

# INF1500 – LOGIQUE DES SYSTÈMES NUMÉRIQUES

## LABORATOIRE 4

### INTRODUCTION AU LANGAGE VHDL

---

## Simulation et implémentation d'un circuit en logique combinatoire en VHDL

---

*Révisions :*

Alexy TORRES AURORA DUGO - V1.0

Automne 2019

Département de génie informatique et de génie logiciel

École Polytechnique de Montréal



# 1 Objectifs

L'objectif de ce laboratoire est de concevoir, simuler et implémenter un circuit en logique combinatoire sur la carte FPGA du laboratoire en utilisant le langage de description de matériel **VHDL**. L'objectif de ce laboratoire est de se familiariser au langage de description matériel **VHDL**. La validation des différentes composantes sera effectuée à travers la simulation et l'implémentation du circuit complet sur la carte FPGA.

## 2 Système à réaliser

Dans ce laboratoire, on vous demande de réaliser un circuit qui intègre :

- MOD\_5, un premier module qui fait le calcul modulo 5. Si on passe le nombre 01011 à notre entrée A. Le module retournera 00001. **Attention, utiliser l'opérateur MOD de VHDL sera compté comme faux, vous devez écrire votre propre circuit.** L'entrée de ce module est sur 5 bits, la sortie de ce module est sur 5 bits.
- MULT\_2, un second module permettant de faire une multiplication par 2 de l'entrée A. L'entrée de ce module est sur 5 bits, la sortie de ce module est sur 5 bits.
- MUX 2:1, un multiplexeur 2 vers 1 permettant de choisir entre les deux modules. Le multiplexeur prends en entrée deux bus de 5 bits, a une entrée de sélection sur 1 bit et a une sortie sur 5 bits.
- BIN\_2\_DEC, un module permettant de transformer la sortie du multiplexeur sur 5 bits en deux sortie sur 4 bits. Ce module est expliqué dans la suite de l'énoncé.
- DISP\_7SEG\_LAB4, un driver permettant d'afficher les deux sorties de BIN\_2\_DEC sur les afficheurs 7 segments de la carte. **Ce module vous est fournis et n'est pas à réaliser.**

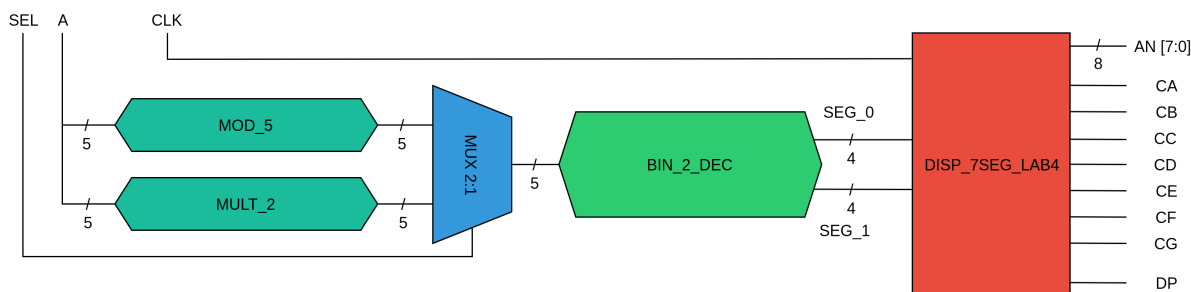


FIGURE 1 – Schéma du circuit à réaliser

Grâce au multiplexeur il est possible de choisir le mode de fonctionnement du système. Les différents modes sont représentés dans le tableau ci-dessous.

SEL	Opération
0	Module 5
1	Multiplication par 2

### 3 Module BIN\_2\_DEC

Le module BIN\_2\_DEC prend en entrée un nombre binaire de 5 bits. Il doit ensuite convertir cette entrée en deux sortie SEG\_0 et SEG\_1.

- SEG\_0 représente les unités du nombre binaire en entrée sur module. Exemple : si l'entrée est 01101 (13) alors la SEG\_0 vaut 0011 (3).
- SEG\_1 représente les dizaines du nombre binaire en entrée sur module. Exemple : si l'entrée est 01101 (13) alors la SEG\_0 vaut 0001 (1).

Une fois la conversion faite, le drivers 7 segments fourni pour ce laboratoire se chargera du reste.

### 4 Module DISP\_7\_SEG\_LAB4

Ce module prend deux entrées : SEG\_0 et SEG\_1. Ces entrées sont expliquées dans la section précédente. Le driver affiche les chiffres envoyés SEG\_0 et SEG\_1 quand ceux-ci sont inférieurs à 10, sinon il affiche le caractère “-”.

Les entrées sont ensuite utilisées par le driver afin de faire afficher les nombres sur les afficheurs 7 segments présents sur la carte. Vous n’avez pas à vous soucier de ce module, il vous est fourni.

### 5 Fichier de contrainte

Pour le fichier de contrainte voici les ports que vous devez utiliser :

```
set_property -dict { PACKAGE_PIN E3      IOSTANDARD LVCMOS33 } [get_ports { CLK }];

set_property -dict { PACKAGE_PIN T10     IOSTANDARD LVCMOS33 } [get_ports { CA }];
set_property -dict { PACKAGE_PIN R10     IOSTANDARD LVCMOS33 } [get_ports { CB }];
set_property -dict { PACKAGE_PIN K16     IOSTANDARD LVCMOS33 } [get_ports { CC }];
set_property -dict { PACKAGE_PIN K13     IOSTANDARD LVCMOS33 } [get_ports { CD }];
set_property -dict { PACKAGE_PIN P15     IOSTANDARD LVCMOS33 } [get_ports { CE }];
set_property -dict { PACKAGE_PIN T11     IOSTANDARD LVCMOS33 } [get_ports { CF }];
set_property -dict { PACKAGE_PIN L18     IOSTANDARD LVCMOS33 } [get_ports { CG }];

set_property -dict { PACKAGE_PIN H15     IOSTANDARD LVCMOS33 } [get_ports { DP }];

set_property -dict { PACKAGE_PIN J17     IOSTANDARD LVCMOS33 } [get_ports { AN[0] }];
```

```

set_property -dict { PACKAGE_PIN J18      IOSTANDARD LVCMOS33 } [get_ports { AN[1] }];
set_property -dict { PACKAGE_PIN T9       IOSTANDARD LVCMOS33 } [get_ports { AN[2] }];
set_property -dict { PACKAGE_PIN J14      IOSTANDARD LVCMOS33 } [get_ports { AN[3] }];
set_property -dict { PACKAGE_PIN P14      IOSTANDARD LVCMOS33 } [get_ports { AN[4] }];
set_property -dict { PACKAGE_PIN T14      IOSTANDARD LVCMOS33 } [get_ports { AN[5] }];
set_property -dict { PACKAGE_PIN K2       IOSTANDARD LVCMOS33 } [get_ports { AN[6] }];
set_property -dict { PACKAGE_PIN U13      IOSTANDARD LVCMOS33 } [get_ports { AN[7] }];

```

Pour les entrées A et SEL, à vous de les placer sur les commutateurs comme à votre habitude.

## 6 Travail à effectuer

Le travail à réaliser est le suivant :

- Décrire le module de calcul du modulo 5 **en VHDL**.
- Décrire le module de multiplication par 2 **en VHDL**.
- Décrire le multiplexeur **en VHDL**.
- Décrire le module binaire vers décimale **en VHDL**.
- Décrire le circuit au complet **en VHDL**.

Pour chaque étape, il est **ÉVIDENT** qu'il faille valider le fonctionnement de son circuit par simulation. **Simulez chaque module indépendamment !** N'oubliez pas de décrire cette étape dans votre rapport !

La dernière étape consiste à implémenter et valider le fonctionnement du circuit sur la carte FPGA. Le fichier de contrainte ainsi que l'implémentation de ce circuit est noté.

## 7 Livrables attendus

Les livrables suivants sont attendus :

- Un rapport pour le laboratoire.
- Le dossier du projet (un dossier contenant **l'intégralité de vos fichiers**).

Le tout à remettre dans une seule archive **zip** avec pour nom matricule1\_matricule2\_lab4.zip à téléverser sur Moodle.

Le rapport doit contenir :

- Une introduction
- Une description pour chacun des modules
- Le code VHDL de chaque module et du système complet
- Une description de la stratégie de test pour chacun des modules et des images de vos simulations
- Une conclusion

Si vous désirez mettre du code VHDL ou autre dans votre rapport, ne faites pas de capture d'écran de l'éditeur. Il est préférable de créer un fichier séparé et d'y faire référence dans

votre rapport.

**Consultez le site Moodle du cours pour la date et l'heure limites de remise des fichiers.**

**L'évaluation sera faite à la prochaine séance de laboratoire.**

## **8 Barème**

La pondération sera donnée de la façon suivante sur 7 :

- 0/7 : l'étudiant n'a rien ou presque rien fait ;
- 1/7 : l'étudiant a réussi à faire approximativement 25% du laboratoire ;
- 2/7 : l'étudiant a réussi à faire un peu moins que la moitié du laboratoire ;
- 3/7 : l'étudiant a réussi à faire un peu plus que la moitié du laboratoire ;
- 4/7 : l'étudiant a réussi à faire presque tout le laboratoire et a démontré une compréhension comportant des faiblesses ;
- 5/7 : l'étudiant a réussi à faire tout le laboratoire et a démontré une compréhension comportant des faiblesses ;
- 6/7 : l'étudiant a réussi à faire presque tout le laboratoire et a démontré une excellente compréhension ;
- 7/7 : l'étudiant a réussi à faire tout le laboratoire et a démontré une excellente compréhension ;

**25% des points sont retranchés par jour de retard!!!**