



The Processor

The Processor

Learning Outcomes be able to:

- explain the impact of transistor miniaturisation
- describe the different levels of detail/abstraction in a processor
 - i.e. transistor, logic gates, registers adders, CPU
- explain the operation of a simple LMC program
- describe the components in the von Neumann architecture
- explain the role of the buses

This computer

Intel Core i5

1.6 GHz

Number of Processors: 1

Total Number of Cores: 2

L2 Cache (per Core): 256 KB

L3 Cache: 3 MB

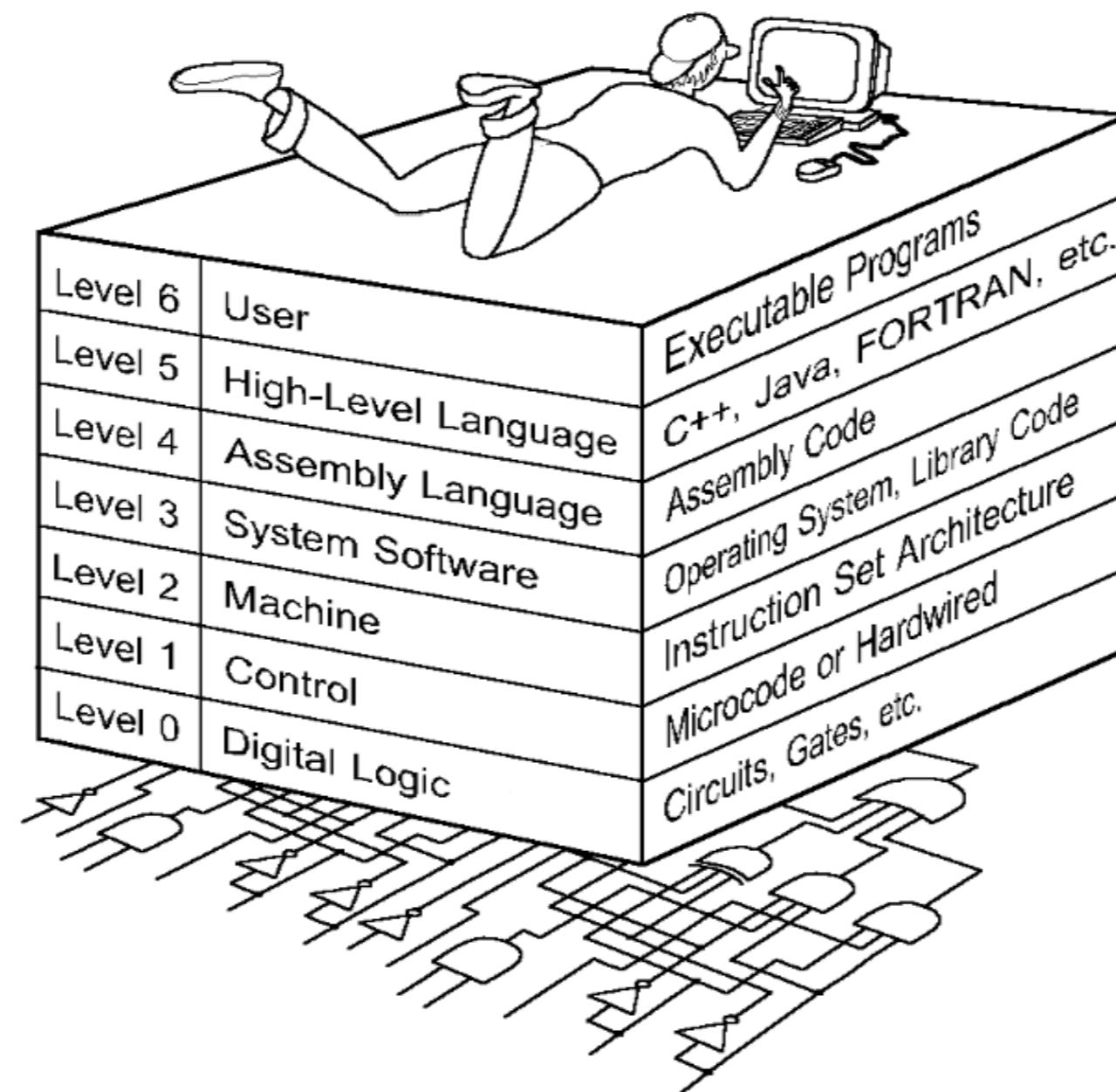
Memory: 8 GB

I/O bus: DMI

121GB SSD (121,018,208,256 bytes,



Computer Hierarchy



Transistors

Semiconductor device used to amplify and switch electronic signals and electrical power

Invented at Bell (AT &T) Laboratories in 1947

First commercially successful transistor was produced by Texas Instruments in 1954.

The first IC (Integrated Circuits: Several transistors on 1 chip) was produced in 1958 at Texas Instruments.



Transistor

Basic building block for all electronic processing and storage.

Ideally suited for Binary operation

Acts as a Switch which is On or Off

Combined to implement logic gates

AND, OR, NOT

Combined to build higher-level structures

Eg Adder

Combined to build processor

more powerful computers

Huge progress on miniaturisation: cheaper, quicker, smaller



Integrated Circuits (IC)

Set of electronic components on a single unit

Also called chip, microchip

Microprocessor - computer processor that implements the functions of a CPU (central processing unit) on a single IC



Intel core i5

Transistor size and density

Manufacturing process

10 micrometers (10^{-6}) in 1971

800 nm (10^{-9}) in 1989

Current production 20-28 nm

Intel 14nm in 2014

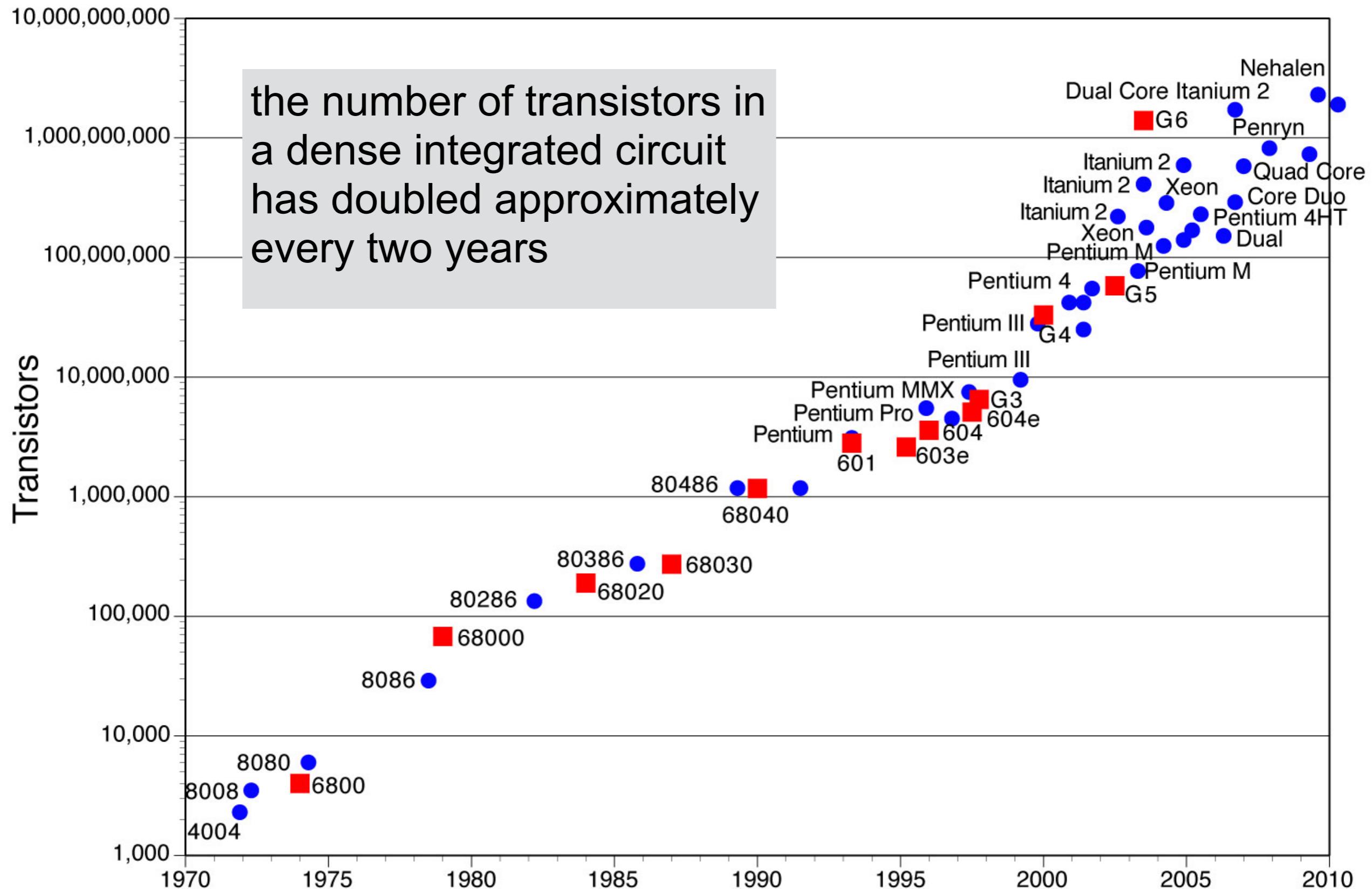
Nvidia 11nm in 2015

Predicted 5nm in 2020

Reaching physically possible limitations using current technologies?

Atom size 0.1nm - 0.5nm

Moore's Law



Google Pixel 2

Qualcomm Snapdragon processor

10-nm

3 billion transistors

64 bit

8 core

2.45GHz

includes a GPU

4GB RAM

64 GB or 128GB storage



Apple iPhone X

A11 Bionic

4.3 billion transistors

10 nm

64bit

6 cores

2.5 GHz

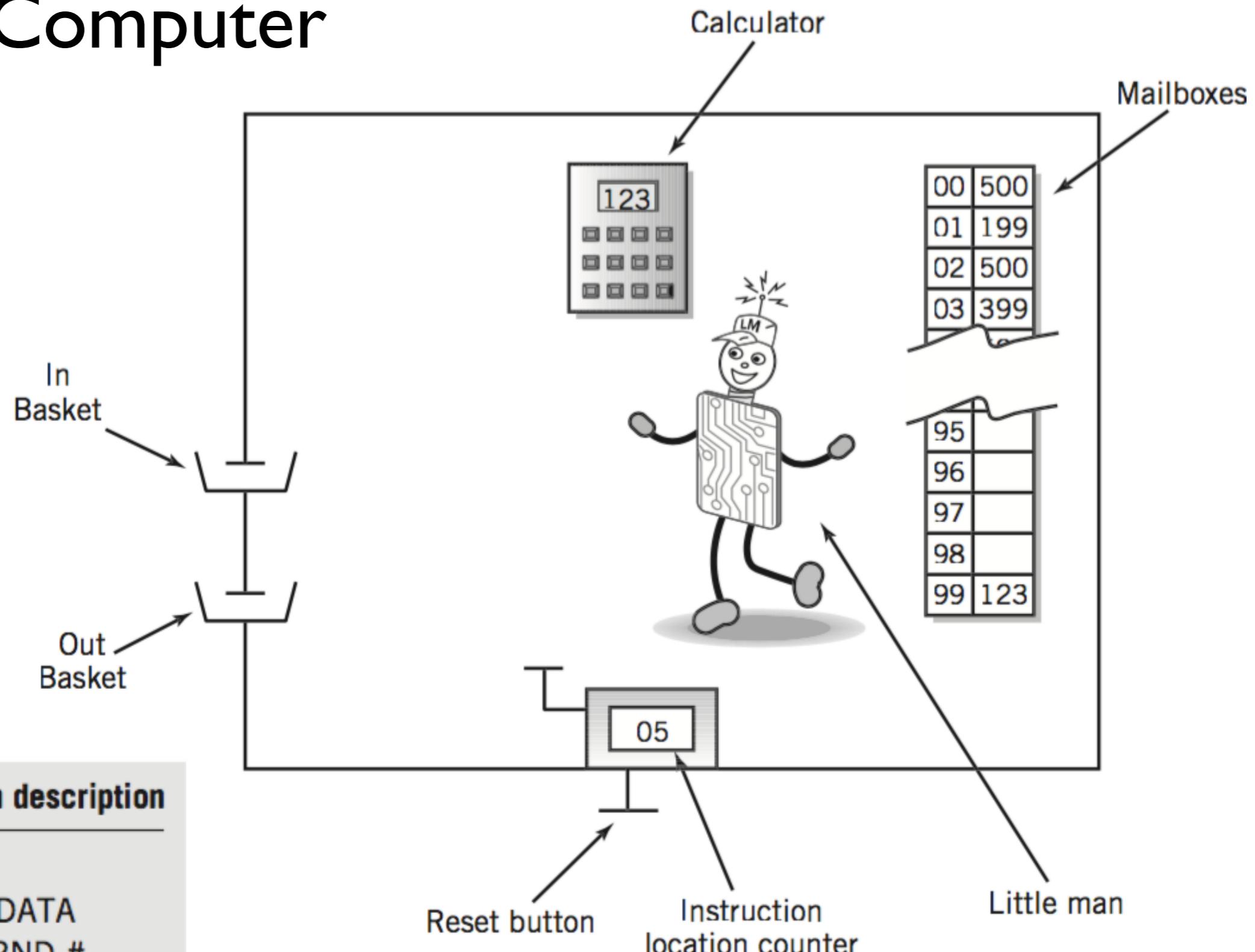
GPU

3GB RAM

64 GB or 128GB storage



Little Man Computer



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

LMC Code

Commands

901 INP

Read the INPUT and store it in the Accumulator

3xx STA

Store the contents of the the Accumulator at address xx

1xx ADD

Add the contents of address xx to the Accumulator

902 OUT

Read the contents of the Accumulator and present it at the output

Add two numbers:

```
00 INP 9 01
01 STA 3 99
02 INP 9 01
03 ADD 1 99
04 OUT 9 02
05 HLT 0 00
```

Another LMC Program

Read in two numbers and write out the larger one

⇒ conditions, branching

5xx LDA

Load the contents of address xx to the Accumulator

2xx SUB

Subtract the contents of address xx from the Accumulator

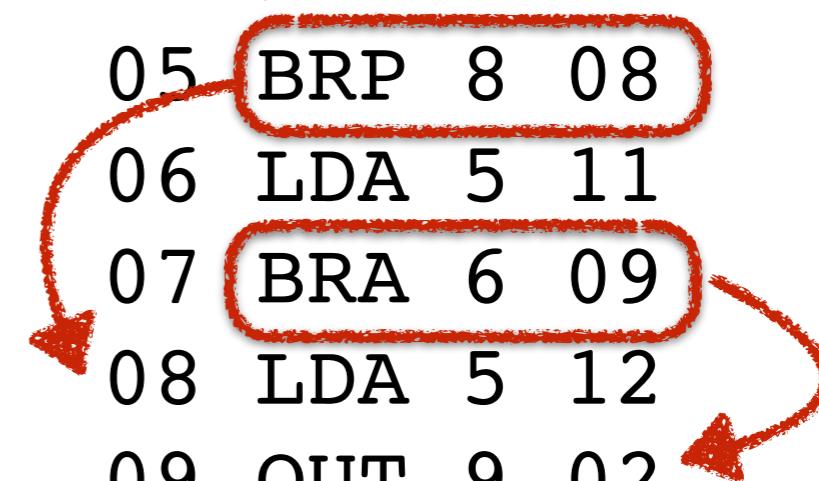
6xx BRA

Branch to xx - change program counter to xx

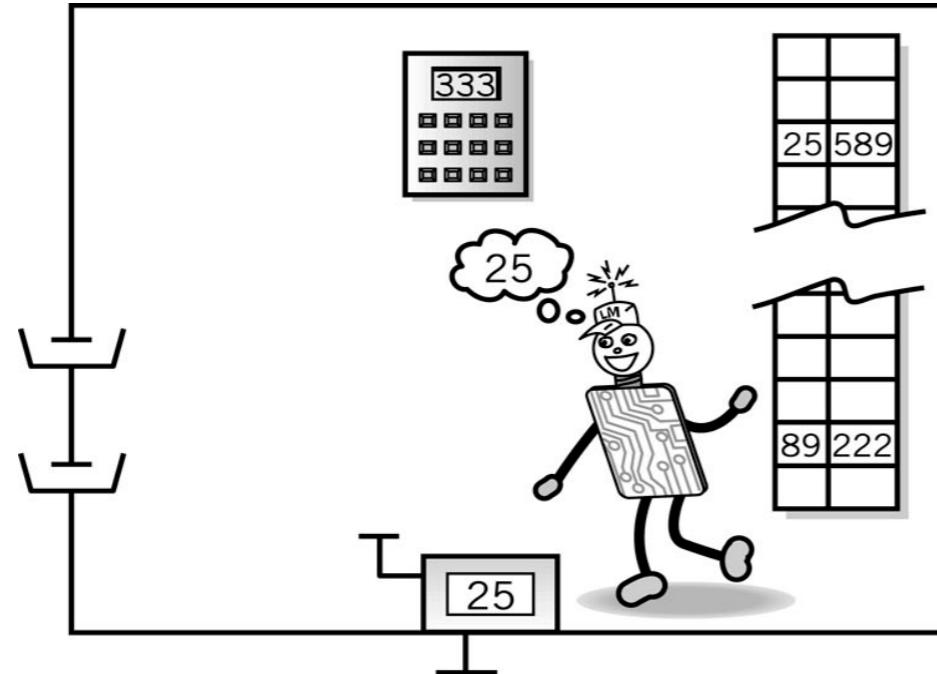
8xx BRP

Branch to xx if contents of Accumulator is positive

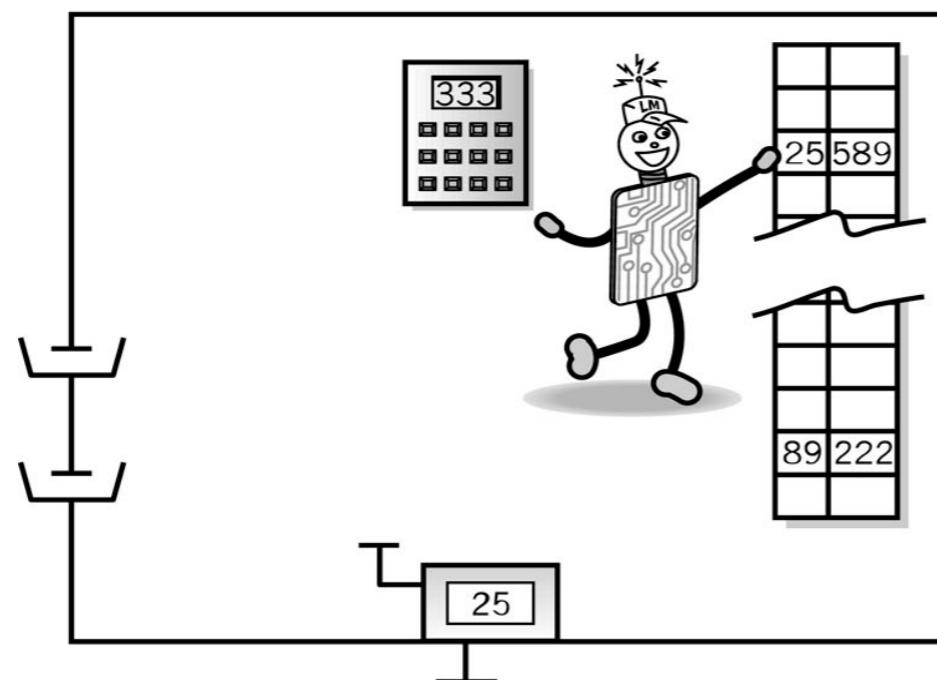
00	INP	9	01
01	STA	3	11
02	INP	9	01
03	STA	3	12
04	SUB	2	11
05	BRP	8	08
06	LDA	5	11
07	BRA	6	09
08	LDA	5	12
09	OUT	9	02
10	HLT	0	00
11	DAT	0	
12	DAT	0	



Little Man Computer

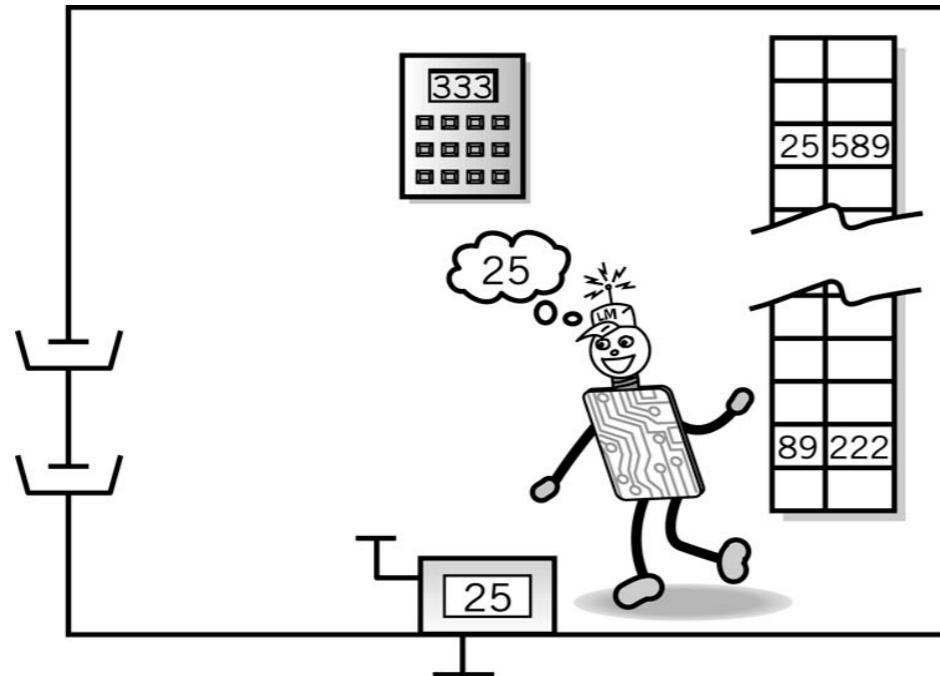


1. Little Man reads the address from the location counter

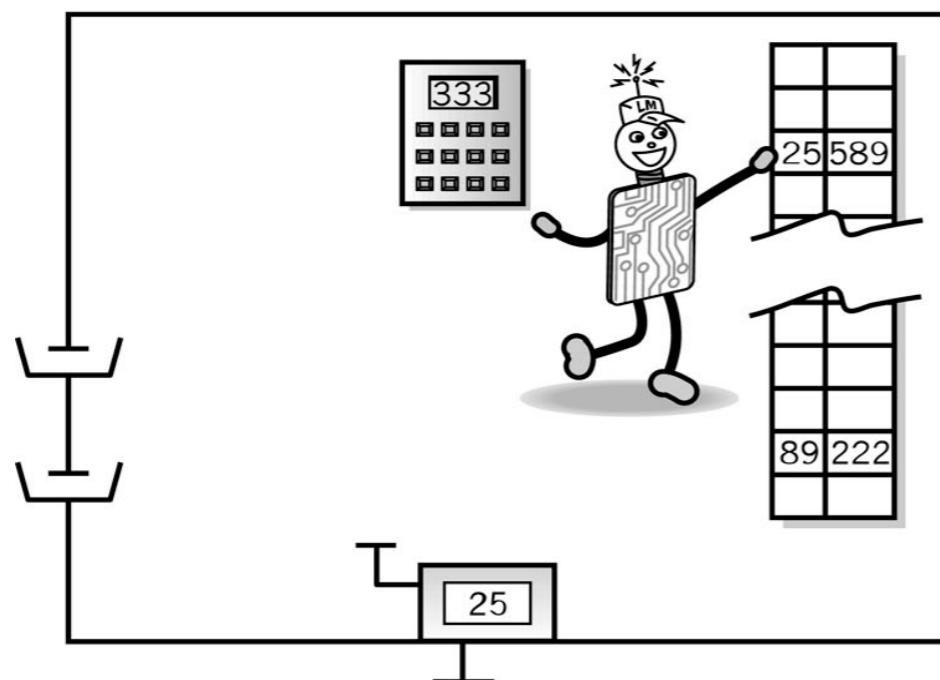


2. He walks over to the mailbox that corresponds to the location counter

FETCH - Little Man Computer

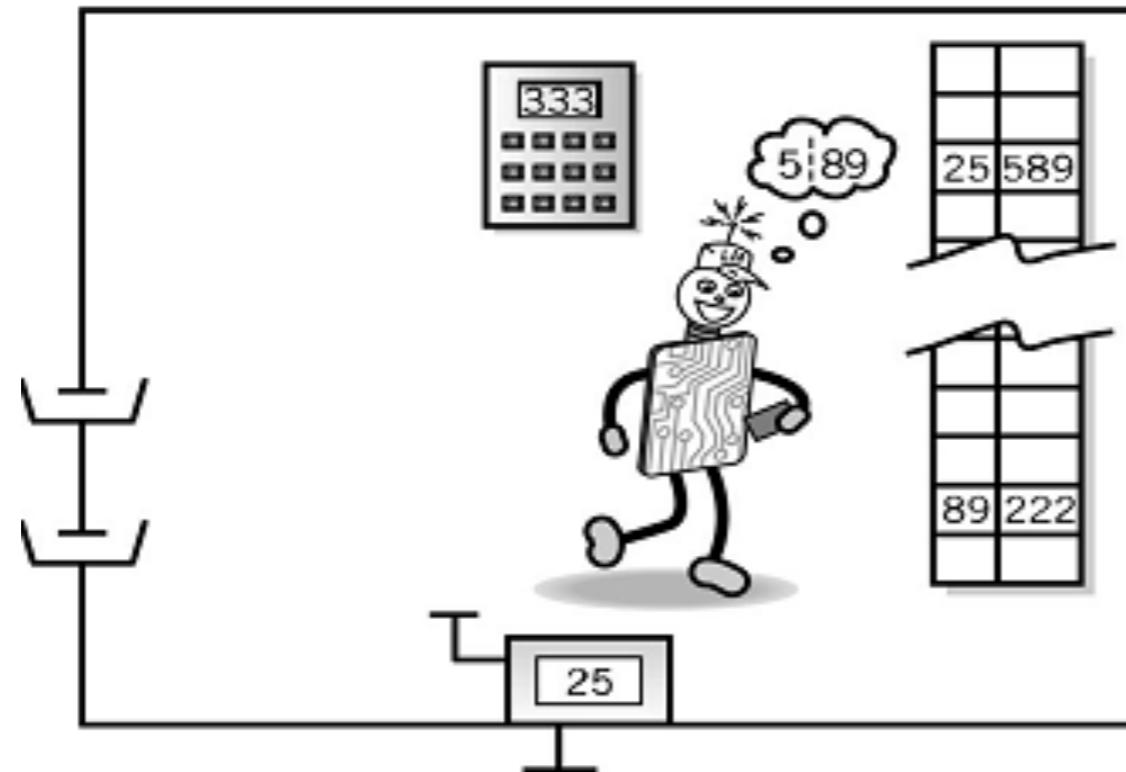


1. Little Man reads the address from the location counter



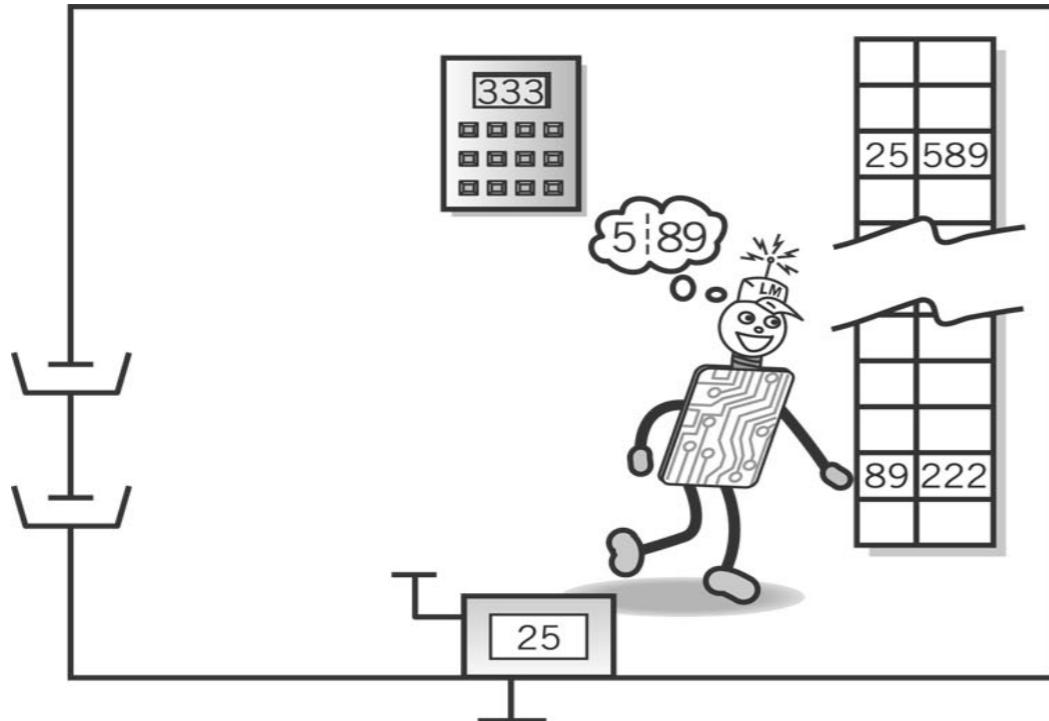
2. He walks over to the mailbox that corresponds to the location counter

FETCH - Little Man Computer



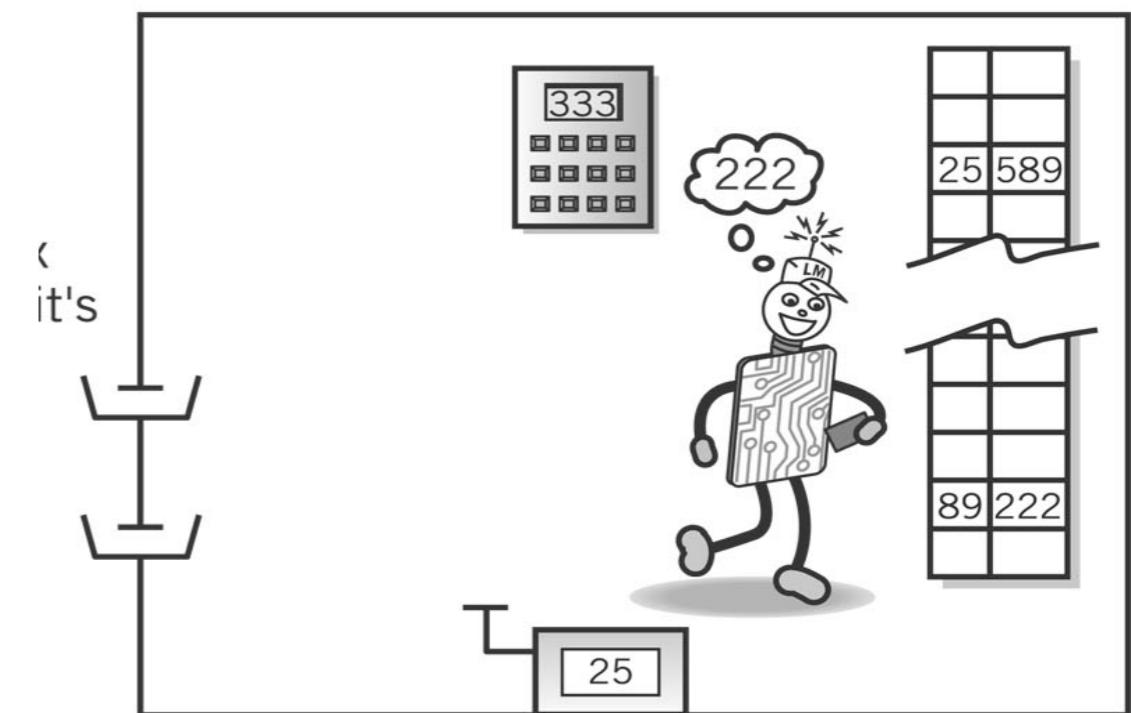
3. And reads the number on the slip of paper (he puts the slip back in case he needs to read it again later)

EXECUTE - Little Man

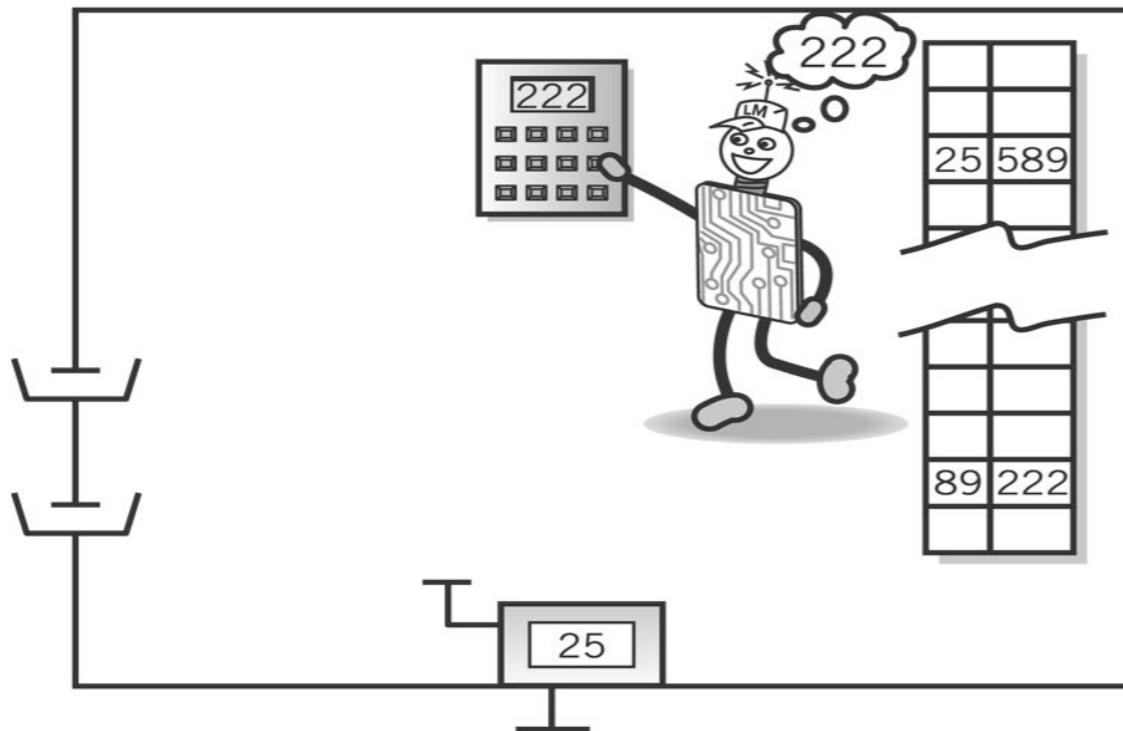


1. The Little Man goes to the mailbox address specified in the instruction he just fetched.

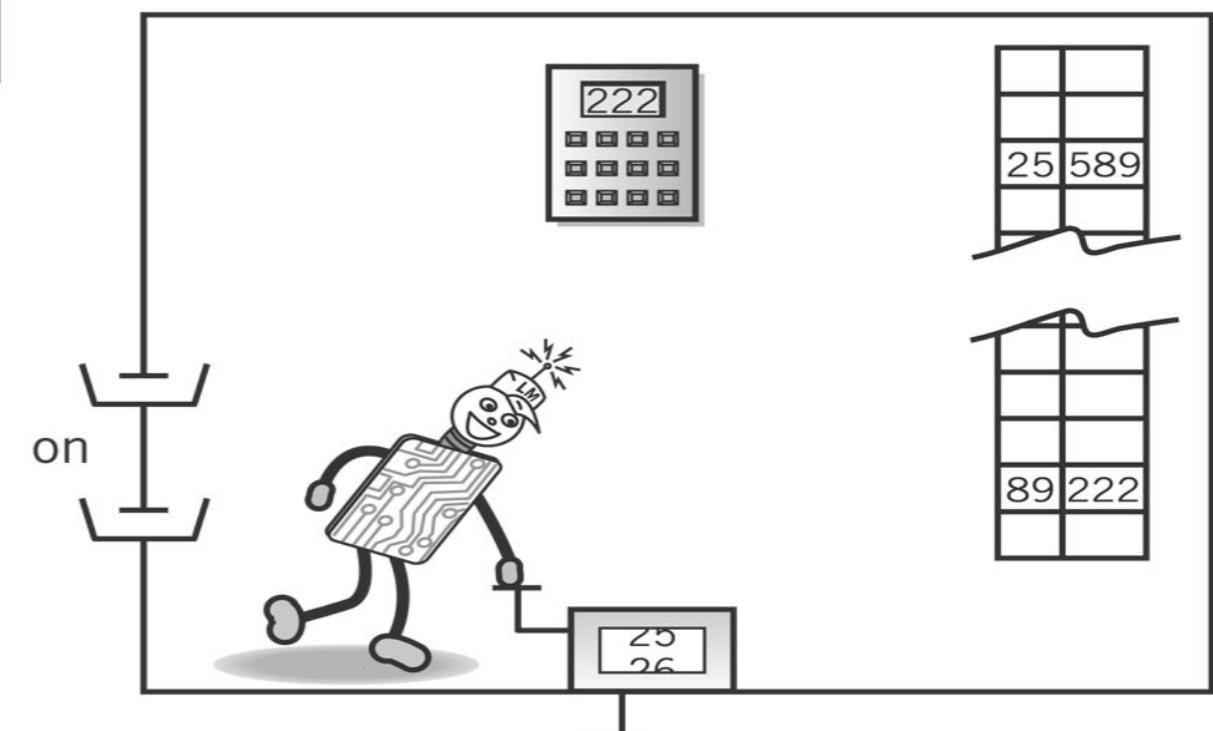
2. He reads the number in that mailbox (he remembers to replace it in case he needs it later).



EXECUTE - Little Man



3. He walks over to the calculator and punches the number in.



4. He walks over to the location counter and clicks it, which gets him ready to fetch the next instruction.

Simulate on LMC

<http://peterhigginson.co.uk/LMC/>

The screenshot shows the Little Man Computer simulation interface. On the left, the assembly language code is displayed:

```
INP    00 INP 9 01
STA 99 01 STA 3 99
INP    02 INP 9 01
ADD 99 03 ADD 1 99
OUT   04 OUT 9 02
HLT    05 HLT 0 00

// Output the sum of two
numbers
```

The CPU section shows the following registers:

- PROGRAM COUNTER: 06
- INSTRUCTION REGISTER: 0
- ADDRESS REGISTER: 00
- ACCUMULATOR: 009

The INPUT field contains the value 4.

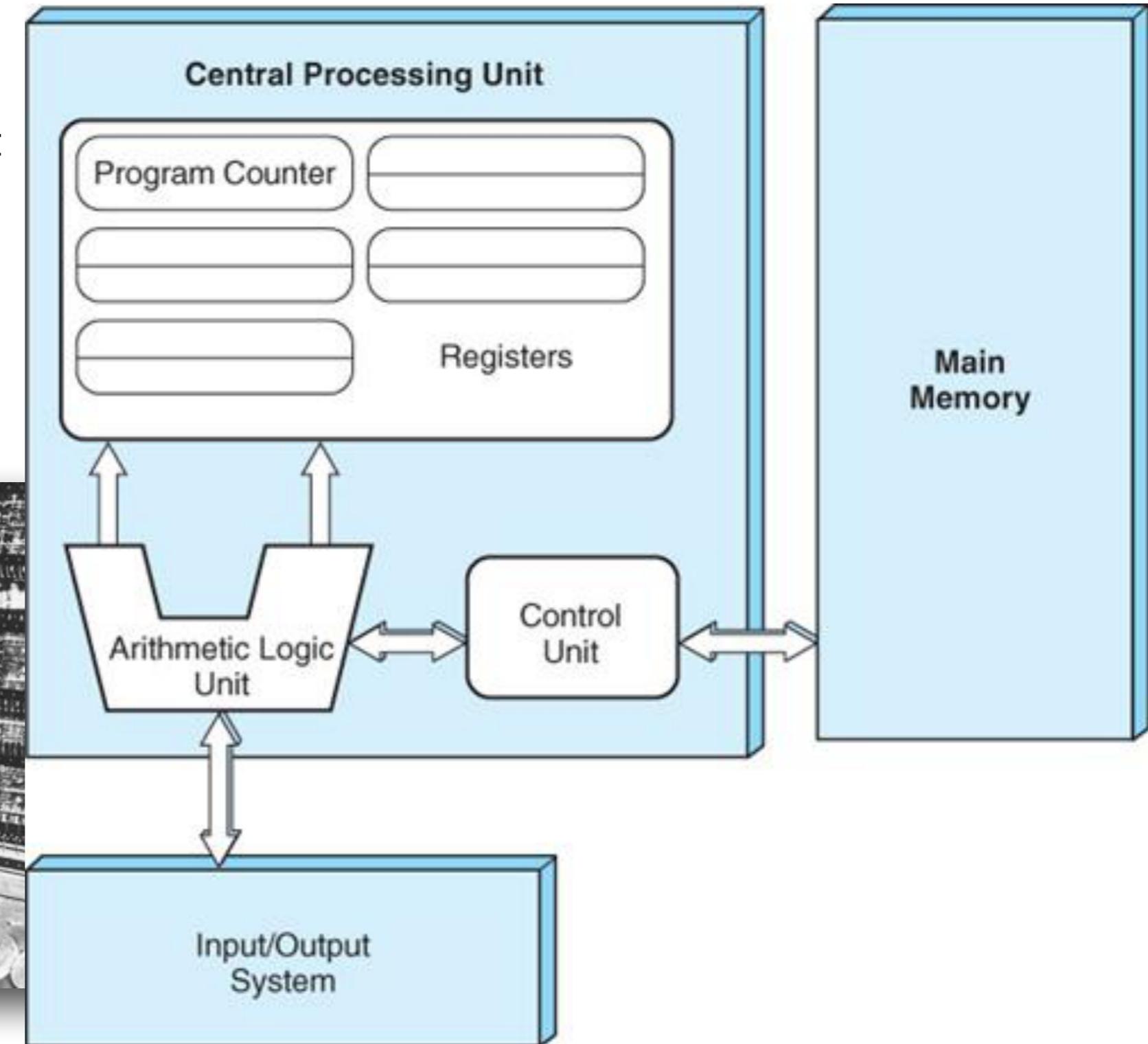
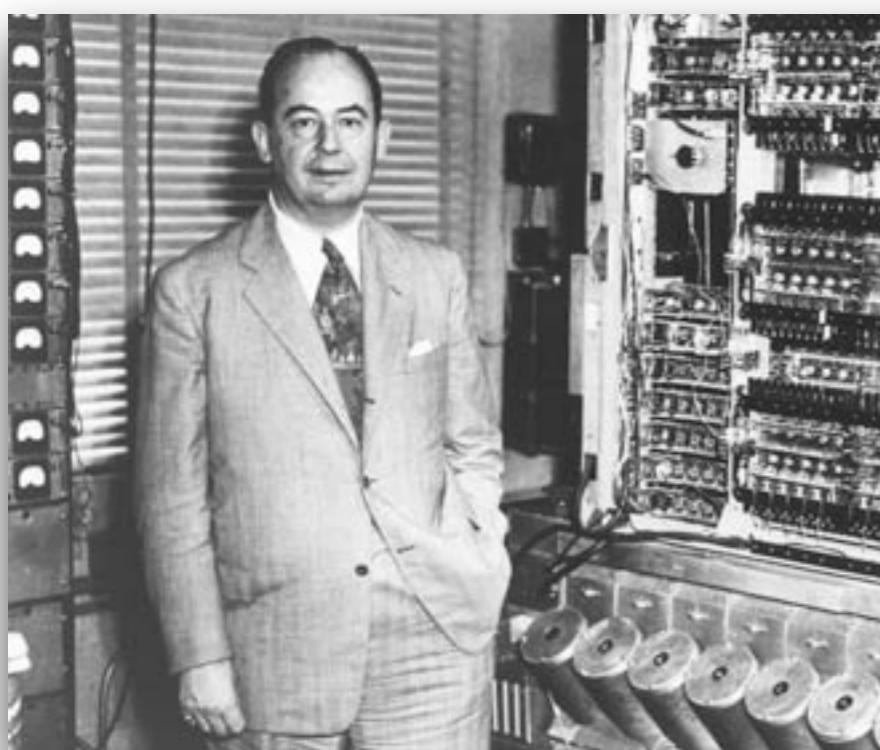
The OUTPUT field shows the value 9.

The RAM section displays memory locations from 0 to 99. A red circle highlights row 6 (locations 6-14), and another red circle highlights location 99 (row 99, column 99).

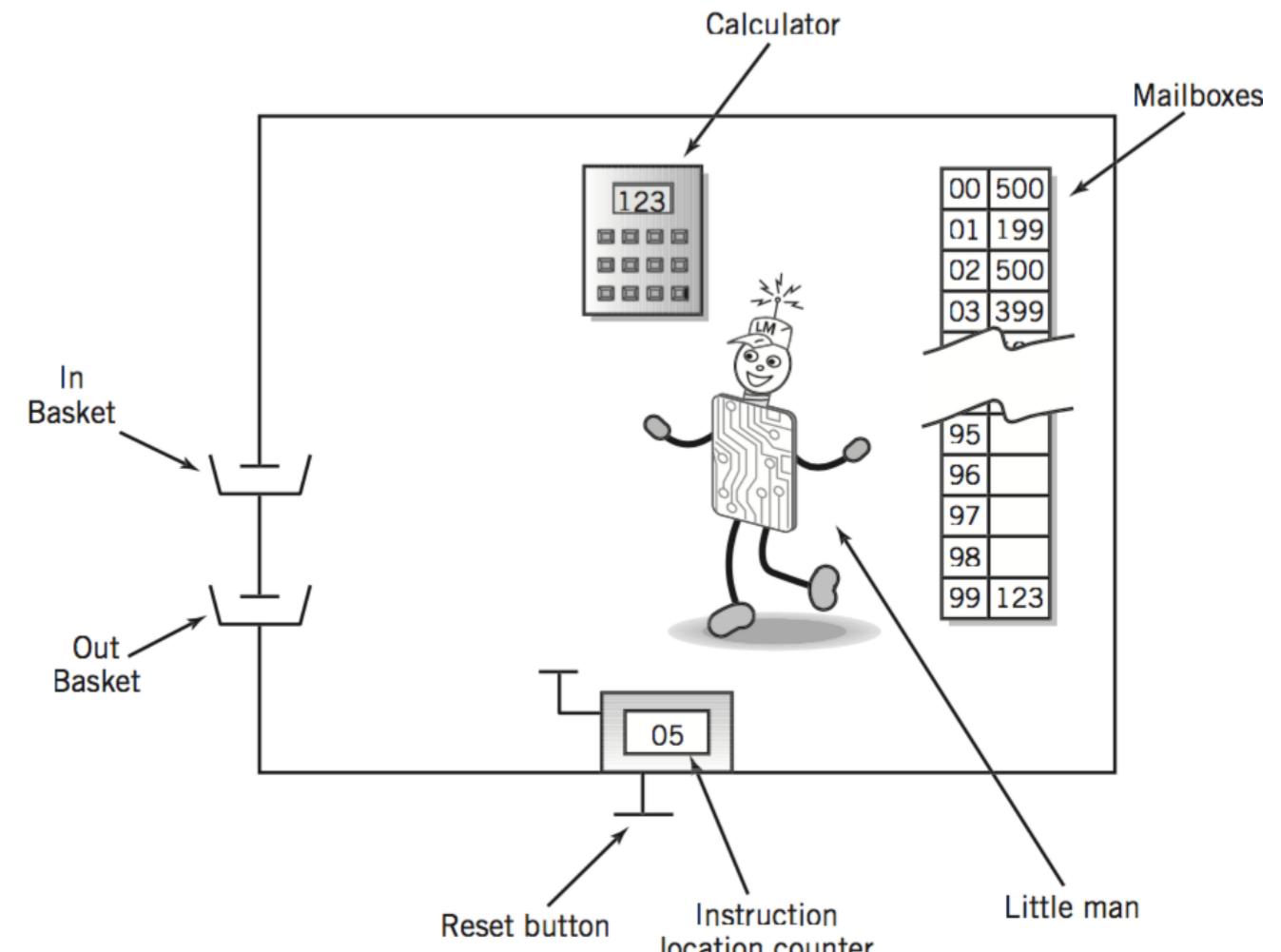
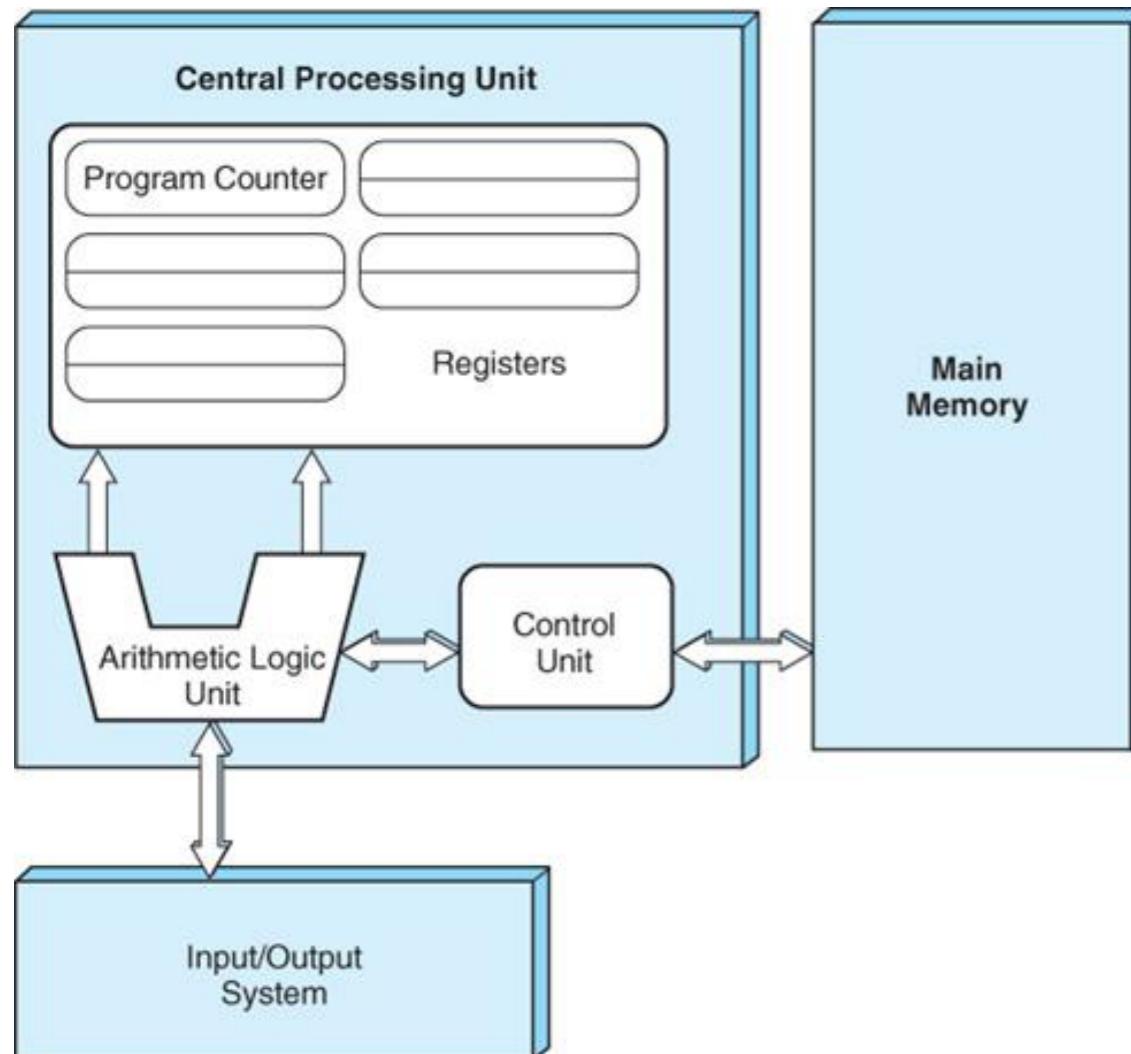
At the bottom, there are buttons for ASSEMBLE INTO RAM, RUN, STEP, RESET, LOAD, HELP, and a dropdown menu set to "add". A message box says "Program HALTED, RESET, LOAD, SELECT or alter memory". The footer includes "show op codes" and copyright information: "©GCSEcomputing.org.uk and Peter Higginson".

Von Neumann Architecture

Stored program concept
Memory is addressed linearly
Memory is addressed without regard to content



Little Man Computer



von Neumann Architecture

CPU

Control Unit

Arithmetic Logic Unit

Registers

- ▶ program counter
- ▶ instruction register
- ▶ address register

Main Memory

I/O System

LMC Cycle

Two stages:

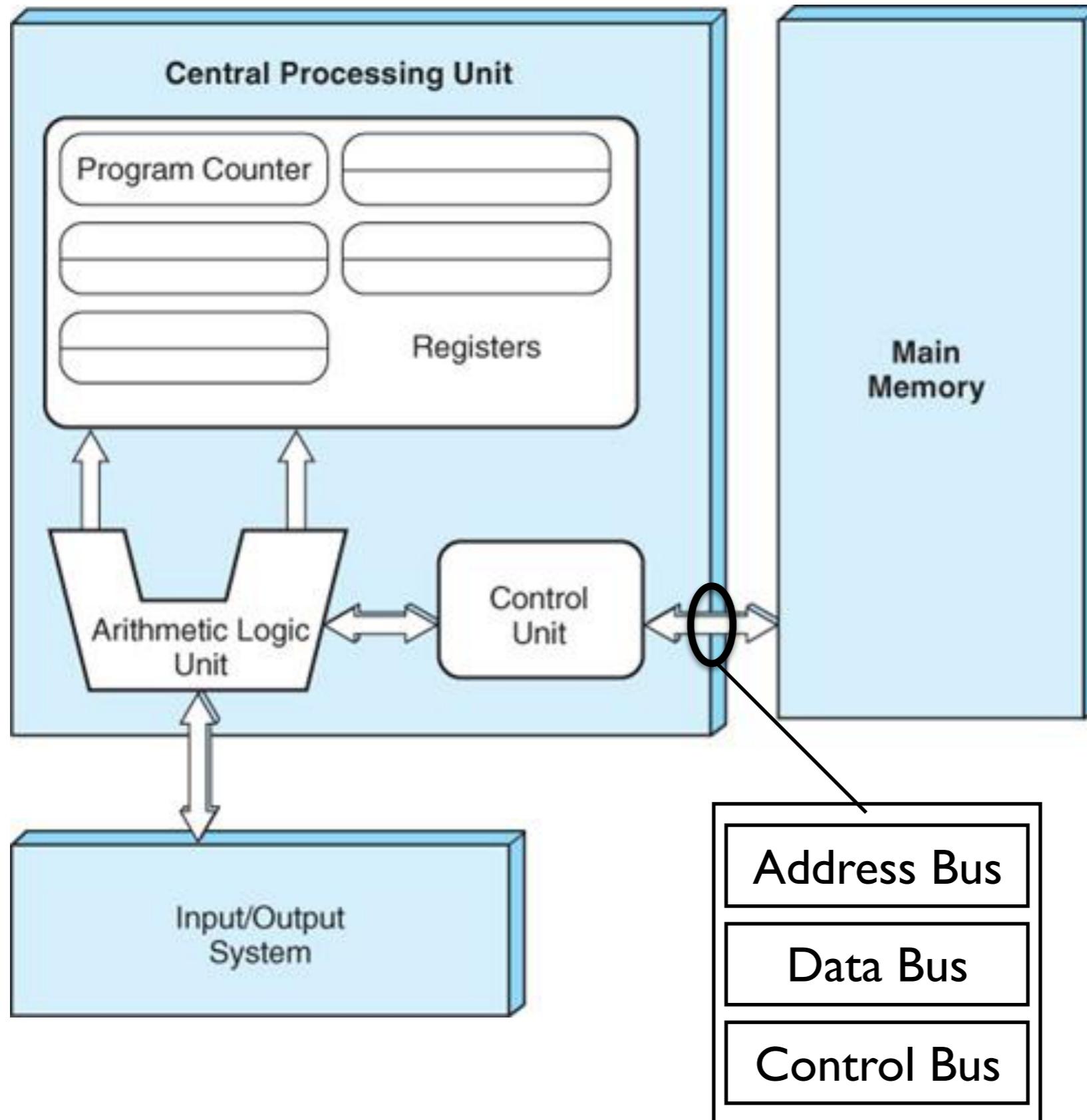
The *fetch* portion of the cycle, in which the Little Man finds out what instruction he is to execute,

The *execute* portion of the cycle, in which he actually performs the work specified in the instruction

Third stage

Decode stage after *fetch*

Busses



Bus width

Address bus

size of addressable memory

Data bus

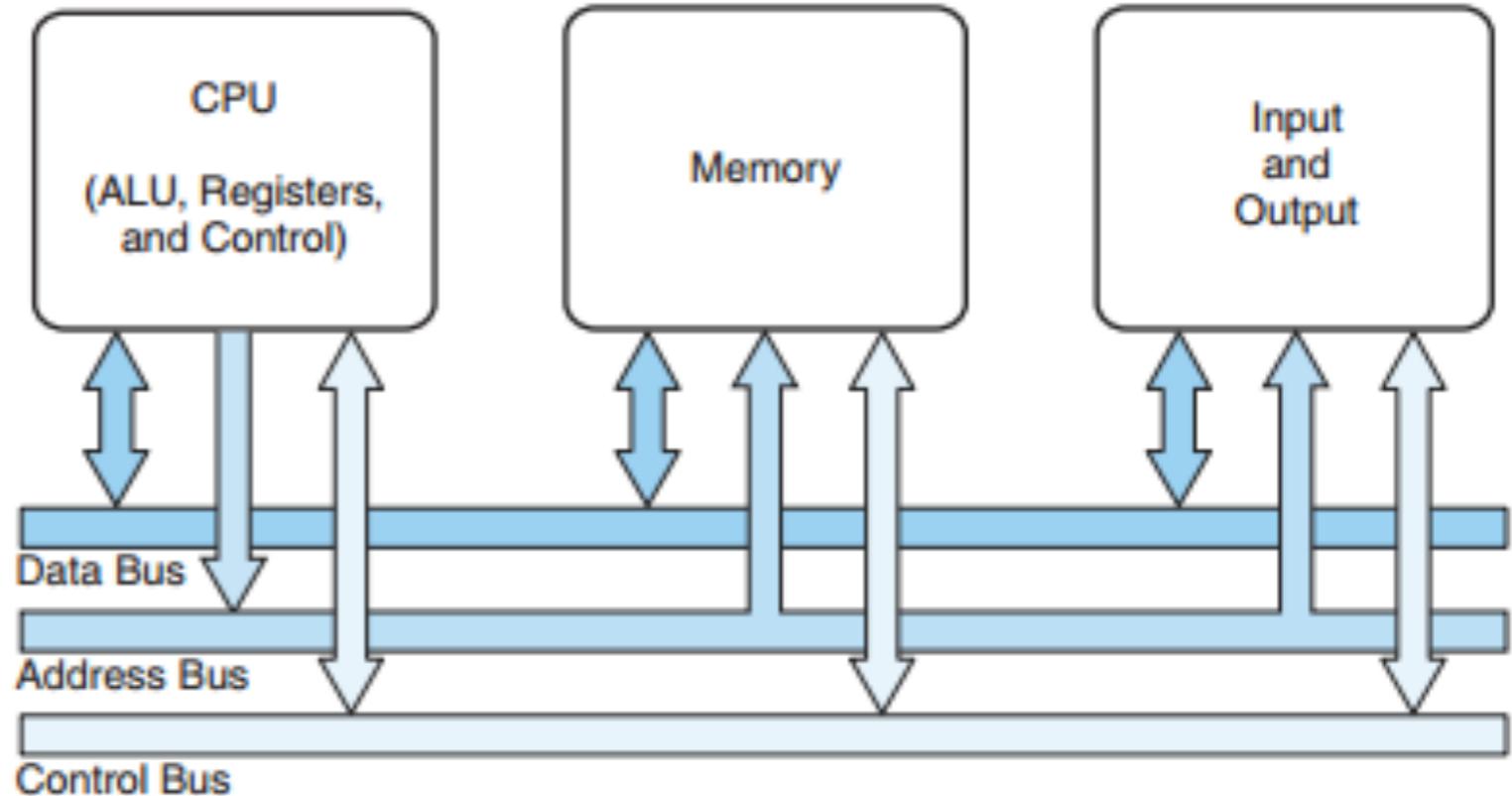
how much data can be fetched in one go

How does CPU work?

Data Bus moves data from main memory to the CPU registers

Address Bus holds the address of the data that the data bus is currently accessing.

Control Bus carries the necessary control signals that specify how the information transfer is to take place.



Addressing

Total Addressable Memory = ($2^{\text{address bus width}}$)

80386SX

16 bit data bus

24 bit address bus

2^{24} multiplied by 16 =
268,435,456 bits
Or 32 megabytes

Pentium

64 bit data bus

32 bit address bus

2^{32} multiplied by 64 =
274877906944 bits
or 32 gigabytes

LMC

What is the width of the address bus?

What is the width of the data bus?

The Processor

Learning Outcomes be able to:

explain the impact of transistor miniaturisation

describe the different levels of detail/abstraction in a processor

- ▶ i.e. transistor, logic gates, registers adders, CPU

explain the operation of a simple LMC program

describe the components in the von Neumann architecture

explain the role of the buses