Examen d'architecture des ordinateurs

Apprentissage Info & réseau

27/11/2013

# Algèbre de Boole (2 pts)

Pa la méthode de votre choix, démontrer que :

# Arithmétique binaire (2 pts)

1. Sur 16 bits, effectuer la somme des deux nombres suivants (codés en hexadécimal)  FEC516 + C34B16
2. Interpréter ce calcul en arithmétique non signée (binaire pur) en donnant les valeurs décimales correspondantes
3. Interpréter ce calcul en arithmétique signée (complément à 2) sur 16 bits en donnant les valeurs décimales correspondantes

# Circuits combinatoires (3 pts)

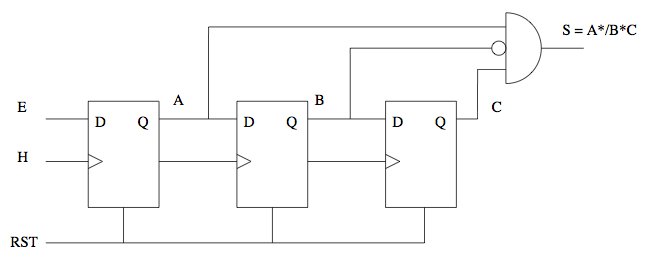
On souhaite concevoir un circuit combinatoire qui calcule le produit de 2 nombres entiers non signés de 2 bits, avec un résultat sur 4 bits ; son interface SHDL est :

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.

# Circuit séquentiel (4 pts)

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



1. Est-ce un circuit de MOORE ou de MEALY ?
2. Énumérer tous les états possibles (les appeler a, b, etc.) et construire le graphe d’états associé.
3. Construire la table de transitions à partir du graphe d’états.
4. Simplifier la table et donner le graphe d’états simplifié. Combien a-t-il d’états ? Pouvez-vous deviner sa fonction ?

# Compteur avec remise à zéro synchrone (3 pts)

Concevoir un compteur 3 bits synchrone d’interface cpt3sclr(rst, clk, sclr: s[2..0]) qui compte normalement lorsque sclr = 0 , et qui repasse à 0 au front d’horloge lorsque sclr = 1. Fournir un schéma ou des équations SHDL.

# Microcommandes associées à une instruction (3 pts)

Compléter le tableau suivant avec la suite des microcommandes à envoyer pour exécuter l'instruction : ld [%r3+%r4], %r5

(On pourra répondre directement sur le sujet ou recopier le tableau sur la copie)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **areg** | **breg** | **dreg** | **cmd\_ual** | **oe\_num** | **write** | **commentaire** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# Programmation de CRAPS (3 pts)

Écrire un sous-programme (sans programme principal) qui calcule dans %r2 la somme des %r1 premiers nombres entiers.