

École Polytechnique de Montréal
Département de génie informatique et génie logiciel

INF1600
Architecture des micro-ordinateurs

TP2
Architecture à deux bus et introduction à l'assembleur IA-32

Soumis par :
William Harvey (1851388) et
Mathieu Bélanger (1850591)
Section 4

1 mars 2017



Exercice 1 : Architecture avec micro-codes

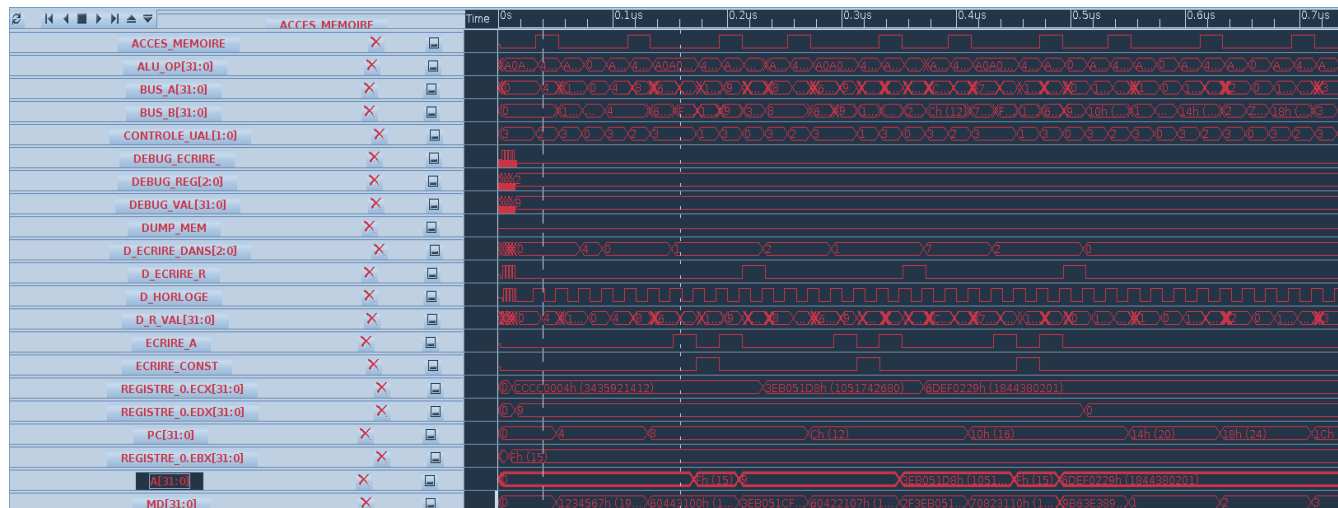
1. Recherche d'instruction :

RTN concret	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	hexa
MA \leftarrow PC;	0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0x3060
MD \leftarrow M[MA] : PC \leftarrow PC+4;	0	1	1	0	1	1	0	0	1	1	0	0	0	0	0	0	0x6CC 0
IR \leftarrow MD;	1	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0x8260

2. Exécution d'une instruction générique :

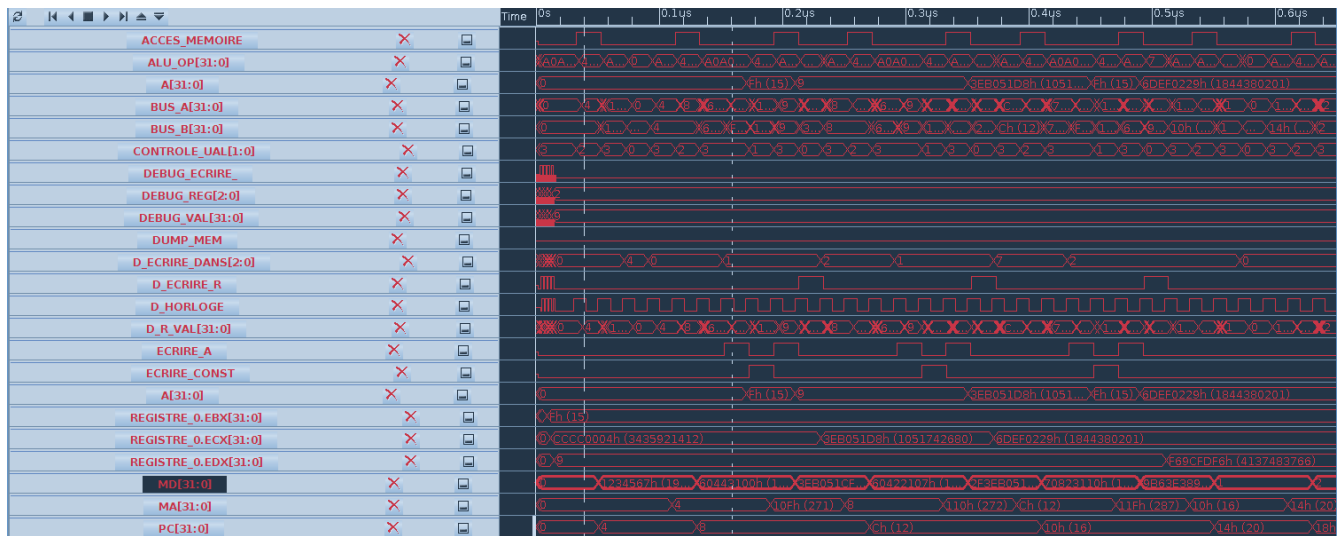
RTN concret	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	hexa
A \leftarrow R[rc];	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	0	0x006E
MA \leftarrow A + IR<11..0>;	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0x1021
MD \leftarrow M[MA] : A \leftarrow R[rb];	0	0	0	0	1	1	0	0	1	1	1	0	1	0	1	0	0x0CE A
R[ra] \leftarrow MD oper A	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0x8210

3. Simulation :



Le microprogramme s'exécute correctement, car on peut voir que lorsque le PC contient l'adresse 0xC (12), le registre %ecx contient 0x6DEF0229. En effet, il s'agit de la bonne exécution de l'opération oper, qui est une addition entre A et MD. Pour preuve, l'addition de 0x2F3EB051 (stockée dans MD) et de 0x3EB051D8 (stockée dans A) est bel et bien 0x6DEF0229. %ecx = MD + A.

4. L'opération NAND :



La valeur stockée dans le registre %edx est 0xF69CFDF6 ce qui représente le résultat exact de l'opération NAND entre la valeur 0x6DEF0229, qui est stockée dans A, et la valeur 0x9B63E389, qui est stockée dans MD. Ainsi, il est possible de savoir que cette opération s'est bien déroulé.

5. Compréhension

a)

On sait que le registre d'instruction IR se compose ainsi :

- IR<31..27> : contient l'opcode
- IR<26..22> : contient le registre ra
- IR<21..17> : contient le registre rb
- IR<16..12> : contient le registre rc
- IR<11..00> : sont des bits inutilisés

Alors, on peut modifier sans problème les 12 premiers bits du registres, car ils sont inutilisés. Toutefois, il est important de conserver la valeur des 20 MSB, car ce sont eux qui donne le sens à l'instruction. Ainsi, on pourrait écrire 0x5555000, car il s'agirait d'une instruction totalement équivalente, puisqu'on ne modifie que les 3 derniers mots.

b)

L'avantage d'une architecture à deux bus est quelle permet d'exécuter plus d'une opération à la fois. Par exemple, nous nous sommes servis de cet avantage à la troisième instruction de la question 2, lorsque nous avons à la fois chargé MD de M[MA] et chargé A de R[rb]. Nous avons travaillé sur les deux bus de cette architecture.

c)

Les instructions de cette architecture peuvent être plus flexibles que celles du processeur étudié à l'exercice 4 du TP1, car on utilise l'architecture de Von Neumann qui nous permet d'accéder directement à la mémoire. De plus, il y a beaucoup plus de registres sur cette architecture que sur celle du TP1 ce qui nous permet d'avoir une plus grande flexibilité au niveau des opérations effectuées.

Exercice 2 : Assembleur avec processeur à pile

*Voir exo2/tp2_2.c
/tp2_2.s

Exercice 3 : Conditions et branchements

*Voir exo3 /tp2_3.c
/tp2_3.s