

**Méthodologie d'intégration de commandes numériques pour dispositifs d'électronique
de puissance basée sur l'utilisation du langage VHDL-AMS**

P. Poure, R. Kadri, L. Hébrard et F. Braun

Laboratoire d'Électronique et de Physique des Systèmes Instrumentaux (LEPSI)
Université Louis Pasteur, IN2P3/CNRS
23 rue du Loess
BP 20
F-67037 Strasbourg Cedex, France
poure@lepsi.in2p3.fr

Méthodologie d'intégration de commandes numériques pour dispositifs d'Électronique de Puissance basée sur l'utilisation du langage VHDL-AMS

Philippe POURE, Rafik KADRI, Luc HEBRARD, Francis BRAUN

Laboratoire d'Électronique et de Physique des Systèmes Instrumentaux
Université Louis Pasteur, IN2P3/CNRS
BP 20 – 23, rue du Loess
67037 STRASBOURG cedex
E-mail : poure@lepsi.in2p3.fr

Résumé

Dans cet article, nous présentons une méthodologie Top-Down, basée sur l'utilisation du langage VHDL-AMS, en vue de l'intégration de commandes numériques de dispositifs d'électronique de puissance. Nous proposons de travailler dans un environnement unique pour la modélisation, la simulation, la validation fonctionnelle et la conception du futur circuit intégré numérique dans son contexte réel analogique et de puissance. La méthodologie proposée est détaillée étape par étape. Elle optimise la durée de développement et propose une solution face aux difficultés rencontrées lors de la conception de circuits intégrés pour la commande de dispositifs de puissance. La méthodologie est illustrée par un exemple dans le domaine du filtrage actif monophasé parallèle. Le flot de conception est détaillé, de la modélisation haut niveau à l'implantation sur cible FPGA.

1. Introduction

L'évolution de la Microélectronique dans le domaine des processeurs numériques de signaux est à l'origine de nombreux progrès en Electronique de Puissance depuis les années 1980. Ainsi, de nombreux travaux de recherche dans le domaine de la commande ont pu être mis en œuvre grâce aux performances des DSPs (Digital Signal Processor) ou des Microcontrôleurs [1]. De même, l'évolution récente des composants numériques de types FPGA (Field Programmable Gate Array) et ASIC (Application Specific Integrated Circuit) permet d'envisager l'intégration complète de commandes numériques de dispositifs d'Electronique de Puissance dans un seul et unique composant [1]. Les avantages d'une telle intégration sont multiples : réduction du nombre de composants, compacité, confidentialité de l'architecture, prototypage rapide de la commande numérique sur FPGA, possibilité d'étendre l'intégration de la partie numérique à l'ensemble du système de commande (commande numérique, drivers, capteurs, électronique analogique de traitement et de conditionnement du signal sur une même puce).

Cependant, le développement rapide de commandes numériques intégrées sur cibles FPGA et/ou ASIC dans le domaine de l'Electronique de Puissance n'est pas seulement dépendant de la maîtrise des procédés et des outils informatiques liés à ces circuits intégrés. Il est fortement lié à la modélisation et à la simulation en boucle fermée du

futur circuit intégré dans son contexte analogique et de puissance. En outre, à l'aide de langages de modélisation de haut niveau tels que le standard IEEE 1076.1-1999, encore appelé VHDL-AMS (Very high speed integrated circuits Hardware Description Language – Analog-Mixed Signal), les différentes phases de conception peuvent être optimisées. Ce langage VHDL-AMS permet de modéliser et de simuler dans un environnement unique les différents éléments d'un système de puissance (électronique analogique et numérique, composants de puissance, mécanique, thermique, capteurs, ...). De plus, la commande numérique peut être modélisée et fonctionnellement validée par simulation mixte en boucle fermée à différents niveaux d'abstraction. Ici, nous présentons tout d'abord une méthodologie de conception dite « Top Down » basée sur l'utilisation du langage VHDL-AMS, optimisant l'intégration de commandes numériques pour dispositifs d'Electronique de Puissance. Ensuite, un exemple d'application dans le domaine du filtrage actif est présenté et le flot de conception détaillé, de la modélisation haut niveau à l'implantation sur cible FPGA.

2. Méthodologie de conception

2.1 Approche générale

Les systèmes de puissance conventionnels sont constitués de composants discrets de natures différentes et ayant chacun des fonctionnalités différentes (figure 1). Ils sont principalement constitués d'éléments de puissance (source, convertisseur statique, charge), de capteurs, d'éléments analogiques et numériques destinés à l'interfaçage et la commande.

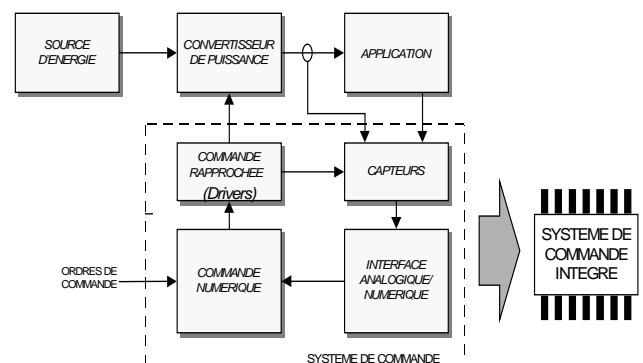


Figure 1 : Système de puissance.

Les outils classiques de conception assistée par ordinateur tels que Matlab-Simulink ainsi que les outils standards de simulation en Electronique de Puissance (Saber, Psim, ...) sont certes adaptés à l'implantation de commandes numériques sur Microcontrôleur, Microprocesseur ou DSP mais ne permettent pas l'implantation numérique sur cibles de type ASIC précaractérisé. Ainsi, afin de garantir un niveau de confiance élevé dans le futur circuit intégré numérique tout en assurant une durée de développement limitée et un coût réduit, nous avons éprouvé la nécessité de modéliser et simuler le système de puissance complet dans un environnement de simulation adapté et unique [2]. La simulation avant prototypage permet de corriger plus facilement les erreurs éventuelles et d'optimiser le coût de développement. Ainsi, sur la base de méthodologies récemment publiées [3] et compte tenu de l'évolution récente des langages de description matériel de haut niveau ou HDL (Hardware Description Languages), nous avons développé un flot de conception optimisé. Les HDLs numériques les plus connus sont VHDL et Verilog. En raison de la normalisation en mars 1999 de l'extension analogique de VHDL (appelée VHDL-AMS), nous avons choisi VHDL pour la modélisation comportementale et structurelle haut niveau de la commande numérique à intégrer. Le langage VHDL est basé sur la notion de modèles temporels discrets et pourrait ainsi être utilisé pour la modélisation comportementale des éléments analogiques et de puissance, si l'on échantillonne les phénomènes continus mis en œuvre [5]. Cependant, VHDL est basé sur la notion d'événements et n'est donc pas le langage le plus adapté aux phénomènes continus. Ainsi, nous avons choisi d'utiliser VHDL-AMS qui, quant à lui, permet de décrire les équations différentielles algébriques ordinaires. VHDL-AMS est en fait la combinaison de deux standards : VHDL 1076-1993 (HDL numérique) et VHDL 1076-1.1999 (extension analogique). De plus, les outils CAO mettant en œuvre ce langage mixte sont commercialisés depuis peu. Ainsi, VHDL-AMS permet de modéliser et de simuler le système de puissance complet. Ensuite, la commande numérique, décrite en VHDL peut alors être implantée indépendamment de la cible physique choisie, FPGA ou ASIC précaractérisé.

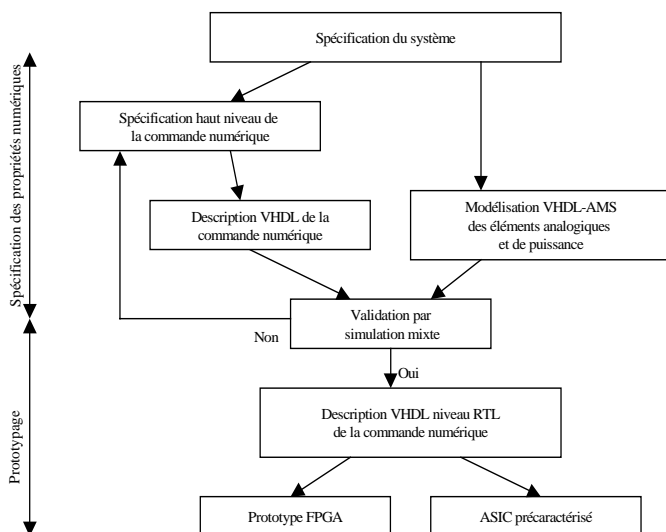


Figure 2 : Méthodologie « Top-Down ».

Le concepteur peut ainsi envisager l'intégration de la commande numérique par une méthodologie « Top-Down » (figure 2), partant d'une étude haut niveau avec des modèles peu détaillés pour affiner pas à pas le modèle de la commande numérique jusqu'à l'obtention d'un modèle synthétisable suffisamment détaillé pour son implantation physique.

2.2 Simulation mixte

La simulation mixte est un élément clé du flot de conception présenté figure 2. Les modèles des éléments analogiques (capteurs, interface, Electronique de Puissance, charge, ...) associés à l'environnement de simulation mixte permettent d'étudier en boucle fermée diverses configurations intégrées d'une même commande numérique.

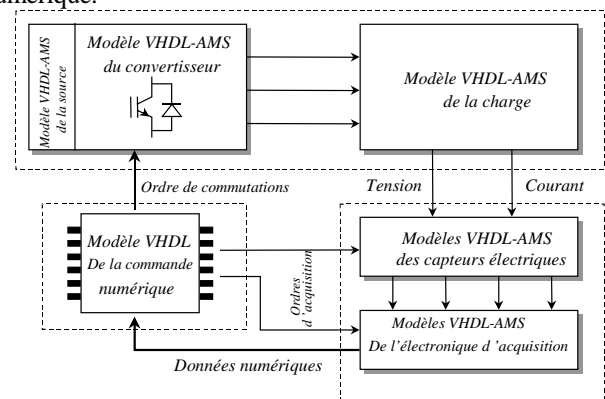


Figure 3 : Principe de la simulation mixte.

Nous proposons ici l'utilisation des outils CAO de la société Anacad pour réaliser la simulation mixte. VHDL est utilisé pour la description de la commande numérique. Les modèles des éléments analogiques et de puissance sont décrits en VHDL-AMS. La figure 3 décrit le principe général de la simulation mixte.

2.3 Flot de conception

Le flot de conception présenté figure 2 est divisé en 2 étapes principales appelées : « Spécification des propriétés numériques » et « Prototypage ».

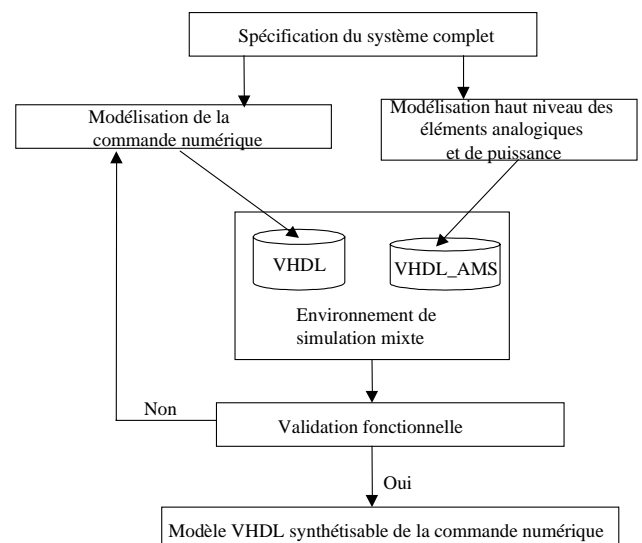


Figure 4 : Spécification des propriétés numériques.

Lors de la première étape, l'ensemble des spécifications du système de puissance est d'abord étudié et défini. Cette étape se conclut par une modélisation VHDL-AMS haut niveau de l'ensemble des éléments du système. La validation fonctionnelle de ces modèles est réalisée par simulation mixte. Ensuite, le concepteur examine l'adaptation numérique de la commande précédemment validée. Cette étape est particulièrement dédiée à l'implantation de l'algorithme de commande; on y étudie l'optimisation des calculs et détermine les propriétés numériques du futur circuit intégré. Pour un algorithme donné, le grand nombre d'implantations envisageables et de propriétés numériques à étudier rend impossible le choix du séquençement de l'algorithme et de ses propriétés numériques sans l'aide de la simulation mixte. Il faut noter que la propriété numérique prépondérante à étudier est le format binaire spécifique de codage des données.

Ensuite, un modèle VHDL synthétisable est décrit au niveau RTL (Register Transfer Level). Ce dernier correspond à l'architecture finale de l'ASIC. Diverses descriptions structurales peuvent être aisément et rapidement testées en réutilisant des fonctions ou blocs précédemment décrits en VHDL et validées lors de projets antérieurs. Cette étude permet de réduire la surface de silicium et d'optimiser le temps de traitement de l'algorithme. A chaque pas de cette étape, chaque modèle VHDL de la commande numérique est validé par simulation mixte en réutilisant les modèles VHDL-AMS analogiques et de puissance décrits initialement lors de la spécification du système. Cette première étape est détaillée sur la figure 4.

A ce niveau du flot de conception et à partir du modèle VHDL synthétisable final, le concepteur peut cibler soit un FPGA, soit placer et router un ASIC précaractérisé. L'avantage majeur de l'utilisation du langage VHDL réside dans la possibilité offerte au concepteur de cibler différents circuits physiques à partir d'une même description VHDL au niveau RTL sans avoir à re-spécifier ou reconsidérer les spécifications de la commande numérique.

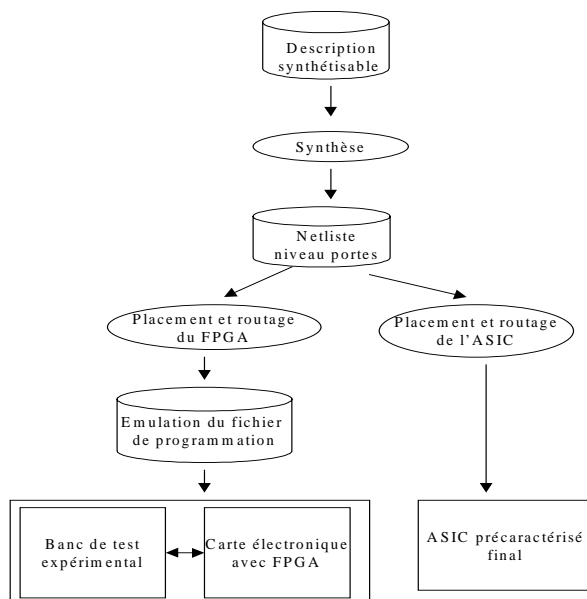


Figure 5 : Prototypage.

L'étape finale ou « Prototypage » consiste en l'implantation physique (figure 5). Le prototype est destiné

à être testé expérimentalement. La synthèse, le placement et le routage du modèle VHDL final au niveau RTL peuvent par exemple être réalisés à l'aide des outils de conception CAO de la société Cadence dans le cas d'un ASIC précaractérisé. Cependant, un délai important s'écoule entre l'envoi en fonderie du circuit ASIC et sa réception par le concepteur. Pour réduire ce délai et bien qu'il ne permette ni d'évaluer la fréquence maximale de fonctionnement de l'ASIC final, ni sa surface de silicium, un prototypage sur cible FPGA est préféré de par la reprogrammabilité intrinsèque au composant FPGA. Dans ce cas, des outils CAO tels que « Synopsys Synthesis Design » peuvent être avantageusement utilisés pour générer la « netlist » nécessaire à la programmation du FPGA ciblé. Ensuite, ce FPGA peut être placé, routé et immédiatement programmé. Une carte électronique générale destinée aux tests sur FPGA de diverses commandes numériques intégrées est en cours de finalisation au sein du laboratoire; elle met en œuvre un composant FPGA de la famille Altera, un Flex10K200, comportant environ 200.000 portes logiques. Finalement, ce FPGA une fois programmé peut ainsi être testé dans son contexte de puissance réel.

3. Un exemple d'implantation : commande numérique intégrée d'un filtre actif monophasé

3.1 Le filtrage actif parallèle

Dans cette partie, nous décrivons l'intégration de la commande numérique d'un onduleur monophasé de tension destiné au filtrage actif parallèle de réseaux électriques. Le principe de base du filtrage actif parallèle est de compenser en temps réel les courants harmoniques générés par des charges dites polluantes ou non linéaires. Ces harmoniques interfèrent avec les autres équipements électroniques connectés au réseau ainsi pollué et le filtrage actif parallèle est un moyen de compenser ces courants polluants.

La commande présentée ici a été développée et validée au sein du laboratoire GREEN de l'Institut National Polytechnique de Lorraine lors de travaux de thèse de doctorat [6]. Les travaux faisant l'objet de cet article ne sont pas destinés à l'étude ou l'optimisation de cette commande. Ils concernent son implantation intégrée sur cible numérique selon la méthodologie décrite précédemment.

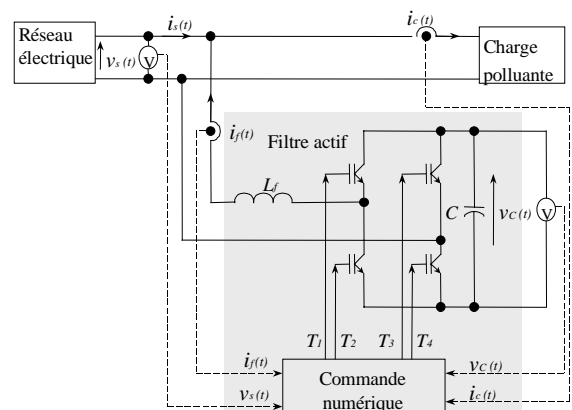


Figure 6 : Filtrage actif parallèle monophasé.

Le filtre actif étudié est constitué d'un onduleur à IGBT (Insulated Gate Bipolar Transistor), d'un condensateur de stockage C et d'une inductance L_f (figure 6). Les commutations des IGBT sont déterminées afin de compenser en temps réel les courants harmoniques. L'algorithme de commande du filtre actif monophasé est décrit figure 7 [6].

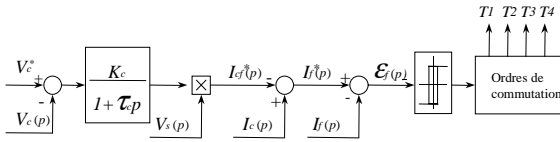


Figure 7 : Algorithme de commande.

3.2 Modélisation comportementale à partir des équations différentielles ordinaires.

A partir de l'algorithme de commande figure 7, nous définissons les équations différentielles suivantes :

$$i_{cf}^*(t) + \tau_c \frac{di_{cf}^*(t)}{dt} = v_s(t) \cdot [k_c (V_c^* - v_c(t))] \quad (1)$$

$$i_f^*(t) = i_c(t) - i_{cf}^*(t) \quad (2)$$

$$\zeta_f(t) = i_f^*(t) - i_f(t) \quad (3)$$

A partir de ces équations, nous avons écrit un modèle VHDL comportemental haut niveau de la commande numérique à implanter. De plus, nous avons écrit des modèles VHDL-AMS haut niveau des éléments analogiques et de puissance. Voici à titre d'exemple la description VHDL-AMS d'un des capteurs de courant (idéal), présentée figure 8.

```

LIBRARY DISCIPLINES;
USE DISCIPLINES.ELECTROMAGNETIC_SYSTEM.ALL;

ENTITY CAP_COURANT IS
  GENERIC ( RED: REAL);
  PORT (
    TERMINAL A : ELECTRICAL;
    TERMINAL B : ELECTRICAL;
    QUANTITY C_OUT: OUT REAL );
END ENTITY CAP_COURANT ;

ARCHITECTURE CAP_COURANT_BHV0 OF CAP_COURANT IS
  QUANTITY V_CAPTEUR1 ACROSS I_CAPTEUR1 THROUGH
  A TO B;
  BEGIN
    V_CAPTEUR1 == 0.0;
    C_OUT == I_CAPTEUR1/RED;
  END ARCHITECTURE CAP_COURANT_BHV0;

```

Figure 8 : Modèle VHDL-AMS d'un capteur de courant.

Le code VHDL associé aux équations (1) à (3) est donné figure 9. Il met en œuvre le format réel. Les figures 8 et 9 attestent des nombreux intérêts du langage VHDL-AMS dans notre contexte d'étude et d'intégration : description temporelle de phénomènes continus, possibilité de modélisation à un niveau d'abstraction élevé et simplicité lors de la description et la simulation en boucle fermée

d'une commande numérique destinée à être intégrée dans un contexte analogique et de puissance.

```

LIBRARY DISCIPLINES;
USE DISCIPLINES.ELECTROMAGNETIC_SYSTEM.ALL;

ENTITY COMMANDE_FILTRE_ACTIF IS
  GENERIC(
    V_INI: REAL;
    TC: REAL;
    KC: REAL;
    HYS: REAL;
    T: REAL;
    RED_COURANT : REAL;
    RED_TENSION : REAL);

  PORT (
    SIGNAL RST: IN BIT;
    SIGNAL CLK: IN BIT;
    QUANTITY V_C: IN REAL;
    QUANTITY V_S: IN REAL;
    QUANTITY I_C: IN REAL;
    QUANTITY I_F: IN REAL;
    SIGNAL T1: OUT BIT;
    SIGNAL T2: OUT BIT;
    SIGNAL T3: OUT BIT;
    SIGNAL T4: OUT BIT );

END ENTITY COMMANDE_FILTRE_ACTIF ;
ARCHITECTURE COMMANDE_FILTRE_ACTIF_BHV0 OF
  COMMANDE_FILTRE_ACTIF IS
  SIGNAL EPTMP: REAL ;
  QUANTITY ICF : REAL ;
  BEGIN
    ICF+TC * ICFDOT == V_S * (KC * (V_INI-V_C));
    P1: PROCESS(CLK)
    BEGIN
      IF CLK = '1' AND CLK'EVENT THEN
        IF RST = '1' THEN
          EPTMP <= (-L_F+I_C-ICF) ;
          IF EPTMP>HYS THEN
            T2 <= '1';T3 <= '1';
          ELSE
            T2 <= '0';T3 <= '0';
          END IF;
          IF EPTMP < -HYS THEN
            T1 <= '1';T4 <= '1';
          ELSE
            T1 <= '0';T4 <= '0';
          END IF;
        ELSE
          EPTMP <= 0.0;
        END IF;
      END IF;
    END PROCESS P1;
  END ARCHITECTURE COMMANDE_FILTRE_ACTIF_BHV0;

```

Figure 9 : Modélisation haut niveau de la commande.

Le système de puissance complet a été simulé à l'aide de l'outil AdvanceMS de la société Anacad pour les paramètres définis tableau 1, dans le cas d'une charge polluante de type PD2 à diodes alimentant une charge (R, L) série.

Simulation Mixte :	
Tension de référence Vc	= 500 V.
Bande d'hystérésis	= 0.5 A
C	= 22 mF, L _f = 5 mH,
T _e	= 10µs, τ _c = 0.0001 s, K _c = 0.02
R	= 40 Ω, L = 110 mH

Tableau 1 : Paramètres de simulation.

Les résultats de simulation en boucle fermée sont présentés figure 10 : de haut en bas, on visualise le courant dans la charge $i_c(t)$, le courant fourni par le filtre $i_f(t)$ et le courant de source $i_s(t)$.

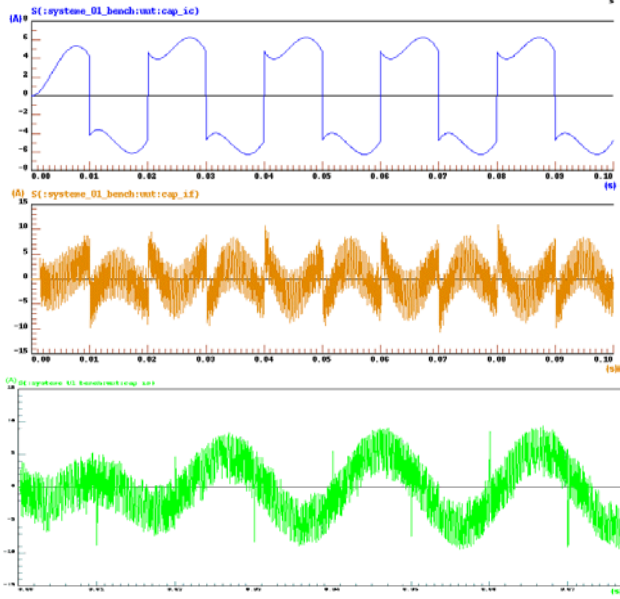


Figure 10 : Résultats de simulation.

Après une phase transitoire d'environ 6 ms, un courant de compensation $i_f(t)$ s'établit en régime permanent et permet de réduire le Taux de Distorsion Harmonique en courant de 14% à 3%. Il est important de noter que le temps de simulation associé aux résultats présentés est très réduit du fait de la modélisation haut niveau. Dans un premier temps, ces résultats de simulation valident les modèles VHDL-AMS haut niveau des éléments analogiques et de puissance qui vont être réutilisés tout au long du flot de conception. De plus, ils valident le principe de commande lui-même.

3.3 Modélisation comportementale à partir des équations échantillonnées

A partir des équations (1) à (3), nous définissons les équations échantillonnées suivantes à l'instant $k.T_e$, où T_e représente la période d'échantillonnage :

$$I_{cf}^*(k.T_e) = V_s(k.T_e) \cdot \left[\frac{T_e}{T_e + \tau_c} \cdot k_c (V_c^* - V_c(k.T_e)) \right] + \frac{\tau_c}{T_e + \tau_c} \cdot I_{cf}^*((k-1).T_e) \quad (4)$$

$$I_f^*(k.T_e) = I_c(k.T_e) - I_{cf}^*(k.T_e) \quad (5)$$

$$\zeta_f(k.T_e) = I_f^*(k.T_e) - I_f(k.T_e) \quad (6)$$

Dans ces équations, deux paramètres génériques, notés red-tension et red-courant, interviennent. Ils sont appliqués aux signaux analogiques en sortie des capteurs de courant et de tension pour travailler en grandeur réduite. Ainsi, l'amplitude des grandeurs à l'entrée des convertisseurs analogiques-numériques appartient à l'intervalle $[-1,1]$. A partir des équations (4) à (6), nous avons écrit une nouvelle « architecture » pour l'« entité » précédente « commande-filtre-actif » (figure 9). Cette dernière utilise toujours le format réel et est décrite figure 11.

```

ARCHITECTURE COMMANDE_FILTRE_ACTIF_BHV1 OF
COMMANDE_FILTRE_ACTIF IS
SIGNAL EPTMP : REAL ;
SIGNAL ICF : REAL := 0.0 ;

BEGIN

    P1: PROCESS(CLK)
    BEGIN
        IF CLK='1' AND CLKEVENT THEN
            IF RST='1' THEN
                ICF <= (((RED_TENSION *
                RED_TENSION * V_S*T*KC*(V_INI/RED_TENSION -
                V_C))/RED_COURANT+TC*ICF)/(TC+T));
                EPTMP <= (-I_F+I_C-ICF) *
                RED_COURANT ;
                IF EPTMP>HYS THEN
                    T2 <= '1';T3 <= '1';
                ELSE
                    T2 <= '0';T3 <= '0';
                END IF;
                IF EPTMP < -HYS THEN
                    T1 <= '1';T4 <= '1';
                ELSE
                    T1 <= '0';T4 <= '0';
                END IF;
            ELSE
                EPTMP <= 0.0;
            END IF;
        END IF;
    END PROCESS P1;

END ARCHITECTURE COMMANDE_FILTRE_ACTIF_BHV1;

```

Figure 11 : Modélisation de la commande.

Ainsi, dans l'environnement de simulation précédemment mis en place et en remplaçant l'architecture appelée « BHV_0 » (figure 9) par l'architecture appelée « BHV_1 » (figure 1), nous obtenons rapidement des résultats de simulation d'allures générales similaires à celles de la figure 10.

3.4 Modélisation structurelle au format réel

Nous cherchons maintenant à étudier et valider une architecture physique de la commande numérique tout en utilisant le format réel pour une optimisation du temps de simulation et de développement.

D'abord, bien que la mesure de quatre grandeurs électriques soit nécessaire à la commande étudiée, le nombre de convertisseurs analogiques-numériques à mettre en œuvre a été optimisé et réduit à un seul grâce à l'utilisation d'un multiplexeur analogique rapide. De plus, la séquence de traitement des calculs a été étudiée et choisie afin de réduire le nombre d'opérateurs et de registres. La multiplication exige généralement une surface de silicium importante lors de son implantation ; ainsi, une seule multiplication est effectuée à chaque pas de calcul. De même, nous avons limité à une multiplication et une addition/soustraction les opérations se déroulant en parallèle.

L'architecture résultante met en œuvre un multiplieur, un additionneur/soustracteur, quatre multiplexeurs, une mémoire, dix registres, un diviseur de fréquence et une machine à états finis. Sa description structurelle a été simulée avec succès en boucle fermée.

3.5 Modélisation structurelle au format binaire spécifique

L'objectif final de cette partie est de décrire au niveau RTL l'architecture validée lors de la précédente étape. Les descriptions précédentes des différents blocs de l'architecture ne sont pas toutes synthétisables. Celles du multiplieur et de l'additionneur/soustracteur doivent être en l'occurrence modifiées et rendues synthétisables. Ceci nécessite l'utilisation d'un format binaire spécifique de codage des données.

Le choix d'un format flottant a été écarté : ce dernier induit un temps d'exécution relativement élevé lors des opérations mathématiques et une très grande surface de silicium. Ainsi, nous avons retenu un format binaire spécifique fixe, en code complément à deux, noté $[s, m_i, m_d]$. s désigne le bit de signe, m_i le nombre de bit de la partie entière et m_d celui de la partie décimale. Les valeurs de m_i et m_d peuvent être définies séparément. Pour notre application, la valeur de m_i a été fixée à 3 bits compte tenu de la valeur maximale à coder pour le séquençement de l'algorithme retenu. La valeur de m_d peut être déterminée à partir d'un critère de minimisation des erreurs d'estimation dues aux calculs numériques. Ces erreurs ne peuvent pas être déterminées théoriquement avec précision ; nous avons choisi de mener cette étude par simulation mixte et avons estimé pour différentes valeurs de m_d et pour un convertisseur analogique-numérique parfait la différence instantanée entre le courant $i_s(t)$ obtenu figure 10 et celui obtenu avec le format $[s, 3, m_d]$. Le format $[s, 3, 16]$ a ainsi pu être retenu.

Nous avons ensuite étudié par simulation l'influence du format du convertisseur analogique-numérique réel sur la précision d'estimation de $i_s(t)$ en considérant les deux formats classiques 10 et 12 bit. Le cas 10 bit a conduit à une enveloppe non sinusoïdale pour $i_s(t)$ et nous avons donc choisi d'utiliser un convertisseur 12 bit.

Finalement, la description VHDL au niveau RTL de la commande numérique du filtre actif a été simulée et validée en boucle fermée dans le cas d'un convertisseur 12 bit pour le format $[s, 3, 16]$.

4. Prototypage sur FPGA

Nous avons conçu une carte de test modulaire pour valider sur cible FPGA la description VHDL synthétisable de la commande numérique. Nous nous sommes attachés à réaliser une carte à usage général et non pas une carte spécifique à notre application. Afin de pouvoir également tester le futur ASIC précaractérisé, nous avons réalisé une carte d'interface commune. La carte FPGA met en œuvre un composant de type Flex10K 200 de la famille Altera ; son test est en cours de finalisation. Elle permettra de valider dans des conditions expérimentales réelles l'intégration de la commande numérique du filtre actif.

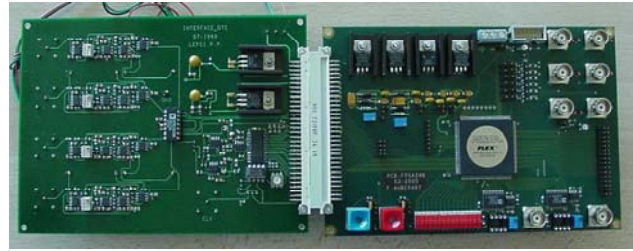


Figure 11 : Cartes de test incluant interface et FPGA.

5. Conclusion

Dans cet article, nous avons présenté une méthodologie Top-Down basée sur l'utilisation du langage VHDL-AMS en vue de l'intégration de commandes numériques de dispositifs d'électronique de puissance. Nous proposons de travailler dans un environnement unique pour la modélisation, la simulation, la validation fonctionnelle et la conception du futur circuit intégré dans son contexte réel analogique et de puissance. La méthodologie proposée optimise la durée de développement et propose une solution face aux difficultés rencontrées lors de la conception de circuits intégrés numériques pour la commande de dispositifs de puissance. De plus, les modèles utilisés pour les divers composants analogiques et numériques une fois validés deviennent des « briques de base » pour de futurs développements. La méthodologie a été illustrée par un exemple dans le domaine du filtrage actif parallèle.

L'intégration de commandes numériques sur ASIC est un premier pas vers l'intégration mixte sur « système sur puce » de l'ensemble du système de commande (figure 1). Un telle puce pourrait à plus long terme intégrer non seulement la partie numérique mais également les capteurs de courant [7] et les drivers des IGBT.

Références

- [1] LE HUY, H.: 'Microprocessors and Digital ICs for Motion Control', *Proceedings of the IEEE*, 1994, **82** (8) pp. 1140-1163.
- [2] AUBEPART, F., POURE, P., CHAPUIS, Y. and BRAUN, F.: 'HDL based methodology for VLSI design of AC motor controller', accepted for publication in IEE Proceedings of Circuits, Devices and Systems, to be published in February 2003.
- [3] RIESGO, T., TORROJA, Y. and DE LA TORRE, E.: 'Design methodologies based on Hardware Description Languages', *IEEE Trans. On Ind. Elec.*, 1999, **46** (1) pp. 3-12.
- [4] 'IEEE Standard VHDL Language Reference Manual', IEEE std 1076-1993.
- [5] AOUNI, A., CIRSTEANU, M., Mc CORMICK, M. AND DINU, A.: 'Vector controlled induction motor drive modeling using VHDL', *Computer aided control systems design IFAC symposium*, 2001, Salford, pp.265-270
- [6] XU J., "Filtrage Actif des Harmoniques des Réseaux de Distribution d'Electricité", *ph. D Thesis*, 1994, Institut National Polytechnique de Lorraine, France.
- [7] FRICK V., HEBRARD L., POURE P., ANSTOTZ F., BRAUN F., "CMOS Microsystem Front-End for MicroTesla Resolution Magnetic Field Measurement", accepted for publication in "Analog Integrated Circuits and Signal Processing".