

Compte rendu de la réunion sur l'action incitative ARP-ISIS autour d'Ardoise à l'ENST, salle Grenat

Ronan KERYELL

10h-13h35, 7 juillet 1999

1 Personnes présentes

Étaient présents (par ordre alphabétique de laboratoire) :

Mohamed AKIL	A2SI-ESIEE
Stéphane GAILLARD	A2SI-ESIEE
Patrice BERTIN	CMA-ENSM
Ronan KERYELL	Département Informatique-ENSTBr
Gérald OUVRAUDOU	Département Informatique-ENSTBr
Didier DEMIGNY	ETIS-ENSEA
Dominique LAVENIER	IRISA
Daniel DOURS	IRIT
Sébastien PILLEMENT	LIRMM
E. BOURENNANE	Université de Bourgogne
GRANADO	LIS-UPMC
SHAWKY	UTC

Pour plus d'information cliquer sur les noms. Je n'ai pas réussi à retrouver tous les prénoms...

2 Ordre du jour

L'ordre du jour proposé était :

- exposé sur l'architecture actuelle de la machine par Didier DEMIGNY ;
- état des lieux logiciels de la machine ;
- applications visées ;
- faire le point sur les équipes toujours partantes ;
- partager les tâches.

3 La réunion

3.1 Exposé sur la chaîne de segmentation

Didier DEMIGNY a fait un exposé sur la chaîne de segmentation envisagée comme cible du projet. C'est un choix parmi d'autres envisageables.

Il s'appuie sur l'expérience de son laboratoire ([ETIS-ENSEA](#)) dans le domaine du traitement d'images et des architectures dans ce domaine.

Dans le cadre du GT7 une méthode de génération de matériel à partir de Signal et Syndex est utilisée.

La chaîne de segmentation consiste grossièrement en un pipeline

- 1° mémoire d'image ;
- 2° lissage, gradient et double seuillage ;
- 3° fermeture des contours ;
- 4° étiquetage.

Lissage de NAGAO : voisinage 5×5 sur lequel on passe des masques de taille 3×3 de forme bizarre. On choisit la moyenne dans le bloc contenant les pixels de plus faible variance. L'algorithme est simplifié en n'utilisant pas la variance.

Le calcul de la norme du gradient sur un voisinage 2×2 est approximée par

$$\|\vec{\nabla}\| \approx \max\left(\left|\frac{\partial}{\partial x}\right|, \left|\frac{\partial}{\partial y}\right|, \frac{3}{4}\left(\left|\frac{\partial}{\partial x}\right| + \left|\frac{\partial}{\partial y}\right|\right)\right)$$

avec une erreur en pratique inférieure à 6 %.

La fermeture des contours est effectuée par un automate cellulaire. Une simplification consiste à le faire fonctionner en mode « data-flow » (itération à la GAUSS-SEIDEL) plutôt qu'en mode SIMD pour limiter le débit mémoire.

L'étiquetage est fait par balayage horizontal et étiquetage local. Une table d'équivalence globale est utilisée pour résoudre les fusions de zones communes.

Une autre approche n'utilisant pas NAGAO est présentée : un filtre récursif de type

$$\frac{1 - \gamma}{1 - \gamma z^{-1}}$$

avec un balayage dans chaque sens appliqué dans chaque direction.

Tous ces algorithmes sont assez différents d'un point de vue organisation et le choix de la taille des paramètres a beaucoup d'importance. Dans le cadre de l'action indicative il faudrait regarder les problèmes de mémoire et d'ordonnancement.

Il faudrait avoir le programme en C de l'application complète.

À l'occasion ce serait bien d'avoir un lien vers les transparents de l'exposé...

3.2 Exposé sur l'architecture du prototype Ardoise

[Didier DEMIGNY](#) a ensuite fait un autre exposé sur l'architecture du premier prototype d'Ardoise.

L'idée d'une machine dynamiquement reconfigurable est qu'elle doit pouvoir mieux utiliser l'électronique qu'elle contient.

La première version de la machine est composée de 3 modules en pipeline sous contrôle d'un FPGA et d'une mémoire de configuration via un bus global de 16 bits et un bus global de configuration de 40 bits (24 bits d'adresse et 16 bits pour la CacheLogic d'Atmel). Le FPGA de contrôle permet de se relier à d'autres systèmes comme des DSP divers et variés.

Chaque module est composé de :

- 1 FPGA Atmel 40K20 5V ;
- 2 bancs mémoires de $256K \times 32$ avec pour chacun 1 bus de données sur 32 bits et 1 bus de d'adresses sur 18 bits ;

- 1 bus de communication de 40 bits à gauche pour le pipeline ;
- 1 bus de communication de 40 bits à droite pour le pipeline.

L'intérêt du reconfigurable :

- méthode de rotation par changement dynamique de coefficient de filtre à chaque ligne (ce serait bien d'avoir le programme de ce système¹) ;
- on peut choisir d'utiliser le meilleur algorithme à un moment donné après une phase de mesure ;
- on ne gagne pas en terme de vitesse par rapport à une implantation tout en ASIC sans contrainte de ressource.

Les performances brutes sont limitées par la taille du circuit virtuel équivalent statique qu'on peut avoir par durée de trame T (si on fonctionne de manière synchrone avec une entrée vidéo). Un circuit se programmant avec une vitesse v d'un certain nombre de portes par secondes, le nombre de portes équivalentes disponibles (c'est à dire la somme de toutes les portes de toutes les configurations) pendant une trame est bornée par :

$$P_{\text{equiv}} = T.v$$

Si on décide de travailler au niveau d'une ligne, on constate que la reconfiguration possible est beaucoup plus limitée et la taille de circuit équivalente de même.

Quelques problèmes qui se posent :

- comment utiliser les ressources ?
- exécution en parallèle ou en série ?
- parallélisme de données ?
- parallélisme de pipeline ?
- comment partitionner l'application ne serait-ce qu'approximativement ? Cf. les méthodes de [Daniel DOURS](#) à l'[IRIT](#).

À l'occasion ce serait bien d'avoir aussi un lien vers les transparents de cet exposé...

3.3 État des lieux sur les outils disponibles

Ensuite a débuté la discussion sur les outils disponibles actuellement autour d'Ardoise.

Pour l'instant il n'y a que la chaîne de développement de FPGA du constructeur (Atmel).

Pour l'instant il n'y a pas eu trop de travail effectué en méthodologie :

- le pôle [ISIS](#) de l'action a avancé depuis 1 an (conception de la machine) ;
- le pôle [ARP](#) de l'action n'a pas démarré : attente des applications et du matériel.

Il n'y a plus qu'un an....

Les outils qui sont nécessaires à court terme :

- système de chargement de configurations ;
- bas niveau : placement des bits de configuration,... ;
- scénarii d'utilisation des configurations ;
- algorithmes à implanter : plusieurs existent déjà mais pas tous, en particulier l'étiquetage des régions.

1. En outre je n'ai pas compris comment cela fonctionne...

Première démonstration à faire avec 1 module interconnecté par exemple à 1 DSP.

Il existe un simulateur à partir d'une description VHDL de l'algorithme et des mémoires qui utilise les outils Synopsys.

Si on veut faire une présentation des différentes parties du projet pour la prochaine édition de Sympa il faudrait que l'on ait avancé d'ici là :

- réalisation d'un prototype ;
- algorithmique ;
- implantation des algorithmes ;
- méthodologie d'optimisation du placement sur l'architecture :
 - bien poser les problèmes ;
 - regarder plusieurs approches et stratégies ;
 - faire un modèle approximatif de coût.

3.4 Partage des tâches du côté ARP

Les actions qui ont été proposées :

IRISA (Dominique LAVENIER) : prendre MPEG2 et voir l'adéquation d'Ardoise par rapport à cette application. À plus long terme prendre les applications et essayer de dériver l'architecture et le programme.

ENSTBr (Ronan KERYELL & Gérard OUVRADOU) : stratégie de séquençement des phases de reconfiguration et de transfert de données et réalisation. Regarder l'approche de parallélisation à grain fin aux applications.

IRIT (Daniel DOURS) : étudier une méthodologie de partitionnement d'une application globale.

Patrice BERTIN du CMA-ENSMF était là comme visiteur invité par Ronan KERYELL. Il travaille sur le langage Jazz, langage de haut niveau de synthèse architecturale se substituant à VHDL et successeur du langage 2Z de DEC-PRL.

Le compilateur Jazz génère des netlists et comprend des annotations permettant de faire du placement dans les CLBs des circuits reconfigurables.

Il est intéressé par l'architecture d'Ardoise et la méthodologie. Le contexte du traitement d'images s'y prête bien. Le couplage fort avec un processeur est toujours intéressant afin de bénéficier de tout un environnement avec plein de mémoire accessible. Il est néanmoins inquiet sur les performances : seulement 30 MHz, débit mémoire faible.

Réponse de Didier DEMIGNY : il s'agit dans un premier temps de limiter les risques et valider le concept.

4 Conclusion

Les actions à court terme :

- préparer la réunion d'octobre 1999. Probablement 2 jours consécutifs : 1 journée ARP puis 1 journée avec toute l'action (ce qui implique réunion à Paris...). Fixer la date dès début septembre ;
- analyser les applications que Mohamed AKIL doit diffuser ;
- penser aux premiers outils à développer ;
- mettre à jour le site www de l'action et le basculer sur le site d'ISIS ;
- définir une bibliographie globale au projet (centrée sur le concept de reconfiguration dynamique par exemple) ;

- ce serait bien que chaque équipe fasse au moins une page [WWW](#) sur sa partie.

Les actions à moyen terme :

- réfléchir aux exposés que l'on pourrait faire à Sympa.