DIRO
IFT 1215

INTRODUCTION AUX SYSTÈMES INFORMATIQUES

LE CPU ET LA MÉMOIRE

Max Mignotte

Département d'Informatique et de Recherche Opérationnelle
Http: //www.iro.umontreal.ca/~mignotte/
E-mail: mignotte@iro.umontreal.ca

LE CPU ET LA MÉMOIRE

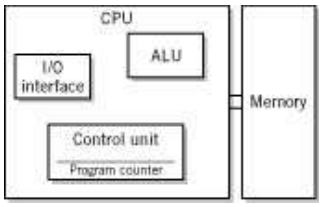
SOMMAIRE

LE CPU ET LA MÉMOIRE
INTRODUCTION & COMPOSANTES DU CPU

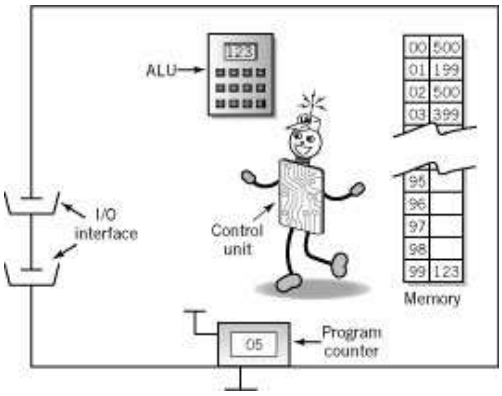
Introduction

Dans les ordinateurs réels, le jeu d'instruction et les opérations sont encodés en binaire et de la logique câblé (basée sur l'algèbre de Boole) remplace le LMC

Composants du CPU



Composant du CPU



The Little Man Computer

- ALU + CU = CPU ([*Central Process Unit*])

- CU ([*Control Unit*]) contrôle, interprète les instructions, lit le compteur d'instruction, et fait la séquence d'action correspondant au cycle *Fetch/Execute*

LE CPU ET LA MÉMOIRE
CPU & REGISTRES

CPU & Registres

- Consiste en plusieurs Circuits Intégrés ([*ICs*])
  - ▷ appelé aussi Microprocesseur

- Le CPU consiste en

1. Une série de Registres (lieu de stockage mémoire temporaire) pour stocker les instructions, les opérandes et le résultat de l'ALU
2. ALU (Unité Arithmétique et Logique), qui s'occupe des opérations arithmétiques et logiques binaires sur les opérandes
3. CU (Unité de Contrôle), véritable "chef d'orchestre" qui s'occupe de contrôler, interpréter et dérouler la séquence d'action correspondant au cycle *Fetch/Execute*

LE CPU ET LA MÉMOIRE
LE CONCEPT DE REGISTRES

Le concept de Registres

• Un registre est un espace de stockage mémoire temporaire dans le CPU

• Les registres peut être utilisé pour contenir un bit ([*flag*] ou plusieurs octets (ex; 128 bits)

• Les registres sont utilisés pour stocker des instructions, I/O adresses ou des codes binaires spéciales ([*flag*])

• Les registres, contrairement à la mémoire ne sont pas adressables de la même façon que la mémoire mais sont manipulé directement par l'unité de contrôle

• Chaque registre à une fonctionnalité particulière,

[*General-Purpose Register*]

1. Compteur de Programme (PC)
2. Registre d'Instruction (IR)
3. Registre d'Adresse Mémoire (MAR)
4. Registre de Donnée Mémoire (MDR)
5. Registre de Statut (ST)
6. Registre

Pour le LMC, 4 registres sont présents
le calculateur + le compteur de programme +
les deux registres I/O

LE CPU ET LA MÉMOIRE
OPÉRATION SUR LES REGISTRES

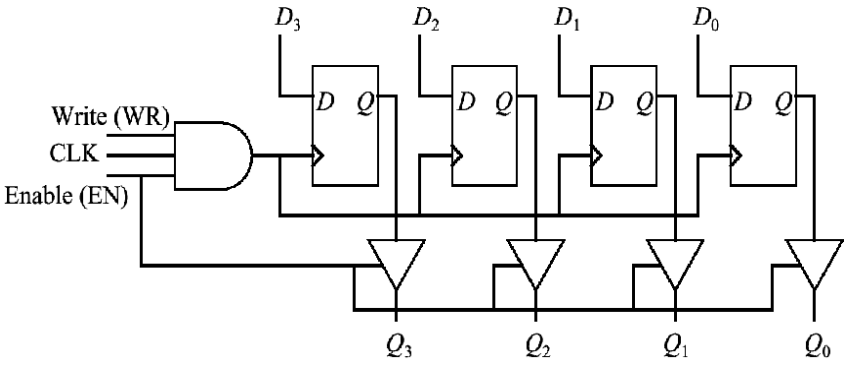
Opérations sur les Registres

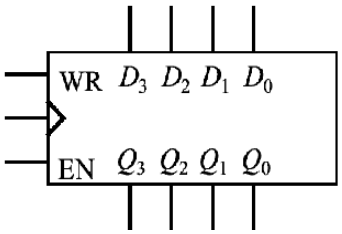
1. Les registres peuvent être chargées avec une valeur contenue dans d'autres registres ou à partir d'une mémoire
2. La donnée d'un registre peut être additionnée/soustraite [a]/[a partir de] la valeur d'un autre registre
3. La donnée d'un registre peut être décalée d'un (ou plusieurs) bits à droite ou à gauche
4. La valeur de la donnée d'un registre peut être initialisée à zéro, complémentée, incrémentée, décrémentée par un ou testée pour certaines conditions (> 0, $= 0$, < 0)

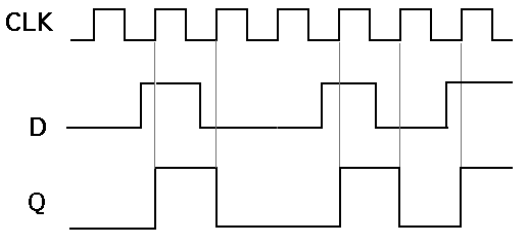
LE CPU ET LA MÉMOIRE
OPÉRATION SUR LES REGISTRES

Un Registre est un groupement de Flip-Flop

Registre 1 Bit

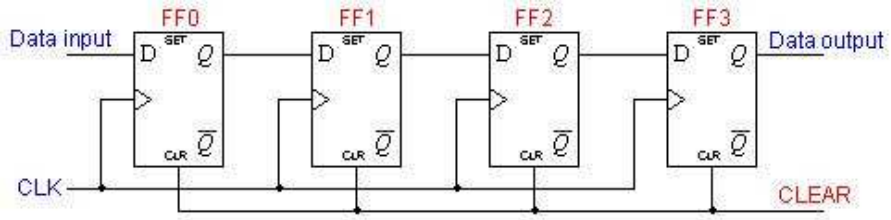






LE CPU ET LA MÉMOIRE
OPÉRATION SUR LES REGISTRES

Registre à Décalage

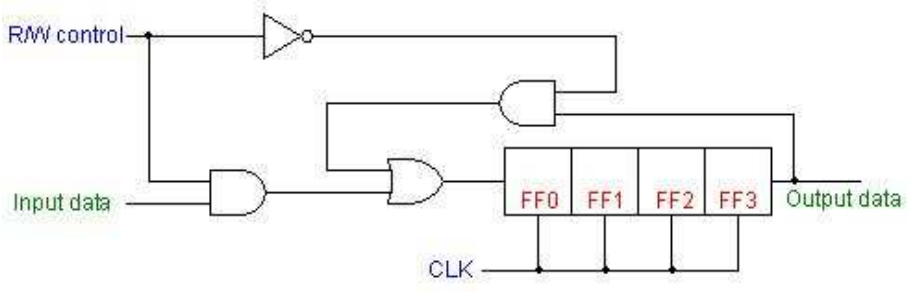


	FF0	FF1	FF2	FF3	
Clear	0	0	0	0	
1001	0	0	0	0	0
100	1	0	0	0	00
10	0	1	0	0	000
1	0	0	1	0	0000
-	1	0	0	1	00000
0000	1	0	0	1	-
000	0	1	0	0	1
00	0	0	1	0	01
0	0	0	0	1	001
-	0	0	0	0	1001

[*Destructive ReadOut*]

LE CPU ET LA MÉMOIRE
OPÉRATION SUR LES REGISTRES

Registre à Décalage



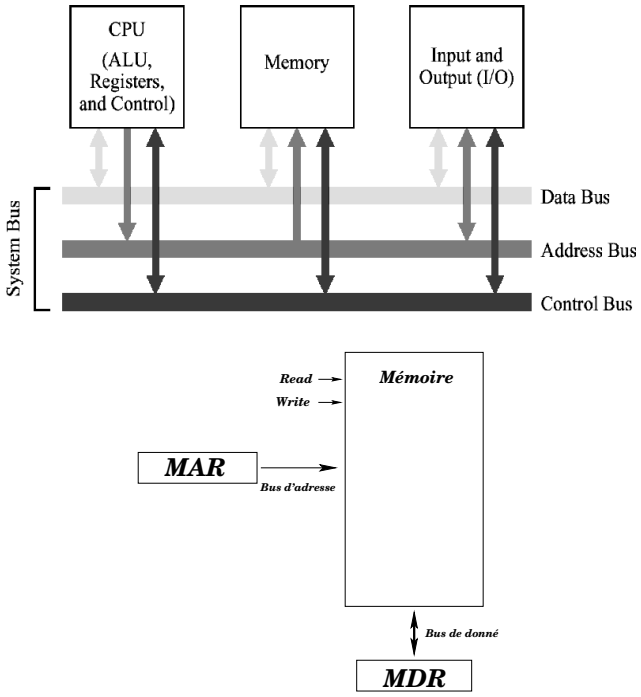
	FF0	FF1	FF2	FF3
Clear	0	0	0	0
1001	0	0	0	0
100	1	0	0	0
10	0	1	0	0
1	0	0	1	0
-	1	0	0	1
	1	1	0	0
	0	1	1	0
	0	0	1	1
	1	0	0	1

[*Non Destructive ReadOut*]

La Mémoire

• Une unité Mémoire consiste en un ensemble de cellule de stockage possédant une adresse propre et pouvant stoker une valeur binaire

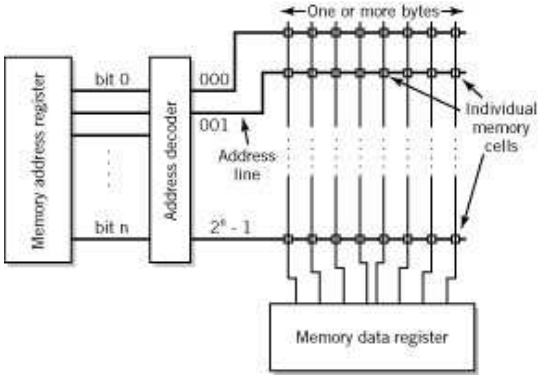
• la capacité d'une mémoire est le nombre d'octets qu'elle peut stocker



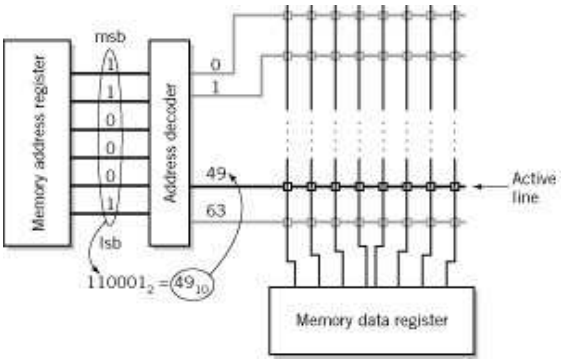
• MAR : [*Memory Address Register*] Registre contenant l'adresse de la mémoire ou sera stocké la donnée

• MDR : [*Memory Data Register*] Registre contenant la donnée/Instruction à être stocké/à être lu de la mémoire

La Mémoire et les Registres MAR & MDR

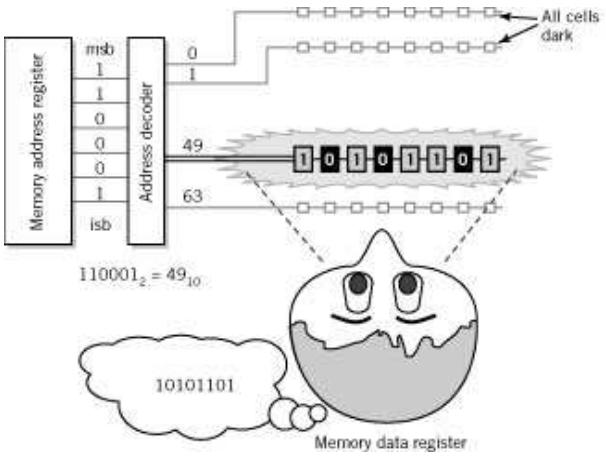


Exemple

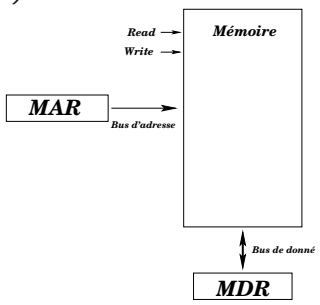


LE CPU ET LA MÉMOIRE
LA MÉMOIRE ET LES REGISTRES MAR & MDR

Analogie Visuelle



• La ligne Lecture/Écriture [*Read/Write*] détermine si la donnée sera transférée de la cellule mémoire au MDR (lecture) ou si celle ci sera transférée du MDR à la cellule mémoire (écriture)



• **Capacité Mémoire**: Nb de Bits dans MAR ▷ Nb d'adresses mémoires pouvant être décodées et Nb de Bits dans chaque cellule mémoire

LE CPU ET LA MÉMOIRE
CYCLE FETCH/EXECUTE

Cycle Fetch/Execute

• L'**UC** (Unité de controle/commande) est l'élément moteur de l'ordinateur qui interprête et execute les instructions du Prog

• C'est un ensemble des dispositifs coordonnant le fonctionnement de l'ordinateur afin de lui faire éxecuter la suite d'opération spécifiée dans les instructions du Prog selon

Le Format d'Instruction
et/ou
l'Architecture de la Machine

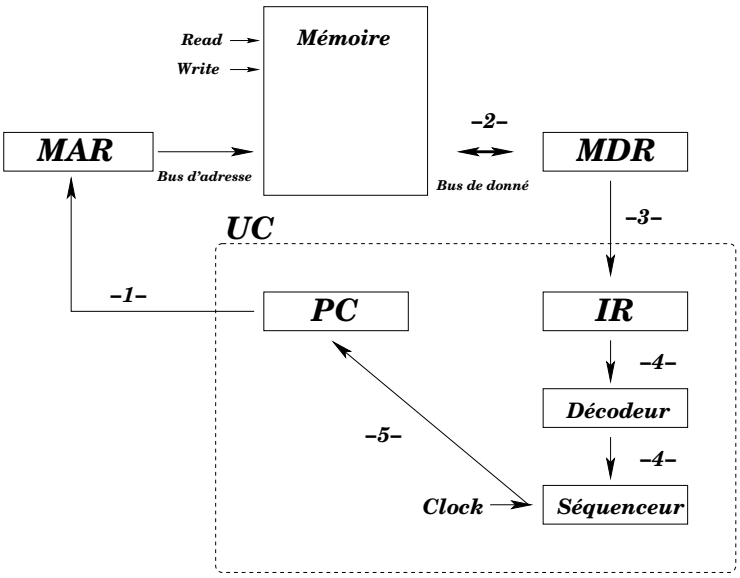
Format d'Instruction

Op-Code	*Operandes*

1. Le Code d'Opération [*Operation Code*] : Exemple: ADD, SUB, LOAD ...
   Le nombre de bits détermine le nombre d'opérations
2. Dans les cas les moins simples, divisés en plusieurs opérandes

LE CPU ET LA MÉMOIRE
CYCLE FETCH/EXECUTE

Cycle Fetch

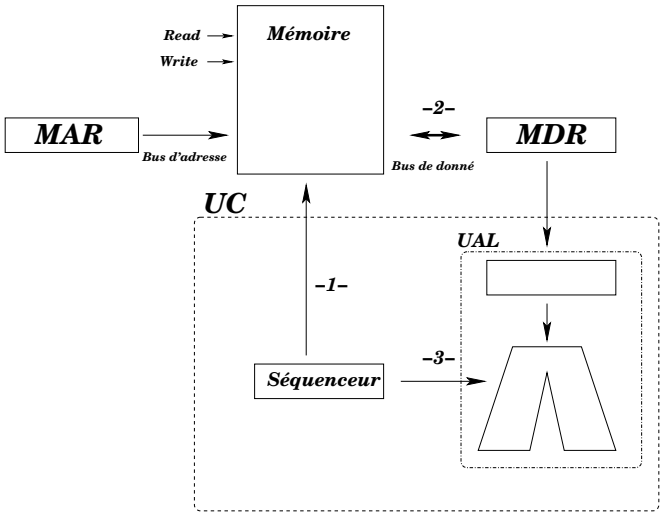


Cycle de Recherche

1. Transfert de la nouvelle adresse du PC ▷ MAR
2. Une impulsion de lecture, générée par l'UC, provoque le transfert de l'instruction cherchée ▷ MDR
3. Transfert de l'instruction dans le IR (instruction= opcode + adresse opérande(s))
4. L'adresse de l'opérande est envoyé au MAR, et le opcode est transmis au décodeur qui détermine le type d'opération demandée et le transmet au séquenceur en envoyant un signal sur la ligne de sortie correspondante
5. CP est incrémenté de un

LE CPU ET LA MÉMOIRE
CYCLE FETCH/EXECUTE

Cycle Execute

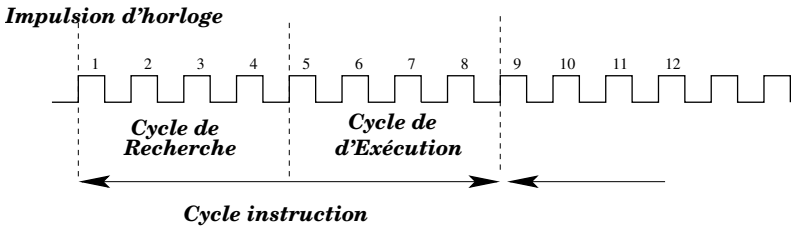


Cycle d'Exécution

1. Le séquenceur envoie le signal de commande vers la mémoire nécessaire pour lire l'opérande à l'adresse stockée dans le MAR et le faire parvenir dans le MDR
2. (a) MDR ▷ UAL et plus précisément dans l'accumulateur (ou tout autre registre de l'UAL affecté à l'opération spécifiée).
   (b) Dans certain cas (ex: mémorisation) ce sera le contenu de l'accumulateur ▷ MDR
   (c) S'il s'agit d'une instruction de Branchement, le champ d'adresse de l'instruction doit être transféré dans le CP
3. L'opération effectuée sous contrôle du séquenceur

LE CPU ET LA MÉMOIRE
CYCLE FETCH/EXECUTE

Synchronisation des Opérations



Séquenceur

Le **séquenceur** est un automate générant les signaux de commande nécessaires pour actionner et contrôler les unités participant à l'exécution d'une instruction donnée

Cet automate peut être réalisé de deux façon différentes

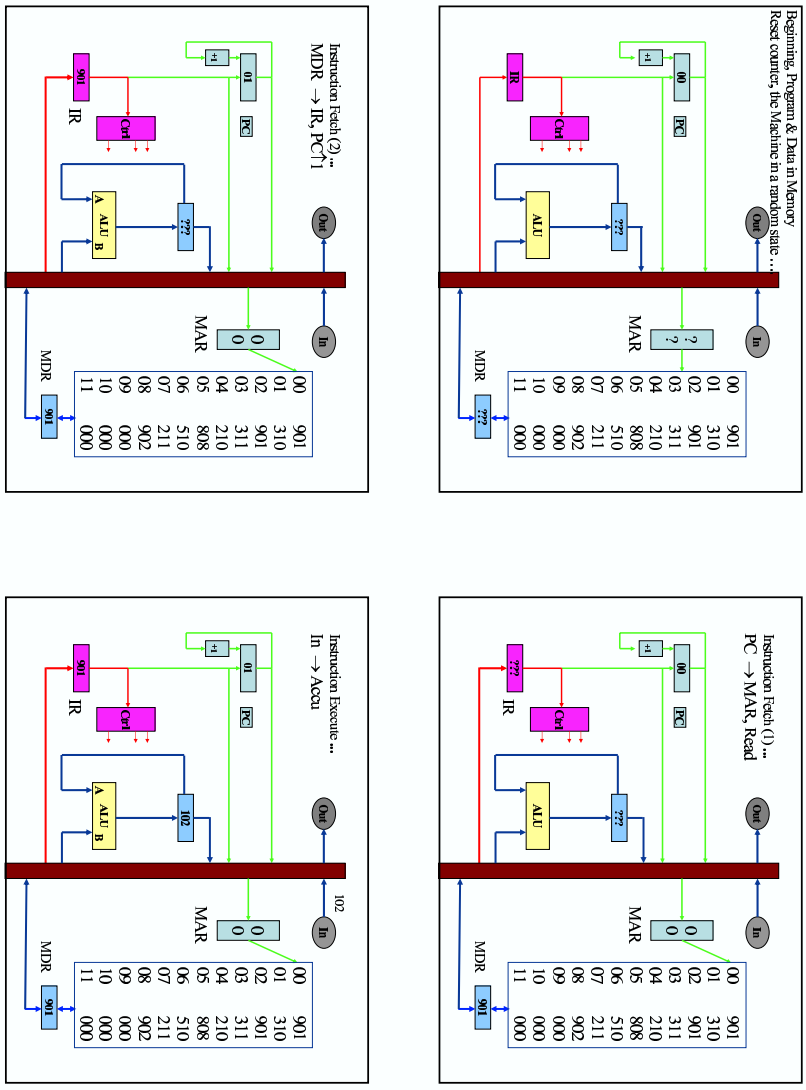
1. Séquenceur câblé
2. Séquenceur micro-programmé

Instructions

Ces instructions ou langage machine résultent de la compilation de langage haut niveau

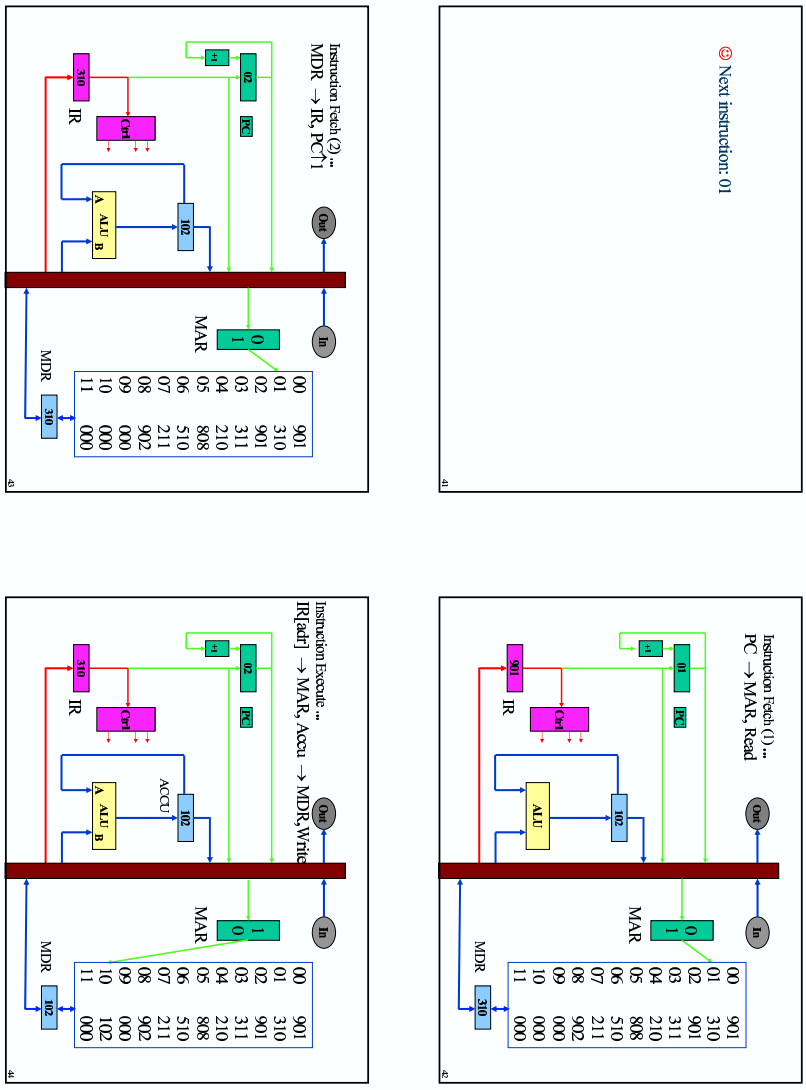
LE CPU ET LA MÉMOIRE

EX.: CYCLE FETCH/EXECUTE (© Irv Englander)



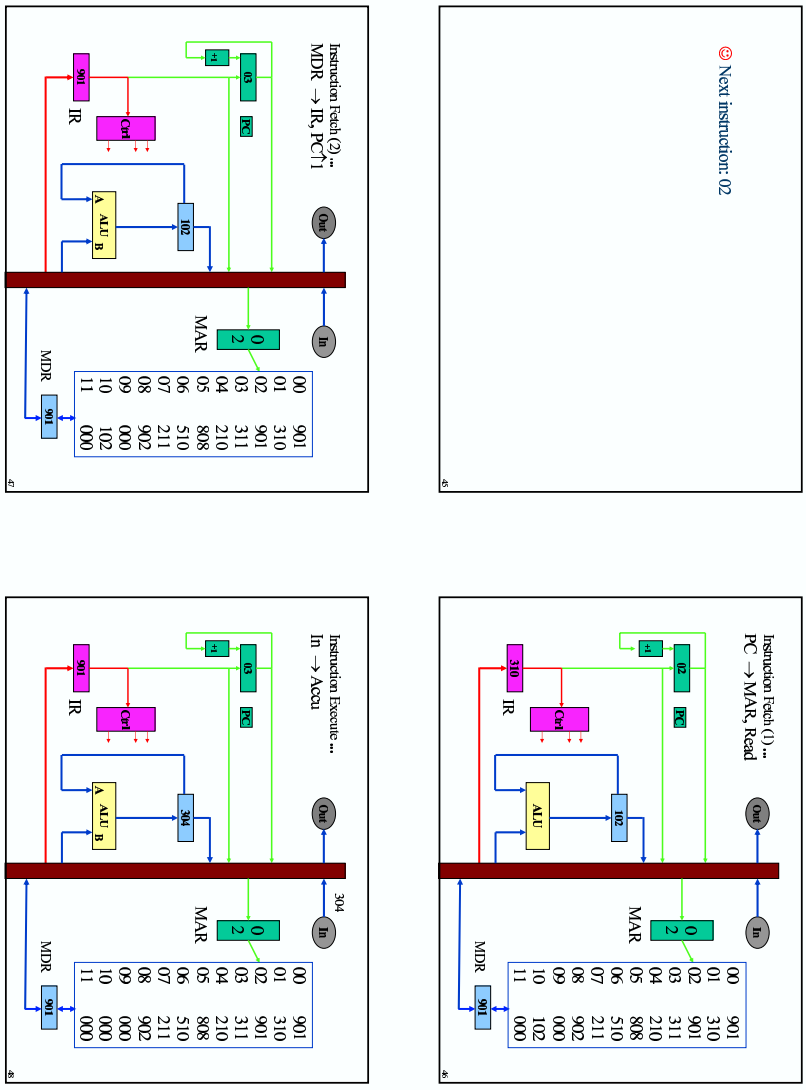
LE CPU ET LA MÉMOIRE

EX.: CYCLE FETCH/EXECUTE (© Irv Englander)

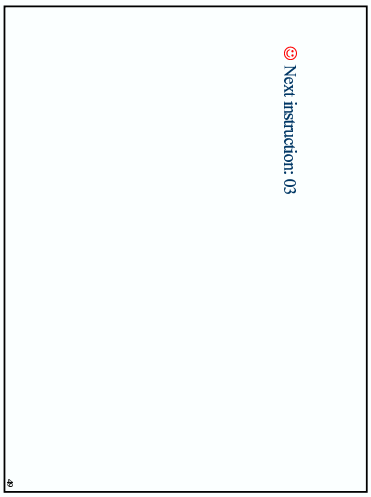


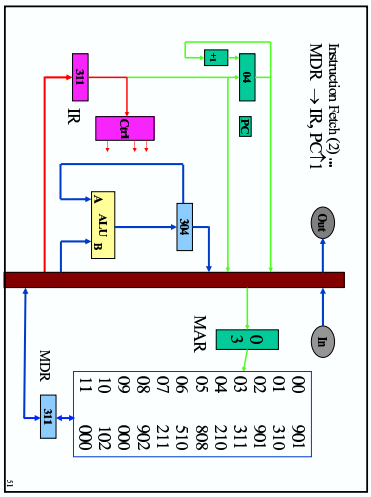
LE CPU ET LA MÉMOIRE

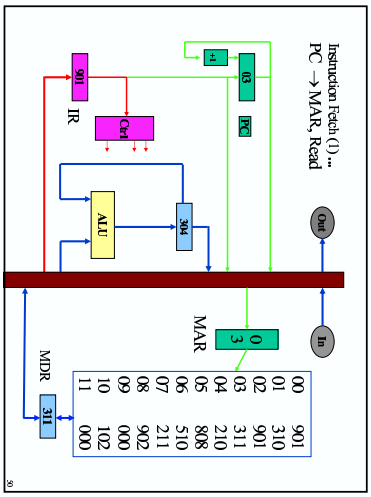
EX.: CYCLE FETCH/EXECUTE (© Irv Englander)

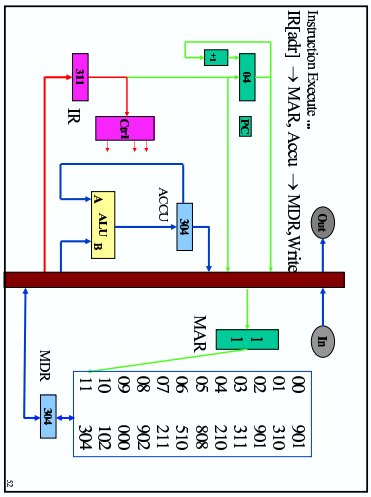


LE CPU ET LA MÉMOIRE
EX.: CYCLE FETCH/EXECUTE (© Irv Englander)



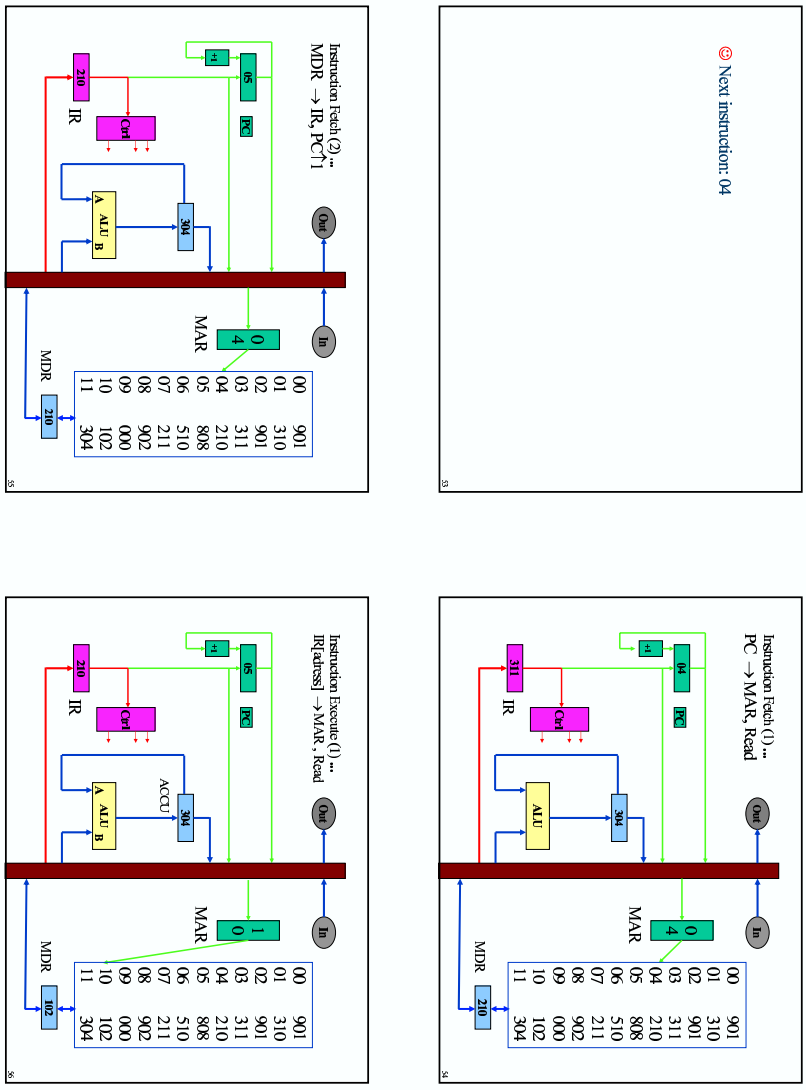






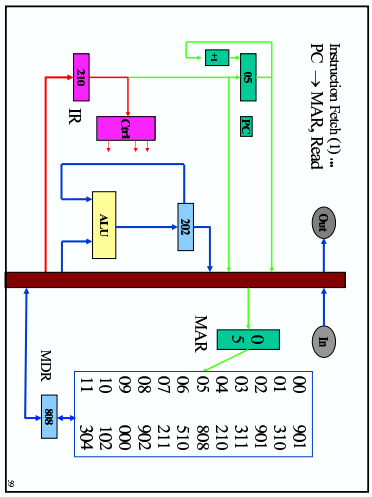
LE CPU ET LA MÉMOIRE

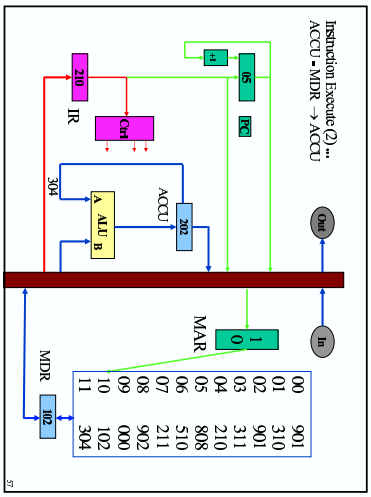
EX.: CYCLE FETCH/EXECUTE (© Irv Englander)

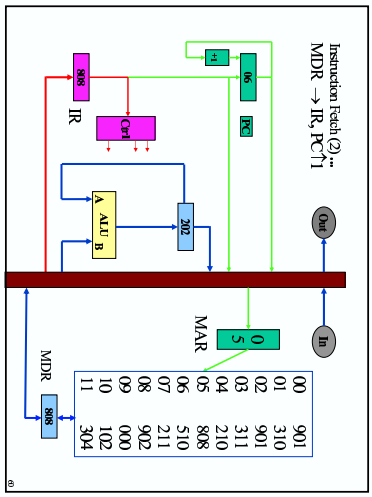


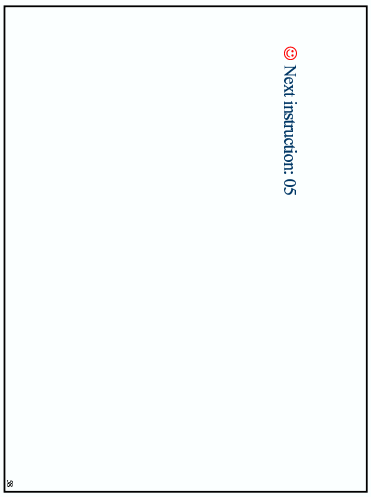
LE CPU ET LA MÉMOIRE

EX.: CYCLE FETCH/EXECUTE (© Irv Englander)



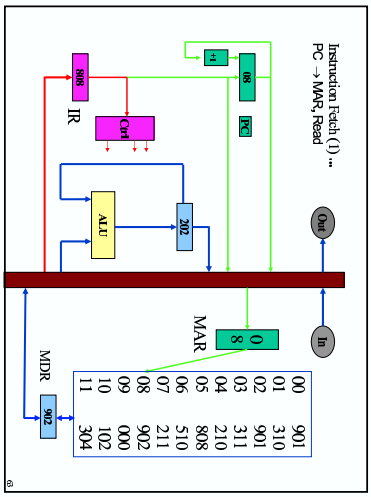


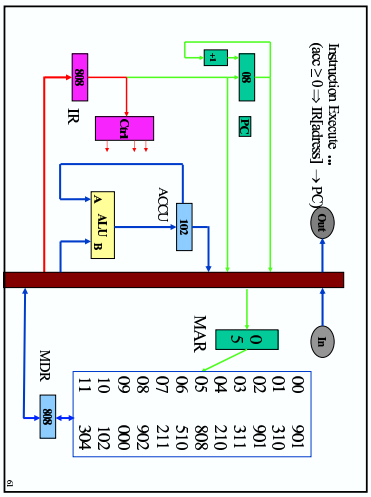


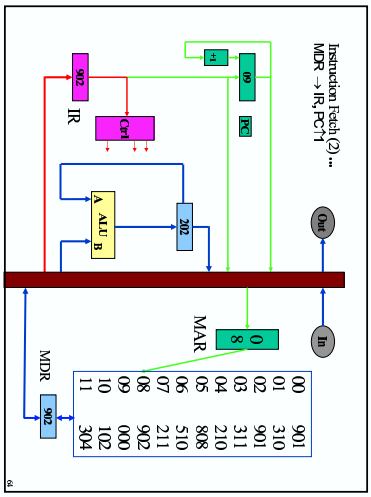


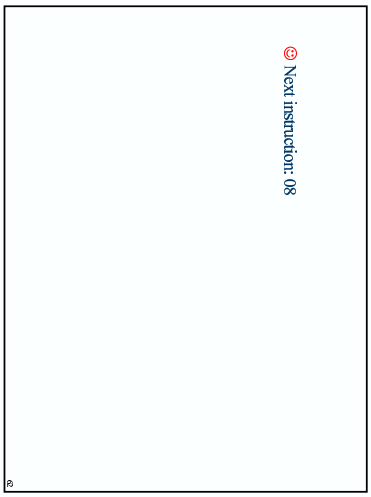
LE CPU ET LA MÉMOIRE

EX.: CYCLE FETCH/EXECUTE (© Irv Englander)



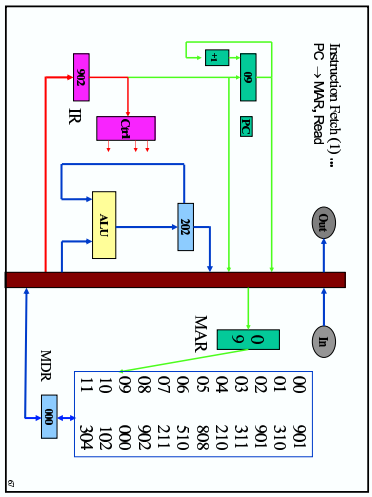


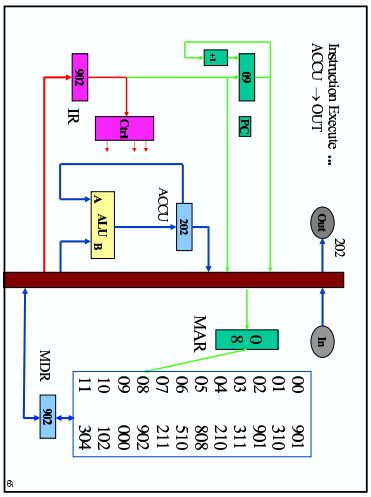


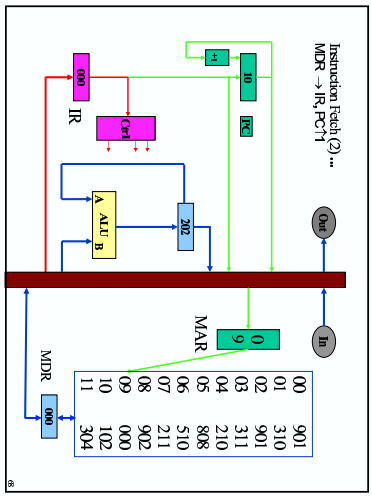


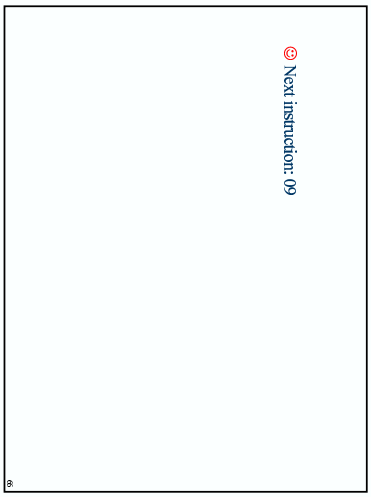
LE CPU ET LA MÉMOIRE

EX.: CYCLE FETCH/EXECUTE (© Irv Englander)



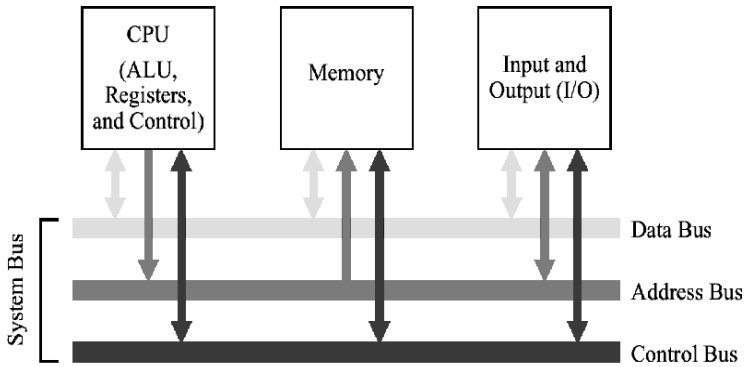






LE CPU ET LA MÉMOIRE
BUS

Bus



• Un Bus est un groupement de conducteur électrique permettant une connexion physique et le transport de signaux entre les différents composants de l'ordinateur

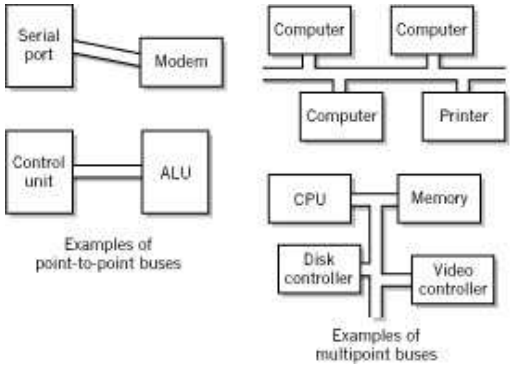
• Utilisé pour transférer donné/instructions entre le CPU, la mémoire et les périphériques

• 3 Types de lignes

1. Les Données
2. Les Adresses
3. Le Contrôle
4. La Puissance

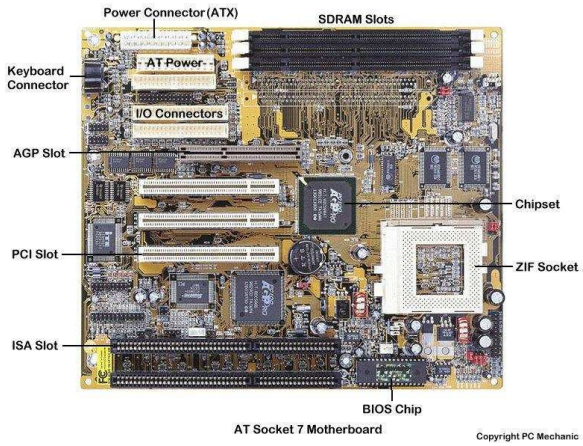
LE CPU ET LA MÉMOIRE
BUS

Type de Bus

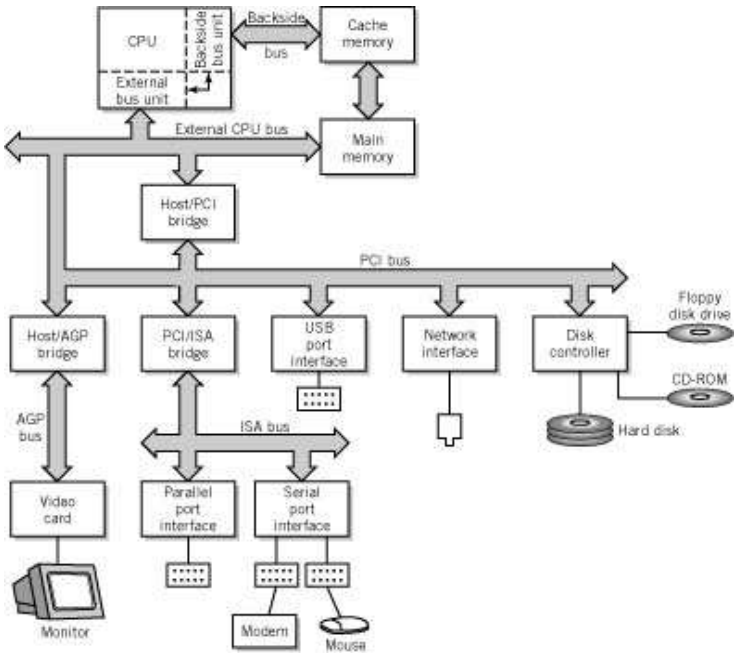


• Bus Multipoint [*multipoint Bus*],[*Broadcast Bus*], Type de Bus dans les réseau ethernet (nécessite une technique d'adressage $\neq$ Bus point à point)

• Le bus interne du CPU n'a pas de nom, le bus externe du CPU peut être le bus PCI [*Peripheral Connect Interface*], VESA [*Video Electronic Standart Architecture*], ISA [*Industry Standart Architecture*], etc.



LE CPU ET LA MÉMOIRE
BUS



• **Un protocole de Bus** est un ensemble de règles permettant la communication et la compréhension entre deux entités

• Un Bus est caractérisé par

1. Le Taux de Transfert (Nb Bits/sec)
2. La largeur des données (32 bits, 64 bits, etc.)
3. La distance entre deux points final du bus
4. Le type de Bus
5. La capacité d'adressage, de contrôle requis, etc.

LE CPU ET LA MÉMOIRE
INSTRUCTIONS DE CONTROLE DE PROGRAMME

Instruction de Controle de Programme