

# Examen d'architecture des ordinateurs

---

Apprentissage Info & réseau

20/12/2018

## 1. Arithmétique binaire (2 pts)

- Sur 8 bits, effectuer la somme des deux nombres suivants (codés en binaire) 1001 0110 + 1000 0100
- Interpréter ce calcul en arithmétique non signée (binaire pur) en donnant les valeurs décimales correspondantes et en indiquant si et pourquoi il y a débordement
- Interpréter ce calcul en arithmétique signée (complément à 2) sur 9 bits en donnant les valeurs décimales correspondantes et en indiquant si et pourquoi il y a débordement

## 2. Simplifications algébriques (2 pts)

Simplifier l'expression suivante, en utilisant les théorèmes de l'algèbre de Boole :

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

## 3. Écriture SHDL (2 pts)

Dessiner le schéma correspondant au module SHDL suivant :

```
module sequence (rst, clk, e : s)
  x := e on clk reset when rst ;
  y := x on clk reset when rst ;
  s = e * /x * y
end module
```

## 4. Conception modulaire (2 pts)

Construire un comparateur non signé de 8 bits, d'interface :

```
ucmp8(a[7..0], b[7..0] : sup, eq)
```

avec 2 comparateurs non signés de 4 bits, d'interface :

```
ucmp4(a[3..0], b[3..0] : sup, eq)
```

### 5. Décompteur avec remise à zéro synchrone (2 pts)

Concevoir un décompteur 3 bits synchrone d'interface `cpt3z(rst, clk, sclr: s[2..0])` qui décompte 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 (2 pts)

Donner la séquence des microcommandes (`areg`, `breg`, `dreg`, `cmd_ual`, `oe_num`, `write`) qui sont appliquées pour exécuter l'instruction assembleur CRAPS : `umulcc %r4, 3, %r1`

### 7. Langage machine (2 pts)

Fournir en binaire et en hexadécimal le code machine de l'instruction assembleur CRAPS : `umulcc %r4, 3, %r1`

### 8. CRAPS : programmation (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 0x1000.