

Examen d'architecture des ordinateurs

Apprentissage Info-réseaux

28/11/2012

1. Algèbre de Boole – simplifications algébriques (2 pts)

Simplifier algébriquement l'expression suivante :

$$S = \bar{A} B C + A C + A \bar{B} \bar{C} + \bar{A} \bar{B}$$

2. Algèbre de Boole - (3 pts)

On veut réaliser un circuit qui indique qu'un nombre SIGNÉ de 2 bits a1,a0 est strictement supérieur à un nombre SIGNÉ de 2 bits b1,b0, d'interface:

```
module scmp2(a1, a0, b1, b0 : sup)
```

Par exemple a1,a0=01 est strictement supérieur à b1,b0=10

- dessiner la table de vérité de sup
- simplifier sup à l'aide d'une table de Karnaugh
- écrire le circuit sous forme d'un module SHDL

3. Arithmétique binaire (2 pts)

- Ecrire 1234 en binaire pur, puis en hexadécimal
- Quel est l'intervalle des valeurs possibles pour des nombres non signés codés en binaire pur sur 16 bits ?
- Quel est l'intervalle des valeurs possibles pour des nombres signés codés en complément à 2 sur 16 bits ?
- Réaliser l'addition sur 16 bits :

```
1010 0110 1111 0000
+ 1000 0000 1100 0011
```

et interprétez les arguments et le résultat en arithmétique signée et non signée. On indiquera dans les 2 cas s'il y a ou non débordement, et pourquoi.

4. Circuit séquentiel - compteur (4 pts)

Ecrire un module SHDL qui réalise un compteur sur 4 bits (a, b, c, d), d'interface :

```
module count4(rst, clk: a, b, c, d)
```

Modifier ce compteur pour qu'il puisse charger une valeur de 4 bits (ai, bi, ci, di) lors de l'arrivée d'une commande load. Son interface sera :

```
module count4i(rst, clk, load, ai, bi, ci, di : a, b, c, d)
```

Lorsque load=0, le compteur compte normalement. Lorsque load=1, au front d'horloge, le compteur se charge avec la valeur (ai, bi, ci, di) et continuera ensuite de compter à partir de cette valeur (dès que load repassera à 0). On pourra décrire le compteur modifié en SHDL ou sous forme d'un schéma, au choix.

5. Circuit séquentiel – graphe d'états (3 pts)

On désire réaliser un circuit marche/arrêt d'un appareil électrique, de type MOORE. Le circuit a 2 bouton d'entrées A et B et une sortie M. Si l'appareil est arrêté (M=0) et que A et B sont relâchés, il se met en marche (M=1) lorsqu'on appuie sur A (B relâché) puis qu'on relâche A (B relâché) puis qu'on appuie sur B (A relâché). Aucune autre séquence ne met en marche un appareil arrêté. Si l'appareil est en marche (M=1) et que A et B sont relâchés, il s'éteint (M=0) lorsqu'on appuie sur B (A relâché), puis qu'on relâche B (A relâché) puis qu'on appuie sur A (B relâché). Aucune autre séquence n'arrête un appareil en marche.

Dessiner le graphe de MOORE de ce circuit, puis la table de transitions, en la simplifiant si c'est possible. Combien de bascules seraient-elles nécessaires à la conception de ce circuit ?

ON NE CONCEVRA PAS LE CIRCUIT.

6. CRAPS : séquencement d'une instruction (2 pts)

Dessiner le sous-graphe de séquencement d'une instruction du type :

```
ld    [%rs1+simml3], %rd
```

7. Programmation de CRAPS (4 pts)

Écrire un programme pour CRAPS qui stocke dans un tableau en mémoire les 100 premières valeurs de la suite de Fibonacci ($u_0=1$, $u_1=1$, $u_{n+2}=u_{n+1}+u_n$). Le tableau sera placé à l'adresse 0x100 ; le programme commencera à l'adresse 0.