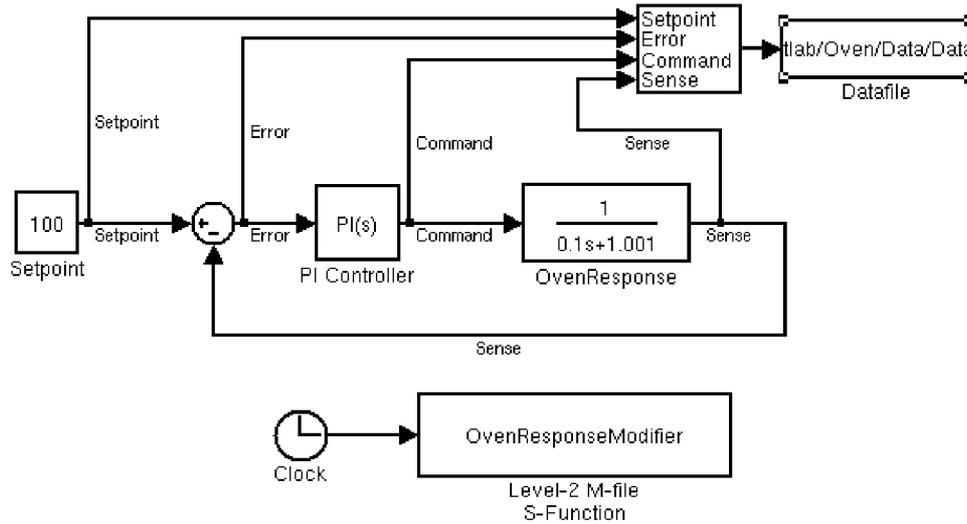


Fig. 31 : Four électrique – modèle OvenPIDTDF sous Simulink

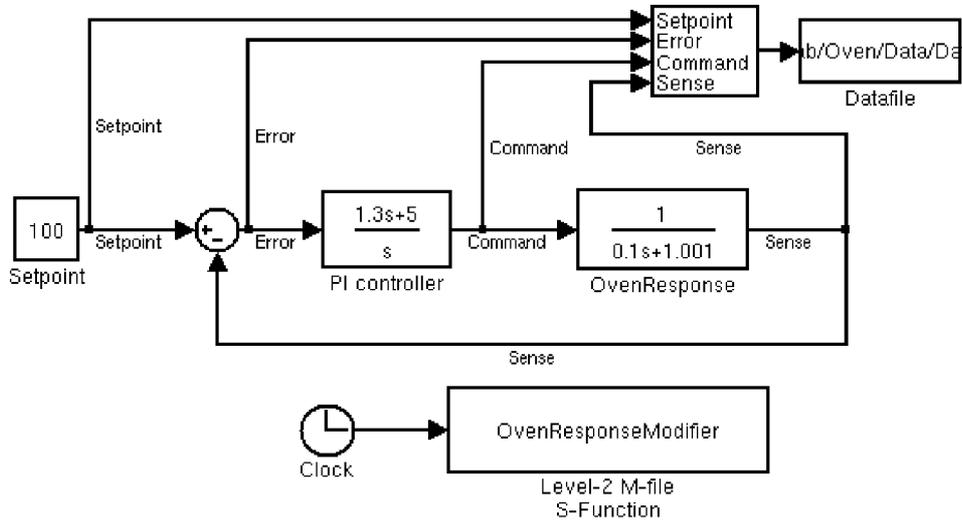


Fig. 32 : Four électrique – modèle OvenLTFTDF sous Simulink

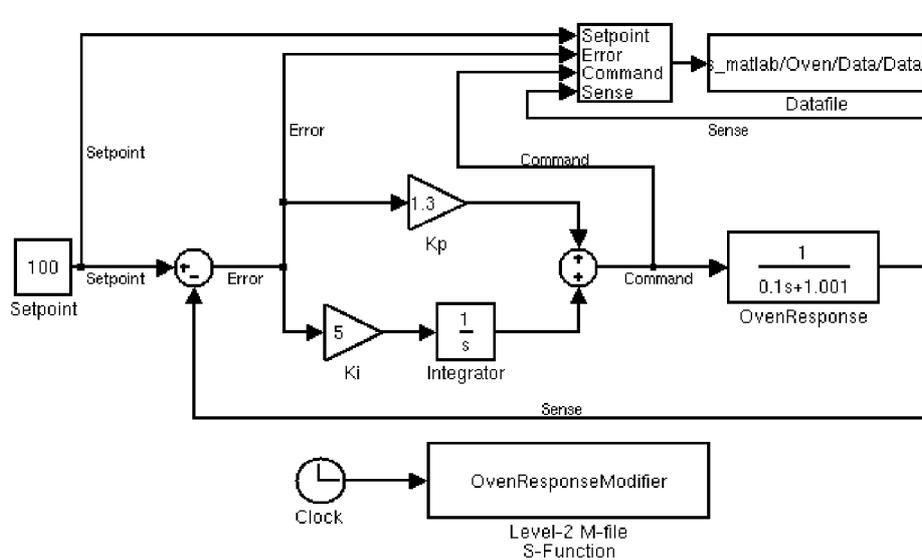


Fig. 33 : Four électrique – modèle OvenLSF sous Simulink

iii) Résultats de simulation

Pour toutes les simulations, les signaux de consigne, de commande et de mesure sont tracés dans un fichier.

Le solveur sélectionné sous Simulink est ode1 qui utilise la méthode de Euler, celle employée par SystemC-AMS avec un pas de temps fixe. Afin d'effectuer une comparaison équitable entre SystemC-AMS et Simulink, les modèles Simulink sont exécutés sans l'interface graphique de Simulink. Ils sont démarrés à l'aide de la commande Matlab `sim(...)` écrite dans un fichier Matlab M-file.

Toutes les simulations SystemC-AMS comme Simulink sont effectuées sur le même ordinateur que celui utilisé pour le cas d'étude du filtre, un PC sous Linux avec les caractéristiques suivantes : CPU Intel Xeon X5570 (2.93 GHz, 8 Mo), RAM 24 Go (1066 MHz), HDD 10000 tours/min. Les déclinaisons bêta 1 et bêta 2 de la version 1.0 de SystemC-AMS ont été utilisées. Pour SystemC, il s'agissait de la version 2.2 tandis que pour Simulink c'était la version 7.10 R2010a.

Les conditions des simulations du four sont indiquées au tableau 9.

Durée simulée	10 s
Pas de temps du simulateur	100 μ s
Résolution du simulateur	100 μ s

Tableau 9 : Paramétrage des simulations du four

Les temps de simulations ont été moyennés sur dix essais. Ils sont mis en perspective fig. 34. Pour SystemC-AMS, ces temps incluent le temps d'élaboration du simulateur bien que celui-ci soit négligeable puisque le nombre d'équations est toujours peu élevé dans ce cas d'étude.

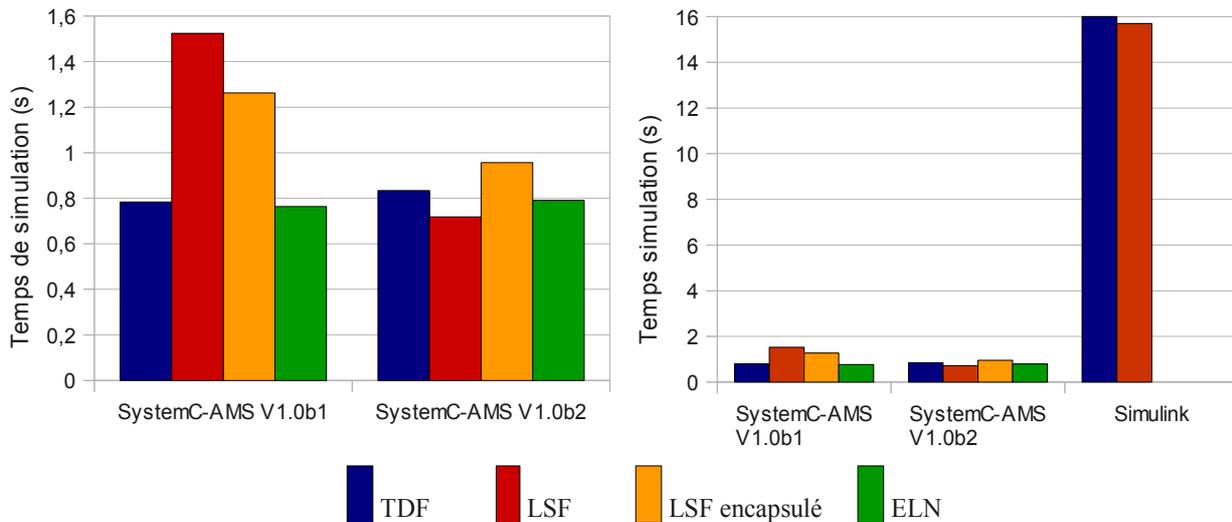


Fig. 34 : Comparaison des temps de simulation du four

Concernant les styles de modélisation SystemC-AMS, le modèle selon le style ELN est un peu plus rapide, bien que très proche (moins de 3 %), du modèle selon le style TDF. Ceci montre que dans le cas d'un réseau électrique dont le nombre d'éléments est réduit, résoudre le réseau électrique ne prend pas plus de temps que de calculer deux fonctions de transfert (une pour le régulateur PI et une pour l'unité de cuisson). Avec la déclinaison bêta 1 de la version 1.0 de SystemC-AMS, le modèle de calcul LSF ne gère pas de façon optimale la boucle de rétroaction. Les simulations du modèle de four selon le style LSF sont les plus lentes. Le style de modélisation LSF encapsulé apporte une réduction du temps de simulation bien que celui-ci reste nettement élevé par rapport aux styles de modélisation TDF et ELN. La gestion des boucles de rétroaction par le modèle de calcul LSF a été grandement améliorée dans la déclinaison bêta 2 de la version 1.0 de SystemC-AMS. En effet, toute chose étant égale par ailleurs, avec la déclinaison bêta 2 les simulations du modèle de four selon le style LSF sont désormais les plus rapides de toutes. Bien que le style de modélisation LSF encapsulé ait aussi bénéficié de cette amélioration, les simulations réalisées avec ce style

de modélisation sont maintenant les plus lentes de toutes. Le style de modélisation LSF encapsulé ne présente désormais plus d'intérêt pour les modélisations comparables à ce cas d'étude, la déclinaison bêta 1 de la version 1.0 de SystemC-AMS étant appelée à disparaître définitivement.

Les deux simulations Simulink renvoyant au style de modélisation TDF, OvenPIDTDF et OvenLTFTDF, durent exactement le même temps. Elles ont été regroupées sous la même barre bleue TDF sur le diagramme de droite de la figure 34. La simulation du modèle Simulink OvenLSF se montre un peu plus rapide que les modèles OvenPIDTDF et OvenLTFTDF bien que ce ne soit pas de façon significative (à peine 2 %). Dans tous les cas, les simulations avec Simulink sont très nettement plus lentes que les simulations avec SystemC-AMS. Le modèle selon le style TDF de SystemC-AMS est vingt fois plus rapide que ces équivalents Simulink. Le modèle selon le style LSF de SystemC-AMS avec la déclinaison bêta 1 de la version 1.0 de SystemC-AMS se montrent dix fois plus rapide que son équivalent Simulink alors que c'est la plus lente des simulations SystemC-AMS dans cette déclinaison. Quand la comparaison est faite avec la déclinaison bêta 2 de la version 1.0 de SystemC-AMS, le modèle selon le style LSF de SystemC-AMS est alors vingt-deux fois plus rapide que son équivalent Simulink.

IV.2.4) Ce qui ressort des cas d'étude

Les deux cas d'étude examinés dans ce chapitre ont d'abord démontré que niveaux d'abstraction et styles de modélisation ne sont pas liés. Le filtre a été décrit avec les quatre styles de modélisation alors qu'il relève du niveau module – selon le processus de raffinement proposé au chapitre III. Le four a été lui aussi entièrement décrit dans chacun des quatre styles de modélisation alors qu'il comportait des éléments à des niveaux d'abstraction différents : IHM au niveau module, unité de cuisson au niveau unité fonctionnelle, régulateur PI au niveau module (modélisation TDF) puis au niveau structure interne du module (modélisations LSF, LSF encapsulé et ELN). Par conséquent, un même niveau d'abstraction peut être décrit avec différents styles de modélisation alors qu'un même style de modélisation peut être employé à différents niveaux d'abstraction.

Pour choisir le style de modélisation de façon pertinente, deux aspects doivent être pris en compte : la durée à simuler et la taille du système modélisé. Ce dernier point implique d'estimer le nombre total de modules dans le style TDF, le nombre de signaux dans le style LSF ou le nombre de nœuds dans le style ELN. En effet, le style TDF, bien que spécialement conçue pour le prototypage virtuel de systèmes mixtes et analogiques [2], n'est pas toujours le plus rapide. Quand le système est de taille réduite, c'est-à-dire constitué d'un nombre restreint d'éléments, les styles de modélisation LSF et ELN offrent une vitesse de simulation plus élevée. Quand la taille du système augmente, la durée de simulation envisagée prend de l'importance. En effet, les styles de modélisation LSF et ELN sont pénalisés par le temps d'élaboration du simulateur. Ainsi, si les durées simulées sont courtes, le temps d'élaboration domine et devient dissuasif pour les systèmes de grande taille. Dès les tailles moyennes, c'est-à-dire une cinquantaine de primitives pour le style ELN et entre cent et deux cents pour le style LSF, le temps total consommé par l'exécution de l'élaboration et de la simulation proprement dite dépasse le temps de simulation du style TDF. Cependant, s'il est envisagé de longues durées de simulation les styles de modélisation LSF et ELN fournissent les vitesses de simulation les plus élevées. Toutefois pour exploiter pleinement ce potentiel, la durée de simulation doit dépasser le temps d'élaboration pour une taille de système donnée.

Quand la taille du système devient très importantes, le style TDF n'est pas en mesure d'apporter une vitesse de simulation supérieure à celle de Simulink et, si la durée à simuler n'est pas très longue, les styles de modélisation LSF et ELN seront encore plus lents. Utiliser SystemC-AMS pour la modélisation des systèmes de très grande taille garde toutefois un intérêt face à Simulink par la possibilité offert par SystemC-AMS de raffiner progressivement et méthodiquement le modèle du système jusqu'à des niveaux de détails inatteignables avec Simulink.

La pertinence du style de modélisation LSF encapsulé a été grandement amoindri avec les améliorations apportées au modèle de calcul LSF dans la déclinaison bêta 2 de la version 1.0 de SystemC-AMS. Pour les systèmes de taille réduite avec des boucles de rétroaction, le style de modélisation LSF apporte désormais la plus grande vitesse de simulation alors qu'il était le plus lent précédemment. Le style de modélisation LSF encapsulé n'amène plus d'amélioration dans ce genre de modélisation. Malgré cela et malgré un effort de codage supplémentaire nécessaire pour encapsuler chaque primitive LSF, ce style de modélisation peut rester attrayant dans le cas de systèmes de très grande taille avec des durées simulées relativement brèves : le style de modélisation LSF encapsulé apporte les performances du style TDF avec le formalisme du style LSF.

Chapitre V)

Modèle d'amplificateur opérationnel simplifié

La description des parties analogiques des systèmes mixtes sur puce est aujourd'hui abordée soit de façon très abstraite avec un point de vue système, soit à l'inverse avec une faible abstraction et une vision détaillée de la mise en œuvre physique. Les descriptions avec un niveau élevé d'abstraction sont purement comportementales et reposent souvent sur des fonctions de transfert, des équations et des algorithmes. Elles conviennent bien pour compléter l'étude des parties numériques aux niveaux les plus abstraits. Cependant, elles masquent complètement les phénomènes physiques qui s'observent sur un circuit réel. Les descriptions proches de la mise en œuvre matérielle imitent de façon précise et détaillée le comportement du circuit réel. Pour atteindre un tel degré de précision, elles sous-entendent des choix de topologie et de technologie. Dans le contexte d'un processus de raffinement méthodique tel que celui proposé au chapitre III, le concepteur définit une architecture et vérifie qu'avec celle-ci le système a un comportement qui respecte les spécifications. Puis, il définit la structure interne des modules constitutifs de cette architecture. La définition de la structure interne des modules analogiques est le niveau adapté pour introduire les principales limitations et les non-linéarités dominantes, celles qui influencent le plus le comportement du module. Il est également le niveau adapté pour s'assurer que, malgré celles-ci, le système respecte toujours les spécifications. Ensuite seulement, le concepteur devrait commencer à élaborer une topologie pour chaque bloc analogique. Le bon déroulement d'un tel processus de raffinement exige des transitions progressives entre les niveaux d'abstraction. Pourtant dans les faits, les concepteurs ont de nos jours tendance à sauter cette étape et à passer directement d'une description avec le point de vue système à une description détaillée proche du matériel et par conséquent à présupposer des choix topologiques et technologiques. Ceci accroît le risque de distorsion entre les niveaux d'abstraction les plus élevés et les plus bas. Introduire progressivement les limitations et les non-linéarités en s'assurant au fur et à mesure que les spécifications sont toujours bien respectées est au contraire une méthode efficace de réduction des risques de discordance entre le modèle comportemental et le circuit réel. De plus, un raffinement dont la progressivité est assurée par le soin apporté aux étapes intermédiaires est un moyen d'aider le concepteur à faire ses choix topologiques et technologiques.

Le présent chapitre montrera à travers le cas d'étude d'un amplificateur opérationnel, à sortie simple en tension, ce qu'est la description d'un bloc analogique à un niveau intermédiaire d'abstraction pour lequel les limitations et les non-linéarités dominantes sont prises en compte sans pour autant faire d'hypothèse topologique ni technologique.

Les signaux analogiques considérés dans le cadre de cette étude sont tous supposés être à temps continu et à énergie finie (signaux aperiodiques) ou puissance finie (signaux périodiques).

V.1) Pourquoi écrire à nouveau un modèle d'amplificateur opérationnel ?

Ce paragraphe explique les raisons pour lesquelles le classique modèle de l'amplificateur idéal ainsi que les nombreux modèles haut niveau d'amplificateur opérationnel que l'on peut trouver dans la littérature ne sont pas adaptés à la description de niveau intermédiaire qui est recherchée ici.

V.1.1) Insuffisances du modèle de l'amplificateur idéal

La structure interne des modules analogiques est décrite par l'association de blocs analogiques eux-mêmes d'abord exprimés sous forme de fonctions mathématiques puis ensuite projetés sur un schéma électronique (cf. § III.3.1). Pour que les limitations et les non-linéarités principales puissent être remontées au niveau de la structure interne du module, afin d'évaluer leur impact sur le comportement du système, celles-ci doivent être exprimées dans les modèles de composants utilisés pour la projection des blocs analogiques.

L'amplificateur opérationnel est très employé pour la réalisation de blocs analogiques comme le gain, l'intégrateur, le dérivateur (différenciateur), le soustracteur, l'additionneur ou encore des filtres élémentaires du premier et du deuxième ordre. Le modèle classique de l'amplificateur idéal est traditionnellement employé pour calculer et simuler les blocs analogiques cités. Il est le plus souvent sous-entendu qu'il s'agit d'un amplificateur en tension. Ce modèle convient-il à notre recherche d'une description de niveau intermédiaire permettant la remontée des limitations et non-linéarités principales ?

Le concept d'amplificateur idéal est énoncé en 1954 par B.D.H. Tellegen dans le cadre d'une recherche d'une série complète d'éléments idéaux de circuit qui associés entre-eux, de façon appropriée, rendent possible la description de tous les circuits électriques y compris ceux au comportement non linéaire (cf. [56]). Après avoir exploré les dipôles linéaires ou non, Tellegen recherchait un quadripôle quasi-actif universel indispensable pour rendre compte de plusieurs comportements non-linéaires. Par quasi-actif Tellegen désigne un quadripôle qui, dans une portion de sa caractéristique, est capable d'amplifier le signal de sortie. En appelant respectivement v_1 et i_1 la tension et le courant d'entrée et v_2 et i_2 la tension et le courant de sortie du quadripôle, et en notant P la puissance, la relation de puissance sur le quadripôle est donnée par :

$$P = v_1 \cdot i_1 + v_2 \cdot i_2 \quad (1)$$

Quand la puissance P est nulle, le quadripôle est passif et sans perte, la puissance en sortie ($v_2 \cdot i_2$) étant égale à la puissance d'entrée ($v_1 \cdot i_1$).

Pour que le quadripôle ait le comportement d'un amplificateur, la puissance mise en jeu sur celui-ci doit ne plus être nulle pour certaines valeurs de tension ou de courant d'entrée : l'énergie en sortie ne provient pas uniquement de l'entrée. Pour réduire le nombre de possibilités et ainsi atteindre un nombre minimal d'éléments de circuit, Tellegen retient deux situations qu'il considère comme les plus simples :

- quand la puissance de sortie est nulle avec une puissance d'entrée non nulle,
- quand la puissance d'entrée est nulle avec une puissance de sortie non nulle.

Le premier cas correspond à un puits d'énergie (sans fond !) qui peut être interprété comme un absorbeur parfait de signal. Sortant du cadre de ce chapitre, il n'est pas examiné plus avant. Le deuxième cas conduisit Tellegen à définir quatre amplificateurs idéaux pour lesquels la puissance mise en jeu sur le quadripôle égale la puissance de sortie.

Cas n°1 : $v_1 \neq 0, i_1 = 0$ et v_1 conditionne v_2

Avec $g \in \mathbb{R}, v_2 = g \cdot v_1$ et i_2 est fixé par la charge

Cas n°2 : $v_1 \neq 0, i_1 = 0$ et v_1 conditionne i_2

Avec $h \in \mathbb{R}, i_2 = h \cdot v_1$ et v_2 est fixé par la charge

Cas n°3 : $v_1 = 0, i_1 \neq 0$ et i_1 conditionne v_2

Avec $m \in \mathbb{R}, v_2 = m \cdot i_1$ et i_2 est fixé par la charge

Cas n°4 : $v_1 = 0, i_1 \neq 0$ et i_1 conditionne i_2

Avec $n \in \mathbb{R}, i_2 = n \cdot i_1$ et v_2 est fixé par la charge

Puisque la puissance d'entrée est nulle alors que celle de sortie ne l'est pas, ce qui signifie un gain infini en puissance, les éléments que Tellegen a décrit sont donc des sources plutôt que des amplificateurs. Ils ont été désignés par la suite comme étant respectivement :

- une source de tension commandée en tension (cas n°1 – *voltage controlled voltage source* – *vcvs*),
- une source de courant commandée en tension (cas n°2 – *voltage controlled current source* – *vccs*),

- une source de tension commandée en courant (cas n°3 – *current controlled voltage source* – *ccvs*),
- une source de courant commandée en courant (cas n°4 – *current controlled current source* – *cccs*).

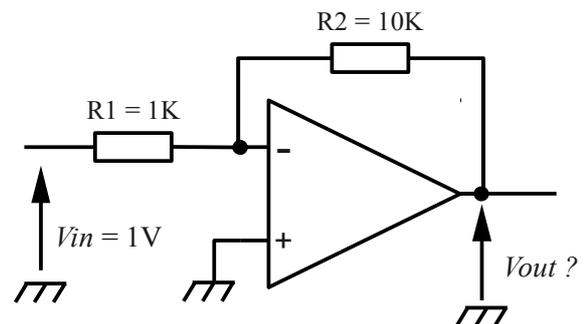
Dans sa recherche d'éléments idéaux de circuit, Tellegen affiche clairement son objectif d'abstraction (cf. [56]) :

« Les éléments de circuit sont ainsi définis par des équations et non pas par des moyens physiques pour les construire. Par exemple, de l'équation $v = L \cdot \frac{di}{dt}$ on ne peut pas conclure que celle-ci décrit une bobine. [...] Les éléments de circuit [...] sont des éléments idéaux qui peuvent être réalisés seulement par approximation. Le problème de leur réalisation est un problème en soi que nous ne discuterons pas. »

Les éléments idéaux de circuit sont donc destinés à transporter une fonction mathématique et non à représenter un composant électronique en tant que tel. Il en découle directement que l'amplificateur idéal ne réalise que la fonction « gain » ($y = kx$, avec k aussi grand que voulu) et, ayant un gain infini en puissance, il ne connaît aucune limitation. Sa sortie peut atteindre n'importe quelle valeur de tension et de courant, et varier instantanément. L'amplificateur idéal ne sature donc jamais et sa vitesse de balayage est infinie.

Le modèle classique de l'amplificateur idéal est communément employé pour donner une première implémentation d'un module analogique. Dans ce contexte, il est le plus souvent compris comme un amplificateur de tension, c'est-à-dire qu'il renvoie au cas n°1 rappelé ci-dessus. Puis la projection des blocs analogiques est accomplie en interprétant chaque amplificateur idéal comme étant un amplificateur opérationnel à sortie en tension. Pour procéder au raffinement progressif recherché ici, le modèle utilisé pour dépeindre l'amplificateur opérationnel doit être capable de refléter ses principales limitations et saturations : au minimum la saturation en amplitude de la tension de sortie et les conséquences d'une vitesse de balayage finie. Or, avec le modèle de l'amplificateur opérationnel idéal, cela est absolument impossible.

Un amplificateur opérationnel est un composant actif puisqu'il peut amplifier l'énergie du signal d'entrée mais ce n'est pas un générateur : l'énergie supplémentaire qui se retrouve dans le signal de sortie n'est pas fournie par l'amplificateur lui-même mais provient d'une source externe à laquelle il est nécessairement connecté. Or, le modèle de l'amplificateur idéal ne spécifie aucune source extérieure. Par conséquent, utiliser le modèle de l'amplificateur idéal en tension conduit soit à commettre des erreurs quand à la prédiction de la tension de sortie soit à faire des hypothèses très contraignantes quant à la réalisation du circuit réel. Examinons le cas classique l'amplificateur en contre-réaction dans le montage inverseur (cf. fig. 35) avec un gain de -10. Quelle est la tension de sortie V_s pour une tension différentielle d'entrée de +1 V ? La réponse qui vient immédiatement à l'esprit est $V_s = -10$ V. Si l'amplificateur idéal est assimilé à une source de tension non-reliée à la masse, cette réponse est valide en toute circonstance. Mais, si l'amplificateur idéal est assimilé à un amplificateur opérationnel, cette réponse réputée évidente impose d'avoir fait deux choix contraignants quant à la mise en œuvre du circuit réel : l'alimentation du circuit doit être symétrique et l'alimentation négative V_{SS} doit être inférieure à -10 V. Or, ces choix ne sont pas forcément acceptables vis à vis du cahier des charges. À notre époque, la majorité des circuits électroniques sont alimentés par des batteries. Il est donc plus facile, plus économe en énergie et moins onéreux de concevoir des alimentations monotension et de tension nominale inférieure à 5 V. À l'inverse, élaborer un système qui devra être alimenté en monotension tout en utilisant le modèle de l'amplificateur idéal pour décrire des structures qui seront réalisées avec des amplificateurs opérationnels, c'est accepter d'utiliser un modèle inadéquat qui peut conduire à des résultats erronés.



$$\frac{V_{out}}{V_{in}} = -\frac{R_2}{R_1} = -10$$

$$V_{out} = \begin{cases} 0 \text{ V} & \text{si } V_{SS} = 0 \text{ V} \\ V_{SS} & \text{si } -10 \text{ V} < V_{SS} < 0 \text{ V} \\ -10 \text{ V} & \text{si } V_{SS} \leq -10 \text{ V} \end{cases}$$

Fig. 35 : Nécessité de spécifier les alimentations d'un amplificateur opérationnel

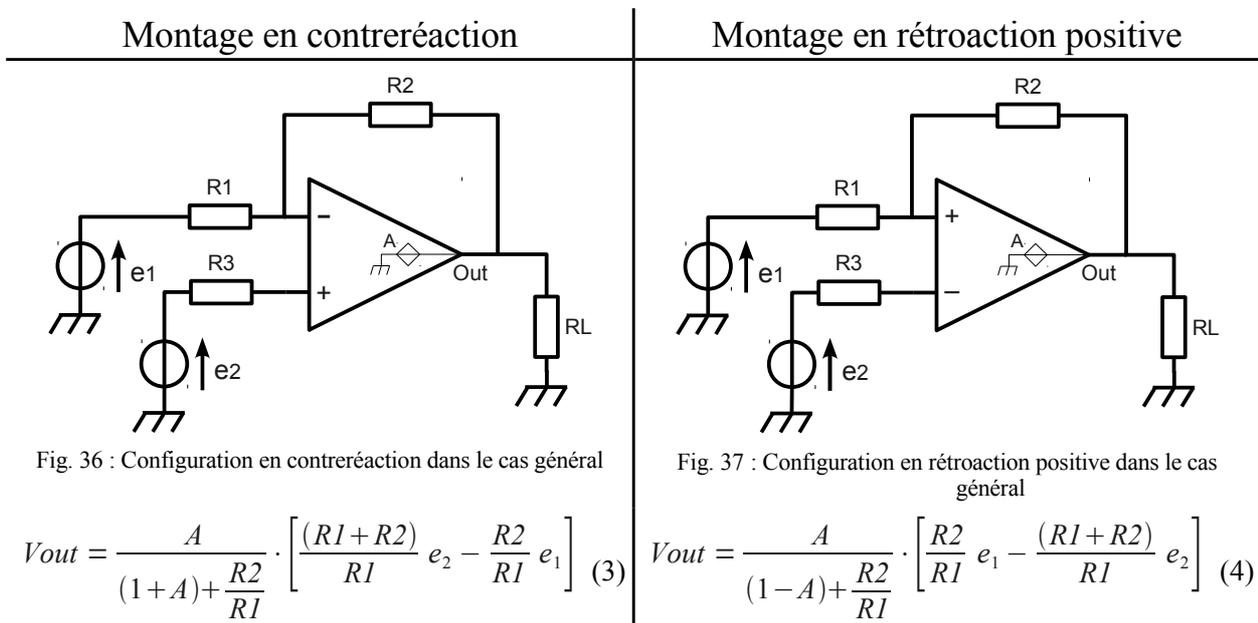
Le modèle classique de l'amplificateur idéal considère que le temps de propagation à travers l'amplificateur est nul. En absence d'inertie, ce modèle est incapable de différencier rétroaction positive et négative, dit autrement de distinguer l'amplificateur monté en réaction de l'amplificateur monté en contre réaction. Ceci se démontre en calculant la tension de sortie en fonction des tensions d'entrées dans les cas du montage en contre réaction (fig. 36) et du montage avec rétroaction positive (fig. 37). La différence entre ces deux schémas est juste l'inversion des entrées de l'amplificateur. Il est considéré que le courant dans chacune des entrées est nul, par conséquent le courant traversant R1 est égal au courant traversant R2 et la chute de tension sur R3 est nulle. Les notations suivantes sont adoptées :

- A : gain statique différentiel en tension de l'amplificateur.
- V^+ : tension sur l'entrée positive de l'amplificateur
- V^- : tension sur l'entrée négative de l'amplificateur
- V_{out} : tension de sortie de l'amplificateur

La tension en sortie de l'amplificateur idéal est régie par l'équation (2)

$$V_{out} = A(V^+ - V^-) \tag{2}$$

Le déroulement du calcul, donné en détail à l'annexe 3, conduit aux résultats suivants.



Le gain statique différentiel A tend vers l'infini, il donc est très grand devant 1 et devant le ratio $R2/R1$.

$$A \gg \frac{R2}{R1} \Rightarrow A + \frac{R2}{R1} \approx A$$

Par conséquent, (3) se simplifie pour devenir :

$$V_{out} \approx \frac{(R1+R2)}{R1} e_2 - \frac{R2}{R1} e_1 \tag{5}$$

$$A \gg \frac{R2}{R1} \Rightarrow -A + \frac{R2}{R1} \approx -A$$

Par conséquent, (4) se simplifie pour devenir :

$$V_{out} \approx \frac{(R1+R2)}{R1} e_2 - \frac{R2}{R1} e_1 \tag{6}$$

Les équations (5) et (6) sont identiques. Le modèle classique de l'amplificateur idéal, dans lequel la tension de sortie est régie par l'équation (2), est incapable de distinguer le montage en rétroaction positive du montage en contre réaction. Dans les deux cas, le modèle se comportera comme si l'amplificateur était en contre réaction. Ce modèle est donc inapte à rendre compte de la dérive rapide de la tension de sortie vers l'un des rails d'alimentation¹¹ que l'on observe avec un amplificateur réel bouclé en rétroaction positive.

Un modèle meilleur que celui de l'amplificateur idéal doit être trouvé.

Pour obtenir une projection correcte des blocs analogiques, il est nécessaire de disposer d'un modèle d'amplificateur en tension sur lequel les alimentations soient exprimées et dont la tension de sortie soit contrainte par les tensions d'alimentations. Destiner à donner une représentation abstraite d'un amplificateur

¹¹ Ce serait vers l'infini en l'absence de saturation de la tension de sortie

opérationnel à sortie en tension, ce modèle devra aussi être capable de reproduire les comportements distincts d'un amplificateur opérationnel quand il est monté en contre-réaction et quand il est monté en réaction, même s'il s'agit d'une reproduction simplifiée. Ce point est essentiel pour détecter les erreurs éventuelles de connexion.

Il y a bien deux niveaux de représentation : d'une part l'amplification c'est-à-dire la fonction « gain » ($y = kx$) et d'autre part le composant électronique amplificateur qui permet une réalisation physique bien qu'imparfaite de la fonction « gain ». Le lecteur doit garder à l'esprit que le propos se place dans le cadre d'un processus de raffinement descendant ce qui signifie que chaque passage vers un niveau d'abstraction plus faible doit s'accompagner d'un apport réel d'informations supplémentaires. Or, le modèle de l'amplificateur idéal avait été imaginé pour restituer la fonction mathématique « gain », il n'apporte pas d'information supplémentaire par rapport à une description dans un formalisme de type flot de signal tel que le modèle de calcul LSF de SystemC-AMS. Par conséquent, quand le niveau d'abstraction réclame de ne voir que la fonction mathématique, il est bien plus performant d'utiliser le formalisme LSF plutôt que l'amplificateur idéal qui ne se détache pas de la représentation sous forme de réseau électrique tout en étant une source potentielle de graves erreurs. Quand le niveau d'abstraction descend au bloc analogique, il est plus pertinent d'utiliser un modèle plus complet que celui de l'amplificateur idéal. Toutefois, bien que plus complet, ce modèle doit rester suffisamment simple pour ne pas impliquer déjà de choix topologiques ni technologiques. En effet, le rôle premier de ce modèle est de mesurer l'impact des limitations et des non-linéarités dominantes sur le comportement du module qui utilise ces blocs analogiques.

V.1.2) Cas du *nullor*

Dans le manuel de référence du langage SystemC-AMS (*Language Reference Manual – LRM*), une note, reproduite ci-dessous, dans le paragraphe consacré à la description de la primitive ELN `sca_nullor` affirme qu'un *nullor* est un modèle d'amplificateur opérationnel idéal.

« NOTE – A *nullor* (a *nullator* - *norator* pair) corresponds to an ideal operational amplifier (an amplifier with an infinite gain). » (cf. [5], § 4.3.1.13, p 83)

Que vaut cette affirmation et le *nullor* peut-il répondre à la présente recherche d'un modèle d'amplificateur en tension capable de rendre compte de la saturation en amplitude de la tension de sortie et des conséquences d'une vitesse de balayage finie sans pour autant faire d'hypothèse topologique ni technologique ?

i) Description du *nullor*

Le concept de *nullor* a été énoncé en 1964 par H.J. Carlin dans [57]. Poursuivant les travaux de Tellegen, Carlin constate qu'un cas n'a pas été traité par celui-ci dans son étude des amplificateurs idéaux (cf. § V.1.1) : le cas où tension et courant sont nuls en même temps à l'entrée de l'amplificateur.

$$((v_I = 0) \text{ ET } (i_I = 0)) \tag{7}$$

Le *nullor* est donc un cas particulier d'amplificateur idéal. Comme la puissance de sortie de l'amplificateur n'est pas nulle, puisque c'est une condition de définition des amplificateurs idéaux, ni la tension ni le courant de sortie ne sont conditionnés par la tension ou le courant d'entrée. Ils peuvent prendre une valeur arbitrairement grande, ceci indépendamment l'un de l'autre. Il en résulte que pour définir la fonction de transfert du *nullor*, une boucle de rétroaction est indispensable. La nature et la valeur des éléments constitutifs de la boucle de rétroaction fixent la valeur de la tension et du courant de sortie du *nullor*, de façon à ce que la condition (7) soit respectée. [58 ; 59]

ii) Pertinence du *nullor* comme modèle d'un amplificateur opérationnel

L'impossibilité de mettre en œuvre le *nullor* sans une boucle de rétroaction, interdit l'emploi du *nullor* pour modéliser un amplificateur opérationnel dans un montage sans rétroaction tel que le montage en comparateur sans hystérésis (voir fig. 38).

N'étant qu'un cas particulier des amplificateurs idéaux, le *nullor* souffre des mêmes insuffisances que ceux-ci pour modéliser un amplificateur opérationnel (cf. § V.1.1).

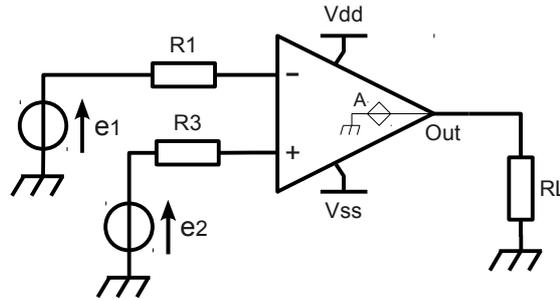


Fig. 38 : Comparateur sans hystérésis

Un montage réalisable avec amplificateur opérationnel mais impossible à modéliser avec un *nullor* car sans boucle de rétroaction

Le *nullor* est définitivement un objet mathématique abstrait. Cependant, son niveau d’abstraction n’est pas nécessairement très élevé. Dans [57], Carlin donne deux exemples de l’utilisation du *nullor*. Le premier exemple est le gyrateur. Le second est la modélisation d’une triode, l’ancêtre des transistors MOSFET actuels ! L’emploi du *nullor* au niveau transistor est toujours d’actualité. En 2003, E. Tlelo-Cuautle et L.A. Sarmiento-Reyes exposent dans [60] une nouvelle technique d’analyse informatique des circuits intégrés analogiques. Cette méthode est la conjugaison de l’analyse nodale pure et de l’analyse symbolique. Grâce à une technique de transformation à base de *nullors* dont ils sont les inventeurs, ils sont en mesure de traiter avec l’analyse nodale des éléments qui ne sont pas directement représentables dans ce type d’analyse, ceci sans devoir recourir à l’analyse nodale modifiée. Ils démontrent la possibilité de traduire un schéma transistor complet en un schéma *nullor* équivalent à travers différents cas d’étude dont un OTA¹² cascodé et replié. La figures 39 reproduit le schéma transistor et le schéma *nullor* de cet OTA.

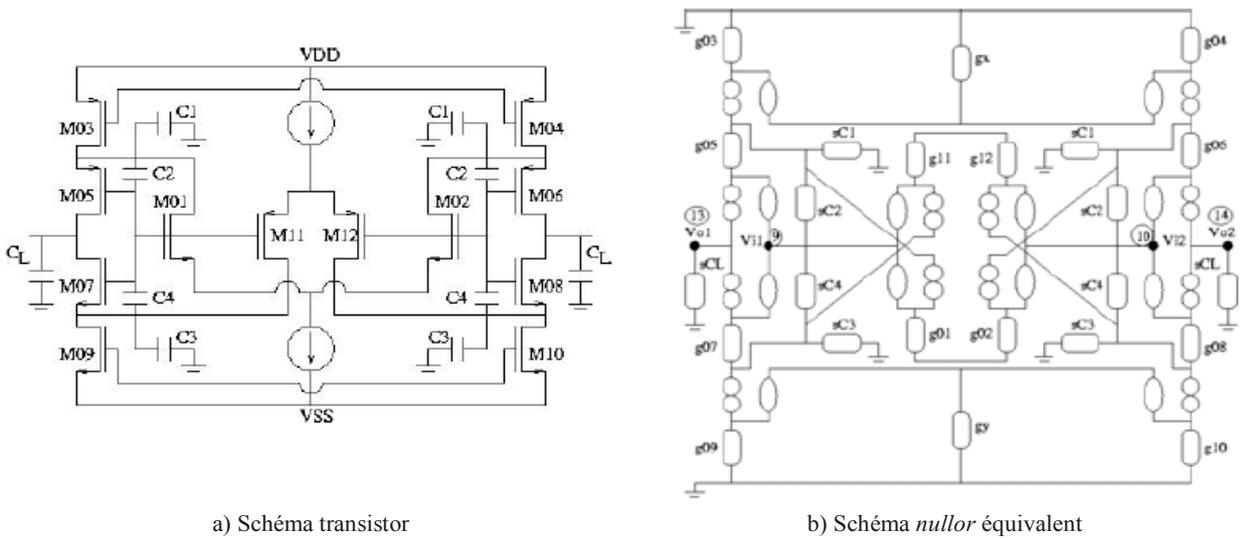


Fig. 39 : OTA cascodé et replié à sortie différentielle
source : [60]

En 2006, D.G Haigh présente dans [61] et [62] une nouvelle génération de *nullors* dont la matrice de calcul contient des coefficients infinis c’est-à-dire des coefficients qui sont des limites d’expressions mathématiques, la limite pouvant être infinie. Le *nullor* classique de Carlin, lui, est calculé à l’aide d’une matrice à coefficients réels finis. Alors que le *nullor* classique de Carlin permet la modélisation de transistors MOSFET idéaux, D.G Haigh parvient, avec le *nullor* à coefficients infinis et une technique appropriée de transformation et de clonage, à modéliser des structures de transistors non-idéaux. Il obtient ainsi des modèles plus précis puisque, par rapport au modèle idéal basé sur des transistors idéaux, le comportement de la structure est altéré par les effets parasites des transistors.

Le *nullor* a été principalement utilisé pour modéliser le suiveur en tension et le suiveur en courant [58], deux montages qui grâce à la facilité d’adaptation d’impédance qu’ils procurent autorisent la conception modulaire des systèmes électriques. C’est en cherchant à matérialiser ces montages que l’amplificateur opérationnel a été employé pour approcher le comportement du *nullor*. En 1977, J.H. Huijsing et J. De Korte

12 OTA : Operational Transconductance Amplifier – Amplificateur opérationnel à transconductance

relèvent dans [58] une insuffisance de cette approche : alors que le *nullor* est un quadripôle avec un port de sortie à deux bornes, l'amplificateur opérationnel n'a qu'une borne de sortie. La conséquence est un nombre important de composants externes pour approcher les montages à base de *nullors*. Toujours dans le même article, J.H. Huijsing et J. De Korte décrivent un circuit intégré plus adapté qui donne la possibilité de mettre en œuvre ces montages avec le nombre minimal de composants requis. Ce composant qu'ils ne nomment alors pas s'avère être un amplificateur opérationnel à sortie différentielle. Il en commencèrent la description en énonçant les déviations par rapport au *nullor* théorique.

Il ressort de cet aperçu que :

- le *nullor* n'est pas une abstraction de l'amplificateur opérationnel, c'est l'amplificateur opérationnel qui est une approximation du *nullor* dans une plage restreinte de tensions et de fréquences. Et, ceci uniquement dans les cas où il est employé dans un montage avec contre réaction.
- le *nullor* n'est pas le modèle d'un composant électronique quel qu'il soit, c'est un objet mathématique qui permet de construire des modèles de composants électroniques. Le niveau d'abstraction auquel il est employé peut être très bas.
- le *nullor* conduit soit à des modèles d'amplificateurs idéaux dans le cas particulier où ils sont montés en contre réaction soit à des modèles au niveau transistor.

Le *nullor* n'est pas adapté pour répondre à la recherche dont ce chapitre est l'objet.

V.1.3) Raisons pour lesquelles les modèles existants d'amplificateurs opérationnels à haut niveau d'abstraction ne conviennent pas non-plus

La littérature contient de nombreux exemples de modèles d'amplificateurs opérationnels à haut niveau d'abstraction. Ces modèles se répartissent en trois catégories :

- ceux extraits du schéma transistor d'un composant réel, comme dans [63] ou [64] ,
- ceux basés sur un réseau électrique équivalent, comme dans [65] ou [66],
- ceux composés par un assemblage de blocs de construction, comme dans [67] ou [68].

Les modèles extraits sont fortement liés à la topologie du composant et à la technologie employée pour le réaliser. L'utilisation de ces modèles impose d'accepter les choix topologiques et technologiques qui ont été fait pour établir le schéma transistor. L'emploi de tels modèles implique donc que le concepteur a déjà franchit l'étape de la structure interne des blocs analogiques. Ce n'est donc pas le niveau d'abstraction qui nous intéresse ici.

Parmi les modèles construits à l'aide d'un réseau électrique équivalent, certains d'entre eux décrivent l'amplificateur au moyen d'un modèle petit signal. Par définition, le modèle petit signal restreint l'amplificateur à un comportement linéaire autour d'un point de polarisation spécifique. Il est donc absolument impossible à un modèle petit signal de refléter un comportement non-linéaire. Même lorsqu'il ne s'agit pas d'un modèle petit signal, modéliser avec un schéma équivalent conduit inévitablement à faire des choix topologiques même si cela est indirect. Ces modèles ne conviennent donc pas.

Dans le cas des assemblages de blocs de construction, chaque bloc est considéré comme faiblement couplés et donc indépendant des autres. Ces blocs de construction peuvent être décrits avec soit un schéma au niveau transistor, soit un réseau électrique équivalent, soit des équations et des fonctions de transfert. Les deux premiers types de description présentent les même inadéquations que celles évoqués pour les deux premières catégories de modèles de haut niveau évoqués ci-dessus. La modélisation des blocs de construction au moyen d'équations ou de fonctions de transfert mérite d'avantage d'attention. Ce dernier type description peut sembler ne guère se différencier de celui employé pour décrire globalement le module analogique – qui est à un niveau d'abstraction supérieur à celui examiné ici. Il convient de prendre garde à ne pas confondre technique de modélisation et niveau d'abstraction. Une décomposition pertinente en blocs de construction introduit des informations supplémentaires et permet de traiter les saturations et autres non-linéarités. Un assemblage judicieusement de fonctions de transfert aboutit à une description plus précise qu'une seule fonction de transfert globale. Toutefois, à notre connaissance, dans la littérature relative aux modèles formés par un assemblage de blocs de construction, les coefficients nécessaires aux équations ou aux fonctions de transfert sont extraits de schémas transistors ou de réseaux électriques équivalents. Les objections déjà émises concernant les choix topologiques induits sont valables ici aussi.

Aucun des modèles existants ne répond au besoin un modèle d'amplificateur en tension capable de rendre compte à la fois de la saturation en amplitude de la tension de sortie et des conséquences d'une vitesse de balayage finie sans pour autant faire d'hypothèse topologique ni technologique. Un nouveau modèle d'amplificateur opérationnel doit être créé : le modèle de l'amplificateur opérationnel simplifié.

V.2) Paramètres requis au niveau transistor à spécifier au niveau bloc analogique

Pour chaque niveau d'abstraction, il est possible de définir des paramètres propres à ce niveau et des paramètres hérités issus du raffinement. Les paramètres propres au niveau sont ceux qui sont nécessaires pour caractériser le niveau d'abstraction considéré mais qui peuvent être masqués au niveau d'abstraction supérieur. Les paramètres hérités sont ceux dont l'influence dépasse le niveau d'abstraction considéré et dont il est nécessaire de tenir compte au niveau immédiatement supérieur, voire au delà. Dans un processus de raffinement bien formé, la caractérisation réalisée à un niveau d'abstraction donné sert de spécification au niveau d'abstraction inférieur pour les paramètres hérités.

Pour s'insérer correctement dans le raffinement des parties analogiques, les modèles d'amplificateurs opérationnels étudiés ici au niveau bloc analogique devront donc spécifier les paramètres hérités pour le niveau transistor. Quels sont -ils ?

P.E. Allen et D.R. Holberg dans [69] listent les conditions limites et les paramètres requis pour la conception des amplificateurs opérationnels au niveau transistor. Cette liste est rappelée ici :

Conditions limites

- Paramètres technologiques liés aux procédés de fabrication (V_T , K , C_{ox} , etc.)
- Tension nominale d'alimentation et plage de tension de fonctionnement
- Courant maximal d'alimentation, consommation maximale de l'amplificateur
- Plage de température en fonctionnement

Paramètres

- Gain statique différentiel en tension* de l'amplificateur
- Produit gain-bande* (*Gain Bandwidth Product – GBP*)
- Temps d'établissement* (*Settling time*)
- Vitesse de balayage* (*Slew rate – Sr*)
- Plage de tension de mode commun* (*Input Common-Mode Range – ICMR*)
- Taux de réjection du mode commun* (*Common-Mode Rejection Ratio – CMRR*)
- Taux de réjection de l'alimentation* (*Power-Supply Rejection Ratio – PSRR*)
- Tension de décalage en entrée* (*Input Offset Voltage*)
- Dynamique de sortie, excursion de tension en sortie (*Output Voltage Swing*)
- Résistance de sortie
- Bruit
- Surface

Parmi ces conditions limites et paramètres, la surface et les paramètres technologiques sont propres au niveau transistor. Ils n'ont pas à apparaître au niveau bloc analogique ni aux niveaux d'abstraction supérieurs. La prise en compte de tous les paramètres et limitations restants conduira à un modèle réaliste de l'amplificateur opérationnel au niveau bloc analogique. Cependant un tel modèle sera encore très détaillé et donc lourd à simuler, ce qui ne correspond pas à ce qui est souhaité dans le cadre d'une exploration d'architecture. Par conséquent, un modèle plus abstrait qui n'intègre que les limitations les plus importantes pour mener à bien le choix d'une implémentation du module analogique est à créer : le modèle de l'amplificateur opérationnel simplifié. Ce modèle qui privilégie une description comportementale en tension est présenté et étudié au chapitre V.3.

* voir glossaire

V.3) Modèle de l'amplificateur opérationnel simplifié

L'amplificateur opérationnel simplifié est un modèle intermédiaire entre l'amplificateur idéal et l'amplificateur réel. Il reproduit certaines des limitations et non-linéarités de l'amplificateur réel mais ignore ou simplifie certains de ces comportements. Ceci permet d'obtenir un modèle d'un niveau d'abstraction relativement élevé tout en intégrant les limitations et non-linéarités les plus importantes. Par exemple, la saturation de la tension de sortie est reproduite, limitation qui n'est pas prise en compte dans le cas de l'amplificateur idéal. A l'inverse il sera considéré que la vitesse de balayage est identique en montée et en descente ce qui n'est pas toujours vrai dans le cas d'un amplificateur réel. La liste des hypothèses retenues est donnée au chapitre V.3.1. Le modèle étudié ici décrit un amplificateur avec une sortie simple en tension.

Le modèle d'amplificateur opérationnel simplifié fait abstraction de l'architecture interne de l'amplificateur. Il est donc indépendant de celle-ci. Il est également indépendant des contraintes dues aux procédés de fabrication et à la technologie à employer pour fabriquer l'amplificateur.

V.3.1) Hypothèses

Les différentes hypothèses retenues pour définir le modèle d'amplificateur opérationnel simplifié sont présentées ici. Les limitations et les non-linéarités prises en compte ou au contraire ignorées ou simplifiées sont listées.

- i) Limitations et non-linéarités prises en compte dans le modèle d'amplificateur opérationnel simplifié**
 - Gain statique différentiel en tension de valeur finie
 - Atténuation en fréquence du gain différentiel en tension, le produit gain-bande est fini
 - Saturation de la tension de sortie et limitation de la dynamique de sortie
 - Limitation de la vitesse de variation de la tension de sortie, la vitesse de balayage de l'amplificateur est finie
 - Réponse indicielle (réponse à l'échelon) en forme d'arc d'exponentielle, si l'amplitude de l'échelon conduit à une vitesse de variation en sortie inférieure à la vitesse de balayage, ou en forme de rampe de pente égale à la vitesse de balayage dans le cas contraire
 - Arrêt de la simulation, avec affichage d'un message d'erreur, si la tension d'alimentation est en dehors de la plage de fonctionnement spécifiée
- ii) Limitations et non-linéarités ignorées ou simplifiées dans le modèle d'amplificateur opérationnel simplifié**
 - Aucune restriction quand au courant fourni par l'alimentation (alimentation parfaite en tension)
 - Les variations des tensions d'alimentation sont suffisamment lentes devant les signaux d'entrée pour être considérées comme des tensions constantes
 - Température sans influence
 - Les courants dans les entrées de l'amplificateur opérationnel sont négligeables (impédance d'entrée infinie)
 - Tension de décalage en entrée nulle
 - Courant de décalage en entrée nul
 - Plage de tension de mode commun en entrée allant de la tension d'alimentation négative à la tension d'alimentation positive
 - Réjection totale du mode commun en entrée
 - Réjection totale des alimentations dans le signal de sortie.
 - Rétablissement instantané après saturation ; temps de reprise (*recovery time*) nul
 - Vitesse de variation de la tension de sortie identique à la montée et à la descente
 - Les signaux appliqués sur les entrées de l'amplificateur sont implicitement écrêtés par rapport aux alimentations de l'amplificateur. Une surtension en entrée n'est pas destructive.
 - Tension de déchet en sortie par rapport à la tension d'alimentation positive identique à la tension de déchet par rapport à la tension d'alimentation négative
 - Résistance série de sortie nulle
 - Amplificateur non bruité

iii) Autres hypothèses pour le modèle d'amplificateur opérationnel simplifié

Il est supposé que la partie intéressante de la réponse en fréquence de l'amplificateur opérationnel simplifié est comprise entre le continu et la fréquence de transition*. L'étude portera sur cette plage de fréquences.

Les hypothèses suivantes sont également faites :

- la réponse transitoire de l'amplificateur est apériodique (pas d'oscillation)
- le pôle dominant de la fonction de transfert est un pôle basse fréquence, de fréquence nettement inférieure à la fréquence de transition
- les autres pôles de la fonction de transfert sont des pôles haute fréquence largement au-delà de la fréquence de transition ; leur influence est négligeable

Les trois hypothèses précédentes sont fortement simplificatrices et élève le niveau d'abstraction.

Compte-tenu des hypothèses précédentes, la réponse en fréquence de l'amplificateur opérationnel simplifié peut être approximée par une fonction de transfert du premier ordre avec un pôle réel. La réponse impulsionnelle sera donc exponentielle.

Puisque la réponse transitoire est apériodique, le temps d'établissement peut être calculé à partir de la réponse impulsionnelle. La constante de temps est déduite de la fréquence de coupure de la réponse en fréquence de l'amplificateur. Le temps d'établissement n'est donc pas exprimé séparément.

V.3.2) Conventions et notations

- A : gain statique différentiel en tension de l'amplificateur.
- V_{dd} : tension du plot d'alimentation positive de l'amplificateur
- V_{ss} : tension du plot d'alimentation négative de l'amplificateur
- V_{sply} : tension d'alimentation vue par l'amplificateur
- $V_{splyMax}$: valeur maximale autorisée pour l'alimentation vue par l'amplificateur
- $V_{splyMin}$: valeur minimale autorisée pour l'alimentation vue par l'amplificateur
- V_{bias} : tension de repos de la sortie
- V^+ : tension sur l'entrée positive de l'amplificateur
- V^- : tension sur l'entrée négative de l'amplificateur
- V_{in} : tension différentielle d'entrée de l'amplificateur
- V_{i_0} : valeur initiale de la tension différentielle d'entrée
- V_{inNS} : tension différentielle d'entrée de l'amplificateur sans saturation
- V_{out} : tension de sortie de l'amplificateur
- V_{o_0} : valeur initiale de la tension de sortie
- V_{outNS} : tension de sortie avec un produit gain-bande fini mais sans saturation ni limitation de la vitesse de variation
- V_{outSat} : tension de sortie avec un produit gain-bande fini et avec saturation mais sans limitation de la vitesse de variation
- V_{outSr} : tension de sortie avec un produit gain-bande fini et vitesse de variation finie mais sans saturation
- V_{oLoss} : tension de déchet en sortie
- V_{oSatH} : seuil de saturation supérieure de la tension de sortie
- V_{oSatL} : seuil de saturation inférieure de la tension de sortie
- $MSatL, MSatH, MNoSat$: modes de comportement de la sortie vis à vis de la saturation en sortie
- $MSrL, MSrH, MNoSr$: modes de comportement de la sortie vis à vis de la limitation de la vitesse de variation de la tension de sortie
- $SatMode$: variable d'état indiquant le mode de comportement de la sortie vis à vis de la saturation
- $SrMode$: variable d'état indiquant le mode de comportement de la sortie vis à vis de la limitation de la vitesse de variation de la tension de sortie

* Voir glossaire

- Sr : vitesse de balayage, vitesse maximale à laquelle l'amplificateur peut effectuer une variation de tension
- f_t : fréquence de transition, fréquence à laquelle l'amplitude de la fonction de transfert vaut 1
- f_c : fréquence de coupure, fréquence à laquelle l'amplitude de la fonction de transfert subit une atténuation de 3 dB par rapport au gain statique en tension
- τ : constante de temps
- GBP : produit gain-bande

Transformée de Laplace

Les notations utilisées pour la transformée de Laplace sont les suivantes.

La variable du temps sera notée t et la variable de Laplace p .

La notation retenue pour la transformée de Laplace F d'une fonction f du temps est la suivante :

$$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}, F(p) = TL[f(t)]$$

La notation retenue pour la transformée inverse de Laplace est la suivante :

$$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}, f(t) = TL^{-1}[F(p)]$$

Pour mémoire, les transformées de Laplace remarquables et les propriétés utilisées dans cet exposé sont rappelées à l'annexe 1.

V.3.3) Définition du modèle de l'amplificateur opérationnel simplifié

Les définitions nécessaires à la construction du modèle d'amplificateur opérationnel simplifié sont réunies dans la première partie de ce paragraphe. Les propriétés que le modèle devra respecter sont données dans la deuxième partie.

i) Définitions

Définition 1 Plage de tensions autorisées

Une plage de tensions autorisées est un intervalle de tension que doivent respecter les tensions appliquées à un circuit électronique pour que celui-ci fonctionne correctement. Il peut y avoir plusieurs plages de tensions autorisées pour un même circuit, chaque intervalle concernant une tension différente.

Définition 2 Tension autorisée

Une tension est dite autorisée quand elle est dans la plage de tensions autorisées qui la contraint.

Postulat 1 Tensions appliquées sur les plots d'alimentation

Dans le modèle d'amplificateur opérationnel simplifié, la tension V_{dd} du plot d'alimentation positive et la tension V_{ss} du plot d'alimentation négative sont assimilées à des constantes réelles telles que :

$$V_{dd} > V_{ss}$$

Considérer que la tension délivrée sur un plot d'alimentation est constante dans le temps et ceci quel que soit le courant absorbé, est une hypothèse simplificatrice. Les sources de tension connectées sur les plots d'alimentations devront être des sources de tension continue parfaites.

Définition 3 Tension d'alimentation vue par l'amplificateur

La tension d'alimentation vue par l'amplificateur, V_{sply} , est la différence de tension entre les plots d'alimentation. Dans ce modèle, elle est constante car les tensions des plots d'alimentation, V_{dd} et V_{ss} , sont constantes.

$$V_{sply} = V_{dd} - V_{ss}$$

Le respect du postulat 1 implique que $V_{sply} > 0$

Définition 4 Plage de tensions autorisées pour l'alimentation de l'amplificateur

La plage de tensions autorisées pour l'alimentation de l'amplificateur est définie par l'intervalle $[VsplyMin, VsplyMax]$ où $VsplyMin$ et $VsplyMax$ sont deux constantes réelles.

La tension d'alimentation vue par l'amplificateur doit être dans cette plage

$$VsplyMin \leq V_{sply} \leq VsplyMax$$

Définition 5 Tension de repos en sortie

La tension de sortie évolue autour de la tension de repos $Vbias$ qui est au point milieu entre les tensions d'alimentation positive et négative.

$$Vbias = \frac{(Vdd - Vss)}{2} + Vss = \frac{Vsply}{2} + Vss$$

Définition 6 Conditions initiales

Au début de la simulation la tension différentielle d'entrée est nulle et la tension de sortie est égale à la tension de repos.

$$\begin{cases} Vi_0 = 0 \\ Vo_0 = Vbias \end{cases}$$

Postulat 2 Valeur du gain statique différentiel en tension

Le gain statique différentiel en tension de l'amplificateur est une constante réelle au moins égale¹³ à 10000.

$$A \geq 10^4$$

Le choix de considérer le gain statique différentiel en tension constant donc indépendant de la tension d'alimentation est une hypothèse simplificatrice qui élève le niveau d'abstraction.

ii) Propriétés

Les propriétés énoncées dans ce paragraphe définissent un cadre au modèle de l'amplificateur opérationnel simplifié. Ce modèle devra donc obligatoirement vérifier ces propriétés.

Propriété 1 Tension de sortie quand la tension différentielle d'entrée est nulle

Quand la tension d'entrée est maintenue à zéro, une fois le transitoire terminé, la tension de sortie est égale à la tension de repos $Vbias$.

Propriété 2 Comportement en absence de boucle de rétroaction

En absence de boucle de rétroaction¹⁴, l'amplificateur opérationnel simplifié doit se comporter comme un comparateur.

Propriété 3 Forme de la réponse du montage en contre réaction

La réponse du montage en contre réaction doit être asymptotiquement stable et apériodique.

Propriété 4 Forme de la réponse du montage en rétroaction positive

La réponse du montage en rétroaction positive doit être instable et apériodique.

13 Le gain statique différentiel de l'AOP 741 était de 10000. Tous les amplificateurs modernes ont des performances supérieures au 741

14 Afin d'éviter les confusions, l'expression « gain en absence de boucle de rétroaction » est préférée ici à « gain en boucle ouverte ». En effet chez certains auteurs, cette dernière expression ne peut s'employer que dans le cas d'un système bouclé et désigne le gain à l'endroit où la boucle est ouverte (en générale à l'entrée soustractive de l'unité mesurant l'erreur). Il s'agit alors du produit du gain de la chaîne directe (qui correspondrait ici à l'amplificateur) par le gain de la boucle de retour.

Propriété 5 Tension de sortie bornée par les seuils de saturation

La tension de sortie de l'amplificateur est toujours comprise entre le seuil de saturation positive V_{oSatH} et le seuil de saturation négative V_{oSatL} .

$$V_{oSatL} \leq V_{out} \leq V_{oSatH}$$

Propriété 6 Vitesse de variation limitée à la vitesse de balayage

Le vitesse de variation de la tension de sortie est bornée entre la vitesse de balayage à la descente $-S_r$ et la vitesse de balayage à la montée S_r .

$$\forall t \in \mathbb{R}, \\ -S_r \leq \frac{dV_{out}(t)}{dt} \leq S_r$$

V.3.4) Étude du modèle sans saturation, ni limitation de la vitesse de variation

Dans un premier temps l'étude porte sur un modèle avec un produit gain-bande fini sans prendre en compte les saturations en entrée et en sortie, ni la limitation de la vitesse de variation de la tension de sortie. La propriété 5 et la propriété 6 ne seront donc pas vérifiées. La fonction de transfert, l'équation temporelle associée, la réponse impulsionnelle et la réponse indicielle (réponse à un échelon) sont examinées.

L'étude des saturations et limitations sera faite au paragraphe V.3.5.

i) Définitions spécifiques au modèle sans saturation, ni limitation

Les définitions suivantes s'ajoutent aux définitions données au § V.3.3)i

Définition 7 Tension différentielle d'entrée sans saturation

La tension différentielle d'entrée sans saturation est la différence de tension entre l'entrée « + » et l'entrée « - » de l'amplificateur en absence de saturation de ces entrées. Elle est appelée V_{inNS} dans le domaine temporel. Sa transformée de Laplace est appelée V_{INNS}

$$\forall t \in \mathbb{R}, V_{inNS}(t) = V^+(t) - V^-(t) \\ \forall t \in \mathbb{R}, \forall p \in \mathbb{C}, V_{INNS}(p) = TL[V_{inNS}(t)]$$

Définition 8 Tension de sortie sans saturation ni limitation de sa vitesse de variation

La fonction temporelle décrivant la tension de sortie sans saturation ni limitation de sa vitesse de variation est appelée V_{outNS} .

Définition 9 Fonction d'étude de la tension de sortie

Pour étudier la fonction gouvernant la tension de sortie, une fonction y est définie comme la différence entre la tension de sortie (sans saturation, ni limitation) et la tension de repos en sortie. Sa transformée de Laplace est appelée Y .

$$\forall t \in \mathbb{R}, y(t) = V_{outNS}(t) - V_{bias} \\ \forall t \in \mathbb{R}, \forall p \in \mathbb{C}, Y(p) = TL[y(t)]$$

ii) Amplificateur non-bouclé : fonction de transfert, équation temporelle, réponses impulsionnelle et indicielle

Compte-tenu des hypothèses énoncées au chapitre V.3.1, la réponse en fréquence de l'amplificateur opérationnel simplifié peut être approximée par une fonction de transfert du premier ordre avec un pôle réel.

L'équation (8) donne la forme typique d'une fonction de transfert d'un système linéaire du premier ordre de gain A , où τ est une constante réelle positive. La figure 40 donne le gabarit du gain en fréquence d'une telle fonction de transfert.

Fonction de transfert de l'amplificateur opérationnel simplifié :

$$\boxed{\forall p \in \mathbb{C}, H(p) = \frac{A}{1 + \tau p}} \quad (8)$$

Pour le modèle d'amplificateur opérationnel simplifié, le coefficient A correspond au gain statique différentiel de l'amplificateur. La constante τ est la constante de temps du système. Elle peut être calculée soit à partir de la fréquence de coupure* à -3 dB, f_c , soit à partir du produit gain-bande* GBP .

$$\tau = \frac{1}{2\pi f_c} \quad (9)$$

$$\tau = \frac{A}{2\pi GBP} \quad (10)$$

L'entrée du système est représentée par la fonction $VinNS$ (cf. définition 7) tandis que la fonction y (cf. définition 9) décrit la sortie de ce même système. Avec les transformées de Laplace de ces deux fonctions, respectivement $VINNS$ et Y , la fonction de transfert H s'écrit :

$$\forall p \in \mathbb{C}, H(p) = \frac{Y(p)}{VINNS(p)}$$

d'où :

$$\forall p \in \mathbb{C}, Y(p) = H(p) \cdot VINNS(p) \quad (11)$$

Pour établir l'équation temporelle, $H(p)$ est remplacée dans (11) en utilisant (8), et on en déduit

$$Y(p) + \tau p \cdot Y(p) = A \cdot VINNS(p) \quad (12)$$

L'utilisation des propriétés de la transformée de Laplace et l'utilisation de la transformée de Laplace inverse (détail du calcul en annexe A 4.1) conduit à :

$$\tau \cdot \frac{d y(t)}{dt} + y(t) = A \cdot VinNS(t)$$

L'équation temporelle générale est obtenue à l'aide de la définition de y .

$$\text{Équation temporelle : } \boxed{\forall t \in \mathbb{R}, \tau \cdot \frac{d VoutNS(t)}{dt} + VoutNS(t) = A \cdot VinNS(t) + Vbias} \quad (13)$$

Pour le calcul de la réponse impulsionnelle, une impulsion de Dirac*, notée δ , est appliquée à l'entrée, c'est-à-dire que l'on pose $VinNS(t) = \delta$. La transformée de Laplace d'une impulsion valant la valeur constante 1, on a $VINNS(p) = 1$. L'équation (11) devient donc ici :

$$Y(p) = \frac{A}{1 + \tau p}$$

L'identification avec une transformée de Laplace remarquable et l'utilisation de la définition de la fonction y conduisent à la formule de la réponse impulsionnelle (détail du calcul en annexe A 4.1).

$$\text{Réponse impulsionnelle : } \boxed{\forall t \in \mathbb{R}, VoutNS(t) = \frac{A}{\tau} \cdot e^{-\frac{t}{\tau}} + Vbias} \quad (14)$$

Pour le calcul de la réponse indicielle, le signal d'entrée est un échelon d'amplitude Ech . En notant $u(t)$ la fonction d'Heaviside*, le signal d'entrée est alors de la forme $VinNS(t) = Ech \cdot u(t)$

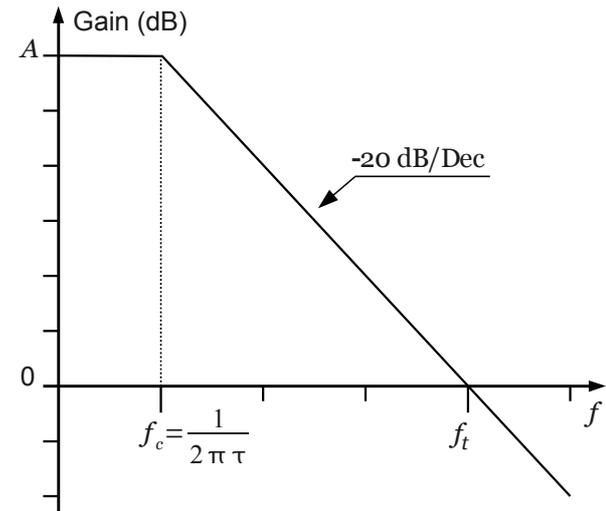


Fig. 40 : Gabarit du gain en fréquence d'un système du premier ordre

* Voir glossaire

Sa transformée de Laplace est alors

$$VINNS(p) = \frac{Ech}{p}$$

L'équation (11) devient :

$$Y(p) = \frac{Ech}{p} \cdot \frac{A}{(1 + \tau p)}$$

L'identification avec une transformée de Laplace remarquable et l'utilisation de la définition de la fonction y conduisent à la formule de la réponse impulsionnelle (détail du calcul en annexe A.4.1).

$$\text{Réponse indicielle : } \boxed{\forall t \in \mathbb{R}, V_{outNS}(t) = A \cdot Ech \cdot \left[1 - e^{-\frac{t}{\tau}} \right] + V_{bias}} \quad (15)$$

iii) Montage en contre réaction

Afin de faciliter la compréhension, le lecteur est invité à se reporter à la figure 36. Le détail des calculs menés dans ce paragraphe est donné à l'annexe A.4.2) et les conditions de stabilité d'un système linéaire sont rappelées à l'annexe 2.

La tension de sortie étant centrée autour de V_{bias} , V_{out} doit être remplacée par la fonction y (cf. définition 9). Les tensions appliquées en entrée du montage étant désignées par e_1 et e_2 , l'analyse du schéma de la figure 36 conduit à :

$$\forall t \in \mathbb{R}, \begin{cases} V^+(t) = e_2(t) \\ V^-(t) = \frac{R1 \cdot y(t) + R2 \cdot e_1(t)}{(R1 + R2)} \end{cases}$$

L'expression de la tension différentielle d'entrée de l'amplificateur en contre réaction en découle directement.

$$\forall t \in \mathbb{R}, V_{inNS}(t) = V^+(t) - V^-(t) = e_2(t) - \frac{R2 \cdot e_1(t)}{(R1 + R2)} - \frac{R1}{(R1 + R2)} \cdot y(t) \quad (16)$$

Soient E_1 et E_2 les transformées de Laplace de e_1 et e_2 , $VINNS$ celle de V_{inNS}

$$\begin{aligned} \forall t \in \mathbb{R}, \forall p \in \mathbb{C}, \\ E_1(p) &= TL[e_1(t)] \\ E_2(p) &= TL[e_2(t)] \\ VINNS(p) &= TL[V_{inNS}(t)] \end{aligned}$$

Avec E_1 , E_2 et $VINNS$, la transformée de Laplace de (16) donne :

$$\forall p \in \mathbb{C}, VINNS(p) = E_2(p) - \frac{R2 \cdot E_1(p)}{(R1 + R2)} - \frac{R1}{(R1 + R2)} \cdot Y(p) \quad (17)$$

Le remplacement de $VINNS(p)$ dans (12) avec la valeur trouvée dans (17) conduit à

$$\forall p \in \mathbb{C}, Y(p) = \frac{A}{\left[\frac{(R1 + R2)}{R1} \cdot (1 + \tau p) + A \right]} \cdot \left[\frac{(R1 + R2)}{R1} \cdot E_2(p) - \frac{R2}{R1} \cdot E_1(p) \right] \quad (18)$$

On pose :

$$\forall p \in \mathbb{C}, VSIG(p) = \frac{(R1 + R2)}{R1} \cdot E_2(p) - \frac{R2}{R1} \cdot E_1(p) \quad (19)$$

$VSIG(p)$ s'interprète comme la transformée de Laplace du signal différentiel appliqué à l'entrée du montage. La fonction de transfert $Hcr(p)$ du montage en contre réaction est donc déterminée par :

$$\forall p \in \mathbb{C}, Y(p) = Hcr(p) \cdot VSIG(p) \quad (20)$$

Fonction de transfert du montage en contre réaction :

$$\boxed{\forall p \in \mathbb{C}, Hcr(p) = \frac{A}{\frac{(RI + R2)}{RI} \cdot (1 + \tau p) + A}} \quad (21)$$

La fonction de transfert $Hcr(p)$ a un pôle p_1 .

$$p_1 = -\frac{1}{\tau} \cdot \left[1 + \frac{A \cdot RI}{(RI + R2)} \right]$$

Réponse transitoire

La stabilité du système est donnée par le signe de la partie réelle du pôle p_1 .

τ est une constante réelle positive (cf. (10) au ii. Dans le montage étudié ici, $R1$ et $R2$ sont des résistances pures, ce sont donc des grandeurs réelles positives. Le gain statique différentiel A est lui aussi réel et positif. Par conséquent, p_1 est un pôle réel négatif.

Avec le modèle de l'amplificateur opérationnel simplifié, **la réponse du montage en contre réaction** est donc **asymptotiquement stable et apériodique**, ce qui est conforme à la réalité.

Réponse indicielle du montage en contre réaction

La réponse indicielle du montage est obtenue en appliquant un échelon de tension sur chacune des entrées.

$$\forall t \in \mathbb{R}, e_1(t) = Ech_1 \cdot u(t) ; e_2(t) = Ech_2 \cdot u(t)$$

Leurs transformées de Laplace respectives sont :

$$\forall p \in \mathbb{C}, E_1(p) = \frac{Ech_1}{p} ; E_2(p) = \frac{Ech_2}{p}$$

L'équation (18) devient alors :

$$\forall p \in \mathbb{C}, Y(p) = \frac{A}{\left[\frac{(RI + R2)}{RI} \cdot (1 + \tau p) + A \right]} \cdot \left[\frac{(RI + R2)}{RI} \cdot \frac{Ech_2}{p} - \frac{R2}{RI} \cdot \frac{Ech_1}{p} \right]$$

L'identification avec des transformées de Laplace remarquables, la restriction du résultat aux temps positifs pour lesquels la fonction d'Heaviside n'est pas nulle et l'utilisation de la définition de la fonction y conduisent à la formule de la réponse indicielle du montage en contre réaction.

Réponse indicielle du montage en contre réaction

$$\boxed{\forall t \in \mathbb{R}, V_{outNS}(t) = \left[1 - e^{-\frac{t}{\tau} \left[1 + \frac{A \cdot RI}{(RI + R2)} \right]} \right] \cdot \frac{1}{\frac{(RI + R2)}{A \cdot RI} + 1} \cdot \left[\frac{(RI + R2)}{RI} \cdot Ech_2 - \frac{R2}{RI} \cdot Ech_1 \right] + V_{bias}} \quad (22)$$

Le rapport des résistances $\frac{R2}{RI}$ étant toujours choisi de façon à être petit devant A , l'approximation suivante est acceptable (justification à l'annexe A 4.2) :

$$\frac{1}{\left[\frac{(RI + R2)}{A \cdot RI} + 1 \right]} \approx 1$$

Par conséquent :

$$\lim_{t \rightarrow \infty} V_{outNS}(t) = \frac{(RI + R2)}{RI} \cdot Ech_2 - \frac{R2}{RI} \cdot Ech_1 + V_{bias} \quad (23)$$

$$\Leftrightarrow \lim_{t \rightarrow \infty} V_{outNS}(t) = \frac{R2}{RI} \cdot [Ech_2 - Ech_1] + Ech_2 + V_{bias} \quad (24)$$

Les valeurs limites (23) et (24) correspondent à la tension de sortie en régime établi. L'équation (23) est à rapprocher de l'équation (5) de la tension de sortie du montage en contre réaction avec l'amplificateur idéal (avec $V_{bias} = 0$).

iv) Montage en rétroaction positive

Afin de faciliter la compréhension, le lecteur est invité à se reporter à la figure 37. Le détail des calculs menés dans ce paragraphe est donné à l'annexe A.4.3) et les conditions de stabilité d'un système linéaire sont rappelées à l'annexe 2.

La tension de sortie étant centrée autour de V_{bias} , V_{out} doit être remplacée par la fonction y (cf. définition 9). Les tensions appliquées en entrée du montage étant désignées par e_1 et e_2 , l'analyse du schéma de la figure 37 conduit à :

$$\forall t \in \mathbb{R}, \begin{cases} V^+(t) = \frac{RI \cdot y(t) + R2 \cdot e_1(t)}{(RI + R2)} \\ V^-(t) = e_2(t) \end{cases} \quad (25)$$

L'expression de la tension différentielle d'entrée de l'amplificateur en contre réaction en découle directement.

$$\forall t \in \mathbb{R}, V_{inNS}(t) = V^+(t) - V^-(t) = \frac{R2 \cdot e_1(t)}{(RI + R2)} - e_2(t) + \frac{RI}{(RI + R2)} \cdot y(t) \quad (26)$$

Soient E_1 et E_2 les transformées de Laplace de e_1 et e_2 , V_{INNS} celle de V_{inNS}

$$\begin{aligned} \forall t \in \mathbb{R}, \forall p \in \mathbb{C}, \\ E_1(p) &= TL[e_1(t)] \\ E_2(p) &= TL[e_2(t)] \\ V_{INNS}(p) &= TL[V_{inNS}(t)] \end{aligned}$$

Avec E_1 , E_2 et V_{INNS} , la transformée de Laplace de (26) donne :

$$\forall p \in \mathbb{C}, V_{INNS}(p) = \frac{R2 \cdot E_1(p)}{(RI + R2)} - E_2(p) + \frac{RI}{(RI + R2)} \cdot Y(p) \quad (27)$$

Le remplacement de $V_{INNS}(p)$ dans (12) avec la valeur trouvée dans (27) conduit à :

$$\forall p \in \mathbb{C}, Y(p) = \frac{A}{\left[A - \frac{(RI + R2)}{RI} \cdot (1 + \tau p) \right]} \cdot \left[\frac{(RI + R2)}{RI} \cdot E_2(p) - \frac{R2}{RI} \cdot E_1(p) \right]$$

Avec $V_{SIG}(p)$ le signal appliqué à l'entrée du montage tel que défini en (19), la fonction de transfert $Hr(p)$ du montage en rétroaction positive est donc déterminée par :

$$\forall p \in \mathbb{C}, Y(p) = Hr(p) \cdot V_{SIG}(p) \quad (28)$$

Fonction de transfert du montage en rétroaction positive

$$\boxed{\forall p \in \mathbb{C}, Hr(p) = \frac{A}{A - \frac{(RI + R2)}{RI} \cdot (1 + \tau p)}} \quad (29)$$

La fonction de transfert $Hr(p)$ a un pôle p_2 .

$$p_2 = \frac{1}{\tau} \cdot \left[\frac{A \cdot R1}{(R1 + R2)} - 1 \right]$$

Réponse transitoire

La stabilité du système est donnée par le signe de la partie réelle du pôle p_2 .

τ est une constante réelle positive (cf. (10) au ii). Dans le montage étudié ici, $R1$ et $R2$ sont des résistances pures, ce sont donc des grandeurs réelles positives. Le gain statique différentiel A est lui aussi réel et positif. Par conséquent, p_2 est un pôle réel. La réponse du montage sera aperiodique. Il faut maintenant étudier le signe de p_2 .

$$p_2 > 0 \Rightarrow \frac{A \cdot R1}{(R1 + R2)} > 1$$

Selon le postulat 2, $A \geq 10^4$. Puisque $R1$ et $R2$ sont des grandeurs positives, leur rapport est également positif. Dans la pratique, le rapport $\frac{R2}{R1}$ reste inférieur à 1000. Par conséquent, le pôle p_2 est toujours positif.

Le pôle p_2 étant un réel toujours positif, avec le modèle de l'amplificateur opérationnel simplifié, **la réponse du montage en rétroaction positive est donc instable et aperiodique**, ce qui est conforme à la réalité.

V.3.5) Étude du modèle avec saturations et vitesse de balayage finie

Le modèle d'amplificateur opérationnel simplifié est ici augmenté de façon à prendre en compte les saturations et les limitations dominantes. L'étude est menée en quatre étapes :

- saturation de la tension en entrée,
- saturation de la tension de sortie
- limitation de la vitesse de variation de la tension de sortie du fait d'une vitesse de balayage finie
- combinaison de la saturation en sortie et de la vitesse de balayage fine

i) Saturation en entrée

Les signaux appliqués sur les entrées V^+ et V^- de l'amplificateur sont implicitement écrêtés : l'amplitude prise en compte ne dépasse pas les tensions d'alimentation de l'amplificateur Vdd et Vss .

La prise en compte de la saturation en entrée conduit à définir la tension différentielle d'entrée Vin . Celle-ci correspond à la tension différentielle d'entrée sans saturation $VinNS$ quand aucune des entrées ne sature.

Définition 10 Tension différentielle d'entrée

$\forall t \in \mathbb{R}$,

$$Vin(t) = \begin{cases} V^+(t) - V^-(t) & \text{Si } [Vss < V^+(t) < Vdd] & \text{et } [Vss < V^-(t) < Vdd] \\ Vdd - V^-(t) & \text{Si } [V^+(t) \geq Vdd] & \text{et } [Vss < V^-(t) < Vdd] \\ Vss - V^-(t) & \text{Si } [V^+(t) \leq Vss] & \text{et } [Vss < V^-(t) < Vdd] \\ V^+(t) - Vdd & \text{Si } [Vss < V^+(t) < Vdd] & \text{et } [V^-(t) \geq Vdd] \\ V^+(t) - Vss & \text{Si } [Vss < V^+(t) < Vdd] & \text{et } [V^-(t) \leq Vss] \\ Vdd - Vss & \text{Si } [V^+(t) \geq Vdd] & \text{et } [V^-(t) \leq Vss] \\ 0 & \text{Si } [V^+(t) \geq Vdd] & \text{et } [V^-(t) \geq Vdd] \\ 0 & \text{Si } [V^+(t) \leq Vss] & \text{et } [V^-(t) \leq Vss] \end{cases}$$

Postulat 3 Entrées indestructibles

La saturation des entrées n'engendre aucun dommage. L'amplificateur continue de fonctionner quelles que soient les tensions appliquées en entrée et quelle qu'en soit la valeur de mode commun. Il s'agit ici d'une hypothèse fortement simplificatrice.

ii) Saturation en sortie

La tension de sortie sature parce qu'elle ne peut dépasser la valeur des tensions d'alimentation. Dans ce qui suit, la tension de sortie avec saturation est appelée V_{outSat} .

Les seuils de saturation positive et négative, V_{oSatH} et V_{oSatL} , sont respectivement la plus grande et la plus petite valeur que peut atteindre la sortie (voir fig. 41). La tension de déchet V_{oLoss} est l'écart entre un seuil de saturation et la tension d'alimentation dont il est le plus proche. Dans le modèle de l'amplificateur opérationnel simplifié, la tension de déchet est identique vis à vis de la tension d'alimentation positive et vis à vis de la tension d'alimentation négative. Elle est en outre toujours inférieure à 20 % de la tension d'alimentation positive¹⁵. Les seuils de saturation en sortie sont normalement toujours inférieurs, en valeur absolue, aux tensions d'alimentation. Cependant, la tension de déchet peut, dans bien des cas, être estimée comme négligeable. Les seuils de saturation en sortie sont donc considérés comme inférieurs ou égaux aux tensions d'alimentation, en valeur absolue.

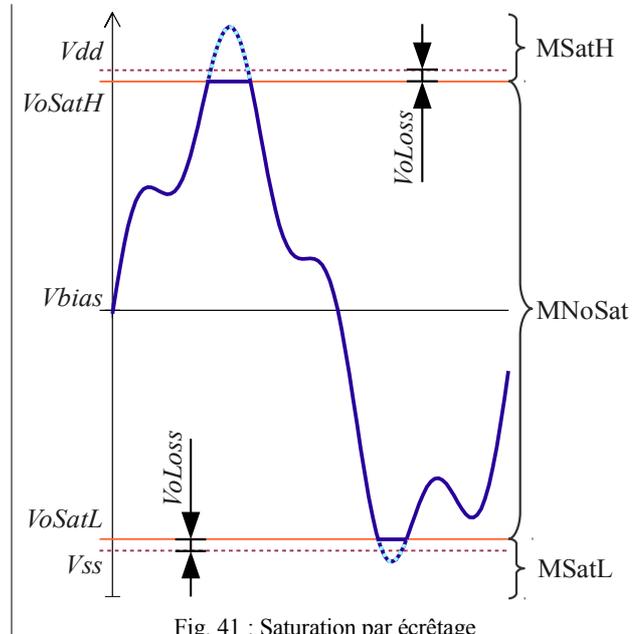


Fig. 41 : Saturation par écrêtage

Postulat 4 Tension de déchet en sortie

$$0 \leq V_{oLoss} < 20\% V_{dd}$$

Définition 11 Seuil de saturation supérieur

$$V_{oSatH} = V_{dd} - V_{oLoss}$$

Définition 12 Seuil de saturation inférieur

$$V_{oSatL} = V_{ss} + V_{oLoss}$$

Pour l'amplificateur opérationnel simplifié, la saturation en sortie est modélisée par un écrêtage. Quand la tension de sortie sans saturation passe au delà d'un des seuils de saturation, la tension de sortie avec saturation est maintenue égale à la tension de ce seuil de saturation. La fonction décrivant la tension de sortie avec saturation est donc définie en trois morceaux. Un mode est associé à chacun de ces trois morceaux selon la définition 13 ci-dessous. La variable $SatMode$ permet de distinguer dans quel mode est la sortie.

Définition 13 Modes de comportement de la sortie vis à vis de la saturation

Les trois modes de comportement de la sortie sont :

- $MSatL$, lorsque la sortie sature par rapport au seuil de saturation inférieur V_{oSatL}
- $MSatH$, lorsque la sortie sature par rapport au seuil de saturation supérieur V_{oSatH}
- $MNoSat$, lorsque la sortie ne sature pas

La variable d'état $SatMode$ associée à la sortie peut prendre l'une des valeurs ci-dessus.

$$SatMode = \{MSatL, MSatH, MNoSat\}$$

¹⁵ La tension de déchet en sortie de l'AOP 741 peut atteindre 3V avec une alimentation de $\pm 15V$. Les performances des amplificateurs modernes sont supérieures au 741. Pour un amplificateur rail à rail, la tension de déchet est inférieure à 1%.

Définition 14 Surveillance de la saturation

La fonction V_{outNS} qui décrit la tension de sortie sans saturation sert de fonction de surveillance pour la saturation. Elle permet de déterminer quand la sortie doit entrer en saturation et à quel moment elle doit en sortir pour retourner en mode non saturé (voir fig. 41).

$$\forall t \in \mathbb{R},$$

$$V_{outSat}(t) = \begin{cases} V_{oSatL} & \text{Si } V_{outNS}(t) \leq V_{oSatL} \\ V_{outNS}(t) & \text{Si } V_{oSatL} < V_{outNS}(t) < V_{oSatH} \\ V_{oSatH} & \text{Si } V_{outNS}(t) \geq V_{oSatH} \end{cases}$$

Définition 15 Transitions entre modes de comportement de la sortie vis à vis de la saturation

Quand la sortie n'est pas saturée, elle peut entrer en saturation par rapport au seuil de saturation inférieur ou au seuil de saturation supérieur. Par contre quand elle sature, elle n'est pas autorisée à passer directement dans l'autre mode de saturation, elle doit d'abord passer par le mode normal, c'est-à-dire non saturé (voir fig. 42).

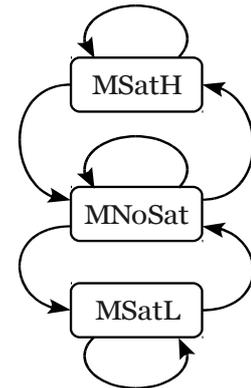


Fig. 42 : Transitions entre modes de saturation

Conditions aux limites lors du changement de mode de saturation et implémentation informatique

Les morceaux de la fonction décrivant la tension de sortie avec saturation sont raccordés. D'un point de vue purement mathématique, la définition 14 est suffisante pour garantir un raccordement correct. Toutefois sur le plan informatique, la simulation étant à pas discret – aussi fin soit-il – et la surveillance de la saturation étant effectuée à posteriori, la tension de sortie doit être corrigée lorsqu'un changement de mode est détecté. Au moment où il est constaté que la sortie entre en saturation, la valeur de la tension de sortie sans saturation V_{outNS} est déjà au delà du seuil de saturation, la tension de sortie avec saturation doit donc être forcée à la valeur du seuil de saturation puisqu'elle ne peut aller au delà (cf. fig. 41). A l'instant t_1 où il est constaté que la sortie quitte la saturation, la tension sans saturation est déjà en deçà du seuil de saturation, alors que sa valeur précédente était encore au delà. Il est nécessaire de ramener la tension de la sortie sans saturation à la valeur de la tension qu'a la sortie avec saturation à cet instant. Il faut donc appliquer l'égalité $V_{outNS}(t_1) = V_{outSat}(t_1)$, avant le passage en mode normal sans saturation.

iii) Limitation de la vitesse de variation

La tension de sortie de l'amplificateur ne peut varier plus vite que la vitesse de balayage de l'amplificateur Sr . Quand la vitesse de variation de la tension de sortie sans limitation devient supérieure à la vitesse de balayage, la vitesse de variation de la tension de sortie avec limitation est maintenue égale à la vitesse de balayage. Voir fig. 43.

La fonction qui décrit la tension de sortie de l'amplificateur avec une vitesse de balayage finie et un produit gain-bande fini est appelée V_{outSr} . Elle est définie en trois morceaux.

Un mode est associé à chacun des trois morceaux de la fonction V_{outSr} selon la définition 16 ci-dessous. La variable $SrMode$ permet de distinguer dans quel mode est la sortie.

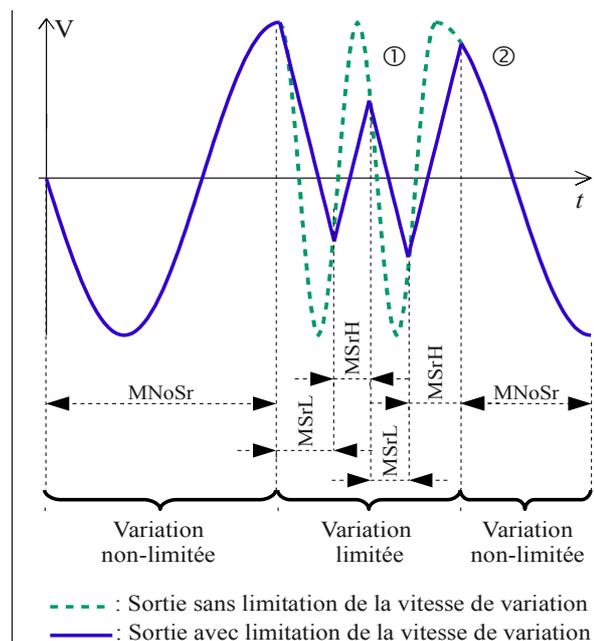


Fig. 43 : Vitesse de variation limitée à la vitesse de balayage

Définition 16 Modes de comportement de la sortie vis à vis de la limitation de la vitesse de variation

Les trois modes de comportement de la sortie sont :

- $MSrL$, lorsque la sortie est décroissante et que sa vitesse de variation est limitée à la vitesse de balayage à la descente $-Sr$
- $MSrH$, lorsque la sortie est croissante et que sa vitesse de variation est limitée à la vitesse de balayage à la montée Sr
- $MNoSr$, lorsque la vitesse de variation de la sortie n'est pas limitée

La variable d'état $SrMode$ associée à la sortie peut prendre l'une des valeurs ci-dessus.

$$SrMode = \{MSrL, MSrH, MNoSr\}$$

Définition 17 Surveillance de la vitesse de variation

La vitesse de variation de la tension de sortie est surveillée en examinant la dérivée de la fonction $VoutNS$ qui décrit la sortie sans saturation mais aussi sans limitation de sa vitesse de variation. Cette examen permet de déterminer à quel moment la sortie change de mode de comportement vis à vis de cette limitation.

$$\forall t \in \mathbb{R}, \quad \frac{dVoutSr(t)}{dt} = \begin{cases} -Sr & Si \quad \frac{dVoutNS(t)}{dt} \leq -Sr \\ \frac{dVoutNS(t)}{dt} & Si \quad -Sr < \frac{dVoutNS(t)}{dt} < Sr \\ +Sr & Si \quad \frac{dVoutNS(t)}{dt} \geq Sr \end{cases}$$

Définition 18 Transitions entre modes de comportement vis à vis de la limitation de la vitesse de variation

Quand la vitesse de variation de la tension de sortie n'est pas limitée, la sortie peut évoluer vers le mode où la vitesse de variation est limité à la vitesse de balayage à la montée Sr ; dans le cas d'un accroissement de la tension de la sortie, ou à la descente $-Sr$, dans le cas d'une décroissance de la tension de la sortie (cf. fig. 44).

Une fois dans un mode avec limitation de la vitesse de variation de la tension de sortie, l'amplificateur ne peut quitter ce mode que si la tension de sortie rattrape la tension d'une sortie théorique à la vitesse de variation non limitée, c'est-à-dire quand $Vout$ rattrape $VoutNS$ (voir zone ② sur fig. 43). Puisque la vitesse de variation de $Vout$ est limitée à la vitesse de balayage de l'amplificateur, ceci ne peut se produire que si la vitesse de variation de $VoutNS$ redevient inférieure à la vitesse de balayage.

La sortie d'un mode avec limitation de la vitesse de variation se fait toujours vers le mode normal sans limitation de la vitesse de variation. Ceci permet d'éviter l'oscillation entre les modes avec limitation $MSrL$ et $MSrH$, cette oscillation n'étant qu'un artéfact de simulation indésirable. Cependant, lors d'un changement rapide de sens de variation, la sortie peut, immédiatement après le retour en mode normal, évoluer vers le mode de limitation opposé à celui qu'elle avait précédemment. De la sorte, la fonction $VoutSr$ décrit correctement la forme triangulaire que prend la tension de sortie avec limitation de la vitesse de variation quand l'entrée varie à une fréquence supérieure à la vitesse de balayage de l'amplificateur.(cf. zone ① sur fig. 43).

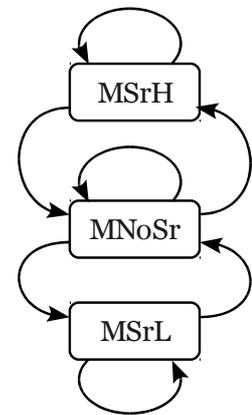


Fig. 44 : Transitions entre modes de limitation de la vitesse de variation

Définition 19 Tension de sortie avec produit gain-bande fini et vitesse de balayage finie

Lorsque la vitesse de variation de la tension de sortie est limitée à la vitesse de balayage de l'amplificateur, la tension de sortie varie selon une rampe. La fonction qui décrit cette tension est une droite de pente Sr quand la tension de sortie s'accroît et de pente $-Sr$ quand elle décroît. Ces portions de droite doivent être

raccordées à la courbe décrite par la fonction V_{outNS} qui décrit l'évolution de la tension de sortie quand la vitesse de variation n'est pas limitée. En appelant t_0 l'instant où la surveillance de la vitesse de variation détecte l'entrée dans un des modes de limitation de la vitesse de variation, et en appelant V_{t_0} la valeur de la tension de sortie à cet instant t_0 , la droite qui décrit alors la tension de sortie passe par le point $[t_0, V_{t_0}]$. La fonction V_{outSr} est donc définie comme suit :

$$\forall t \in \mathbb{R},$$

$$V_{outSr}(t) = \begin{cases} -Sr \cdot t + Sr \cdot t_0 + V_{t_0} & \text{Si } \frac{dV_{outNS}(t)}{dt} \leq -Sr \\ V_{outNS}(t) & \text{Si } -Sr < \frac{dV_{outNS}(t)}{dt} < Sr \\ Sr \cdot t - Sr \cdot t_0 + V_{t_0} & \text{Si } \frac{dV_{outNS}(t)}{dt} \geq Sr \end{cases}$$

Conditions aux limites lors du changement de mode de limitation et implémentation informatique

Les morceaux de la fonction décrivant la tension de sortie avec saturation sont raccordés. D'un point de vue purement mathématique, la définition 19 est suffisante pour garantir un raccordement correct. Toutefois sur le plan informatique, la simulation étant à pas discret – aussi fin soit-il – et la surveillance de la saturation étant effectuée à posteriori, la tension de sortie doit être corrigée lors d'un changement de mode. Au moment où il est constaté que la vitesse de variation de la tension de sortie dépasse la vitesse de balayage de l'amplificateur, la valeur de la tension de sortie excède la valeur qu'elle aurait dû avoir si la limitation de la vitesse de variation avait été respectée. La tension de sortie doit donc être corrigée en refaisant le calcul à partir de la valeur qu'elle avait avant le dépassement de la vitesse de balayage en lui ajoutant une variation égale au produit de la vitesse de balayage par la durée du pas de simulation. En appelant Δt le pas de temps de simulation, la tension de sortie, au moment où le dépassement de la vitesse de balayage est constaté, doit donc être forcée à la valeur $V_{t_0} = V_{outSr}(t_0) = V_{outSr}(t_0 - \Delta t) + Sr \cdot \Delta t$ quand la tension de sortie croit et à la valeur $V_{t_0} = V_{outSr}(t_0) = V_{outSr}(t_0 - \Delta t) - Sr \cdot \Delta t$ quand la tension de sortie décroît.

A l'instant t_1 où il est constaté que la vitesse de variation de la tension de sortie redevient inférieure à la vitesse de balayage de l'amplificateur, la tension de sortie à la vitesse de variation non limitée, V_{outNS} , peut être très éloignée de la tension de sortie avec limitation. Il est impératif que la tension de la sortie sans limitation soit ramenée à la valeur qu'a, à cet instant t_1 , la tension de sortie avec limitation. Il faut donc appliquer l'égalité $V_{outNS}(t_1) = V_{outSr}(t_1)$, avant le passage en mode normal sans limitation.

V.3.6) Modèle complet de l'amplificateur opérationnel simplifié

Le modèle complet de l'amplificateur opérationnel simplifié est construit à partir des fonctions V_{in} , V_{outNS} , V_{outSat} et V_{outSr} . Les fonctions V_{outSat} et V_{outSr} suffisent pour exprimer la fonction V_{out} décrivant la tension de sortie de l'amplificateur opérationnel simplifié avec produit gain-bande fini, saturations et vitesse de balayage finie. En effet, V_{outSat} et V_{outSr} utilisent V_{outNS} dans leur construction, tandis que V_{outNS} utilise la fonction V_{in} .

Définition 20 Tension de sortie de l'amplificateur opérationnel simplifié

Les démonstrations prouvant que le modèle d'amplificateur opérationnel simplifié vérifie les propriétés énoncées au chapitre V.3.3)ii sont données à l'annexe A 4.4).

$$\forall t \in \mathbb{R},$$

$$V_{out}(t) = \begin{cases} V_{outSatL} & \text{Si } (V_{outNS}(t) \leq V_{outSatL}) \\ (-Sr \cdot t + Sr \cdot t_0 + V_{t_0}) & \text{Si } (V_{outSatL} < V_{out}(t) < V_{outSatH}) \text{ et } \left(\frac{dV_{outNS}(t)}{dt} \leq -Sr \right) \\ V_{outNS}(t) & \text{Si } (V_{outSatL} < V_{out}(t) < V_{outSatH}) \text{ et } \left(-Sr < \frac{dV_{outNS}(t)}{dt} < Sr \right) \\ (Sr \cdot t - Sr \cdot t_0 + V_{t_0}) & \text{Si } (V_{outSatL} < V_{out}(t) < V_{outSatH}) \text{ et } \left(\frac{dV_{outNS}(t)}{dt} \geq Sr \right) \\ V_{outSatH} & \text{Si } (V_{outNS}(t) \geq V_{outSatH}) \end{cases}$$

Quand la sortie sature, la tension est constante, sa vitesse de variation est donc nulle. Par conséquent, la surveillance de la vitesse de variation n'a pas lieu et la variable d'état $SrMode$ est forcée alors à $MNoSr$.

Définition 21 Transitions supplémentaires dues à la combinaison de la saturation et de la limitation de la vitesse de variation

La combinaison de la saturation et de la limitation de la vitesse de variation, tout en conservant les changements de modes qui ont été énoncés à la définition 15 et à la définition 18, ajoute les changements de modes suivants.

- Quand la tension de sortie croît avec une vitesse de variation limitée par la vitesse de balayage à la montée Sr , l'amplificateur peut évoluer vers le mode de saturation par rapport au seuil supérieur $VoSatH$.
- Quand la tension de sortie décroît avec une vitesse de variation limitée par la vitesse de balayage à la descente $-Sr$, l'amplificateur peut évoluer vers le mode de saturation par rapport au seuil inférieur $VoSatL$.

La figure 45 illustre tous les changements de modes possibles suite à la combinaison de la saturation et de la limitation de la vitesse de variation.

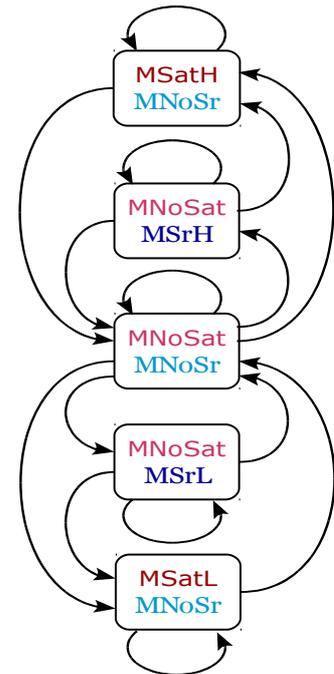


Fig. 45 : Transitions entre tous les modes du modèle complet de l'amplificateur opérationnel simplifié

V.3.7) Résultats obtenus avec l'implémentation en SystemC-AMS

Le modèle de l'amplificateur opérationnel simplifié, qui a été élaboré au long des paragraphes V.3.1 à V.3.6, a été implémenté en SystemC-AMS avec le modèle de calcul ELN. La saturation des entrées n'a cependant pas été mise en œuvre dans cette implémentation. Le code source correspondant est disponible sur le site internet du laboratoire TIMA. Il peut être téléchargé librement depuis la page « Autres productions » à l'URL :

<http://tima.imag.fr/tima/fr/mediatheque/productions.html>

Ce paragraphe présente les résultats des tests menés avec le modèle de l'amplificateur opérationnel simplifié, d'abord dans un montage sans rétroaction puis dans un montage avec rétroaction négative (contre-réaction) et rétroaction positive. Afin de démontrer que l'implémentation vérifie bien les propriétés énoncées au §V.3.3)ii, l'atténuation et le déphasage en fréquence, la saturation et la limitation du taux de variations ont été évalués au cours des deux campagnes de tests. Les tests avec rétroaction prouvent la capacité du modèle d'amplificateur opérationnel simplifié à correctement différencier un amplificateur monté en contre-réaction d'un amplificateur monté en rétroaction positive.

Toutes les courbes présentées dans le présent sous-chapitre sont des courbes de simulation obtenues avec l'implémentation en SystemC-AMS du modèle de l'amplificateur opérationnel simplifié.

i) Tests sans rétroaction

Les tests sans rétroaction sont effectués avec le circuit de la figure 46. Pour ces tests les caractéristiques suivantes sont appliquées au modèle (valeurs issues de la fiche technique de l'AD8541) :

- $GBP = 1$ MHz
- $A = 40000$ V/V
- $f_c = 25$ Hz
- $Sr = 0,92$ V/ μ s

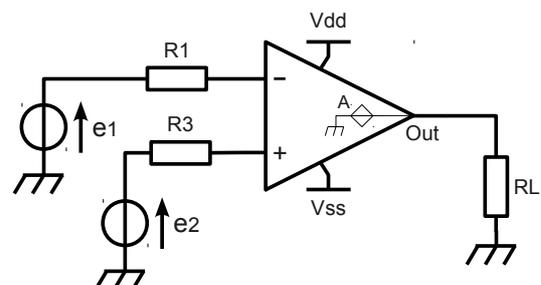


Fig. 46 : Circuit de test sans boucle de rétroaction (MoC ELN)

Amplification différentielle et réjection du mode commun

Les courbes de la figure 47 montre l’amplification différentielle et la réjection du mode commun. Dans la zone ①, alors que les tensions sur les deux entrées sont progressivement décalées de la même quantité, la tension de sortie V_{out} ne se décale pas. Le mode commun est donc totalement rejeté. En zone ②, quand les deux entrées sont pilotées par des sinusoïdes de même fréquence, même phase, même amplitude et même tension de décalage, alors la sortie prend sa valeur de repos, soit zéro volt dans le cas d’une alimentation symétrique et la moitié de la tension d’alimentation dans le cas d’une alimentation monotension. En zone ③, alors que les entrées sont pilotées par des sinusoïdes de même fréquence, même amplitude et même tension de décalage mais de phase opposée, la tension de sortie est d’amplitude double par rapport à la zone ① où seule l’entrée moins de l’amplificateur était attaquée par une sinusoïde, le générateur e_2 imposant une tension continue sur l’entrée plus. L’amplification est donc purement différentielle. Les courbes obtenues avec l’alimentation monotension sont superposables avec celles obtenues avec une alimentation symétrique de même amplitude. Elles sont cependant centrées autour de la moitié de la tension d’alimentation dans le cas d’une alimentation monotension alors qu’elles sont centrées autour de zéro dans le cas d’une alimentation symétrique. Ceci montre la capacité du modèle à prendre en compte la tension d’alimentation.

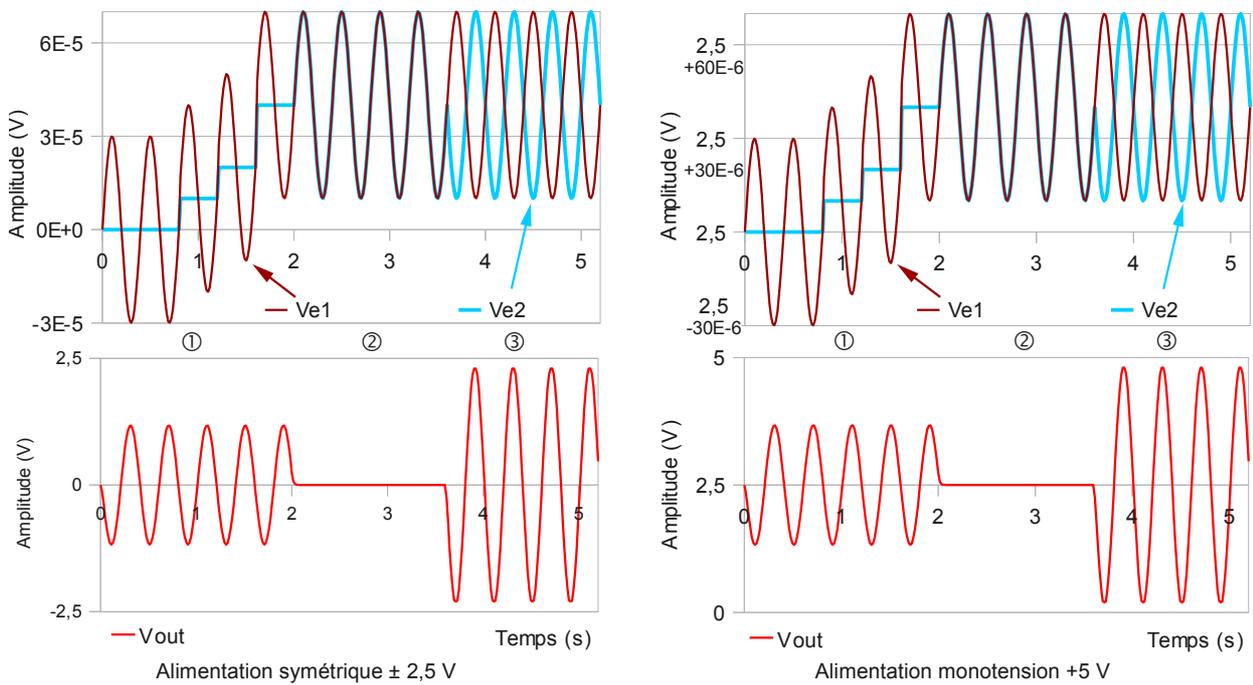


Fig. 47 : Amplification différentielle et réjection du mode commun

Gain et déphasage en fréquence

Sur la figure 48, la courbe verte claire marquée $A \cdot V_{in}$ représente le produit de la tension différentielle d’entrée par le gain intrinsèque de l’amplificateur. C’est la tension qu’il y aurait en sortie sans atténuation ni déphasage en fréquence. Il est à noter que c’est la tension de sortie produite par le modèle de l’amplificateur idéal. Cette tension permet de montrer la capacité du modèle de l’amplificateur opérationnel simplifié à rendre correctement compte de l’atténuation et du déphasage en fréquence (courbe rouge V_{out}). À la fréquence de coupure f_c , l’atténuation est de 3 dB ($A \cdot V_{in} = 2 \text{ V}$; $V_{out} = 1,4 \text{ V}$) et le déphasage est de 45° , ce qui est conforme à ce qui est attendu.

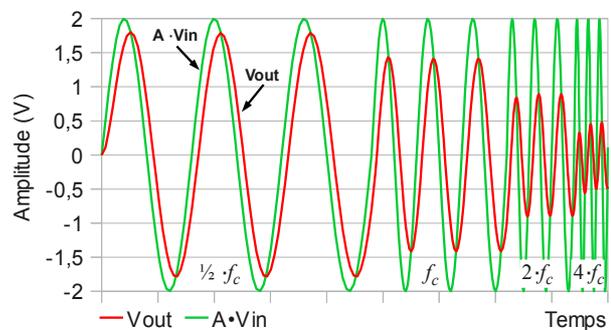


Fig. 48 : Atténuation en fréquence

Les figures 49 et 50 donnent respectivement le gain et le déphasage dans le plan de Bode. Ces courbes ont été construites par points relevés dans le domaine temporel grâce à une volubulation*. Elles sont conformes à ce qui est attendu.

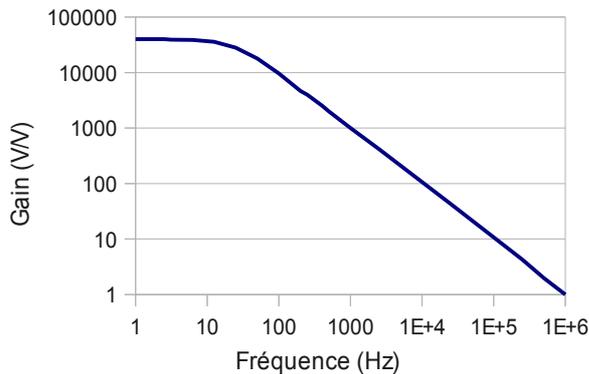


Fig. 49 : Gain en fréquence en absence de boucle de rétroaction

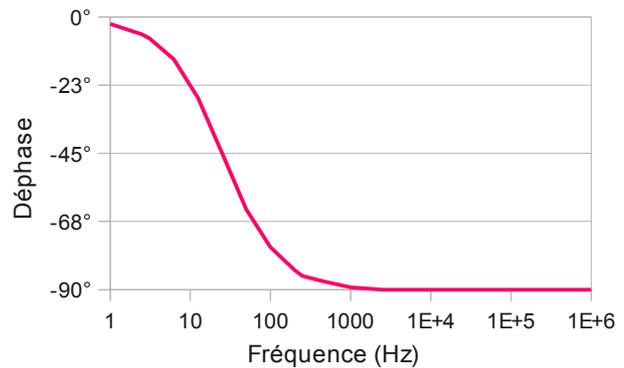


Fig. 50 : Déphasage en fréquence en absence de boucle de rétroaction

Saturation

La saturation de la tension de sortie est illustrée par la figure 51. Au départ, la tension sortie ne sature pas et est centrée autour de la tension de repos qui est à mi-chemin entre les tensions d'alimentation. Dans cet exemple, l'alimentation de l'amplificateur est monotension et vaut +5 V. La tension de repos vaut donc +2,5 V. Les points ① à ④ montrent que le décalage progressif de la tension d'entrée conduit à l'écrêtage progressif de la tension de sortie jusqu'à la rendre constante et égale à la valeur du seuil de saturation (point ④). Il est à remarquer que le seuil de saturation est écarté de la tension d'alimentation de la valeur de la tension de déchet $VoLoss$. Ce qui est observé, points ① à ④, par rapport au seuil de saturation supérieur est également observé, points ⑥ à ⑧, par rapport au seuil de saturation inférieur. Le passage du ④ au ⑤ montre qu'alors que la tension d'entrée est suffisamment décalée pour provoquer la saturation complète de la sortie, une augmentation suffisante de l'amplitude du signal d'entrée peut amener les sommets du signal amplifié à repasser sous le seuil de saturation. Ces sommets sont alors correctement reproduits. Au point ⑨, alors que la tension de sortie est centrée autour de la tension de repos, l'amplitude du signal d'entrée est telle que la sortie sature à la fois par rapport au seuil de saturation supérieur et au seuil de saturation inférieur. Les points ⑥ et ⑩ montrent que la sortie sort correctement d'une situation de saturation.

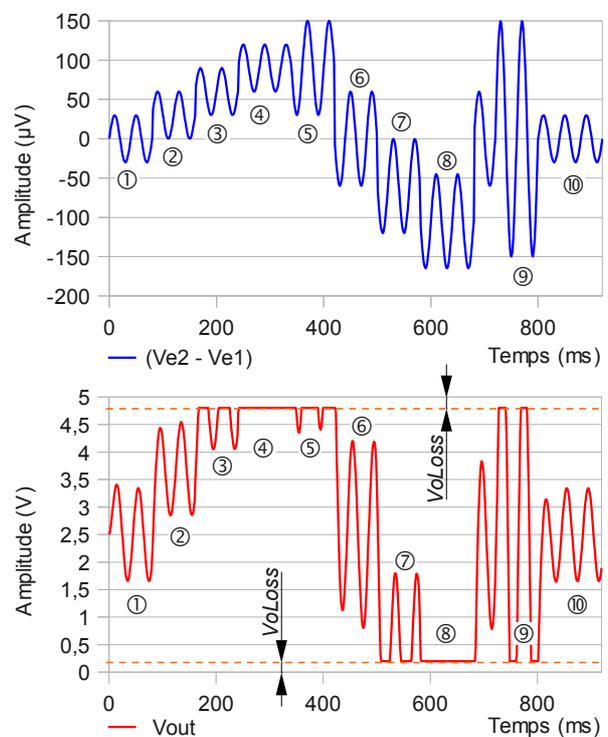


Fig. 51 : Saturation

Comparateur

Les tests précédents de saturation valident également le fonctionnement en comparateur.

Avec les valeurs numériques choisies ici, un signal d'entrée d'amplitude crête supérieure à $\frac{(2,5 \text{ V} - 0,2 \text{ V})}{40000} = 57,5 \mu\text{V}$ provoque la saturation de la sortie. Il s'en suit que lorsque la tension sur l'entrée plus dépasse celle sur l'entrée moins d'au moins $57,5 \mu\text{V}$, la tension de sortie égale la tension du

* Voir glossaire

seuil de saturation supérieur. Symétriquement, quand la tension sur l'entrée plus est inférieure d'au moins $57,5 \mu\text{V}$ à celle sur l'entrée moins, la tension de sortie égale la tension du seuil de saturation inférieur. C'est ce qui est observé sur la figure 51 aux points ③ à ⑤ et ⑦ à ⑨.

Limitation de la vitesse de variation

La figure 52 montre un signal d'entrée dont la vitesse de variation change à cause d'un changement d'amplitude. L'observation de la sortie pour ce signal d'entrée met en évidence que la vitesse de variation de la tension de sortie est maintenue égale à la vitesse de balayage de l'amplificateur chaque fois qu'elle aurait dû devenir supérieure à la vitesse de balayage, s'il n'y avait pas eu de limitation. La vitesse de variation étant constante, les changements d'amplitude s'effectuent en rampe. Le signal de sortie se déforme alors pour devenir une triangulaire bien que l'entrée soit une sinusoïde. Lorsque la vitesse de variation de la tension de sortie redevient inférieure à la vitesse de balayage de l'amplificateur, la forme du signal est restaurée. Ceci est conforme à ce qui est observé avec un amplificateur réel.

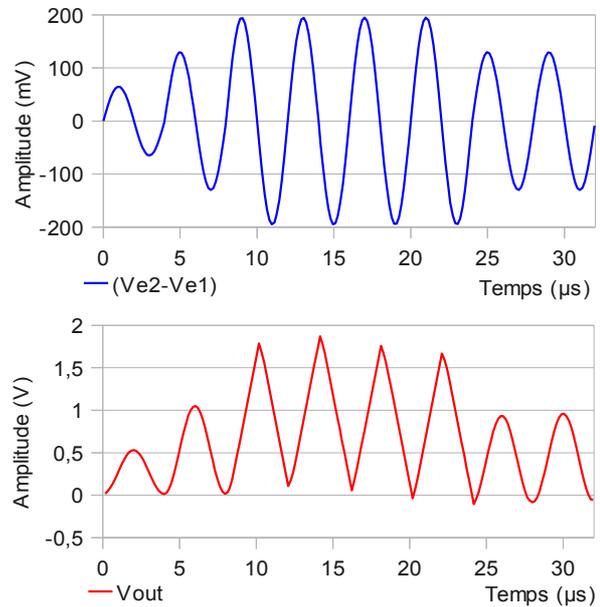


Fig. 52 : Limitation de la vitesse de variation de la sortie à la vitesse de balayage de l'amplificateur

Courant

Avoir implémenté le modèle de l'amplificateur opérationnel simplifié à l'aide du modèle de calcul ELN de SystemC-AMS permet de rendre compte de la consommation électrique de l'amplificateur. Le courant débité par la sortie de l'amplificateur provient de ses plots d'alimentation. Pour le modèle de l'amplificateur opérationnel simplifié, quand la sortie débite un courant positif (courant sortant), celui-ci est intégralement prélevé sur le rail d'alimentation positif et quand elle débite un courant négatif (courant entrant), celui-ci est intégralement prélevé sur le rail d'alimentation négatif. A ces courants, s'ajoute le courant de repos de l'amplificateur. Ce dernier s'écoule en permanence du plot d'alimentation positif vers le plot d'alimentation négatif. Il correspond à la consommation propre de l'amplificateur.

Pour les tests en courant, les caractéristiques suivantes sont appliquées au modèle :

- courant de repos de l'amplificateur, $I_q = 50 \mu\text{A}$
- courant de court-circuit (destructeur), $I_{sc} = 60 \text{ mA}$

Si au cours du fonctionnement le courant débité par la sortie est amené à dépasser le courant de court-circuit, il y a alors un fort risque de destruction de l'amplificateur. C'est pourquoi si une telle situation est rencontrée en cours de simulation, la simulation est arrêtée et un message d'alerte est affiché.

La figure 53 décrit le circuit de test qui a été utilisé pour évaluer le courant consommé par l'amplificateur et pour dresser les courbes de la figure 54. Le courant traversant la résistance R_{ic} symbolise la consommation du reste du circuit. Dans cet exemple, il est supposé que l'activité du reste du circuit demeure inchangée et donc que le courant I_{ic} est constant. Il apparaît clairement sur les courbes de la figure 54 que l'amplificateur consomme même au repos quand la sortie ne débite aucun courant. A chaque

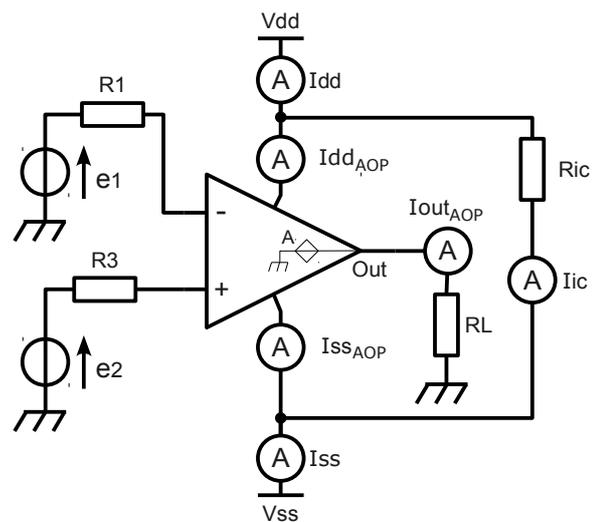


Fig. 53 : Circuit de test pour l'évaluation du courant

instant, le courant consommé par l'amplificateur correspond au courant de sortie auquel s'ajoute le courant de repos. Quand le courant de sortie est positif, l'alimentation positive est mise à contribution et seul le courant de repos traverse le plot d'alimentation négative ($I_{ss_{AOP}} = I_q$). Symétriquement quand le courant de sortie est négatif, l'alimentation négative est mise à contribution et seul le courant de repos traverse sur le plot d'alimentation positive ($I_{dd_{AOP}} = I_q$). Les courants I_{dd} et I_{ss} débités par les sources de tension V_{dd} et V_{ss} sont constitués respectivement par les courants $I_{dd_{AOP}}$ et $I_{ss_{AOP}}$, absorbés par l'amplificateur sur ses plots d'alimentation positive et négative, auxquels s'ajoutent le courant absorbé par le reste du circuit.

$$I_{dd} = I_{dd_{AOP}} + I_{ic}$$

$$I_{ss} = I_{ss_{AOP}} + I_{ic}$$

$$I_{dd_{AOP}} = \begin{cases} I_{out_{AOP}} + I_q & \text{Si } I_{out_{AOP}} > 0 \\ I_q & \text{Si } I_{out_{AOP}} \leq 0 \end{cases}$$

$$I_{ss_{AOP}} = \begin{cases} I_{out_{AOP}} - I_q & \text{Si } I_{out_{AOP}} < 0 \\ -I_q & \text{Si } I_{out_{AOP}} \geq 0 \end{cases}$$

ii) Test avec boucle de rétroaction

Ce paragraphe s'attache d'abord à démontrer la capacité du modèle de l'amplificateur opérationnel simplifié à distinguer correctement un montage en contre-réaction d'un montage avec rétroaction positive. Ensuite, il se portera sur des tests spécifiques aux montages avec rétroaction. Il ne reprend donc pas tous les tests effectués pour le montage sans rétroaction.

Différenciation entre contre-réaction et rétroaction positive

Le circuit de la figure 55 est utilisé pour évaluer la capacité du modèle de l'amplificateur opérationnel simplifié à différencier un amplificateur en contre-réaction d'un amplificateur en rétroaction positive. Dans ce circuit, les deux amplificateurs ont exactement les mêmes caractéristiques. Les boucles de rétroaction sont réalisées avec des résistances de même valeur. Les mêmes signaux attaquent les entrées des amplificateurs. La seule différence est l'inversion du branchement des entrées entre l'amplificateur en contre-réaction (marqué AOP1 sur fig. 55) et celui en rétroaction positive (marqué AOP2 sur fig. 55).

La figure 56 montre les résultats obtenus avec les mêmes signaux d'entrée que ceux appliqués lors du test sur la réjection du mode commun et l'amplification différentielle (cf. fig. 47). Dès le démarrage, la sortie de AOP2 dérive vers le seuil de saturation supérieur. Cette dérive est une croissance exponentielle comme le fait apparaître le zoom n°1 reporté sur la figure 57. Ce comportement correspond à une réponse instable et apériodique comme cela avait été prédit lors de l'étude de la fonc-

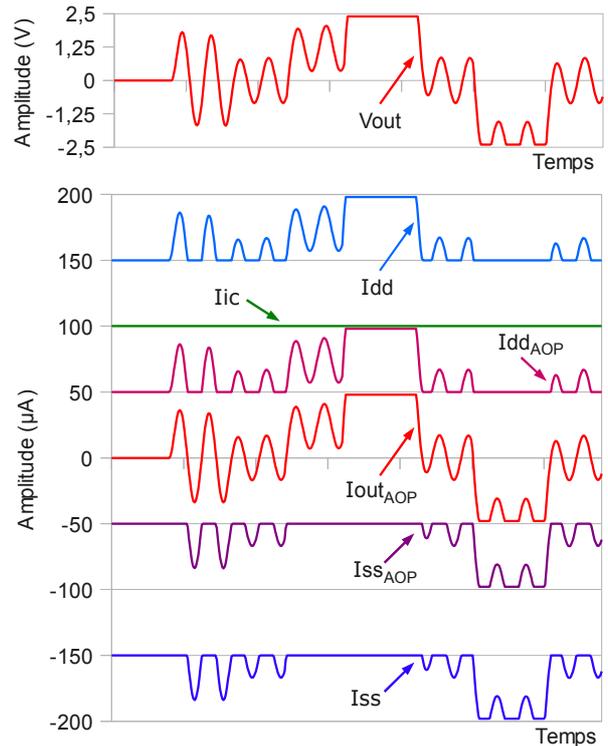


Fig. 54 : Courant à travers l'amplificateur

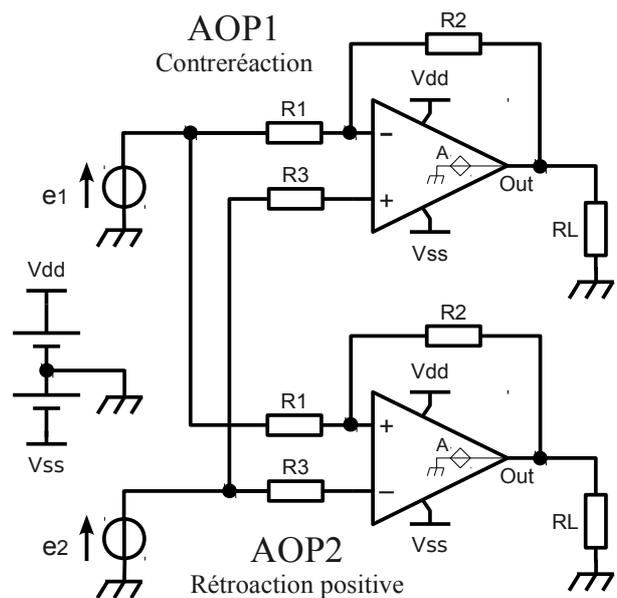


Fig. 55 : Circuit de test avec boucles de rétroaction (MoC ELN)

tion de transfert du montage en rétroaction positive (cf. V.3.4)iv, p. 71). La courbe de la sortie de AOP1 sur la figure 56 montre que celui-ci amplifie bien la différence entre les tensions d'entrée du montage en contre-réaction. Dans cette exemple, les résistances de la boucle de rétroaction ont été choisi de façon à avoir un rapport de 20, c'est-à-dire $\frac{R2}{R1} = 20$. Les courbes de la figure 56 permettent de constater que c'est aussi le rapport entre la tension de sortie de AOP1 et la différence des tensions d'entrée. Le gain du montage en contre-réaction est bien égal au rapport des résistances de la boucle comme cela était attendu.

Il peut paraître surprenant au premier abord de ne pas avoir une tension en sortie de AOP1 qui soit nulle quand la différence de tension est nulle entre les entrées du montages e_1 et e_2 (cf. point ② sur fig. 56). Ceci est dû à la contre-réaction et provient du fait que les entrées ne sont pas nulles à ce moment là (cf. point ② sur fig. 47, p. 78). Compte tenu du circuit de test utilisé ici, la tension différentielle d'entrée de l'amplificateur en contre-réaction n'est pas nulle quand la différence de tension entre les entrées du montage est nulle. L'explication est donnée en reprenant l'équation (19) qui exprime, dans le domaine de Laplace, la tension différentielle d'entrée du montage en contre-réaction (cf. V.3.4)iii, p. 69).

$$\forall p \in \mathbb{C},$$

$$V_{SIG}(p) = \frac{(R1 + R2)}{R1} \cdot E_2(p) - \frac{R2}{R1} \cdot E_1(p) \quad (19)$$

L'équation temporelle de la tension différentielle d'entrée du montage en contre-réaction est extraite de (19) par transformée de Laplace inverse.

$$\forall t \in \mathbb{R},$$

$$V_{sig}(t) = \frac{(R1 + R2)}{R1} \cdot e_2(t) - \frac{R2}{R1} \cdot e_1(t) \quad (30)$$

$$\Leftrightarrow V_{sig}(t) = e_2(t) + \frac{R2}{R1} \cdot [e_2(t) - e_1(t)] \quad (31)$$

Quand la différence de tension entre les entrées est nulle, la tension différentielle d'entrée est égale à la tension délivrée par la source e_2 . Or, e_2 est connectée sur l'entrée moins de AOP1. La contre-réaction gommant le gain intrinsèque de l'amplificateur, il s'en suit que la tension de sortie est elle-même égale à e_2 . Ceci s'observe sur la figure 58 où le signal délivré par e_2 a été superposé sur le zoom n°2 de la figure 56.

$$\forall t \in \mathbb{R},$$

$$e_1(t) = e_2(t)$$

$$\Rightarrow V_{sig}(t) = e_2(t) \Rightarrow V_{out_AOP1}(t) = e_2(t)$$

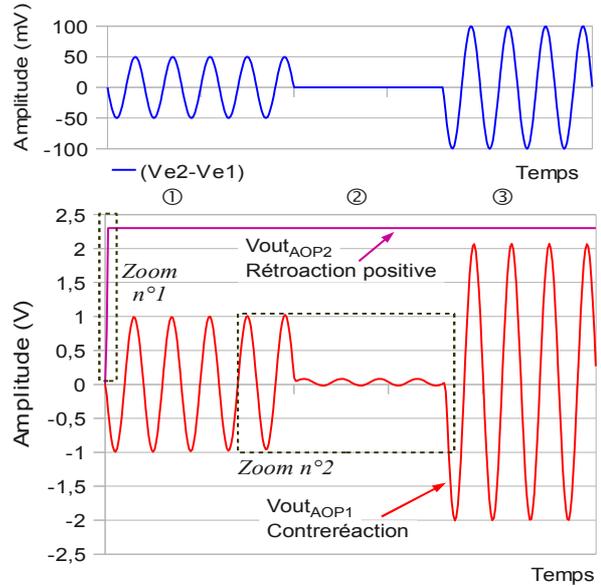


Fig. 56 : Différenciation entre rétroactions positive et négative

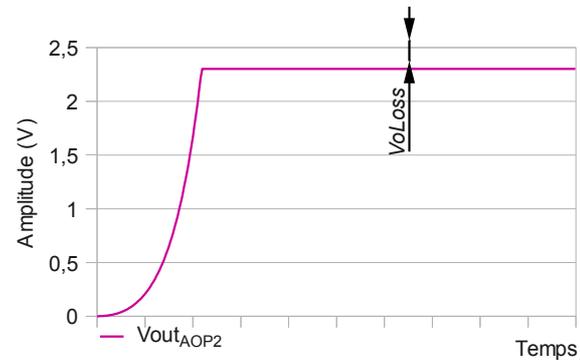


Fig. 57 : Dérive exponentielle due à la rétroaction positive

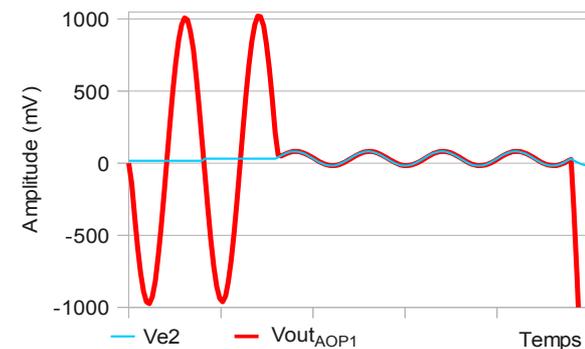


Fig. 58 : Tension de sortie de l'amplificateur en contre-réaction égale à la tension e_2 (entrée "+") quand les signaux d'entrée sont identiques

Atténuation et déphasage en fréquence (montage en contre réaction)

Les courbes de la figure Fig. 59 donnent respectivement le gain et le déphasage dans le plan de Bode du montage en contre réaction comparés au gain et déphasage du circuit sans boucle de rétroaction (cf. fig. 49 et fig. 50). Ces courbes ont été construites par points relevés dans le domaine temporel grâce à une volubulation. Conformément à ce qui est attendu, elles montrent que la contre réaction accroît la bande passante au détriment du gain. Ces relevés ont été réalisés avec un montage inverseur de gain de 20.

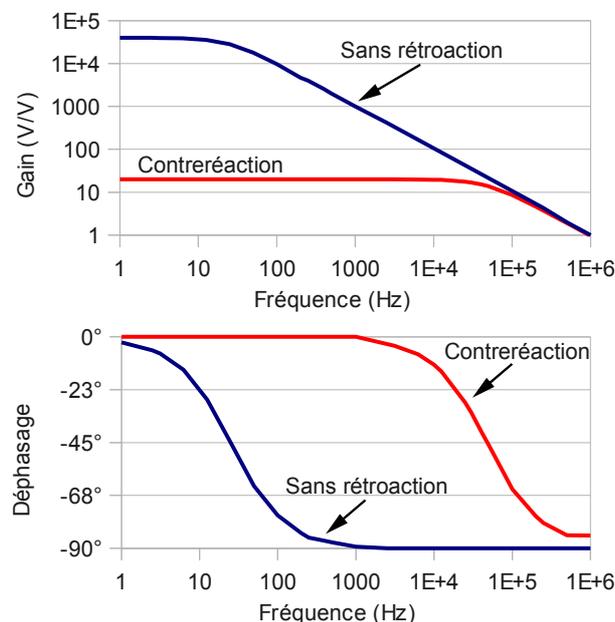


Fig. 59 : Gain et déphasage en fréquence avec et sans rétroaction

V.3.8) Contraintes d'utilisation du modèle d'amplificateur opérationnel simplifié

Les simulateurs numériques de part leur constitution, et aussi parce qu'ils s'exécutent sur des ordinateurs qui sont eux-mêmes des machines numériques¹⁶, sont nécessairement à pas discret. Ceci signifie que les variables évoluent par sauts de valeur et que, même dans le cas de la simulation d'une fonction continue, le simulateur produit une version discrétisée de l'image de la fonction simulée.

Une des techniques pour reconstruire la courbe de la fonction à partir des points calculés par le simulateur consiste à relier deux points consécutifs par un segment de droite (interpolation linéaire d'ordre 1). C'est une technique très répandue car elle est rapide et facile à mettre en œuvre, ne nécessitant pas de calcul supplémentaire. L'inconvénient est que pour obtenir une reconstruction convenable, le nombre de points à calculer doit être bien au-delà de ce qu'exige le strict respect du théorème de l'échantillonnage de Shannon. La figure 60 montre qu'une sinusoïde est reconstituée de façon satisfaisante avec 25 segments de droite par période (nombre d'intervalles déterminé empiriquement), là où le théorème de Shannon n'exige que deux points par période. En première approche, le pas de simulation est donc réglé à $1/25$ e de la période du signal le plus rapide à examiner.

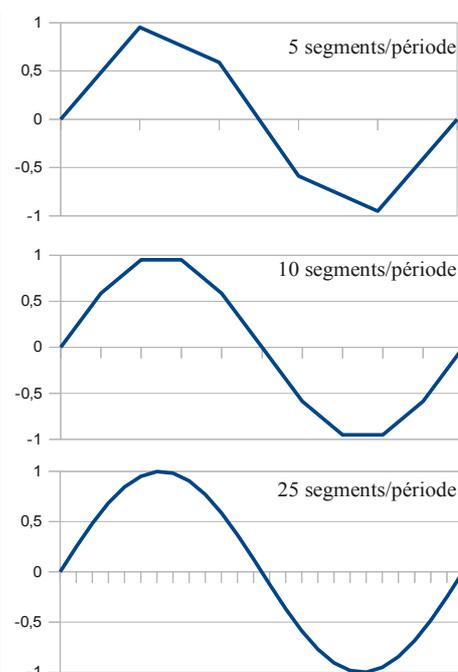


Fig. 60 : Reconstruction d'une sinusoïde en fonction du nombre de segments par période

¹⁶ Le succès des microprocesseurs numériques a fait disparaître les calculateurs analogiques.

L'utilisation d'un pas de simulation discret à aussi une répercussion sur le calcul des dérivées. Aussi petit soit le pas de simulation, il n'est pas infiniment petit au sens mathématique du terme. Un simulateur numérique est donc incapable de calculer exactement une dérivée. Il ne peut en fournir qu'une approximation (par exemple, à l'aide d'une des techniques aux différences finies). La valeur du pas de temps de simulation a une grande influence sur la précision de calcul et par suite sur la qualité des résultats produits par le modèle.

Alors que le modèle de l'amplificateur idéal utilise une équation algébrique pour le calcul de la tension de sortie (cf. (2)), le modèle de l'amplificateur opérationnel simplifié utilise une équation différentielle (cf. (13)) pour calculer la tension de sortie en absence de saturation et de limitation de la vitesse de variation. L'élément clé qui distingue donc ces deux modèles est la dérivée de la tension de sortie utilisée dans le modèle de l'amplificateur opérationnel simplifié. La conséquence est que le modèle de l'amplificateur opérationnel simplifié est sensible à l'estimation de cette dérivée que fournit le simulateur numérique. Si la dérivée de la tension de sortie est sous évaluée, au point de devenir négligeable devant la valeur de la tension de sortie, alors le modèle de l'amplificateur opérationnel simplifié se rapproche du modèle de l'amplificateur idéal et comporte alors les mêmes limitations et insuffisances que celui-ci (cf. § V.1.1).

La méthode d'Euler explicite est la technique d'évaluation de la dérivée qui a été retenue par l'équipe du Fraunhofer pour réaliser la preuve de concept (*Proof of Concept – PoF*), seule implémentation de SystemC-AMS existant au moment où est écrit ce mémoire. SystemC-AMS, comme SystemC, a été conçu pour simuler des signaux fonction du temps. Dans ce contexte, le pas de simulation est un pas de temps. Dans la méthode d'Euler, la dérivée est approchée en divisant la différence entre la valeur obtenue au cycle de simulation courant et la valeur obtenue au cycle de simulation précédent, par la durée du pas de temps du simulateur. En appelant Δt le pas de temps du simulateur et $VoutNS[t]$ la fonction discrétisée, selon le pas de simulation Δt , de la fonction $VoutNS(t)$ ¹⁷ qui sert à calculer la tension de sortie en absence de saturation et de limitation de la vitesse de variation, la valeur approchée de la dérivée $\frac{\Delta VoutNS[t]}{\Delta t}$ s'obtient de la façon suivante par la méthode d'Euler explicite :

$$\begin{aligned} \Delta t &= [t] - [t-1] \\ VoutNS[t] &= VoutNS(t) \\ VoutNS[t-1] &= VoutNS(t - \Delta t) \\ \frac{dVoutNS(t)}{dt} &\approx \frac{\Delta VoutNS[t]}{\Delta t} = \frac{VoutNS[t] - VoutNS[t-1]}{\Delta t} \end{aligned}$$

L'équation différentielle calculée par le simulateur n'est pas l'équation (13) mais plutôt l'équation suivante :

$$\begin{aligned} \forall t \in \mathbb{R}, \\ \frac{\tau \cdot \Delta VoutNS[t]}{\Delta t} + VoutNS[t] &= A \cdot VinNS[t] + Vbias \\ \Leftrightarrow VoutNS[t] &= A \cdot VinNS[t] - \frac{\tau}{\Delta t} \cdot \Delta VoutNS[t] + Vbias \end{aligned}$$

Pour un fonctionnement correct la contribution de la tension différentielle d'entrée $VinNS$ et celle de la variation de la tension de sortie $\Delta VoutNS$ doivent être équilibrées. Leurs pondérations respectives, A et $\frac{\tau}{\Delta t}$ doivent donc être identiques.

$$\begin{aligned} A &= \frac{\tau}{\Delta t} \\ \Leftrightarrow \Delta t &= \frac{\tau}{A} \end{aligned} \tag{32}$$

En remplaçant τ par son expression en fonction du produit gain-bande (cf. (10)), on obtient :

$$\Delta t = \frac{1}{2\pi \cdot GBP} \tag{33}$$

Toutefois, si sur l'ensemble du montage, les signaux les plus rapides ont une période inférieure à 25 fois Δt , alors il convient de fixer le pas de simulation à une durée égale à 1/25e de la période du signal le plus

¹⁷ Remarquer l'emploi des crochets à la place des parenthèses pour noter la fonction échantillonnée

rapide afin de respecter les 25 segments par périodes identifiés fig. 60. Ceci conduit à la démarche suivante pour régler le pas de simulation.

Définition 22 Valeurs du pas de simulation à respecter pour le bon fonctionnement du modèle de l'amplificateur opérationnel simplifié

Avec Sr la vitesse de balayage de l'amplificateur, GBP le produit gain-bande, $Tmin$ la période du signal le plus rapide appliqué au montage et $NbSeg$ le nombre de segments par période du signal le plus rapide, pour que le bon fonctionnement du modèle d'amplificateur opérationnel simplifié soit assuré, le pas de simulation Δt doit être choisi de la sorte :

$$\Delta t = \begin{cases} \frac{1}{2\pi \cdot GBP} & \text{Si } Tmin \geq \frac{NbSeg}{2\pi \cdot GBP} \\ \frac{Tmin}{NbSeg} & \text{Si } Tmin < \frac{NbSeg}{2\pi \cdot GBP} \end{cases} \quad \begin{array}{l} \text{Valeur recommandée} \\ NbSeg = 25 \text{ segments/période} \end{array}$$

La valeur recommandée pour le nombre de segments $NbSeg$ par période du signal le plus rapide est empirique. Elle est donc sujette à optimisation.

La vitesse de variation de la tension de sortie $Vout$ ne peut, par définition, dépasser la vitesse de balayage de l'amplificateur. La plus grande variation de cette tension entre deux pas de simulation consécutifs est donc égale au produit de la vitesse de balayage par la durée du pas de simulation.

$$\begin{aligned} \left| \frac{\Delta Vout[t]}{\Delta t} \right| &\leq Sr \\ \Rightarrow |\Delta Vout[t]| &\leq Sr \cdot \Delta t \end{aligned} \quad (34)$$

La variation de la tension de sortie entre deux pas de simulation consécutifs est obtenue avec la définition 22 :

$$\begin{cases} |\Delta Vout[t]| \leq \frac{Sr}{2\pi \cdot GBP} & \text{Si } Tmin \geq \frac{NbSeg}{2\pi \cdot GBP} \\ |\Delta Vout[t]| \leq \frac{Sr \cdot Tmin}{NbSeg} & \text{Si } Tmin < \frac{NbSeg}{2\pi \cdot GBP} \end{cases} \quad (35)$$

La conséquence de la nécessité de fixer le pas de simulation en fonction du produit gain-bande de l'amplificateur et non-plus seulement en fonction de la fréquence des signaux appliqués au montage, est que le nombre de points à calculer peut devenir très grand quand ces signaux ont des périodes longues devant le pas de simulation. Il s'en suit un accroissement du temps de simulation comme l'illustrent les résultats rassemblés dans le tableau 10. Ces résultats ont été produits avec le circuit de test avec boucle de rétroaction de la figure 55 pour lequel les paramètres ont été réglés de la façon suivante.

- $e1$ sinusoïde d'amplitude 0,5 V crête dont la fréquence est changée au cours des tests
- $e2 = 0$ V
- Alimentations : $Vdd = 5V$, $Vss = 0V$
- Rapport des résistances de boucle $\frac{R2}{R1} = 4$; la tension de sortie de l'amplificateur en contre-réaction $AOP1$ est par conséquent une sinusoïde d'amplitude 2 V crête
- Données enregistrées dans le fichier de traces : les tensions de $e1$ et $e2$, leur différence et les tensions de sortie des deux amplificateurs

La fréquence à partir de laquelle la variation de la tension de sortie est limitée à la vitesse de balayage est indiquée dans la colonne F_{maxSr} . La fréquence du signal d'entrée est maintenue inférieure à cette valeur.

Les essais ont été effectués sur un ordinateur type PC sous Linux avec les caractéristiques suivantes :

- processeur Intel® Core™ i3-2120, 2 cœurs physiques cadencés à 3.30 GHz, 4 CPU, cache 3 Mo
- RAM 4 Go, DDR3 cadencée à 1333 MHz

- fichier d'échange mémoire de 4 Go
- disque dur SATA 3.0, 7200 tr/min, cache 16 Mo
- système d'exploitation Debian GNU/Linux 6.0 *Squeeze*, noyau 2.6.32-5-amd64, avec interface graphique GNOME 2.30.2

La modélisation du circuit de la figure 55 comporte 28 équations au total. Un amplificateur contribue pour 7 équations dont une seule est différentielle. Chaque essai simule dix périodes du signal d'entrée eI . Les durées relevées ont été mesurées à l'aide de la fonction `clock()` de la bibliothèque C pour Linux. Elles incluent le temps d'élaboration du simulateur SystemC-AMS et elles ont été moyennées sur 10 simulations.

Exemple d'amplificateur opérationnel	F_{maxSr} (2V crête)	Pas de simulation Δt	Fréquence f_e du signal d'entrée (0,5 V crête)	Ratio $\frac{GBP}{f_e}$	Nombre total de points calculés	Temps de simulation pour 10 périodes
AD8541 $GBP = 1$ MHz $Sr = 0,92$ V/ μ s $Iq = 45$ μ A	73 kHz	159 ns	43 kHz	23,3	1463 pts	12 ms
			300 Hz	3333	209644 pts	1,8 s
			30 Hz	33333	2096437 pts	19,5 s
AD8500 $GBP = 7$ kHz $Sr = 4$ V/ms $Iq = 0,75$ μ A	318 Hz	22,7 μ s	300 Hz	23,3	1467 pts	10 ms
			30 Hz	233	14661 pts	120 ms
			0,21 Hz	33333	2094434 pts	19,1 s

Tableau 10 : Lien entre produit gain-bande, nombre de points calculés et temps de simulation

Plus le ratio entre le produit gain-bande et la fréquence du signal d'entrée est grand, plus le nombre de points calculés est grand et plus la simulation est longue. Un même ratio aboutit à un même temps de simulation. Par conséquent à fréquence identique, la simulation sera plus rapide avec un amplificateur ayant un produit gain-bande bas qu'avec un produit gain-bande élevé. Il est à noter que cette amélioration de performance de simulation trouve écho dans la performance d'un circuit réel. En effet, pour avoir un produit gain-bande plus élevé, il est nécessaire de d'avoir des sources de courant d'intensité plus élevée ce qui se traduit par une consommation statique plus importante. Ceci peut être constaté avec les deux amplificateurs opérationnels faible consommation pris en exemple, l'AD8541 et l'AD8500. Sous 5V, l'AD8541 possède un produit gain-bande cent quarante trois fois plus élevé que celui de l'AD8500 mais affiche également une consommation statique (Iq) soixante fois plus grande que celle de l'AD8500. Utiliser un amplificateur avec un produit gain-bande élevé par rapport à la fréquence des signaux à traiter n'est pas optimum en terme de consommation ni en terme de surface puisqu'une source de courant plus intense implique la mise en œuvre de transistors plus grands voire de davantage de transistors. Il n'est cependant pas toujours possible d'éviter une telle situation soit parce que la plage de fréquences des signaux appliqués sur l'ensemble du montage est très étalée, soit parce que le circuit est construit à partir d'un assemblage de blocs matériels propriétaires (*Intellectual Property – IP*). Dans le cas, d'une large plage de fréquences, le produit gain-bande doit-être suffisamment élevé pour pouvoir traiter sans atténuation la plus haute des fréquences utiles alors que des signaux beaucoup plus lents doivent être considérés pour le même montage. L'intérêt des *IP* réside dans la réutilisation sans modification d'architectures ou de circuits déjà éprouvés. Même si ceux-ci ne sont pas parfaitement optimisés pour l'étude d'un nouveau système, leur emploi réduit le temps et le cout de conception tout en rendant possible la mise en œuvre d'éléments dont l'équipe de conception n'a pas la complète maîtrise technique.

Dans le but de réduire le volume de données générées et de réduire le temps de simulation, les concepteurs de SystemC-AMS ont prévu un mécanisme de décimation. Par défaut, tous les points calculés sont enregistrés. Une fois le mécanisme de décimation activé en spécifiant un coefficient de décimation, appelé ici CD , ce mécanisme n'écrit dans le fichier de données qu'une fois tous les CD points. Ce mécanisme n'a pas besoin d'être activé quand le signal le plus rapide a une période T_{min} inférieure à 25 fois la valeur donnée par (33) puisque dans ce cas le pas de simulation est calculé de façon à produire 25 points par période (cf. définition 22, avec $NbSeg = 25$). Quand le signal le plus rapide a une période T_{min} supérieure à 25 fois la valeur donnée par (33), la durée à obtenir entre deux points consécutifs après décimation est $1/25e$ de T_{min} . Le coefficient de décimation à appliquer est donc le rapport entre cette durée et celle du pas de simulation.

Coefficient de décimation

$$CD = \frac{T_{min}}{25 \cdot \Delta t}$$

$$\Leftrightarrow CD = \frac{2 \pi \cdot GBP \cdot T_{min}}{25} \quad (36)$$

Les tests présentés au tableau 10, ont été refaits avec la décimation activée, toute chose étant égale par ailleurs. Les résultats consignés dans le tableau 11 montrent que la réduction de temps de simulation obtenue grâce à la décimation est d'autant plus importante que le ratio entre le produit gain-bande et la fréquence du signal d'entrée est grand. La réduction de temps de simulation est néanmoins significative dans tous les cas puisque son ratio est compris entre 2,5 et 4.

Il est à noter que SystemC-AMS autorise la modification du coefficient de décimation en cours de simulation, sous réserve que la simulation soit suspendue au moment du changement. Il est donc possible d'affiner la décimation afin d'obtenir la meilleure adéquation en fonction du contexte du moment.

Exemple d'amplificateur opérationnel	Fréquence f_e du signal d'entrée	Ratio $\frac{GBP}{f_e}$	Coefficient de décimation	Temps de simulation pour 10 périodes		Points enregistrés (taille du fichier)	
				Sans décimation	Avec décimation	Sans décimation	Avec décimation
AD8541 $GBP = 1 \text{ MHz}$ $Sr = 0,92 \text{ V}/\mu\text{s}$ $Iq = 45 \mu\text{A}$	43 kHz	23,3	5	12 ms	4 ms	1463 pts (96,3ko)	293 pts (19,2 ko)
	300 Hz	3333	837	1,8 s	500 ms	209644 pts (13,5 Mo)	250 pts (16,5 ko)
	30 Hz	33333	8377	19,5 s	5,3 s	2096437 pts (135 Mo)	250 pts (16,4 ko)
AD8500 $GBP = 7 \text{ kHz}$ $Sr = 4 \text{ V}/\text{ms}$ $Iq = 0,75 \mu\text{A}$	300 Hz	23,3	5	10 ms	4 ms	1467 pts (96,4 ko)	294 pts (19 ko)
	30 Hz	233	58	120 ms	34 ms	14661 pts (964 ko)	253 pts (16,7 ko)
	0,21 Hz	33333	8377	19,1 s	4,7 s	2094434 pts (138 Mo)	250 pts (17 ko)

Tableau 11 : Réduction du temps de simulation par décimation du nombre de points enregistrés

Chapitre VI)

Perspectives

Les travaux menés ainsi que les rencontres et les échanges scientifiques effectués tout au long de cette thèse ont éveillé des réflexions quand aux usages de SystemC-AMS ou à l'évolution de la conception assistée par ordinateur des systèmes hétérogènes à signaux mixtes et analogiques. Certaines de ces réflexions ont muri davantage et sont devenues assez élaborées. D'autres sont restées au niveau d'une simple idée à creuser. Ces réflexions se regroupent en trois catégories : utiliser SystemC-AMS pour enrichir les modèles numériques, améliorer le langage, progresser vers la génération automatique des systèmes hétérogènes.

VI.1) Introduire ou mesurer des effets analogiques dans les modèles de circuits logiques

Bien que spécialement conçu pour l'étude des circuits mixtes et analogiques, SystemC-AMS peut être mis à profit pour compléter les modèles de circuits numériques initialement décrits en SystemC.

SystemC-AMS peut aussi servir à mesurer des grandeurs analogiques dans un circuit numérique comme la consommation électrique. Une telle évaluation peut aider à choisir l'architecture la moins gourmande en énergie ou encore à dimensionner une alimentation ou un système de refroidissement. Des travaux en ce sens ont été entrepris par Laurent Bousquet pour intégrer des informations de consommation électrique dans des modèles analogiques à haut niveau d'abstraction (cf. [70 – 72]).

Tout circuit numérique est constitué de transistors. Sur le silicium, il s'agit donc d'un composant analogique avec un comportement fortement non-linéaire. Ils est donc soumis à des phénomènes analogiques dont certains ont des répercussions sur le comportement au point de pouvoir altérer le comportement numérique s'ils ne sont pas évalués correctement. SystemC-AMS peut donc servir à enrichir les modèles des circuits numériques écrits en SystemC avec des informations concernant ces phénomènes analogiques ayant potentiellement des conséquences sur le comportement. Parmi ceux-ci, les temps de propagation des signaux font partie de ceux qui ont le plus souvent des conséquences. Leur impact peut être ressenti même aux niveaux d'abstraction les plus élevés par le retard ou les erreurs dans l'acheminement des données qu'ils engendrent. Une réflexion est menée ci-après sur la façon de les intégrer dans la description numérique et sur la manière d'utiliser SystemC-AMS dans ce but.

VI.1.1) Éléments généraux et définitions à propos des temps de propagation

Les temps de propagation et d'établissement examinés dans cette réflexion concernent les circuits CMOS. Pour que l'évaluation de leur impact s'inscrive dans le cadre d'un processus de raffinement progressif et structuré, leur modélisation doit s'ajouter au modèle numérique sans nécessiter la réécriture de celui-ci. L'approche la plus pertinente consiste à adjoindre des blocs de retard à chaque sortie (éventuellement sur les entrées en fonction du contexte) du modèle numérique non temporalisé. Ce principe est illustré par la figure 61.

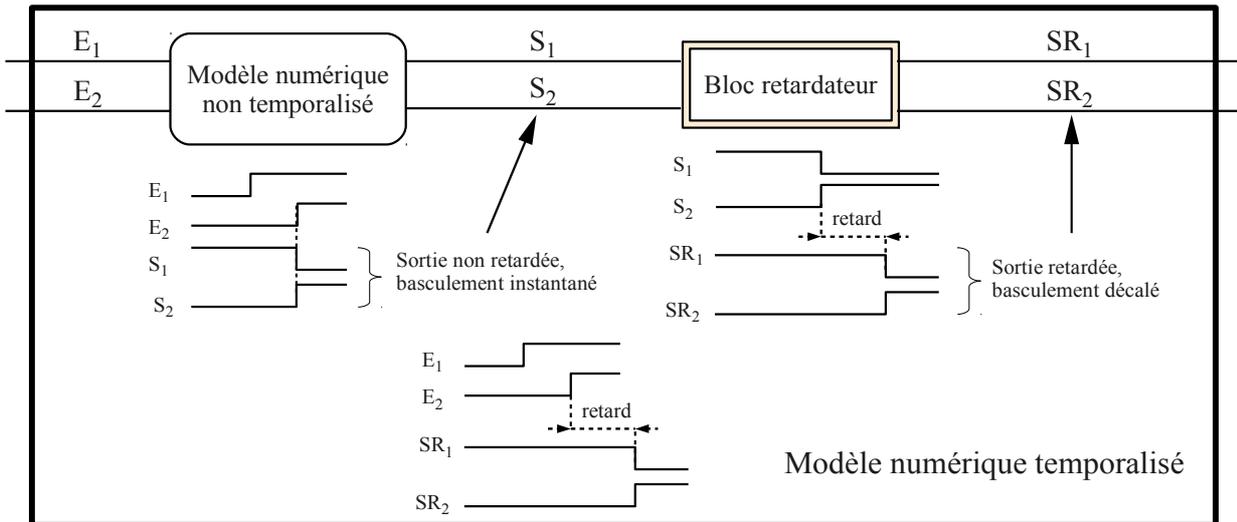


Fig. 61 : Temporalisation des modèles numériques par ajout d'un bloc retardateur

Définition 23 Temps de propagation

Le temps de propagation est défini comme le temps séparant le moment où le signal en entrée est à 50 % de sa valeur finale, du moment où la sortie est à 50 % de sa valeur finale. Le temps de propagation est désigné ici par t_{pHL} quand la sortie passe de 1 à 0 logique (transition négative) et par t_{pLH} quand elle passe de 0 à 1 logique (transition positive). Les temps t_{pHL} et t_{pLH} peuvent être différents.

Quand une porte a plusieurs entrées, le temps de propagation est évalué par rapport à l'entrée qui, la dernière, remplit la condition nécessaire pour le basculement de la sortie. Ceci est illustrée fig. 62. Le terme d'entrée doit être ici compris au sens très large. Il s'agit non seulement d'une entrée de donnée mais aussi d'une entrée de sélection (ex : multiplexeur), d'horloge ou d'activation. Avec une telle approche l'analyse présentée ici peut ainsi être généralisée à n'importe quel circuit logique CMOS. Dans le cas d'un circuit logique à plusieurs sorties, il convient d'appliquer sur chaque sortie le temps de propagation correspondant à la transition en cours.

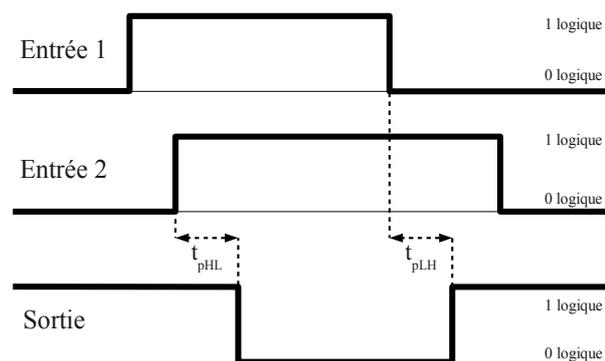


Fig. 62 : Temps de propagation à la descente et à la montée

Pour des raisons de simplicité, la suite du propos est développée sur une seule sortie et ne sera indiquée comme entrée que celle qui est la cause du changement d'état de la sortie.

Le temps de propagation est lié à la topologie interne du circuit numérique : nombres d'étages et dimensionnement des sources internes de courant. Pour un circuit élémentaire donné (topologie arrêtée et sources de courant dimensionnées), le temps de propagation de chaque sortie est conditionné par le nombres d'entrées qu'elle doit piloter. Une fois l'insertion faite dans un circuit plus complexe, chaque sortie de chacun des circuits élémentaires a un nombre défini d'entrées à piloter. Ce nombre ne change pas au cours du fonctionnement. Toutefois le temps de propagation peut changer en cours de fonctionnement : la température et la tension d'alimentation du circuit le font varier.

Le temps moyen de propagation, t_p , d'un circuit logique CMOS peut être approché par une fonction inverse de la tension d'alimentation, V_{dd} . La formule générale est donnée en (37). Les coefficients k , a et b dépendent de la technologie employée et de la topologie de la porte. Ils sont caractéristiques de la bibliothèque de portes utilisée pour réaliser le circuit numérique. Ils sont en général spécifiés pour une valeur de référence de la capacité chargeant la sortie de la porte logique. Quand les temps de propagations sont différents à la montée et à la descente, ce sont les coefficients k , a et b qui sont différents. Il y a donc une équation pour la transition positive et une autre pour la transition négative.

$$t_p = \frac{1}{k \cdot V_{dd} - a} + b \tag{37}$$

Une loi plus complète pourrait être établie en observant les variations du temps de propagation en fonction de la température sur une série de circuits réels (une telle information est en générale consignée dans les fiches techniques établies par les fondeurs de composants).

Les modèles temporalisés doivent comporter une entrée donnant la mesure de la tension d'alimentation et de la température du circuit si celle-ci est prise en compte.

La prise en compte du temps de propagation peut être faite avec différents niveaux d'abstraction. Dans un premier temps la sortance de chaque sortie est considérée comme infinie, c'est à dire que le signal délivré par une sortie n'est pas altéré, ni en amplitude ni en temps d'établissement, par le nombre d'entrées qu'elle doit piloter. Avec cette simplification, soit les signaux délivrés sont considérés comme strictement rectangulaires et le temps de propagation est alors modélisé comme un retard pur, soit ils sont considérés comme trapézoïdaux, les transitions sont des rampes, et le temps de propagation est liés aux temps de montée et de descente. Pour la prise en compte de la sortance, un modèle plus précis est nécessaire. Dans ce modèle, c'est la combinaison de la résistance sérielle de la sortie avec la capacité de chaque entrée pilotée par cette sortie qui conditionne le temps d'établissement du signal de sortie.

VI.1.2) Modélisation du temps de propagation par des signaux rectangulaires retardés

Le niveau d'abstraction est ici encore relativement élevé puisque les signaux d'entrée et de sortie sont purement booléens : les amplitudes sont logiques et les transitions instantanées (cf. fig. 63).

La sortie du bloc retardateur est une version retardée du signal reçu sur son entrée. Le retard modélise le temps de propagation à travers le circuit numérique temporalisé.

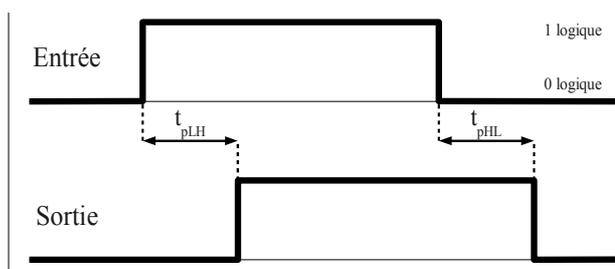


Fig. 63 : Temps de propagation modélisé par un retard

Dans ce modèle :

- Il est supposé que la sortie ne bascule pas si l'impulsion en entrée a une durée plus courte que le temps de propagation du circuit (principe d'inertie).
- La tension d'alimentation ne sert qu'à calculer la valeur du retard.
- La porte écrête sa tension d'alimentation : si, en cours de simulation, la tension d'alimentation dépasse le maximum défini à l'élaboration, elle est alors forcée à cette tension maximale. Si, en cours de simulation, la tension d'alimentation devient négative, elle est alors considérée comme valant 0 volt.
- La capacité de charge sur la sortie de la porte logique est ignorée. Elle n'influence pas le temps de propagation. La sortance de la porte est considérée comme infinie.

Le comportement de la porte logique est décrit par la machine à états montrée fig. 64. Sur cette figure, les états stables sont symbolisés par des ovales et les états transitoires par des rectangles. Les flèches pleines décrivent les transitions normales tandis que les flèches en pointillées renvoient au cas de transitions abandonnées du fait d'une impulsion en entrée plus courte que le temps de propagation de la porte.

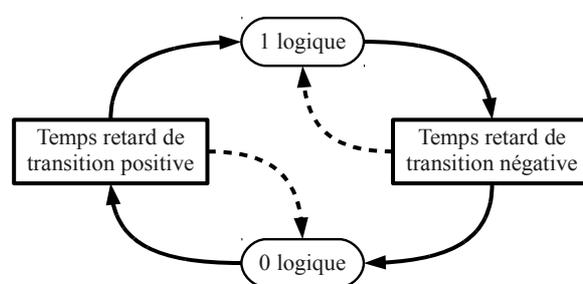


Fig. 64 : Comportement avec un temps de propagation modélisé par un retard

Lorsqu'un front est détecté en entrée, la tension d'alimentation est lue et le temps de propagation est calculé selon l'équation (37). Pendant la phase transitoire, l'entrée est constamment évaluée. Si, avant que le délai correspondant au temps de transition ne soit écoulé, elle reprend la valeur correspondant à l'état logique de départ, alors la transition est abandonnée et la sortie ne bascule pas. Si pendant tout le

temps de transition, l'entrée à conservée la valeur indiquant le changement d'état de la sortie, alors une fois écoulé le délais correspondant au temps de transition, la sortie est positionnée à l'état logique correspondant à la valeur en entrée.

SystemC avec la fonction `wait(...)` et avec les paramètres de délai dans les canaux et les initiateurs de port dans le cas de la modélisation transactionnelle, dispose de tout le nécessaire pour modéliser le temps de propagation de cette manière. Utiliser SystemC-AMS dans ce cas, n'apporterait rien de plus voire pourrait s'avérer pénalisant à cause d'une simulation continue, dont le pas de temps est fixe dans la actuelle du standard (V 1.0), comparée à la simulation événementielle de SystemC.

VI.1.3) Transition en rampe, temps de propagation fonction des temps de transition

Dans le présent modèle, la transition entre les niveaux logiques est effectuée selon une rampe de pente déterminée. La pente de la rampe peut être différente selon qu'il s'agit d'une transition positive ou d'une transition négative. Il n'y a pas de lien entre les pentes des rampes du signal d'entrée et du signal de sortie.

i) Niveaux logiques, temps de transition, temps de propagation

Afin de caractériser les rampes, les niveaux logiques doivent être exprimés en terme d'amplitude. Le moment et la durée pendant lesquels chaque rampe a lieu doivent être définis.

Définition 24 Niveaux logiques

Les niveaux logiques doivent être exprimés. Pour les circuits logiques CMOS, un niveau est assimilé à un 1 logique quand il est supérieur ou égale à 90 % de la tension d'alimentation V_{dd} et à un 0 logique quand il est inférieur ou égale à 10 % de V_{dd} . Entre 10 % et 90 % de V_{dd} , un signal a une valeur logique indéterminée qui n'est ni 1 ni 0.

Définition 25 Temps de montée, de descente, de transition

Le temps de montée t_r d'une sortie logique est le temps mis par le signal de cette sortie pour passer de 10 % à 90 % de V_{dd} (transition positive). Le temps de descente t_f de la sortie logique est le temps mis par le signal de cette sortie pour passer de 90 % à 10 % de V_{dd} (transition négative). Ces temps, qui peuvent être différents l'un de l'autre, sont caractéristiques du circuit logique considéré. Ils définissent la pente de la rampe de transition entre niveaux logiques. Voir fig. 65.

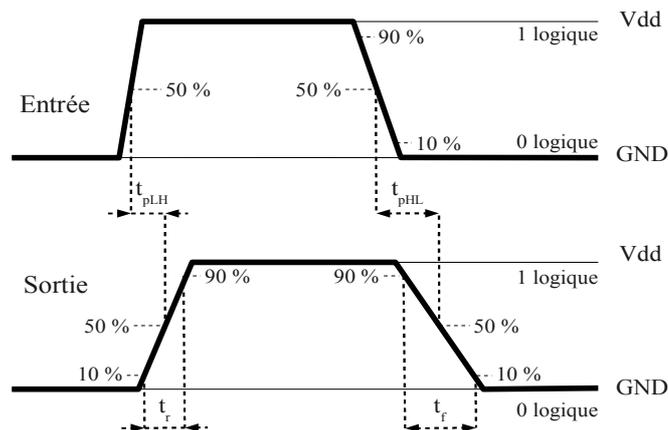


Fig. 65 : Transitions normales en rampe

Le terme « temps de transition » désigne sans distinction soit le temps de montée soit le temps de descente. Le temps de transition dépend de la tension d'alimentation et de la température de fonctionnement du circuit¹⁸. Cependant de part la définition des temps de montée et de descente, le temps de transition représente toujours 80 % du temps mis par la sortie pour passer complètement d'un niveau logique à l'autre.

¹⁸ Dans le cas général, les temps de transition et de réponse à un échelon dépendent aussi de la capacité chargeant la sortie. Cependant dans le modèle avec transition en rampe, la sortance est considérée comme infinie donc la capacité en sortie n'a pas d'influence.

Définition 26 Temps de réponse à un échelon (*step response time*)

Le temps de réponse à un échelon est le temps qui sépare le commencement de la transition en sortie et le moment où le signal de sortie franchit 50 % de V_{dd} . Il est corrélé au temps de transition et tout comme lui, il dépend de la tension d'alimentation et de la température de fonctionnement du circuit¹⁸.

Le temps de réponse à un échelon est notée t_{sHL} pour une transition négative et t_{sLH} pour une transition positive. Voir fig. 66.

Décomposition du temps de propagation

Structurellement, chaque sortie d'un circuit logique CMOS est reliée à ses entrées à travers des capacités parasites en série. La conséquence est que l'échelon d'entrée se propage partiellement jusqu'à la sortie à travers ces capacités parasites et s'additionne à la tension de sortie, d'où un dépassement de la tension nominale (*overflow*). La durée et l'amplitude de ce dépassement dépendent de la raideur du front du signal en entrée. La porte doit évacuer les charges supplémentaires ainsi apportées en sortie avant de pouvoir commencer sa transition. Il s'en suit un retard de la réponse de la porte à l'échelon d'entrée. Plus le front en entrée est raide et plus la sortie tarde à commencer sa transition par rapport au moment où le signal d'entrée atteint sa valeur finale (V_{dd} pour un front montant ou 0 V pour un front descendant). A l'inverse plus la pente du signal en entrée est faible et plus la sortie commence tôt sa transition par rapport au signal d'entrée. Quand la pente du signal d'entrée devient très faible, il est possible que la sortie commence sa transition avant même que le signal d'entrée est atteint sa valeur finale. [73]

Dans le cas de transitions rapides, telles qu'exposées dans [73] et [74], le moment où la sortie commence sa transition est toujours postérieure au moment où l'entrée franchit 50% de V_{dd} . De plus, les temps de réponse à un échelon, de transition et de propagation sont indépendants de la raideur du front du signal en entrée. Par conséquent, pour une tension d'alimentation et une capacité de sortie données, ces temps sont constants. La transition rapide en entrée est aussi une condition pour que le temps de propagation puisse être calculé par l'équation (37) sans tenir compte de la raideur du front du signal d'entrée.

Définition 27 Temps mort

Le temps mort correspond au retard de la porte à répondre à l'échelon d'entrée. Dans ce modèle, les transitions en entrée seront toujours considérées comme des transitions rapides pour le circuit étudié. Le temps mort est donc une grandeur strictement positive.

Le temps mort est noté t_{dtHL} pour une transition négative et t_{dtLH} pour une transition positive.

Dans ce modèle de transition en rampe, le dépassement de l'amplitude en sortie sera ignorée car, dans la plupart des cas réels, il ne représente que quelques pourcents de la tension nominale. Seul le retard est pris en compte. Le temps de propagation (cf. définition 23) peut être modélisé par la succession de deux temps : un temps mort (*dead time*), suivi du temps de réponse à un échelon comme cela est illustrée sur la figure 66.

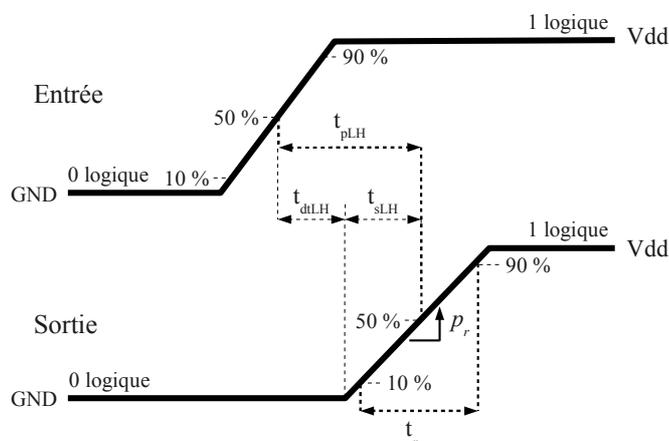


Fig. 66 : Décomposition du temps de propagation

Puisque le temps de propagation et le temps de réponse à un échelon sont indépendants de la raideur du front du signal en entrée, le temps mort le sera aussi. Il sera donc lui aussi constant pour une tension d'alimentation et une capacité de sortie données. Malgré cela, le comportement décrit reste réaliste puisque le moment où commence la transition en sortie, par rapport au moment où finit celle de l'entrée, varie en fonction de la pente d'entrée. Cela est montré figure 67. On notera que conformément à ce qui a été expliqué plus haut, plus la pente en entrée est raide plus la transition en sortie est retardée par rapport à la fin du front du signal d'entrée. Dans le cas d'une faible pente en entrée, la sortie commence sa transition avant la fin de la transition en entrée.

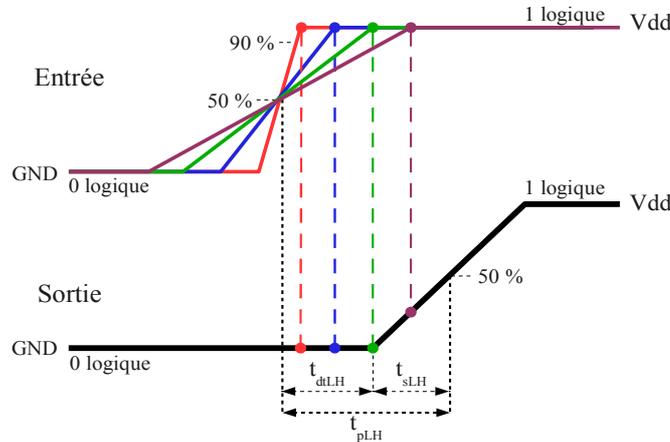


Fig. 67 : Description réaliste grâce au temps mort avant le temps de réponse à l'échelon

Cependant, tel que ce modèle est défini, si le signal d'entrée a une pente plus faible que celle de la sortie, la sortie peut achever sa transition avant la fin de la transition de l'entrée. Ce comportement ne peut se rencontrer que dans le cas d'une transition d'entrée lente pour le circuit¹⁹. Or, dans le cas d'une transition lente en entrée, la transition en sortie commence avant que l'entrée n'ait dépassée les 50% de V_{dd} (temps mort négatif) et la pente du signal de sortie est d'autant plus faible que la pente du signal d'entrée est faible. Le présent modèle devient inadapté. Toutefois, un circuit logique réel n'est pas conçu pour fonctionner avec des signaux d'entrée de très faible pente. Les fiches techniques des constructeurs indiquent un taux de montée (inverse de la pente) minimum que les signaux d'entrée doivent respecter pour garantir le bon fonctionnement du circuit. C'est pour cette raison qu'il a été choisi de ne pas traiter les transitions lentes en entrée dans ce modèle.

Calcul du temps mort et de la pente de la rampe de transition

Le temps mort et la pente de la rampe sont les données nécessaires pour générer la transition en sortie.

Le temps mort est calculé comme la différence entre le temps de propagation et le temps de réponse à l'échelon (voir fig. 66). La formule du temps mort à la descente t_{dHL} est donnée par l'équation (38). La formule du temps mort à la montée t_{dLH} est donnée par l'équation (39). Dans ces équations les temps de propagation à la descente t_{pHL} et à la montée t_{pLH} sont obtenus par l'équation (37).

$$t_{dHL} = t_{pHL} - t_{sHL} \quad (38)$$

$$t_{dLH} = t_{pLH} - t_{sLH} \quad (39)$$

Selon la définition 25 et parce que les transitions sont modélisées par des rampes, le temps de transition représente 80 % du temps mis par la sortie pour passer complètement d'un niveau logique à l'autre. Le temps de réponse à l'échelon (t_{sHL} pour une transition négative et t_{sLH} pour une transition positive) représente 50 % de cette même durée (voir fig. 66). Il s'en suit les relations suivantes entre les deux.

$$t_f = 80\% \cdot (2 t_{sHL}) \quad (40)$$

$$t_r = 80\% \cdot (2 t_{sLH}) \quad (41)$$

$$t_{sHL} = \frac{5}{8} \cdot t_f \quad (42)$$

$$t_{sLH} = \frac{5}{8} \cdot t_r \quad (43)$$

¹⁹ La notion de transition rapide ou lente de l'entrée est relative. Cela est expliqué dans [74].

Quand les temps de transition ne sont pas renseignés, le temps de réponse à l'échelon sera fixé à 50% du temps de propagation.

La pente de la rampe de transition est calculée à partir du temps de réponse à l'échelon.

$$p_f = -\frac{\frac{1}{2}V_{dd}}{t_{sHL}} = -\frac{V_{dd}}{2t_{sHL}} \quad (44)$$

$$p_r = \frac{\frac{1}{2}V_{dd}}{t_{sLH}} = \frac{V_{dd}}{2t_{sLH}} \quad (45)$$

ii) Transitions normales, transitions critiques

La figure 65 présente les transitions normales c'est-à-dire quand le signal d'entrée passe totalement d'un niveau logique à un autre et que le front suivant en entrée se produit après que la sortie ait fini sa transition.

Les transitions critiques sont les transitions avortées et les impulsions brèves en entrée dont les définitions suivent.

Définition 28 Transition avortée (*glitch*)

Une transition en entrée est avortée quand, dans le cas d'un front montant, le signal revient à GND sans avoir dépassé 90 % de V_{dd} (partie gauche de fig. 68). De même pour un front descendant, la transition est avortée quand le signal d'entrée revient à V_{dd} sans être passé sous 10 % de V_{dd} (partie droite de fig. 68).

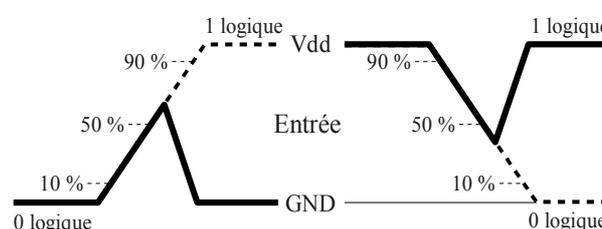


Fig. 68 : Transitions avortées

Définition 29 Impulsion brève en entrée

Une impulsion brève est la succession en entrée, d'un front et d'un front de sens opposé avant que la porte n'ait fini sa transition en sortie (fig. 69). La différence essentielle avec la transition avortée est que, dans le cas de l'impulsion brève, le signal d'entrée effectue des transitions complètes entre les deux niveaux logiques. La porte commence donc normalement sa transition en sortie mais n'a pas le temps de la terminer. La situation critique provient de la brièveté dans la succession des fronts. Cette situation se rencontre quand le circuit est attaqué en entrée par des signaux trop rapides pour lui.

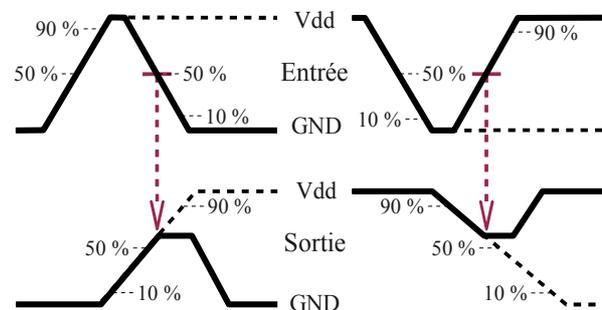


Fig. 69 : Impulsions brèves en entrée

L'impulsion brève est positive dans le cas d'un front montant suivi d'un front descendant et négative dans le cas d'un front descendant suivi d'un front montant.

iii) Comportement de la sortie lors d'une transition critique

Deux cas peuvent se présenter selon que le signal revient à l'état initial pendant ou après le temps mort en sortie. Le cas d'un front montant en entrée sert d'illustration. Un comportement symétrique existe avec un front descendant en entrée.

Transition critique achevée avant la fin du temps mort

Quand, lors d'une transition critique, le signal d'entrée franchit de nouveau les 50 % de V_{dd} avant la fin du temps mort en sortie, le signal de sortie n'effectue pas de transition (phénomène d'inertie). En effet, au moment où le signal d'entrée repasse sous les 50 % de V_{dd} , la sortie se trouve dans la situation d'une transition négative. Elle doit commencer le temps mort dû au front descendant en entrée alors même que le temps

mort dû au front montant précédent n'est pas terminé. Par conséquent la sortie ne changera pas d'état comme l'illustre la figure 70.

Puisque le temps de propagation et par conséquent le temps mort n'est compté qu'à partir du moment où le signal d'entrée traverse 50 % de V_{dd} , une transition avortée dont l'amplitude ne franchit pas 50 % de V_{dd} sera toujours ignorée par la porte (aucune variation en sortie).

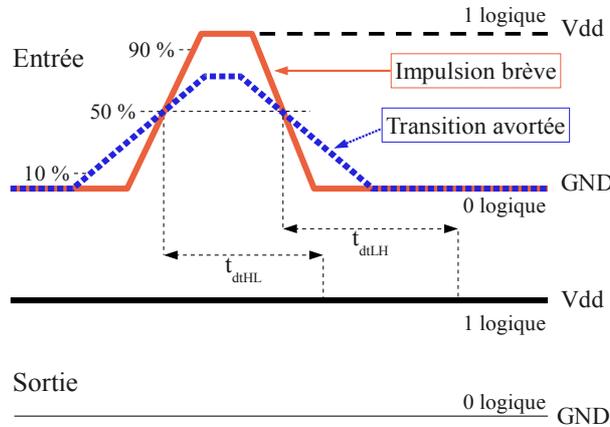


Fig. 70 : Transitions critiques plus brèves que le temps mort

Transition critique plus longue que le temps mort

Quand lors d'une transition critique le signal d'entrée est au delà des 50 % de V_{dd} à la fin du temps mort t_{dtHL} en sortie, la sortie commence sa transition comme dans le cas normal : le signal de sortie décroît varie selon une rampe de pente p_r , pour une transition positive, ou de pente p_f , dans le cas une transition négative. Au moment où le signal d'entrée repasse sous les 50 % de V_{dd} , la porte se retrouve dans la même situation qu'un front descendant en entrée. Le signal de sortie reste inchangé (plateau) pendant la durée du temps mort t_{dtLH} puis il varie dans le sens opposé à celui d'avant le plateau, avec la pente correspondante p_f ou p_r . Il se produit donc une transition avortée (*glitch*) en sortie qu'illustre la figure 71.

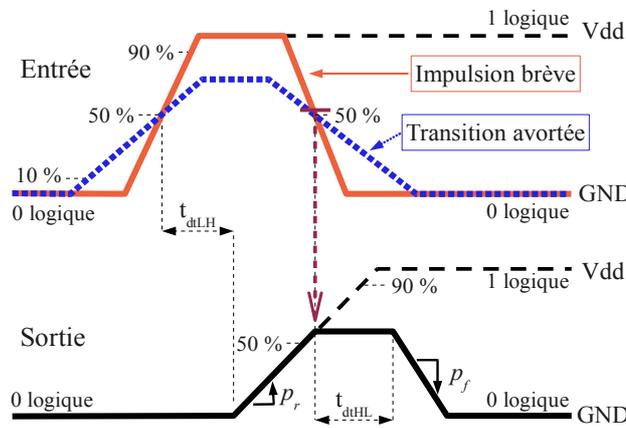


Fig. 71 : Transitions critiques plus longues que le temps mort

iv) Limites du modèle

La liste ci-dessous énumère les approximations et présupposés qui sont faits dans le modèle présenté dans ce chapitre. Les approximations et présupposés déjà indiqués dans les paragraphes précédents sont repris dans cette liste.

- Les transitions entre les niveaux logiques sont effectuées selon une rampe de pente déterminée.
- Les transitions en entrée seront toujours considérées comme des transitions rapides pour le circuit. Les transitions lentes ne sont pas traitées dans ce modèle.

- Le temps de propagation est modélisé par la succession d'un temps mort et du temps de réponse à un échelon.
- Quand les temps de transition ne sont pas renseignés, le temps mort et le temps de réponse à l'échelon sont fixés par défaut à 50% du temps de propagation.
- Une transition avortée en entrée dont l'amplitude ne franchit pas 50 % de V_{dd} sera toujours ignorée par le circuit.
- La marge de bruit est ignorée, la même définition des niveaux logiques s'applique aussi bien pour un signal d'entrée que pour un signal de sortie.
- Les variations de V_{dd} sont beaucoup plus lentes que les transitions. V_{dd} ne change pas au cours d'une transition.
- Au cours du fonctionnement, V_{dd} ne dépasse jamais la valeur nominale spécifiée lors de la construction du modèle.
- Le signal d'entrée est implicitement écrêté : l'amplitude prise en compte en entrée ne dépasse pas la tension d'alimentation de la porte. Une surtension en entrée n'est pas destructive.
- Le signal de sortie est implicitement écrêté : l'amplitude de la sortie ne dépasse jamais les rails d'alimentations. Les dépassements lors des transitions (*overshoot & undershoot*) sont donc ignorés.
- Les capacités de charge sur les sorties du circuit numérique sont ignorées. Elles n'influencent pas le temps de propagation dans ce modèle. La sortance du circuit est considérée comme infinie.

v) Comportement de la sortie logique temporalisée avec des transitions en rampe

Le comportement d'une sortie logique d'un circuit numérique décrit selon le modèle avec des transitions en rampe, est décrit par la machine à états montrée fig. 72.

Sur la figure 72, les états stables sont symbolisés par des ovales et les états transitoires par des rectangles. Les flèches pleines décrivent les transitions normales tandis que les flèches en pointillées renvoient aux transitions critiques. Le passage d'un temps mort à l'autre se produit dans le cas d'une transition critique plus brève que le temps mort (cf. fig. 70). Le passage direct d'une transition à un temps mort, sans passer par un état stable, a lieu dans le cas d'une transition critique plus longue que le temps mort (cf. fig. 71).

Le modèle de calcul (MoC) SystemC-AMS à utiliser est le TDF. La méthode `processing()` du MoC TDF recueillera l'algorithme décrivant le comportement conformément à la machine à états de la figure 72. Cette méthode est appelée cycliquement par le simulateur. Une première lecture et mémorisation de V_{dd} est à faire dans la fonction d'initialisation, la méthode `initialize()` du MoC TDF.

Le pas de simulation (*time step*) est à choisir de telle sorte qu'il y ait au moins dix points calculés pendant les transitions les plus rapides c'est-à-dire quand la porte est alimentée à la tension maximale.

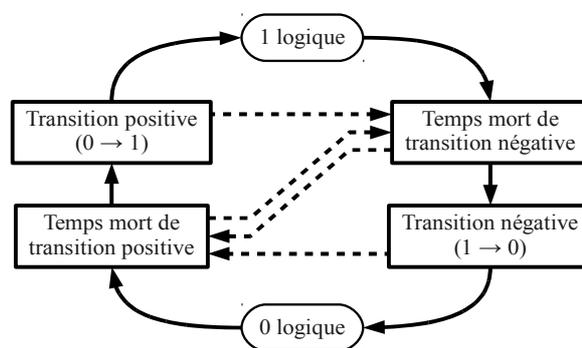


Fig. 72 : Comportement avec des transitions en rampe

VI.1.4) Transition en exponentielle, temps de transition dépendant de la capacité de charge

Le modèle présenté schématiquement dans ce paragraphe, tout en étant capable lui-aussi de traiter les transitions critiques, ajoute la prise en compte de l'influence de la charge d'une sortie sur le temps de transition de celle-ci. Pour cela, la description adoptée est plus proche de la réalité matérielle.

Une entrée de circuit logique CMOS est une commande de grille. À cause de la capacité de grille des transistors MOSFET, chaque entrée d'un circuit logique CMOS présente une capacité en parallèle avec la

masse. La valeur de la capacité est principalement liée à la technologie employée et au dimensionnement des transistors d'entrée.

Les sorties d'un circuit numérique CMOS sont des sorties en tension. Mais chacune d'elle a une résistance série non nulle. Il s'agit de la résistance du transistor en saturation qui connecte la sortie soit avec le rail d'alimentation positive dans le cas d'un '1' logique (PMOS), soit avec le rail d'alimentation négative dans le cas d'un '0' logique (NMOS). La résistance série de la sortie peut donc être différente pour une transition positive et une transition négative, d'où potentiellement des temps de montée et de descente différents. La valeur de la résistance série d'un transistor en saturation dépend de la technologie employée mais surtout de la structure et du dimensionnement de l'étage de sortie. L'ingénieur qui en est chargé peut fortement agir sur cette valeur.

Ceci est reproduit en utilisant le modèle de calcul ELN de SystemC-AMS pour ajouter, au module numérique à temporaliser, un condensateur en parallèle sur chacune de ses entrées et une résistance en série avec chacune de ses sorties, comme l'illustre la figure 73 a. Des convertisseurs sont nécessaires pour assurer la bonne synchronisation entre le domaine analogique (ELN) et le domaine numérique (noté DE pour *discrete event*). La connexion d'une sortie d'un module temporalisé à une entrée d'un autre module temporalisé constitue un réseau RC (cf. fig. 73 b). Le signal sur ce nœud a une forme exponentielle fidèle à la réalité. Quand une même sortie est reliée à plusieurs entrées, les capacités des entrées sont en parallèle et s'additionnent. La constante de temps augmente. Ainsi le temps de transition de la sortie est conditionné par la charge de la sortie. Quand le nombre d'entrées à piloter par une même sortie est trop élevé, des transitions critiques (cf. VI.1.3)ii) apparaissent à cause de la constante de temps devenue trop importante, dit autrement à cause de temps de transition excessivement allongés. Il en résulte des dysfonctionnements du circuit. Avec ce modèle, les concepteurs devraient obtenir des informations quant à la valeur de sortance minimale à prévoir lors du dimensionnement du circuit au niveau transistor.

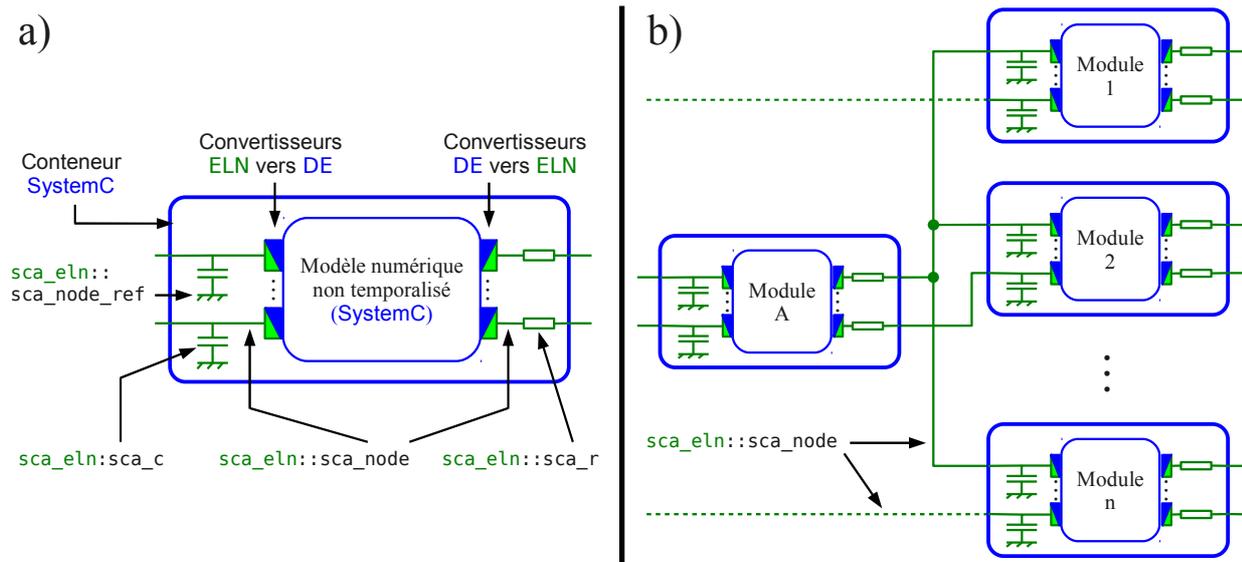


Fig. 73 : Modélisation des interconnexions par un RC – transitions en arcs d'exponentielle
 a) Encapsulation du module numérique non temporalisé
 b) Interconnexions entre modules temporalisés avec des nœuds ELN

Les convertisseurs entre le domaine analogique et le domaine numérique seront un peu plus complexes que ceux livrés en standard avec SystemC-AMS. Pour les entrées, le convertisseur pour passer du domaine analogique vers le domaine numérique (ELN vers DE) doit remplir le rôle d'un comparateur en délivrant un signal booléen sur sa sortie. Il peut y avoir un seuil de basculement différent pour les transitions positives et les transitions négatives (hystérésis). Concernant les sorties, le schéma de la figure 73 a été très simplifié dans le but d'en faciliter l'explication. Le convertisseur à utiliser pour passer du domaine numérique vers le domaine analogique (DE vers ELN) devrait davantage ressembler au schéma de la figure 74. Avec ce schéma l'influence de la tension d'alimentation est conservée et la constante de temps est différenciée entre les transitions positives et négatives.

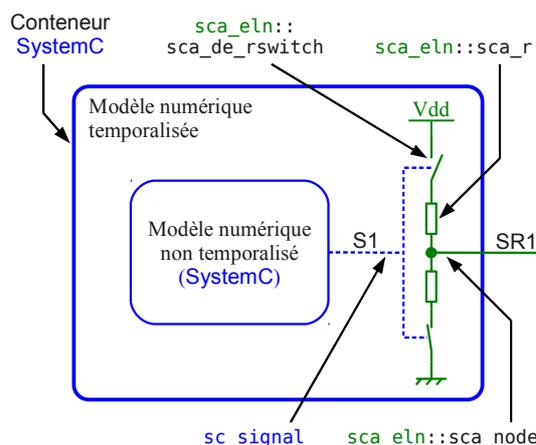


Fig. 74 : Schéma détaillé d'un convertisseur DE vers ELN pour une sortie numérique

VI.2) Améliorer SystemC-AMS

SystemC-AMS a montré ses possibilités et, son adoption par les industriels, même si elle n'est pas encore généralisée, dénote d'un potentiel certain à aider les concepteurs des systèmes hétérogènes complexes. Mais dans sa définition actuelle selon la version 1.0 du manuel de référence du langage [5], SystemC-AMS est limité par certains manques et insuffisances pour complètement répondre au besoin desdits concepteurs.

En premier lieu, la méthode pour calculer la dérivée est à améliorer. Le solveur analogique de SystemC-AMS utilise la méthode d'Euler qui assimile la dérivée au taux d'accroissement entre deux échantillons consécutifs. Cette méthode a l'avantage d'être facile à mettre en œuvre et à ne requérir que très peu de calculs, elle est donc très rapide. Mais, elle est imprécise et fragile. En effet, le résultat dépend fortement du rayon de courbure local de la courbe, dont la dérivée est évaluée, par rapport à la distance séparant les échantillons. La valeur de la dérivée produite par le solveur analogique de SystemC-AMS est très sensible au pas de temps de la simulation. C'est cette sensibilité qui a conduit à établir un critère de réglage de ce pas de temps en fonction du produit gain-bande pour la bonne utilisation du modèle de l'amplificateur opérationnel simplifié présenté au chapitre V (voir le §V.3.8). Ce problème risque de s'amplifier dans la future version de SystemC-AMS avec l'adaptation dynamique du pas de temps. Suite à un changement de la finesse du pas de temps en cours simulation, un signal qui n'a pas changé pourrait être considéré comme ayant une dérivée différente à cause de la sensibilité de la méthode de Euler au pas de temps. Cependant, SystemC-AMS est prévu pour la simulation rapide. Il ne faut donc pas qu'un remplacement de la méthode de calcul de la dérivée anéantisse l'attrait de ce langage en détériorant sa vitesse de simulation. Il convient donc d'être prudent avec les méthodes itératives ou par approximations successives, certes précises mais lourdes en calculs. Une méthode utilisant trois échantillons consécutifs pourrait donner des résultats satisfaisants en s'inspirant du théorème des accroissement finis (moyenne de Lagrange), voir fig. 75. L'erreur qui consisterait à assimiler la pente de la tangente au point situé entre les deux autres comme étant égale à la pente de la sécante reliant les points extérieurs sera toujours moindre que celle commise par la méthode de Euler qui donne une dérivée à droite différente de la dérivée à gauche (point anguleux).

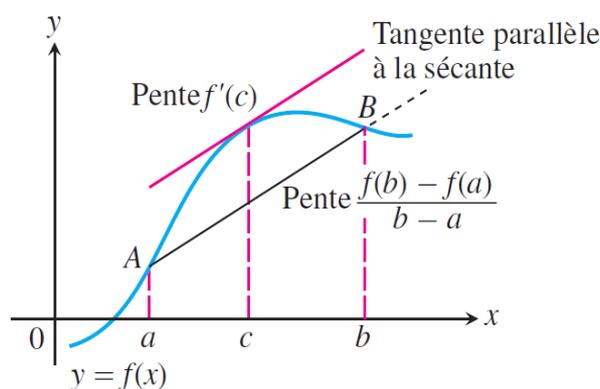


Fig. 75 : Théorème des accroissements finis source [75]

Dans la version 1.0 du standard, SystemC-AMS ne dispose pas de solveur pour résoudre des systèmes d'équations différentielles non linéaires. Par conséquent, seuls les systèmes dont le comportement peut être décrit algorithmiquement ou à l'aide d'un système d'équations différentielles linéaires peuvent être gérés par SystemC-AMS. Ceci s'avère insuffisant quand il s'agit de modéliser des systèmes évoluant dans des milieux non linéaires comme par exemple en magnétisme. L'état de l'art, au chapitre II, a montré que des travaux ont déjà été menés pour palier à ce manque (cf. [10 ; 36]). Il est nécessaire de faire évoluer la norme pour intégrer ces résultats et les rendre disponibles pour l'ensemble des utilisateurs de SystemC-AMS au moyen de modèles de calculs supplémentaires.

Le cas de l'amplificateur opérationnel simplifié a montré la possibilité de modéliser certaines non linéarités au moyen de fonctions linéaires par morceaux. Mais la mise en œuvre s'est révélée compliquée parce que la matrice d'équations est résolue par le solveur analogique en une séquence ininterrompue de calculs. Quand une primitive ELN est écrite, il est possible d'enregistrer des fonctions de rappel qui seront exécutées à chaque pas de temps. Selon leur enregistrement, le rappel de ces fonctions est effectué soit avant, soit après le calcul de la matrice d'équations. Pour les variables qui doivent être tracées, les nouvelles valeurs ne sont écrites dans les fichiers de traces qu'une fois achevée l'exécution des fonctions rappelées après le calcul de la matrice. Il est donc possible d'apporter des corrections sur une nouvelle valeur avant son écriture dans un fichier de trace. Mais si la variable, dont il faut corriger la valeur, est utilisée dans la matrice d'équations pour calculer d'autres variables, la valeur corrigée ne pourra être prise en compte qu'au prochain pas de calcul. Il se produit un décalage très gênant qui peut dans certains cas donner des résultats inappropriés. Une modification du déroulement du processus de calcul de la matrice d'équations ouvrirait des perspectives intéressantes pour la modélisation des non-linéarités. En interrompant le calcul de la matrice d'équations chaque fois qu'une nouvelle valeur est obtenue, il serait alors possible d'exécuter une fonction de rappel. Cette fonction pourrait être employée pour corriger la valeur que vient de fournir le solveur. Les calculs pourraient alors être poursuivis dans le même pas de simulation avec la valeur corrigée. Avec cette méthode, il serait par exemple possible de reproduire l'évolution non linéaire d'un courant de diode en fonction de la tension à ses bornes tout en utilisant le solveur linéaire. Une relation linéaire fictive entre le courant de diode et les tensions d'anode et de cathode est à inscrire dans la matrice d'équations de façon à réserver un tour de calcul pour ce courant et indiquer sa dépendance à ces tensions. Cette relation de dépendance force le solveur à calculer les tensions d'anode et de cathode avant le courant à travers la diode. Une fois le courant calculé par le solveur avec la relation fictive, la fonction de rappel est appelée. Elle contient la formule exacte en fonction de la tension aux bornes de la diode. Elle effectue le calcul du courant avec la nouvelle valeur de tension aux bornes de la diode. Puis, elle remplace le contenu obtenu avec la relation fictive par le résultat exact obtenu avec la formule. Ce résultat exact est disponible pour la suite des calculs et pour continuer la résolution de la matrice d'équations. Par l'intervention des fonctions de rappel pendant la résolution de la matrice d'équations, et non plus seulement avant ou après, il serait donc possible de reproduire des variations non linéaires de signaux tout en gardant le solveur linéaire et de ce fait en conservant une grande vitesse de simulation.

Les tests exposés au chapitre IV.2 ont montrés que la phase d'élaboration pouvait représenter une part importante voire prépondérante dans le temps total des simulations qui utilisent SystemC-AMS. Dissocier l'élaboration de l'exécution proprement dite de la simulation offrirait l'avantage de pouvoir relancer une simulation à partir d'une élaboration obtenue précédemment. Ainsi, dans le cas où la description de la partie analogique n'a pas changée, la simulation ne serait plus pénalisée par la répétition de l'élaboration à chaque démarrage. Elle en serait ainsi nettement plus rapide (cf. fig. 25). Grâce à cette dissociation, le travail pourrait être réparti entre des équipes spécialisées dans la créations de modèles et des équipes chargées du développement des systèmes qui seraient utilisatrices de ces modèles. Ceci irait donc dans le sens de la réemployabilité réclamée par les industriels. Les mécanismes rendant possible cette dissociation sont à imaginer, mais ils pourraient être mis en place selon le principe des bibliothèques de code compilé, si répandu en C/C++.

Une autre amélioration qui, pour sembler anodine n'en rendrait pas moins l'usage de SystemC-AMS bien plus pratique, consisterait à revoir le mécanisme de trace. Un module analogique (au sens de SystemC-AMS, classe dérivée de la classe `sca_module`) est dit traçable quand ses instances peuvent générer des traces et être enregistrées comme telles. L'enregistrement des objets à tracer se fait à l'extérieur des modules analogiques. Mais, dans l'implémentation actuelle (la preuve de concept de l'institut Fraunhofer), un seul identifiant est affecté par module analogique. Ainsi, il est impossible de créer un module traçable avec plus

d'une variable à tracer (la dernière variable du module écrase les précédentes puisqu'il n'y a qu'un seul identifiant). Il est impossible de faire la trace de toutes les entrées et sorties d'un même module en enregistrant une seule fois le module dans la liste des objets à tracer. Pour obtenir que toutes les entrées et toutes les sorties d'un même module apparaissent dans les traces, il faut ajouter, un par un dans la liste des objets à tracer, chacun des signaux ou des nœuds (selon le modèle de calcul) connectés aux ports ou terminaux du module. Quand le nombre d'entrées ou de sorties est conséquent, cette opération se révèle vite fastidieuse et, par suite, source d'erreurs fréquentes. Disposer d'un identifiant par objet à tracer et non par module, laisserait une totale liberté au créateur de composants de choisir ce qui doit être tracé. L'utilisateur du composant, de son côté aurait la possibilité d'enregistrer un nombre potentiellement grand de variables à tracer en un nombre réduit d'opérations et sans le risque d'oublier un des signaux.

VI.3) Vers la génération assistée par ordinateur de systèmes hétérogènes

La conception des systèmes hétérogènes à signaux mixtes et analogiques connaîtra un immense progrès le jour où il sera possible de les générer automatiquement par ordinateur. Ceci implique de pouvoir automatiquement synthétiser les parties analogiques à partir de descriptions de haut niveau, ce qui n'est pas encore le cas aujourd'hui. Pour atteindre cet objectif, il est indispensable qu'une méthode générique de synthèse analogique soit définie, que des bibliothèques de structures formalisées ou de composants existent pour chaque niveau d'abstraction et que des moteurs d'inférence soient construits pour exploiter ces bibliothèques en appliquant les règles de la méthode de synthèse.

Le projet CHAMS du Laboratoire d'Informatique de Paris, LIP6, qui a été lancé en 2012 par P. Renault, C. Braunstein et R. Iskander, a pour objectif de synthétiser les blocs matériels propriétaires analogiques (*analog Intellectual Property – aIP*) décrits au niveau bloc analogique voire à un niveau plus élevé. Il s'agit d'obtenir automatiquement la description de la structure au niveau des transistors et le dimensionnement de ces derniers. Des méthodes de dimensionnement automatique des transistors à partir des paramètres électriques spécifiés par le modèle d'abstraction supérieure, ont déjà été développées au LIP6. Mais, la correspondance entre les modèles est encore faite par le concepteur au moyen de simulations électriques. L'apport du projet sera la mise au point d'une méthode formelle de vérification d'équivalence électriques entre différentes architectures de composants analogiques. La méthode devra pouvoir être mise en œuvre automatiquement par un traitement informatique.

Concernant les bibliothèques, une bibliothèque de composants complexes au niveau module analogique a été proposée par l'Université technologique de Vienne pour la réalisation de la partie radio des systèmes communicants sans fil. Cette bibliothèque contient par exemple un modèle d'amplificateur faible bruit (*Low Noise Amplifier – LNA*), différents modulateurs ou démodulateurs mais aussi des éléments plus abstraits comme une unité de calcul de FFT ou un générateur d'un signal gaussien (voir [76]). Le modèle de l'amplificateur opérationnel simplifié, présenté au chapitre V, donne un aperçu du type de composant et du niveau d'abstraction des modèles que devrait contenir une bibliothèque de composants au niveau bloc analogique. Pour que de telles bibliothèques de structures et de composants s'inscrivent dans la perspective de la génération automatique de systèmes hétérogènes, elles doivent être étoffées mais aussi complétées par d'autres afin que la couverture des domaines soit la plus complète. La constitution des bibliothèques pourrait être accélérée par la mise en commun et le partage de modèles de structures au niveau unité fonctionnelle analogique, au niveau module analogique et au niveau bloc analogique. Pour y parvenir, la mise en commun se ferait dans une base de données relationnelle et un outil d'exploration permettrait aux concepteurs de systèmes hétérogènes d'interroger cette base de données, à distance via internet, à la recherche des éléments dont ils ont besoin. Une approche de cette nature a été proposée dans le cadre du projet *ModelLib* (pour *Model Library*)[10]. Pour obtenir un fonctionnement performant et efficace, il sera nécessaire d'établir une classification méthodique des structures analogiques selon leur domaine d'application, leur rôle, leurs caractéristiques et leur niveau d'abstraction. Cette classification devra être normalisée de façon à être partagée par tous. Il sera également nécessaire de normaliser les interfaces et les règles de constitution des structures enregistrées. De la sorte, ces dernières deviendraient indépendantes des outils de développement et donc interchangeables entre ces outils et entre les équipes.

Chapitre VII)

Conclusion

Le succès des systèmes embarqués, sur puce ou en boîtier, est accompagné du désir de disposer de plus en plus de fonctionnalités intégrées ensemble. La complexité de tels systèmes a augmenté et leur nature a changé. Initialement entièrement numériques, ces systèmes sont devenus à signaux mixtes quand ils ont intégré un lien radiofréquence pour une communication sans fil. Ils sont ensuite devenus hétérogènes quand leurs capteurs ou leurs actionneurs n'ont plus été uniquement électriques. Au cours de cette évolution, le rôle du logiciel enfoui est resté central. Ce dernier est devenu de plus en plus complexe au fur et à mesure que le matériel à contrôler devenait plus complexe. Maîtriser cette complexité, à la fois matérielle et logicielle, tout en contenant le temps et le coût de développement, a imposé une élévation du niveau d'abstraction et l'adoption d'une démarche de conception descendante. Les prototypes virtuels sont devenus incontournables pour réussir l'élaboration du système et la mise au point du logiciel enfoui. Un prototype virtuel est un modèle du système entier qui restitue les interactions entre les parties analogiques et les parties numériques et permet ainsi d'en tenir compte dès le début du développement. Mais disposer de prototypes virtuels ne suffit pas, une méthodologie est essentielle pour organiser le passage des niveaux les plus abstraits du système virtuel aux détails les plus fins de l'électronique du système réel. La mise en œuvre d'un processus méthodique de raffinement descendant aide la réflexion des concepteurs à progresser à travers les phases de conception d'un circuit que sont l'exploration d'architecture, l'implémentation de l'architecture sélectionnée et la réalisation de l'électronique correspondant à l'implémentation élaborée.

Les efforts déployés au cours des dernières décennies pour améliorer le flot de conception des circuits numériques ont abouti à la mise au point d'un processus de raffinement descendant efficace pour les parties numériques. La conception d'un système numérique est désormais possible dans un temps relativement court. Cependant, il n'existait pas de processus équivalent pour le développement des parties analogiques. Dans le but de répondre à ce besoin, cette thèse introduit un processus de raffinement progressif et méthodique des parties analogiques. L'attention a été plus particulièrement portée sur la définition des niveaux analogiques les plus abstraits et sur la mise en correspondance des niveaux d'abstraction entre parties analogiques et parties numériques. Le principe clé, repris du raffinement descendant des parties numériques, est qu'après raffinement, un niveau d'abstraction doit être conforme aux spécifications reçues du niveau d'abstraction supérieur et doit servir de spécification au niveau d'abstraction inférieur. Cette démarche structurée est adaptée à la cosimulation des parties analogiques et numériques ainsi qu'à la gestion des interfaces entre ces parties. Les non-linéarités dominantes des parties analogiques ou les effets analogiques dans les parties numériques peuvent être pris en compte à des niveaux d'abstraction relativement élevés. Leurs conséquences sur le comportement du système peuvent donc être évaluées très tôt dans le cycle de conception.

L'hétérogénéité des systèmes implique de pouvoir réaliser des simulations conjointes entre différents domaines de la physique. SystemC-AMS, l'extension du langage SystemC pour les systèmes à signaux mixtes et analogiques, simplifie la simulation globale en apportant un environnement de simulation unifié. Grâce à ses différents modèles de calcul intégrés et à sa capacité de gérer différents niveaux d'abstraction, la modélisation globale d'un système à signaux mixtes et analogiques est plus facile à réaliser et plus efficace en simulation. Toutefois, le choix du style de modélisation le mieux adapté à chaque niveau d'abstraction est crucial pour atteindre le meilleur compromis entre vitesse de simulation et précision. Une étude a été menée au cours de cette thèse dans le but d'aider le concepteur à faire le meilleur choix à chaque niveau d'abstraction. En se plaçant dans le contexte de la méthode proposée dans cette thèse pour le raffinement des parties analogiques, les styles de modélisation possibles avec SystemC-AMS à chaque étape du raffinement ont été examinés de façon à comprendre leur impact sur la simulation. Les différentes vitesses de simulation obtenues avec ces styles de modélisation ont été comparées entre elles et avec MATLAB/Simulink. L'étude

indique les niveaux d'abstraction du processus de raffinement analogique pour lesquels SystemC-AMS peut-être employé et précise pour chacun d'eux les styles de modélisation qui sont les plus adaptés.

L'interface entre les modèles issus de l'exploration d'architecture et les modèles plus fin requis pour l'implémentation est une question qui reste entière. La description des parties analogiques des systèmes mixtes et analogiques est aujourd'hui abordée soit de façon très abstraite avec un point de vue système, soit à l'inverse avec une faible abstraction et une vision détaillée de la mise en œuvre physique. Les descriptions avec un niveau élevé d'abstraction sont purement comportementales et conviennent bien pour compléter l'étude des parties numériques aux niveaux les plus abstraits. Cependant, elles masquent complètement les phénomènes physiques qui s'observent sur un circuit réel. Les descriptions proches de la mise en œuvre matérielle imitent de façon précise et détaillée le comportement du circuit réel. Mais, pour atteindre un tel niveau de précision, elles sous-entendent des choix de topologie et de technologie. Les concepteurs ont, de nos jours, tendance à passer directement d'une description avec le point de vue système à une description détaillée proche du matériel. Il tiennent par conséquent pour acquis des choix topologiques et technologiques, pourtant présumés, sans avoir évalué l'impact des non-linéarités et des limitations inévitablement introduites par ces choix dans le comportement du système. Ceci accroît le risque de distorsion entre les niveaux d'abstraction les plus élevés et les plus bas. Le processus de raffinement des parties analogiques exposé dans cette thèse prône une transition plus progressive. Avant de commencer à élaborer une topologie pour chaque bloc analogique, il convient de s'assurer que le comportement du système respecte toujours les spécifications malgré les limitations et les non-linéarités dues à la mise en œuvre de composants électroniques. La définition de la structure interne des modules analogiques est le niveau adapté pour introduire les limitations et les non-linéarités qui peuvent le plus influencer le comportement du module. Pour cela, la structure interne du module doit être décrites avec des blocs analogiques dont le modèle reproduit les limitations et les non-linéarités dominantes. Puisque cette étape doit précéder la détermination de la topologie et de la technologie à utiliser pour chaque bloc analogique, elle implique que les modèles de blocs analogiques ne doivent faire aucune hypothèse topologique ni technologique. Afin d'illustrer ce que devrait être de tels modèles de blocs analogiques, et ainsi démontrer la faisabilité du concept, un modèle d'amplificateur opérationnel a été élaboré en SystemC-AMS. Ce modèle restitue les limitations et les non-linéarités dominantes d'un amplificateur opérationnel que sont la saturation, la vitesse de balayage finie et l'atténuation du gain en fréquence. Pourtant, il garde un niveau d'abstraction suffisamment élevé pour rester indépendant de toute hypothèse sur la structure interne de l'amplificateur ou sur la technologie à employer. Un raffinement dont la progressivité est assurée par le soin apporté aux étapes intermédiaires est, d'une part une méthode efficace de réduction des risques de discordance entre le modèle comportemental et le circuit réel et, d'autre part un moyen d'aider les concepteurs à faire leurs choix topologiques et technologiques.

Les travaux consignés dans cette thèse contribuent à faire progresser les méthodes de conception des systèmes électroniques hétérogènes et posent des jalons vers une possible automatisation du flot. Je suis convaincu que de prochaines étapes rendront possible une conception guidées, partiellement automatisée. Enfin, il est difficile de dire si SystemC-AMS s'imposera comme un standard incontournable mais il est certain que les concepts mis en œuvre feront école. Disposer simultanément de plusieurs styles de modélisation par la mise en œuvre de différents modèles de calcul et l'utilisation de simulateurs différents, disposer de passerelles entre les mondes de l'analogique, du numérique, du logiciel, voire avec d'autres domaines de la physique comme la mécanique ou l'optique, sont autant de principes judicieux pour constituer une approche qui permettra aux concepteurs de se focaliser sur le développement de leur produit et non sur la mise en œuvre d'une cosimulation.

Annexes

Annexe 1)

Transformée de Laplace : rappels succincts

A 1.1) Définitions

Les signaux analogiques étudiés dans le cadre de cette thèse sont des signaux causaux à temps continu et à énergie finie (signaux aperiodiques) ou puissance finie (signaux periodiques). Les fonctions utilisées pour les modéliser seront donc causales et à temps continu. Il est supposé que toutes ces fonctions vérifient les conditions d'existence de la transformée de Laplace : elles sont continues, au moins par morceaux, et d'ordre exponentiel à l'infini c'est-à-dire qu'à partir d'une certaine valeur de la variable, elles sont majorée par une exponentielle.

Une fonction f d'ordre exponentiel a à l'infini vérifie ;

$$\exists \{t_n, M, a\} \in \mathbb{R}^3, M > 0, a > 0$$
$$t > t_n \rightarrow |f(t)| < M \cdot e^{a \cdot t}$$

Le nombre imaginaire de carré égal à -1 est noté j : $j^2 = -1$

La variable dans le domaine temporel est notée t . La variable dans le domaine de Laplace est notée p . \mathbb{C} est une variable complexe à partie réelle positive.

$$\sigma \in \mathbb{R}^+, \omega \in \mathbb{R}, p \in \mathbb{C}$$
$$p = \sigma + j\omega$$

La transformée de Laplace F d'une fonction f du temps est définie par :

$$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}, F(p) = TL[f(t)] = \int_0^{+\infty} f(t) \cdot e^{-p \cdot t} dt$$

Les conditions d'existence énoncées plus haut garantissent que l'intégrale est convergente.

La transformée inverse de Laplace est définie par :

$$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}, f(t) = TL^{-1}[F(p)] = \frac{1}{2\pi j} \cdot \int_{\sigma - j\omega}^{\sigma + j\omega} f(t) \cdot e^{p \cdot t} dp$$

A 1.2) Propriétés utilisées dans l'exposé

Linéarité

Soient deux fonctions f et g réelles, causales et de transformées de Laplace respectives F et G .

$$\{a, b\} \in \mathbb{R}^2, \forall t \in \mathbb{R}, \forall p \in \mathbb{C}$$
$$TL[a \cdot f(t) + b \cdot g(t)] = a \cdot F(p) + b \cdot G(p)$$

Théorème de la dérivation

Pour toute fonction f réelle, causale et de transformée de Laplace F , la transformée de Laplace de la dérivée de f vaut :

$$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}$$

$$TL \left\{ \frac{d f(t)}{dt} \right\} = p \cdot F(p) - f(0)$$

Grâce au théorème de la dérivation, il est possible de remplacer une équation différentielle par une équation algébrique.

A 1.3) Transformées de Laplace remarquables utilisées dans l'exposé

Originale	Transformée de Laplace
$\forall t \in \mathbb{R}$ $f(t)$	$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}$ $F(p) = TL[f(t)]$
Impulsion de Dirac δ	1
Fonction d'Heaviside – échelon unité $u(t)$	$\frac{1}{p}$
Échelon d'amplitude Ech $Ech \in \mathbb{R}, Ech \cdot u(t)$	$\frac{Ech}{p}$
Exponentielle $a \in \mathbb{R}, e^{-a \cdot t} \cdot u(t)$	$\frac{1}{p+a}$

Annexe 2)

Stabilité d'un système linéaire : rappels

Pour l'automaticien un système est stable si, après avoir été écarté de sa position d'équilibre, ce système abandonné à lui-même revient à sa position d'équilibre. Pour répondre à cette exigence, la réponse impulsionnelle $h(t)$ d'un système linéaire stable doit tendre vers zéro quand le temps tend vers l'infini.

$$\forall t \in \mathbb{R}, \lim_{t \rightarrow \infty} h(t) = 0$$

La fonction de transfert d'un système est la transformée de Laplace de sa réponse impulsionnelle. Sous réserve de pouvoir écrire la fonction de transfert sous la forme d'une fraction irréductible de polynômes (degré du numérateur inférieur à celui du dénominateur), la stabilité du systèmes est évaluée par l'examen des pôles de sa fonction de transfert. Il est à noter que les systèmes étudiés étant décrits à l'aide de fonctions réelles dans le domaine temporel, quand la fonction de transfert a des pôles complexes ceux-ci sont toujours par paires conjuguées. Les démonstrations établissant le lien entre la nature des pôles de la fonction de transfert et la forme de la réponse impulsionnelle correspondante peuvent être trouvées dans des ouvrages de mathématiques consacrés à la transformée de Laplace ou dans des ouvrages d'automatique sur les systèmes linéaires asservis. Seuls les résultats sont rappelés ici.

- Si tous les pôles ont une partie réelle négative alors le systèmes est stable. Par corolaire, si au moins un des pôles a une partie réelle positive, le système est instable.
- Si tous les pôles sont réels purs alors la réponse impulsionnelle est apériodique. Par corolaire, si au moins un des pôles a une partie imaginaire non-nulle, la réponse impulsionnelle est oscillatoire.
- Un pôle réel pur à partie réelle positive produit une réponse exponentielle croissante dans le domaine temporel, c'est l'instabilité apériodique
- Un pôle réel pur à partie réelle négative produit une réponse exponentielle décroissante dans le domaine temporel, c'est la stabilité asymptotique apériodique
- Un pôle réel nul produit dans le domaine temporel une réponse constante décalée de la position d'équilibre, c'est l'instabilité astatique
- Une paire de pôles imaginaires purs produit dans le domaine temporel une oscillation sinusoïdale entretenue, ni amortie ni divergente
- Une paire de pôles complexes à partie réelle positive et partie imaginaire non-nulle produit dans le domaine temporel une oscillation sinusoïdale dont l'amplitude croit exponentiellement, c'est l'instabilité oscillatoire
- Une paire de pôles complexes à partie réelle négative et partie imaginaire non-nulle produit dans le domaine temporel une oscillation sinusoïdal exponentiellement amortie, c'est la stabilité asymptotique oscillatoire

Annexe 3)

Amplificateur idéal : détail du calcul du gain en tension

La tension en sortie de l'amplificateur idéal est régie par l'équation (2)

$$V_{out} = A(V^+ - V^-) \quad (2)$$

Un tel modèle sous entend que l'amplificateur est alimenté par deux alimentations symétriques et que la tension de sortie évolue autour de 0.

Les tensions V^+ et V^- en entrée de l'amplificateur sont calculées respectivement pour le montage en contre-réaction et pour le montage en rétroaction positive. Afin de faciliter la lecture, les figures 36 et 37 sont rappelées.

Montage en contre-réaction

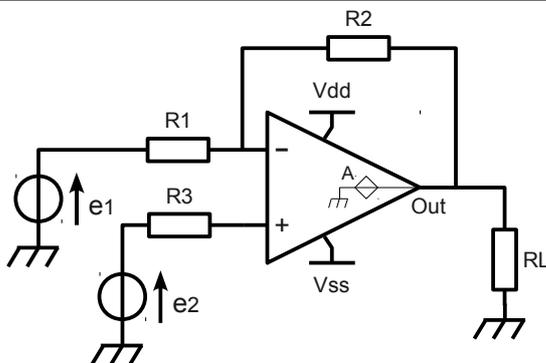


Fig. 36 : Configuration en contre-réaction dans le cas général

$$V^+ = e_2 \quad (46)$$

$$I = \frac{e_1 - V^-}{R1} ; I = \frac{V^- - V_{out}}{R2}$$

$$\Rightarrow \frac{e_1 - V^-}{R1} = \frac{V^- - V_{out}}{R2}$$

$$\Leftrightarrow R1(V^- - V_{out}) = R2(e_1 - V^-)$$

$$\Leftrightarrow R1 \cdot V^- - R1 \cdot V_{out} = R2 \cdot e_1 - R2 \cdot V^-$$

$$\Leftrightarrow (R1 + R2) \cdot V^- = R1 \cdot V_{out} + R2 \cdot e_1$$

$$V^- = \frac{R1 \cdot V_{out} + R2 \cdot e_1}{(R1 + R2)} \quad (48)$$

Montage en rétroaction positive

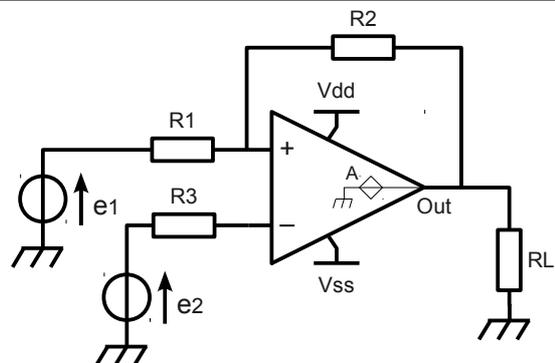


Fig. 37 : Configuration en rétroaction positive dans le cas général

$$V^- = e_2 \quad (47)$$

$$I = \frac{e_1 - V^+}{R1} ; I = \frac{V^+ - V_{out}}{R2}$$

$$\Rightarrow \frac{e_1 - V^+}{R1} = \frac{V^+ - V_{out}}{R2}$$

$$\Leftrightarrow R1(V^+ - V_{out}) = R2(e_1 - V^+)$$

$$\Leftrightarrow R1 \cdot V^+ - R1 \cdot V_{out} = R2 \cdot e_1 - R2 \cdot V^+$$

$$\Leftrightarrow (R1 + R2) \cdot V^+ = R1 \cdot V_{out} + R2 \cdot e_1$$

$$V^+ = \frac{R1 \cdot V_{out} + R2 \cdot e_1}{(R1 + R2)} \quad (49)$$

Montage en contre réaction	Montage en rétroaction positive
Avec (46) et (48), l'équation (2) devient	Avec (47) et (49), l'équation (2) devient
$V_{out} = A \cdot \left[e_2 - \frac{R1 \cdot V_{out} + R2 \cdot e_1}{(R1 + R2)} \right]$ $\Leftrightarrow V_{out}(R1 + R2) = A \cdot [e_2(R1 + R2) - R1 \cdot V_{out} - R2 \cdot e_1]$ $\Leftrightarrow V_{out}(R1 + R2) = A \cdot e_2(R1 + R2) - A \cdot R1 \cdot V_{out} - A \cdot R2 \cdot e_1$ $\Leftrightarrow V_{out}(R1 + R2) + A \cdot R1 \cdot V_{out} = A \cdot e_2(R1 + R2) - A \cdot R2 \cdot e_1$ $\Leftrightarrow V_{out}[(1 + A) \cdot R1 + R2] = A \cdot [e_2(R1 + R2) - R2 \cdot e_1]$ $\Leftrightarrow V_{out} = \frac{A \cdot [e_2(R1 + R2) - R2 \cdot e_1]}{(1 + A) \cdot R1 + R2}$ $V_{out} = \frac{A}{(1 + A) + \frac{R2}{R1}} \cdot \left[\frac{(R1 + R2)}{R1} e_2 - \frac{R2}{R1} e_1 \right] \quad (50)$	$V_{out} = A \cdot \left[\frac{R1 \cdot V_{out} + R2 \cdot e_1}{(R1 + R2)} - e_2 \right]$ $\Leftrightarrow V_{out}(R1 + R2) = A \cdot [R1 \cdot V_{out} + R2 \cdot e_1 - e_2(R1 + R2)]$ $\Leftrightarrow V_{out}(R1 + R2) = A \cdot R1 \cdot V_{out} + A \cdot R2 \cdot e_1 - A \cdot e_2(R1 + R2)$ $\Leftrightarrow V_{out}(R1 + R2) - A \cdot R1 \cdot V_{out} = A \cdot R2 \cdot e_1 - A \cdot e_2(R1 + R2)$ $\Leftrightarrow V_{out}[(1 - A) \cdot R1 + R2] = A \cdot [R2 \cdot e_1 - e_2(R1 + R2)]$ $\Leftrightarrow V_{out} = \frac{A \cdot [R2 \cdot e_1 - e_2(R1 + R2)]}{(1 - A) \cdot R1 + R2}$ $V_{out} = \frac{A}{(1 - A) + \frac{R2}{R1}} \cdot \left[\frac{R2}{R1} e_1 - \frac{(R1 + R2)}{R1} e_2 \right] \quad (51)$
Le gain statique différentiel A tend vers l'infini, il donc est très grand devant 1 et devant le ratio $R2/R1$.	Le gain statique différentiel A tend vers l'infini, il donc est très grand devant 1 et devant le ratio $R2/R1$.
$A \gg 1 \Rightarrow (1 + A) \approx A$ $\Rightarrow (1 + A) + \frac{R2}{R1} \approx A + \frac{R2}{R1}$ $A \gg \frac{R2}{R1} \Rightarrow A + \frac{R2}{R1} \approx A$	$A \gg 1 \Rightarrow (1 - A) \approx -A$ $\Rightarrow (1 - A) + \frac{R2}{R1} \approx -A + \frac{R2}{R1}$ $A \gg \frac{R2}{R1} \Rightarrow -A + \frac{R2}{R1} \approx -A$
Par conséquent, (50) se simplifie pour devenir :	Par conséquent, (51) se simplifie pour devenir :
$V_{out} \approx \frac{(R1 + R2)}{R1} e_2 - \frac{R2}{R1} e_1 \quad (52)$	$V_{out} \approx - \left[\frac{R2}{R1} e_1 - \frac{(R1 + R2)}{R1} e_2 \right]$
$\Leftrightarrow V_{out} \approx \left(1 + \frac{R2}{R1} \right) e_2 - \frac{R2}{R1} e_1 \quad (54)$	$V_{out} \approx \frac{(R1 + R2)}{R1} e_2 - \frac{R2}{R1} e_1 \quad (53)$
$\Leftrightarrow V_{out} \approx \frac{R2}{R1} (e_2 - e_1) + e_2 \quad (56)$	$\Leftrightarrow V_{out} \approx \left(1 + \frac{R2}{R1} \right) e_2 - \frac{R2}{R1} e_1 \quad (55)$
$\Leftrightarrow V_{out} \approx \frac{R2}{R1} (e_2 - e_1) + e_2 \quad (57)$	$\Leftrightarrow V_{out} \approx \frac{R2}{R1} (e_2 - e_1) + e_2 \quad (57)$
Le gain en boucle fermée est donc :	Les équations (53), (55) et (57) sont respectivement identiques aux équations (52), (54) et (57).
$G = \frac{R2}{R1}$	Le modèle classique de l'amplificateur idéal, dans lequel la tension de sortie est régie par l'équation (2), est incapable de distinguer le montage en rétroaction positive du montage en contre réaction. Dans les deux cas, le modèle se comportera comme si l'amplificateur était en contre réaction. Ce modèle est donc inapte à rendre compte de la dérive rapide de la tension de sortie vers l'un des rails d'alimentation que l'on observe avec un amplificateur réel bouclé en rétroaction positive.
Quand $e_1 = 0$ alors (54) devient	
$V_{out} \approx \left(1 + \frac{R2}{R1} \right) e_2$	
C'est le montage non-inverseur classique.	
Quand $e_2 = 0$ alors (54) devient	
$V_{out} \approx - \frac{R2}{R1} e_1$	
C'est le montage inverseur classique.	Un meilleur modèle doit être trouvé.

Annexe 4)

Amplificateur opérationnel simplifié

L'annexe A 4.1) détaille les calculs qui ont permis d'établir la fonction de transfert, l'équation temporelle, la réponse impulsionnelle et la réponse indicielle de l'amplificateur opérationnel simplifié non-bouclé.

L'annexe A 4.4) démontre que le modèle d'amplificateur opérationnel simplifié vérifie les propriétés énoncées au §V.3.3)ii.

Des éléments essentiels permettant l'implémentation du modèle de l'amplificateur opérationnel simplifié sont rassemblés aux annexes A 4.5) et A 4.6)

A 4.1) Amplificateur non-bouclé : détails des calculs

Compte-tenu des hypothèses énoncées au chapitre V.3.1, la réponse en fréquence de l'amplificateur opérationnel simplifié peut être approximée par une fonction de transfert du premier ordre avec un pôle réel. L'équation (8) et la figure 76, rappelées ici, donne respectivement l'expression mathématique et le gabarit typique d'une fonction de transfert du premier ordre.

Fonction de transfert de l'amplificateur opérationnel simplifié :

$$\forall p \in \mathbb{C}, H(p) = \frac{A}{1 + \tau p} \quad (8)$$

Pour le modèle d'amplificateur opérationnel simplifié, le coefficient A correspond au gain statique différentiel de l'amplificateur. La constante de temps τ se calcule à partir de la fréquence de coupure à -3 dB f_c , qui est un paramètre caractéristique de l'amplificateur, selon l'équation (10). La fréquence de coupure étant une grandeur réelle positive, la constante de temps est réelle et positive.

$$\tau = \frac{1}{2\pi f_c} \quad (10)$$

L'équation (11) rappelée ci-dessous exprime, à l'aide de la fonction de transfert H , la relation dans le domaine de Laplace entre l'entrée V_{INNS} et la fonction Y d'étude de la tension de sortie (cf. définition 9).

$$\forall p \in \mathbb{C}, Y(p) = H(p) \cdot V_{INNS}(p) \quad (11)$$

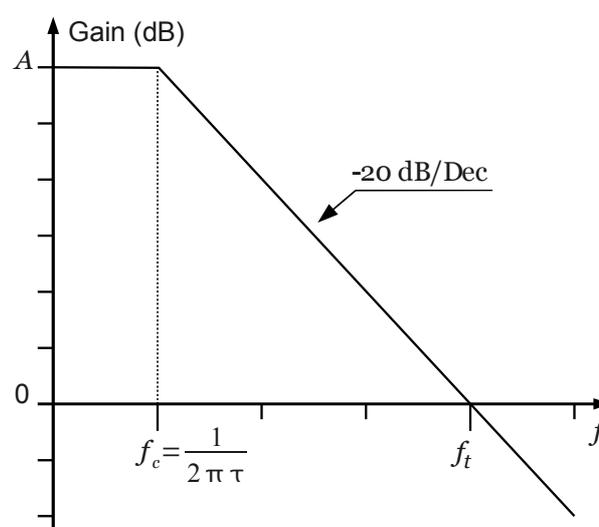


Fig. 76 : Gabarit du gain en fréquence d'un système du premier ordre

A 4.1.1) Équation temporelle

Pour établir l'équation temporelle, $H(p)$ est remplacée dans (11) en utilisant (8), ce qui donne $\forall p \in \mathbb{C}$,

$$Y(p) = \frac{A}{1 + \tau p} \cdot VINNS(p)$$

$$\Leftrightarrow Y(p) + \tau p \cdot Y(p) = A \cdot VINNS(p) \quad (12)$$

Le théorème de la dérivation dans le domaine de Laplace spécifie que pour toute fonction f réelle, causale et de transformée de Laplace F , la transformée de Laplace de la dérivée de f vaut :

$$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}, F(p) = TL[f(t)]$$

$$TL\left\{\frac{df(t)}{dt}\right\} = p \cdot F(p) - f(0)$$

En appliquant ce théorème à la fonction y , on obtient

$$\forall t \in \mathbb{R}, \forall p \in \mathbb{C}, Y(p) = TL[y(t)]$$

$$TL\left\{\frac{dy(t)}{dt}\right\} = p \cdot Y(p) - y(0) \quad (58)$$

Lemme 1 Conditions initiales

Le respect des conditions initiales données à la définition 6 implique $y(0) = 0$.

Preuve

D'après la définition 9 :

$$\forall t \in \mathbb{R}, y(t) = VoutNS(t) - Vbias$$

$$y(0) = VoutNS(0) - Vbias$$

$VoutNS(0)$ correspond à la valeur initiale de la tension de sortie V_{O_0} . Or, la définition 6 spécifie que V_{O_0} est égale à $Vbias$, d'où :

$$y(0) = V_{O_0} - Vbias = Vbias - Vbias = 0$$

Il découle du Lemme 1 et de (58) que :

$$TL\left\{\frac{dy(t)}{dt}\right\} = p \cdot Y(p) \quad (59)$$

Grâce à l'équation (59), la transformée de Laplace de la dérivée de y est identifiée dans (12) et finalement la transformée de Laplace inverse de (11) donne :

$$\tau \cdot \frac{d y(t)}{dt} + y(t) = A \cdot VinNS(t)$$

L'équation temporelle générale est obtenue à l'aide de la définition de y .

$$\tau \cdot \frac{d\{VoutNS(t) - Vbias\}}{dt} + VoutNS(t) - Vbias = A \cdot VinNS(t)$$

$$\text{Équation temporelle : } \boxed{\forall t \in \mathbb{R}, \tau \cdot \frac{d VoutNS(t)}{dt} + VoutNS(t) = A \cdot VinNS(t) + Vbias} \quad (13)$$

A 4.1.2) Réponse impulsionnelle

Pour le calcul de la réponse impulsionnelle, une impulsion de Dirac, notée δ , est appliquée à l'entrée, c'est-à-dire que l'on pose $VinNS(t) = \delta$. La transformée de Laplace d'une impulsion valant la valeur constante 1, on a $VINNS(p) = 1$. L'équation (11) devient donc ici :

$$\begin{aligned}
 Y(p) &= H(p) \\
 \Leftrightarrow Y(p) &= \frac{A}{1+\tau p} \\
 \Leftrightarrow Y(p) &= \frac{A}{\tau} \cdot \frac{1}{p+\frac{1}{\tau}}
 \end{aligned}$$

On reconnaît ici la transformée de Laplace de la fonction :

$$y(t) = \frac{A}{\tau} \cdot e^{-\frac{t}{\tau}}$$

La réponse impulsionnelle est obtenue à l'aide de la définition de y .

$$V_{outNS}(t) - V_{bias} = \frac{A}{\tau} \cdot e^{-\frac{t}{\tau}}$$

Réponse impulsionnelle : $\forall t \in \mathbb{R}, V_{outNS}(t) = \frac{A}{\tau} \cdot e^{-\frac{t}{\tau}} + V_{bias}$ (14)

A 4.1.3) Réponse indicielle

Pour le calcul de la réponse indicielle, le signal d'entrée est un échelon d'amplitude Ech . En notant $u(t)$ la fonction d'Heaviside*, le signal d'entrée est alors de la forme :

$$\forall t \in \mathbb{R}, V_{inNS}(t) = Ech \cdot u(t)$$

Sa transformée de Laplace est alors

$$\forall p \in \mathbb{C}, V_{INNS}(p) = \frac{Ech}{p}$$

L'équation (11) devient :

$$\begin{aligned}
 Y(p) &= H(p) \cdot \frac{Ech}{p} \\
 \Leftrightarrow Y(p) &= \frac{Ech}{p} \cdot \frac{A}{1+\tau p}
 \end{aligned}$$

Pour continuer le calcul, $Y(p)$ doit être décomposée en éléments simples. Soit à déterminer deux réels a et b tels que :

$$Y(p) = \frac{a}{p} + \frac{b}{1+\tau p}$$

d'où :

$$Y(p) = \frac{Ech}{p} \cdot \frac{A}{(1+\tau p)} = \frac{a}{p} + \frac{b}{1+\tau p}$$

Calcul de a :

$$\begin{cases}
 p \cdot Y(p) = p \cdot \frac{Ech}{p} \cdot \frac{A}{(1+\tau p)} = p \cdot \frac{a}{p} + p \cdot \frac{b}{1+\tau p} \\
 p \cdot Y(p) = \frac{A \cdot Ech}{1+\tau p} = a + p \cdot \frac{b}{1+\tau p} \\
 p=0 \Rightarrow a = A \cdot Ech
 \end{cases}$$

* voir glossaire

Calcul de b :

$$\begin{cases} (1+\tau p) \cdot Y(p) = (1+\tau p) \cdot \frac{Ech}{p} \cdot \frac{A}{(1+\tau p)} = (1+\tau p) \cdot \frac{a}{p} + (1+\tau p) \cdot \frac{b}{(1+\tau p)} \\ (1+\tau p) \cdot Y(p) = \frac{A \cdot Ech}{p} = (1+\tau p) \cdot \frac{a}{p} + b \\ p = -\frac{1}{\tau} \Rightarrow b = -A \cdot Ech \cdot \tau \end{cases}$$

Par conséquent, avec $\begin{cases} a = A \cdot Ech \\ b = -A \cdot Ech \cdot \tau \end{cases}$:

$$\begin{aligned} Y(p) &= A \cdot Ech \cdot \left[\frac{1}{p} - \frac{\tau}{1+\tau p} \right] \\ \Leftrightarrow Y(p) &= A \cdot Ech \cdot \left[\frac{1}{p} - \frac{1}{p + \frac{1}{\tau}} \right] \end{aligned}$$

On reconnaît ici la transformée de Laplace de la fonction :

$$\begin{aligned} y(t) &= A \cdot Ech \cdot \left[u(t) - e^{-\frac{t}{\tau}} \cdot u(t) \right] \\ \Leftrightarrow y(t) &= A \cdot Ech \cdot \left[1 - e^{-\frac{t}{\tau}} \right] \cdot u(t) \\ \Leftrightarrow \forall t > 0, \quad y(t) &= A \cdot Ech \cdot \left[1 - e^{-\frac{t}{\tau}} \right] \end{aligned}$$

La réponse indicielle est obtenue à l'aide de la définition de y .

$$V_{outNS}(t) - V_{bias} = A \cdot Ech \cdot \left[1 - e^{-\frac{t}{\tau}} \right]$$

$$\text{Réponse indicielle : } \boxed{\forall t \in \mathbb{R}, V_{outNS}(t) = A \cdot Ech \cdot \left[1 - e^{-\frac{t}{\tau}} \right] + V_{bias}} \quad (15)$$

A 4.2) Montage en contre-réaction : détail des calculs

Afin de faciliter la compréhension, le lecteur est invité à se reporter à la figure 36. Le calcul des tensions appliquées sur les entrées de l'amplificateur en contre-réaction s'obtient avec un raisonnement similaire à celui tenu à l'annexe 3 pour établir les équations (46) et (48). Cependant, la tension de sortie étant centrée autour de V_{bias} , V_{out} doit être remplacée par la fonction y (cf. Définition 9). Les tensions appliquées en entrée du montage étant toujours désignées par e_1 et e_2 , un tel raisonnement conduit à :

$$\forall t \in \mathbb{R}, \begin{cases} V^+(t) = e_2(t) \\ V^-(t) = \frac{R1 \cdot y(t) + R2 \cdot e_1(t)}{(R1 + R2)} \end{cases}$$

L'expression de la tension différentielle d'entrée de l'amplificateur en contre-réaction en découle directement.

$$\begin{aligned} \Rightarrow V_{inNS}(t) &= V^+(t) - V^-(t) \\ \Leftrightarrow V_{inNS}(t) &= e_2(t) - \frac{R1 \cdot y(t) + R2 \cdot e_1(t)}{(R1 + R2)} \\ \forall t \in \mathbb{R}, \quad V_{inNS}(t) &= e_2(t) - \frac{R2 \cdot e_1(t)}{(R1 + R2)} - \frac{R1}{(R1 + R2)} \cdot y(t) \end{aligned} \quad (16)$$

Soient E_1 et E_2 les transformées de Laplace de e_1 et e_2

$$\begin{aligned}\forall t \in \mathbb{R}, \\ E_1(p) &= TL[e_1(t)] \\ E_2(p) &= TL[e_2(t)]\end{aligned}$$

Avec E_1 et E_2 , la transformée de Laplace de (16) donne :

$$\forall p \in \mathbb{C}, \text{VINNS}(p) = E_2(p) - \frac{R2 \cdot E_1(p)}{(R1+R2)} - \frac{R1}{(R1+R2)} \cdot Y(p) \quad (17)$$

$\text{VINNS}(p)$ est remplacée dans (12) (rappelée ci-dessous) avec la valeur trouvée dans (17).

$$\begin{aligned}\forall p \in \mathbb{C}, Y(p) + \tau p \cdot Y(p) &= A \cdot \text{VINNS}(p) \quad (12) \\ \Leftrightarrow Y(p) + \tau p \cdot Y(p) &= A \cdot \left[E_2(p) - \frac{R2 \cdot E_1(p)}{(R1+R2)} \right] - \frac{A \cdot R1}{(R1+R2)} \cdot Y(p) \\ \Leftrightarrow Y(p) \cdot \left[1 + \tau p + \frac{A \cdot R1}{(R1+R2)} \right] &= A \cdot \left[E_2(p) - \frac{R2 \cdot E_1(p)}{(R1+R2)} \right] \\ \Leftrightarrow Y(p) \cdot \left[\frac{(R1+R2)}{R1} \cdot (1 + \tau p) + A \right] &= A \cdot \left[\frac{(R1+R2)}{R1} \cdot E_2(p) - \frac{R2}{R1} \cdot E_1(p) \right] \\ \Leftrightarrow Y(p) &= \frac{A}{\left[\frac{(R1+R2)}{R1} \cdot (1 + \tau p) + A \right]} \cdot \left[\frac{(R1+R2)}{R1} \cdot E_2(p) - \frac{R2}{R1} \cdot E_1(p) \right] \quad (18)\end{aligned}$$

On pose :

$$\forall p \in \mathbb{C}, \text{VSIG}(p) = \frac{(R1+R2)}{R1} \cdot E_2(p) - \frac{R2}{R1} \cdot E_1(p) \quad (19)$$

$\text{VSIG}(p)$ s'interprète comme la transformée de Laplace du signal différentiel appliqué à l'entrée du montage. La fonction de transfert $Hcr(p)$ du montage en contre-réaction est donc déterminée par :

$$\forall p \in \mathbb{C}, Y(p) = Hcr(p) \cdot \text{VSIG}(p) \quad (20)$$

Fonction de transfert du montage en contre-réaction :

$$\boxed{\forall p \in \mathbb{C}, Hcr(p) = \frac{A}{\frac{(R1+R2)}{R1} \cdot (1 + \tau p) + A}} \quad (21)$$

$$\begin{aligned}\Leftrightarrow Hcr(p) &= \frac{A}{\frac{(R1+R2)}{R1}} \cdot \frac{1}{\left\{ \tau p + 1 + \frac{A \cdot R1}{(R1+R2)} \right\}} \\ \Leftrightarrow Hcr(p) &= \frac{A \cdot R1}{(R1+R2) \tau} \cdot \frac{1}{\left\{ p + \frac{1}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1+R2)} \right] \right\}}\end{aligned}$$

La fonction de transfert $Hcr(p)$ a un pôle p_1 .

$$p_1 = -\frac{1}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1+R2)} \right]$$

i) Réponse transitoire

Lemme 2 La fonction $VoutNS$ vérifie la propriété 3

Le modèle d'amplificateur avec un produit gain-bande fini mais sans saturation ni limitation de la vitesse de variation, permet de réaliser un montage en contre-réaction dont la réponse est asymptotiquement stable et apériodique.

Preuve

La stabilité du système est donnée par le signe de la partie réelle du pôle p_I .

τ est une constante réelle positive (cf. (10) au V.3.4)ii. Dans le montage étudié ici, R_1 et R_2 sont des résistances pures, ce sont donc des grandeurs réelles positives. Le gain statique différentiel A est réel et positif. Par conséquent, p_I est un pôle réel négatif.

Avec le modèle de l'amplificateur opérationnel simplifié, **la réponse du montage en contre réaction** est donc **asymptotiquement stable et apériodique**, ce qui est conforme à ce qui à la réalité.

ii) Réponse indicielle du montage en contre réaction

La réponse indicielle du montage est obtenue en appliquant un échelon de tension sur chacune des entrées.

$$\forall t \in \mathbb{R}, e_1(t) = Ech_1 \cdot u(t) ; e_2(t) = Ech_2 \cdot u(t)$$

Leurs transformées de Laplace respectives sont :

$$\forall p \in \mathbb{C}, E_1(p) = \frac{Ech_1}{p} ; E_2(p) = \frac{Ech_2}{p}$$

L'équation (20) devient alors :

$$\forall p \in \mathbb{C},$$

$$Y(p) = \frac{A}{\left[\frac{(R_1 + R_2)}{R_1} \cdot (1 + \tau p) + A \right]} \cdot \left[\frac{(R_1 + R_2)}{R_1} \cdot \frac{Ech_2}{p} - \frac{R_2}{R_1} \cdot \frac{Ech_1}{p} \right]$$

$$\Leftrightarrow Y(p) = \frac{A}{\frac{(R_1 + R_2)}{R_1}} \cdot \frac{1}{\left\{ \tau p + 1 + \frac{A \cdot R_1}{(R_1 + R_2)} \right\}} \cdot \left[\frac{(R_1 + R_2)}{R_1} \cdot \frac{Ech_2}{p} - \frac{R_2}{R_1} \cdot \frac{Ech_1}{p} \right]$$

$$\Leftrightarrow Y(p) = \frac{1}{p} \cdot \frac{1}{\left\{ \tau p + 1 + \frac{A \cdot R_1}{(R_1 + R_2)} \right\}} \cdot \left\{ \frac{A \cdot R_1}{(R_1 + R_2)} \cdot \left[\frac{(R_1 + R_2)}{R_1} \cdot Ech_2 - \frac{R_2}{R_1} \cdot Ech_1 \right] \right\}$$

Pour simplifier les écritures et décomposer $Y(p)$ en éléments simples, on pose :

$$\Gamma = \left\{ \frac{A \cdot R_1}{(R_1 + R_2)} \cdot \left[\frac{(R_1 + R_2)}{R_1} \cdot Ech_2 - \frac{R_2}{R_1} \cdot Ech_1 \right] \right\}$$

$$Y(p) = \frac{1}{p} \cdot \frac{1}{\left[\tau p + 1 + \frac{A \cdot R_1}{(R_1 + R_2)} \right]} \cdot \Gamma = \left\{ \frac{a}{p} + \frac{b}{\left[\tau p + 1 + \frac{A \cdot R_1}{(R_1 + R_2)} \right]} \right\} \cdot \Gamma$$

$$\left| \begin{array}{l} p \cdot Y(p) = \frac{1}{\left[\tau p + 1 + \frac{A \cdot R_1}{(R_1 + R_2)} \right]} \cdot \Gamma = \left\{ a + \frac{b \cdot p}{\left[\tau p + 1 + \frac{A \cdot R_1}{(R_1 + R_2)} \right]} \right\} \cdot \Gamma \\ p=0 \Rightarrow \frac{1}{1 + \frac{A \cdot R_1}{(R_1 + R_2)}} = a \\ p=0 \Rightarrow a = \frac{(R_1 + R_2)}{(R_1 + R_2) + A \cdot R_1} \end{array} \right.$$

$$\left\{ \begin{array}{l} \left[\tau p + 1 + \frac{A \cdot R1}{(R1 + R2)} \right] \cdot Y(p) = \frac{1}{p} \cdot \Gamma = \left\{ \frac{a \cdot \left[\tau p + 1 + \frac{A \cdot R1}{(R1 + R2)} \right]}{p} + b \right\} \cdot \Gamma \\ p = -\frac{1}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1 + R2)} \right] \Rightarrow \frac{-\tau}{1 + \frac{A \cdot R1}{(R1 + R2)}} = b \\ p = -\frac{1}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1 + R2)} \right] \Rightarrow b = -\frac{\tau \cdot (R1 + R2)}{(R1 + R2) + A \cdot R1} \end{array} \right.$$

d'où

$$Y(p) = \left\{ \frac{1}{p} - \frac{\tau}{\left[\tau p + 1 + \frac{A \cdot R1}{(R1 + R2)} \right]} \right\} \cdot \frac{(R1 + R2)}{(R1 + R2) + A \cdot R1} \cdot \Gamma$$

soit encore en remplaçant Γ par sa valeur.

$$Y(p) = \left\{ \frac{1}{p} - \frac{\tau}{\left[\tau p + 1 + \frac{A \cdot R1}{(R1 + R2)} \right]} \right\} \cdot \frac{A \cdot R1}{(R1 + R2) + A \cdot R1} \cdot \left[\frac{(R1 + R2)}{R1} \cdot Ech_2 - \frac{R2}{R1} \cdot Ech_1 \right]$$

$$Y(p) = \left\{ \frac{1}{p} - \frac{1}{p + \frac{1}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1 + R2)} \right]} \right\} \cdot \frac{A \cdot R1}{(R1 + R2) + A \cdot R1} \cdot \left[\frac{(R1 + R2)}{R1} \cdot Ech_2 - \frac{R2}{R1} \cdot Ech_1 \right]$$

Par transformée de Laplace inverse, on déduit :

$$\begin{aligned} y(t) &= \left[u(t) - \left(e^{-\frac{t}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1 + R2)} \right]} \right) \cdot u(t) \right] \cdot \frac{A \cdot R1}{(R1 + R2) + A \cdot R1} \cdot \left[\frac{(R1 + R2)}{R1} \cdot Ech_2 - \frac{R2}{R1} \cdot Ech_1 \right] \\ \Leftrightarrow y(t) &= u(t) \cdot \left[1 - e^{-\frac{t}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1 + R2)} \right]} \right] \cdot \frac{A \cdot R1}{(R1 + R2) + A \cdot R1} \cdot \left[\frac{(R1 + R2)}{R1} \cdot Ech_2 - \frac{R2}{R1} \cdot Ech_1 \right] \\ \Leftrightarrow \forall t > 0, y(t) &= \left[1 - e^{-\frac{t}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1 + R2)} \right]} \right] \cdot \frac{1}{\frac{(R1 + R2)}{A \cdot R1} + 1} \cdot \left[\frac{(R1 + R2)}{R1} \cdot Ech_2 - \frac{R2}{R1} \cdot Ech_1 \right] \end{aligned}$$

La réponse indicielle du montage en contre réaction est obtenue à l'aide de la définition de y .

Réponse indicielle du montage en contre réaction

$$\forall t \in \mathbb{R}, V_{outNS}(t) = \left[1 - e^{-\frac{t}{\tau} \cdot \left[1 + \frac{A \cdot R1}{(R1 + R2)} \right]} \right] \cdot \frac{1}{\frac{(R1 + R2)}{A \cdot R1} + 1} \cdot \left[\frac{(R1 + R2)}{R1} \cdot Ech_2 - \frac{R2}{R1} \cdot Ech_1 \right] + V_{bias} \quad (22)$$

Le rapport des résistances $\frac{R2}{R1}$ est toujours choisi de façon à être petit devant A , le gain statique différentiel en tension de l'amplificateur. Dans la pratique $\frac{R2}{R1} \leq 1000 \Rightarrow \frac{(R1 + R2)}{R1} \leq 1001$ tandis que $A \geq 10^4$, par conséquent :

$$1 + \frac{A \cdot R1}{(R1 + R2)} \geq 1 + \frac{10^4}{1001} \Leftrightarrow 1 + \frac{A \cdot R1}{(R1 + R2)} \geq 10$$

Avec le montage en contre réaction le transitoire sera plus de dix fois plus court que dans le cas de l'amplificateur non bouclé.

En outre puisque A , $R1$ et $R2$ sont positifs :

$$\begin{cases} 0 < \frac{(RI + R2)}{RI} \leq 1001 \\ A \geq 10^4 \Leftrightarrow 0 < \frac{1}{A} \leq \frac{1}{10^4} \end{cases}$$

d'où

$$\begin{aligned} 0 &< \frac{(RI + R2)}{A \cdot RI} \leq \frac{1001}{10^4} \\ \Leftrightarrow 1 &< \frac{(RI + R2)}{A \cdot RI} + 1 \leq \frac{1001}{10^4} + 1 \\ \Leftrightarrow 1 &< \frac{(RI + R2)}{A \cdot RI} + 1 \leq \frac{1001 + 10^4}{10^4} \\ \Leftrightarrow \frac{10^4}{1001 + 10^4} &\leq \frac{1}{\frac{(RI + R2)}{A \cdot RI} + 1} < 1 \end{aligned}$$

Dans le cas le plus défavorable pour lequel $A = 10^4$ et $\frac{R2}{RI} = 1000$ on a :

$$\frac{1}{\left[\frac{(RI + R2)}{A \cdot RI} + 1 \right]} = \frac{10^4}{1001 + 10^4} \approx 0,91$$

Avec $A = 10^5$, ce qui est courant de nos jours, le rapport ci-dessus a alors pour valeur

$$\frac{1}{\left[\frac{(RI + R2)}{A \cdot RI} + 1 \right]} = \frac{10^5}{1001 + 10^5} \approx 0,99$$

L'approximation suivante est donc acceptable :

$$\frac{1}{\left[\frac{(RI + R2)}{A \cdot RI} + 1 \right]} \approx 1$$

Par conséquent :

$$\lim_{t \rightarrow \infty} V_{outNS}(t) = \frac{(RI + R2)}{RI} \cdot Ech_2 - \frac{R2}{RI} \cdot Ech_1 + V_{bias} \quad (23)$$

$$\Leftrightarrow \lim_{t \rightarrow \infty} V_{outNS}(t) = \frac{R2}{RI} \cdot [Ech_2 - Ech_1] + Ech_2 + V_{bias} \quad (24)$$

Les valeurs limites (23) et (24), qui correspondent à la tension de sortie en régime établi, sont à rapprocher respectivement des équations (54) et (56) de la tension de sortie du montage en contre réaction avec l'amplificateur idéal (avec $V_{bias} = 0$).

A 4.3) Montage en rétroaction positive : détail des calculs

Afin de faciliter la compréhension, le lecteur est invité à se reporter à la figure 37.

Le calcul des tensions appliquées sur les entrées de l'amplificateur en rétroaction positive s'obtient avec un raisonnement similaire à celui tenu à l'annexe 3 pour établir les équations (47) et (49). Cependant, la tension de sortie étant centrée autour de V_{bias} , V_{out} doit être remplacée par la fonction y (cf. Définition 9). Les tensions appliquées en entrée du montage étant toujours désignées par e_1 et e_2 , un tel raisonnement conduit à :

$$\forall t \in \mathbb{R}, \begin{cases} V^+(t) = \frac{R1 \cdot y(t) + R2 \cdot e_1(t)}{(R1 + R2)} \\ V^-(t) = e_2(t) \end{cases}$$

L'expression de la tension différentielle d'entrée de l'amplificateur en contre-réaction en découle directement.

$$\begin{aligned} \Rightarrow VinNS(t) &= V^+(t) - V^-(t) \\ \Leftrightarrow VinNS(t) &= \frac{R1 \cdot y(t) + R2 \cdot e_1(t)}{(R1 + R2)} - e_2(t) \\ \forall t \in \mathbb{R}, VinNS(t) &= \frac{R2 \cdot e_1(t)}{(R1 + R2)} - e_2(t) + \frac{R1}{(R1 + R2)} \cdot y(t) \end{aligned} \quad (26)$$

Soient E_1 et E_2 les transformées de Laplace de e_1 et e_2

$$\begin{aligned} \forall t \in \mathbb{R}, \\ E_1(p) &= TL[e_1(t)] \\ E_2(p) &= TL[e_2(t)] \end{aligned}$$

Avec E_1 et E_2 , la transformée de Laplace de (26) donne :

$$\forall p \in \mathbb{C}, VINNS(p) = \frac{R2 \cdot E_1(p)}{(R1 + R2)} - E_2(p) + \frac{R1}{(R1 + R2)} \cdot Y(p) \quad (27)$$

$VINNS(p)$ est remplacée dans (12) (rappelée ci-dessous) avec la valeur trouvée dans (27).

$$\begin{aligned} \forall p \in \mathbb{C}, Y(p) + \tau p \cdot Y(p) &= A \cdot VINNS(p) \\ \Leftrightarrow Y(p) + \tau p \cdot Y(p) &= A \cdot \left[\frac{R2 \cdot E_1(p)}{(R1 + R2)} - E_2(p) \right] + \frac{A \cdot R1}{(R1 + R2)} \cdot Y(p) \\ \Leftrightarrow Y(p) \cdot \left[1 + \tau p - \frac{A \cdot R1}{(R1 + R2)} \right] &= A \cdot \left[\frac{R2 \cdot E_1(p)}{(R1 + R2)} - E_2(p) \right] \\ \Leftrightarrow Y(p) &= \frac{A}{\left[\frac{(R1 + R2)}{R1} \cdot (1 + \tau p) - A \right]} \cdot \left[\frac{R2}{R1} \cdot E_1(p) - \frac{(R1 + R2)}{R1} \cdot E_2(p) \right] \\ \Leftrightarrow Y(p) &= \frac{A}{\left[A - \frac{(R1 + R2)}{R1} \cdot (1 + \tau p) \right]} \cdot \left[\frac{(R1 + R2)}{R1} \cdot E_2(p) - \frac{R2}{R1} \cdot E_1(p) \right] \\ \Leftrightarrow Y(p) &= \frac{A}{\left[A - \frac{(R1 + R2)}{R1} \cdot (1 + \tau p) \right]} \cdot VSIG(p) \end{aligned} \quad (12)$$

Avec $VSIG(p)$ le signal appliqué à l'entrée du montage tel que défini en (19), la fonction de transfert $Hr(p)$ du montage en rétroaction positive est donc déterminée par :

$$Y(p) = Hr(p) \cdot VSIG(p) \quad (28)$$

Fonction de transfert du montage en rétroaction positive :

$$\boxed{\forall p \in \mathbb{C}, Hr(p) = \frac{A}{A - \frac{(R1 + R2)}{R1} \cdot (1 + \tau p)}} \quad (29)$$

$$\Leftrightarrow Hr(p) = \frac{A}{\frac{(R1 + R2)}{R1}} \cdot \frac{1}{\left\{ \frac{A \cdot R1}{(R1 + R2)} - 1 - \tau p \right\}}$$

$$\Leftrightarrow Hr(p) = \frac{A \cdot RI}{(-\tau) \cdot (RI + R2)} \cdot \frac{1}{\left\{ p + \frac{1}{\tau} \cdot \left[1 - \frac{A \cdot RI}{(RI + R2)} \right] \right\}}$$

La fonction de transfert $Hr(p)$ a un pôle p_2 .

$$p_2 = -\frac{1}{\tau} \cdot \left[1 - \frac{A \cdot RI}{(RI + R2)} \right]$$

$$p_2 = \frac{1}{\tau} \cdot \left[\frac{A \cdot RI}{(RI + R2)} - 1 \right]$$

Réponse transitoire

Lemme 3 La fonction $VoutNS$ vérifie la propriété 4

Le modèle d'amplificateur avec un produit gain-bande fini mais sans saturation ni limitation de la vitesse de variation, permet de réaliser un montage en rétroaction positive dont la réponse est asymptotiquement instable et apériodique.

Preuve

La stabilité du système est donnée par le signe de la partie réelle du pôle p_2 .

τ est une constante réelle positive (cf. (10) au V.3.4)ii). Dans le montage étudié ici, $R1$ et $R2$ sont des résistances pures, ce sont donc des grandeurs réelles positives. Le gain statique différentiel A est lui aussi réel et positif. Par conséquent, p_2 est un pôle réel. La réponse du montage sera apériodique. Il faut maintenant étudié le signe de p_2 . Il est donné par le signe du terme entre crochets.

$$sign[p_2] = sign \left[\frac{A \cdot RI}{(RI + R2)} - 1 \right]$$

$$p_2 > 0 \Rightarrow \frac{A \cdot RI}{(RI + R2)} > 1$$

$R1$ et $R2$ étant des grandeurs positives, leur rapport est positif.

$$\frac{A \cdot RI}{(RI + R2)} > 1 \Leftrightarrow A > \frac{RI + R2}{RI}$$

$$\Leftrightarrow A > 1 + \frac{R2}{RI} \tag{60}$$

Selon la postulat 2, $A \geq 10^4$. Pour respecter (60), il faut :

$$\frac{R2}{RI} < 10^4 - 1 \tag{61}$$

Dans la pratique le rapport des résistances $\frac{R2}{RI}$ étant au plus de 1000, la condition posée par (61) est toujours respectée donc le pôle p_2 est toujours positif.

Le pôle p_2 étant un réel toujours positif, avec le modèle de l'amplificateur opérationnel simplifié, **la réponse du montage en rétroaction positive est donc instable et apériodique**, ce qui est conforme à ce qui est recherché.

A 4.4) Démonstration des propriétés vérifiées par le modèle de l'amplificateur opérationnel simplifié

Cette annexe prouve que le modèle d'amplificateur opérationnel simplifié vérifie les propriétés énoncées au §V.3.3)ii.

Propriété 1 Tension de sortie quand la tension différentielle d'entrée est nulle

Quand la tension d'entrée est maintenue à zéro, une fois le transitoire terminé, la tension de sortie est égale à la tension V_{bias} .

$$\forall t \in \mathbb{R}, \quad V_{in}(t) = 0 \Rightarrow \lim_{t \rightarrow +\infty} V_{out}(t) = V_{bias}$$

Preuve

La fonction V_{outNS} est décrite par l'équation différentielle (13). Quand la tension d'entrée est à zéro, l'équation (13) devient :

$$\begin{aligned} \tau \cdot \frac{d V_{outNS}(t)}{dt} + V_{outNS}(t) &= V_{bias} \\ \Leftrightarrow \tau \cdot \frac{d [V_{outNS}(t) - V_{bias}]}{dt} + V_{outNS}(t) - V_{bias} &= 0 \end{aligned}$$

En utilisant la fonction y (cf. définition 9), cette équation différentielle devient :

$$\tau \cdot \frac{d y(t)}{dt} + y(t) = 0$$

La solution de cette équation différentielle est $y(t) = e^{-\frac{t}{\tau}}$ or $\lim_{t \rightarrow \infty} e^{-\frac{t}{\tau}} = 0$ donc $\lim_{t \rightarrow \infty} y(t) = 0$.

Par conséquent $\lim_{t \rightarrow \infty} V_{outNS}(t) - V_{bias} = 0 \Leftrightarrow \lim_{t \rightarrow \infty} V_{outNS}(t) = V_{bias}$.

Pour $t = 5\tau$, $y(5\tau) = e^{-5} \Rightarrow y(5\tau) < 7 \cdot 10^{-3} \Rightarrow \{V_{outNS}(5\tau) - V_{bias}\} < 7 \cdot 10^{-3}$

Par conséquent si l'entrée est maintenue égale à zéro pour un temps d'au moins cinq fois la constante de temps τ , la tension de sortie, sans saturation ni limitation de sa vitesse de variation, est suffisamment proche de la tension de repos pour être confondue avec elle.

Premier cas : Au moment où la tension d'entrée devient nulle, la sortie n'est ni en mode de saturation ni en mode de limitation de son taux de montée.

Dans ce cas, la tension de sortie est décrite par la fonction V_{outNS} , c'est-à-dire : $V_{out} = V_{outNS}$.

Par conséquent $\lim_{t \rightarrow \infty} V_{out}(t) = V_{bias}$.

Deuxième cas : Au moment où la tension d'entrée devient nulle, la sortie ne sature pas mais sa vitesse de variation est limitée à la vitesse de balayage de l'amplificateur.

Dans ce cas, la tension de sortie est décrite par la fonction V_{outSr} qui est conditionnée par la dérivée de V_{outNS} . Comme V_{outNS} devient constante (égale à V_{bias}) quand la tension d'entrée est maintenue suffisamment longtemps à zéro, sa dérivée tend progressivement vers zéro et de ce fait celle-ci redevient nécessairement inférieure à la vitesse de balayage de l'amplificateur. L'amplificateur quitte donc le mode de limitation de la vitesse de variation de sa tension de sortie dans lequel il était, et revient dans le mode normal pour lequel $V_{out} = V_{outNS}$. On se retrouve dans le premier cas pour lequel la propriété a déjà été vérifiée.

Troisième cas : Au moment où la tension d'entrée devient nulle, la sortie est en saturation.

Dans ce cas, la tension de sortie est décrite par la fonction V_{outSat} qui est conditionnée par l'amplitude de V_{outNS} par rapport aux seuils de saturation.

Quelle est la position de la tension de repos par rapport aux seuils de saturation ? La définition 5 donne l'expression de la tension de repos V_{bias} . Cette expression n'est pas modifiée en ajoutant et retranchant V_{oLoss} .

$$\begin{aligned}
V_{bias} &= \frac{(V_{dd} - V_{ss})}{2} + V_{ss} \\
\Leftrightarrow V_{bias} &= \frac{(V_{dd} - V_{ss})}{2} + V_{ss} + V_{oLoss} - V_{oLoss} \\
\Leftrightarrow V_{bias} &= \frac{(V_{dd} - V_{ss} - 2 \cdot V_{oLoss})}{2} + V_{ss} + V_{oLoss} \\
\Leftrightarrow V_{bias} &= \frac{([V_{dd} - V_{oLoss}] - [V_{ss} + V_{oLoss}])}{2} + V_{ss} + V_{oLoss}
\end{aligned}$$

On reconnaît l'expression des seuils de saturation V_{oSatL} et V_{oSatH} (cf. Définition 11 et Définition 12). D'où :

$$V_{bias} = \frac{(V_{oSatH} - V_{oSatL})}{2} + V_{oSatL} \quad (62)$$

La tension de repos V_{bias} est au point milieu entre les seuils de saturation.

Si la sortie est en saturation alors V_{outNS} est au delà des seuils de saturation. Quand la tension d'entrée devient nulle, V_{outNS} devenant égale à V_{bias} et V_{bias} étant au point milieu entre les seuils de saturation, l'amplificateur sort nécessairement du mode de saturation puisque V_{outNS} rentre entre les seuils de saturation.

Si la vitesse à laquelle V_{outNS} devient égale à V_{bias} est supérieure à la vitesse de balayage de l'amplificateur, la sortie passera en mode avec limitation de sa vitesse de variation sinon elle passera en mode normal sans saturation ni limitation de la vitesse de variation. Pour ces deux modes la propriété 1 a déjà vérifiée.

Conclusion : la propriété 1 est vérifiée en toute circonstance.

Propriété 2 Comportement en absence de boucle de rétroaction

En absence de boucle de rétroaction, l'amplificateur opérationnel simplifié doit se comporter comme un comparateur. Une très faible tension d'entrée provoque la saturation de la sortie.

Preuve

Cette propriété n'a de sens que si la sortie n'est pas déjà saturée.

Cette propriété s'évalue en appliquant un échelon en entrée. La réponse indicielle de l'amplificateur opérationnel simplifié est donnée par l'équation (15).

$$\begin{aligned}
\forall t \in \mathbb{R}, V_{out}(t) &= A \cdot Ech \cdot \left[1 - e^{-\frac{t}{\tau}} \right] + V_{bias} \\
\lim_{t \rightarrow \infty} V_{out}(t) &= A \cdot Ech + V_{bias}
\end{aligned}$$

L'amplitude minimale d'un échelon $EchSatH$ provoquant la saturation de la sortie de l'amplificateur par rapport au seuil de saturation supérieur est calculée de la façon suivante :

$$\begin{aligned}
V_{oSatH} &= A \cdot EchSatH + V_{bias} \\
\Leftrightarrow V_{oSatH} - V_{bias} &= A \cdot EchSatH \\
\Leftrightarrow EchSatH &= \frac{V_{oSatH} - V_{bias}}{A}
\end{aligned}$$

Plus l'excursion de tension est importante et le gain statique différentiel en tension est faible, plus l'amplitude minimale sera importante. Dans le cas d'un AOP LM741C alimenté en $\pm 15V$, le seuil de saturation supérieur est de 13V et le gain statique différentiel²⁰ de 15000. Ceci donne un échelon minimal saturant $EchSatH$ de 866 μV . Dans un montage réel, cette valeur serait dans le bruit. Pour résister au bruit, les comparateurs intégrés, tels ceux de la série LT6700, ont un hystérésis de plusieurs millivolts et souvent de plusieurs dizaines de millivolts. La démonstration faite ici par rapport au seuil de saturation supérieur peut être conduite de façon similaire par rapport au seuil de saturation inférieur.

En absence de boucle de rétroaction, l'amplificateur opérationnel simplifié se comporte comme un comparateur, la propriété 2 est donc vérifiée.

²⁰ La valeur de 10000 indiquée précédemment est obtenue pour une alimentation de $\pm 5V$

Propriété 3 Forme de la réponse du montage en contre réaction

La réponse du montage en contre réaction doit être asymptotiquement stable et apériodique.

Preuve

La stabilité d'un montage s'évalue par rapport à la réponse impulsionnelle du montage. Une impulsion de Dirac présente un taux de montée et de descente théoriquement infinie donc très supérieur à la vitesse de balayage de l'amplificateur. Lors du front montant de l'impulsion, la sortie commencera à évoluer avec une vitesse de variation limitée à la vitesse de balayage puis, lors du front descendant de l'impulsion, elle inversera son évolution une vitesse de variation limitée à la vitesse de balayage de signe opposé. Du fait de la brièveté de l'impulsion et de la limitation de la vitesse de variation, la sortie n'aura pas le temps de s'éloigner beaucoup de sa tension de repos V_{bias} . Elle n'atteindra jamais un seuil de saturation durant l'impulsion. Une fois l'impulsion terminée, le signal appliqué à l'entrée est maintenu indéfiniment à zéro. La sortie quittera donc le mode de limitation de la vitesse de variation, comme cela a été démontré lors de la vérification de la propriété 1, et elle reviendra dans le mode normal sans limitation de la vitesse de variation ni saturation. Dans le mode normal, $V_{out} = V_{outNS}$ or le lemme 2 a prouvé que la fonction V_{outNS} vérifie la propriété 3.

Par conséquent, la fonction V_{out} vérifie la propriété 3.

Propriété 4 Forme de la réponse du montage en rétroaction positive

La réponse du montage en rétroaction positive doit être instable et apériodique.

Preuve

Pour les mêmes raisons que celles évoquées pour la propriété 3, l'amplificateur opérationnel simplifié soumis à une impulsion évolue brièvement en mode avec limitation de la vitesse de variation puis revient dans le mode normal sans limitation de la vitesse de variation ni saturation. Sa sortie ne passe jamais en saturation. Dans le mode normal, $V_{out} = V_{outNS}$ or le lemme 3 a prouvé que la fonction V_{outNS} vérifie la propriété 4. Par conséquent, la fonction V_{out} vérifie la propriété 4.

Propriété 5 Tension de sortie bornée par les seuils de saturation

La tension de sortie de l'amplificateur est toujours comprise entre le seuil de saturation positive V_{oSatH} et le seuil de saturation négative V_{oSatL} .

$$V_{oSatL} \leq V_{out} \leq V_{oSatH}$$

Preuve

De part sa construction même (cf. définition 20), la fonction V_{out} est bornée par V_{oSatL} et V_{oSatH} .

Propriété 6 Vitesse de variation limitée à la vitesse de balayage

La vitesse de variation de la tension de sortie est bornée entre la vitesse de balayage à la descente $-S_r$ et la vitesse de balayage à la montée S_r .

$$\forall t \in \mathbb{R}, \\ -S_r \leq \frac{d V_{out}(t)}{dt} \leq S_r$$

Preuve

Quand la sortie sature, la tension est constante, sa vitesse de variation est donc nulle et par conséquent comprise entre la vitesse de balayage à la descente $-S_r$ et la vitesse de balayage à la montée S_r .

Quand la sortie ne sature pas, par construction (cf. définition 20), la vitesse de variation de la tension de sortie est laissée libre uniquement si celle-ci est strictement comprise entre la vitesse de balayage à la descente $-S_r$ et la vitesse de balayage à la montée S_r .

Quand la sortie ne sature pas et que la vitesse de variation dépasse la vitesse de balayage à la montée, la tension de sortie est alors modélisée par la fonction $\{S_r \cdot t - S_r \cdot t_0 + V t_0\}$ dont la dérivée vaut S_r .

Quand la sortie ne sature pas et que la vitesse de variation dépasse la vitesse de balayage à la descente, la tension de sortie est alors modélisée par la fonction $\{-Sr \cdot t + Sr \cdot t_0 + Vt_0\}$ dont la dérivée vaut $-Sr$.

Par conséquent, en toute circonstance : $\forall t \in \mathbb{R}, -Sr \leq \frac{d V_{out} Sr(t)}{dt} \leq Sr$

A 4.5) Paramètres du modèle de l'amplificateur opérationnel simplifié

Les paramètres du modèles de l'amplificateur opérationnel simplifiés se répartissent en deux groupes : les paramètres ajustables et les paramètres internes. Les paramètres ajustables sont manipulés par le concepteur de circuit qui utilise le modèle de l'amplificateur opérationnel simplifié. Ils reçoivent une valeur par défaut indiquée ci-dessous afin de faciliter l'usage du modèle en autorisant le concepteur à ne pas toujours devoir spécifier tous les paramètres. Les paramètres internes ne sont pas directement affectés par le concepteur mais sont calculés à partir des paramètres ajustables et de la tension d'alimentation vue par l'amplificateur. Les formules de calcul sont rappelées ci-dessous. Les paramètres ajustables sont publics, au sens de la programmation orientée objet, tandis que les paramètres internes sont protégés, toujours au sens de la programmation orientée objet.

Paramètres ajustables

Les valeurs par défaut des paramètres ajustables correspondent aux valeurs typiques sous 5V de l'AD8541.

Nom	Désignation	Valeur par défaut
A (ou A_{vo})	Gain statique différentiel en tension	40000 (≈ 92 dB)
$V_{splyMax}$	Valeur maximale autorisée pour l'alimentation vue par l'amplificateur	6 V
$V_{splyMin}$	Valeur minimale autorisée pour l'alimentation vue par l'amplificateur	2,5 V
$VoLoss$	Tension de déchet en sortie	0,1 V
GBP	Produit gain-bande	1 MHz
Sr	Vitesse de balayage	0,92 V/ μ s
Iq	Courant de repos (consommation propre de l'amplificateur)	45 μ A
Isc	Courant de court-circuit (courant maximal de sortie)	60 mA

Paramètres internes

- V_{sply} : tension d'alimentation vue par l'amplificateur
 $V_{sply} = V_{dd} - V_{ss}$
- V_{bias} : tension de repos de la sortie
 $V_{bias} = \frac{(V_{dd} - V_{ss})}{2} + V_{ss} = \frac{V_{sply}}{2} + V_{ss}$
- V_{in} : tension différentielle d'entrée de l'amplificateur
 $V_{in}(t) = V^+(t) - V^-(t)$
- $VoSatH$: seuil de saturation supérieure de la tension de sortie
 $VoSatH = V_{dd} - VoLoss$
- $VoSatL$: seuil de saturation inférieure de la tension de sortie
 $VoSatL = V_{ss} + VoLoss$
- τ : constante de temps
 $\tau = \frac{1}{2\pi f_c}$

A 4.6) Algorithme de la tension de sortie de l'amplificateur opérationnel simplifié

Afin de faciliter l'implémentation, l'algorithme pour le calcul de la tension dans le modèle de l'amplificateur opérationnel simplifié est présenté ici en entier. La lecture de cet algorithme est rendue plus aisée en s'appuyant sur les figures 41 et 43 qui sont reproduites ci-dessous à la suite de l'algorithme.

$$V_{in}(t) = V^+(t) - V^-(t)$$

$$V_{sply} = V_{dd} - V_{ss}$$

$$V_{bias} = \frac{V_{sply}}{2} + V_{ss}$$

$$V_{oSatH} = V_{dd} - V_{oLoss}$$

$$V_{oSatL} = V_{ss} + V_{oLoss}$$

$$\tau \cdot \frac{d V_{outNS}(t)}{dt} + V_{outNS}(t) = A \cdot V_{inNS}(t) + V_{bias}$$

```

Si (SatMode = MSatL) alors
  /* Saturation par rapport au seuil inférieur */
  Si (VoutNS(t) > VoSatL) alors
    /* Retour en mode non-saturé */
    Vout(t) = VoutNS(t)
    SatMode = MNoSat
  Fin Si
  Si (VoutNS(t) ≤ VoSatL) alors
    /* L'amplificateur reste en saturation */
    Vout(t) = VoSatL
    SatMode = MSatL
  Fin Si
Fin Si /* Fin saturation par rapport au seuil inférieur */

Si (SatMode = MNoSat) alors
  /* Mode non-saturé : examiner en premier la vitesse de variation */
  Si SrMode = MSrL alors
    /* Vitesse de variation limitée à la descente */
    Si (VoutNS(t) - Vout(t) > (-ε)) alors
      /* Écart avec la fonction de surveillance est inférieur à l'erreur spécifiée ε
      ⇒ Retour à une vitesse de variation non-limitée */
      Vout(t) = VoutNS(t)
      SrMode = MNoSr
    Sinon
      /* Maintient de la limitation de la vitesse de variation à la vitesse de balayage à la descente */
      Vout(t) = -Sr·(t) + Sr·t0 + Vt0
      SrMode = MSrL
    Fin Si
  Fin Si /* Fin vitesse de variation limitée à la descente */

Si SrMode = MNoSr alors
  /* Vitesse de variation non-limitée */
  Si  $\left(\frac{d V_{out}(t)}{dt} \leq -Sr\right)$  alors
    /* Limitation de la vitesse de variation à la vitesse de balayage à la descente */
    Vout(t) = -Sr·(t) + Sr·t0 + Vt0
    SrMode = MSrL

```

```

Fin Si
Si  $\left(-Sr < \frac{d Vout(t)}{dt} < Sr\right)$  alors
  /* La vitesse de variation reste non-limitée */
   $Vout(t) = VoutNS(t)$ 
  ( $SrMode = MNoSr$ )
Fin Si
Fin Si
Si  $\left(\frac{d Vout(t)}{dt} \geq Sr\right)$  alors
  /* Limitation de la vitesse de variation à la vitesse de balayage à la montée */
   $Vout(t) = Sr \cdot (t) - Sr \cdot t_0 + Vt_0$ 
   $SrMode = MSrH$ 
Fin Si
Fin Si /* Fin vitesse de variation non-limitée */

Si  $SrMode = MSrH$  alors
  /* Vitesse de variation limitée à la montée */
  Si  $(VoutNS(t) - Vout(t) < \epsilon)$  alors
    /* Écart avec la fonction de surveillance est inférieur à l'erreur spécifiée  $\epsilon$ 
    ⇒ Retour à une vitesse de variation non-limitée */
     $Vout(t) = VoutNS(t)$ 
     $SrMode = MNoSr$ 
  Sinon
    /* Maintient de la limitation de la vitesse de variation à la vitesse de balayage à la montée */
     $Vout(t) = Sr \cdot (t) - Sr \cdot t_0 + Vt_0$ 
     $SrMode = MSrH$ 
  Fin Si
Fin Si /* Fin vitesse de variation limitée à la montée */

/* Mode non-saturé : examiner l'amplitude par rapport aux seuils de saturation */
Si  $(Vout(t) \leq VoSatL)$  alors
  /* Passage en saturation par rapport au seuil inférieur */
   $Vout(t) = VoSatL$ 
   $SatMode = MSatL$ 
   $SrMode = MNoSr$  /* La tension ne varie plus quand la sortie est en saturation */
Fin Si

Si  $(VoSatL < Vout(t) < VoSatH)$  alors
  /* Pas de saturation : l'équation de la tension de sortie ne doit pas être touchée ici, l'amplificateur
  pouvant être en mode normal tout comme en limitation de la vitesse de variation */
   $SatMode = MNoSat$ 
Fin Si

Si  $(Vout(t) \geq VoSatH)$  alors
  /* Passage en saturation par rapport au seuil supérieur */
   $Vout(t) = VoSatH$ 
   $SrMode = MNoSr$  /* La tension ne varie plus quand la sortie est en saturation */
Fin Si
Fin Si /* Fin du mode non-saturé */

Si  $(SatMode = MSatH)$  alors
  /* Saturation par rapport au seuil supérieur */
  Si  $(VoutNS(t) < VoSatH)$  alors
    /* Retour en mode non-saturé */

```

$V_{out}(t) = V_{outNS}(t)$
 $SatMode = MNoSat$

Fin Si

Si $(V_{outNS}(t) \geq V_{oSatH})$ alors

/* L'amplificateur reste en saturation */

$V_{out}(t) = V_{oSatH}$

$SatMode = MSatH$

Fin Si

Fin Si /* Fin saturation par rapport au seuil supérieur */

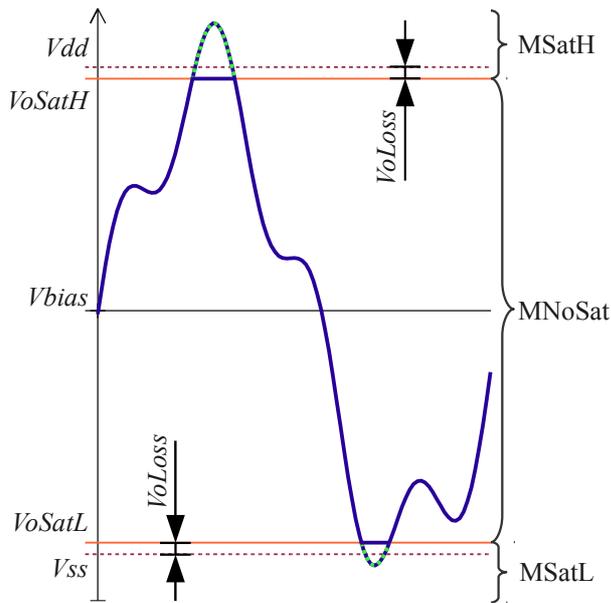


Fig. 41 : Saturation par écrêtage

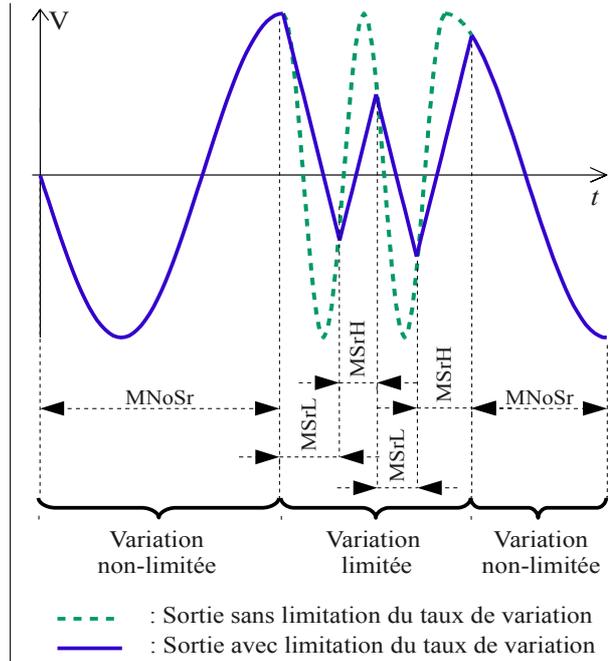


Fig. 42 : Transitions entre modes de saturation

Annexe 5)

Fonction inverse translatée

Calcul des coefficients de translation et du facteur d'échelle connaissant trois points appartenant à la courbe translatée

L'équation (63) est l'équation générale d'une fonction inverse translatée. Les coefficients de translation a et b sont explicités Fig. 77. Le facteur d'échelle k est explicité Fig. 78.

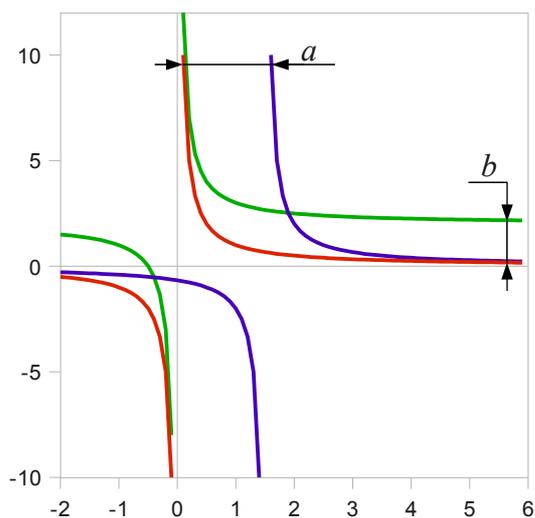


Fig. 77 : Coefficients de translation

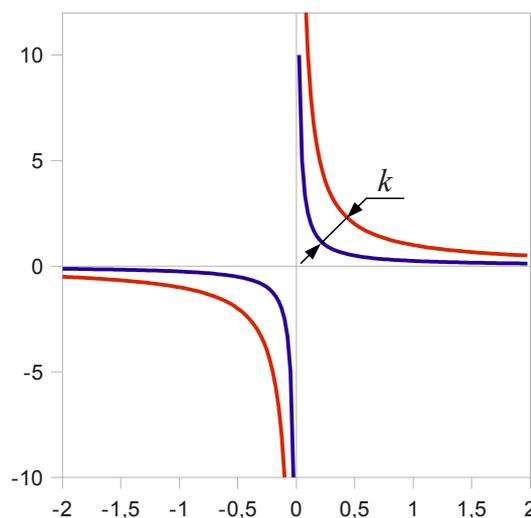


Fig. 78 : Facteur d'échelle

$$k, a, b \in \mathbb{R}; k \neq 0;$$

$$y = \frac{1}{kx - a} + b$$

(63)

Établissement des formules de calcul

Soient trois points $P_1(x_1, y_1)$, $P_2(x_2, y_2)$ et $P_3(x_3, y_3)$ distincts appartenant à la même branche d'une courbe inverse translatée, tels que $x_1 < x_2 < x_3$.

A partir de (63), on déduit le système d'équation (64).

$$\begin{cases} y_1 = \frac{1}{kx_1 - a} + b \\ y_2 = \frac{1}{kx_2 - a} + b \\ y_3 = \frac{1}{kx_3 - a} + b \end{cases} \quad (64)$$

On calcule $(y_1 - y_2)$:

$$\begin{aligned}
 y_1 - y_2 &= \frac{1}{kx_1 - a} - \frac{1}{kx_2 - a} \\
 \Leftrightarrow y_1 - y_2 &= \frac{(kx_2 - a) - (kx_1 - a)}{(kx_1 - a)(kx_2 - a)} = \frac{kx_2 - kx_1}{(kx_1 - a)(kx_2 - a)} \\
 \Leftrightarrow (kx_1 - a)(kx_2 - a) &= \frac{k(x_2 - x_1)}{(y_1 - y_2)} \\
 \Leftrightarrow k^2 x_1 x_2 - akx_1 - akx_2 + a^2 &= \frac{k(x_2 - x_1)}{(y_1 - y_2)} \\
 \Leftrightarrow a^2 - ak(x_1 + x_2) + k^2 x_1 x_2 - \frac{k(x_2 - x_1)}{(y_1 - y_2)} &= 0 \tag{65}
 \end{aligned}$$

De manière similaire, les équations (66) et (67) sont obtenues par le calcul de $(y_1 - y_3)$ et de $(y_2 - y_3)$.

$$a^2 - ak(x_1 + x_3) + k^2 x_1 x_3 - \frac{k(x_3 - x_1)}{(y_1 - y_3)} = 0 \tag{66}$$

$$a^2 - ak(x_2 + x_3) + k^2 x_2 x_3 - \frac{k(x_3 - x_2)}{(y_2 - y_3)} = 0 \tag{67}$$

On calcule $\{(65) - (66)\}$:

$$\begin{aligned}
 ak[(x_1 + x_3) - (x_1 + x_2)] + k^2(x_1 x_2 - x_1 x_3) + k \left[\frac{(x_3 - x_1)}{(y_1 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] &= 0 \\
 \Leftrightarrow a(x_3 - x_2) + kx_1(x_2 - x_3) + \left[\frac{(x_3 - x_1)}{(y_1 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] &= 0 \\
 \Leftrightarrow a = -\frac{kx_1(x_2 - x_3)}{(x_3 - x_2)} - \frac{1}{(x_3 - x_2)} \cdot \left[\frac{(x_3 - x_1)}{(y_1 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] \\
 \Leftrightarrow a = kx_1 - \frac{1}{(x_3 - x_2)} \cdot \left[\frac{(x_3 - x_1)}{(y_1 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] & \tag{68}
 \end{aligned}$$

On calcule $\{(65) - (67)\}$:

$$\begin{aligned}
 ak[(x_2 + x_3) - (x_1 + x_2)] + k^2(x_1 x_2 - x_2 x_3) + k \left[\frac{(x_3 - x_2)}{(y_2 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] &= 0 \\
 \Leftrightarrow a(x_3 - x_1) + kx_2(x_1 - x_3) + \left[\frac{(x_3 - x_2)}{(y_2 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] &= 0 \\
 \Leftrightarrow a = -\frac{kx_2(x_1 - x_3)}{(x_3 - x_1)} - \frac{1}{(x_3 - x_1)} \cdot \left[\frac{(x_3 - x_2)}{(y_2 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] \\
 \Leftrightarrow a = kx_2 - \frac{1}{(x_3 - x_1)} \cdot \left[\frac{(x_3 - x_2)}{(y_2 - y_3)} - \frac{(x_2 - x_1)}{(y_1 - y_2)} \right] & \tag{69}
 \end{aligned}$$

De (68) et (69), on déduit :

$$\begin{aligned}
 k x_1 - \frac{1}{(x_3-x_2)} \left[\frac{(x_3-x_1)}{(y_1-y_3)} - \frac{(x_2-x_1)}{(y_1-y_2)} \right] &= k x_2 - \frac{1}{(x_3-x_1)} \left[\frac{(x_3-x_2)}{(y_2-y_3)} - \frac{(x_2-x_1)}{(y_1-y_2)} \right] \\
 k(x_2-x_1) &= \frac{1}{(x_3-x_1)} \left[\frac{(x_3-x_2)}{(y_2-y_3)} - \frac{(x_2-x_1)}{(y_1-y_2)} \right] - \frac{1}{(x_3-x_2)} \left[\frac{(x_3-x_1)}{(y_1-y_3)} - \frac{(x_2-x_1)}{(y_1-y_2)} \right] \\
 k &= \frac{1}{(x_2-x_1)(x_3-x_1)} \left[\frac{(x_3-x_2)}{(y_2-y_3)} - \frac{(x_2-x_1)}{(y_1-y_2)} \right] - \frac{1}{(x_2-x_1)(x_3-x_2)} \left[\frac{(x_3-x_1)}{(y_1-y_3)} - \frac{(x_2-x_1)}{(y_1-y_2)} \right] \\
 k &= \left[\begin{array}{l} \frac{(x_3-x_2)}{(x_2-x_1)(x_3-x_1)(y_2-y_3)} - \frac{1}{(x_3-x_1)(y_1-y_2)} \\ - \frac{(x_3-x_1)}{(x_2-x_1)(x_3-x_2)(y_1-y_3)} + \frac{1}{(x_3-x_2)(y_1-y_2)} \end{array} \right]
 \end{aligned}$$

Réduction au même dénominateur :

$$\begin{aligned}
 k &= \left[\begin{array}{l} \frac{(x_3-x_2)^2 (y_1-y_2)(y_1-y_3)}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \\ - \frac{(x_2-x_1)(x_3-x_2)(y_1-y_3)(y_2-y_3)}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \\ - \frac{(x_3-x_1)^2 (y_1-y_2)(y_2-y_3)}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \\ + \frac{(x_2-x_1)(x_3-x_1)(y_1-y_3)(y_2-y_3)}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \end{array} \right] \\
 k &= \frac{\left[\begin{array}{l} (x_3-x_2)^2 (y_1-y_2)(y_1-y_3) \\ - (x_2-x_1)(x_3-x_2)(y_1-y_3)(y_2-y_3) \\ - (x_3-x_1)^2 (y_1-y_2)(y_2-y_3) \\ + (x_2-x_1)(x_3-x_1)(y_1-y_3)(y_2-y_3) \end{array} \right]}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \\
 k &= \frac{\left[\begin{array}{l} (x_3-x_2)^2 (y_1-y_2)(y_1-y_3) \\ + (x_2-x_1) (y_1-y_3)(y_2-y_3)[(x_3-x_1)-(x_3-x_2)] \\ - (x_3-x_1)^2 (y_1-y_2)(y_2-y_3) \end{array} \right]}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \\
 k &= \frac{\left[\begin{array}{l} (x_3-x_2)^2 (y_1-y_2)(y_1-y_3) \\ + (x_2-x_1)^2 (y_1-y_3)(y_2-y_3) \\ - (x_3-x_1)^2 (y_1-y_2)(y_2-y_3) \end{array} \right]}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \tag{70}
 \end{aligned}$$

Le coefficient de translation a est calculé avec (69) et (70) :

$$a = k x_2 - \frac{1}{(x_3-x_1)} \left[\frac{(x_3-x_2)}{(y_2-y_3)} - \frac{(x_2-x_1)}{(y_1-y_2)} \right]$$

$$\begin{aligned}
a &= k x_2 - \frac{(x_3 - x_2)(y_1 - y_2) - (x_2 - x_1)(y_2 - y_3)}{(x_3 - x_1)(y_1 - y_2)(y_2 - y_3)} \\
a &= \left(\frac{x_2 \cdot \begin{bmatrix} (x_3 - x_2)^2 (y_1 - y_2)(y_1 - y_3) \\ + (x_2 - x_1)^2 (y_1 - y_3)(y_2 - y_3) \\ - (x_3 - x_1)^2 (y_1 - y_2)(y_2 - y_3) \end{bmatrix}}{\begin{bmatrix} (x_2 - x_1)(x_3 - x_1)(x_3 - x_2)(y_1 - y_2)(y_1 - y_3)(y_2 - y_3) \\ (x_3 - x_2)(y_1 - y_2) - (x_2 - x_1)(y_2 - y_3) \end{bmatrix}} \right) \\
a &= \left(\frac{x_2 \cdot \begin{bmatrix} (x_3 - x_2)^2 (y_1 - y_2)(y_1 - y_3) \\ + (x_2 - x_1)^2 (y_1 - y_3)(y_2 - y_3) \\ - (x_3 - x_1)^2 (y_1 - y_2)(y_2 - y_3) \end{bmatrix}}{\begin{bmatrix} (x_2 - x_1)(x_3 - x_1)(x_3 - x_2)(y_1 - y_2)(y_1 - y_3)(y_2 - y_3) \\ (x_2 - x_1)(x_3 - x_2)^2 (y_1 - y_2)(y_1 - y_3) - (x_2 - x_1)^2 (x_3 - x_2)(y_1 - y_3)(y_2 - y_3) \end{bmatrix}} \right) \\
a &= \frac{\begin{bmatrix} (x_3 - x_2)^2 x_2 (y_1 - y_2)(y_1 - y_3) \\ + (x_2 - x_1)^2 x_2 (y_1 - y_3)(y_2 - y_3) \\ - (x_3 - x_1)^2 x_2 (y_1 - y_2)(y_2 - y_3) \\ - (x_3 - x_2)^2 (x_2 - x_1)(y_1 - y_2)(y_1 - y_3) \\ + (x_2 - x_1)^2 (x_3 - x_2)(y_1 - y_3)(y_2 - y_3) \end{bmatrix}}{(x_2 - x_1)(x_3 - x_1)(x_3 - x_2)(y_1 - y_2)(y_1 - y_3)(y_2 - y_3)} \\
a &= \frac{\begin{bmatrix} (x_3 - x_2)^2 (y_1 - y_2)(y_1 - y_3)[x_2 - (x_2 - x_1)] \\ + (x_2 - x_1)^2 (y_1 - y_3)(y_2 - y_3)[x_2 + (x_3 - x_2)] \\ - (x_3 - x_1)^2 x_2 (y_1 - y_2)(y_2 - y_3) \end{bmatrix}}{(x_2 - x_1)(x_3 - x_1)(x_3 - x_2)(y_1 - y_2)(y_1 - y_3)(y_2 - y_3)} \\
a &= \frac{\begin{bmatrix} (x_3 - x_2)^2 x_1 (y_1 - y_2)(y_1 - y_3) \\ + (x_2 - x_1)^2 x_3 (y_1 - y_3)(y_2 - y_3) \\ - (x_3 - x_1)^2 x_2 (y_1 - y_2)(y_2 - y_3) \end{bmatrix}}{(x_2 - x_1)(x_3 - x_1)(x_3 - x_2)(y_1 - y_2)(y_1 - y_3)(y_2 - y_3)} \tag{71}
\end{aligned}$$

Du système d'équations (64), on déduit :

$$\begin{cases} y_1 = \frac{1}{k x_1 - a} + b \\ y_2 = \frac{1}{k x_2 - a} + b \\ y_3 = \frac{1}{k x_3 - a} + b \end{cases} \Leftrightarrow \begin{cases} y_1 - b = \frac{1}{k x_1 - a} \\ y_2 - b = \frac{1}{k x_2 - a} \\ y_3 - b = \frac{1}{k x_3 - a} \end{cases} \Leftrightarrow \begin{cases} k x_1 - a = \frac{1}{y_1 - b} \\ k x_2 - a = \frac{1}{y_2 - b} \\ k x_3 - a = \frac{1}{y_3 - b} \end{cases}$$

On calcule de $(k x_2 - k x_1)$

$$\Rightarrow k x_2 - k x_1 = \frac{1}{y_2 - b} - \frac{1}{y_1 - b}$$

$$\Leftrightarrow k(x_2 - x_1) = \frac{(y_1 - b) - (y_2 - b)}{(y_2 - b)(y_1 - b)}$$

$$\Leftrightarrow k(x_2 - x_1) = \frac{(y_1 - y_2)}{y_1 y_2 - b(y_1 + y_2) + b^2}$$

$$\Leftrightarrow b^2 - b(y_1 + y_2) + y_1 y_2 = \frac{(y_1 - y_2)}{k(x_2 - x_1)}$$

$$\Leftrightarrow b^2 - b(y_1 + y_2) + y_1 y_2 - \frac{(y_1 - y_2)}{k(x_2 - x_1)} = 0 \quad (72)$$

De manière similaire, le calcul de $(k x_3 - k x_1)$ aboutit à l'équation (73).

$$b^2 - b(y_1 + y_3) + y_1 y_3 - \frac{(y_1 - y_3)}{k(x_3 - x_1)} = 0 \quad (73)$$

On calcule $\{(72) - (73)\}$:

$$b(y_1 + y_3) - b(y_1 + y_2) + y_1 y_2 - y_1 y_3 - \frac{(y_1 - y_2)}{k(x_2 - x_1)} + \frac{(y_1 - y_3)}{k(x_3 - x_1)} = 0$$

$$\Leftrightarrow b(y_3 - y_2) - y_1(y_3 - y_2) - \frac{(y_1 - y_2)}{k(x_2 - x_1)} + \frac{(y_1 - y_3)}{k(x_3 - x_1)} = 0$$

$$\Leftrightarrow b(y_3 - y_2) = y_1(y_3 - y_2) + \frac{(y_1 - y_2)}{k(x_2 - x_1)} - \frac{(y_1 - y_3)}{k(x_3 - x_1)}$$

$$\Leftrightarrow b = y_1 + \frac{1}{k(y_3 - y_2)} \left[\frac{(y_1 - y_2)}{(x_2 - x_1)} - \frac{(y_1 - y_3)}{(x_3 - x_1)} \right]$$

$$\Leftrightarrow b = y_1 + \frac{1}{k(y_2 - y_3)} \left[\frac{(y_1 - y_3)}{(x_3 - x_1)} - \frac{(y_1 - y_2)}{(x_2 - x_1)} \right]$$

$$\Leftrightarrow b = y_1 + \frac{1}{k} \cdot \frac{(x_2 - x_1)(y_1 - y_3) - (x_3 - x_1)(y_1 - y_2)}{(x_2 - x_1)(x_3 - x_1)(y_2 - y_3)}$$

k est remplacé en utilisant (70).

$$b = y_1 + \left\{ \frac{(x_2 - x_1)(x_3 - x_1)(x_3 - x_2)(y_1 - y_2)(y_1 - y_3)(y_2 - y_3)}{\begin{bmatrix} (x_3 - x_2)^2 (y_1 - y_2)(y_1 - y_3) \\ + (x_2 - x_1)^2 (y_1 - y_3)(y_2 - y_3) \\ - (x_3 - x_1)^2 (y_1 - y_2)(y_2 - y_3) \end{bmatrix}} \right. \\ \left. \times \frac{(x_2 - x_1)(y_1 - y_3) - (x_3 - x_1)(y_1 - y_2)}{(x_2 - x_1)(x_3 - x_1)(y_2 - y_3)} \right\}$$

$$b = y_1 + \frac{(x_3 - x_2)(y_1 - y_2)(y_1 - y_3) [(x_2 - x_1)(y_1 - y_3) - (x_3 - x_1)(y_1 - y_2)]}{\begin{bmatrix} (x_3 - x_2)^2 (y_1 - y_2)(y_1 - y_3) \\ + (x_2 - x_1)^2 (y_1 - y_3)(y_2 - y_3) \\ - (x_3 - x_1)^2 (y_1 - y_2)(y_2 - y_3) \end{bmatrix}} \quad (74)$$

Récapitulation des formules de calcul de k , a et b :

$$k = \frac{\begin{bmatrix} (x_3-x_2)^2 (y_1-y_2)(y_1-y_3) \\ + (x_2-x_1)^2 (y_1-y_3)(y_2-y_3) \\ - (x_3-x_1)^2 (y_1-y_2)(y_2-y_3) \end{bmatrix}}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \quad (70)$$

$$a = \frac{\begin{bmatrix} (x_3-x_2)^2 x_1 (y_1-y_2)(y_1-y_3) \\ + (x_2-x_1)^2 x_3 (y_1-y_3)(y_2-y_3) \\ - (x_3-x_1)^2 x_2 (y_1-y_2)(y_2-y_3) \end{bmatrix}}{(x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)} \quad (71)$$

$$b = y_1 + \frac{(x_3-x_2)(y_1-y_2)(y_1-y_3) [(x_2-x_1)(y_1-y_3) - (x_3-x_1)(y_1-y_2)]}{\begin{bmatrix} (x_3-x_2)^2 (y_1-y_2)(y_1-y_3) \\ + (x_2-x_1)^2 (y_1-y_3)(y_2-y_3) \\ - (x_3-x_1)^2 (y_1-y_2)(y_2-y_3) \end{bmatrix}} \quad (74)$$

En posant :

$$Dx21 = (x_2-x_1) \quad Dy12 = (y_1-y_2)$$

$$Dx31 = (x_3-x_1) \quad Dy13 = (y_1-y_3)$$

$$Dx32 = (x_3-x_2) \quad Dy23 = (y_2-y_3)$$

$$Numk = \begin{bmatrix} (x_3-x_2)^2 (y_1-y_2)(y_1-y_3) \\ + (x_2-x_1)^2 (y_1-y_3)(y_2-y_3) \\ - (x_3-x_1)^2 (y_1-y_2)(y_2-y_3) \end{bmatrix} = \begin{bmatrix} Dx32^2 Dy12 Dy13 \\ + Dx21^2 Dy13 Dy23 \\ - Dx31^2 Dy12 Dy23 \end{bmatrix}$$

$$Numa = \begin{bmatrix} (x_3-x_2)^2 x_1 (y_1-y_2)(y_1-y_3) \\ + (x_2-x_1)^2 x_3 (y_1-y_3)(y_2-y_3) \\ - (x_3-x_1)^2 x_2 (y_1-y_2)(y_2-y_3) \end{bmatrix} = \begin{bmatrix} Dx32^2 Dy12 Dy13 x_1 \\ + Dx21^2 Dy13 Dy23 x_3 \\ - Dx31^2 Dy12 Dy23 x_2 \end{bmatrix}$$

$$Numb = (x_3-x_2)(y_1-y_2)(y_1-y_3) [(x_2-x_1)(y_1-y_3) - (x_3-x_1)(y_1-y_2)]$$

$$Numb = Dx32 Dy12 Dy13 [Dx21 Dy13 - Dx31 Dy12]$$

$$Denomk = (x_2-x_1)(x_3-x_1)(x_3-x_2)(y_1-y_2)(y_1-y_3)(y_2-y_3)$$

$$Denomk = Dx21 Dx31 Dx32 Dy12 Dy13 Dy23$$

Les équations (70), (71) et (74) peuvent s'écrire :

$$k = \frac{Numk}{Denomk} = \frac{Dx32^2 Dy12 Dy13 + Dx21^2 Dy13 Dy23 - Dx31^2 Dy12 Dy23}{Dx21 Dx31 Dx32 Dy12 Dy13 Dy23} \quad (75)$$

$$a = \frac{Numa}{Denomk} = \frac{Dx32^2 Dy12 Dy13 x_1 + Dx21^2 Dy13 Dy23 x_3 - Dx31^2 Dy12 Dy23 x_2}{Dx21 Dx31 Dx32 Dy12 Dy13 Dy23} \quad (76)$$

$$b = y_1 + \frac{Numb}{Numk} = y_1 + \frac{Dx32 Dy12 Dy13 [Dx21 Dy13 - Dx31 Dy12]}{Dx32^2 Dy12 Dy13 + Dx21^2 Dy13 Dy23 - Dx31^2 Dy12 Dy23} \quad (77)$$

Vérification sur un exemple numérique

Supposons que les coefficients à déterminer sont $k = 0,8$; $a = 0,32$ et $b = 1,2$. A partir de trois points appartenant à la courbe d'équation $y = \frac{1}{0,8x - 0,32} + 1,2$, les équations (70), (71) et (74) doivent permettre de retrouver les valeurs de k , a et b . On considère les trois points suivants :

$$P_1 \begin{cases} x_1 = 0,5 \\ y_1 = 13,7 \end{cases}$$

$$P_2 \begin{cases} x_2 = 1,65 \\ y_2 = 2,2 \end{cases}$$

$$P_3 \begin{cases} x_3 = 4,71 \\ y_3 = 1,49 \end{cases}$$

D'où :

$$Dx21 = (x_2 - x_1) = 1,65 - 0,50 = 1,15$$

$$Dy12 = (y_1 - y_2) = 13,7 - 2,20 = 11,50$$

$$Dx31 = (x_3 - x_1) = 4,71 - 0,50 = 4,21$$

$$Dy13 = (y_1 - y_3) = 13,7 - 1,49 = 12,21$$

$$Dx32 = (x_3 - x_2) = 4,71 - 1,65 = 3,06$$

$$Dy23 = (y_2 - y_3) = 2,2 - 1,49 = 0,71$$

$$Numk = Dx32^2 Dy12 Dy13 + Dx21^2 Dy13 Dy23 - Dx31^2 Dy12 Dy23$$

$$Numk = (3,06^2 \times 11,50 \times 12,21) + (1,15^2 \times 12,21 \times 0,71) - (4,21^2 \times 11,50 \times 0,71)$$

$$Numk = 1181,54$$

$$Numa = Dx32^2 Dy12 Dy13 x_1 + Dx21^2 Dy13 Dy23 x_3 - Dx31^2 Dy12 Dy23 x_2$$

$$(3,06^2 \times 11,50 \times 12,21 \times 0,5)$$

$$+ (1,15^2 \times 12,21 \times 0,71 \times 4,71)$$

$$- (4,21^2 \times 11,50 \times 0,71 \times 1,65)$$

$$Numa = 657,40 + 54,00 - 238,78$$

$$Numa = 472,71$$

$$Numb = Dx32 Dy12 Dy13 [Dx21 Dy13 - Dx31 Dy12]$$

$$Numb = 3,06 \times 11,50 \times 12,21 \cdot (1,15 \times 12,21 - 4,21 \times 11,50)$$

$$Numb = -14769,26$$

$$Denomk = Dx21 Dx31 Dx32 Dy12 Dy13 Dy23$$

$$Denomk = 1,15 \times 4,21 \times 3,06 \times 11,50 \times 12,21 \times 0,71$$

$$Denomk = 1476,98$$

$$k = \frac{Numk}{Denomk} = \frac{1181,54}{1476,98} = 0,80$$

OK

$$a = \frac{Numa}{Denomk} = \frac{472,71}{1476,98} = 0,32$$

OK

$$b = y_1 + \frac{Numb}{Numk} = 13,7 + \frac{-14769,26}{1181,54} = 1,2$$

OK

Les équations (70), (71) et (74) et leurs formes contractées (75), (76) et (77) sont validées.

Glossaire

Analogicien

Expert en conception de circuits analogiques

ASIC

Circuit intégré spécifique à une application – *Application Specific Integrated Circuit*

CAO (CAD)

Conception assistée par ordinateur – *Computer Assisted Design*

Conservatif

Un système ou un modèle est dit conservatif (au sens de la conservation de la charge électrique) s'il se conforme aux lois de Kirchhoff. Un modèle conservatif est capable de rendre compte des interactions et des effets mutuels entre blocs. Par exemple, les effets de l'impédance doivent être restitués correctement.

Un système non conservatif, souvent traité comme un flot orienté de signal ou de donné, ne tient pas compte de ces effets mutuels. Par exemple, une variation brutale de charge n'altère en aucune façon la valeur de sortie d'une sortie, même temporairement.

Le système d'équations d'un modèle conservatif est plus important en taille et, dans bien des cas, en complexité que celui d'un modèle non-conservatif. Par conséquent, les modèles non-conservatifs simulent plus rapidement.

Fréquence de coupure

Fréquence caractéristique à laquelle le gain intrinsèque en tension de l'amplificateur est atténué de 3 dB par rapport au gain statique différentiel. La fréquence de coupure définit la bande passante.

Fréquence de transition

Fréquence caractéristique à laquelle le gain intrinsèque en tension de l'amplificateur vaut 1 du fait de l'atténuation en fréquence.

Fonction d'Heaviside

Fonction échelon d'amplitude 1.

$$\forall t \in \mathbb{R}, u(t) = \begin{cases} 0 & \text{Si } t < 0 \\ 1 & \text{Si } t \geq 0 \end{cases}$$

Fonction de transfert

Fonction caractérisant la relation entre la sortie et l'entrée d'un système. Dans le cas d'un système à temps continu, la fonction de transfert est définie comme la transformée de Laplace de la réponse impulsionnelle du système.

En notant V_e le signal temporel appliqué à l'entrée du système et V_s le signal de sortie dans le domaine temporel, en notant VE et VS es transformées de Laplace respectives de V_e et V_s , la fonction de transfert H du système s'écrit :

$$\forall p \in \mathbb{C}, H(p) = \frac{VS(p)}{VE(p)} \quad \text{soit encore} \quad VS(p) = H(p) \cdot VE(p)$$

Gain intrinsèque

Valeur de gain propre que présente un étage amplificateur en l'absence de boucle de rétroaction

Gain statique différentiel en tension de l'amplificateur opérationnel

C'est le gain intrinsèque en tension que présente l'amplificateur pour un signal continu en entrée. C'est une grandeur caractéristique d'un amplificateur opérationnel.

Impulsion de Dirac

D'un point de vue mathématique, l'impulsion de Dirac est une distribution singulière qui, lorsque qu'elle est centrée autour d'un réel a , transforme une fonction réelle f en la valeur que cette fonction possède en a . La distribution de Dirac centrée en a est notée δ_a .

Soit une fonction f définie sur $Df \subset \mathbb{R}$

$$\forall a \in Df \\ \langle \delta_a, f \rangle = f(a)$$

Le plus souvent la distribution de Dirac est utilisée en la centrée autour de zéro. Dans ce cas, la notation est allégée en supprimant l'indice sous la lettre δ

$$\delta = \delta_0$$

La principale caractéristique d'une distribution de Dirac est la suivante :

$$\int_{\mathbb{R}} \delta dt = 1$$

D'un point de vue physique, l'impulsion de Dirac modélise une impulsion de largeur infinitésimale et d'amplitude infinie. L'infini étant une notion critiquable en physique, l'impulsion de Dirac s'interprète comme une impulsion extrêmement brève et d'amplitude extrêmement grande par rapport aux autres signaux appliqués au système étudié.

JFET

Transistor à effet de champ à jonction – *Junction Field Effect Transistor*

Méthodologie

Ensemble de règles et de démarches adoptées pour conduire une recherche

Modèle exécutable

Modèle écrit dans un langage informatique approprié qui peut être exécuté sur un ordinateur pour simuler, vérifier ou aider à construire l'objet modélisé.

Modèle de calcul

Formalisme de modélisation utilisé pour composer un modèle exécutable. Ce formalisme constitue la sémantique du modèle par la réunion des éléments syntaxiques du langage de modélisation, des règles syntaxiques d'assemblage des éléments syntaxiques et des règles de composition des règles syntaxiques.

MOSFET

Transistor à effet de champ à grille isolée – *Metal Oxide Silicon Field Effect Transistor*

OTA (*Operational Transconductance Amplifier*)

Amplificateur opérationnel à transconductance. Alors que l'amplificateur opérationnel impose sur sa sortie une tension proportionnelle à la différence de tension entre ses deux entrées, l'OTA fournit en sortie un courant proportionnel à la différence de tension entre ses deux entrées.

PLL

Boucle à verrouillage de phase – *Phase Locked Loop*

Pôle

Valeur qui annule le dénominateur d'une fraction de polynômes – racine du dénominateur

Produit gain-bande (*Gain Bandwidth Product – GBP*)

Le produit gain-bande est une grandeur caractéristique d'un amplificateur opérationnel. C'est le produit du gain intrinsèque que présente l'amplificateur à une fréquence donnée par la valeur de cette fréquence. Ce produit qui traduit l'atténuation en fréquence du gain intrinsèque est constant au delà de la fréquence de coupure f_c .

$$\forall f \geq f_c, \text{ GBP} = A(f) \cdot f$$

Il est souvent calculer à la fréquence de coupure, le gain étant alors considéré comme encore égale au gain statique différentiel A .

$$GBP = A \cdot f_c$$

La fréquence de transition f_t étant par définition la fréquence à laquelle le gain intrinsèque de l'amplificateur vaut 1, le produit gain-bande est aussi égal à la valeur de la fréquence de transition.

$$A(f_t) = 1 \Rightarrow GBP = f_t$$

RTL

Description au niveau transfert entre registres – *Register Transfer Level*

SiP

Système en boîtier – *System in Package*

Système mixte / circuit mixte

Système ou circuit électronique comportant des parties numériques et des parties analogiques interconnectées et interagissant entre elles

SoC

Système sur puce – *System on Chip*

Taux de réjection de l'alimentation (*Power-Supply Rejection Ratio – PSRR*)

Valeur absolue du ratio entre la variation de la tension d'alimentation vue par l'amplificateur (cf. définition 3) et la variation qu'elle entraîne dans la valeur de la tension de décalage en entrée. Le taux de réjection de l'alimentation caractérise quelle quantité d'ondulation résiduelle présente dans la tension d'alimentation se retrouve dans la tension de sortie de l'amplificateur.

La réjection totale des alimentations, c'est-à-dire l'absence d'influence la variation de la tension d'alimentation sur la tension de sortie, implique un taux de réjection d'alimentation infini. Ceci est évidemment impossible à réaliser avec un circuit réel.

Taux de réjection du mode commun (*Common-Mode Rejection Ratio – CMRR*)

Valeur absolue du ratio entre la variation la tension de mode commun en entrée de l'amplificateur et la variation qu'elle entraîne dans la valeur de la tension de sortie.

La réjection totale du mode commun, c'est-à-dire l'absence d'influence la variation de la tension de mode commun en entrée sur la tension de sortie, implique un taux de réjection du mode commun infini. Ceci est évidemment impossible à réaliser avec un circuit réel.

Temps d'établissement (*Settling time*)

Mesuré avec un échelon appliqué en entrée, le temps d'établissement est le temps mis par la sortie pour entrer et se maintenir dans une bande d'erreur prédéfinie autour de la valeur finale visée.

Tension de décalage en entrée (*Input Offset Voltage*)

Tension continue à appliquer entre les entrées de l'amplificateur opérationnel non-bouclé pour annuler la composante continue présente dans la tension de sortie.

Tension de mode commun (en entrée) (*Common-Mode Input Voltage*)

Moyenne arithmétique des tensions appliquées sur les entrées de l'amplificateur

TLM

Niveau de modélisation transactionnelle – *Transaction Level Modeling*

VCO

Oscillateur contrôlé en tension – *Voltage Controlled Oscillator*

Vitesse de balayage (*Slew rate – Sr*)

Vitesse maximale à laquelle l'amplificateur peut effectuer une variation de tension

Vobulation (*Wobulation*)

Balayage fréquentiel réalisé dans le domaine temporel destiné à analyser le comportement en fréquence d'un système. Le balayage est obtenu en changeant régulièrement la fréquence du signal périodique attaquant l'entrée du système étudié. Le signal d'entrée utilisé dans ce cas est le plus souvent une sinusoïde.

Bibliographie

- [1] C. Grimm, M. Barnasconi, A. Vachoux & K. Einwich. "An Introduction to Modeling Embedded Analog/Mixed-Signal Systems using SystemC-AMS Extensions". *Open SystemC Initiative (OSCI)*, June 2008.
En ligne sur http://publik.tuwien.ac.at/files/PubDat_171466.pdf
(consulté le : 03 févr. 2012)
- [2] M. Barnasconi. "SystemC AMS Extensions: Solving the Need for Speed". *Open SystemC Initiative (OSCI)*, May 2010.
En ligne sur http://www.accelera.org/community/articles/amsspeed/community/articles/amsspeed/SystemC_AMS_Solving_the_Need_for_Speed_May_2010.pdf
(consulté le : 13 août 2012)
- [3] K. Einwich, C. Grimm, W. Granig, G. Noessing, W. Scherr, S. Scotti, M. Barnasconi, G. Zucchelli & A. Vachoux. "Requirements Specification for SystemC Analog Mixed Device (AMS) Extensions". *OSCI*, March 2010.
- [4] M. Barnasconi & C. Grimm. Open SystemC Initiative (OSCI). "SystemC AMS extensions V1.0 User's Guide". March 2010.
- [5] M. Barnasconi, K. Einwich, C. Grimm & A. Vachoux. Open SystemC Initiative (OSCI). "Standard SystemC AMS extensions V1.0 Language Reference Manual". March 2010.
- [6] Laurence W. Nagel. "SPICE2: A Computer Program to Simulate Semiconductor Circuits". *Thèse de l'EECS Department, University of California, Berkeley*, 1975.
En ligne sur <http://www.eecs.berkeley.edu/Pubs/TechRpts/1975/9602.html>
(consulté le : 8 août 2012)
- [7] E. Christen & K. Bakalar. "VHDL-AMS-a hardware description language for analog and mixed-signal applications". *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 46, n°10, pp. 1263 -1272, oct 1999.
DOI : 10.1109/82.799677
- [8] S. Chandrasekaran, J. Barby, X. Bestel, S. Bresticker, K. Cameron, G. Coram, D. Cronauer, J. David, P. Floyd, G. Helwig, J. Hou, M. Kole, K. Kundert, P. Liebmann, M. Mierzwinski, D. Miller, A. Muranyi, M. Mirmak, P. O'Halloran, M. O'Leary, J. Sanders, D. Sharrit, S. Sutherland, P. Tamhankar, B. Troyanovsky & I. Yusim. Accellera Organization, Inc. "Verilog-AMS Language Reference Manual". June 2009.
En ligne sur <http://www.accelera.org/downloads/standards/v-ams/>
(consulté le : 11 août 2012)
- [9] K. Wills. "Viewlogic is first to integrate VHDL, Verilog and MAST simulators". *Business Wire*, Feb. 1995.
En ligne sur [http://www.thefreelibrary.com/Viewlogic is first to integrate VHDL, Verilog and MAST simulators.-a016534769](http://www.thefreelibrary.com/Viewlogic+is+first+to+integrate+VHDL,+Verilog+and+MAST+simulators.-a016534769)
(consulté le : 11 août 2012)
- [10] Torsten Mähne. "Efficient Modelling and Simulation Methodology for the Design of Heterogeneous Mixed-Signal Systems on Chip". *Thèse de l'École polytechnique fédérale de Lausanne (EPFL)*, April 2011.
DOI : 10.5075/epfl-thesis-4993
En ligne sur <http://library.epfl.ch/theses/?nr=4993>
(consulté le : 03 févr. 2012)

-
- [11] F. Pecheux, C. Lallement & A. Vachoux. "VHDL-AMS and Verilog-AMS as alternative hardware description languages for efficient modeling of multidiscipline systems". *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 24, n°2, pp. 204 - 225, feb. 2005.
DOI : 10.1109/TCAD.2004.841071
- [12] "SABER". *Synopsys*,
En ligne sur <http://www.synopsys.com/Systems/Saber/Pages/default.aspx>
(consulté le : 13 août 2012)
- [13] Luca Ferro. "Vérification de propriétés logico-temporelles de spécifications SystemC TLM". *Thèse de l'Université de Grenoble*, July 2011.
En ligne sur http://tel.archives-ouvertes.fr/index.php?halsid=hs7m7scn20p28tgqlefmdd0f7&view_this_doc=tel-00633069&version=2
(consulté le : 13 août 2012)
- [14] E. Sperling & J. Blyler. "New Kinds Of Hybrid Chips". *System-Level Design Community*, June 2012.
En ligne sur <http://chipdesignmag.com/sld/blog/2012/06/28/new-kinds-of-hybrid-chips/>
(consulté le : 14 août 2012)
- [15] P.J. Ashenden, P.A. Wilsey & D.E. Martin. "SUAVE: painless extension for an object-oriented VHDL". *VHDL International Users' Forum, 1997. Proceedings*, pp. 60 -67, oct 1997.
DOI : 10.1109/VIUF.1997.623930
- [16] M. Radetzki. "Overview of Objective VHDL Language Features". *Proceedings of the Forum on Design Languages (FDL 99), Lyon, , 1999*.
- [17] Peter J. Ashenden, Philip A. Wilsey & Dale E. Martin. "Critique of the Objective VHDL Language Definition". 2000.
En ligne sur <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.39.4701>
(consulté le : 14 août 2012)
- [18] S. L. Mufreni & M. Radetzki. "Objective VHDL". *Seminar Embedded System, Institut für Technische Informatik*, 2006.
En ligne sur http://www.iti.uni-stuttgart.de/~radetzki/Seminar06/06_report.pdf
(consulté le : 14 août 2012)
- [19] V. Berman. "A Tale of Two Languages: SystemC and SystemVerilog". *Chip Design Magazine*, July 2005.
En ligne sur <http://chipdesignmag.com/display.php?articleId=116&issueId=11%09>
(consulté le : 14 août 2012)
- [20] John Aynsley. "SystemC versus SystemVerilog". *Doulos*, Feb. 2009.
En ligne sur http://www.doulos.com/knowhow/video_gallery/#anchor3
(consulté le : 14 août 2012)
- [21] M. Bowen. "Handel-C Language Reference Manual". *Embedded Solutions Limited*, 2001.
En ligne sur <http://www.pa.msu.edu/hep/d0/12/Handel-C/Handel%20C.PDF>
(consulté le : 15 août 2012)
- [22] "Handel-C Synthesis Methodology". *Mentor Graphics*,
En ligne sur <http://www.mentor.com/products/fpga/handel-c/>
(consulté le : 15 août 2012)
- [23] R. Dömer. "The SpecC Language". *University of California, Irvine*, 2001.
En ligne sur <http://www.cecs.uci.edu/~specc/language.pdf>
(consulté le : 15 août 2012)
- [24] "Available Accellera Systems Initiative Standards". *Accellera Systems Initiative*, 2012.
En ligne sur <http://www.systemc.org/downloads/standards/>
(consulté le : 15 août 2012)
-

- [25] K. Karnane, G. Curtis & R. Goering. "Solutions For Mixed-Signal Soc Verification". *Cadence Design Systems*, 2009.
En ligne sur http://www.cadence.com/rl/Resources/white_papers/ms_soc_verification_wp.pdf
(consulté le : 03 févr. 2012)
- [26] R. Goering. "Magma Design claims first parallel fast Spice". *EE Times*, March 2007.
En ligne sur http://www.eetindia.co.in/ART_8800456102_1800000_NP_e05da045.HTM
(consulté le : 16 août 2012)
- [27] M. Santarini. "Nascentric intros Fast Spice simulator for digital, AMS, analog designs". *EDN*, Feb. 2008.
En ligne sur <http://www.edn.com/design/integrated-circuit-design/4327967/Nascentric-intros-Fast-Spice-simulator-for-digital-AMS-analog-designs>
(consulté le : 16 août 2012)
- [28] M. Vasilevski, F. Pecheux, N. Beilleau, H. Aboushady & K. Einwich. "Modeling and Refining Heterogeneous Systems With SystemC-AMS: Application to WSN". *Design, Automation and Test in Europe, 2008. DATE '08*, pp. 134 -139, March 2008.
DOI : 10.1109/DATE.2008.4484675
- [29] W. Hartong & S. Cranston. "Real Valued Modeling for Mixed Signal Simulation". *Cadence Design Systems*, Jan. 2009.
En ligne sur http://www.cadence.com/rl/Resources/application_notes/real_number_appNote.pdf
(consulté le : 03 févr. 2012)
- [30] Agilent technologies. "ADS Ptolemy Simulation". 2004.
En ligne sur <http://cp.literature.agilent.com/litweb/pdf/ads2004a/pdf/ptolemy.pdf>
(consulté le : 18 août 2012)
- [31] S. Orcioni, M. Ballicchia, G. Biagetti, R. d'Aparo & M. Conti. "System Level Modelling of RF IC in SystemC-WMS". *EURASIP Journal on Embedded Systems*, vol. 2008, n°1, pp. 371768, 2008.
DOI : 10.1155/2008/371768
En ligne sur <http://jes.erasipjournals.com/content/2008/1/371768>
(consulté le : 19 août 2012)
- [32] H. Al-Junaid & T. Kazmierski. "Analogue and mixed-signal extension to SystemC". *Circuits, Devices and Systems, IEE Proceedings -*, vol. 152, n°6, pp. 682 – 690, dec. 2005.
DOI : doi:10.1049/ip-cds:20045204
En ligne sur <http://eprints.soton.ac.uk/262792/>
(consulté le : 6 août 2012)
- [33] Hessa Al-Junaid, Tom Kazmierski & Leran Wang. "SystemC-A modeling of an automotive seating vibration isolation system". *Forum on Specification and Design Languages (FDL 2006)*, 2006.
En ligne sur <http://eprints.soton.ac.uk/263695/>
(consulté le : 6 août 2012)
- [34] "SystemC Analog/Mixed-signal Working Group (AMSWG)". *Accellera Systems Initiative*,
En ligne sur <http://www.accellera.org/activities/committees/systemc-ams/>
(consulté le : 20 août 2012)
- [35] A. Vachoux, C. Grimm & K. Einwich. "SystemC-AMS requirements, design objectives and rationale". *Design, Automation and Test in Europe Conference and Exhibition, 2003*, pp. 388 – 393, 2003.
DOI : 10.1109/DATE.2003.1253639
- [36] T. Uhle & K. Einwich. "A SystemCAMS extension for the simulation of non-linear circuits". *SOC Conference (SOCC), 2010 IEEE International*, pp. 193 -198, sept. 2010.
DOI : 10.1109/SOCC.2010.5784751
- [37] M. Damm, J. Haase & C. Grimm. "Co-Simulation of mixed HW/SW and Analog/RF systems at architectural level". *Behavioral Modeling and Simulation Workshop, 2008. BMAS 2008. IEEE International*, pp. 84 – 89, sept. 2008.
DOI : 10.1109/BMAS.2008.4751245

-
- [38] M. Damm, C. Grimm, J. Haas, A. Herrholz & W. Nebel. "Connecting SystemC-AMS models with OSCI TLM 2.0 models using temporal decoupling". *Specification, Verification and Design Languages, 2008. FDL 2008. Forum on*, pp. 25 -30, sept. 2008.
DOI : 10.1109/FDL.2008.4641416
- [39] A. Lévêque, F. Pecheux, M.M. Louerat, H. Aboushady & M. Vasilevski. "SystemC-AMS Models for Low-Power Heterogeneous Designs: Application to a WSN for the Detection of Seismic Perturbations". *23th International Conference on Architecture of Computing Systems (ARCS 2010)*, Feb. 2010.
En ligne sur <http://www.vde-verlag.de/proceedings-en/453222027.html>
(consulté le : 17 mai 2012)
- [40] A. Massouri, A. Lévêque, L. Clavier, M. Vasilevski, A. Kaiser & M.M. Louerat. "Baseband Fading Channel Simulator for Inter-Vehicle Communication using SystemC-AMS". *2010 IEEE International Behavioral Modeling and Simulation Conference (BMAS)*, pp. 2.3, Sept. 2010.
En ligne sur <http://hal.archives-ouvertes.fr/hal-00632156>
(consulté le : 17 mai 2012)
- [41] Fabio Cenni. "Modélisation à haut niveau de systèmes hétérogènes, interfaçage analogique /numérique". *Thèse de l'Université de Grenoble*, April 2012.
- [42] "SystemC AMS Day 2011 - Industry Adoption of the SystemC AMS Standard". *Accellera Systems Initiative*, May 2011.
En ligne sur http://www.accellera.org/news/events/systemc_ams_day
(consulté le : 21 août 2012)
- [43] "SystemC AMS 2.0 Draft Standard". *Accellera Systems Initiative*, March 2012.
En ligne sur http://www.accellera.org/activities/committees/systemc-ams/ams_2_draft_public_review/
(consulté le : 21 août 2012)
- [44] I. O'Connor, B. Courtois, K. Chakrabarty & M. Hampton. "Heterogeneous Systems on Chip and Systems in Package". *Design, Automation Test in Europe Conference Exhibition, 2007. DATE '07*, pp. 1 -6, april 2007.
DOI : 10.1109/DATE.2007.364683
- [45] Alex Doboli, Adrian Nunez-Aldana, Nagu Dhanwada, Sree Ganesan & Ranga Vemuri. "Behavioral synthesis of analog systems using two-layered design space exploration". *Proceedings of the 36th annual ACM/IEEE Design Automation Conference*, pp. 951–957, 1999.
DOI : 10.1145/309847.310105
En ligne sur <http://doi.acm.org/10.1145/309847.310105>
(consulté le : 17 mai 2012)
- [46] Adrian Nunez-Aldana & Ranga Vemuri. "An analog performance estimator for improving the effectiveness of CMOS analog systems circuit synthesis". *Proceedings of the conference on Design, automation and test in Europe*, 1999.
DOI : 10.1145/307418.307533
En ligne sur <http://doi.acm.org/10.1145/307418.307533>
(consulté le : 3 août 2012)
- [47] T.D. Burd & R.W. Brodersen. "Energy efficient CMOS microprocessor design". *System Sciences, 1995 (HICCS 1995). Proceedings of the Twenty-Eighth Hawaii International Conference on*, pp. 288–297 vol.1, Jan. 1995.
DOI : 10.1109/HICSS.1995.375385
- [48] David Rios-Arambula, Aurelien Buhrig, Gilles Sicard & Marc Renaudin. "On the use of Feedback Systems to Dynamically Control the Supply Voltage of Low-Power Circuits". *Journal of Low Power Electronics*, vol. 2, n°1, pp. 45-55, April 2006.
DOI : doi:10.1166/jolpe.2006.006
En ligne sur <http://www.ingentaconnect.com/content/asp/jolpe/2006/00000002/00000001/art00007>
(consulté le : 14 mai 2012)
-

- [49] A. Guyot. "Op Ar".
En ligne sur http://users-tima.imag.fr/cis/guyot/Cours/Oparithm/francais/Op_Ar2.htm
(consulté le : 1er mai 2012)
- [50] Chingyei Chung, Shou-Yen Chao & M. F. Lu. "Nonlinear feedback system for an inverter-based ring oscillator". *WSEAS Trans. Cir. and Sys.*, vol. 8, n°7, pp. 537–547, July 2009.
En ligne sur <http://dl.acm.org/citation.cfm?id=1639537.1639538>
(consulté le : 14 mai 2012)
- [51] H. Graeb, S. Zizala, J. Eckmueller & K. Antreich. "The sizing rules method for analog integrated circuit design". *Computer Aided Design, 2001. ICCAD 2001. IEEE/ACM International Conference on*, pp. 343–349, 2001.
DOI : 10.1109/ICCAD.2001.968645
- [52] T. Massier, H. Graeb & U. Schlichtmann. "The Sizing Rules Method for CMOS and Bipolar Analog Integrated Circuit Synthesis". *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 27, n°12, pp. 2209–2222, Dec. 2008.
DOI : 10.1109/TCAD.2008.2006143
- [53] F. Bouchhima, M. Briere, G. Nicolescu, M. Abid & E.M. Aboulhamid. "A SystemC/Simulink Co-Simulation Framework for Continuous/Discrete-Events Simulation". *Behavioral Modeling and Simulation Workshop, Proceedings of the 2006 IEEE International*, pp. 1–6, Sept. 2006.
DOI : 10.1109/BMAS.2006.283461
- [54] R. Narayanan, N. Abbasi, M. Zaki, G. Al Sammane & S. Tahar. "On the simulation performance of contemporary AMS hardware description languages". *Microelectronics, 2008. ICM 2008. International Conference on*, pp. 361–364, dec. 2008.
DOI : 10.1109/ICM.2008.5393509
- [55] P.A. Hartmann, P. Reinkemeier, A. Rettberg & W. Nebel. "Modelling control systems in SystemC AMS - Benefits and limitations". *SOC Conference, 2009. SOCC 2009. IEEE International*, pp. 263 – 266, sept. 2009.
DOI : 10.1109/SOCCON.2009.5398043
- [56] B. D. H. Tellegen. "La recherche pour une série complète d'éléments de circuit idéaux non=linéaires (sic)". *Milan Journal of Mathematics*, vol. 25, pp. 134-144, 1955.
En ligne sur <http://dx.doi.org/10.1007/BF02923815>
(consulté le : 3 août 2012)
- [57] H. Carlin. "Singular Network Elements". *Circuit Theory, IEEE Transactions on*, vol. 11, n°1, pp. 67 – 72, mar 1964.
DOI : 10.1109/TCT.1964.1082264
- [58] J.H. Huijsing & J.De. Korte. "Monolithic nullor-a universal active network element". *Solid-State Circuits, IEEE Journal of*, vol. 12, n°1, pp. 59 – 64, Feb. 1977.
DOI : 10.1109/JSSC.1977.1050841
- [59] C. Toumazou, G. Moschytz & B. Gilbert. "Analog Amplifiers Architectures: Gain Bandwidth Trade-Offs" in *Trade-Offs in Analog Circuit Design : the Designer's Companion*. Chap. 7, pp. 207 – 225, Editions Kluwer Academic Publishers, 2002.
- [60] E. Tlelo-Cuautle & L.A. Sarmiento-Reyes. "A pure nodal-analysis method suitable for analog circuits using nullors". *Journal of Applied Research and Technology*, vol. 1, n°3, pp. 235-247, Oct. 2003.
En ligne sur <http://www.revistas.unam.mx/index.php/jart/article/view/17512>
(consulté le : 19 juil. 2012)
- [61] D.G. Haigh. "Analytic approach to nullor transformations for FET circuit synthesis. Part I. Nullator-norator tree transformations". *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, pp. 4 pp., may 2006.
DOI : 10.1109/ISCAS.2006.1693812

-
- [62] D.G. Haigh. "Analytic approach to nullor transformations for FET circuit synthesis. Part II. Nullator-norator re-pairing and cloning". *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, pp. 4 pp., may 2006.
DOI : 10.1109/ISCAS.2006.1693813
- [63] G.R. Boyle, D.O. Pederson, B.M. Cohn & J.E. Solomon. "Macromodeling of integrated circuit operational amplifiers". *Solid-State Circuits, IEEE Journal of*, vol. 9, n°6, pp. 353 -364, Dec. 1974.
DOI : 10.1109/JSSC.1974.1050528
- [64] G. Oltean. "High Level Model for a Simple Transconductance Operational Amplifier". *Computer as a Tool, 2005. EUROCON 2005. The International Conference on*, pp. 1243 -1246, Nov. 2005.
DOI : 10.1109/EURCON.2005.1630181
- [65] B. Perez-Verdu, J.L. Huertas & A. Rodriguez-Vazquez. "A new nonlinear time-domain op-amp macromodel using threshold functions and digitally controlled network elements". *Solid-State Circuits, IEEE Journal of*, vol. 23, n°4, pp. 959 -971, Aug. 1988.
DOI : 10.1109/4.347
- [66] Huabiao Qin & Fei Wang. "Modeling of Operational Amplifier based on VHDL-AMS". *Electronics, Circuits and Systems, 2006. ICECS '06. 13th IEEE International Conference on*, pp. 894 -897, Dec. 2006.
DOI : 10.1109/ICECS.2006.379933
- [67] He Zhang & Guoyong Shi. "Symbolic behavioral modeling for slew and settling analysis of operational amplifiers". *Circuits and Systems (MWSCAS), 2011 IEEE 54th International Midwest Symposium on*, pp. 1-4, Aug. 2011.
DOI : 10.1109/MWSCAS.2011.6026581
- [68] C.T. Chuang. "Analysis of the settling behavior of an operational amplifier". *Solid-State Circuits, IEEE Journal of*, vol. 17, n°1, pp. 74 -80, Feb. 1982.
DOI : 10.1109/JSSC.1982.1051689
- [69] Philip E. Allen & Douglas R. Holberg. *CMOS Analog Circuit Design*, Second Edition, Editions Oxford Press, 2002.
ISBN : 978-0-19-511644-1
- [70] Laurent Bousquet, Fabio Cenni & Emmanuel Simeu. "Inclusion of Power Consumption Information in High-Level Modeling of Linear Analog Blocks". *J. Low Power Electronics*, vol. 7, n°4, pp. 541-551, 2011.
DOI : 10.1166/jolpe.2011.1156
En ligne sur <http://www.ingentaconnect.com/content/asp/jolpe/2011/00000007/00000004/art00010>
(consulté le : 27 août 2012)
- [71] L. Bousquet, F. Cenni & E. Simeu. "SystemC-AMS high-level modeling of linear analog blocks with power consumption information". *Test Workshop (LATW), 2011 12th Latin American*, pp. 1 -6, march 2011.
DOI : 10.1109/LATW.2011.5985924
- [72] Laurent Bousquet & Emmanuel Simeu. "High-level modeling of power consumption in active linear analog circuits". *Proceedings of the great lakes symposium on VLSI*, pp. 87-90, 2012.
DOI : 10.1145/2206781.2206804
En ligne sur <http://doi.acm.org/10.1145/2206781.2206804>
(consulté le : 27 août 2012)
- [73] J.M. Daga & D. Auvergne. "A comprehensive delay macro modeling for submicrometer CMOS logics". *IEEE Journal of Solid-State Circuits*, vol. 34, n°1, pp. 42-55, Jan. 1999.
DOI : 10.1109/4.736655
En ligne sur <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=736655&isnumber=15899>
(consulté le : 03 févr. 2012)
-

-
- [74] P. Maurine, M. Rezzoug, N. Azemard & D. Auvergne. "Transition time modeling in deep submicron CMOS". *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 21, n°11, pp. 1352–1363, 2002.
DOI : 10.1109/TCAD.2002.804088
En ligne sur http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1047054
(consulté le : 03 févr. 2012)
- [75] "Theoreme de Rolle et theoreme de la moyenne de Lagrange". *Cheneliere Education*,
En ligne sur
http://www.cheneliere.info/cfiles/complementaire/calcul_diffenriel_11_eme_edition/Theoreme_de_Rolle_et_theoreme_de_la_moyenne_de_Lagrange.pdf
(consulté le : 26 août 2012)
- [76] J. Ou, P. Brunmayr, M. Farooq, J. Haase, C. Grimm, M. Damm & J. Wenninger. "Tu vienna systemc ams communications library". *TU Vienna*, March 2010.
En ligne sur http://www.systemc-ams.org/BB_library.html
(consulté le : 27 août 2012)

Publications dans le cadre de la thèse

- Franck Pagnat, Laurent Fesquet, Katell Morin-Allory.
“Model of a Simple yet effective Operational Amplifier”
International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), Seville (Spain), Sept. 2012
- Franck Pagnat, Katell Morin-Allory, Laurent Fesquet.
“A Refinement Process for Top-Down Mixed-Signal Designs Thanks to SystemC-AMS”
New Circuits and Systems Conference (NEWCAS), 2011 IEEE 9th International, pp.378 – 381, June 2011
ISBN: 978-1-61284-135-9
DOI: [10.1109/NEWCAS.2011.5981249](https://doi.org/10.1109/NEWCAS.2011.5981249)
On line at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5981249 (last visit: 28 Sept. 2011)
- Franck Pagnat, Laurent Bousquet, Katell Morin-Allory, Laurent Fesquet
“A Performance Comparison Between the SystemC-AMS Models of Computation”
Proceedings of edaWorkshop 2011, pp. 13 – 18, May 2011, Dresden, Germany.
- Franck Pagnat, Laurent Bousquet, Katell Morin-Allory, Laurent Fesquet
“Analog Design Abstraction Levels and SystemC-AMS Models of Computation”
SystemC-AMS Day 2011: Industry Adoption of the SystemC AMS Standard, poster session, May 2011, Dresden, Germany.
On line at: http://www.systemc.org/news/events/systemc_ams_day (last visit: 4 July 2011)

Méthode de modélisation et de raffinement pour les systèmes hétérogènes

Résumé

Les systèmes sur puces intègrent aujourd'hui sur le même substrat des parties analogiques et des unités de traitement numérique. Tandis que la complexité de ces systèmes s'accroissait, leur temps de mise sur le marché se réduisait. Une conception descendante globale et coordonnée du système est devenue indispensable de façon à tenir compte des interactions entre les parties analogiques et les parties numériques dès le début du développement. Dans le but de répondre à ce besoin, cette thèse expose un processus de raffinement progressif et méthodique des parties analogiques, comparable à ce qui existe pour le raffinement des parties numériques. L'attention a été plus particulièrement portée sur la définition des niveaux analogiques les plus abstraits et à la mise en correspondance des niveaux d'abstraction entre parties analogiques et numériques. La cohérence du raffinement analogique exige de détecter le niveau d'abstraction à partir duquel l'utilisation d'un modèle trop idéalisé conduit à des comportements irréalistes et par conséquent d'identifier l'étape du raffinement à partir de laquelle les limitations et les non linéarités aux conséquences les plus fortes sur le comportement doivent être introduites. Cette étape peut être d'un niveau d'abstraction élevé. Le choix du style de modélisation le mieux adapté à chaque niveau d'abstraction est crucial pour atteindre le meilleur compromis entre vitesse de simulation et précision. Les styles de modélisations possibles à chaque niveau ont été examinés de façon à évaluer leur impact sur la simulation. Les différents modèles de calcul de SystemC-AMS ont été catégorisés dans cet objectif. Les temps de simulation obtenus avec SystemC-AMS ont été comparés avec Matlab Simulink. L'interface entre les modèles issus de l'exploration d'architecture, encore assez abstraits, et les modèles plus fins requis pour l'implémentation, est une question qui reste entière. Une bibliothèque de composants électroniques complexes décrits en SystemC-AMS avec le modèle de calcul le plus précis (modélisation ELN) pourrait être une voie pour réussir une telle interface. Afin d'illustrer ce que pourrait être un élément d'une telle bibliothèque et ainsi démontrer la faisabilité du concept, un modèle d'amplificateur opérationnel a été élaboré de façon à être suffisamment détaillé pour prendre en compte la saturation de la tension de sortie et la vitesse de balayage finie, tout en gardant un niveau d'abstraction suffisamment élevé pour rester indépendant de toute hypothèse sur la structure interne de l'amplificateur ou la technologie à employer.

Mots clés : niveaux d'abstraction, raffinement, conception analogique, synthèse haut niveau, signaux mixtes, modélisation, SystemC, AMS, SystemC-AMS, modèles de calcul, temps de simulation

Abstract

Systems on Chip (SoC) embed in the same chip analogue parts and digital processing units. While their complexity is ever increasing, their time to market is becoming shorter. A global and coordinated top-down design approach of the whole system is becoming crucial in order to take into account the interactions between the analogue and digital parts since the beginning of the development. This thesis presents a systematic and gradual refinement process for the analogue parts comparable to what exists for the digital parts. A special attention has been paid to the definition of the highest abstracted analogue levels and to the correspondence between the analogue and the digital abstraction levels. The analogue refinement consistency requires to detect the abstraction level where a too idealised model leads to unrealistic behaviours. Then the refinement step consist in introducing – for instance – the limitations and non-linearities that have a strong impact on the behaviour. Such a step can be done at a relatively high level of abstraction. Correctly choosing a modelling style, that suits well an abstraction level, is crucial to obtain the best trade-off between the simulation speed and the accuracy. The modelling styles at each abstraction level have been examined to understand their impact on the simulation. The SystemC-AMS models of computation have been classified for this purpose. The SystemC-AMS simulation times have been compared to that obtained with Matlab Simulink. The interface between models arisen from the architectural exploration – still rather abstracted – and the more detailed models that are required for the implementation, is still an open question. A library of complex electronic components described with the most accurate model of computation of SystemC-AMS (ELN modelling) could be a way to achieve such an interface. In order to show what should be an element of such a library, and thus prove the concept, a model of an operational amplifier has been elaborated. It is enough detailed to take into account the output voltage saturation and the finite slew rate of the amplifier. Nevertheless, it remains sufficiently abstracted to stay independent from any architectural or technological assumption.

Keywords: abstraction levels, refinement, analogue design, high-level synthesis, modelling, mixed signal, SystemC, AMS, SystemC-AMS, model of computation, MoC, simulation speed