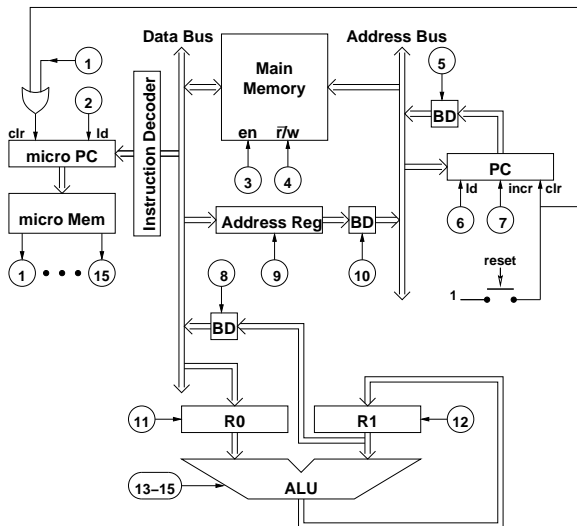


Le premier ordinateur



Zones fonctionnelles

- l'unité arithmétique et logique (l'unité de calcul)
- les registres (zone de stockage des données)
- les séquenceur (zone dirigeant le tout)

Calcul et mémorisation

- la zone de calcul
 - l'UAL effectue les opérations mathématiques
 - l'UAL prend les opérandes des opérations depuis les registres
 - l'UAL renvoie les résultats dans les registres
- la zone de stockage
 - les registres sont la zone de stockage interne du processeur
 - plus rapide qu'un accès en mémoire
 - pour alimenter les registres, le processeur cherche les données en mémoire principale
 - la circulation des données s'effectue via les bus

Séquenceur

- les registres et l'UAL sont régis par des commandes
- le séquenceur récupère les instructions stockées dans la mémoire principale
- ces instructions sont décodées puis envoyées au registre d'instructions
- Il décide alors quels sont les transferts de données nécessaires et les commandes à envoyer.
- Le travail du séquenceur est régi par une horloge (plus l'horloge est rapide, plus le séquenceur et donc le processeur travaillent vite)

Les instructions

- Le processeur exécute les instructions se trouvant en mémoire principale
- ces instructions sont stockées en codes numériques
- langage assembleur : traduction “mnémoniques” des codes numériques
- on programme avec des langages évolués (C, C++, Java, etc.) qui sont assemblés par le compilateur.

Exemples de commandes assembleur

Instructions de chargement et de stockage

LDIMM xx LD xx	Load immediate xx Load xx	$xx \rightarrow R0$ $MEM[xx] \rightarrow R0$	La valeur xx est placée dans le registre R0 La valeur contenue à l'adresse xx est copiée dans le registre R0
ST xx	Store xx	$R1 \rightarrow MEM[xx]$	La valeur du registre R1 est copiée à l'adresse xx

Instructions concernant l'Unité Arithmétique et Logique

COPY SHL	Copy Shift left	$R0 \rightarrow R1$ décalage à gauche($R1$) $\rightarrow R1$	Les bits Z, N, C et V de l'UAL sont mis à jour Le bit de poids fort est copié dans la retenue (C)
SHR	Shift right	décalage à droite($R1$) $\rightarrow R1$	Les bits Z, N et V de l'UAL sont mis à jour Le bit de poids faible est copié dans la retenue(C)
ADD	Add	$R1 + R0 \rightarrow R1$	Les bits Z, N, C et V de l'UAL sont mis à jour
SUB	Sub	$R1 - R0 \rightarrow R1$	Les bits Z, N, C et V de l'UAL sont mis à jour
AND	And	$R1 \text{ et } R0 \rightarrow R1$	Les bits Z, N, C et V de l'UAL sont mis à jour
OR	Or	$R0 \text{ ou } R1 \rightarrow R1$	Les bits Z, N, C et V de l'UAL sont mis à jour
NOT	Not	non $R1 \rightarrow R1$	Les bits Z, N, C et V de l'UAL sont mis à jour

Instructions de saut en mémoire centrale

(Les sauts conditionnels dépendent du résultat de la dernière instruction exécutée par l'UAL)

JAL xx JN xx JZ xx JC xx JV xx	Jump always Jump negative Jump zero Jump carry Jump overflow	aller à l'adresse xx si négatif aller à l'adresse xx si zéro aller à l'adresse xx si retenue aller à l'adresse xx si débordement aller à l'adresse xx	
--	--	---	--