

Acquisition de plusieurs positions via le CPU Différence de comportement entre le logiciel et le matériel

Objectif :

Cette manipulation a pour objectif de mettre en évidence les différences de comportement entre le logiciel (**séquentiel**) et le matériel (**concurrent**). La première partie montrera qu'une lecture séquentielle de 3 positions ne permet pas d'assurer des valeurs cohérentes entre celles-ci. Ensuite, l'étudiant devra adapter l'interface afin de garantir une acquisition correcte de 3 positions. Le but est d'acquérir les positions ayant toute la même référence de temps (en anglais : timestamp). Il s'agit de prendre une photo instantanée des 3 positions, même si les temps d'acquisition et de transfert sont différent pour chacune d'entre elle.

A rendre :

Ce laboratoire sera évalué. Il sera réalisé par groupe de 2 étudiants.

Vous devrez rendre à l'issu de ce travail de laboratoire 2 fichiers, soit:

- **Un fichier PDF** avec votre rapport de laboratoire comprenant toutes les explications sur les étapes demandées dans l'énoncé (1^{ère} et 2^{ème} parties) ainsi que la **copie** de tous les documents nécessaires pour la correction (scan documents manuscrit, schéma Logisim, programme C, etc.)
- **Une archive** zip ou tar.gz avec les fichiers suivants pour les **parties 1 et 2**, soit :
 - o les fichiers *.cir, *.c, et autres fichiers modifiés
 - sans** le répertoire "include" (il fait 3.3MB!). Merci

Vous devez déposer vos fichiers sur la page IFS du site Moodle (Cyberlearn).

Description :

Les nouveautés de ce laboratoire :

Carte Max-V connectée connecteur 80 pôles de la carte REPTAR :

Utilisation d'une carte Max-V comprenant un système numérique qui génère la position d'une machine avec 3 axes: X, Y et Z. Le bouton rotatif avec codeur incrémental de 4096 pas est utilisé pour simuler un déplacement de la machine.

La position de la machine correspond aux équations suivantes:

- $Y = X + 500, Z = -X$
- la position de l'axe X varie de + 7000 à -7000 (butée mécanique virtuelle)

Dans Logisim :

Circuit FPGA_top

- Connecteur 80 pôles permettant la connexion avec la carte Max-V qui modélise le fonctionnement d'une machine 3 axes (X, Y, Z).
- Les signaux suivants sont connectés au circuit "lba_user_interface", soit:
 - o vecteur 16 bits avec la position et le nom de chaque axe (voir ci-après)
pos_axe_x[15..0], pos_axe_y[15..0], pos_axe_z[15..0]
 - o signal pour initialiser la position de la machine: init_axe
la position d'initialisation est : X = 0, Y = 500, Z = 0

Circuit lba_user_interface

- Toutes les connexions nécessaires pour réaliser le laboratoire, soit les leds, switch, affichage 7 segments, les 3 positions de la machine.

Plan d'adressage proposé :

Adresse	Read	Write
	D15 0	D15 0
0x1900 0000	Cst (valeur à définir)	not used
0x1900 0002	[15..8] "0..0"; [7..0] Leds7...0	[15..8] reserved; [7..0] Leds7...0
0x1900 0004	[15..8] Aff_7seg2"; [7..0] Aff_7seg1	[15..8] Aff_7seg2"; [7..0] Aff_7seg1
0x1900 0006	[15..8] "0..0"; [7..0] switch7..0	not used
0x1900 0008	[15..2] "0..0"; [1..0] Statut	[15..2]: reserved; [0] Init
0x1900 000A	<i>reserved</i>	<i>reserved</i>
0x1900 000C	<i>reserved</i>	<i>reserved</i>
0x1900 000E	<i>reserved</i>	<i>reserved</i>
0x1900 00100	Pos-X : [15..14] Axe, [13..0] Position	<i>not used</i>
0x1900 00102	Pos -Y : [15..14] Axe, [13..0] Position	<i>not used</i>
0x1900 00104	Pos -Z : [15..14] Axe, [13..0] Position	<i>not used</i>
0x1900 00106	<i>reserved</i>	<i>reserved</i>
0x1900 00108	<i>reserved</i>	<i>reserved</i>
0x1900 0010A	<i>reserved</i>	<i>reserved</i>
0x1900 0010C	<i>reserved</i>	<i>reserved</i>
0x1900 0010E	<i>reserved</i>	<i>reserved</i>

Explications sur le plan d'adressage :

- Switch7..0 lecture des 8 switch de la FPGA SP6
- Leds7..0 écriture/lectures des 8 leds de la FPGA SP6
- Aff_7seg écriture/lectures affichage 7seg. 1 et 2 de la FPGA SP6
- Statut Vous devez gérer deux bits de Statut. Il seront placés aux bits [1..0]. Voici le fonctionnement de ces 2 bits :
 - o Statut[1] indique si l'interface dispose d'un système de capture des 3 positions (photo instantanée).
 - o Statut[0] indique qu'une photo a été prise lorsque le système dispose d'un système de capture. Ce bit est valide uniquement si Statut[1] = '1'. Le fonctionnement sera spécifié pour la seconde partie du laboratoire.
- Init commande d'initialisation de la position initiale, soit : X = 0, Y = 500 et Z = 0. Il sera placé au bit [0] de l'adresse.
- Pos-n lecture des 3 positions (axe: X, Y ou Z) avec code de l'axe
 - o Bits [13..0] : position sur 14 bits en représentation signée en C2.
 - o Bits [15..14] : code de l'axe sur 2 bits, soit:
axe X : "00", axe Y : "01", axe Z : "10" (code "11" reserved)

Description des circuits fournis :

Circuit FPGA_top

- Connexion directe dans la FPGA entre le SW5 et le point de l'afficheur 7 segments est réalisée afin de valider que le design est programmé sur la FPGA.

Circuit lba_user_interface

- La gestion des switchs et les leds de la FPGA est déjà fournis.
- Vous devrez faire les modifications nécessaires afin de passer à un décodage d'une zone de 16 adresses. Le projet fourni gère une zone de 8 adresses.

Syllabus :

Vous devez modifier le bloc lba_usr_interface pour ajouter les accès spécifiés, dans le plan d'adressage, pour les 3 positions, la gestion des 2 bits de statut et du signal Init.

Vous disposez d'un fichier C qui gère la sortie du programme (switch 4 du CPU). Vous devez compléter le programme afin de réaliser les fonctionnalités suivantes :

- Tant que le SW1 est actif, vous lisez successivement l'état des compteurs de position, puis les comparer et les afficher sur la console comme suit :
 - o Si les 3 positions sont correctes (voir formule), vous affichez :
 - OK : statut, position-X, position-Y, position-Z**Rem** : afficher ce texte toujours sur la même ligne (utiliser \r)
 - o Si les positions sont incorrectes, vous affichez :
 - ER : statut, position-X, position-Y, position-Z**Rem** : afficher ce texte sur une ligne différente (utiliser \n)
- Une pression sur le switch SW5 permet d'allumer successivement l'un des segments de l'afficheur 7SEG1 dans le sens horaire.

- Le switch SW8 permet l'initialisation de la machine, soit $X = 0$, $Y = 500$ et $Z = 0$.
- Le switch SW3 allume la Led0 et le switch SW4 allume la Led1.

Travail demandé :

Première partie :

- 1) Compléter le bloc lba_usr_interface fourni afin de répondre au plan d'adressage présenté précédemment. Celui-ci doit permettre tous les accès indiqués dans les spécifications données précédemment.
Pour la 1^{ère} partie les bits de Statut[1..0] seront mis à "00" (fixe).
- 2) Simuler vos blocs dans Logisim afin d'en vérifier le fonctionnement (rendre les chronogrammes).
- 3) Réaliser l'intégration de votre interface. Si cela vous semble pertinent, réaliser quelques tests avec les commandes de u-boot (mw et md).
- 4) Compléter le programme C fourni pour répondre aux fonctionnalités demandées précédemment.
Vous devez expliquer pourquoi vous obtenez parfois des écarts entre les lectures des 3 positions.
- 5) Vous devez faire valider par le professeur ou l'assistant la mise en évidence de ces erreurs de lectures des 3 positions.

Seconde partie :

La seconde partie de la manipulation consiste à modifier l'interface afin d'ajouter la possibilité de réaliser une acquisition correcte des 3 positions. C'est le switch SW2 qui permet d'activer cette fonctionnalité lorsqu'il est actif. Vous devez prévoir une solution permettant de réaliser une « photo » des 3 positions prises sur commande du CPU. Vous devez expliquer votre solution et comment vous répartissez les rôles entre le logiciel et le matériel. Vous devez gérer les 2 bits de Statut[1..0]. Vous ferez les adaptations nécessaires du plan d'adressage, puis vous modifierez le projet Logisim en conséquence de même que le programme C. Votre solution doit permettre d'avoir les 2 types d'acquisition selon l'état de SW2

Les fonctionnalités créées en 1^{ère} partie doivent être conservées.

Travail demandé pour 2^{ème} partie :

- 6) Concevoir une solution qui permette de résoudre le problème rencontré lors de la première partie. Vous devez expliquer votre solution et donner les modifications du plan d'adressage.
- 7) Réaliser les modifications nécessaires dans le projet Logisim et du programme C. Puis vous devez tester votre solution et démontrer, quel que soit les variations du bouton incrémental, vous n'avez jamais d'erreur entre les 3 positions mesurées.
- 8) Vous devez faire valider par le professeur ou l'assistant votre solution finale.