

VTF 2010 - Design de systèmes de traitement vidéo temps réel par FPGA

Labo Nios 26/11/2010 – V1.0
Temps prévu : 3 périodes encadrées

Prof. M. Starkier
Assistant : S.Masle

Objectifs du laboratoire

Les objectifs de ce laboratoire sont de :

1. Se familiariser avec l'environnement ALTERA d'implémentation d'un processeur embarqué NIOS II et les outils de développement logiciel.
2. Ajouter et interfacier un processeur NIOS II, en tant que contrôleur d'un module de traitement du signal vidéo sur FPGA.
3. Contrôler à partir du PC, via USB et par une interface graphique, les paramètres de traitement de la vidéo.

Note : Le labo s'effectuera à partir du design réalisé au cours du labo 1 (initiation) - réglage des niveaux RVB. Vous pouvez utiliser, au choix, votre propre design ou la solution fournie.

Fichiers et documents fournis

Les fichiers et les documents se trouvent sur le site RedS :
http://www.reds.ch/Formations/Master/VTF/VTF_doc.aspx

Vous trouverez les informations nécessaires à la réalisation des différentes étapes de ce laboratoire dans les documents suivants :

- NIOS II Reference Handbook : Manuel de référence hardware du Nios
- NIOS II Software Reference Handbook
- My First Nios II Software : Tutorial

Chargez et décompressez les dossiers **Labo_initiation_solution.zip**, et **Labo_NIOS2.zip**

- **Labo_initiation_solution.zip** contient la solution complète du Labo Initiation
- **Labo_NIOS2.zip** contient :
 - L'Interface graphique – contrôleur avec 3 sliders - sur PC et communication USB.

N'oubliez pas de sauvegarder votre répertoire de travail sur les machines du labo. Le répertoire étudiant est effacé à chaque extinction de machine.

Travail à effectuer

1. Validation du design « réglage des niveaux RVB » du labo 1 initiation

Ce design va être utilisé comme base pour la suite du labo. Si vous n'utilisez pas votre design, mais la solution fournie, vous devez (impérativement) compiler le code, générez le bitstream et le tester sur la carte. Ceci doit être fait en premier, avant toute modification du code.

2. Ajout du Nios et premier programme

Ajoutez le Nios au design « réglage des niveaux RVB » en tenant compte des consignes ci-dessous :

- Utilisez un Nios II/s, avec un cache d'instruction de 32Kbytes, et le Jtag Debug Module niveau 1.
- Composants nécessaires :
 - On-Chip Memory : RAM
 - Block type : Auto
 - Cocher : Initialize memory content
 - Data width : 32
 - Memory size : au moins 65536 bytes
 - Jtag UART
 - PII à 85Mhz
 - 2 Interval timer : un à 10ms, un à 10us
 - Hardware options : full featured
 - PIO (Parallel I/O) port d'entrée /sortie

Testez le Nios avec un programme simple : Allumage / extinction d'une led. La led doit être connectée au PIO.

Note : Suivez le tutorial : **My First Nios II Software**, qui contient un exemple d'allumage de led.

3. Contrôle des niveaux RVB par sliders sur PC

Effectuer l'interfaçage des trois registres « niveaux RVB » avec des PIOs. L'interface graphique (3 sliders) est un exécutable qui envoie des commandes texte via l'USB. Il faut se connecter au Nios avant de bouger les sliders (menu Connect -> JTAG UART).

L'interface graphique envoie des commandes texte qui contiennent successivement :

- 1- l'identifiant de la couleur à modifier (red,blue,green)
- 2- la valeur en decimal (0 à 511)
- 3- la lettre 'f'.

Exemple :

- déplacement du slider rouge, l'interface envoie : red348f
- déplacement du slider vert , l'interface envoie : green26f
- déplacement du slider bleu, l'interface envoie : blue423f

Ces commandes sont reçues (caractères ASCII) - après transmission par USB - par le composant Jtag UART. Utilisez le code C ci-dessous :

```
fp = fopen ("/dev/jtag_uart", "r+"); //Open file for reading and writing
.....
prompt = getc(fp); // Get a character from the JTAG UART
.....
fclose(fp);
```