

IFT 1227 – Architecture des ordinateurs

Devoir 2

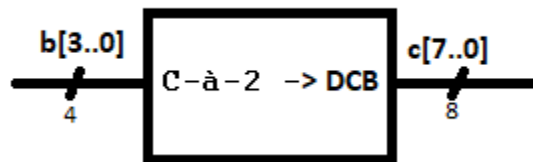
- Remise : Le 12 mars 2024 avant minuit au plus tard.

1. Conception des circuits combinatoires (VHDL)

a) Modélisation comportementale

Concevoir un circuit combinatoire convertisseur complément à 2 sur 4 bits vers le code BCD (Binary Coded Decimal) ou DCB (décimal codé en binaire) sur 8 bits. Dans cette convention, les signes sont encodés par les combinaisons binaires 1011 (« + ») et 1101 (« - ») dans les 4 bits les plus significatifs, et les 4 bits restants encodent en binaire un chiffre en base 10.

Le circuit doit prendre un bus de 4 entrées **b** (**b3** . . **b0**) et produire un bus de 8 sorties **c** (**c7** ... **c0**).

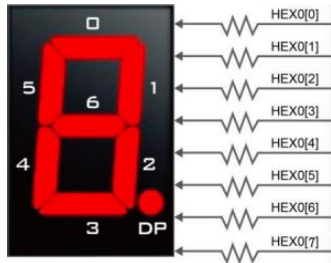


Par exemple, le code binaire en entrée 0111 doit générer la chaîne 1011 0111 qui encode « + » et « 7 ». Un autre exemple, le code binaire 1111, encodant -1 en complément à 2, doit générer en sortie la chaîne 1101 0001 (« - » « 1 »).

- #### b) Créer un autre module comportemental en VHDL pour le circuit suivant : le circuit doit prendre un bus de 8 entrées **c** (**c7** . . **c0**) et produire deux vecteurs de bits de 7 sorties chacun, **affS** et **affCh**.



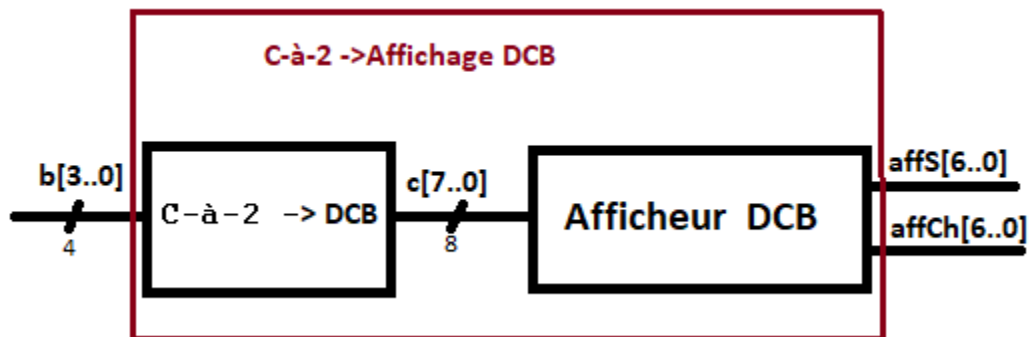
Ce circuit reçoit en entrée un code BCD (binaire codé décimal) et produit deux vecteurs de bits qui permettent de configurer les deux afficheurs 7 segments. **affS** contrôle l'afficheur du signe et **affCh** celui du chiffre. En cas de signe "+", l'afficheur du signe doit être éteint, et en cas de signe "-", le segment 6 doit être activé. Le deuxième vecteur de bits, **affCh**, affiche le chiffre en base 10 encodé par les 4 bits les moins significatifs de l'entrée **c**.



Pour pouvoir utiliser les afficheurs 7 segments sur la carte, il faudra attribuer la valeur 0 pour allumer un segment et 1 pour éteindre.

c) Modélisation structurelle

Créer un module structurel qui prendra comme entrée une valeur en complément à 2 sur 4 bits et générera les deux vecteurs de contrôle pour les deux afficheurs.

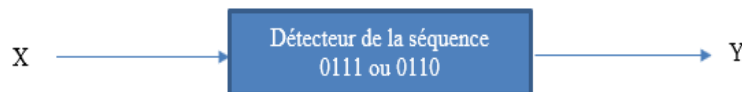


Pour ce modèle, placez le code des deux modules comportementaux dans le sous-répertoire "**files**" créé dans le répertoire de votre projet du modèle structurel.

- d) Synthétisez le circuit *Complément-à-2 vers Affichage BCD* et vérifiez son comportement en connectant les entrées du circuit aux interrupteurs SW[3] – SW[0], et les sorties **affS** et **affCh** aux afficheurs HEX1 et HEX0 respectivement.

I. Circuits logiques séquentiels

Concevoir un circuit séquentiel qui prend en entrée, de manière séquentielle, un à un, des bits, et produit 1 en sortie lorsque les quatre derniers bits lus sont 0111 ou 0110, et 0 en sortie sinon. Il faut bien sûr produire 0 en sortie lorsque moins de quatre bits ont été lus.



Exemple de comportement : X : 0101100101111011

Y : 0000010000010000

Temps : —————>

- a) Créer une machine à états de type Moore;
- b) Codez la machine en VHDL ;
- c) Écrire un testbench auto-vérificateur en VHDL et placez-le dans le répertoire de votre projet FSM;
- d) Vérifiez le comportement de votre circuit par simulation;

Remise

La remise électronique : Créer un fichier compressé (*.zip) contenant 2 répertoires de vos 2 projets (q.1(c) et q.2(b)) + le fichier pdf du rapport.

Le rapport doit contenir :

- q.1(a), q.1(b), q.1(c) - images des circuits synthétisés (netlist);
- q.2.a – le diagramme de transition d'états (FSM) de votre machine;
- q.2.b – image du circuit synthétisé (Tools > Netlist Viewers > RTL viewer);
- q.2.c – code source de votre testbench

Le rapport pourrait être écrit à la main (version scannée doit être convertie en format pdf) ou par traitement de texte. Le rapport devra être clair, **présentable** et concis en format pdf.

Barème de correction

Numéro 1 – 50 pts; 1a – 15; 1b – 15; 1c – 15; 1d – 5;

Numéro 2 – 45 pts; 2a – 10; 2b – 15; 2c – 10, 2d – 10;

Présentation de rapport, respect des directives - 5;

Total : 100 points

Bon travail!