
RESUME

Les premiers compilateurs de silicium étaient principalement de deux types :

- ceux qui ont pour point d'entrée une description structurelle,
- ceux qui ont pour fichier d'entrée une description comportementale.

L'utilisation d'un compilateur du premier type permet d'obtenir un système VLSI dont l'architecture peut être quelconque. Néanmoins, la conception de cette architecture est à la charge de l'utilisateur dans la mesure où la compilation est conduite à partir d'une représentation structurelle.

Un compilateur du deuxième type permet de générer un système VLSI dont l'architecture est pré-spécifiée. Cependant, l'avantage de tels types de compilateurs est que les efforts de l'utilisateur sont concentrés non pas sur la conception d'architectures, mais plutôt sur la conception de systèmes.

Les travaux de recherche actuels dans ce domaine visent la conception de compilateurs ayant les avantages de ceux des deux types, c'est à dire conduire la compilation à partir d'une description comportementale tout en générant des circuits à caractéristiques différentes (variées). Ainsi, il est possible d'introduire des contraintes de surface, de vitesse et de consommation de puissance afin de guider le processus de synthèse. En outre, les technologies actuelles permettent d'intégrer tous les composants du système sur la même puce (*system on chip*), ce qui engendre des problèmes de recherche supplémentaires (la conception de systèmes monopuce requiert un savoir pluridisciplinaire allant du logiciel aux phénomènes physiques des technologies submicroniques : développement d'APIs, de systèmes d'exploitation en temps réel, de pilotes pour des périphériques aussi variés que nombreux, conception de circuits mixtes numérique/analogique, de circuits RF, résolution des phénomènes électro-thermiques, etc ...). La prise en considération de tous ces problèmes nécessitant alors toute une compagnie, le projet a été cerné à l'aspect *Synthèse Hardware de systèmes digitaux à intégrer sur la même puce*. Toutefois, l'architecture à générer permet à plusieurs processeurs et mémoires de coexister avec la partie matérielle du système, ce qui permet de paralléliser même la partie Software et d'anticiper sur la possibilité d'intégrer une partie logicielle. Les grandes étapes de ce travail sont les suivantes :

- Génération d'une structure de données à partir d'une description en SYSTEMC pour la synthèse du compilateur
- Détermination du graphe des tâches
- Décomposition du système en sous-systèmes VLSI en tenant compte du compromis surface – vitesse – consommation de puissance
- Synthèse des tâches
- Définition du protocole de communications régissant les sous-systèmes
- Synthèse et analyse de chaque sous-système VLSI

Enfin, notons que la plupart des problèmes traités dans cette thèse ne sont pas polynomiaux : Ils sont \sum_k^P , et les plus simples d'entre-eux sont \sum_1^P (=NP) complets.

Mots-clé : compilation de silicium, synthèse de haut niveau, systèmes monopuce, embarqués, temps réel, optimisations, contraintes de surface, vitesse, consommation de puissance, complexité algorithmique