

Projet FxPSynthesis :

Synthèse de circuits utilisant l'arithmétique virgule fixe

Type du projet : Action initiative

Coordinateur du projet :

Roselyne CHOTIN-AVOT, Maître de conférences, équipe CIAN

Membres du projet :

Thibault HILAIRE, Maître de conférences, équipe PEQUAN

Descriptif du Projet

Le but de ce projet est de rapprocher deux outils (développés respectivement dans les équipes CIAN et PEQUAN) qui sont finalement à utiliser à la suite l'un l'autre dans un flot de conception :

- FiPoGen est un outil développé dans le cadre de la thèse de Benoit Lopez et du projet ANR DEFIS (*Design of fixed-point embedded systems, 2011-2014*) permettant de transformer un algorithme en du code virgule fixe (plus précisément, un graphe d'opérations virgule fixe, avec calcul des erreurs commises, optimisations des largeurs des opérateurs (sous contrainte d'erreur de sortie), etc.). FiPoGen est principalement dédié à l'implantation des filtres linéaires récurrents, dans la mesure où certaines parties (évaluation de la dynamique, etc.) ne sont garanties que dans ce cadre pour le moment.
- Stratus est un outil développé dans l'équipe CIAN permettant de décrire des composants matériels numériques sous forme de générateur dont les paramètres peuvent être aussi bien matériels (largeur des données, type d'opérateur etc.) que logiciels (système de numération, contraintes etc.). Il permet notamment d'adapter facilement un composant à son contexte d'utilisation.

L'équipe CIAN développe des circuits mixtes analogique/numérique. Ces circuits sont décrits à un haut niveau d'abstraction en utilisant des réels. Lorsqu'il faut les synthétiser, il est nécessaire de trouver la bonne représentation virgule fixe, ainsi que l'enchaînement des calculs virgule fixe satisfaisant les contraintes numériques du circuit (erreur sur la sortie, par exemple), ce que permet l'outil FiPoGen. Une fois cela, le graphe de calcul est utilisé par l'outil Stratus pour générer le circuit adéquat.

La mise en commun des deux outils permettrait de générer directement un circuit numérique à partir de la description haut niveau du circuit mixte analogique/numérique, ce qui est en enjeu majeur dans le domaine de la conception mixte analogique/numérique.

Étude théorique

La mise en commun des deux outils pour proposer un flot de conception complet, nécessite tout d'abord d'identifier les verrous technologiques existants. Ces verrous peuvent être de plusieurs sortes :

- Le niveau de description du circuit initial n'est pas adapté à l'outil FiPoGen
- Le type de circuit n'est pas pris en compte par l'outil FiPoGen
- Les opérateurs utilisés ne sont pas présents dans l'outil Stratus
- Etc.

Cette identification pourra se faire à l'aide d'exemples concrets de circuits qui ont été synthétisés par l'équipe CIAN, et d'exemple académiques classiques traités dans l'équipe Pequan.

Objectifs atteints l'an passé

L'an dernier, nous avons obtenu le financement partiel d'un stage de M2 et le stagiaire que nous avons engagé nous a fait faux bon. Nous avons donc pris un élève ingénieur pendant l'été ce qui nous a obligé à revoir les objectifs à la baisse. L'intégration de FiPoGen et Stratus a néanmoins été totalement réalisée, dans le sens où Stratus peut désormais prendre en entrée le graphe de calcul obtenu par FiPoGen et générer l'architecture matérielle correspondante. Ceci a pu être validé sur l'ensemble des exemples traité par FiPoGen.

Objectifs pour l'année à venir

Même si l'intégration des outils Stratus et FiPoGen est fonctionnelle, il reste maintenant à exploiter leurs interactions possibles, et voir comment exploiter de manière conjointe les phases d'optimisation. Par exemple, lors de sa phase d'optimisation, FiPoGen devrait être capable d'interroger le moteur de Stratus afin d'obtenir une estimation des performances de l'architecture matérielle et d'en tenir compte. De plus il serait bien de pouvoir prendre en entrée une description Simulink de l'algorithme à implémenter (standard utilisé dans l'industrie).

Pour l'année à venir, nous comptons :

- Terminer la rédaction de l'article commun pour la revue IEEE Circuits&Systems
- Intégrer à FiPoGen l'importation de descriptions Simulink
- Etendre les exemples traités par FiPoGen aux circuits mixtes analogiques/numériques
- Enfin tenter d'exploiter de manière conjointe les différents degrés de liberté de chaque outil.

Il sera alors possible, partant d'un exemple concret et d'une contrainte sur la qualité numérique de l'implémentation, de générer automatiquement une implémentation matérielle, avec garantie sur la qualité numérique et sur les performances de l'implantation matérielle.

Importance du projet pour le LIP6

Ce projet est l'occasion de continuer la collaboration entre deux équipes (CIAN et PEQUAN) et de développer un outil commun au cœur de la problématique de la conception de circuits mixtes analogique/numérique. Il sera une première étape pour établir une collaboration nationale ou internationale au travers d'un projet de plus grande ampleur (projet ANR par exemple).

Il s'inscrit dans une thématique « *systèmes embarqués* » plus générale avec des applications possibles dans différents domaines, dont la santé (prothèses, implants, etc.), le traitement du signal (radio logicielle, réseau de capteur etc.). Cet outil commun pourra alors y être utilisé pour accélérer le développement d'applications.

Annexe financière

Nous présentons une demande de budget qui concerne le fonctionnement pour l'année 2015. Le financement est destiné, pour une large part, à financer un stage (sujet en annexe). Le reste du financement sera destiné à couvrir des frais de missions associés (une conférence internationale et deux déplacements nationaux, prévus pour préparer un projet de type ANR).

Type	Coût
Stage	3000€
Missions	1000€
Total	4000€

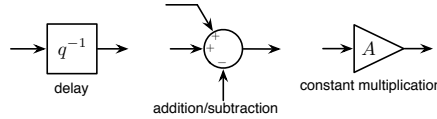
Fixed-Point implementation of Simulink block diagrams

Advisor: Thibault HILAIRE (thibault.hilaire@lip6.fr)

Context:

Simulink (or Simulink-like softwares) allows to design and simulate systems (dynamic systems, signal processing or control systems, etc.) described with by block diagrams. Block diagrams are visual representations of dynamic systems, and they consist of interconnected blocks performing some operations on the input(s) in order to produce output(s).

We focus here on linear algorithms, *ie* algorithms that only used additions/subtractions, multiplications by constants and delay operators. They can be described in block diagrams with the following blocks:



On the other hands, it has been shown that such algorithms can be exactly described with a dedicated state-space equation (denoted SIF, for Specialized Implicit Framework), that allow to represent the operations and the way they are chained:

$$\begin{pmatrix} \mathbf{J} & \mathbf{0} & \mathbf{0} \\ -\mathbf{K} & \mathbf{I}_n & \mathbf{0} \\ -\mathbf{L} & \mathbf{0} & \mathbf{I}_p \end{pmatrix} \begin{pmatrix} \mathbf{t}(k+1) \\ \mathbf{x}(k+1) \\ \mathbf{y}(k) \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathbf{M} & \mathbf{N} \\ \mathbf{0} & \mathbf{P} & \mathbf{Q} \\ \mathbf{0} & \mathbf{R} & \mathbf{S} \end{pmatrix} \begin{pmatrix} \mathbf{t}(k) \\ \mathbf{x}(k) \\ \mathbf{u}(k) \end{pmatrix}$$

where $\mathbf{u}(k)$ is the inputs vector at time k , $\mathbf{y}(k)$ the outputs vector, $\mathbf{x}(k)$ collects the states (informations stored from one step to another) and \mathbf{t} the intermediates values. The \mathbf{J} matrice is lower triangular, with 1's on the diagonal. The structure (sparsity) of the matrices $\mathbf{K}, \mathbf{L}, \mathbf{M}, \mathbf{N}, \mathbf{P}, \mathbf{Q}, \mathbf{R}, \mathbf{S}$ reflects the structure of the operations, and their coefficients the constants used.

With these framework have been developed tools and methods for the generation of fixed-point algorithms, specially *FiPoGen* and *Stratus*, both developed in the lab:

- *FiPoGen* helps to automatically produce reliable fixed-point algorithms for embedded targets and solve the tradeoff between implementation cost and numerical properties (specially for finite precision arithmetic)
- *Stratus* is a generic hardware component generator, where the operators can be parametrized by their bit width, numeration system, etc.

Subject:

The aim of this internship is to transform Simulink block diagrams into VHDL code. First, simulink block diagrams will be transformed in equivalent¹ SIF realizations, and then transformed in hardware implementation with *FiPoGen* and *Stratus* tools. The outline will be the following:

¹equivalent in the sense that it has the same input-output relationship, but also preserves the structure of the computations.

1. Study `.mdl` structure file (or `.sxl`), and generate a block diagram in Simulink (a `.mdl` or `.sxl`) from a given SIF realization;
2. Develop an algorithm that will iterate on a block diagram (on the equivalent graph) in order to aggregate the blocks and build the equivalent SIF;
3. Use *FiPoGen* to generate fixed-point algorithm from the various sum-of-products of the SIF;
4. Use *Stratus* to generate hardware implementation from fixed-point algorithm.

We will first start with basic block diagrams (where the signals are scalar), and extend it, if possible, to more complex diagrams (with vectors and some non-linear blocks).

This internship can be followed by a 6 months engineer position (CDD) at Laboratoire d'Informatique de Paris 6 (LIP6), if the student is enough motivated and the internship successful.

Prerequisites: programming skills (Python is preferable, but it can be C or C++), graph theory, linear algebra. It is not mandatory to know Simulink.