

Mannuel d'Utilisation

ABDERRAHMANI Ghouzali, CARRE Clement, KHAMARI Nor
el kitem, LOUALIA Kamillia, OUCHENE Rachida,
REKIK Ines, REYNAUD Julien,
TOUAZI Lylia

21 Mai 2019

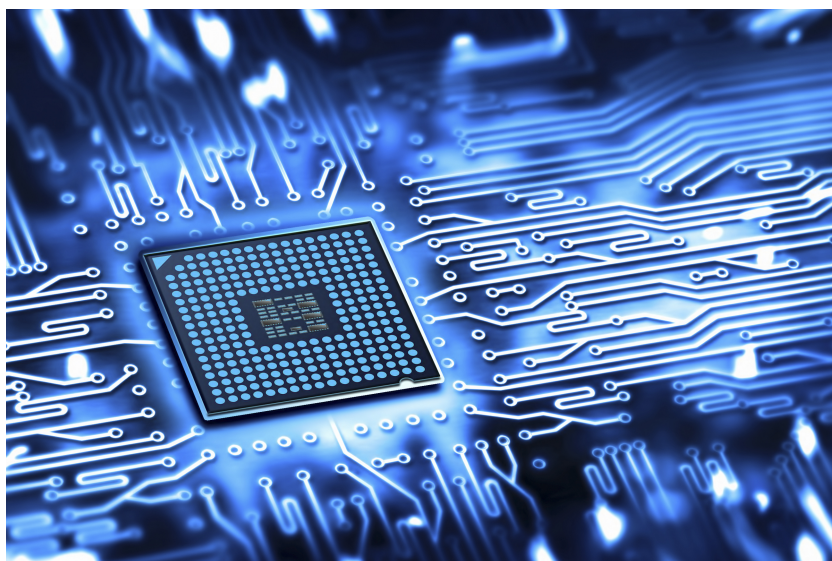


Table des matières

1	INTRODUCTION	3
2	Installation et exécution	3
2.1	Installation	3
2.2	Exécution	3
3	Nomenclature du fichier	3
4	Interface	4
4.1	Interface principale	4
4.2	Deuxième Interface	5
4.3	Troisième Interface	5
4.4	Quatrième Interface	6
4.5	Cinquième Interface	6

1 INTRODUCTION

Notre projet intitulé Émulation d'un processeur RISC consiste à émuler le fonctionnement d'un tel processeur en respectant son jeu d'instruction élémentaire. Cet émulateur disposera d'un programme assembleur donné en entrée et devra pouvoir l'accepter après avoir vérifié sa compatibilité avec le jeu d'instruction du processeur.

2 Installation et exécution

2.1 Installation

Une archive .tar.gz vous est fourni, il faudra l'extraire pour pouvoir utiliser l'application. Aussi, il vous faudra installer la bibliothèque GTK+ 2.0 en tapant les lignes de commande suivantes sur le terminal :

- Pour mettre à jour les paquets, utilisez la commande : `sudo apt-get update`
- Pour installer le paquet de développement, utilisez la commande : `sudo apt-get install libgtk2.0-dev`
- Puis vérifier l'installation, utilisez la commande : `sudo apt-get install libgtk2.0-0`

2.2 Exécution

Pour exécuter l'application, il vous faudra décompresser le fichier .zip, se positionner dans le répertoire de l'application /RISC. Ensuite, ouvrir un terminal , taper la commande `make` pour compiler le programme.

3 Nomenclature du fichier

Les fichiers acceptés par l'application doivent être des fichiers texte composés de l'adresse des variables au début du fichier précédé d'un '@' suivie d'une ligne vide, puis d'instructions selon le modèle du jeu d'instructions donné.

Exemple :

- @a = 1000 (de cette façon on représente les adresses)
- @y = 1100
- @c = 1200
- (ligne vide)
- LW R1,@a[R0] (On représente les instructions de cette façon)
- LW R3,@b[R0]
- ADD R5,R1,R0
- LW R5,@c[R0]
- SUB R5,R5,R1
- SW @c[R2],R5
- ADD R2,R2,R1
- SLT R4,R2,R3
- BNEZ R4,-8

4 Interface

4.1 Interface principale

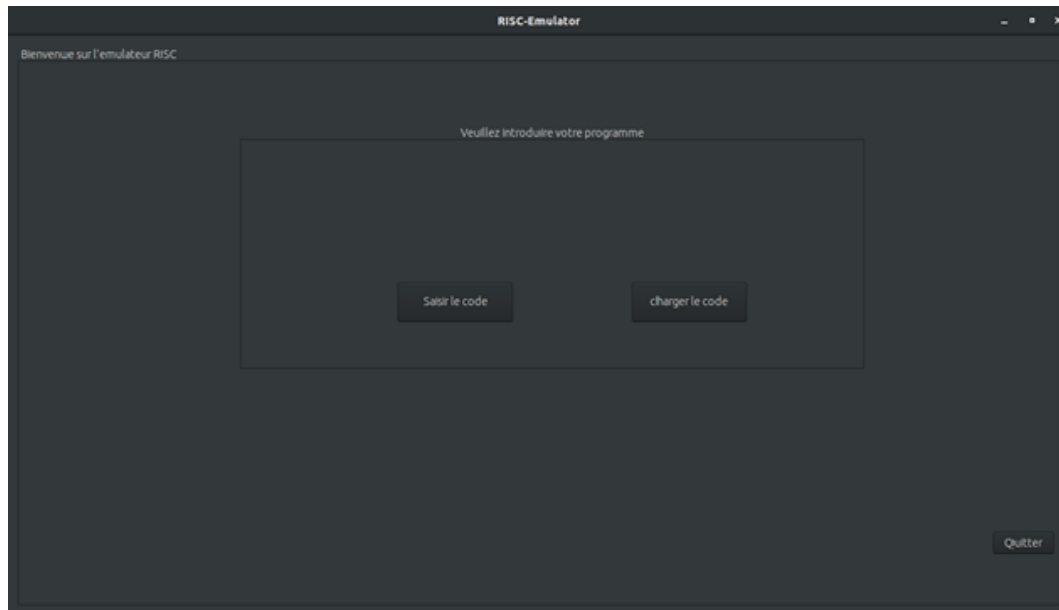


FIGURE 1 – Interface principale

Le programme démarre avec cette fenêtre : elle affiche deux boutons principaux au milieu pour donner le choix à l'utilisateur de saisir le code ou bien le charger dans un répertoire, et un bouton quitter en bas pour quitter le programme. Pour ouvrir ou charger un fichier contenant le code : le programme est fourni avec 2 codes de test : `code1.txt` et `code2.txt` qui seront fournis dans le répertoire `/code`.

4.2 Deuxième Interface

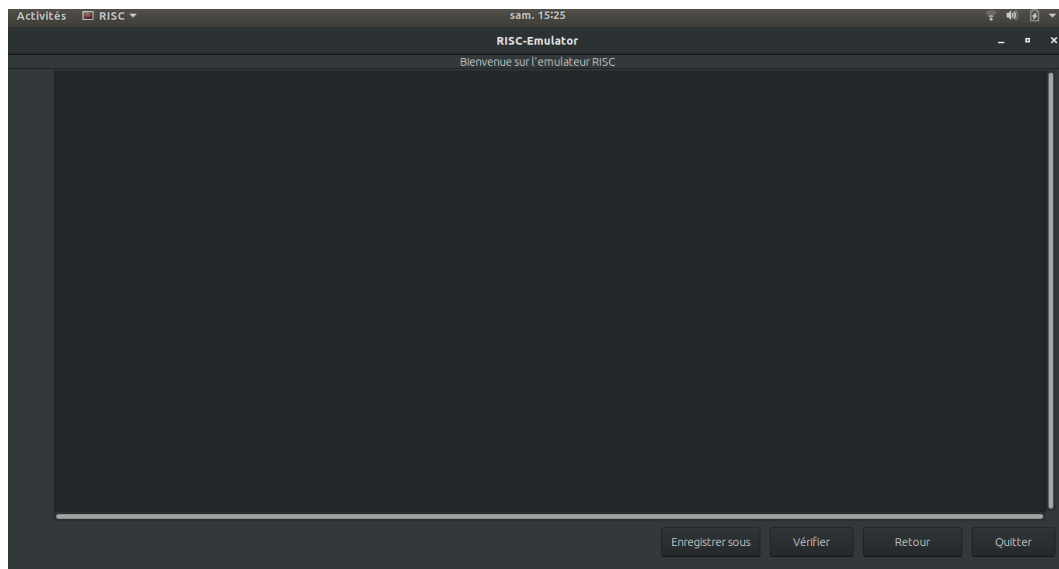


FIGURE 2 – Saisie du Code

Cette fenêtre affiche l'éditeur de texte pour permettre la saisie du code, et quatre boutons en bas, un pour enregistrer le code saisi, un pour lancer l'analyse, un pour revenir à la fenêtre principale, et le dernier pour quitter le programme.

4.3 Troisième Interface

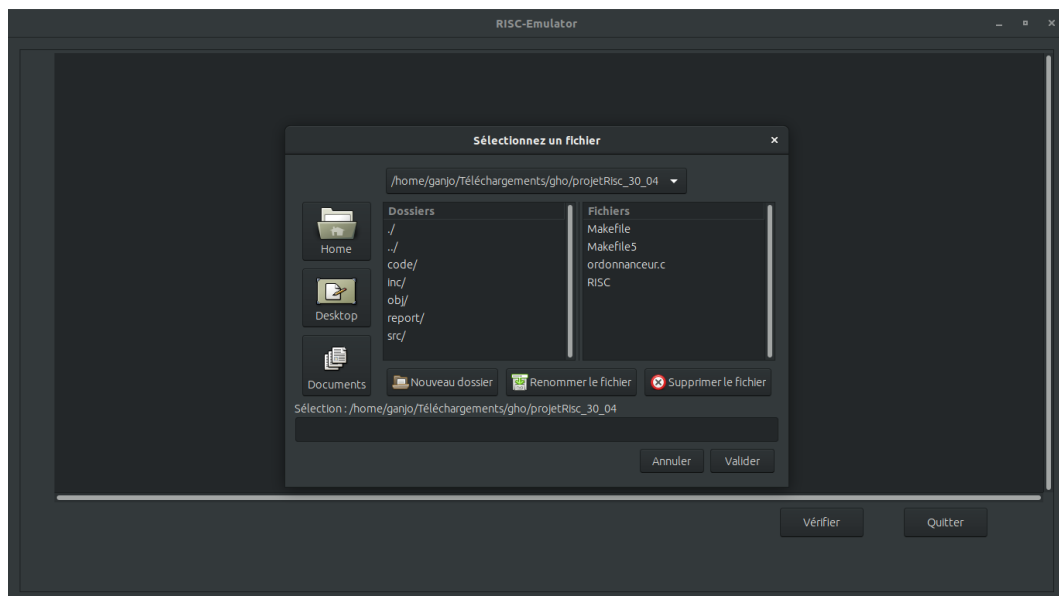


FIGURE 3 – Chargement du Code

La fenêtre de chargement du code, une fenêtre dialogue pour sélectionner le fichier texte contenant le programme, et afficher ce dernier dans l'éditeur de texte.

4.4 Quatrième Interface

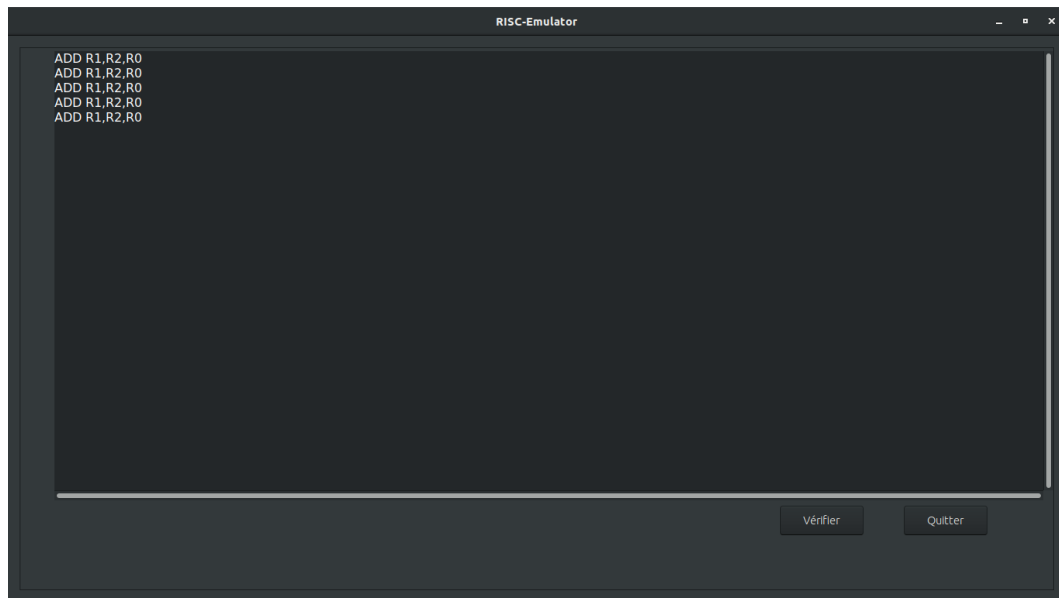


FIGURE 4 – Code Chargé

Une fois le fichier texte est choisi, son contenu est affiché sur l'éditeur texte, puis quatre buttons en bas, un pour l'enregistrement en cas de modification du code, un pour lancer l'analyse du code, un pour revenir et le dernier pour quitter le programme

4.5 Cinquième Interface

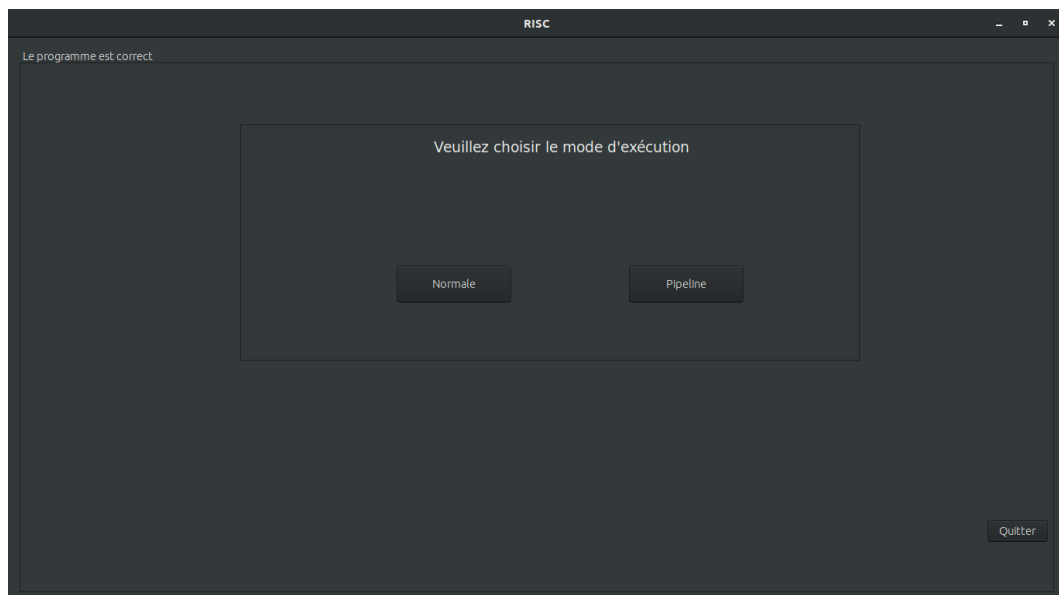


FIGURE 5 – Choix d'Exécution

Cette fenêtre offre deux choix à l'utilisateur grâce au deux buttons du milieu , afin de choisir le mode d'exécution , et le buttons en bas pour quitter le programme , et enfin un petit message

en haut pour préciser que le programme après l'analyse est correct. Pour exécuter le mode d'émulation désiré, il suffit de cliquer sur un des deux boutons : Le bouton "Normale" exécute le mode séquentiel sur le code. Le bouton "Pipeline" exécute le mode pipeline sur le code.