Examen d'architecture des ordinateurs

1ère année département informatique ENSEEIHT

20/01/2014

1. Algèbre de Boole (2 pts)

Pa la méthode de votre choix, démontrer que :

$$A.(B \oplus C \oplus D) = A.B \oplus A.C \oplus A.D$$

2. Arithmétique binaire (2 pts)

a. Sur 16 bits, effectuer la somme des deux nombres suivants (codés en hexadécimal) C516 + 4B16

b. Interpréter ce calcul en arithmétique non signée (binaire pur) en donnant les valeurs décimales correspondantes

c. Interpréter ce calcul en arithmétique signée (complément à 2) sur 16 bits en donnant les valeurs décimales correspondantes

3. Circuits combinatoires (3 pts)

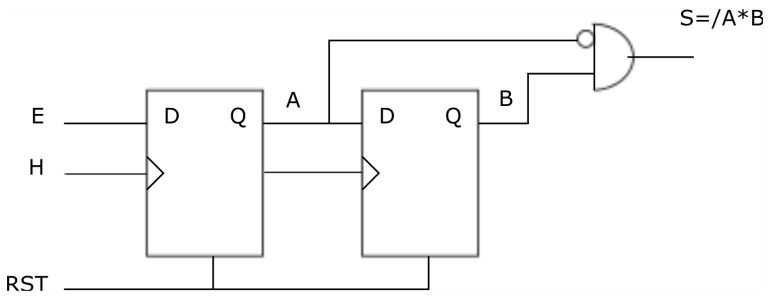
On souhaite concevoir un circuit combinatoire qui calcule le produit de 2 nombres entiers non signés de 2 bits, avec un résultat sur 4 bits ; son interface SHDL est :

```
module mult(a[1..0], b[1..0] : s[3..0])
```

Donner la table de vérité de ce circuit combinatoire, simplifier les équations des sorties et écrire le module en SHDL.

4. Circuit séquentiel (4 pts)

On considère le circuit séquentiel suivant, d'entrée E, sortie S et horloge H :



a. Est-ce un circuit de MOORE ou de MEALY (justifier) ?

b. En partant de l'état initial où toutes les bascules sont à 0, énumérer tous les états possibles en leur donnant des noms (a, b, etc.) et reconstituer le graphe d'états du circuit.

c. Construire la table de transitions à partir du graphe d'états.

d. Simplifier la table et donner le graphe d'états simplifié. Combien a-t-il d'états ? Pouvez-vous deviner la fonction du circuit ?

5. Circuit de time-out (3 pts)

On souhaite ajouter un module time_out au processeur craps, version simplifiée du module pwm, et fonctionnant avec l'interface suivant :

```
module time_out (rst, clk, din[31..0] : to[31..0], to_out)
```

- si din[31..0] est égal à 0, le module reste à l'arrêt ;
- sinon :
  - si to[31..0] est égal à 0, din[31..0] est chargé dans to[31..0] et to_out est mis à 0
  - sinon to[31..0] est décrémenté à chaque top d'horloge et to_out passe à 1 quand to[31..0] atteint la valeur 0

a- donner la description en shdl de to[0], to[1], to[2], to_out (Un module zero32(e[31..0]):zero) est fourni) (1 point)

b- ce module doit être accessible, dans CRAPS, via l'adresse 0x70000000 dans laquelle on peut écrire la valeur initiale du time_out (pour le lancer) et on peut lire la valeur courante du time_out. Dire dans quelle(s) partie(s) du processeur CRAPS on doit effectuer les ajouts nécessaires ; en donner la description en shdl (2 points)

6. Ajout d'une nouvelle instruction dans CRAPS (3 pts)

On souhaite rajouter dans CRAPS de nouvelles instructions « compare and branch » de syntaxe assembleur :

```
cmpb<cond> %ri, %rj, label
```

Le terme <cond> est identique à celui des instructions b<cond>

Par exemple, l'instruction : `cmpbeq %r1, %r2, loop` est équivalente à la suite des deux instructions :

```
cmp   %r1, %r2
beq   loop
```

a- Choisir un format pour cette instruction (1 point)

b- Donner le graphe correspondant à ajouter dans le séquenceur, en y indiquant la condition de chaque transition et les valeurs des microcommandes correspondantes (2 points)

7. Programmation de CRAPS (3 pts)

a- Ecrire un sous-programme qui affiche le contenu du registre %r1 sur l'afficheur de la carte.

b- Ecrire un programme qui lit les switches et affiche la valeur correspondante lorsque l'on appuie sur le bouton btn3, et s'arrête dès que la valeur des switches est égale à 0.