

# Architecture : circuits numériques et éléments d'architecture

## Examen de TP

ENSIMAG 1A

2020-2021

Consignes et conseils :

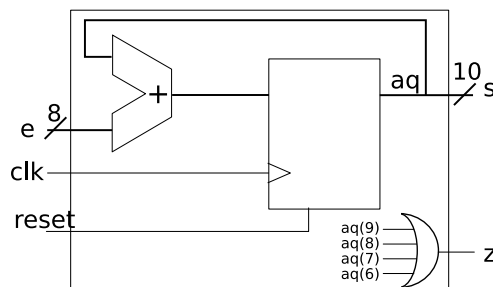
- Durée : 2h
- A titre préventif, remplissez vos nom et prénom dans le fichier `/sujet/qui.txt`. En cas de soucis sur votre session, ça sera une aide précieuse.
- Le barème est donné à titre indicatif.
- Les exercices sont indépendants et peuvent être traités dans le désordre.
- **IMPORTANT** : votre copie est à rendre sous TEIDE (teide.ensimag.fr accessible via le navigateur de la session d'examen) à l'issue de l'épreuve. Les retards non autorisés seront pénalisés.
  - Votre archive au format tgz<sup>1</sup> ne contiendra que les fichiers VHDL modifiés et les chronogrammes demandés (png et fichier de configuration wcfg).
  - Les chronogrammes réalisés doivent être enregistrés sous forme d'images. Utilisez l'outil de capture d'écran de la session d'examen (Activities->Type to search...->Screenshot) et pensez à rendre vos captures lisibles (choix du zoom, séparateurs et pertinence des signaux).
  - Si vous souhaitez commenter vos réponses, insérez directement vos commentaires dans les fichiers VHDL.
- Pour rappel, pour utiliser les outils, il est nécessaire d'initialiser les variables d'environnement avec : `source /bigsoft/Xilinx/Vivado/2019.1/settings64.sh`

### Ex. 0 : Survie (0 pts)

Remplissez par précaution le fichier `/sujet/qui.txt`.

### Ex. 1 : Bête de somme (6 pts)

Dans cet exercice, on vous demande d'implanter le circuit suivant :



**Question 1** Implantez ce circuit en complétant le fichier `vhd/accu.vhd` sans utiliser d'instanciation (pas de port map).

---

1. `tar cvfz rendu.tgz <fichiers à rendre>`

**Question 2** On vous fournit le fichier de test `tb_accu.vhd` qui permet de simuler le fonctionnement du circuit. Pour rappel, la simulation se lance avec la commande (depuis le répertoire `exo1`) :

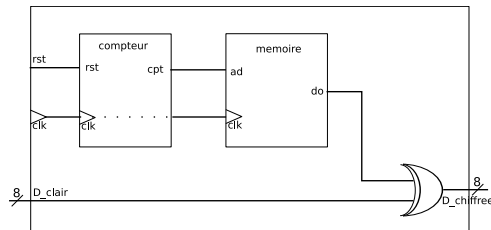
```
make run_simu TOP=accu
```

Imprimez un chronogramme sous le nom "exo1.png". Votre chronogramme affichera les entrées et les sorties du circuit sous un radix approprié (entier non signé pour les vecteurs) et avec un niveau de zoom adéquat.

Vous enregistrerez la configuration d'affichage dans le fichier "exo1.wcfg"

## Ex. 2 : Chiffrement XOR (5 pts)

Dans cet exercice, on veut réaliser un module de chiffrement XOR avec une taille de clé paramétrable via un paramètre générique. Concrètement, on va utiliser une clé de 512 octets stockée dans une mémoire. La taille de clé paramétrable permettra d'indiquer combien d'octets seront utilisés dans cette clé. Nous utiliserons un compteur paramétrable pour couvrir les indices de la taille de clé choisi (de 0 à *taille* - 1), et ainsi récupérer octet par octet la clé de manière cyclique. Le message à chiffrer arrive en flux continu sur l'entrée *D\_clair* du module. Le message chiffré s'obtient par un simple XOR bit-à-bit entre cette entrée et l'octet courant de la clé. Les modules compteur et mémoire vous sont fournis. Il vous reste donc à implanter le circuit suivant :



**Question 1** Implantez ce circuit en complétant le fichier `vhd/chiffreur.vhd` en utilisant les modules fournis.

**Question 2** On vous fournit le fichier de test `tb_chiffreur.vhd` qui permet de simuler le fonctionnement du circuit. La simulation se lance avec la commande (depuis le répertoire `exo2`) :

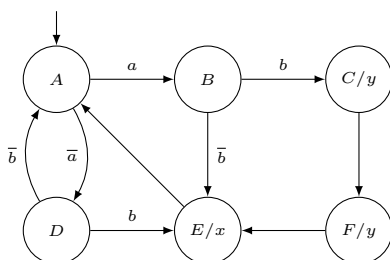
```
make run_simu TOP=chiffreur
```

Imprimez un chronogramme sous le nom "exo2.png". Votre chronogramme affichera les entrées et les sorties du circuit, ainsi que l'état du compteur. Les signaux seront affichés sous un radix approprié et avec un niveau de zoom adéquat.

Vous enregistrerez la configuration d'affichage dans le fichier "exo2.wcfg".

## Ex. 3 : Automate (8 pts)

Soit l'automate suivant :



**Question 1** Implantez cet automate en VHDL dans le fichier `vhd/fsm.vhd` en utilisant un codage implicite.

**Question 2** On vous fournit le fichier de test `tb_fsm.vhd` qui permet de simuler le fonctionnement de votre automate. La simulation se lance avec la commande (depuis le répertoire `exo3`) :

```
make run_simu TOP=fsm
```

Vous imprimerez votre chronogramme sous le nom "exo3.png" en y affichant les valeurs de l'état courant, des entrées et des sorties. Vous ajusterez le niveau de zoom afin de démontrer que votre automate fonctionne correctement.

Vous enregistrerez la configuration d'affichage dans le fichier "exo3.wcfg"