

TP n° 5 : Utilisation du processeur NIOS II avec instructions «spécialisées»

0. Introduction

L'objectif de ce TP est d'utiliser le processeur NIOS II (cœur logiciel) sur une carte FPGA d'Altera pour exécuter des programmes C et mesurer leurs temps d'exécution.

On examinera comment

- compiler une version du processeur à l'aide du logiciel Quartus II
- charger le processeur et l'environnement associé dans le FPGA (charger le fichier de configuration du processeur sur la carte à l'aide de la connexion USB – Blaster)
- charger un programme C dans la mémoire du processeur (charger le fichier correspondant au programme dans le FPGA) et l'exécuter
- écrire des programmes C sous l'environnement IDE, les charger dans le FPGA et les exécuter
- ajouter des instructions spécialisées au processeur NIOS
- mesurer les temps d'exécution des programmes.

1. Développement d'un programme sous l'environnement IDE

Ouvrir le logiciel NIOS IDE

Ouvrir une nouvelle application C : *new* | *CC++ Application* et choisir *Hello_world*. Choisir pour SOC Builder System C : *\niosII_cyclone_1c20\quartus_project\system.ptf*

Activer le projet en cliquant dessus.

Dans le menu *Projet* | *Propriétés* | *C/C++ Build*, sélectionner la configuration *Release* et un niveau d'optimisation (-O2 ou -O3)

Puis faire *Build Projet*.

Puis exécuter le projet avec *Run* | *Run as* | *Nios Hardware*

2. Mesure des temps d'exécution du programme *aes_8bits.c*

Les programmes *aes_8bits_V1.c* et *aes_8bits_V2.c* sont disponibles à l'adresse

La version V1 permet de mesurer les temps d'exécution des fonctions *aes_encrypt_ecb* et *aes_decrypt_ecb*. **Mesurer les temps d'exécution de ces deux fonctions en nombre de cycles d'horloge du processeur.**

On notera que le programme dispose d'un affichage qui permet de vérifier que les fonctions de cryptage et de décryptage sont correctes.

La version V2 permet de mesurer les temps d'exécution de chaque fonction individuelle, ce qui permet d'estimer les fonctions qu'il est intéressant d'accélérer. **Mesurer les temps d'exécution des différentes fonctions en nombre de cycles d'horloge du processeur.**

3. Utilisation d'instructions spécialisées dans le programme *aes_8bits.c*

Chargement de la configuration sur le FPGA

Dans le TP 4, on a écrit et validé des opérateurs décrits en VHDL pour spécifier des instructions spécialisées.

a) Ouvrir sous Quartus le projet *standard.qpf* contenu dans l'archive *M1_TP4.rar* (site du cours). Définir les instructions spécialisées et ajouter les au processeur NIOS avec SOPC builder sous la rubrique « custom instructions »

- b) Quand toutes les instructions ont été ajoutées, générer le fichier comprenant tous les composants nécessaires avant la compilation VHDL du « système complet : processeur, mémoire et composants d'interface » via la fonction « generate ». Après cette étape, effectuer la compilation du VHDL généré.
- c) A la fin de la compilation, charger le fichier standard.sof sur la carte FPGA à l'aide du module « programmer ».

Mesure des temps d'exécution du programme AES_8bits.c avec accélération matérielle.

- a) Ajouter les #define correspondant aux instructions spécialisées.
- b) Ecrire les versions accélérées avec des instructions spécialisées des différentes fonctions C des fonctions cryptage et décryptage.
- c) Exécuter la version accélérée d'AES_8bits.c en vérifiant que les résultats du cryptage et décryptage sont corrects.
- d) Mesurer les temps d'exécution. Quelle est l'accélération obtenue ? Vous préciserez quelles sont les instructions spécialisées utilisées et les fonctions accélérées.

4. Devoir 2

Le devoir 2 est constitué des TP4 et TP5. Le compte rendu est à rendre par binôme à de@lri.fr au plus tard le 7 Novembre.

5. Annexes (à rajouter)

Mesure des temps d'exécution

Chargement d'une configuration sur le FPGA (processeurs + opérateurs spécialisés)

Modification du processeur et ajout de nouvelles instructions.