Projet du module : A.O

Simulateur Motorola 6809
Microprocesseur

Présenté par :

BILAL ZINEDDINE / ACHRAF SEHLAOUI / YAHIA RIADI / YASMINE HAMDANI
NIZAR SENBATI

Encadre par : Mr. BENALLA Hicham

PLAN:

A. Microprocesseur Motorola 6809

B. les techniques de simulation

C. structure - hiéararchie du projet

D. les interfaces

E. Démonstration - Pratique

Microprocesseur Motorola 6809

Introduit en 1978, le Motorola 6809 est un microprocesseur 8 bits qui s'est distingué dans l'histoire de l'informatique.

Motorola 6809

le Motorola 6800, il a apporté des améliorations notables en termes de performance et de capacités. Parmi ses caractéristiques marquantes, on peut citer :

1. **Architecture avancée :** Contrairement à de nombreux autres processeurs 8 bits de l'époque, le 6809 disposait d'un jeu d'instructions plus riche et d'une structure de registres plus complexe, offrant ainsi une plus grande flexibilité pour la programmation.
2. **Modes d'adressage** : Le 6809 a introduit un large éventail de modes d'adressage, ce qui était plutôt inhabituel pour les processeurs de sa catégorie à cette époque.
3. **Performance accrue** : Il était considéré comme l'un des microprocesseurs 8 bits les plus puissants en termes de traitement des instructions complexes.

Impact dans l'Industrie :

1. **Compréhension Historique** : Un simulateur permet d'explorer et de comprendre l'architecture et la programmation des premiers microprocesseurs, essentiels à l'histoire de l'informatique.
2. **Développement de Logiciels Rétro** : Il offre une plateforme pour développer et tester des logiciels conçus pour le 6809, sans nécessiter le matériel physique, qui est devenu rare et précieux.
3. **Outil Éducatif** : Les simulateurs servent à enseigner les concepts de base de l'architecture des microprocesseurs et de la programmation bas-niveau, compétences toujours pertinentes dans l'enseignement de l'informatique.
4. **Préservation du Patrimoine Numérique** : Ils jouent un rôle crucial dans la préservation des logiciels classiques, permettant à ces programmes de continuer à fonctionner sur des systèmes modernes.

les techniques de simulation

Langage de Programmation : Java

Java est célèbre pour sa devise Écrire une fois, exécuter partout.

Java dispose d'un vaste écosystème d'outils et de bibliothèques qui facilitent le développement de simulateurs complexes.

Swing est une bibliothèque GUI puissante pour Java, permettant de créer des interfaces utilisateur riches et réactives.

Eclipse est un environnement de développement intégré (IDE) populaire qui fournit des fonctionnalités avancées pour faciliter le processus de développement logiciel.

structure - hiérarchie du projet

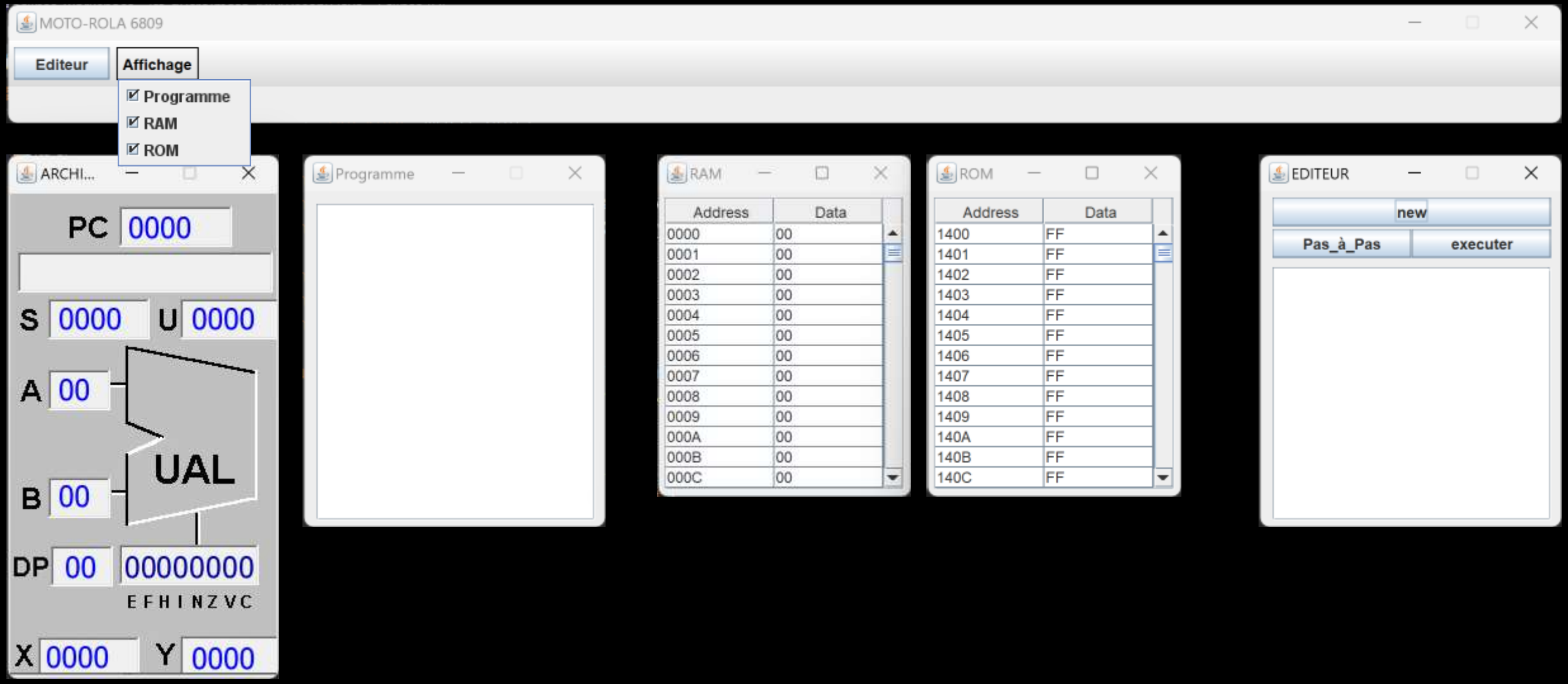
1. **ArchitecteInterne.java** : Ce fichier est au cœur de l'interface utilisateur du simulateur. Il gère l'affichage et la mise à jour des informations critiques liées à l'architecture interne du simulateur. En utilisant les éléments de l'interface graphique Java Swing, Architecte Interne sert de tableau de bord interactif, offrant aux utilisateurs une vue d'ensemble des processus en cours dans le simulateur.

2. **Editeur.java** : Ce fichier représente l'éditeur de code intégré au simulateur. Il fournit une interface pour écrire et exécuter des scripts ou des instructions qui seront traitées par le simulateur. L'éditeur est équipé de fonctionnalités telles que la mise en évidence de syntaxe et l'exécution pas à pas, facilitant ainsi pour les utilisateurs la création et le débogage de leurs programmes.

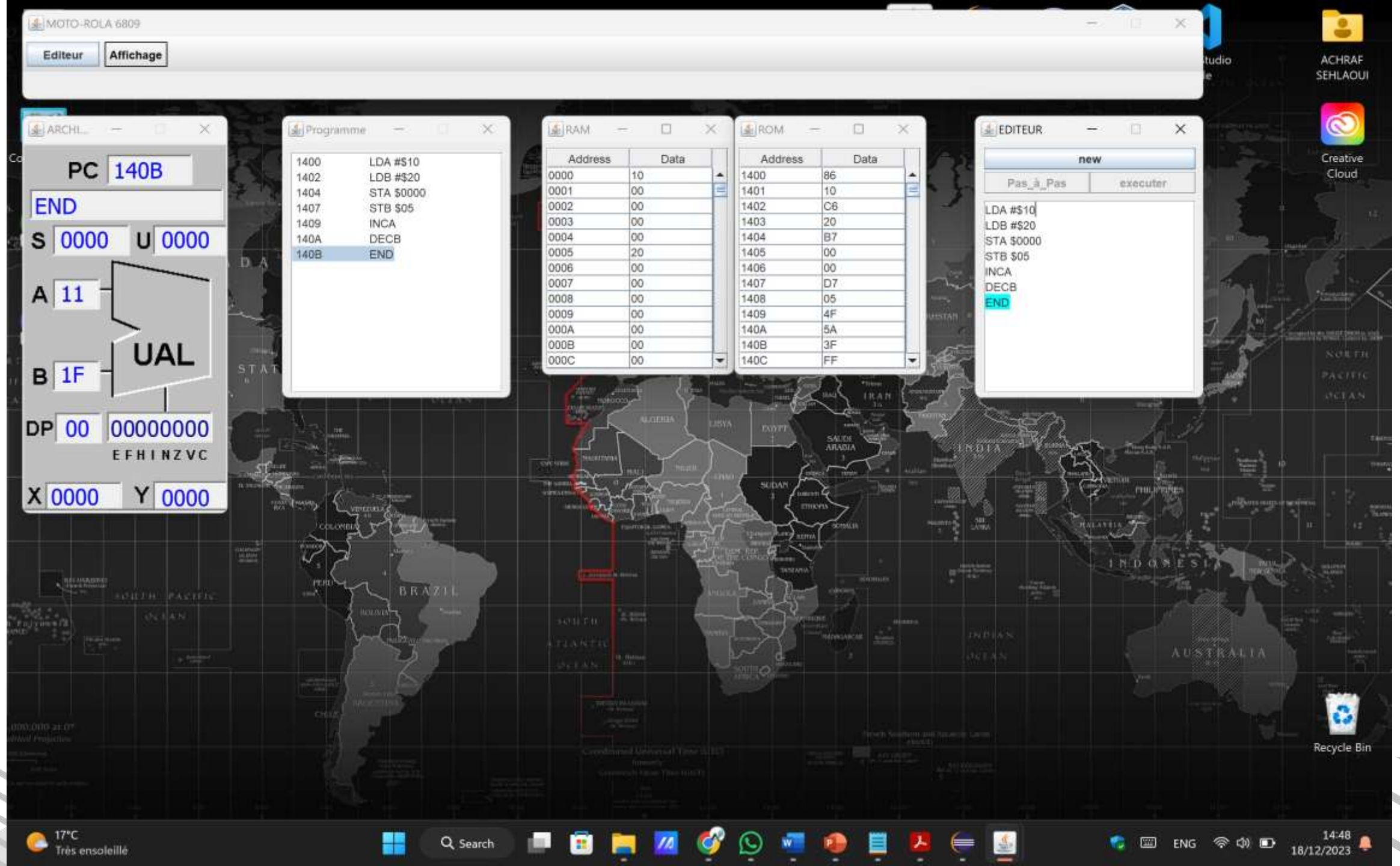
3. **Menu.java** : Ce fichier gère la barre de menu du simulateur, offrant un accès rapide et intuitif aux différentes fonctionnalités du logiciel. Il agit comme une interface centrale pour naviguer entre les différents composants du simulateur, tels que l'éditeur de code, les vues de la RAM et de la ROM, et d'autres outils utiles.

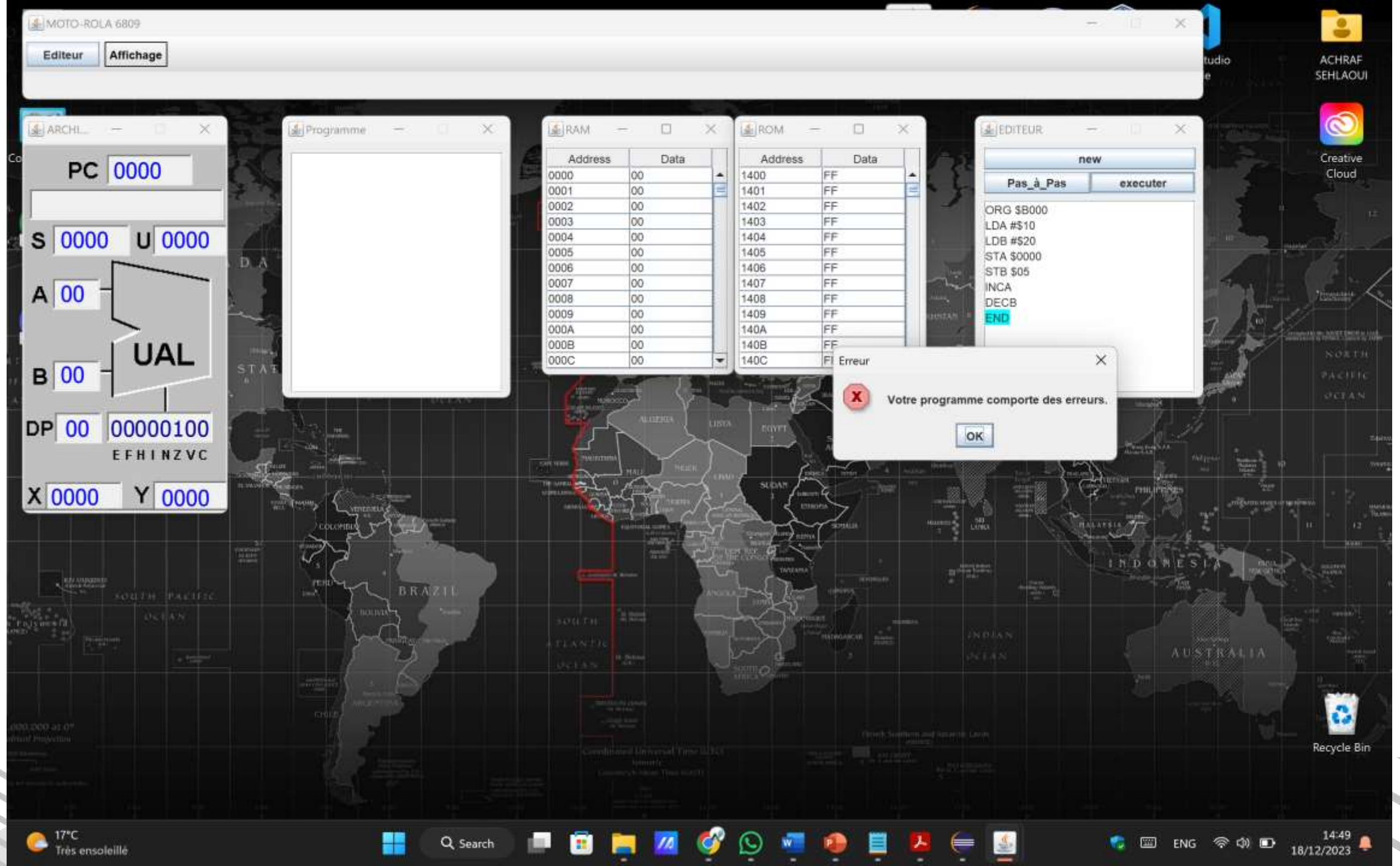
4. **RAM.java et ROM.java :** Ces fichiers sont responsables de simuler les fonctionnalités de la mémoire vive (RAM) et de la mémoire morte (ROM) du système informatique. Ils offrent une interface pour visualiser et manipuler les données stockées dans ces composantes mémoire, permettant aux utilisateurs de comprendre et d'analyser comment les données sont traitées et conservées dans le simulateur.

les interfaces


MOTO-ROLA 6809
Editeur
Affichage
Programme
RAM
ROM
ARCHI...
PC 0000
S 0000
U 0000
A 00
B 00
UAL
DP 00
00000000
E F H I N Z V C
X 0000
Y 0000
Programme
RAM
Address
Data
0000 00
0001 00
0002 00
0003 00
0004 00
0005 00
0006 00
0007 00
0008 00
0009 00
000A 00
000B 00
000C 00
ROM
Address
Data
1400 FF
1401 FF
1402 FF
1403 FF
1404 FF
1405 FF
1406 FF
1407 FF
1408 FF
1409 FF
140A FF
140B FF
140C FF
EDITEUR
new
Pas_à_Pas
executer

Démonstration - Pratique


MOTO-ROLA 6809
Editeur
Affichage
ARCHI...
PC 140B
END
S 0000 U 0000
A 11
B 1F
UAL
DP 00 00000000
EFHINZVC
X 0000 Y 0000
Programme
1400 LDA #$10
1402 LDB #$20
1404 STA $0000
1407 STB $05
1409 INCA
140A DECB
140B END
RAM
Address Data
0000 10
0001 00
0002 00
0003 00
0004 00
0005 20
0006 00
0007 00
0008 00
0009 00
000A 00
000B 00
000C 00
ROM
Address Data
1400 86
1401 10
1402 C6
1403 20
1404 B7
1405 00
1406 00
1407 D7
1408 05
1409 4F
140A 5A
140B 3F
140C FF
EDITEUR
new
Pas_à_Pas
executer
LDA #$10
LDB #$20
STA $0000
STB $05
INCA
DECB
END
ACHRAF SEHLAOUI
Creative Cloud
Recycle Bin
17°C
Très ensoleillé
Search
ENG
14:48
18/12/2023


MOTO-ROLA 6809
Editeur
Affichage
ARCHI...
PC 0000
S 0000
U 0000
A 00
B 00
UAL
DP 00
00000100
EFHINZVC
X 0000
Y 0000
Programme
RAM
Address
Data
0000 00
0001 00
0002 00
0003 00
0004 00
0005 00
0006 00
0007 00
0008 00
0009 00
000A 00
000B 00
000C 00
ROM
Address
Data
1400 FF
1401 FF
1402 FF
1403 FF
1404 FF
1405 FF
1406 FF
1407 FF
1408 FF
1409 FF
140A FF
140B FF
140C FF
EDITEUR
new
Pas_à_Pas
executer
ORG $B000
LDA #$10
LDB #$20
STA $0000
STB $05
INCA
DECB
END
Erreur
Votre programme comporte des erreurs.
OK
ACHRAF SEHLAOUI
Creative Cloud
Recycle Bin
17°C
Très ensoleillé
Search
ENG
14:49
18/12/2023

MERCI
POUR VOTRE ATTENTION