



# ELECINF102

## Processeurs et Architectures Numériques

Contrôle de rattrapage

Lundi 15 septembre 2012

Sans calculatrice

Document autorisé : une feuille recto-verso

Durée: 1h30

Ce contrôle comporte trois parties **indépendantes** :

1. Questions de cours
2. Schémas d'architecture : Un dispositif de compte à rebours
3. Codage SystemVerilog : Un compteur modulo 10 ou 3...

N'oubliez pas d'inscrire nom, prénom, et numéro de casier sur votre copie.

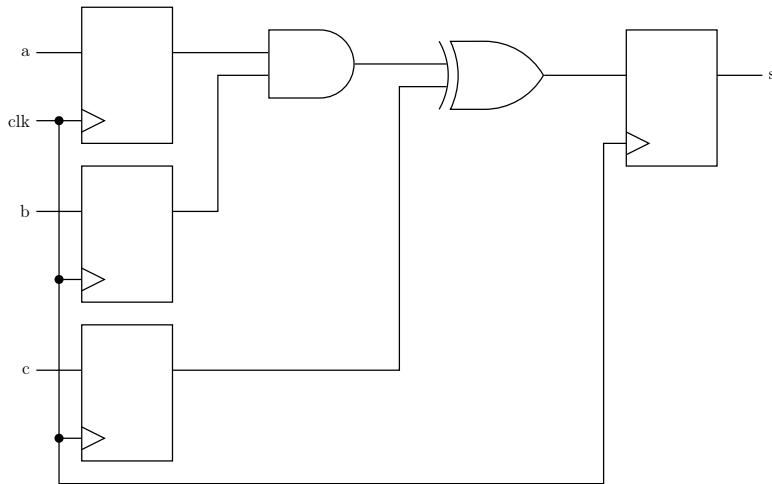
Bon courage !

## 1 Questions de cours

**Question 1** : Indiquez la plage des valeurs représentables en utilisant une représentation entière non signée sur  $n$  bits.

Même question lorsque l'on utilise une représentation en complément à 2 sur  $n$  bits.

**Question 2** : Soit la portion de circuit suivante :



Calculez la fréquence maximale de l'horloge (clk) pour que cette portion de circuit fonctionne correctement sachant que :

- le temps de propagation des portes AND est de 1 ns,
- le temps de propagation des portes XOR est de 2 ns,
- le temps de propagation ( $t_{co}$ ) des bascules est de 1 ns,
- le temps de pré-positionnement ( $t_{su}$ ) des bascules est de 1 ns,
- le temps de maintien ( $t_h$ ) des bascules est de 0,5 ns.

**Question 3** : Décrivez, de façon concise, le comportement d'une bascule D. N'oubliez pas d'indiquer le rôle de chacun des signaux.

**Question 4** : Quelle est la différence entre un signal de remise à zéro (*reset*) *synchrone* et *asynchrone* ?

**Question 5** : Comment le nano-processeur (tel que vu en cours) détecte t'il que la donnée qu'il reçoit de la mémoire est une instruction, l'adresse d'un opérande ou la valeur d'un opérande ?

**Question 6** : Qu'est-ce qu'un mapping mémoire (cartographie mémoire) pour un processeur ?

**Question 7** : Complétez la phrase : un compteur sur 8 bits est un automate (une machine à états finis) à au moins ..... états.

**Question 8** : Considérez deux nombres codés en représentation entière non signée sur  $n$  bits. Combien de bits sont nécessaires pour représenter la *somme* de ces deux nombres.

**Question 9** : Considérez deux nombres codés en représentation entière signée en complément à 2 sur  $n$  bits. Combien de bits sont nécessaires pour représenter le *produit* de ces deux nombres.

**Question 10** : Le langage SystemVerilog est un langage de description de matériel. En quoi est-ce différent d'un langage de programmation classique tel que Java ou C ?

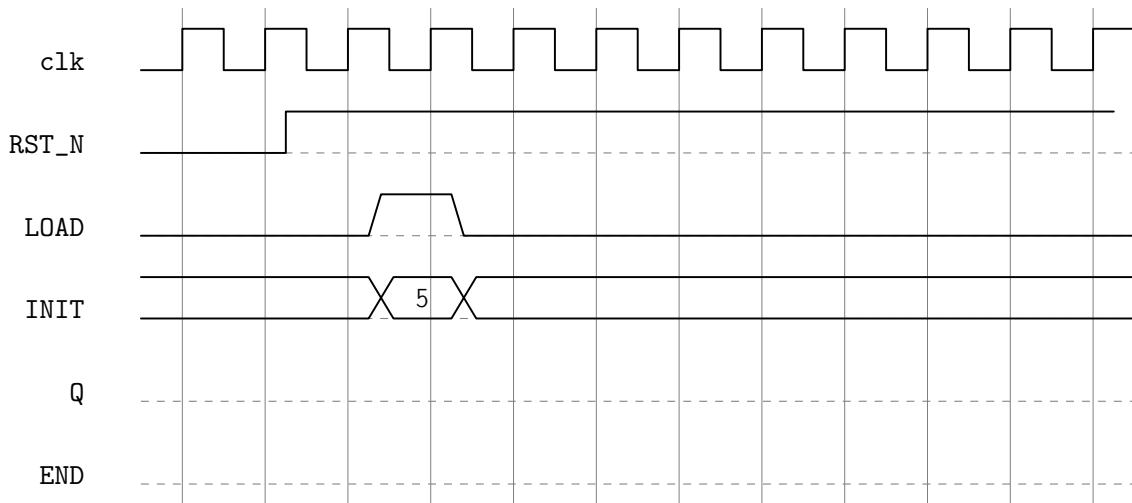
## 2 Compte à rebours

Nous voulons concevoir un système synchrone sur 4 bits permettant de réaliser un compte à rebours.

Les spécifications du système sont les suivantes :

- Un signal **RST\_N** force son état **Q** à zéro de façon *asynchrone*.
- Un signal **LOAD** permet de le charger l'état du système **Q** avec une valeur initiale de début du compte à rebours. Cette valeur est comprise entre 0 et 15 et vient de l'entrée **INIT** du système.
- Une fois arrivé à l'état zéro le système s'arrête et génère un signal **END** indiquant la fin.

1. Complétez le chronogramme suivant en respectant les spécifications ci-dessus.



2. Dessinez le schéma d'un opérateur permettant de savoir si un nombre **A** codé sur 4 bits est égal à zéro en utilisant des portes logiques de base.

Cet opérateur aura en entrée les 4 bits du nombre  $A = a_3, a_2, a_1, a_0$  et en sortie un bit **Z** tel que :

$$Z = \begin{cases} 1 & \text{si et seulement si } A == 0 \\ 0 & \text{si non} \end{cases}$$

3. Dessinez le schéma complet du système en utilisant l'opérateur précédent et des portes logiques combinatoires et séquentielles ainsi qu'éventuellement des opérateurs arithmétiques vus en cours.
4. On veut pouvoir relancer plusieurs fois le compte à rebours sans être obligé de le charger avec une nouvelle valeur. On ajoute le signal **RESTART** pour pouvoir re-déclencher le compte à rebours à partir de la dernière valeur chargée.

Proposez une nouvelle architecture (re-faites un schéma) qui prend en compte cette nouvelle fonctionnalité.

### 3 Compteur modulo

On souhaite concevoir un module permettant de réaliser un compteur synchrone permettant au choix un comptage modulo 10 ou modulo 3.

Voici la description exhaustive des entrées et des sorties de ce module :

Signal	Direction	Nombre de bits	Description
clk	Entrée	1	L'horloge
reset_n	Entrée	1	Signal de remise à zéro asynchrone actif à l'état bas
comptage	Entrée	1	Signal autorisant le comptage (s'il est à 1 lors d'un front d'horloge, le compteur s'incrémentera, sinon, il garde sa valeur)
m3	Entrée	1	Signal indiquant si le comptage est modulo 3 (m3=1) ou modulo 10 (m3=0). Au front d'horloge, si le mode de comptage change alors, le compteur doit <ul style="list-style-type: none"><li>• repartir à zéro si la valeur courante n'est pas compatible avec le modulo choisi,</li><li>• continuer en séquence si la valeur courante est compatible avec le modulo choisi.</li></ul>
valeur	Sortie	4	Valeur actuelle du compteur

Écrivez le code SystemVerilog d'un tel module. Ce code doit être complet, il doit contenir la définition du nom du module, des ses entrées/sorties, ainsi que le code interne.