

PROJET DE SYSTEMES NUMERIQUES 2018-2019

REVEIL NUMERIQUE

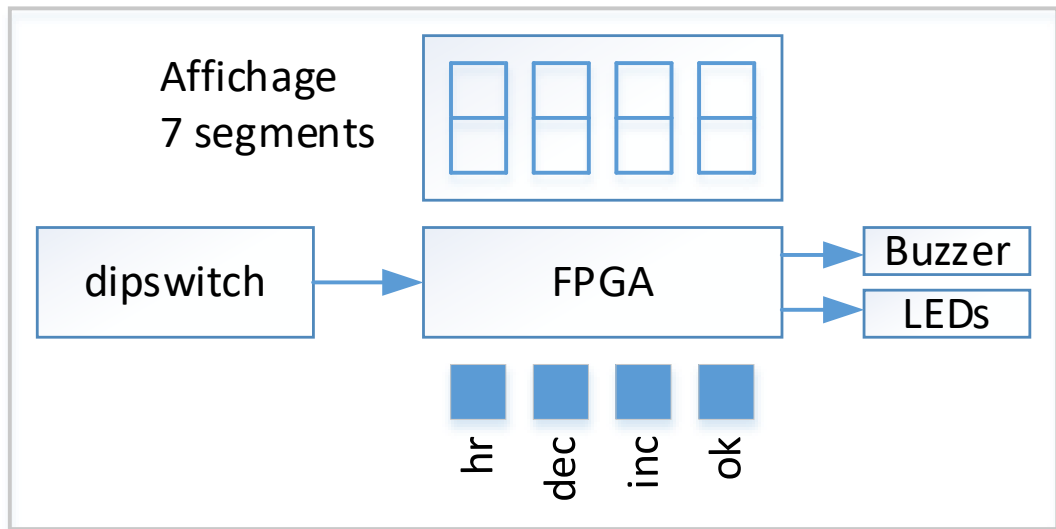
He-Arc / Olivier Gloriod, Yves Meyer / novembre 2018

INTRODUCTION

Ce projet consiste à décrire en VHDL un réveil numérique afin d'être synthétisé sur la carte Kit Xilinx 7.

- Le travail en dehors des heures de cours est indispensable à l'aboutissement du projet.
- Chaque personne doit tenir à jour un « cahier de laboratoire » individuel où il note proprement ce qu'il a fait personnellement et quand il l'a fait (date et heures), les problèmes rencontrés, la raison de certains choix, etc. Le but est d'avoir une traçabilité individuelle du déroulement du projet. Chaque étudiant doit m'envoyer par email son cahier de laboratoire à la fin de chaque leçon.
- Au délai de restitution, chaque groupe doit rendre (les documents fait proprement à la main et scannés sur les imprimantes suffisent largement) :
 - Le cahier de laboratoire en format pdf.
 - Les schémas bloc en format pdf.
 - Le graphe des états en format pdf.
 - Le répertoire du projet Xilinx Vivado avec les fichiers sources et le(s) banc(s) de test (testbench).
 - Faire une démonstration sur le kit Xilinx lors de la dernière leçon. Des questions seront posées afin de déterminer la compréhension du travail réalisé.
- Délais pour la restitution : INF1a et b mercredi 23 janvier 2019 en fin de leçon.
- Il est autorisé de collaborer entre les groupes, par contre la copie ou la transmissions de documents interdits.
- Tous les documents non restitués dans les délais ou copiés entres groupes seront taxés de la note 1.
- La note du projet a le même poids qu'un CP.
- Ne pas réinventer la roue, les blocs simples étudiés en cours avec quelques modifications suffisent ne pas utiliser de bloc complexes !

CAHIER DES CHARGES PRELIMINAIRE



Le réveil permet d'afficher l'heure actuelle ou l'heure de réveil sur les 4 afficheurs 7 segments sous la forme dizaines des heures (HD), unités des heures (HU), dizaines des minutes (MD) et unités des minutes (MU).

Les fonctions sont commandées par 4 boutons poussoirs.

En plus du réglage du réveil par les boutons poussoirs 13 dip-switch permettent d'introduire l'heure ou l'heure de réveil.

En parallèle du buzzer une LED est allumée pour visualiser l'alarme. Un bit du dip switch permet d'empêcher l'enclenchement du buzzer.

Le réglage de l'heure et du réveil se fait à l'aide des boutons poussoirs.

En mode réglage le chiffre réglé doit clignoter à une fréquence de 4Hz.

Les différentes fonctions et boutons

- Le terme "activer" pour l'alarme indique que l'alarme pourra sonner quand l'heure sera égale à l'heure de réveil sans tenir compte des secondes.
- Le terme "arrêter" indique que l'alarme ne sonne plus.

Affichage et réglages

Hors mode réglage :

1. Le choix de l'affichage heure ou réveil se fait avec le bouton HR.
2. Le chargement de l'heure ou de l'heure de réveil à l'aide des dip-switch se fait avec un bouton à choix.
3. Le passage en mode réglage se fait avec le bouton OK.

Dans le mode réglage :

1. Le passage du réglage de HD à HU à MD à MU et sortie du mode réglage se fait avec le bouton OK.
2. L'augmentation des chiffres avec le bouton INC.
3. La diminution des chiffres avec le bouton DEC.
4. Le chargement par dip-switch n'est pas possible

Réveil et alarme

Hors mode réglage :

1. Quand l'alarme sonne elle est désactivée et la pression de l'un des boutons l'arrête. Aucune autre fonction n'est activée.
2. Le passage en mode réglage désactive l'alarme.
3. L'activation de l'alarme se fait avec un bouton à choix.
4. Le chargement de l'heure ou de l'heure de réveil avec les dip-switch ne désactive pas l'alarme.

Dans le mode réglage :

1. L'alarme est désactivée
2. Le chargement par dip-switch n'est pas possible

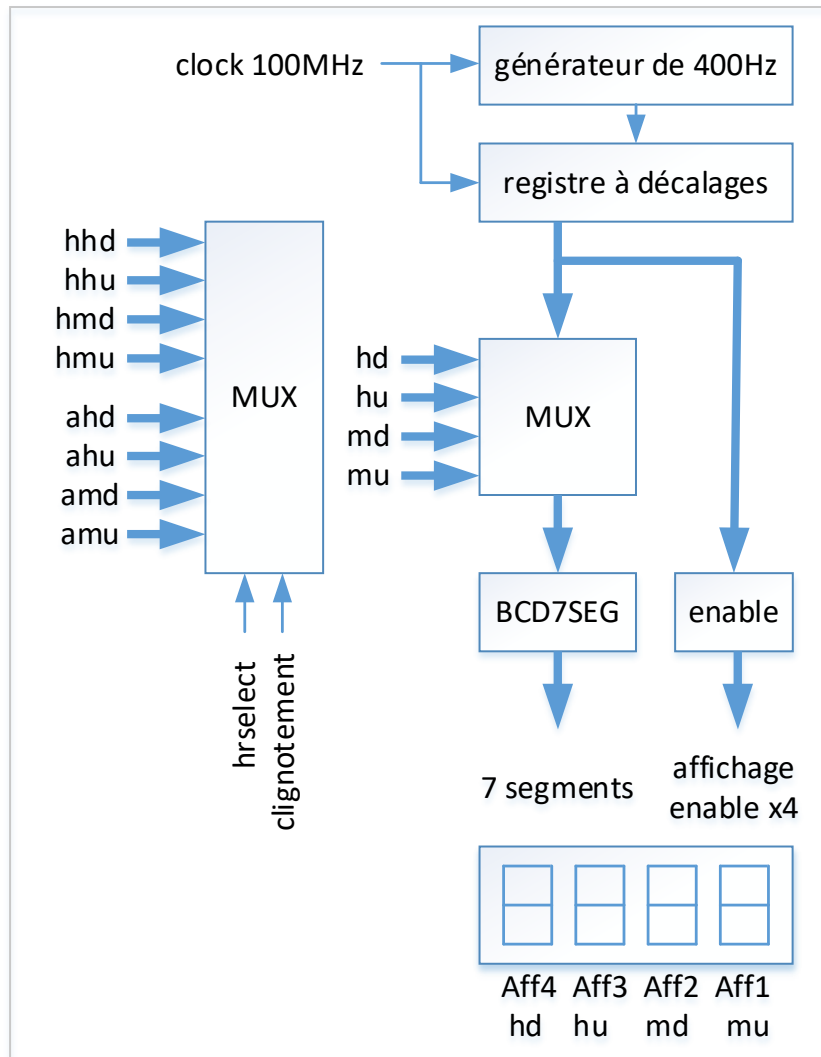
LEDs

Les LED (bicolores ou barregraphe) servent à indiquer ce qui est affiché (heure ou réveil), mode normal ou réglage, alarme activé, alarme enclenchée ainsi que d'autres informations permettant d'aider au debug.

Travail à réaliser

1. Dessiner le schéma bloc du composant affichage qui contient lui-même les blocs suivants :

multiplexeurs
 décodeurs BCD 7 segments
 registre à décalage
 diviseur de fréquence



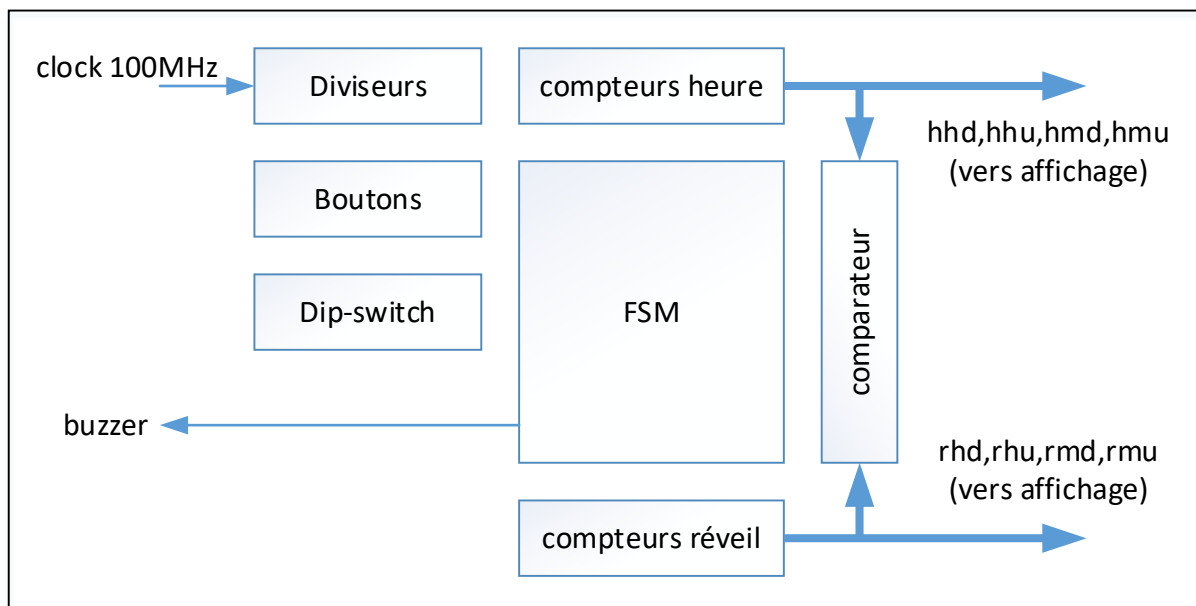
2. Dessiner le schéma bloc du composant compteurs qui contient lui-même les blocs suivants :

Ce bloc est composé de 4 compteurs BCD (HD, HU, MD, MU) cascades.

Le même bloc peut être utilisé pour les heures et pour l'alarme, il faut bien réfléchir à tous les signaux nécessaires (load, enable, inc, dec,...)

3. Dessiner le schéma bloc top du réveil qui contient lui-même les bocs suivants, l'intérieur des composants ne doit pas être dessiné :

- Diviseurs: Un premier diviseur divise le clk (100MHz) pour fabriquer le signal enable des minutes pour les compteurs. Un deuxième diviseur pour le clignotement en mode réglage à 4Hz rapport cyclique 50%, utiliser une bascule T.
Prévoir une entrée test sur ces diviseurs pour pouvoir diviser par un petit nombre (exemple 10) pour diminuer le nombre de cycles d'horloge de la simulation.
- Synchro: synchronise les 4 boutons poussoir avec le clk y compris détection de flanc.
- FSM: la machine d'états qui gère l'horloge.
- Heure: Composant compteurs (heuresMinutes pour l'heure).
- Reveil: Composant compteurs (heuresMinutes pour l'heure de réveil).
- Affichage: Composant qui sert à l'affichage.
- Compare: compare l'heure et l'heure de réveil.



4. Dessiner le graphe des états de la machine d'états.
5. Créer un projet Vivado.
6. Dans le projet Vivado, créer et écrire le VHDL structurel pour les blocs Top, compteurs et affichage.
7. Dans le projet Vivado, créer et écrire le VHDL comportemental des autres blocs.
8. Créer dans le projet Vivado créer un banc de test (testbench) au top de la hiérarchie permettant de tester tous les cas du réveil.

9. Débugger votre système en traçant dans le simulateur les signaux internes (compteurs, affichages, ...) jusqu'à ce que tout fonctionne. Attention cette phase prend du temps ne pas s'y prendre à la dernière minute !
10. Une fois la totalité du projet testé en simulation, créer un fichier de contrainte
11. Synthétiser, implémenter et programmer l'application sur le Kit Xilinx 7 et tester sur le kit.