

Laboratoire 03 – Utilisation des interruptions

Objectif

Ce laboratoire a pour but d'utiliser les interruptions du HPS. Il s'agira d'ajouter la possibilité de générer des interruptions à un PIO que vous utilisez déjà. Vous devrez comprendre le mécanisme des interruptions sur la Cyclone V SoC afin de bien gérer la gestion de celles-ci.

Spécifications

Vous complétez votre travail du laboratoire précédent avec les spécifications suivantes :

L'appui sur KEY2 ou KEY3 génère une interruption permettant de réaliser les fonctions suivantes :

- Appui sur KEY2 : l'affichage des LEDs et des afficheurs 7 segments subit une rotation à droite.
- Appui sur KEY3 : l'affichage des LEDs et des afficheurs 7 segments subit une rotation à gauche.

Documents à rendre

Vous devrez rendre un rapport à l'issue de ce laboratoire contenant les explications sur les différentes étapes de la réalisation de votre système.

Vous devez également rendre une archive avec les sources du projet pour Quartus, Qsys et le programme C. Utiliser le Makefile fourni pour générer votre archive à rendre en tapant "make zip" dans un terminal.

Les fichiers sont à rendre sur Moodle

Travail demandé

- 1) Récupérer votre projet du laboratoire FPGA_GPIO, il vous servira de base pour ce travail. Y incorporer les fichiers source supplémentaires obtenus dans le repo en faisant un « git pull »
- 2) Ouvrir votre projet Qsys, éditer le composant HPS pour activer les interruptions nécessaires.
- 3) Configurer le PIO connecté aux boutons (KEY) pour qu'il puisse générer une interruption lors du flanc montant (« rising edge »).
- 4) Câbler le PIO dans Qsys.
- 5) Générer les fichiers HDL du projet Qsys.
- 6) Synthétiser et faire le placement routage du projet.
- 7) Modifier la configuration mémoire de votre projet "Altera Monitor Program" pour qu'il alloue une portion pour les vecteurs d'interruptions. Dans la configuration du projet, aller dans le dernier onglet "Memory Settings" et choisir "Exceptions" dans la section "Linker Section Presets".
- 8) Compléter votre code pour intégrer les nouvelles spécifications.
 - Compléter la routine d'interruption `__cs3_isr_irq(void)` du fichier `exceptions.c` fourni. La routine d'interruption doit être la plus rapide possible.
 - Configurer le PIO et le GIC pour gérer les interruptions en provenance du PIO.
 - Initialiser le mode interruption du processeur en utilisant les fonctions `set_A9_IRQ_stack(void)` et `enable_A9_interrupts(void)` du fichier `exceptions.c` fourni.
 - Réaliser le comportement de rotation des affichages.
- 9) Compiler le code et le tester sur la carte DE1-SoC.
- 10) Faire valider votre montage par le professeur ou l'assistant.