

Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud

Travail de diplôme 2005

Département Electricité et Informatique



Travail de diplôme 2005

Lionel YERSIN

Responsable IETR: Olivier DEFORGES Professeur : Cédric BORNAND

Haute Ecole d'Ingénierie et de Gestion du Canton de Vaud

Objectifs

Vidéo

LARMPEG4

- Étendre le processus de prototypage à des architectures hétérogènes comprenant des DSP et
- mise en œuvre des communications entre DSP et FPGA à travers des IPs spécifiques
- gestion automatique de l'instanciation et du contrôle des interfaces de communication
- Intégration du codeur LAR, en validant les travaux déjà réalisés, puis en y ajoutant certaines fonctionnalités (couche entropique, transformée Hadamard, ...)

Encadrement

- Mickaël RAULET
- Olivier DEFORGES

Outils de développement

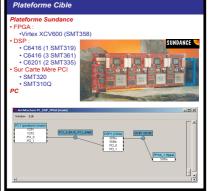
- Développement VHDL et simulation :
 - Mentor HDL Designer
 - Mentor Modelsim
- Synthèse et placement routage :
- Xilinx ISE / XST
- Génération automatique de code :
 - **SynDEx**

Méthodologie de prototypage sur architectures parallèles et mixtes Méthodologie AAA - SynDEx (http://www.SynDEx.org) Bibliothèques de transformation AAA : Algorithme Adéquation Architect Virtex XCV600 (SMT358 Génération de code Algorithme Adéquation Architecture C6416 (3 SMT361) C6201 (2 SMT335) ur Carte Mère PCI SMT320 Générique Dépendent de Dépendent de SynDEx.m4x l'architecture l'application Plate-formes - Sundance - LIMTS

- DSP Processeur

Composants

Dépendent du Dépendent du type de circuit type de média CP.m4x (C62x, C64x, FPGA)
SDB.m4x (C62x, C64x, FPGA)
Bifo.m4x (C62x, FPGA)
Bifo.DMA.m4x (C62x, FPGA)
US_PCI_RAM.m4x (FC, C62x, C64x)
US_PCI_SAM.m4x (FC, C62x, C64x)
TCP.m4x (C62x, PC) C62x m4x C64x m4x Pentium.m4x



Principe du codeur LAR

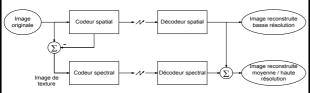
Codeur constitué de deux couches :

Exécutifs

Génériques

Exécutifs

- codeur spatial pour les forts taux de compression
- codeur spectral chargé de coder la texture



- Résolution adaptée à l'activité locale de l'image selon un gradient morphologique
- → codage de l'image par blocs de tailles variables
- · Quantification et prédiction DPCM des valeurs des blocs en fonction de la taille des blocs
- → codage de l'image par blocs de tailles variables

Résultats

- Projet basé sur la méthode de prototypage pour architectures hétérogènes
- Temps nécessaire pour coder une image de taille 64x64 pixels, sans les temps de transferts :
- → FPGA cadencé à 33 MHz : 163 µs
- → DSP cadencé à 400 MHz : 593 µs
- Résultat visuel de l'image quantifiée brute en sortie du codeur spatial:





Utilisation des ressources du FPGA pour des images 64x64 pixels:

→ Blocs RAM : 21 x 4'096 bits → Flip Flops 4'604 33% : 4'448 64% → Cellules logiques → Look Up Tables : 4'834 34%









