

La synthèse de haut niveau d'un modem digital intelligent

Abdellatif Mtibaa & Mohamed Abid

Laboratoire d'Automatique et d'Informatique Industrielle
Ecole Nationale d'Ingénieurs de Monastir
Rue Ibn Jazzar 5019 Monastir - Tunisie -
Tél. +216 3 464 703, Fax +216 3 461900

RÉSUMÉ

Le papier discute l'intégration des applications complexes dans des circuits à partir des spécifications de haut niveau. Il s'agit de la conception d'un MODEM digital intelligent utilisant la synthèse de haut niveau. Le flot de conception adopté permet, à partir des spécifications de haut niveau indépendantes de la technologie, l'étude de plusieurs alternatives de réalisations, le choix rapide des solutions, la réduction des erreurs pendant les premières phases de la conception et de la réutilisation des modules déjà synthétisés ou existants. Le temps de conception est réduit par l'automatisation des différentes phases de conception en utilisant le même langage de description (VHDL) à tous les niveaux.

ABSTRACT

This paper discusses the integration of complex applications in a circuits from high level specifications. It consists on the design of an intelligent digital MODEM using the high level synthesis. The adapted design flow allows, from high level specifications independent of the technology, the study of many realizations alternatives, the fast choice of solutions, the errors reduction during the first design steps and the reusing of already existant or synthesized modules. The design time is reduced by automatization of different design steps by using the same description language (VHDL) in all levels.

1 Introduction

Les MODEM (ETCD) sont des dispositifs qui poursuivent leur progression, grâce à leurs atouts de sécurité et de facilité d'utilisation. Ces derniers sont les éléments moteurs des chaînes de transmission de données. Ils sont destinés à mettre en forme l'information pour sa transmission sur la ligne analogique. Cette fonction est réalisée le plus souvent par modulation-démodulation d'un signal auxiliaire porteur. Notre choix de cette application se justifie par les multiples facettes que posent certains constituants. Elle intègre de plus en plus de fonctionnalité par la multitude des recommandations UIT-T.

La compétitivité du développement d'un ETCD nécessite des méthodes nouvelles qui permettent de maîtriser la complexité, l'implantation de nouvelles recommandations, d'améliorer les performances du système et d'assurer une évolutivité vis-à-vis des changements technologiques.

Afin d'évaluer l'apport des techniques nouvelles de conception et de ses outils, nous avons adopté une approche méthodologique reposant sur la synthèse de haut niveau, appelée encore synthèse architecturale ou comportementale. Le but visé dans cette démarche est tout d'abord d'élever de plus en plus le niveau d'abstraction de la conception afin de permettre la production d'une série de circuits à partir des spécifications les plus proches du cahier des charges et de favoriser la maintenabilité et la réutilisabilité. Le domaine de la compilation de comportement, bien que récent, a déjà

fait l'objet d'une littérature abondante, qui reflète un grand besoin dans ce domaine. De nombreux systèmes de synthèse de haut niveau existent. Il s'agit essentiellement d'outils universitaires qui ont atteint la maturité et qui commencent à émerger dans l'industrie [3].

2 Démarche et environnement

La démarche de conception, reposant sur la synthèse de haut niveau (SHN), et l'environnement retenus pour notre méthodologie sont présentés par la figure 1. Ce flot présente trois étapes essentielles, à savoir : spécification, synthèse et implantation physique. Ces étapes peuvent être divisées en phases de conception proprement dites et en étapes de validation. Les phases de validation sont essentielles afin d'assurer que le circuit physique conçu possède bien les propriétés désirées, quelle que soit la qualité des outils de synthèse utilisés [9]. L'étape de la synthèse débute par une synthèse de haut niveau (appelée encore synthèse architecturale ou comportementale) [4, 6]. Celle-ci permet, à partir d'une description comportementale, la génération d'une architecture au niveau transfert de registres (RTL) composée d'un contrôleur et d'un chemin de données. L'environnement utilisant l'outil de synthèse AMICAL du groupe SLS du TIMA [7, 5], est utilisé pour la synthèse des descriptions VHDL [8] comportementales. Il combine la synthèse comportementale avec une méthodologie permettant la réutilisation des modules existants (par exemples des UAL ou plus complexes comme des contrôleurs ou des DSP). Commencant par une description

initiale, la synthèse avec cet outil se compose de plusieurs boucles de raffinements permettant d'explorer les différentes solutions d'architectures. Lorsqu'une solution satisfaisante est obtenue, la synthèse au niveau plus bas peut être commencée. De plus, en utilisant l'expérience acquise dans la spécification, le concepteur peut facilement améliorer sa description comportementale et recommencer une autre boucle de synthèse architecturale. En plus, l'outil AMICAL offre une interactivité avec le concepteur, ce qui permet de bien optimiser l'architecture du circuit. La synthèse logique permet de générer une description structurelle, généralement une liste d'interconnexion d'éléments physiques de base (portes, bascules) décrits dans la bibliothèque de la technologie cible adoptée par le concepteur, à partir d'équations booléennes [Cebe95]. L'étape de synthèse physique peut être utilisée aussi bien pour les ASIC que pour les circuits programmables FPGA ou CPLD [2, 10].

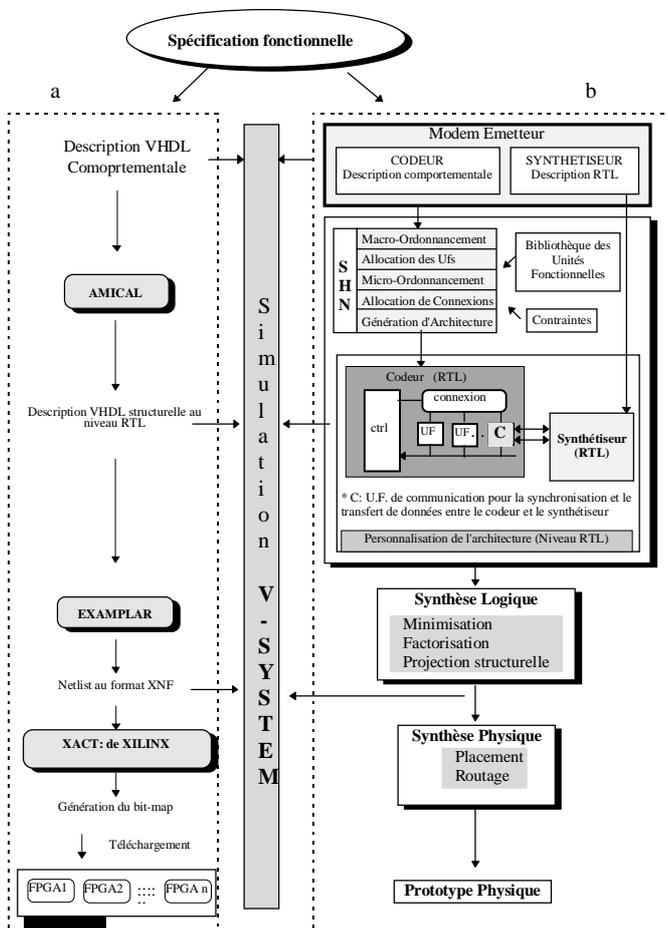


Figure 1 : Flot de conception poursuivie et outils de synthèse et de simulation utilisés

3 Spécification du MODEM

Nous nous intéressons à la réalisation des systèmes électroniques complexes, d'une façon purement matérielle. En effet et d'une manière générale, un système se traduit par un circuit, mais dans le cas des systèmes complexes, il peut s'agir d'une association de circuits (processeur). Le problème du découpage en plusieurs circuits se pose. Cette décomposition est appelée décomposition fonctionnelle ou partitionnement architectural ou encore conception

architecturale [1, 7]. Elle se fait sous forme d'un sous-ensemble de modules qui peuvent être spécialisés ou tous identiques, et qui peuvent être synchrones ou asynchrones. Ces modules sont couplés entre eux par un réseau d'interconnexions pour les échanges d'informations. Ils sont aussi couplés avec l'environnement par les entrées et sorties du système.

Les travaux de spécification appliqué au MODEM ont servi à élaborer une formalisation des exigences textuelles exprimées par la multitude des normes UIT-T. Au cours de cette étape, nous avons décrit toute la fonctionnalité de l'ETCD ainsi que l'interactivité avec son environnement. L'étape du partitionnement architectural est réalisée manuellement au niveau système. Une vue du système partitionné est montrée par la figure 2.

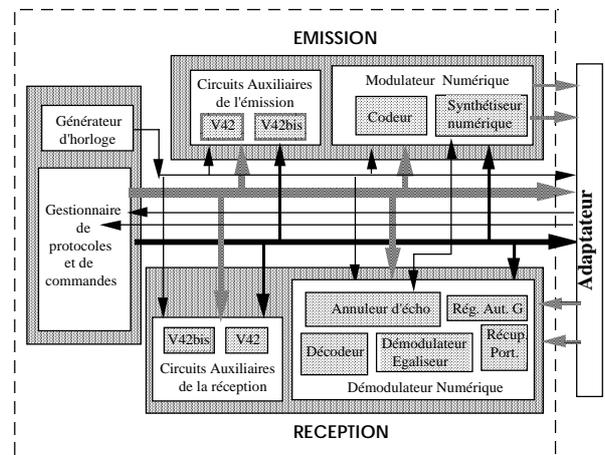


Figure 2 : Décomposition fonctionnelle de l'ETCD multifonctions

Nous nous intéressons dans ce travail à la partie Modulateur numérique (multi-recommandations). Ce dernier permet la réalisation de deux tâches concurrentes et fonctionnant à des vitesses différentes : le codage et la modulation. Par conséquent, cette partie a été modélisée par deux "process". Le codeur est synthétisé à partir d'une description VHDL comportementale. Il réalise les fonctions de l'embrouillage, du codage à q-aires, du codage différentiel, du codage convolutif et de l'affectation spatiale du signal de constellation. Il fait appel au module "synthétiseur numérique" qui constitue un module prédéfini et qui réalise numériquement la modulation de fréquence, de phase et la modulation combinée en amplitude et en phase.

4 Synthèse architecturale de l'émetteur

Le processus de la synthèse architecturale est formé par trois grandes étapes qui sont l'ordonnancement, l'allocation et la génération de l'architecture.

L'outil de synthèse architecturale adopté dans notre méthodologie, AMICAL, commence par deux types d'informations : la description du module en VHDL comportementale et une bibliothèque externe d'unités fonctionnelles. Ces unités fonctionnelles personnalisent les fonctions et les procédures utilisées dans le code VHDL. Pendant les différentes étapes de synthèse de haut niveau, ces unités fonctionnelles sont considérées comme des boîtes noires. L'architecture générée au niveau transfert de

registres (RTL) est composée d'un contrôleur et d'un chemin de données. Cette architecture est à base de bus ou de multiplexeur. Après comparaison des deux types d'architecture, nous choisissons l'architecture optimale.

Dans notre cas, le codeur est synthétisé à partir d'une description VHDL comportementale. Il fait appel au module "synthétiseur numérique" qui constitue un module prédéfini (description RTL). L'environnement utilisé permet donc de considérer l'abstraction de ce module. Il est introduit dans le flot de synthèse architecturale comme le montre la figure 1b.

Après la simulation comportementale, la synthèse architecturale et l'exploration architecturale des différents styles de descriptions et de synthèses avec des contraintes variées (bus, multiplexeurs), les résultats de la solution retenue sont donnés par les figures 3 et 4.

La description comportementale de départ comporte 203 lignes de code VHDL. L'architecture générée automatiquement par AMICAL est décrite par 3067 lignes de code VHDL au niveau RTL. La synthèse architecturale génère une architecture composée par un contrôleur comprenant 58 états et une machine d'états finis de 153 transitions. Le contrôleur commande la partie opérative à travers 82 lignes de contrôles. La partie opérative, obtenue automatiquement, est constituée par 4 unités fonctionnelles (Embrouillage, Codage convolutif, ALU, Opérations sur les registres), de deux unités de communication (gestion et commande de l'EPROM de séquences et l'interface avec le synthétiseur numérique), de 28 registres, de 7 flags registres et de 19 multiplexeurs. La surface générée, après une synthèse logique automatique est de 2408 portes et 2242 bascules.

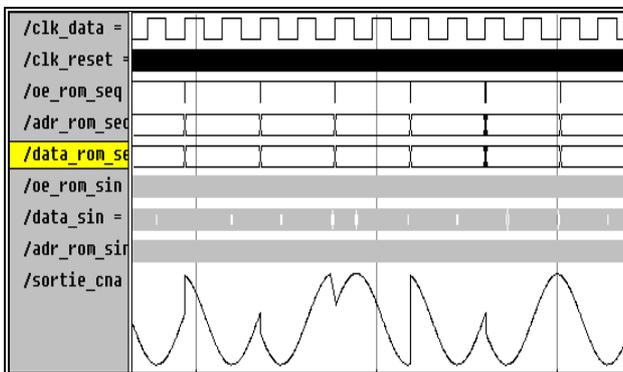


Figure 3 : Copie d'écran du résultat de la simulation pour la génération du signal analogique de ligne.

5 Conclusion

Ce papier a présenté la démarche méthodologique et l'environnement utilisés pour la conception de haut niveau des systèmes complexes. Cette démarche se repose sur la synthèse de haut niveau. Au cours de cette étape de synthèse, en plus de la description comportementale, nous avons fait appel à la réutilisation d'un module existant modélisé et synthétisé au niveau RTL. Pour la généralisation de cette notion de réutilisation, Nous pouvons donc introduire des composants déjà synthétisés en comportementale soit à une description RTL ou même, à d'autres composants qui sont synthétisés par d'autres outils

de synthèse comportementale.

En plus, cette démarche nous a permis la validation de la spécification et la réduction des erreurs de conception et de modélisation pendant une phase avancée de la conception.

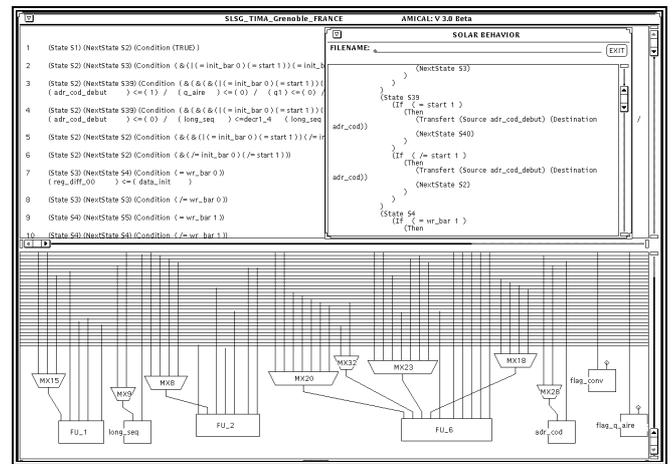


Figure 4 : Copie d'écran du résultat de la SHN.

6 Remerciements

Le présent travail rentre dans le cadre de la conception et l'intégration des systèmes de communications complexes. Il s'inscrit dans le cadre d'un projet de coopération " Réseaux Formation-Recherche Franco-Tunisien" entre le groupe SLS du TIMA de l'INPG et l'ENIM - **TUNISIE**.. A cet effet, nous tenons à remercier Mr AA Jerraya, Chef de groupe SLS.

7 Références

- [1] Calvez J.-P. "Spécification et conception des ASICs", Masson, Paris 1993.
- [2] Galelio "HDL Synthesis Manuel", Exemplar logic INC. 815 Atlantic, 1995.
- [3] Courtois B. "CAD and Testing of ICs and Systems : Where are we going ?", Journal Of Microelectronic Systems Integration, Vol. 2 N°3, 1994.
- [4] Gajski D.D., Dutt N.D., Wu A.C-H. and Lin S.Y-L. "High-Level Synthesis, Introduction to Chip and System Design", Kluwer Academic Publishers, 1992.
- [5] Jerraya A.A., Ding H., Kission P. and Rahmouni M. "Behavior synthesis and component reuse with VHDL", Kluwer Academic Publishers, 1996.
- [6] Kission P., Ding H., Jerraya A.A. "VHDL Based Methodology for hierarchy and component re-use at the behavior Level", EuroDAC/EuroVHDL, 1995.
- [7] Kission P. "Exploration de la hiérarchie et la réutilisation de blocs existant par la synthèse de haut niveau", Thèse de l'INPG, TIMA Grenoble 1996.
- [8] IEEE "VHDL Language Reference Manual", Standard 1076/B, IEEE, Juin 1987.
- [9] Model Technology "V-System/Windows, User's Manual : VHDL, Simulation for Pcs Running Windows & Windows NT", Ver 4.3, June 1995
- [10] Xilinx Inc. "User guide : Hardware & peripherel", Octobre 1995.