

TD d'architecture des ordinateurs III

Micro-programmation

Exercice 1 : Modes d'adressage

On considère l'organisation à bus unique des chemins de données dans l'unité centrale (UC) de la Figure 1.

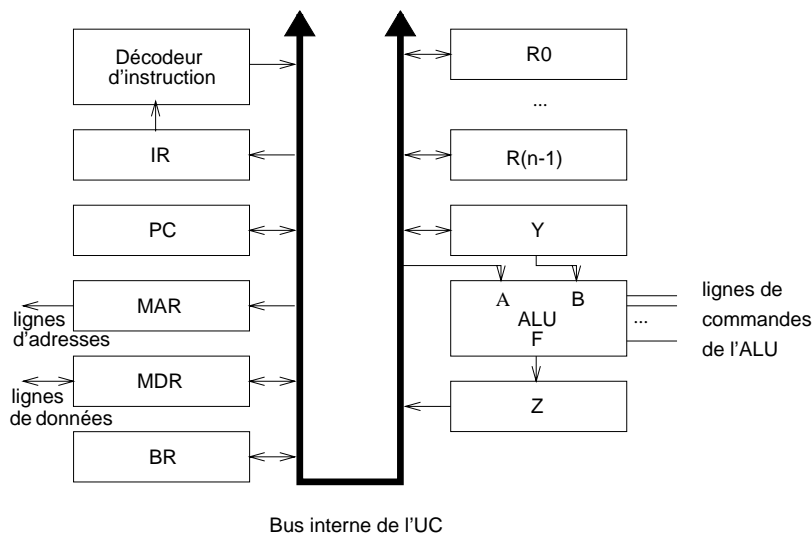


FIG. 1 – Organisation interne de l'unité centrale

On rappelle la signification des différentes abréviations :

TAB. 1 – Nom des registres

Abréviation	Termes anglais	Termes français
IR	Instruction Register	Registre d'Instruction
MDR	Memory Data Register	Registre de Données Mémoire
MAR	Memory Address Register	Registre d'Adresses Mémoire
PC	Program Counter	Compteur Ordinal
SP	Stack Pointer	Pointeur de Pile
ALU	Arithmetical and Logical Unit	Unité Arithmétique et Logique

- Supposez que les temps de propagation le long du bus et à travers l'ALU soient respectivement 20 et 100 ns. Le temps d'établissement d'un registre est de 10 ns. Quel est le temps minimum pour effectuer chacune des opérations suivantes (donner la séquence de micro-instructions en se basant sur la table 2 et les chronogrammes des signaux à envoyer aux éléments utilisés) :
 - Transfert d'une donnée d'un registre à un autre.
 - Incrémenter le compteur de programme.
- On suppose qu'une instruction est composée de deux mots (le premier mot indiquant l'opération et le mode d'adressage et le second contenant un nombre NB) et qu'on dispose des micro-instructions de la table 2. Écrire la séquence des étapes de contrôle nécessaires pour chacune des trois instructions suivantes en utilisant la structure de bus de la figure 1 :

- Additionner le nombre NB au registre R1.
- Additionner le contenu de la case mémoire NB au registre R1.
- Additionner le contenu de la case mémoire dont l'adresse est dans la case mémoire NB au registre R1.

TAB. 2 – Micro-instructions disponibles

Reg out	Sort le contenu du registre Reg sur le bus
Reg in	Met la valeur du bus dans le registre Reg
Lecture	Effectue une lecture de la mémoire à l'adresse MAR et met le résultat dans MDR
Écriture	Effectue une écriture dans la mémoire à l'adresse MAR de la valeur contenue dans MDR
Attente	Permet d'attendre la fin d'une lecture ou d'une écriture
ADD	Additionne les entrées A et B de l'ALU, met le résultat sur la sortie de l'ALU ($F=A+B$)
INCRA	Incréméte l'entrée A de l'ALU ($F=A+1$)
DECRA	Décréméte l'entrée A de l'ALU ($F=A-1$)
INCRB	Incréméte l'entrée B de l'ALU ($F=B+1$)
DECRB	Décréméte l'entrée B de l'ALU ($F=B-1$)
REPA	Reporte l'entrée A de l'ALU sur sa sortie ($F=A$)
REPB	Reporte son entrée B de l'ALU sur sa sortie ($F=B$)

Exercice 2 : JSR et RTS

On considère une organisation à deux bus des chemins de données dans l'unité centrale de la Figure 2.

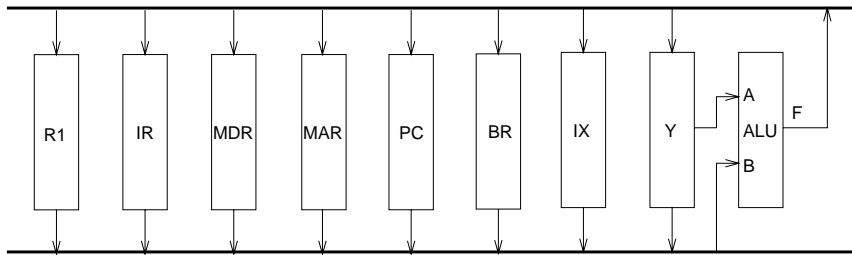


FIG. 2 – Organisation interne de l'unité centrale

Les abréviations et les micro-instructions disponibles sont les mêmes que celles de l'exercice 1.

1. Écrivez la séquence des étapes de contrôle nécessaires à l'obtention et l'exécution de l'instruction JSR (Jump to Sub-Routine). Dans cette machine, l'instruction JSR occupe deux mots. Le premier mot est le code opération et le second mot contient l'adresse de départ du sous-programme. L'adresse de retour est sauvegardée dans une pile mémoire. Un pointeur de pile (SP) est utilisé pour désigner en permanence le sommet de cette pile. On fera en sorte que $[SP]+1$ pointe toujours sur l'élément du sommet de la pile (et $[SP]+2$ sur le second élément de la pile, etc.).
2. Écrivez aussi l'instruction RTS (ReTurn from Sub-routine).