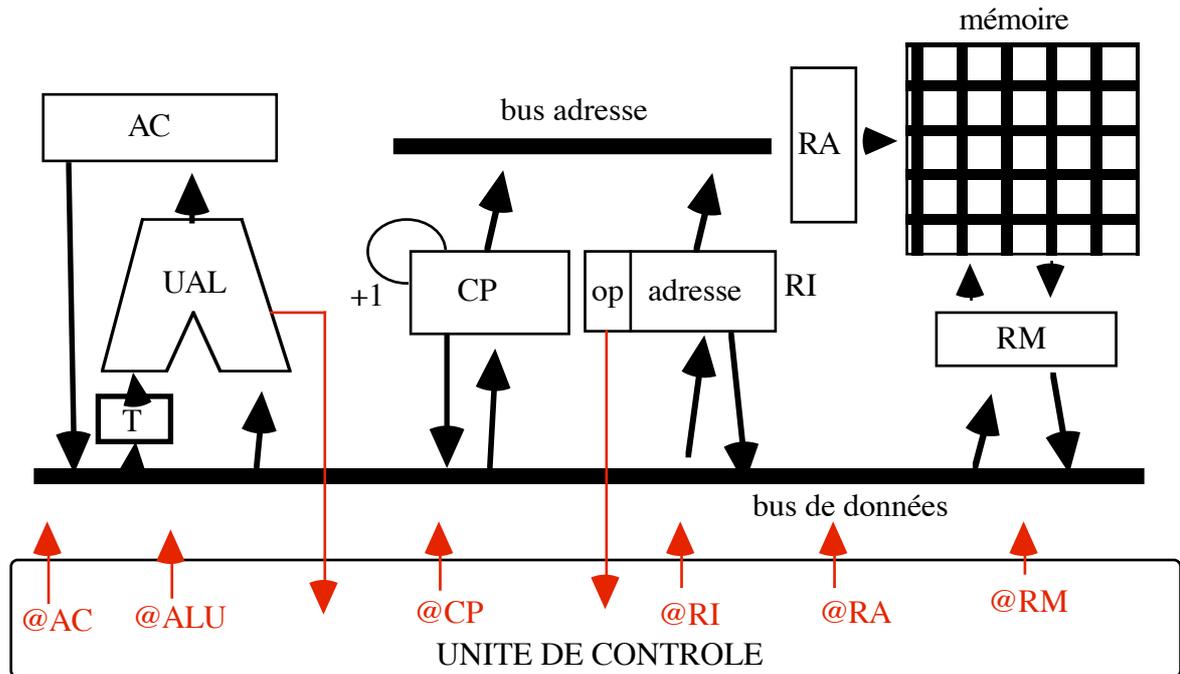


# VARI - Exercice Dirigé

## Architecture des ordinateurs

### Exercice 1 - Un processeur minimal

Reprenons l'architecture d'un processeur minimal vue en cours.



*Rappeler le rôle des différents organes dans le schéma ci-dessus*

*Que manque t'il surtout dans cette architecture minimale ?*

On suppose que tous les registres sont de 64 bits, ainsi que les mots mémoires.

*Quelle est la taille de l'espace mémoire adressable ?*

*Quels sont les nombres manipulables par l'UAL ?*

*Comment peuvent être codées les instructions de ce processeur ? Quel est l'inconvénient de cette approche et comment le résoudre ?*

### Exercice 2 - Exemple de programme

La machine ci-dessus dispose d'instructions de trois types, qui sont toutes codées sur un mot-mémoire de 64 bits.

- des instructions sans opérande :

RAZ : (AC) := 0

INC : (AC) := (AC) + 1

DEC : (AC) := (AC) - 1

HALT : arrêt du programme

- des instructions à 1 opérande :

SET valeur : (AC) := valeur

LOAD case : (AC) := (case)

STORE case : (case) := (AC)

ADD case : (AC) := (AC) + (case)

SUB case : (AC) := (AC) - (case)

EQU case : si (AC) = (case) alors (AC) := 0

INF case : si (AC) < (case) alors (AC) := 0

BRA valeur : si (AC) = 0 alors (CP) := (CP) + valeur

- des instructions d'adressage indirect :

LOADi case : (AC) := ((case))

STOREi case : ((case)) := (AC)

ADDi case : (AC) := (AC) + ((case))

SUBi case : (AC) := (AC) - ((case))

EQUi case : si (AC) = ((case)) alors (AC) := 0

INFi case : si (AC) < ((case)) alors (AC) := 0

BRAi case : si (AC) = 0 alors (CP) := (CP) + (case)

On veut écrire un programme dont la seule fonction est de se dupliquer dans la mémoire, par saut de 1000 cases. On supposera que la position en mémoire de la première instruction est connue. Le programme s'arrête quand toute la mémoire est parcourue.

*Ecrire ce programme en supposant que ses données ne sont pas dupliquées.*

*Que manque-t'il au jeu d'instruction pour que les données soient elles aussi dupliquées ?*

*Nommer un type de programme utilisant ce mécanisme. Quels sont les problèmes posés par ce type de programme ?*

### **Exercice 3 - Chronogramme d'exécution**

*Décrire le séquençement de l'instruction ADD 155*

On indiquera le déroulement des phases :

- de recherche de l'instruction courante ("fetch"),
- de décodage ("decode"),
- d'exécution ("execute").

Chaque étape correspond à une action élémentaire (micro-instruction) de l'unité de contrôle.

*Décrire le séquençement de l'instruction ADDi 155*

*Combien de cycles d'horloge sont maintenant nécessaires ?*

*Quelle valeur doit-on choisir pour le temps de cycle élémentaire ?*