

Projet PROG5 2014-2015
Simulateur ARM
Rapport de Projet

SOULIER Clément
SENECLAUZE Pierre
DUPLAN Maxime
BERNE Corentin

January 14, 2015

1 Structure du code

Nous avons développé l'ensemble du projet dans le squelette fournis, et nous n'avons rien modifié dedans.

2 Liste des fonctionnalités

Description	Fichier	Etat
Accès mémoire	memory.c	Réalisée et testée
Gestion des instructions	arm_instruction.c	Réalisée et testée
Traitement des données	arm_data_processing.c	Réalisée et testée
Accès à la mémoire	arm_load_store.c	Réalisée et testée
Rupture de séquence	arm_branch_other.c	Réalisée et testée
Gestion des interruptions et des exceptions	arm_exception.c	Réalisée et testée

3 Tests effectués

Deux fichiers de tests ont été créés en plus de ceux fournis. Ces deux fichiers permettent de traiter l'ensemble des instructions développées au cours du projet, et les résultats sont concluants. Les fichiers de tests fournis avec le sujet fonctionnent également sans problème. Concernant le test des interruptions et des exceptions, nous avons modifié l'exemple 2 fournit.

4 Journal de la progression

Durant la majorité du développement, nous avons travaillé sur des parties différentes. Cependant, nous avons fait état de l'avancement de nos parties respectives chaque jour.

4.1 Jour 1: Lundi 5 janvier

Tout le groupe :

- Rencontre avec les autres membres du groupe.
- Prise de connaissance du sujet
- Mise en place d'un Github.
- Réalisation des fonctions d'accès à la mémoire.
- Compréhension et début de la réalisation du chargement des instructions.

4.2 Jour 2: Mardi 6 janvier

Tout le groupe :

- Fin de la réalisation générale du chargement des instructions.
- Répartition des différentes parties entre les membres du groupe :
 - Maxime : data_processing.
 - Pierre : branch_other.
 - Clément & Corentin : load_store.

Maxime :

- Création du squelette de switch des 16 opérations.
- Début du traitement des data_processing_shift.

Pierre :

- Début et fin de branch_other.

Clément & Corentin :

- Réalisation de load_store

4.3 Jour 3: Mercredi 7 janvier

Maxime :

- Fin du traitement des décalages pour les data_processing_shift.
- Mise en place des algorithmes basiques des instructions arithmétiques.

Pierre :

- Amélioration et correction du traitement des instructions.

Clément & Corentin :

- Réalisation de load_store multiple

4.4 Jour 4: Jeudi 8 janvier

Maxime :

- Fin du traitement des data_processing_shift.
- Début et fin des data_processing_immediat.

Pierre :

- Réalisation de MRS pour miscellaneous.
- Corrections sur les branchements.

Clément & Corentin :

- Correction des bugs sur les macros et du premier octet des instructions à 1.
- Réalisation de LDRH et STRH.
- Tests de LDR, STR, LDRB et STRB

4.5 Jour 5: Vendredi 9 janvier**Maxime :**

- Correction d'un bug de signe.
- Fin du fichier data_processing.

Pierre :

- Début de la mise en place des interruptions.
- Amélioration des traces.

Clément & Corentin :

- Modifications et corrections diverses sur load_store.

4.6 Jour 6: Lundi 12 janvier**Tout le groupe :**

- Division des tâches restantes :
 - Pierre & Maxime : interruptions et exceptions.
 - Clément & Corentin : jeux de test.

Pierre & Maxime :

- Début du traitement des interruptions.

Clément & Corentin :

- Tests avec les jeux d'essais fournis.

4.7 Jour 7: Mardi 13 janvier

Pierre & Maxime :

- Traitement des interruptions. De nombreux bugs trouvés et résolus pour certains.

Clément & Corentin :

- Création et utilisation d'un test de tri par sélection.

4.8 Jour 8: Mercredi 14 janvier

Pierre & Maxime :

- Fin du traitement des interruptions.
- Ecriture du rapport.

Clément & Corentin :

- Création et utilisation d'un second jeu de test.
- Résolution de bugs mineurs.

Tout le groupe

- Préparation de l'oral

A ce stade, l'ensemble du sujet est traité.