

TD 04 : Langage VHDL

Instruction séquentielle

TD VHDL / FPGA

Dr. Lezzar

Table des matières



Objectifs	3
I - Exercice 01	4
II - Exercice 02	5
III - Exercice 03	6
IV - Exercice 04	7
V - Exercice 05	8
VI - Exercice 06	9

Objectifs

- Maîtriser la structure process.
- Connaître la programmation synchrone et asynchrone.
- savoir l'architecture Comportementale.

Exercice 01

I

Écrire le code VHDL de la bascule JK synchrone.

Méthode

Sensible sur le front montant de l'horloge

Complément

Cette bascule présente des entrées de présélection SET et de remise à zéro RESET (active à zéro) prioritaire a l'entrée mis à un SET (active à zéro), tous les deux sont asynchrone de l'horloge.

Table de vérité d'une bascule J.K

	Pr	Cl	h	J	K	Q ⁺	
Mode Asynchrone	0	0	X	X	X	X	État interdit
	0	1	X	X	X	1	Remise à 1
	1	0	X	X	X	0	Remise à 0
Mode Synchrone	1	1	0/1	x	x	Q ⁻	Etat mémoire
	1	1	↓	0	0	Q ⁻	Etat mémoire
	1	1	↓	0	1	0	Remise à 0
	1	1	↓	1	0	1	Remise à 1
	1	1	↓	1	1	\overline{Q}	Basculement

Exercice 02

II

Écrire le code VHDL d'un compteur binaire 8 bits (qui compte de 0 à 255) et ayant une entrée clear; en utilisant une description comportementale. avec

1. La méthode synchrone
2. La méthode asynchrone

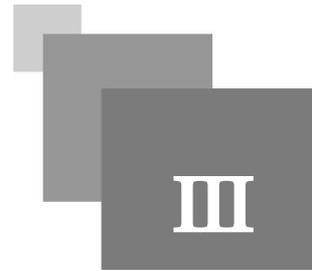
Conseil

Utiliser une variable qui fait le comptage.

Complément

Utiliser les bibliothèque arithmétique qui faite les opération (+,-,*,/) et numérique pour convertir un signal binaire en un entier.

Exercice 03



Soit un compteur/décompteur avec chargement synchrone binaire 4 bits ayant les entrées suivantes :

P(4bits), load(1bit), ck, clear, count_ena(1bit), direction (1bit)

Complément

Le compteur fonctionne comme suit :

- Si clear =0, on une remise a zero
- Si load=1 et clear=1, alors chargement par p (sortie (q)=p)
- Si le front montant alors :
 - Si count_ena=1 et direction =0 décomptage
 - Si count_ena=1 et direction =1 comptage

1. Écrire son code VHDL en utilisant une description comportementale

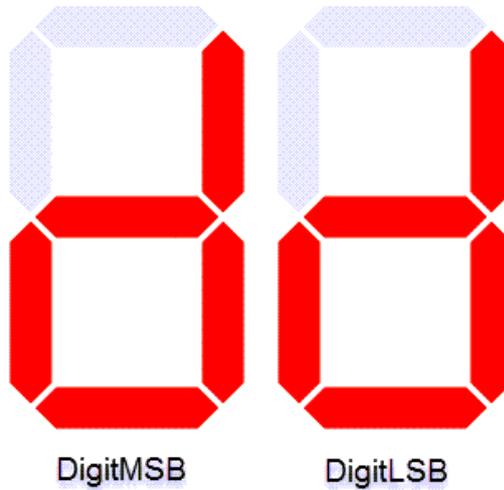
Exercice 04

IV

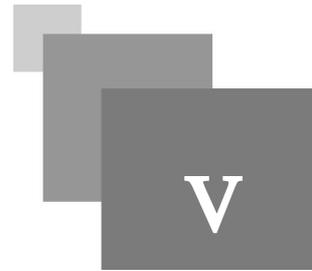
Écrire son code VHDL d'un compteur BCD à deux digits qui affiche les chiffre de 1 à 99

 *Complément*

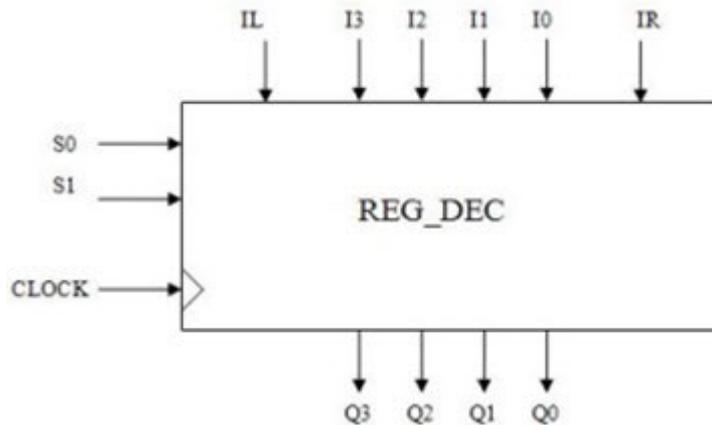
Deux digits c'est à dire 2 signaux à 4bits. DigitLSB (4 bits), DigitMSB (4 bits).



Exercice 05



Le circuit d'un registre à décalage synchrone (front montant) est donné sur la figure suivante :



Complément

4 entrées (I3, I2, I1, I0)/ 4 sorties (Q3, Q2, Q1, Q0)

Une entrée horloge CLOCK.

2 entrées de sélection (S0, S1) qui permettent de choisir le mode de sélection

S0	S1	Fonction réalisée	Opération
	0	$Q[3..0] \leq Q[3..0]$	Pas de chargement
0	1	$Q[3..0] \leq I[3..0]$	Chargement parallèle
1	0	$Q[3..1] \leq Q[2..0], Q[0] \leq IR$	Décalage à gauche avec IR
1	1	$Q[2..0] \leq Q[3..1], Q[3] \leq IL$	Décalage à droite avec IL

1. Écrire l'entité qui décrit la vue externe de ce registre.
2. Donner une architecture comportementale (tableau ci-dessus) en utilisant un process.

Méthode

Dans le process, vous pouvez utiliser l'instruction if... Then.....elsif et l'instruction for.....loop.

Exercice 06

VI

Écrire un code VHDL de reconnaissance de forme de 8 bits qui fonctionne comme suite :

- Les bits sont reçus séquentiellement dans un seul signal "dataIN"
- Les 8 bits d'identification "pattern" sont reçus en parallèle, "load"=1 indique la réception de nouvelle bits à identifier.
- La correspondance se fait par le "found" qui sera mis à 1 si le signal "pattern" est le même que le signal "data", c'est-à-dire il y a une identification.
- Un signal "clk" synchronise le tous.
- "Reset" permet de mètre à zéro le système.

Complément

Inputs: Reset, clk, pattern (8 bits), dataIN, load

Output: found