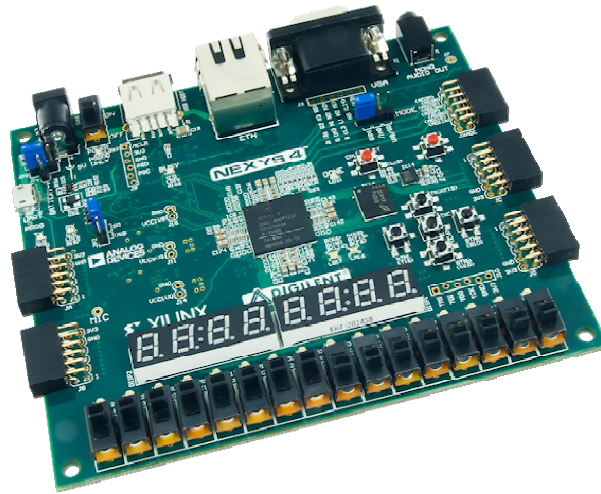


# EN201 : TP Inteface Lecteur MP3

## 1. Présentation de la carte NEXYS 4 :



Le circuit à configurer est un FPGA Xilinx Artix 7 qui est inséré au sein d'une carte NEXYS4 commercialisée par la société Digilent. La référence complète du circuit FPGA est **XC7A100T-1CSG324**. Cette carte comprend, outre le circuit, quatre afficheurs 7 segments, un quartz qui fournit par défaut un signal d'horloge de 100 MHz, des LEDs, des interrupteurs et d'autres périphériques comme décrit sur la Figure 1. Un site web dédié à cette carte contient toutes les informations nécessaires à son utilisation :

<https://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,1184&Prod=NEXYS4>

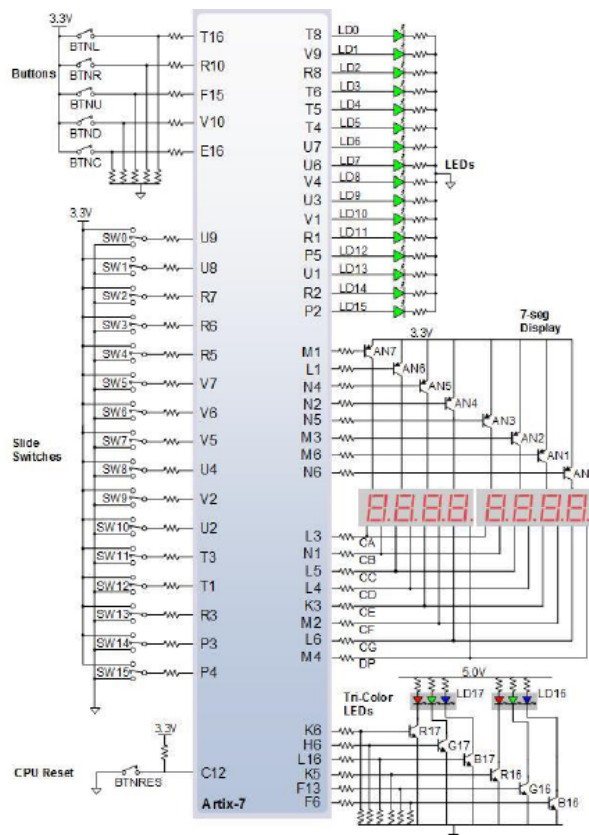


Figure 1 : schéma bloc des accès E/S au circuit FPGA de la carte NEXYS4

## Particularité dans la gestion des afficheurs 7 segments :

La carte comprend huit afficheurs à anode commune comme le montre la Figure 2. Cela signifie que chaque segment est activé (allumé) en appliquant un 0 sur l'entrée adéquate. Par exemple, pour afficher le nombre deux, il faut la combinaison a=0, b=0, c=1, d=0, e=0, f=1 et g=0 sur les entrées.

Par ailleurs, un seul des afficheurs est accessible à un instant donné. C'est pourquoi, il faudra se baser sur la persistance rétinienne (capacité de l'œil à conserver des images vues superposées à une image que l'on est en train de voir) pour gérer l'affichage.

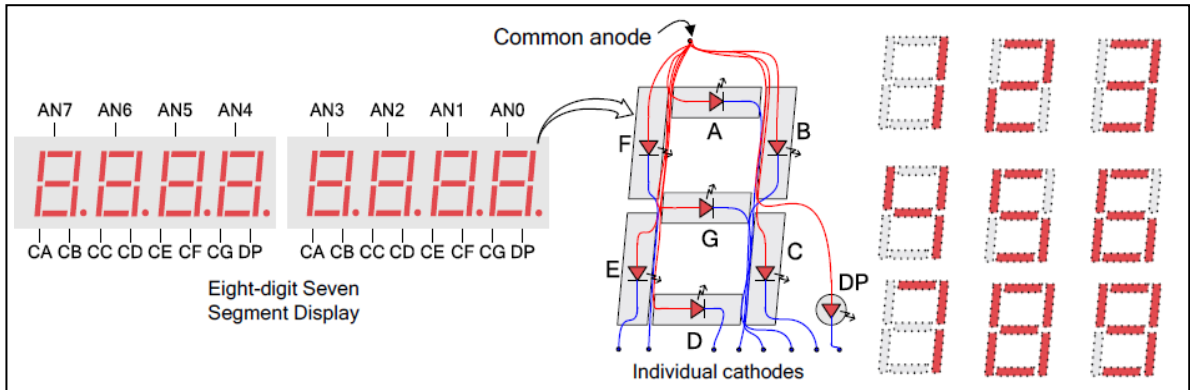


Figure 2 : principe de fonctionnement des afficheurs 7-segments

## Affectation des entrées/sorties du circuit FPGA:

L'affectation des entrées/sorties du circuit FPGA implique la spécification d'un fichier de type XDC. Les affectations nécessaires à chacun des périphériques sont indiquées sur la carte NEXYS4. Un fichier de type XDC générique est disponible sur le site de Digilent.

Par exemple, l'affectation des entrées/sorties pour accéder aux huit afficheurs 7 segments est :

```
set_property PACKAGE_PIN N6 [get_ports AN[7]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[7]]
set_property PACKAGE_PIN M6 [get_ports AN[6]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[6]]
set_property PACKAGE_PIN M3 [get_ports AN[5]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[5]]
set_property PACKAGE_PIN N5 [get_ports AN[4]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[4]]
set_property PACKAGE_PIN N2 [get_ports AN[3]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[3]]
set_property PACKAGE_PIN N4 [get_ports AN[2]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[2]]
set_property PACKAGE_PIN L1 [get_ports AN[1]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[1]]
set_property PACKAGE_PIN M1 [get_ports AN[0]]
set_property IOSTANDARD LVCMOS33 [get_ports AN[0]]
```

L'accès au quartz d'horloge implique quant à lui l'affectation suivante :

```
set_property PACKAGE_PIN E3 [get_ports horloge]
set_property IOSTANDARD LVCMOS33 [get_ports horloge]
create_clock -add -name horloge -period 10.00 -waveform {0 5} [get_ports horloge]
```

## 2. Cahier des charges :

### 2.1. Objectif :

L'architecture à concevoir, puis à prototyper sur une carte NEXYS4 devra permettre de gérer les interfaces d'un lecteur MP3. Dans le détail, il faudra afficher sur 3 afficheurs 7-segments la valeur courante d'un compteur/décompteur de 1 à 599. L'affichage devra avoir la cadence suivante : unité 100 ms, dizaine 1s et centaine 10s. De plus, un nombre compris entre 1 et 9 plus un point devra être affiché sur un quatrième afficheur 7-segments. Le compteur associé devra pouvoir être incrémenté ou décrétementé par deux boutons poussoirs.

Par ailleurs, un codage sera affiché sur les 4 autres afficheurs 7-segments. Quatre valeurs seront nécessaires suivant les états de la machine d'états :

- [--] : ETAT INIT ou ETAT STOP
- ---] : ETAT PLAY-FWD
- [--- : ETAT PLAY-BWD
- ---- : ETAT PAUSE

### 2.2. Fonctionnement :

Le principe d'utilisation de la carte contenant l'architecture demandée est le suivant. Avant tout chose, il faut initialiser toutes les bascules de l'architecture en positionnant *RAZ* à 1, puis en le remettant à 0. Quatre des afficheurs doivent alors afficher **5.001** pour les fonctions de comptages/décomptages.

La fréquence de fonctionnement de l'horloge associée au projet est de 100 MHz. Cela signifie que des signaux d'activation sur les fronts montants de l'horloge (*Clock Enable*) sont nécessaires pour afficher de l'information à des fréquences exploitables par l'œil humain. Dans ce projet, deux signaux d'activation sont nécessaires :

- $CE_{\text{perception}}$  : ce signal permet de cadencer les huit afficheurs 7 segments en tenant compte de la perception rétinienne.  
⇒ *Fréquence* ~ 3 kHz
- $CE_{\text{affichage}}$  : ce signal permet de gérer le défilement des nombres entre 1 et 599  
⇒ *Fréquence* ~ 10 Hz (100 ms)

### 2.3. Consignes à respecter pour la définition de l'architecture :

L'entité devra respecter le schéma suivant au niveau du nom et du type des E/S :

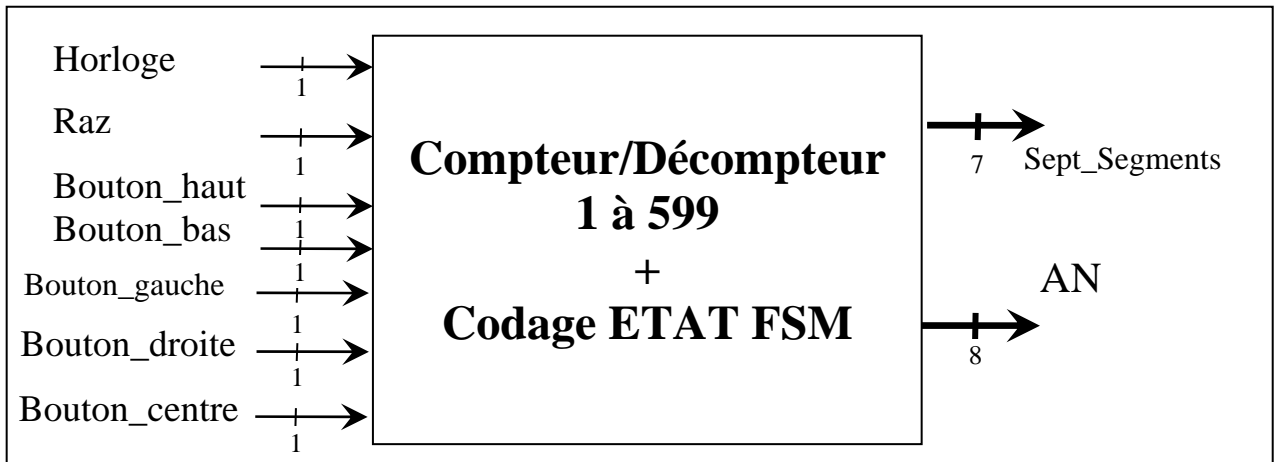


Figure 3 : entité de l'architecture

#### Entrées :

- ✓ Horloge : horloge du circuit fournie par un quartz à 100MHz,
- ✓ Raz : remise à zéro synchrone de l'architecture globale,
- ✓ Bouton\_haut : entrée d'activation de la fonction de comptage 1 à 9
- ✓ Bouton\_bas : entrée d'activation de la fonction de décomptage 1 à 9
- ✓ Bouton\_gauche : entrée de commande de la machine d'états (B\_LEFT)
- ✓ Bouton\_droite : entrée de commande de la machine d'états (B\_RIGHT)
- ✓ Bouton\_centre : entrée de commande de la machine d'états (B\_CENTER)

#### Sorties :

- ✓ Sept\_Segments : affichage sur les 7-segments de la donnée courante du compteur et du codage (*Bus de 8 bits*).
- ✓ AN : indication de l'afficheur 7-segments courant, (*Bus de 8 bits*).

Afin de vous aider dans la réalisation de ce travail, nous vous donnons la découpe d'architecture. Cette dernière est constituée de 8 modules comme décrit dans la Figure 4.

Les 8 modules de l'architecture principale ont les fonctionnalités suivantes :

- ✓ **Gestion des signaux d'activation (*gestion\_freq*)** : génération de deux signaux d'activation sur les fronts montants de l'horloge (*Clock Enable*). Les fréquences ont été définies préalablement à la page 1 du document.
- ✓ **Filtrage séquentiel associée aux boutons (*detec\_impulsion*)** : synchronisation des 5 entrées associées aux 5 boutons poussoirs à l'aide de bascules. Utilisation du signal d'horloge pour la synchronisation.
- ✓ **Machine à états finis (*fsm\_MP3*)** : machine à états finis du projet. Elle a été spécifiée puis décrite au cours du TD n° 1 du module EN201.
- ✓ **Fonction compteur/décompteur (*cpt\_1\_599*)** : il s'agit du compteur/décompteur dont la

- ✓ valeur varie de 1 à 599 en fonction des signaux Play-Pause, Restart et Forward provenant de la FSM. Utilisation du signal  $CE_{\text{affichage}}$  comme signal d'activation.
- ✓ **Fonction compteur/décompteur ( $cpt\_1\_9$ )** : il s'agit d'un compteur/décompteur dont la valeur varie de 1 à 9 en fonction des signaux Volume\_UP et Volume\_DW provenant de la FSM.
- ✓ **Transcodage pour afficheur sept segment ( $Trans$ )** : les valeurs issues des blocs compteur/décompteur représentent des nombres entiers codés sur n-bits. Ce bloc permet donc le transcodage de ces informations vers un format compatible avec un afficheur sept segments. Il permet aussi de spécifier le codage retenu sur les quatre autres afficheurs 7-segments.
- ✓ **Sélection de l'anode ( $mod8$ )** : les huit afficheurs sept-segments sont alimentés par huit anodes différentes qu'il faut sélectionner de façon cyclique. Ce bloc génère les signaux d'activation de ces anodes à l'aide d'un compteur modulo 8.
- ✓ **Module d'affichage 7 segments ( $mux8$ )** : commandé par le signal provenant du module mod8, ce multiplexeur sélectionne l'une des huit données à afficher sur les afficheurs 7 segments de manière synchrone avec le choix de l'anode. C'est un point important de l'architecture pour organiser correctement l'affichage.

Par ailleurs, l'architecture est entièrement synchrone. Ceci implique :

- les bascules, les registres et les compteurs sont commandés par la même horloge.
- la copie de l'entrée sur la sortie d'un bloc séquentiel sur le front montant de l'horloge est autorisée par un signal de type CE (Clock Enable).
- l'entrée synchrone de mise à 0 n'est utilisée qu'à l'initialisation du circuit. **Elle ne doit en aucun cas être utilisée pendant le fonctionnement normal du circuit.**

#### 2.4. Consignes à respecter pour le déroulement du projet

Il est demandé de travailler de manière progressive avec des étapes intermédiaires devant être validée par votre encadrant :

- a-) L'utilisation des outils ne doit débiter que lorsque le circuit est découpé en fonctions élémentaires (compteur, registre, décodeur, etc) et que les interconnexions entre les modules sont parfaitement définies.
- b) Dans un deuxième temps, il faut se focaliser sur chacun des modules et les valider par des simulations comportementales à l'aide de testbenchs décrits en VHDL.
- c) Enfin, il faut associer les 7 blocs constituant l'architecture au sein d'une description VHDL. La fonctionnalité du module final devra également être simulée avant la synthèse, l'implémentation, la génération du bitstream et la configuration du circuit FPGA.

Remarque : les descriptions de certains modules nécessaires à ce travail sont similaires à celles du projet LOTO du semestre 5. Il est donc conseillé que réutiliser autant que possible vos descriptions VHDL de l'année passée.

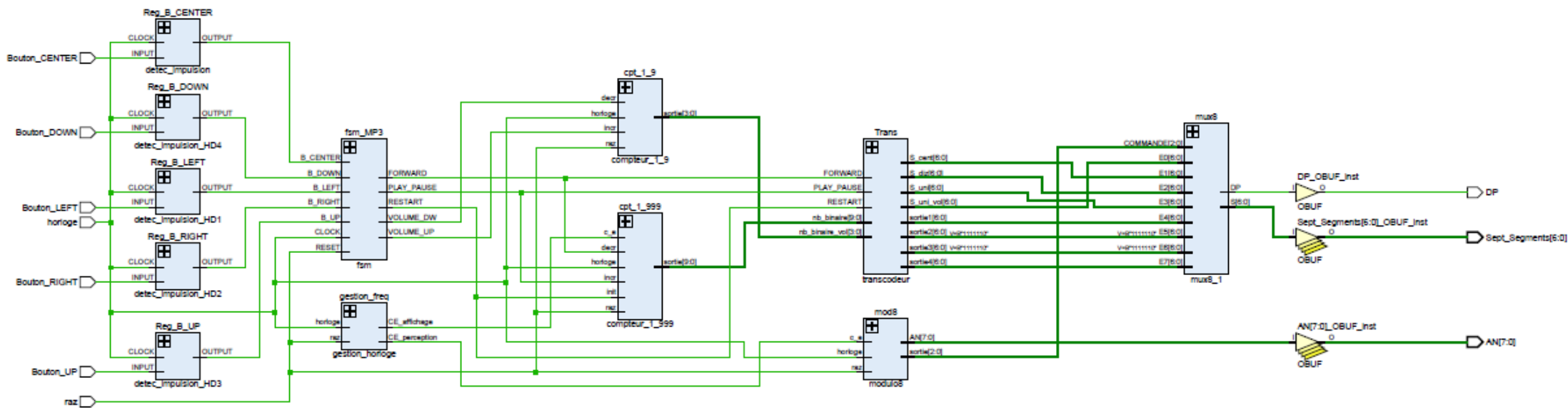


Figure 4 : architecture globale du projet

