

AutoSIN

—

TP3 : Logique séquentielle en VHDL



télécom
saint-étienne

2025 – 2026

Objectifs de la séance

- Concevoir un système numérique séquentiel,
- Pratiquer la conception structurelle,
- Vérifier en simulation le bon fonctionnement d'un système séquentiel.

Elle n'est pas rappelée dans cet énoncé mais vous devez appliquer la méthodologie de conception numérique sur FPGA vue au TP précédent.

Matériel fourni

Pour les exercices suivants, vous réaliserez les étapes de conception et de vérification en utilisant les feuilles à disposition.

Pour la mise en œuvre finale, vous disposez de cartes de développement DE10-Lite et du câble USB associé permettant de les connecter au PC pour les configurer.

Validation

Lorsque vous avez terminé un exercice, appelez l'enseignant afin qu'il valide avec vous le bon fonctionnement du circuit.

Exercice 1 : Compteur à valeur maximale modifiable

La description en VHDL d'un compteur à valeur maximale modifiable est rappelée ci-dessous.

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY compteur_max IS

    PORT (
        clk      : IN  STD_LOGIC;
        rst_n    : IN  STD_LOGIC;
        max      : IN  STD_LOGIC_VECTOR (3 DOWNTO 0);
        valeur   : OUT STD_LOGIC_VECTOR (3 DOWNTO 0));

END compteur_max;

ARCHITECTURE archi OF compteur_max IS

    SIGNAL valeur_interne : STD_LOGIC_VECTOR (3 DOWNTO 0);

BEGIN

    PROCESS (clk, rst_n)
    BEGIN
        IF rst_n = '0' THEN
            valeur_interne <= (OTHERS => '0');
        ELSE
            IF rising_edge(clk) THEN
                IF valeur_interne = max THEN
                    valeur_interne <= (OTHERS => '0');
                ELSE
                    valeur_interne <= STD_LOGIC_VECTOR (UNSIGNED (valeur_interne) + 1);
                END IF;
            END IF;
        END IF;
    END PROCESS;

    valeur <= valeur_interne;

END archi;
```

Conception et vérification Suivre les étapes 2 à 9 du flot de conception pour concevoir ce compteur et vérifier son bon fonctionnement **en simulation seulement**.

Évaluation À partir des rapports de placement/routage, donner le nombre de LUT (*logic elements*) et de bascules D (*registers*) occupés par l'implantation du circuit.

Q1 Quelle est la valeur maximale atteinte par le compteur ? Est-ce cohérent ?

Q2 Que remarquez-vous quant à la fréquence de comptage du compteur ? Est-il utilisable en l'état pour décrire un chronomètre ?

Exercice 2 : Compteur avec autorisation de comptage

Modifier la description précédente pour y ajouter la possibilité d'**autoriser** le comptage.

Q1 Si l'on souhaite ajouter cette possibilité, doit-on modifier l'entité ? Si oui, comment ? Doit-on modifier l'architecture ?

Q2 Quel périphérique de la carte est adapté pour générer **manuellement** le signal d'autorisation ?

Q3 Quel périphérique de la carte est adapté pour afficher la valeur du compteur ?

Conception et vérification Suivre les étapes 2 à 9 du flot de conception pour concevoir ce compteur et vérifier son bon fonctionnement en simulation puis sur la carte de développement.

Évaluation À partir des rapports de placement/routage, donner le nombre de LUT (*logic elements*) et de bascules D (*registers*) occupés par l'implantation du circuit.

Q4 A-t-on résolu le problème relevé en fin d'exercice 1 ?

Exercice 3 : Génération d'un signal d'autorisation

Repartir de la description initiale du compteur à valeur maximale modifiable de l'exercice 1 et la modifier pour permettre au compteur d'indiquer, sur une sortie dédiée, que sa valeur maximale est atteinte.

Q1 Si l'on souhaite ajouter cette possibilité, doit-on modifier l'entité? Si oui, comment? Doit-on modifier l'architecture?

Q2 Le signal d'horloge disponible sur la carte Altera a une fréquence de 50 MHz. Si l'on veut générer un signal d'autorisation pour le compteur gérant les **secondes**, jusqu'à quelle valeur faut-il compter? Combien de bits seront alors nécessaires pour stocker la valeur interne du compteur?

Conception et vérification Suivre les étapes 2 à 9 du flot de conception pour concevoir ce compteur et vérifier son bon fonctionnement **en simulation seulement**.

Évaluation À partir des rapports de placement/routage, donner le nombre de LUT (*logic elements*) et de bascules D (*registers*) occupés par l'implantation du circuit.

Exercice 4 : Chiffre des secondes d'un chronomètre

On souhaite mettre en œuvre le chiffre des secondes d'un chronomètre. On utilisera pour cela un compteur, qui comptera de 0 à 9 et sera incrémenté chaque seconde. Sa sortie sera affichée sur un afficheur 7 segments.

Pour cette description, réutiliser les composants décrits précédemment!

Pour cela, incluez les fichiers .vhd écrits précédemment dans le projet, sans les copier, mais en allant les chercher dans l'arborescence.

Conception et vérification Suivre les étapes 2 à 9 du flot de conception pour concevoir le chiffre des secondes du chronomètre et vérifier son bon fonctionnement en simulation puis sur la carte de développement.

Évaluation À partir des rapports de placement/routage, donner le nombre de LUT (*logic elements*) et de bascules D (*registers*) occupés par l'implantation du circuit.

Exercice 5 : Chronomètre à 3 chiffres

On souhaite mettre en œuvre un chronomètre affichant les chiffres des décisecondes, des secondes et des décisecondes.

Conception et vérification Suivre les étapes 2 à 9 du flot de conception pour concevoir le chronomètre et vérifier son bon fonctionnement en simulation puis sur la carte de développement.

Évaluation À partir des rapports de placement/routage, donner le nombre de LUT (*logic elements*) et de bascules D (*registers*) occupés par l'implantation du circuit.