

Architecture des ordinateurs

Architecture de Von Neumann
Unité de commande

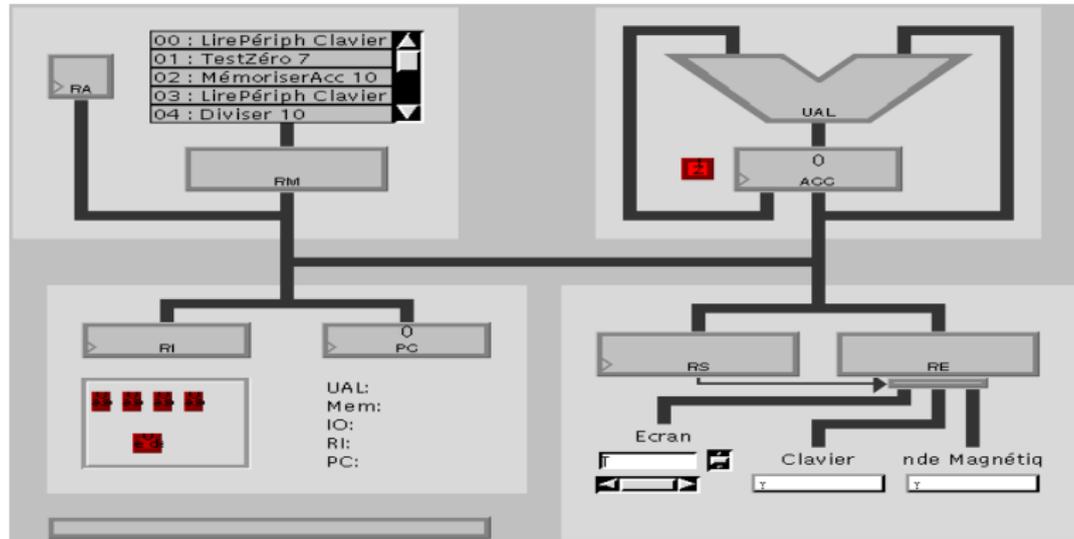
Matthias Colin et Florent Devin

EISTI



Architecture de Von Neumann

Machine de Von Neumann



Machine de Von Neumann

Fonctionnement

- extraction de l'instruction depuis la mémoire
- analyse de l'instruction ;
- recherche, dans la mémoire, les données sur lesquelles porte l'instruction ;
- déclenchement de l'opération adéquate sur l'unité de calcul ou encore l'unité d'entrées/sorties ;
- éventuellement rangement du résultat dans la mémoire.

Unité de commande

Unité de commande

Fonctionnement

- Extraction de l'instruction depuis la mémoire.
- À quelle adresse va-t-elle lire cette instruction ?
- Nécessité de disposer d'un registre particulier qui, en début du cycle d'exécution d'une instruction, contiendra cette adresse.
- *compteur ordinal*, appelé aussi compteur de programme ou encore *Program Counter (PC)*

Unité de commande

Fonctionnement

- Ensuite, analyse de l'instruction.
- Mémorisation nécessaire pour cette analyse
- *registre instruction* (RI)
- Circuit spécial
 - Logique combinatoire
 - Extraction du code opération
 - Extraction des arguments
 - Analyse de certaines parties de l'instruction

Unité de commande

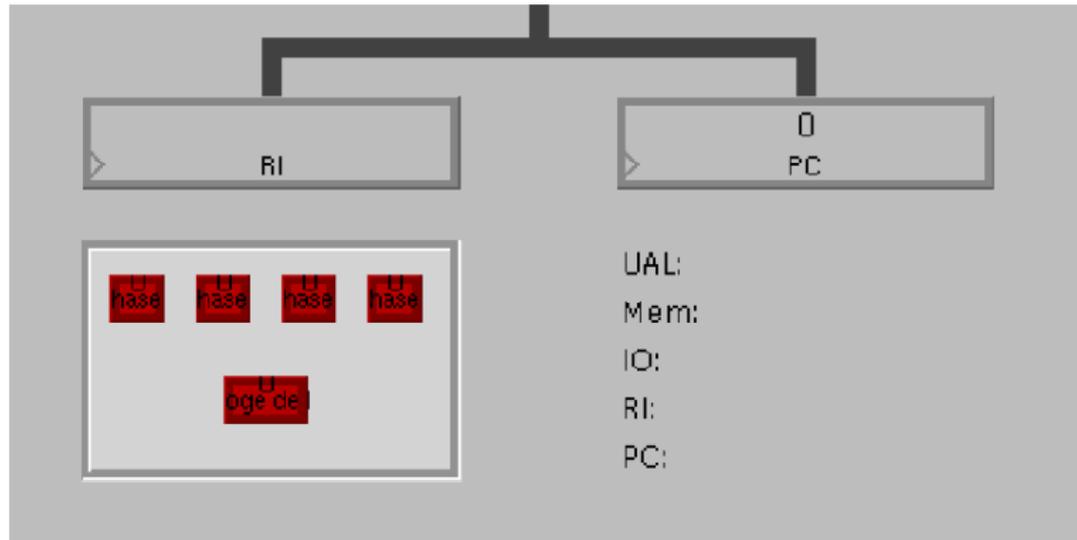


FIG.: Unité de contrôle

Unité de commande

Remarques

- Programme : suite d'instructions
- Nécessité de le charger en mémoire
- Rôle d'un programme particulier appelé le *chargeur*.
 - lecture d'un programme sur support externe
 - Stockage en mémoire ?
- Qui charge le chargeur ?
 - *microchargeur* : circuit câblé

Micro-opérations

Introduction

- Exécution d'un programme : exécution séquentielle d'instructions.
- Une instruction : un cycle d'instruction
- Cycle d'instruction : plusieurs phases (fetch, decode, ...).
- Une phase du cycle d'instruction : une ou plusieurs étapes simples appelées micro-opérations.
- Micro-opérations : opérations fonctionnelles atomiques du processeur.

Micro-opérations

Optimisations

- Regroupages possible pour exécution en parallèle
 - Respecter la dépendance des données
 - Éviter les conflits
- Supposons que nous disposons des registres PC, IR, MAR et MBR.
- Un cycle d'instruction : fetch-decode, indirect, execute, et interrupt.

Micro-opérations

Fetch

- Quatre micro-opérations

$$t_1 \text{ MAR} \leftarrow \text{PC}$$

$$t_2 \text{ MBR} \leftarrow \text{mémoire}$$

$$\text{PC} \leftarrow \text{PC} + T^a$$

$$t_3 \text{ IR} \leftarrow \text{MBR}$$

^aoù T est la taille d'une instruction et les t_i représentent des unités de temps de valeur égale (le compteur ordinal aurait pu être incrémenté lors de t_3).

Micro-opérations

Indirect - chargement

- Si indirect nécessité de refaire la phase

t_1 MAR \leftarrow IR(adresse)^a

t_2 MBR \leftarrow mémoire

t_3 IR(adresse) \leftarrow MBR(adresse)

^aoù IR(adresse) identifie le champs adresse du registre IR.

Micro-opérations

Interrupt

- Après le cycle execute : test effectué
- Séquence peut être
 - t_1 MBR \leftarrow PC
 - t_2 MAR \leftarrow adresse de sauvegarde
PC \leftarrow adresse de routine d'interruption
 - t_3 mémoire \leftarrow MBR

Micro-opérations

Execute

- Imprévisible
- n différents opcodes $\Leftrightarrow n$ différentes séquences de micro-opérations

Exemple

- Instruction ADD R1,X.
 - t_1 MAR \leftarrow IR(adresse)
 - t_2 MBR \leftarrow mémoire
 - t_3 R1 \leftarrow R1 + MBR

Contrôle du processeur

Rôle

- L'unité de commande doit exécuter deux tâches basiques
 - le séquençement des micro-opérations
 - l'exécution des micro-opérations

Contrôle du processeur

Catégories de micro-opérations

- transfert de données entre registres
- transfert de données d'un registre vers une interface externe (e.g., bus)
- transfert de données d'une interface externe (e.g., bus) vers un registre
- opération sur l'ALU en utilisant des registres pour les opérandes sources et le résultat

Contrôle du processeur

Interactions avec l'extérieur : entrées

- l'horloge : une micro-opération s'exécute en un top d'horloge ;
- le registre d'instruction : l'opcode permet de déterminer quelle séquence de micro-opérations doit être exécutée lors d'un cycle exécute ;
- le registre de condition : utilisé pour déterminer le statut du processeur et pour les instructions conditionnelles ;
- des signaux de contrôle du bus de contrôle : par exemple pour détecter une interruption ou recevoir les acquittements des unités externes.

Contrôle du processeur

Interactions avec l'extérieur : sorties

- des signaux de contrôle internes au processeur, de 2 types
 - des signaux déclenchant des transferts entre registre
 - des signaux déclenchant des opérations sur l'ALU
- des signaux de contrôle transportés par le bus de contrôle, de 2 types
 - à destination de la mémoire
 - à destination des modules d'E/S

Contrôle du processeur

Exemple : fetch

- un signal autorisant le chargement de MAR par le contenu de PC
- simultanément
 - un signal plaçant le contenu du MAR sur le bus d'adresse
 - l'activation d'une ligne lecture mémoire du bus de contrôle
 - un signal autorisant le chargement de MBR par le bus de données
 - un signal déclenchant une opération incrémentant le PC (ce qui peut être fait indépendamment de l'ALU si on considère une logique dédié)

Contrôle du processeur

Exemple : fetch

- un signal autorisant le chargement de IR par MBR
- Ensuite, le contenu de IR est analysé pour savoir quel cycle doit suivre (indirect ou execute).
- Les cycles indirect et interrupt fonctionnent de manière similaire.
- Le cycle execute est différent

Contrôle du processeur

Organisation

- L'organisation typique de la CPU comprend un bus interne (ou un ensemble de bus internes) pour la circulation interne des différentes informations (données, instructions, adresses).
- Ce bus relie tous les registres internes et l'ALU.
- Chaque signal de contrôle interne émis par l'unité de commande (exceptés les signaux déclenchant des opérations sur l'ALU) contrôle en fait l'accès à ce bus.

Implantation

Différentes techniques

- Deux techniques d'implantation
 - implantation matérielle
 - implantation micro-programmée