

Déploiement optimisé d'applications sur FPGA autour d'un NoC

Matthieu PAYET - ADACSYS / Laboratoire Hubert Curien
 Virginie FRESSE - Laboratoire Hubert Curien
 Frédéric ROUSSEAU - Laboratoire TIMA
 Pascal REMY - ADACSYS



Introduction

Les réseaux sur puce (NoC) sont maintenant matures pour l'interconnexion à grande échelle d'IP (« Intellectual Property ») composant une application (ensemble de fonctions abstraites à implanter sur matériel).

Des problématiques apparaissent : compatibilité des interfaces et gestion de cette nouvelle couche de communication.

Nous présentons dans ce poster un flot complet permettant une implantation optimisée d'un NoC (une application sous forme de réseau) sur une architecture matérielle à base de FPGA.

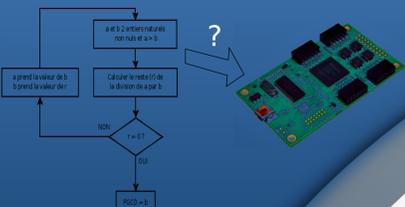
Objectif

Rendre accessible aux concepteurs l'implantation optimisée d'applications décrites sous forme de diagramme d'activités sur une plateforme à base de FPGA en profitant des avantages que nous apportent les NoC en terme d'extensibilité et de vérification.

Méthode

Problèmes :

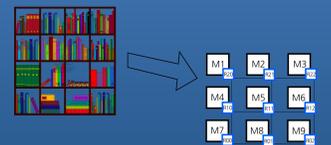
J'ai une carte basée sur un FPGA. Comment l'exploiter afin de réaliser mon application ?



L'implantation est-elle optimisée ?

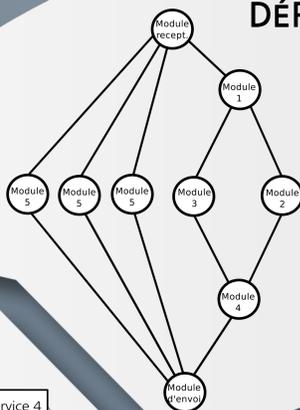
Solution proposée :

Construire une bibliothèque d'IP afin de permettre une synthèse de haut niveau d'un NoC.



Dimensionner le réseau pour qu'il consomme le maximum de ressources disponibles sur le FPGA.

DÉPENDANCES DE PAQUETS



Les modules (blocs IP) fournissant les services spécifiés (à partir d'une bibliothèque) sont sélectionnés, et les dépendances de données sont déduites de leurs interfaces.

Le graphe est ensuite développé jusqu'à utiliser le maximum de ressources disponibles.

ALGORITHME

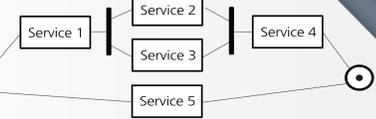
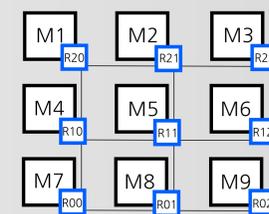


Diagramme d'activités UML 2.0

INTERCONNEXIONS DE MODULES

Une représentation des interconnexions est utilisée pour construire la topologie logique de l'application.

Un module est soit associé à un fichier VHDL ou Verilog, soit il est "abstrait", c'est-à-dire que le "top" est généré pour instancier les modules.

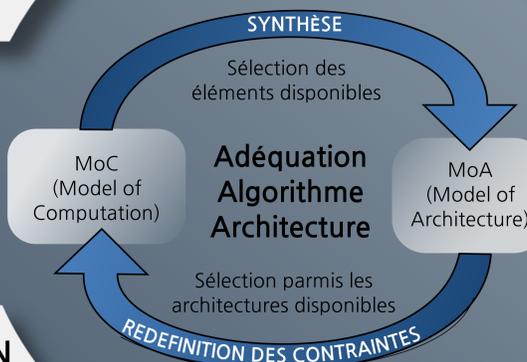


Un NoC est utilisé comme support de communication.

Bibliothèque



- Utilisation en ressources
- Sources
- interfaces



EMULATION ou SIMULATION

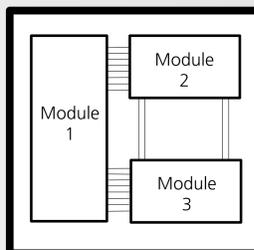
Si le fonctionnement n'est pas satisfaisant, le réseau est vérifié en simulation puis sur FPGA (AVA-Soft).

Sinon, l'implantation est terminée.



Le trafic réseau de l'application est émulé à l'aide de générateurs et récepteurs de trafic.

DESCRIPTION RTL



Le code VHDL est ensuite généré automatiquement à partir des spécifications XML.

C'est le point d'entrée du flot propriétaire d'implantation pris en charge dans l'outil développé.

Résultats

Nous avons développé un outil qui suit le flot proposé nommé YANGO (Yet Another NoC Generator and Optimizer).

La génération du code à partir du XML permet de s'affranchir de contraintes du VHDL (instanciation conditionnelle en autres).

Nous travaillons actuellement sur l'optimisation du graphe de dépendances de paquets et la génération d'adaptateurs réseau.

Conclusions

Le flot proposé assure la continuité entre les différentes synthèses à partir de la spécification algorithmique

Nous avons utilisé un support XML afin d'associer à un module ses caractéristiques physiques (utilisation en ressources, fréquence maximale).

Un boucle d'optimisation à partir des résultats d'implantation et d'émulation permet l'adéquation entre l'algorithme et l'architecture

Travail futur

Nous allons développer un NoC rendant l'extension de l'application à une architecture multi-FPGA possible. Nous utiliserons les informations de trafic dans la bibliothèque afin de vérifier l'application finale et d'ajuster les paramètres de QoS du réseau.



COMMUNAUTÉS DE RECHERCHE ACADÉMIQUE Rhône-Alpes



T.I.C. ET USAGES INFORMATIQUES INNOVANTS