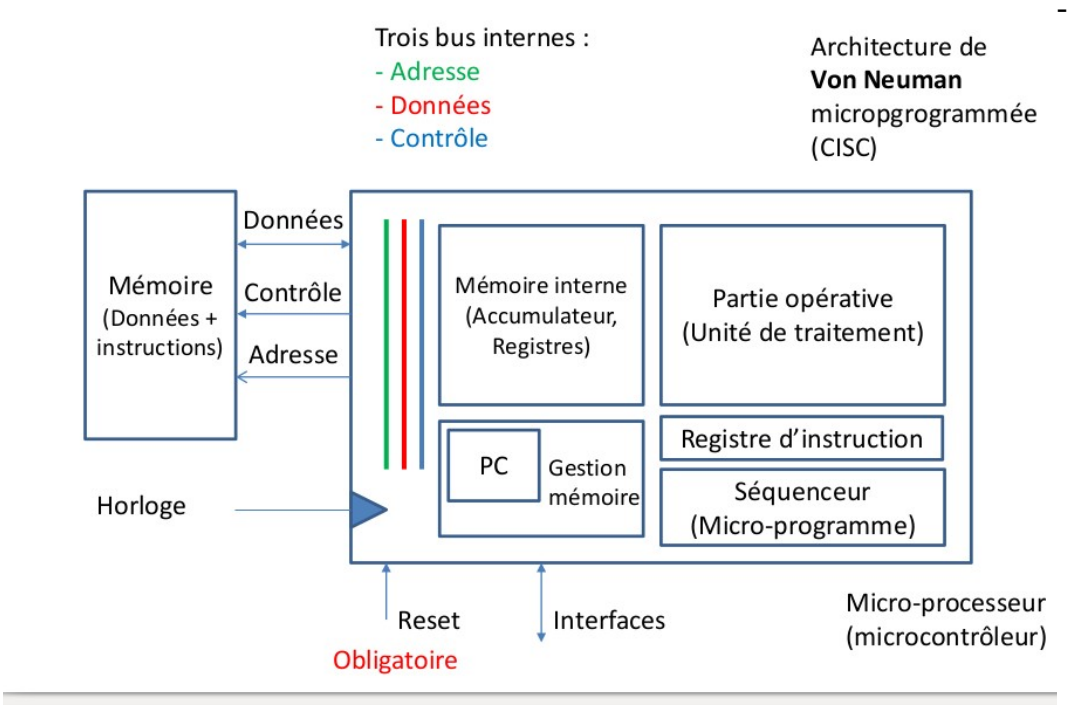
Module 3

Bus de contyrole, externe IHM, interne : transmet les instruction aux different systeme
-**opcode** (ou *code opération*) qui détermine la nature de l'instruction
-Un **registre** est un emplacement de mémoire interne à un processeur.
-**pointeur d'instruction** *(PC*) est le registre (souvent nommé **PC**) qui contient l'adresse mémoire de l'instruction en cours d'exécution ou prochainement exécutée
-**séquenceur** commande et contrôle le fonctionnement du système
-un **accumulateur** est un registre spécial, incorporé dans certaines architectures de processeur, où les résultats intermédiaires de l'UAL, ou ALU .
-ALU est l'organe de l'ordinateur chargé d'effectuer les calculs
-Un **bus informatique** est un dispositif de transmission de données partagé entre plusieurs composants d'un système numériqu
-Fetch= déclenchement du sous-séquenceur
-1cycle = 1 coup dhorloge
-Code machine (ou instruction machine) =Mot binaire codant l'instruction
• (Code) mnémonique : pour le programmeur =Mot « en clair » pour le manipuler (MOVE ADD)
• Microprogramme =Suite de microcommandes réalisant l'instruction
• Assembleur= Mnémoniques ➔ codes machines, Calcule sauts
• Désassembleur (fonction inverse de l'assembleur)

-



-Chaque instruction est codée sur 2 octets :
.Code opération : pour le registre d'instruction
(définit le mode d'adressage)
.Opérande ou adresse

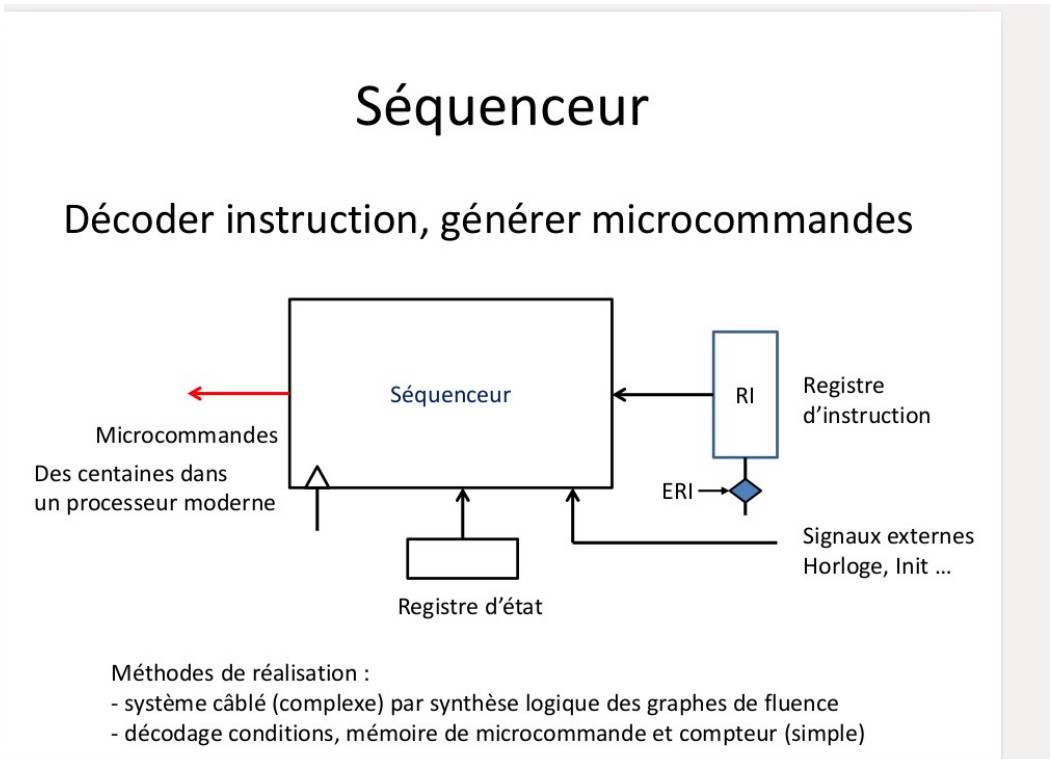
-pour architecture von deu mann tj 4 etapes :
1)Fetch, recherche de l'instruction.

2)Decode, interprétation de l'instruction (opération et opérandes).

3)Execute, exécution de l'instruction.

4)Writeback, écriture du résultat.

Sequenceur fonctionnement :
• Entrées séquenceur
– Instruction courante et opérande
– Prise en compte du registre d'état
– Prend en compte les signaux externes
• Sorties
– Activation et séquencement des accès aux registres
– Fabrication du code opération (ALU)
– Calcul des adresses de saut
– Signaux de gestion externe



-Les modes d'adressage définis dans une architecture régissent la façon dont les instructions en langage machine identifient leurs opérandes. Un mode d'adressage spécifie la façon dont est calculée l'*adresse mémoire effective* d'un opérande à partir de valeurs contenues dans des registres et de constantes contenues dans l'instruction ou ailleurs dans la machine.

Archi de Von Neumann : définitions
• Une unité de traitement (UT)
• Une mémoire (données + instructions)
• Un canal d'échange (BUS) partagé
1- UT : opérations élémentaires câblées
Traitement déclenché par une commande appelée instruction.
L'ensemble des instructions nommé le jeu d'instructions.
2- Exécution des tâches complexes : suite d'instructions exécutée séquentiellement nommée programme
3- Programme et données stockés dans une mémoire
= machine à programme enregistré.
4- Certaines instructions : ruptures de séquences conditionnelles

Module 3bis

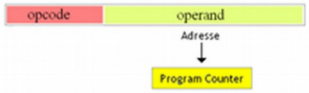
• Le programme commence à l'adresse 0
• On ne peut pas échanger avec l'extérieur
– Clavier, écran ...
• On ne peut pas réagir à une sollicitation
– Capteur, détecteur …

• Accumulateur
une mémoire interne « multiple »
– Chaque position est nommée registre avec adresse
– L'ensemble est nommé banc de registres
• Banc de registres
– Plusieurs banc possibles (chaque entrée de l'accu)
– Banc spécialisés possibles
• Données/adresses
• Entiers/flottants
– Eventuellement « switchables » (échangeables)
• Permet de gagner du temps pur changer de contexte

-Pile (stack) : mémoire à gestion first-in/last-out

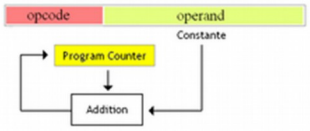
Modes d'adressage : branchements

– **Absolu** : l'adresse est comprise dans l'instruction



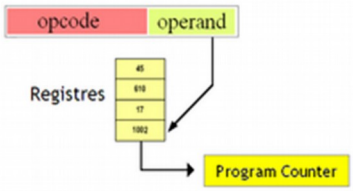
PC ← valeur

– **Relatif** : on indique juste un décalage
(permet d'avoir des instructions plus courtes, programmes translatables



PC ← PC + Dec

– **Indirect**
Si adresse de branchement inconnue à la compilation ou dépend du contexte.
Solutions : (ancien) code auto-modifiant
(moderne) adressage indirect
Adresse stockée dans un registre



PC ← (Reg)

Mécanisme d'interruptions
Si le processeur doit prendre en compte une entrée extérieure (bouton, capteur, clavier…)
• Le processeur vit sa vie
• Mécanisme câblé le prévient si activité externe
• Processeur termine action, sauvegarde contexte (pile) puis
se branche sur le programme de traitement prévu
• Traitement terminé
restitution du contexte et reprise

Adresse de la routine d'interruption
Classique : Base + n*numero_interruption
Exemple d'organisation
- le périphérique « lance « une interruption
- le périphérique s'identifie (donne son numéro)
- le numéro de l'interruption est associé à une adresse
- cette adresse contient l'adresse effective de la routine
(le sous-programme peut être localisé n'importe où
avec une longeur quelconque)

Pile : différentes possibilités : Sauvegarde/changement de contexte
– Par une pile externe (mémoire) [Toujours possible]
• Quasi illimité (limité par la mémoire)
– Par une pile interne (plus rapide)
• Nombre de niveau dépend de la taille
– Par switch de registre
• Limité mais extrêmement rapide

• Un interruption peut être
– Masquée : elle ne sera pas prise en compte
– Non masquée : elle sera prise en compte
– Non masquable : on ne peut pas l'ignorer
– Prioritaire : traitement rapide obligatoire
2types de traitement de routine : interruptible et non interruptible (à priorité logique)

RISC : Reduced Instruction Set Computer
– Processeur MIPS en 1986
Idée pour toutes les instructions
– Longueur de code = 1 mot
– Durée d'exécution = 1 cycle machine

RISC : Reduced Instruction Set Computer
– Instruction simple, longueur constante
– Beaucoup de registres usage général (plusieurs bancs ?)
– Transfert simples Mémoire registre
– Instructions générales : action sur les registres
– Nombre réduit de format de données
-compilateurs complexes
-pas de séquenceur (ou séquenceur très simple)
-programmes long (taille mémoire)

CISC : Complex Instruction Set Computer
– 1 instruction 1 à 100 cycles
– Longueur des instructions : 1 à 10 octets
– Toutes les instructions accèdent à la mémoire
– Modes d'adressage complexes
-compilateurs simples
-séquenceurs TRES compliqués

• Von Neumann
– Une seule mémoire pour données/instructions
• Harvard
– Une mémoire instruction
– Une mémoire données
bus séparés

Module 4 :

-Largeur de bus de données (nb) : nb de fils-
-Largeur de bus d'adresse (na) : nombre de fils
-Bus de contrôle : fils définissant le mode d'accès
-Capacité : 2 na mots de nb bits

• Mémoire synchrone/asynchrone
– Protocole d'accès basé ou pas sur une horloge
• Mémoire à adresse multiplexée
– Adresse présentée en deux fois : ligne/colonne
• Mémoire à adresse/données multiplexées
– Adresses et données sont sur les mêmes fils
• Cache (antémémoire)
– mémoire rapide entre processeur et la mémoire centrale ou entre deux niveaux de mémoire

• Temps d'accès : entre demande et réponse
• Temps de cycle : entre deux demandes
(lié au débit)
• Mode burst : plusieurs données à la suite

• Mémoire virtuelle : espace mémoire utilisé par les programmes du processeur
• Mémoire réelle : ensemble de supports permettant de « mapper » la mémoire virtuelle : mémoire centrale, disques durs, SSD, Cloud

ROM : mémoire à lecture seule
PROM : ROM programmable une seule fois
REPROM : ROM programmable plusieurs fois
• FLASH : Effaçable par page (voir histoire)

RAM (Read Access Memory)
⇒ SDRAM et RDRAM
SDRAM plus rapide

• DRAM : point mémoire capacité à 1T (condo)
Techno DRAM
• Dynamique : charge dans un condensateur
• Rafraîchissement : lecture puis réécriture (10 ns)
• Moins rapide que la SRAM

Inconvénient DRAM
• Les timings dépendent de la techno
• Le processeur attend « juste ce qu'il faut »
• Pas de référence temporelle, ni de handshake
– Pas de synchronisation avec le bus (ou le système)
• Difficulté de conception avec F↗
⇒ synchrone

• SRAM : points mémoire RS à 6T
– Volatile : perte info avec alimentation
SDRAM (SDR-SDRAM) (synchrone)
• Commandes synchronisées avec une horloge
• Pilotée par un séquenceur
• Architecture pipeline
– Traitement entrée avant fin opération précédente
⇒ Latence

• FLASH : interrupteur piloté par grille flottante
– Charge piégées par effet tunnel dans une grille isolée
– Pas de charge : interrupteur ouvert
– Charge : interrupteur fermé

Flash : Techno
Architecture NOR ou NAND
Grosses contraintes à l'effacement/écriture
– 10.000 à 100.000 cycles par cellule (↗)

Memoire cache
• Mémoire rapide entre le processeur et une mémoire plus lente
≠ tampon (buffer) : juste copie temporaire
principe
• Zone mémoire copiée dans le cache (mode burst)
• On travaille sur la copie
• Le processeur demande une donnée
Gestionnaire de cache vérifie disponibilité
disponible (hit) : donnée transmise
indisponible (miss) : demande au fournisseur
Le fournisseur fournit l'info

Le cache stocke (localité)

Chipset
• Composants « compagnons » d'un µproc
• Rôle : gérer des flux données entre le µproc et les divers composants et sous-ensembles de composants de la carte mère

Module 5

• La fréquence est passée de qq kHz à des GHz
– Problèmes : température/gestion horloge globale
– Presque plus de capacité d'évolution
• Taille puces ↗
rendement de fabrication ↘

Comparaison : avantages

• CISC
– Moins d'instructions pour une fonction
– Microprogrammation : correction jeu d'instructions
– Instructions très complexes/très rapides
• RISC
– Architecture globalement moins complexe
– Consommation d'énergie moindre
– Interruptions plus rapides

inconvénients

• CISC
– Surface importante de silicium (optimisation archi ?)
– Consommation importante
– Compilateurs complexes
– Temps développement/validation importants
• RISC
– Taille du programme plus importante (20 % à 50%)
– Instruction à taille fixe pas toujours optimisée
– Accès multiples à la mémoire pénalisés (chaînes,…)

Pipeline RISC (classique)
Recherche instruction
Décodage instruction
Recherche opérandes
Exécution opération
Sauvegarde résultats

pipeline ⇒ on predit les instructions en lancant plusieurs jeux dinstructions en meme temps

Thread o
Exécution d'une suite d'instructions.
Partie indépendante du code pouvant être exécutée seule

Multithreading
• Le système d'exploitation exécute les différents threads en temps partagé sur le processeur
– Thread après thread
– Ou entrelacement : Changement de contexte

Multi-core (cœurs)
• Plusieurs CPU en parallèle dans un même processeur

MODULE 6

Hyperthreading
• Processeur 4 cœurs physiques, 8 cœurs logiques
(de 4 à 8 threads)
Attention : pas 8 cœurs physiques

IPC : **communication inter-processus**
SMT : Le simultaneous multithreading forme de multithreading, une technique qui consiste, à augmenter le TLP (*thread level parallelism*), c'est-à-dire le parallélisme des threads.
TDP : L'**enveloppe thermique, (** *Thermal Design Power)*
Le TDP d'un processeur est utile à un fabricant de système de refroidissement pour ordinateur, ou de manière plus générale à un assembleur d'ordinateur ou à un utilisateur final d'ordinateur.

Le cache de premier niveau (L1), plus rapide et plus petit (cache de données pouvant être séparé du cache d'instructions) ;

Le cache de second niveau (L2), moins rapide et plus gros ;

Le cache de troisième niveau (L3), encore moins rapide et encore plus gros

GPU : Graphical Processing Unit

• Architecture spécialisée pour le graphique

CPU : Un **processeur** (ou **unité centrale de traitement**, **UCT**, en anglais *central processing unit*, CPU) est un composant présent dans de nombreux dispositifs électroniques qui exécute les instructions machine des programmes informatiques.