

Nom :  
Prénom :  
Groupe IUT :

/20

Auto1 SIN  
Lundi 6 janvier 2025

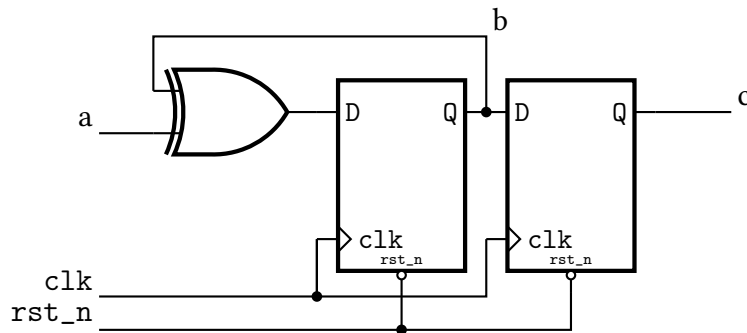
*Aucun document autorisé. Calculatrice non autorisée. Durée : 1h30.  
Les exercices sont indépendants et peuvent être réalisés dans n'importe quel ordre.  
Toutes les réponses doivent être justifiées, et les calculs détaillés.*

## Exercice 1 : Compréhension

/3

### Circuit mystère

Soit le circuit numérique suivant :



Q1

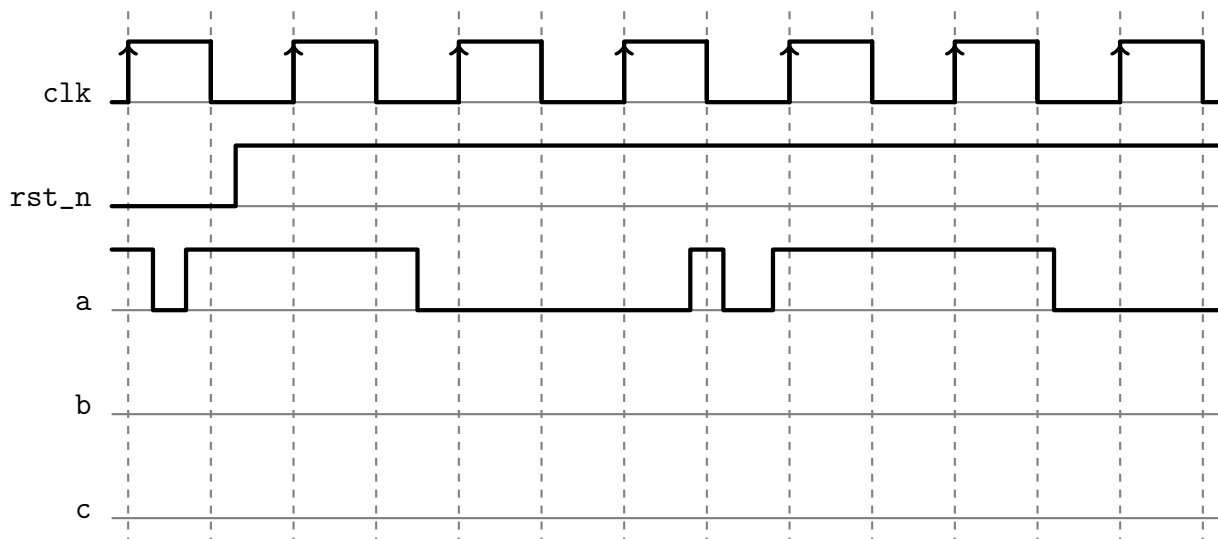
Ce circuit est-il synchrone ?

/0,5

Q2

Compléter dans le chronogramme suivant les valeurs des signaux b et c :

/1



## Code VHDL mystère

Soit le code VHDL suivant où f et s sont de type `STD_LOGIC` :

```
PROCESS (clk, rst_n)
BEGIN
  IF rst_n = '0' THEN
    s <= '0';
    f <= '0';
  ELSE
    IF rising_edge(clk) THEN
      s <= a;
      f <= b;
    END IF;
  END IF;
END PROCESS;

c <= f AND s;
```

Q3

/0,5

Combien de bascules D seront utilisées par Quartus lors de la synthèse ?

Q4

/1

Dessiner le schéma correspondant.

## Exercice 2 : Mesure et affichage d'une durée d'appui

/15

### Présentation

On souhaite concevoir un système permettant de mesurer avec une précision de 200 ms la durée d'appui sur un bouton poussoir, et de l'afficher sur un afficheur dit "thermomètre" à cinq LEDs. La Figure 1 montre différents affichages possibles :

- Initialement, l'affichage est vide, toutes les LEDs sont éteintes ●, comme en Figure 1a.
- Pour des durées d'appui comprise entre 200 ms et 1 s, une LED supplémentaire s'allume ○ chaque fois qu'une durée de 200 ms s'est écoulée. Ceci est illustré en Figures 1b à 1e.
- Si la durée d'appui est supérieure ou égale à 1 s, alors toutes les LEDs s'allument, comme en Figure 1f.



(a) Affichage initial vide



(b) Après un appui d'une durée  $d \in [200 \text{ ms}; 400 \text{ ms}[$



(c) Après un appui d'une durée  $d \in [400 \text{ ms}; 600 \text{ ms}[$



(d) Après un appui d'une durée  $d \in [600 \text{ ms}; 800 \text{ ms}[$



(e) Après un appui d'une durée  $d \in [800 \text{ ms}; 1 \text{ s}[$



(f) Après un appui d'une durée  $d \geq 1 \text{ s}$

FIGURE 1 – Différents affichages possibles sur l'afficheur "thermomètre"

### Attention :

- le reset est asynchrone et **actif à l'état bas** : c'est le seul moyen de réinitialiser le système,
- le bouton poussoir, comme ceux des cartes utilisées en TP, est **actif à l'état bas**,
- les cinq LEDs, quant à elles, sont **allumées par un niveau logique haut**.

### Q5

/2

Représenter ce système, vu de l'extérieur, par une boîte nommée et indiquer les ports d'entrée et de sortie de ce composant en les nommant. Les LEDs sont indépendantes dans ce système : chacune est connectée à un port dédié.

## 2.1 Générateur d'impulsions avec autorisation

/3,5

Le premier composant génère une impulsion d'une durée d'une période d'horloge toutes les 200 ms tant que son entrée ena est à l'état haut. Son ENTITY est donnée ci-dessous : c'est donc essentiellement un compteur avec autorisation indiquant que son maximum est atteint.

```
ENTITY gen_ena_200ms IS
  PORT (
    clk_10kHz    : IN  STD_LOGIC;
    rst_n        : IN  STD_LOGIC;
    ena          : IN  STD_LOGIC;
    max_atteint  : OUT STD_LOGIC);
END gen_ena_200ms;
```

Q6

/1

L'horloge du système a une fréquence de 10 kHz. Jusqu'à combien doit compter le compteur pour générer une impulsion toutes les 200 ms ? Combien de bits sont nécessaires pour stocker la valeur du compteur ?

Q7

/0,5

Faire les déclarations nécessaires dans l'ARCHITECTURE, avant le BEGIN.

```
ARCHITECTURE archi OF gen_ena_200ms IS

BEGIN
```

Q8

/2

Compléter finalement l'ARCHITECTURE pour décrire ce générateur d'impulsion.

```
BEGIN

PROCESS (clk, rst_n)
BEGIN
    IF rst_n = '0' THEN

    ELSE
        IF rising_edge(clk) THEN

        END IF;
    END IF;
END PROCESS;

END archi;
```

## 2.2 Compteur mono-coup

/3,5

Le compteur de cinquièmes de seconde est incrémenté lorsqu'il reçoit une impulsion du composant décrit précédemment. Ce compteur est dit **mono-coup** car une fois sa valeur maximale atteinte il y reste jusqu'au prochain reset.

Q9

/1

Compléter l'ENTITY de ce compteur monocoup.

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY compteur_monocoup_0_5 IS
  PORT (

);
END compteur_monocoup_0_5;
```

Q10

/0,5

Faire les déclarations nécessaires dans l'ARCHITECTURE, avant le BEGIN.

```
ARCHITECTURE archi OF compteur_monocoup_0_5 IS

BEGIN
```

Q11

/2

Compléter finalement l'ARCHITECTURE pour décrire ce compteur mono-coup.

```
BEGIN

PROCESS (clk, rst_n)
BEGIN
  IF rst_n = '0' THEN

  ELSE
    IF rising_edge(clk) THEN

    END IF;
  END IF;
END PROCESS;

END archi;
```

### 2.3 Afficheur thermomètre

/3,5

Q12

/1,5

Compléter ci-dessous la table de vérité du décodeur permettant d'afficher une valeur binaire sur l'afficheur thermomètre.

Entrées			Sorties				

Q13

Décrire en VHDL le décodeur.

/2

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY decodeur_thermo IS
  PORT (

);
END decodeur_thermo;

ARCHITECTURE archi OF decodeur_thermo IS

BEGIN

END archi;
```

## 2.4 Câblage final

/2,5

Q14

/2,5

En considérant que les **COMPONENT** associés aux entités précédentes ont déjà été déclarés, réalisez les instantiations nécessaires pour le “câblage” final du système.

```
BEGIN
```

```
END archi;
```

```
1 ARCHITECTURE archi OF detect_appui IS
2
3   TYPE etat IS (arret, marche);           --
4   SIGNAL etat_actuel : etat;
5   SIGNAL bouton_s    : STD_LOGIC;
6
7 BEGIN
8
9   PROCESS (clk, rst_n)
10  BEGIN
11    IF rst_n = '0' THEN
12      etat_actuel <= arret;
13    ELSIF rising_edge(clk) THEN
14      bouton_s <= bouton;
15
16      CASE etat_actuel IS
17
18        WHEN arret =>
19          IF bouton_s = '1' THEN
20            etat_actuel <= marche;
21          END IF;
22
23        WHEN marche =>
24          IF bouton_s = '1' THEN
25            etat_actuel <= arret;
26          END IF;
27
28        WHEN OTHERS =>
29          etat_actuel <= arret;
30
31      END CASE;
32    END IF;
33  END PROCESS;
34
35 END archi;
```

Q15

/0,5

Comment s'appelle la déclaration utilisée à la ligne 3? Quel est son intérêt?

**Q16**

**/1**

Dessiner le diagramme états-transitions de cette machine d'états.



**Q17**

**/0,5**

Expliquer le problème lors de la mise en œuvre de cette machine d'états dans un circuit réel, en particulier le lien avec la fréquence d'horloge du circuit.

