

TP6: Exploration architecturale complète

1. 0. Objectif
2. 1. Première implémentation
3. 2. Prise en compte du coût
4. 3. Exploration architecturale
5. 4. Compte-Rendu

TP Précédent: MjpegCourse/Coproc

0. Objectif

Au cours des TP précédents, vous avez eu l'occasion de prendre en mains les méthodes suivantes de paramétrage:

- Placement des ressources logicielles
- Modification des tailles de caches
- Dédoublage du flot de traitement
- Architecture multi-cluster
- Introduction de coprocesseurs matériels

Dans ce TP, on vous laisse libre de faire une exploration architecturale avec tous ces paramètres. La contrainte sera une contrainte de coût:

- En surface de circuit
- En fréquence de travail
- En consommation

Pour faire un test plus réaliste que ceux mis en oeuvre aux TP précédents, on utilisera une vidéo de 160x120 pixels. Elle est fournie dans `~pouillon/mjpeg_160x120.raw`.

Attention, la taille nécessaire en pile pour la fonction `Libu` dépend de la largeur de l'image, redimensionnez votre pile en fonction de ce paramètre !

La seule contrainte finale sur le SoC est la fréquence de la vidéo décompressée. On cherche à lire cette vidéo à 25 fps.

Une fréquence raisonnable pour votre SoC sera entre 100 Mhz et 300 Mhz.

1. Première implémentation

? Que proposez-vous à priori comme architecture ? Combien de clusters (si architecture clusterisée), combien de processeurs, quelles tailles de caches, quelle occupation en mémoire ? ...

2. Prise en compte du coût

On donne les estimateurs suivants pour l'évaluation de la surface (procédé de fabrication 90 nm):

- Cache: $0.008\text{mm}^2 + 0.05\text{mm}^2 * \text{size}$ (en Ko)
- MultiRam: $0.05\text{mm}^2 * \text{size}$ (en Ko)
- Mips: 0.16mm^2

- Contrôleur Mwmr: 0.013mm^2 par canal MWMR
- LocalCrossbar: $0.005\text{mm}^2 * \text{nb_initiateur} * \text{nb_cible}$
- Vgmn: $0.01\text{mm}^2 * \text{nb_initiateur} * \text{nb_cible}$ (si archi clusterisée, $\text{nb_initiateur} = \text{nb_cible} = \text{nb_clusters}$)
- Coprocesseur IDCT matériel: $0.01\text{mm}^2 + 70\text{mm}^2 / \text{latence}$ (en cycles)
- Tg: 0.3mm^2
- Ramdac: $0.01\text{mm}^2 + 0.05\text{mm}^2 * \text{size ram video}$ (en Ko, 1 pixel = 1 octet)

Ces valeurs sont données à titre indicatif, pour une technologie fictive. Elle devraient être redéfinies pour le procédé de fabrication visé.

? Évaluez le coût de votre SoC en surface.

? On veut finalement évaluer la consommation du SoC, on utilise la fonction de coût donnant `conso` *proportionnel* à `fréquence * surface`. Donnez la consommation de votre circuit.

3. Exploration architecturale

? Essayez de modifier votre design pour minimiser la consommation tout en remplissant la contrainte temporelle de 25 fps.

? Créez une feuille de calcul dans OpenOffice reprenant toutes ces contraintes et les fonctions de coût. Insérez dans chaque ligne les caractéristiques des SoC que vous simulez.

? Choisissez le meilleur SoC.

4. Compte-Rendu

Vous rendrez deux fichiers (pas une archive tar):

- Un rapport nommé `binome0_binome1.pdf`
- La feuille de calcul nommée `binome0_binome1.ods`
- La description DSX de votre "meilleur" SoC nommée `binome0_binome1.py`

Ces fichiers devront être livrés avant le mardi 20 mars 2007, 18h00 à [MailAsim:nipo Nicolas Pouillon]