

# Examen d'architecture des ordinateurs

---

*Apprentissage Info & réseau*

*27/11/2014*

## 1. Algèbre de Boole (2 pts)

On considère l'expression :

$$s = a \bar{c} \bar{d} + \bar{a} b c + a b \bar{c} \bar{d} + b c d + a b$$

- La simplifier algébriquement
- La simplifier à l'aide d'une table de Karnaugh

## 2. Arithmétique binaire (2 pts)

- Sur 9 bits, effectuer la somme des deux nombres suivants (codés en binaire) 110001110 + 011010111
- Interpréter ce calcul en arithmétique non signée (binaire pur) en donnant les valeurs décimales correspondantes
- Interpréter ce calcul en arithmétique signée (complément à 2) sur 9 bits en donnant les valeurs décimales correspondantes

## 3. Circuits combinatoires (3 pts)

Concevoir, par la méthode de votre choix, un circuit qui calcule la valeur absolue d'un nombre codé en complément à 2 sur 4 bits. Donnez un schéma ou des équations logiques.

## 4. Circuit séquentiel (5 pts)

On veut réaliser une serrure à code simplifiée. Le clavier est composé de 3 touches : '#', 'A', 'B' et une led s'allume lorsque la bonne séquence a été tapée au clavier. La séquence d'ouverture est : '#', 'B', 'A'. Le circuit à réaliser aura l'interface :

```
module serrure(rst, clk, dieze, a, b : ouvert)
```

On fait les hypothèses suivantes :

- l'horloge `clk` est suffisamment rapide pour ne manquer aucun événement sur les touches
- le clavier est conçu pour qu'à chaque instant, il y ait zéro ou une touche appuyée, jamais plusieurs
- entre 2 appuis successifs, les trois touches sont relâchées

La séquence précise d'événements sur les touches qui déclenche l'ouverture est : appui sur #, aucun appui, appui sur B, aucun appui, appui sur A. Tout appui ensuite arrête l'ouverture et commence une nouvelle séquence.

- expliciter les vecteurs des entrées et des sorties et dessiner un graphe de Moore de ce système
- simplifier éventuellement ; quelle est la taille du vecteur d'états ?
- NE PAS synthétiser ce circuit

## 5. Compteur avec remise à zéro synchrone (2 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.

## 6. Microcommandes associées à une instruction (2 pts)

Fournir la suite des microcommandes qui sont produites lors de l'exécution du morceau de programme suivant :

```
        add    %r0, 1, %r1
loop:   addcc  %r1, -1, %r1
        bne    loop
```

## 7. Programmation de CRAPS (4 pts)

Réaliser par programme la serrure codée décrite au 4- :

- #, A et B seront placés sur les switches `sw[2]`, `sw[1]`, `sw[0]` respectivement
- le signal d'ouverture sera affiché sur la led `ld[0]`

Le programme bouclera tant que la bonne séquence n'aura pas été produite ; quand la bonne séquence sera produite, il s'arrêtera (instruction `ba` bouclée sur elle-même)