

MINI-PROJET AVEC L'ARCHITECTURE Y86

Nathan VANBESELAERE, Sacha DUPERRET

19 avril 2023

Résumé

Rapport détaillant le travail effectué en binôme pour adapter l'architecture Y86 aux consignes.

Table des matières

1	Exercice 1	1
1.1	Question 1	1
1.2	Question 2	2
2	Exercice 2	2
2.1	Question 1	2
2.2	Question 2	2
3	Exercice 3	2
3.1	Question 1	2
3.2	Question 2	2
4	Exercice 4	2
4.1	Question 1	2
4.2	Question 2	2

1 Exercice 1

Code disponible "Exercice1.json"

1.1 Question 1

Nous supprimons l'instruction :

```
intsig MRMOVL                                'instructionSet.get("mrmovl").icode'
```

Nous modifions également l'instruction set pour que RMMOVL ait un icode = 4 et ifun = 0, et MRMOVL un icode = 4 avec un ifun = 1. Le code source Y86 compile bien, le code hexadécimal est bien de 40 et 41 pour RMMOVL (ifun = 0) et RMMOVL (ifun = 1) respectivement.

1.2 Question 2

Nous supprimons l'ensemble des occurrences de MRMOVL. Dans les cas où MRMOVL était dissocié de RMMOVL, nous ajoutons l'instruction :

```
|| icode == RMMOVL && ifun == 1
```

permettant ainsi d'exécuter correctement les instructions demandées.

2 Exercice 2

Code disponible "Exercice2.json"

2.1 Question 1

Nous modifions le instruction set en ajoutant STRGL avec un icode = 14 et un ifun = 0. Nous testons cette nouvelle instruction. **A FAIRE**

2.2 Question 2

Nous ajoutons l'instruction

```
intsig STRGL                                'instructionSet.get("strgl").icode'
```

permettant de donner un icode à l'instruction STRGL.

3 Exercice 3

Code disponible "Exercice3.json"

3.1 Question 1

3.2 Question 2

4 Exercice 4

Code disponible "Exercice4.json"

4.1 Question 1

4.2 Question 2