Projet - contrôleur VGA

FPGA et VHDL

Efrei Paris : M1 2017-2018

KUATE Hervé ; BAH Jedeine

SOMMAIRE

1. Introduction
2. Spécifications
   1. Description fonctionnelle
   2. Structure.
3. Détails de bloc
   1. Diviseur d’horloge
   2. Synchronisor
   3. Bloc Principale
4. Gestion de la mémoire

Conclusion

1. Introduction

Le but de ce travail pratique est de nous exercer sur les connaissances entrepris en classe et nous apprendre à faire à utiliser un FPGA (Spartan 3) en langage VHDL.

Tout au long de notre Travail, deux outils principaux ont été utilisé parmi lesquelles :

* Ordinateur
* Application Xilinx ISE.
* Spartan 3
* Moniteur compatible VGA

1. Spécifications
   1. Description fonctionnelle

Notre dispositif permet d’afficher des images sur l’écran via le câble VGA.

* 1. Implémentation

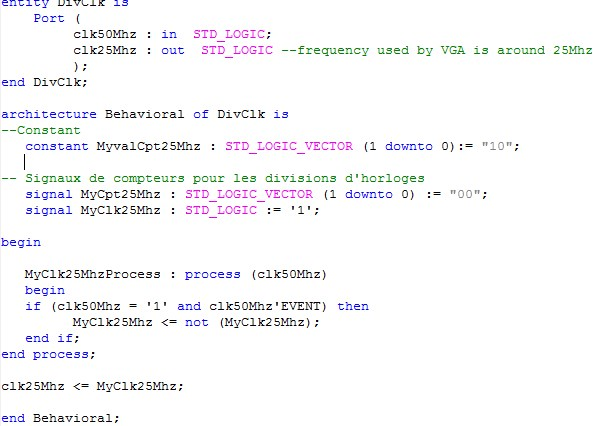
Le matériel utilisé est le FPGA XILINX Spartan 3 XC3S200 à 50 Mhz. La sortie VGA est la principale sortie utilisée. Il nous faudra également utiliser la SRAM disponible sur la carte.

* 1. Structure.

2 principales blocs ont été utilisés :

* Diviseur d’horloge : pour divisé la frequence de la carte de 50mhz à 25 mhz
* Synchroniseur

1. Détails des blocs
   1. Diviseur d’horloge



Description fonctionnelle: convertir la frequence de base de la carte en 25Mhz

* 1. Synchronisor

Description fonctionnelle : assez complexe, il contient plusieurs fonctions parmi lesquels

* Gestion des signaux de synchronisation de l’horloge hv et vs :
* Gestion des couleurs de sortie
* Balayage vidéo sur l’écran
* Mini test d’affichage (drapeau français)

Lors de nos recherches sur la gestion des signaux de synchronisation, nous avons remarquer que le timing standard du VGA a changé

Ancien timing :

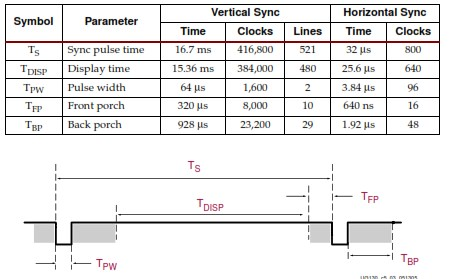


Figure 1 Ancient timing

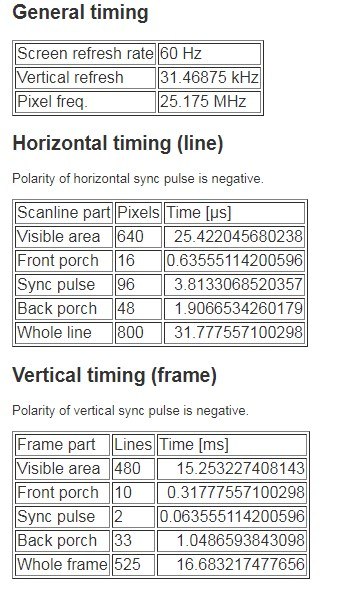
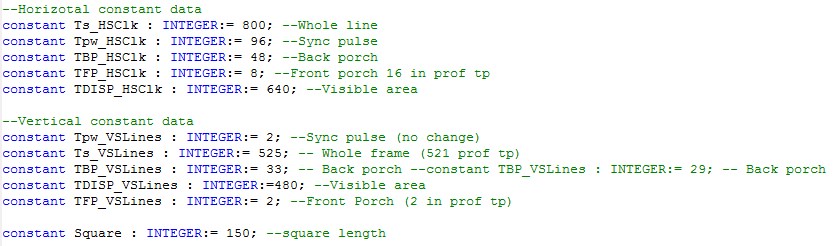


Figure 2 Nouvelle timing

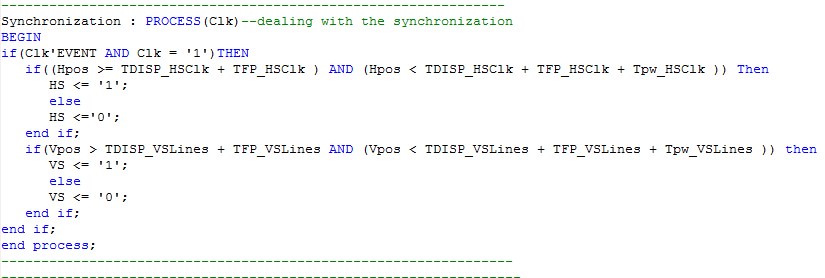
Ce changement est l’une des grandes difficultés rencontrées lors de notre travail.

Nous avons représenté ces constant sur notre *synchronisor* via le script :



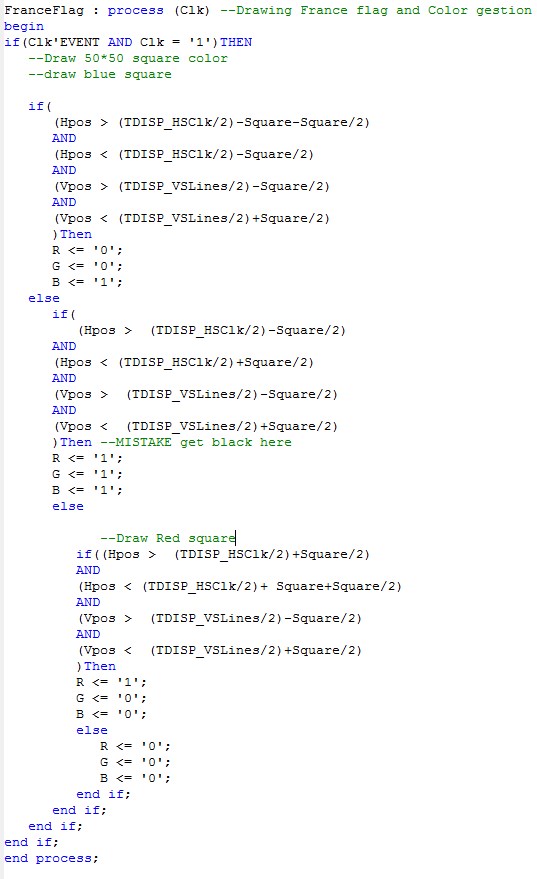
**-Ballayage vidéo :**

Le ballayage vidéo a été faite par implémentation de deux boucles  *for* en language VHDL :



**-test d’affichage du drapeu francais :**

Très simple à implementer, il faut juste borné les position de X et de Y auquel les valeurs de nos couleurs doivent changer, on contient le code :



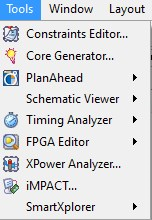
* 1. Bloc principale

Notre blocs principales, le *TP2\_VGA*, prend nos deux sous bloc *DicClk* et *Synchronisor* comme instance .

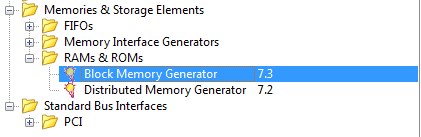
X:\Herve\Data\ShareX\Screenshots\2017-12\_pn_2017-12-20_23-23-57.jpg

1. Gestion de la mémoire

Le dernier exercice nos impose l’utilisation d’un block de ram RAMb16S1. Après des recherches, nous avons découvert qu’il vous parcourir le Core Generator pour en créer ce block.

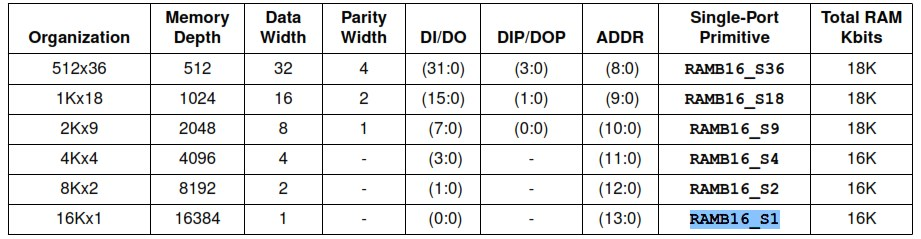


Puis, ouvrir le *Block Memory Generator*



Là-bas, il nous faut configurer notre ram pour qu’elle correspond au RAMB16S1. Après avoir feuileté le datasheet du Spartan 3, nous avons observé les paramètre Memory Depth, Data Width et Organisation à mettre sur notre bloc de mémoire pour qu’elle correspond.

Malheureusement, nous avons pas peu mettre en pratique nos bloc créer car manque de connaissance sur l’utilisation de la ram.



Conclusion

Les cartes FPGA sont très complexes à comprendre à cause des feuilletages sur leur datasheet.

Tout le travail n’a pas pu être effectué à cause du sur chargement de temps et surtout du manque de connaissance sur la notion de la mémoire.