



DIRO  
IFT 1215

# INTRODUCTION AUX SYSTÈMES INFORMATIQUES

## THE LITTLE MAN COMPUTER

*Max Mignotte*

Département d'Informatique et de Recherche Opérationnelle  
Http: [//www.iro.umontreal.ca/~mignotte/](http://www.iro.umontreal.ca/~mignotte/)  
*E-mail: [mignotte@iro.umontreal.ca](mailto:mignotte@iro.umontreal.ca)*

# THE LITTLE MAN COMPUTER

## SOMMAIRE

|   |    |
|---|----|
| Introduction: The Little Man Computer ..... | 2  |
| Composants et Dispositions .....            | 3  |
| Opérations et Jeux d'Instructions .....     | 5  |
| Exécution d'un Programme .....              | 8  |
| Exemple de Programme .....                  | 9  |
| Cycle d'Instruction .....                   | 10 |
| Exemple de Programme .....                  | 12 |
| Observations et Remarques .....             | 13 |

# THE LITTLE MAN COMPUTER

## INTRODUCTION

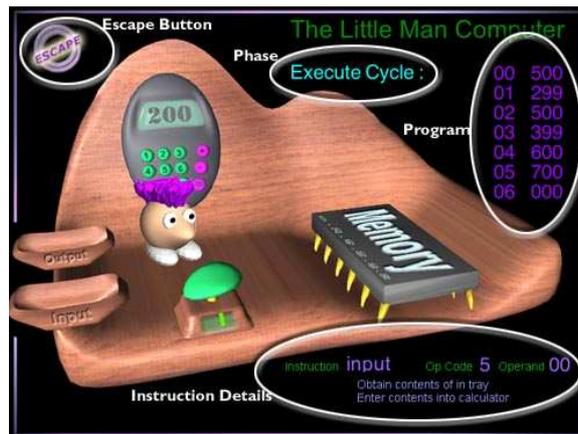
### Introduction: "The Little Man Computer"

---

- Appelé le "Little Man Computer" (LMC) et créé par Dr. Stuart Madnick au MIT en 1965 et modifié en 1979

<http://web.mit.edu/smadnick/www/home.html>

- Modèle simple décrivant le fonctionnement (simplifié) d'un ordinateur réel à travers ses cinq composantes de bases et de leurs interactions (cycle "Fetch/Execute")
- Ce modèle introduit un jeu d'instruction (en langage machine) simplifié et opérations associés et de quelle façon celles ci sont exécutés dans le LMC pour faire des programmes

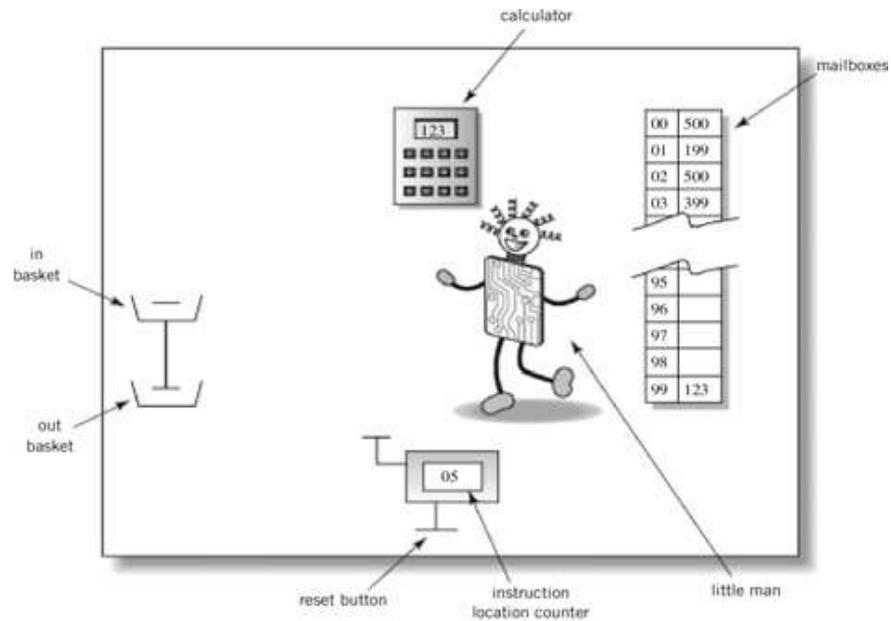


The Little Man Computer

# THE LITTLE MAN COMPUTER

## COMPOSANTS ET DISPOSITION

### Composants & Disposition



Englander: The Architecture of Computer  
Hardware and Systems Software, 2nd edition  
Chapter 6, Figure 06-01

- 100 Boites à lettres, chacune étant identifiée par une adresse (comprise entre 00-99) et permettant de stocker un nombre décimal de 3 chiffres
- Une calculatrice qui peut être utilisé pour entrer un nombre (positif), le stocker provisoirement, et faire des additions et soustractions
- Un compteur (d'instruction) manuel à deux digits (avec Reset externe)
- Un panier "IN" et un panier "OUT"
- Le LMC

# THE LITTLE MAN COMPUTER

## COMPOSANTS ET DISPOSITION

- **Calculatrice**

Comprend deux lumières qui peuvent être vu par le LMC, une de ces lumières s'allume quand le nombre est égale à zéro. L'autre s'allume quand le nombre est positif

- **Bouton START**

L'opérateur du LMC (et non le LMC) presse le Bouton START pour initialiser le compteur d'instruction à zéro et avertit le LMC qu'il peut exécuter le programme stocké dans les boîtes aux lettres

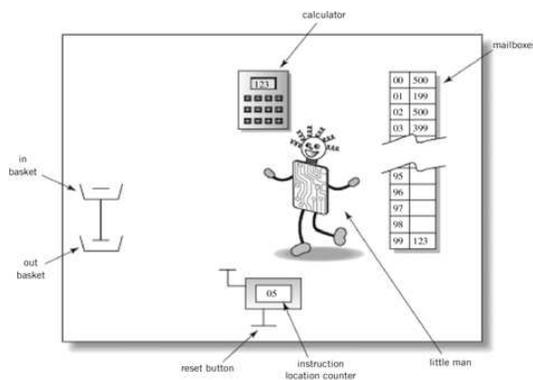
- **Compteur d'instruction**

Affiche l'adresse de la boîte au lettre dans laquelle la **prochaine** instruction est stockée. Le LMC peut changer le numéro dans le compteur d'instruction (soit par incrémentation ou remplacement) et l'utilisateur du LMC peut ré-initialiser ce compteur à 00 en utilisant le bouton START

- **Pannier "In" et "Out"**

Un utilisateur communique avec le LMC en

- ▷ Posant un chiffre de trois digits dans "IN"
- ▷ Prenant un chiffre de trois digits dans "OUT"



# THE LITTLE MAN COMPUTER

## OPÉRATIONS ET JEUX D'INSTRUCTIONS

### Opérations

---

- Le premier digit du nombre de trois chiffres est utilisé pour dire au LMC quel type d'opération il doit faire
- Dans quelques cas, une opération demandera au LMC d'utiliser une boîte aux lettres particulière pour stocker ou retrouver une donnée (i.e., un nombre de 3 chiffres)

|                    |  |                                      |
|--------------------|--|--------------------------------------|
| 3                  |  | 25                                   |
| <i>Instruction</i> |  | <i>Adresse de la boîte au lettre</i> |

- La partie instruction du nombre de 3 digits s'appelle le code d'opération [*operation-code*] [*op-code*]

### Jeux d'instructions

---

| FORMAT      | MNEMONIC             | SENS  |
|-------------|----------------------|---|
| 3 <i>xx</i> | <b>STO</b> <i>xx</i> | Stoke la valeur du calculateur dans la boîte au lettre <i>xx</i>  |
| 4 <i>xx</i> | <b>STA</b> <i>xx</i> | Stoke la partie adresse de la valeur du calculateur (2 derniers digits) dans la partie adresse de l'instruction contenu dans la boîte au lettre <i>xx</i> |

# THE LITTLE MAN COMPUTER

## OPÉRATIONS ET JEUX D'INSTRUCTIONS

| FORMAT      | MNEMONIC      | SENS  |
|-------------|---------------|---|
| 1 <i>xx</i> | ADD <i>xx</i> | Additionne le contenu de la boîte au lettre <i>xx</i> au nombre affiché par le calculateur  |
| 2 <i>xx</i> | SUB <i>xx</i> | Soustrait le contenu de la boîte au lettre <i>xx</i> au nombre affiché par le calculateur   |
| 5 <i>xx</i> | LDA <i>xx</i> | Charge le contenu de la boîte au lettre <i>xx</i> dans le calculateur   |
| 6 <i>xx</i> | B <i>xx</i>   | Initialise le compteur d'instruction avec le nombre <i>xx</i><br>▷ Prochaine instruction à exécuter contenu dans la boîte au lettre <i>xx</i>                                 |
| 7 <i>xx</i> | BZ <i>xx</i>  | Si valeur du calculateur = 0 <b>alors</b> initialise le compteur d'instruction avec <i>xx</i><br>▷ Prochaine instruction à exécuter contenu dans la boîte au lettre <i>xx</i> |

# THE LITTLE MAN COMPUTER

## OPÉRATIONS ET JEUX D'INSTRUCTIONS

| FORMAT      | MNEMONIC     | SENS  |
|-------------|--------------|---|
| 8 <i>xx</i> | BP <i>xx</i> | Si valeur du calculateur $\geq 0$<br><b>alors</b> initialise le compteur<br>d'instruction avec <i>xx</i><br>▷ Prochaine instruction<br>à exécuter contenue dans la<br>boite au lettre <i>xx</i> |
| 901         | INPUT        | Lit un nombre provenant du<br>du panier "IN" et le place<br>dans le calculateur   |
| 902         | OUTPUT       | Copie le nombre présent dans<br>la calculateur et le place<br>dans le panier "OUT"  |
| 000         | STOP         | Stoppe le LMC   |

# THE LITTLE MAN COMPUTER

## EXÉCUTION D'UN PROGRAMME

### Exécution d'un programme

Pour exécuter un programme selon le modèle LMC, on doit suivre les étapes suivantes

1. Charge [*Load*] les instructions du programme dans les boîtes aux lettres en partant de la boîte no 00
2. Place la (les) donnée(s) qui sera utilisée par le programme dans le panier "IN" (dans l'ordre ou le programme les utilisera)
3. Presse le Bouton RESET pour initialiser le compteur d'instruction à 00 et avertir le LMC qu'un programme doit être exécuté

### *Cycle d'Instruction*

1. Le LMC lit la valeur contenue  $x$  dans le compteur d'instruction
2. Le LMC lit la valeur  $y$  contenue dans la boîte d'adresse  $x$  [*cycle Fetch*]
3. Le LMC exécute l'instruction  $y$  [*cycle Execute*]
4. Le LMC incrémente le compteur d'instruction par "1"
5. Retour à l'étape 1.

# THE LITTLE MAN COMPUTER

## EXEMPLE DE PROGRAMME

### Exemple de Programme

Écrire un programme qui additionne deux nombres ensemble avec le jeux d'instruction vu précédemment

|            |     |                            |
|------------|-----|----------------------------|
| LDA        | 5xx | Load                       |
| STO        | 3xx | Store                      |
| ADD        | 1xx | Add                        |
| SUB        | 2xx | Subtract                   |
| IN         | 901 | Input                      |
| OUT        | 902 | Output                     |
| COB or HLT | 000 | Coffee break (or Halt)     |
| BRZ        | 7xx | Branch if zero             |
| BRP        | 8xx | Branch if positive or zero |
| BR         | 6xx | Branch unconditional       |
| DAT        |     | Data storage location      |

| Mailbox code |     | Instruction description |
|--------------|-----|-------------------------|
| 00           | 901 | INPUT                   |
| 01           | 399 | STORE DATA              |
| 02           | 901 | INPUT 2ND #             |
| 03           | 199 | ADD 1ST # TO IT         |
| 04           | 902 | OUTPUT RESULT           |
| 05           | 000 | STOP                    |
| 99           |     | DATA                    |

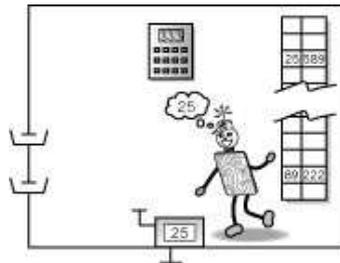
# THE LITTLE MAN COMPUTER

## CYCLE D'INSTRUCTION

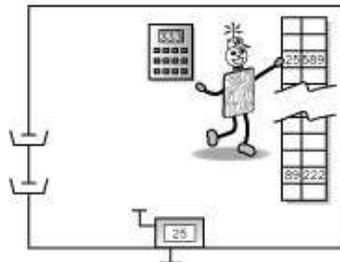
### Partie FETCH Du Cycle d'Instruction

---

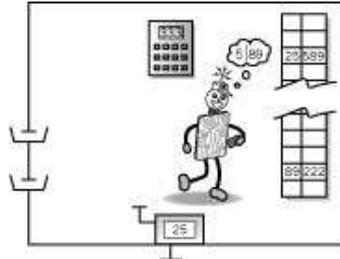
Pour L'instruction **LOAD**



(1) The Little Man reads the address from the location counter



(2) ... walks over to the mailbox that corresponds to the location counter



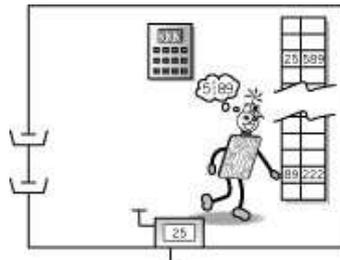
(3) ... and reads the number on the slip of paper. (He then puts the slip of paper back, in case he should need to read it again later.)

# THE LITTLE MAN COMPUTER

## CYCLE D'INSTRUCTION

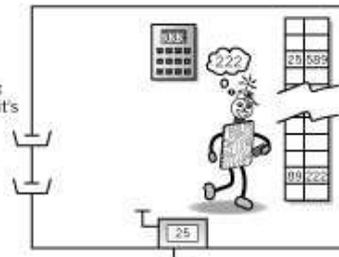
### Partie EXECUTE Du Cycle d'Instruction

#### Pour L'instruction **LOAD**

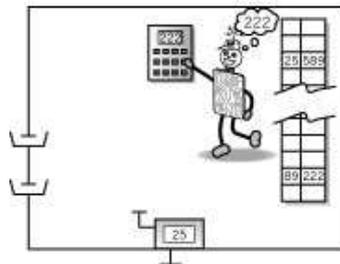


(1) The Little Man goes to the mailbox address specified in the instruction he previously fetched

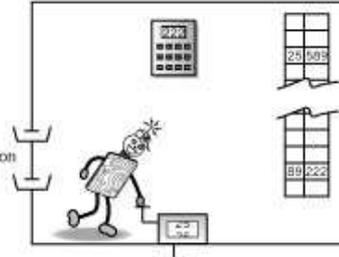
(2) ... he reads the number in that mailbox (he remembers to replace it in the case it's needed again)



(3) ... he walks over to the calculator and punches the number in



(4) ... finally, he walks over to the location counter and clicks it, which gets him ready to fetch the next instruction



# THE LITTLE MAN COMPUTER

## EXEMPLE DE PROGRAMME

### Exemple de Programme

Écrire un programme qui trouve la valeur absolue de la différence entre deux nombre avec le jeux d'instruction

|            |     |                            |
|------------|-----|----------------------------|
| LDA        | 5xx | Load                       |
| STO        | 3xx | Store                      |
| ADD        | 1xx | Add                        |
| SUB        | 2xx | Subtract                   |
| IN         | 901 | Input                      |
| OUT        | 902 | Output                     |
| COB or HLT | 000 | Coffee break (or Halt)     |
| BRZ        | 7xx | Branch if zero             |
| BRP        | 8xx | Branch if positive or zero |
| BR         | 6xx | Branch unconditional       |
| DAT        |     | Data storage location      |

|    |     |    |     |                         |
|----|-----|----|-----|-------------------------|
| 00 | IN  |    | 901 |                         |
| 01 | STO | 10 | 310 |                         |
| 02 | IN  |    | 901 |                         |
| 03 | STO | 11 | 311 |                         |
| 04 | SUB | 10 | 210 |                         |
| 05 | BRP | 08 | 808 | test                    |
| 06 | LDA | 10 | 510 | negative; reverse order |
| 07 | SUB | 11 | 211 |                         |
| 08 | OUT |    | 902 | print result and        |
| 09 | COB |    | 000 | stop.                   |
| 10 | DAT | 00 | 000 | used for data           |
| 11 | DAT | 00 | 000 | "                       |

# THE LITTLE MAN COMPUTER

## OBSERVATIONS ET REMARQUES

### Observations et Remarques

---



Von Neumann développa en 1945-1951 les grandes lignes de ce qui allait être le principe de fonctionnement des ordinateurs modernes

### Architecture de Von Neumann

- D'autres architectures ont été testé et construite mais aucune d'entre elles n'ont été commercialisées !
- L'architecture des ordinateurs reste virtuellement inchangé depuis 1951, alors que la technologie des composant évolue si vite (cf. Loi de Moore)

### Concepts clés de l'Architecture de Von Neumann

1. Concept de Programme stocké en mémoire (mémoire stockant les données et le programme)
2. La mémoire est adressée linéairement (adresse numérique séquentielle unique pour chaque espace mémoire)
3. Chaque espace mémoire possède une adresse et un contenu tous deux étant différencié
4. Les instructions s'exécutent linéairement à moins d'une instruction spécifique de branchement