

UNIVERSITÉ DE MONTRÉAL

IFT1215 - INTRODUCTION AUX SYSTEMES INFORMATIQUES

Exercice 9

18 mars 2019

6.12

Le programme donné effectue plutôt le calcul $IN_3 - (IN_1 + IN_2)$. Pour corriger le problème, puisque l'instruction SUB XY effectue $A - XY \rightarrow A$, il est nécessaire d'enregistrer notre troisième entrée dans une variable temporaire. De cette façon, on peut charger la somme dans l'accumulateur afin d'y soustraire IN_3 .

7.2

a)

On rappelle que le cycle fetch-execute de l'instruction LDA XY est

PC \rightarrow MAR
MDR \rightarrow IR
IR[adr.] \rightarrow MAR *on cherche le contenu de l'adresse XY*
MDR \rightarrow A *et on le charge dans l'accumulateur*
PC + 1 \rightarrow PC

PC : contient la prochaine ligne **21**

IR : contient la dernière instruction lue (LDA 50) **550**

MAR : contient la dernière adresse accédée **50**

MDR : contient le contenu du MAR en tout temps **724**

b)

PC	IR	MAR	MDR	A	
21	450	50	724	724	contenus précédent
21	450	21	151	724	PC \rightarrow MAR
21	151	21	151	724	MDR \rightarrow IR
21	151	51	006	724	IR[adr] \rightarrow MAR
21	151	51	006	730	A + MDR \rightarrow A

7.7

a)

faire un décalage d'un nombre signé, dans un registre, de deux bits à gauche a pour effet de multiplier par 2^2 le contenu du registre. Cependant, il est fort probable que cela cause un overflow, puisque la taille d'un registre est fixe, rendant le résultat incorrect. Un décalage vers la droite produit la division entière par 2 et donnera toujours un résultat valide.

b)

De la même façon, le décalage à gauche ne sera pas toujours correct mais cette fois si, le décalage vers la droite cause problème puisque l'insertion de 0 à droite transforme un nombre négatif en nombre positif.

c)

Ce changement permet maintenant d'effectuer la division entiere par 2 de facon valide pour tout les nombres. Il est claire que :

- l'ajout de zero a l'avant d'un nombre deja positif ne causera pas de changement de signes.
- l'ajout de un a l'avant d'un nombre negatif, ne change pas le signe non plus. Vous pouvez verifier que la nature du complement a 2 rend le resultat valide.

7.9

PC \rightarrow MAR
MDR \rightarrow IR
99 - A \rightarrow A
A + 1 \rightarrow A
PC + 1 \rightarrow PC

PC \rightarrow MAR
MDR \rightarrow IR
MDR \rightarrow A
A - MDR \rightarrow A
PC + 1 \rightarrow PC

7.12

PC \rightarrow MAR (Premier fetch)
MDR \rightarrow IR

IR \rightarrow IAR (On sauvegarde la premiere adresse)
PC + 1 \rightarrow PC

PC \rightarrow MAR (deuxieme fetch)
MDR \rightarrow IR
IR[adr] \rightarrow MAR
MDR \rightarrow A (On sauvegarde le contenu de la seconde adresse dans A)

IAR[adr] \rightarrow MAR (On accede au contenu de la premiere adresse)
MDR \rightarrow IAR (on sauvegarde le contenu dans IAR. La seconde adresse est toujours dans MAR)
A \rightarrow MDR (on remplace le contenu de la premiere adresse)

IR[adr] \rightarrow MAR
IAR \rightarrow MDR (on remplace le contenu de la seconde adresse)
PC + 1 \rightarrow PC
