

Université de Cergy-Pontoise

Projet d'Architecture des Ordinateurs

Rapport de projet

LUNDI 04 AVRIL 2016

Auteurs :

Paul VALENTIN

Thomas LEFEBVRE

Jury :

Laurent RODRIGUEZ

Alexandre MARCASTEL

Table des matières

1	Introduction	2
1.1	Cadre et objectif	2
1.2	Membres	2
1.3	Outils	2
1.4	Contenu du rapport et informations complémentaires	2
2	Fonctionnement du processeur	3
2.1	Spécifications	3
2.2	Principaux composants du processeur	4
3	Problèmes rencontrés	6
3.1	Logistique	6
3.2	Humain	6
3.3	Solutions	6
4	Conclusion	7

1 Introduction

1.1 Cadre et objectif

Ce projet rattaché au module Architecture des Ordinateurs a été réalisé dans le cadre du semestre 6 de la 3ème année de licence à l'université de Cergy-Pontoise. Le but du projet a été de concevoir un processeur 16 bits capable d'exécuter un jeu d'instruction définies.

1.2 Membres

- Paul VALENTIN
- Thomas LEFEBVRE

1.3 Outils

Materiel

Pour ce faire nous avons utilisé un *FPGA* de type *Cyclone II* qui est une carte programmable servant à simuler le comportement d'un processeur.

Logiciel

Cette carte a été utilisé en association avec le langage de description de matériel VHDL (VHSIC Hardware Description Language) nous permettant comme son nom l'indique de décrire les différents composants du processeur ainsi que leur comportement. Au niveau logiciel nous avons utilisé également utilisé les logiciels Quartus et ModelSim pour l'écriture, la compilation et le test logiciel du code écrit en VHDL.

La synchronisation du travail entre les différents membres du groupe a été faite via un dépôt Git et le service Github nous permettant de collaborer plus facilement à distance.

1.4 Contenu du rapport et informations complémentaires

Ce rapport de projet sert principalement à expliciter les caractéristiques de notre processeur, sa conception ainsi que son fonctionnement. Nous aborderons également les différents problèmes qui ont pu subvenir lors de la réalisation de ce projet.

Nous ne voyons pas d'inconvénient à la réutilisation de notre code, merci tout de même d'en indiquer les auteurs. En ce qui concerne l'exécution du code, une carte de type FPGA est nécessaire pour son exécution physique mais on peut également utiliser ModelSim/Quartus pour en simuler le fonctionnement via *Test Benchs*

2 Fonctionnement du processeur

2.1 Spécifications

- Processeur 16bits
- Entièrement générique (donc potentiellement n bits)
- 5 opérations (MV, MVI, ADD, SUB, MULT)

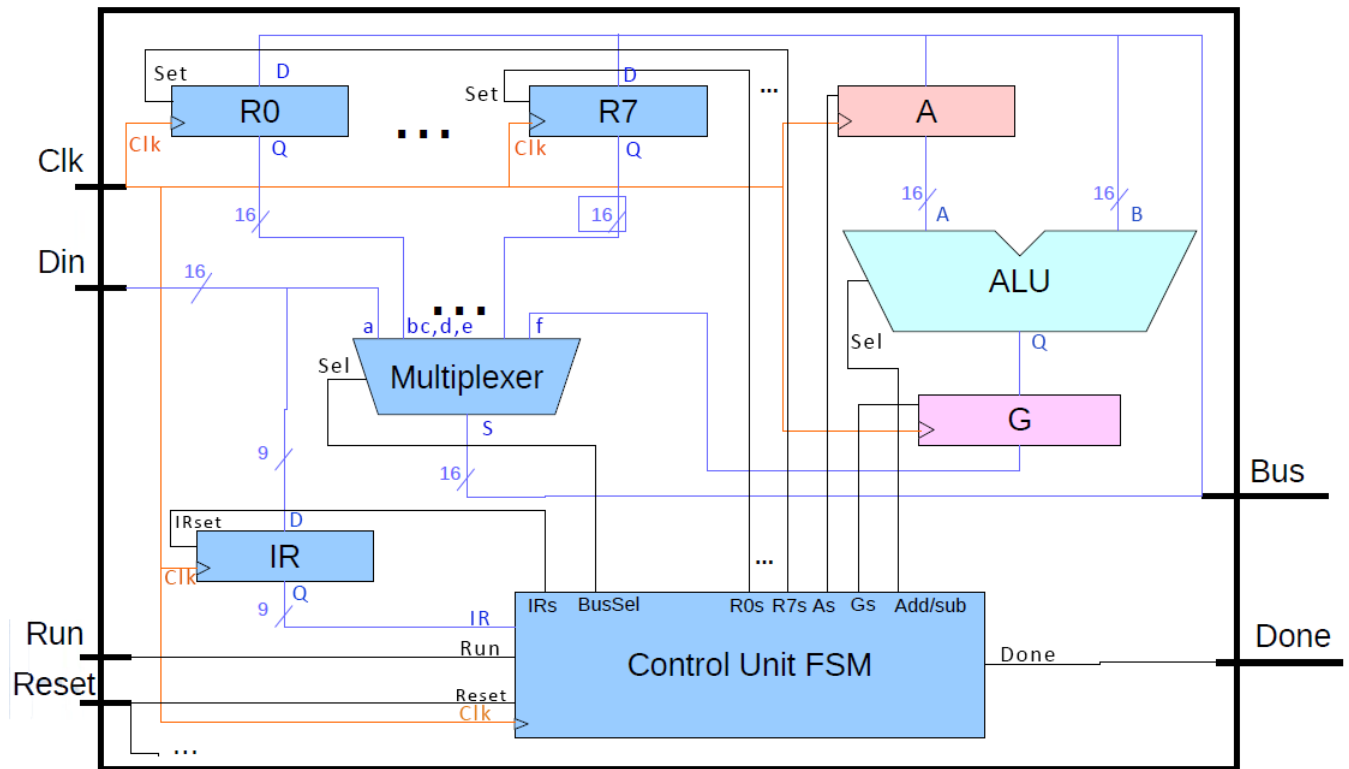


FIGURE 1 – Schéma représentatif du processeur

2.2 Principaux composants du processeur

FSM

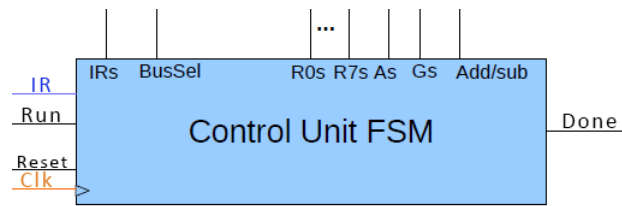


FIGURE 2 – Schéma de la FSM

La FSM (Finite State Machine) est ici l'unité de contrôle de notre processeur, elle dirige les opérations au sein de notre processeur (chemin des instructions)

CODOP :

- **0000** - MVI - Déplacement immédiat des données dans un registre
- **0001** - MV - Déplacement des données d'un endroit à l'autre
- **0010** - ADD - Addition
- **0011** - SUB - Soustraction
- **0100** - MULT - Multiplication

ALU

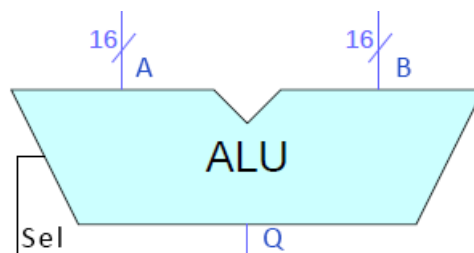


FIGURE 3 – Schéma de l'ALU

L'Unité Arithmétique et Logique est chargé d'effectuer les différentes opérations définies dans la FSM. Ces opérations sont envoyées sous forme d'instructions stockées dans les registres et sélectionnées grâce au MUX.

Registres

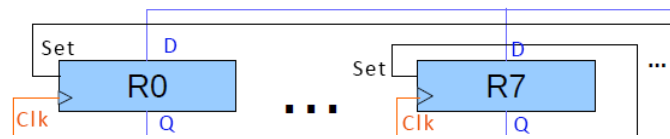


FIGURE 4 – Schéma des registres

Les registres servent à stocker les instructions traitées par l'ALU.

Registre d'instruction (IR)

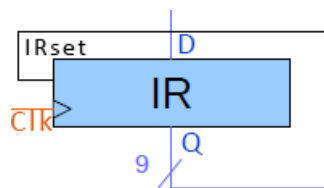


FIGURE 5 – Schéma du registre d'instruction

Le registre d'instruction stocke directement les instructions en cours d'exécution.

Multiplexeur (MUX)

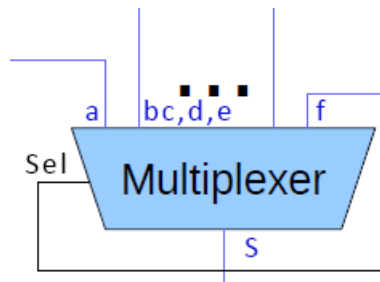


FIGURE 6 – Schéma du MUX

Le MUX sélectionne une entrée parmi N et renvoie le contenu de cette entrée (instruction) en sortie. Ils permettent donc d'aiguiller l'information dans le circuit.

3 Problèmes rencontrés

Au cours de ce projet, plusieurs problèmes ont été rencontrés, cette partie donne le détail de ces différents problèmes ainsi que les solutions apportées.

3.1 Logistique

Incompatibilités et difficulté d'installation

Premièrement nous avons eu des problèmes au niveau de l'installation des outils. En effet, il n'a pas été facile de trouver une version accessible de Quartus et nous nous sommes donc tournés vers ModelSim pour travailler en dehors de l'université. Aussi l'interface de Quartus s'avère assez lourde et austère, n'a pas été facile de prime abord. Quand aux incompatibilités logicielles Quartus/ModelSim n'étant pas disponibles sur Mac, nous avons utilisé un environnement Windows/Linux.

Pertes

Nous avons également perdu une bonne partie des composants réalisés en TD et ce car nous n'étions pas regardant quand à la sauvegarde des fichiers .vhd écrit durant les TD du fait que notre binôme de TD ne devait pas constituer, à la base, notre groupe de projet.

3.2 Humain

Temps

Les problèmes précédents ont pu retarder la réalisation et la finalisation de notre processeur. De plus, la période durant laquelle nous devions réaliser ce processeur était particulièrement chargée en cours assez denses et en projets tout aussi importants que celui-ci (projet de synthèse, projet de GPI) et nous ne pouvions par conséquent pas donner la priorité à notre processeur.

3.3 Solutions

Les solutions à ces différents problèmes ont été mises en oeuvre le plus souvent dès leur apparition. En ce qui concerne le problème d'incompatibilité logiciel avec MacOS (système utilisé par un membre du groupe), il n'a pas été bloquant et nous avons pu continuer à travailler sur environnement Windows/Linux. Concernant la prise en main du logiciel, cela s'est fait au cours du projet et pour ce qui est du temps limite nous avons dû concentrer nos efforts sur la fin du projet pour terminer le processeur.

4 Conclusion

En conclusion, ce projet a été pour nous l'occasion d'aller au delà de la théorie sur l'Architecture des Ordinateurs et de pouvoir enfin aborder la partie pratique de la réalisation d'un processeur. En effet, ce projet s'est inscrit dans la continuité de module d'Architecture et du projet correspondant en L2 (réalisation d'un processeur 4bits sur Logisim).

Nous tenons à adresser nos remerciements à M.Laurent Rodriguez, enseignant du cours d'Architecture et M.Alexandre Marcastel présent lors des séances de TD et tous deux jury lors de la soutenance de projet ainsi qu'à M.Benoit Miramond qui a réalisé, conjointement avec M.Laurent Rodriguez, le pdf du cours d'Architecture des ordinateurs.