INTRODUCTION TO COMPUTER SCIENCE

Dr. Yasmine El-Glaly

Fall 2013

Ch.2: Data Manipulation

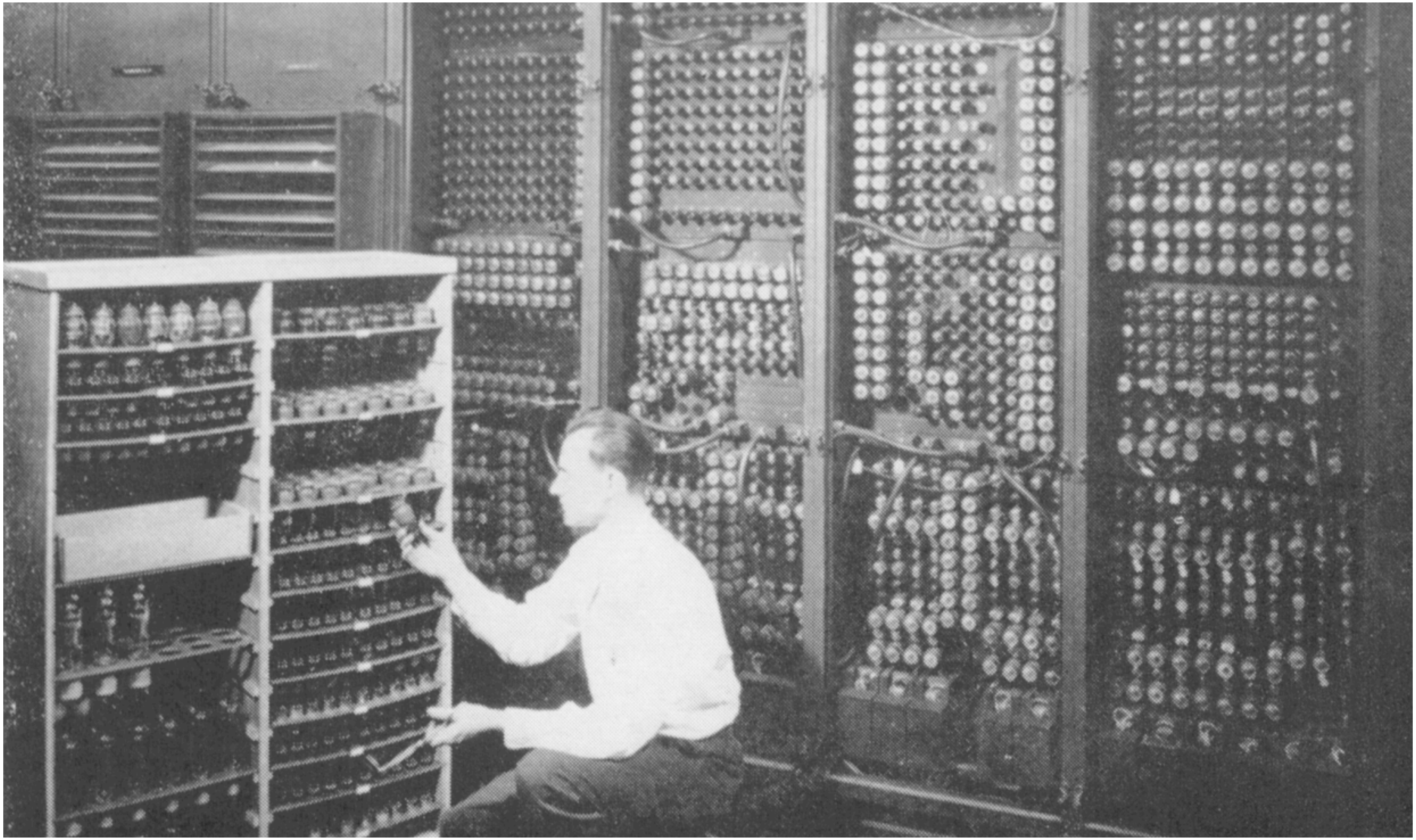
- Computer Architecture
- Machine Language
- Program Execution

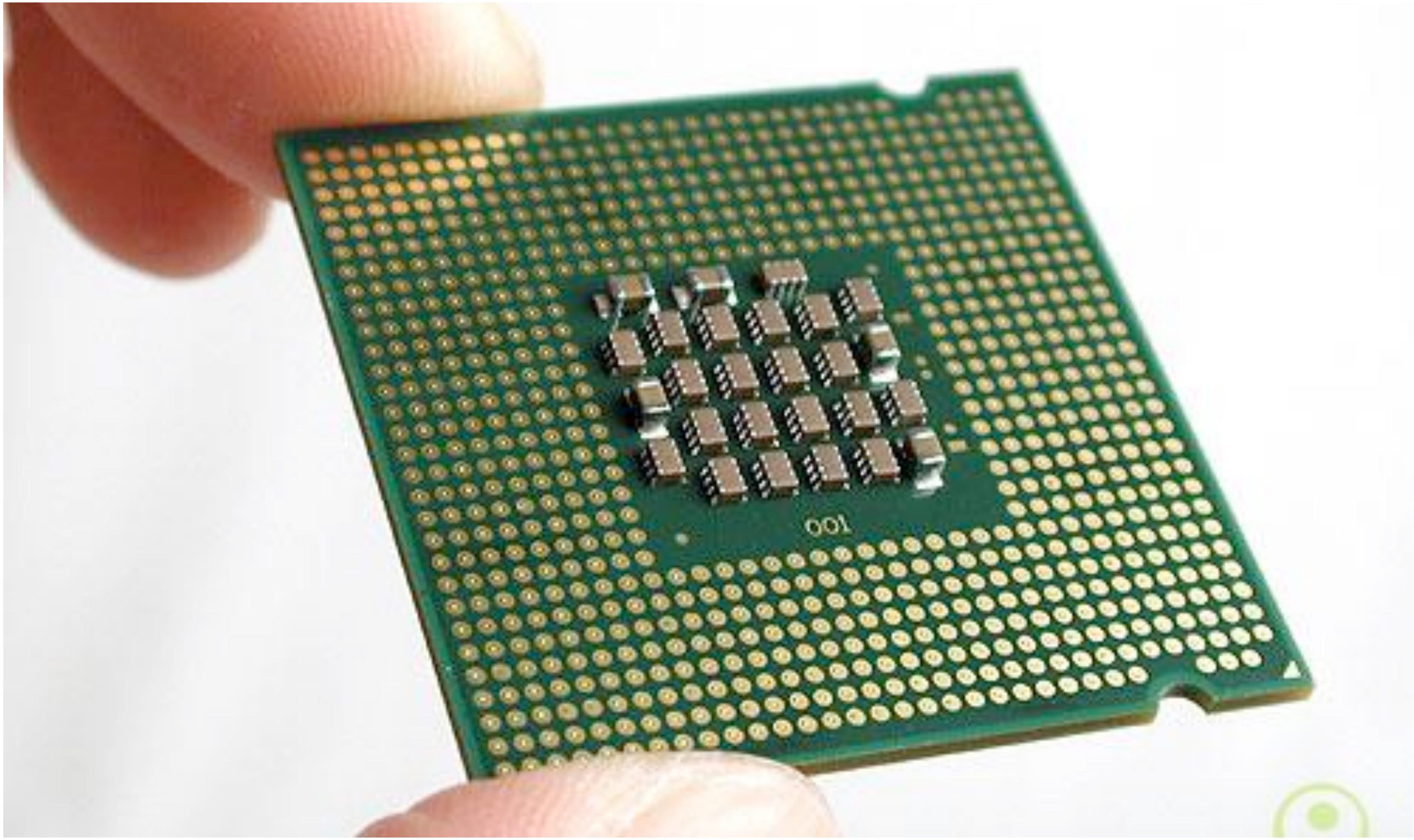
Computer Architecture

Machine Language

Program Execution

- The circuitry in a computer that controls the manipulation of data is called the **central processing unit,** or **CPU**
  - Whose connecting pins plug into a socket mounted on the machine’s main circuit board (called the **motherboard).**



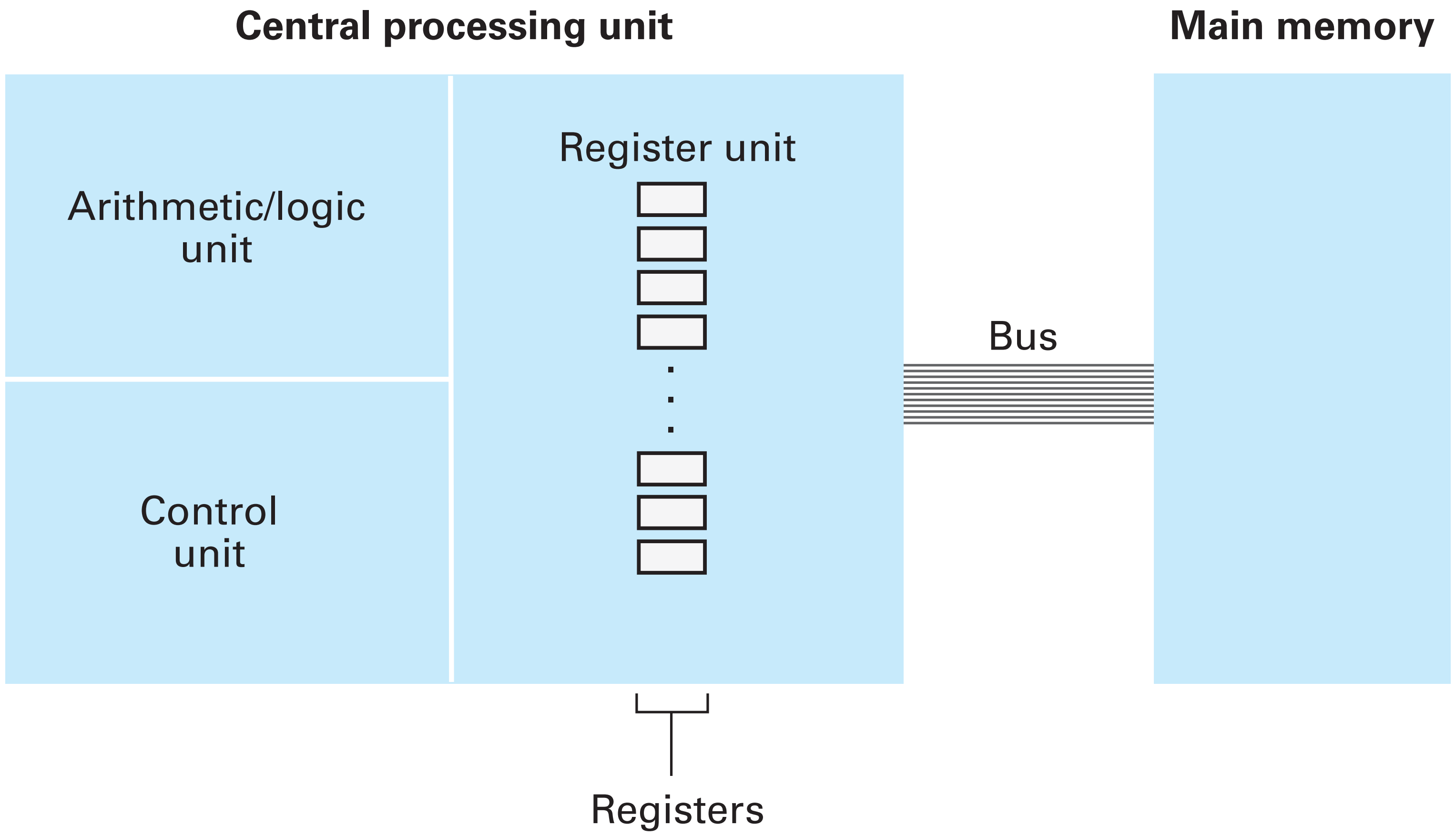


CPU Basics

- A CPU consists of
  - **arithmetic/logic unit (ALU),** which contains the circuitry that performs operations on data (such as addition and subtraction);
  - **control unit,** which contains the circuitry for coordinating the machine's activities;
  - **register unit,** which contains data storage cells (similar to main memory cells), called **registers,** that are used for temporary storage of information within the CPU.
    - Some of the registers within the register unit are considered **general-purpose registers** whereas others are **special-purpose registers.**

CPU Basics

- General-purpose registers serve as temporary holding places for data being manipulated by the CPU.



Computer Architecture

Machine Language

Program Execution

- Von Neumann architecture
  - architecture where program stored in memory

Computer Architecture

Machine Language
Program Execution

- Adding 2 values stored in memory:

  1. Get first value in a register
  2. Get second value in a register
  3. Add results in ALU — result in a register
  4. Store result in memory (or a register)

Computer Architecture

Machine Language

Program Execution

- CPUs are designed to recognize instructions encoded as bit patterns.
- This collection of instructions along with the encoding system is called the **machine language.**

Computer Architecture

Machine Language

Program Execution

- CPU that executes a minimal set of machine instructions is called a **reduced instruction set computer (RISC)**
  - efficient, fast, and less expensive
- CPUs that execute a large number of instructions, even though many of them are technically redundant, leads to **complex instruction set computer (CISC)**
  - better cope with the ever increasing complexities of today’s software.

Computer Architecture

Machine Language

Program Execution

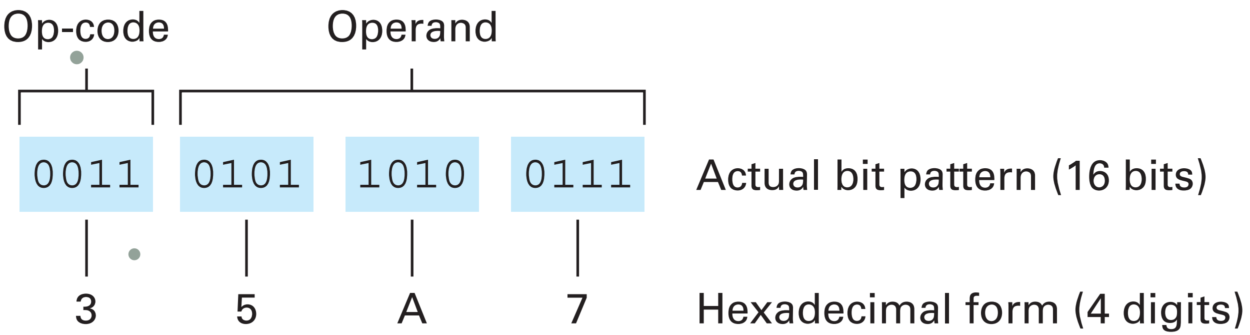
- Machine’s instructions can be categorized into 3 groupings:
  - (1) the data transfer group,
    - instructions that request the movement of data from one location to another
    - E.g. *LOAD, STORE*
  - (2) the arithmetic/logic group, and
    - Add, Boolean operations AND, OR, and XOR
  - (3) the control group.
    - instructions that direct the execution of the program rather than the manipulation of data
    - E.g. JUMP

Computer Architecture

Machine Language

Program Execution

- Example machine language — Appendix C Instruction:
  - 4 bits op-code
  - 12 bits operands



Op-code

Operand

0011 0101 1010 0111

Actual bit pattern (16 bits)

3 5 A 7

Hexadecimal form (4 digits)

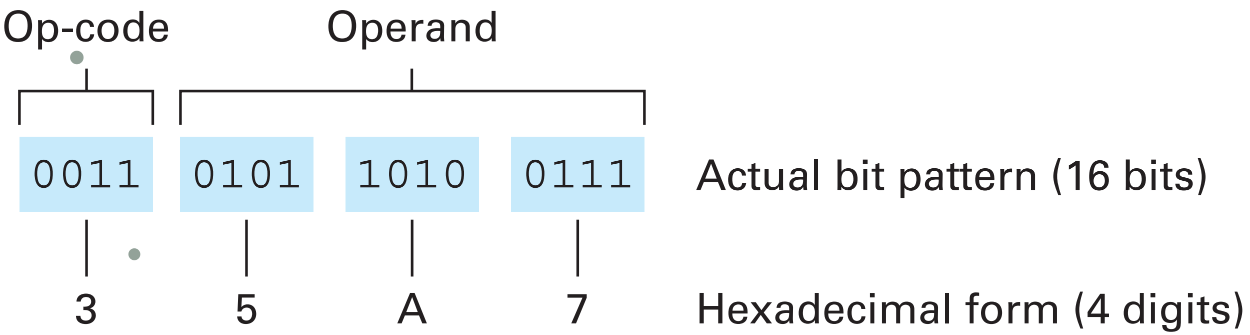
- How many general purpose registers are there?
- A. 4 B. 8 C. 12
- D. 16 E. 32

Computer Architecture

Machine Language

Program Execution

- Example machine language — Appendix C Instruction:
  - 4 bits op-code
  - 12 bits operands



- How many general purpose registers are there?
- A. 4 B. 8 C. 12
- D. 16 E. 32

Example machine language

Instructions:

Op-code	Operands	Meaning
1	RXY	Load reg R from memory cell XY
2	RXY	Load reg R with value XY
3	RXY	Store contents of reg R in cell XY
4	$0RS$	Move contents of reg R to reg S
5	RST	Add two's compl. contents of reg S to reg T; store result in R
6	RST	Foating point add
7	RST	OR
8	RST	AND
9	RST	XOR
A	$R0X$	Rotate reg R X bits to right
B	RXY	Jump to XY if $c(R) = c(0)$
C	000	HALT

Note operands are hexadecimal.

Example machine language

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.

Computer Architecture

Machine Language

Program Execution

- A computer follows a program stored in its memory by copying the instructions from memory into the CPU as needed.
- Once in the CPU, each instruction is decoded and obeyed
- execution process
  - **instruction register** and the **program counter**

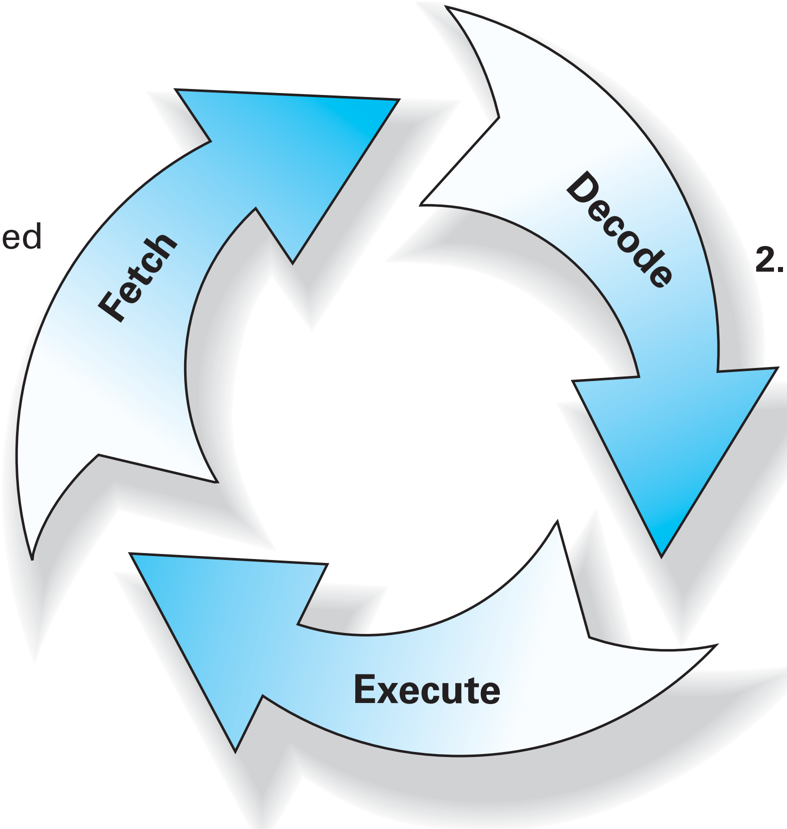
Computer Architecture

Machine Language

Program Execution

- The CPU performs its job by continually repeating an algorithm that guides it through a three-step process known as the Machine cycle:
  - fetch — get next instr., increment program counter by 2
  - Decode
  - execute (instr)

The Machine Cycle



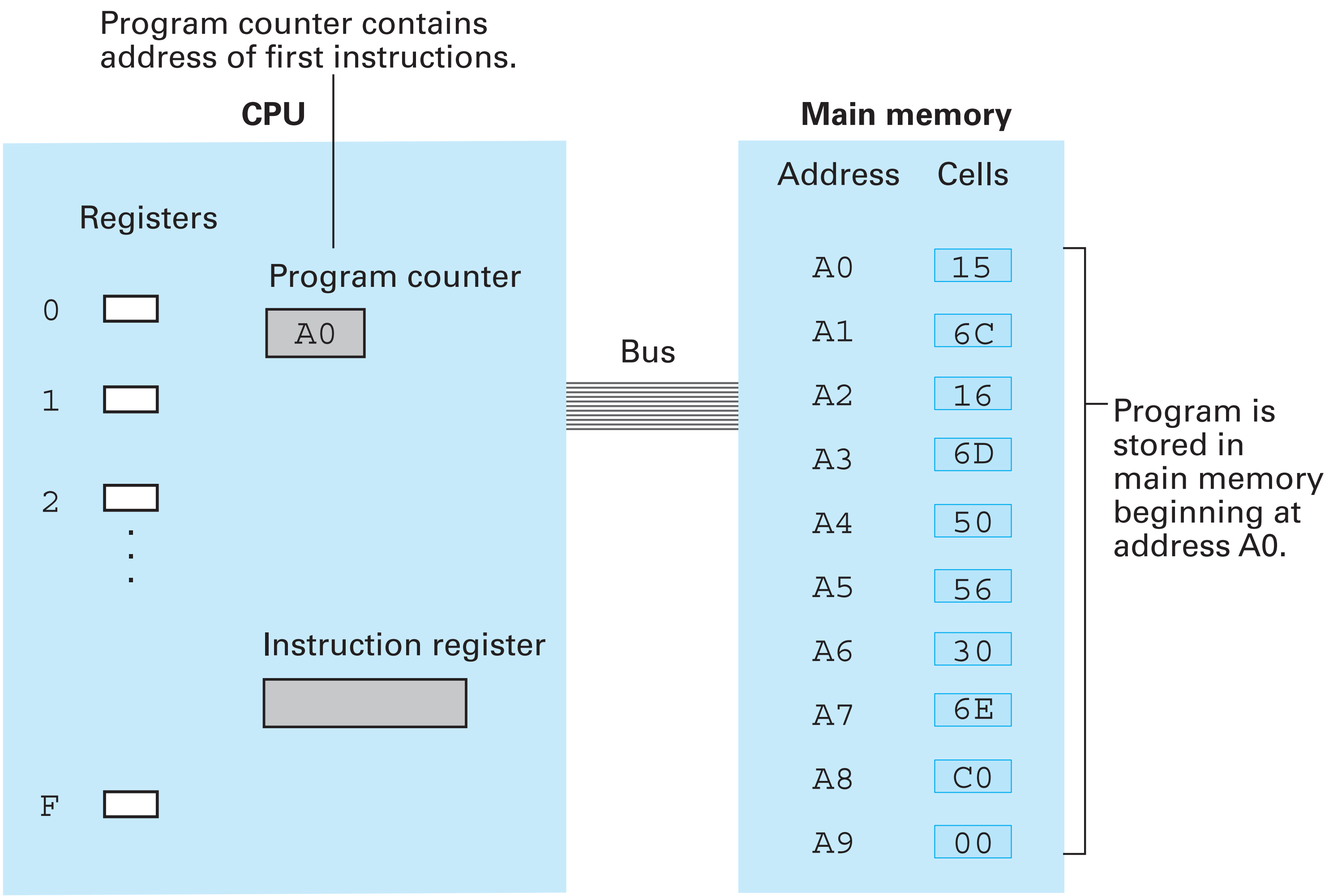
1. Retrieve the next instruction from memory (as indicated by the program counter) and then increment the program counter.

2. Decode the bit pattern in the instruction register.

3. Perform the action required by the instruction in the instruction register.

An Example of Program Execution

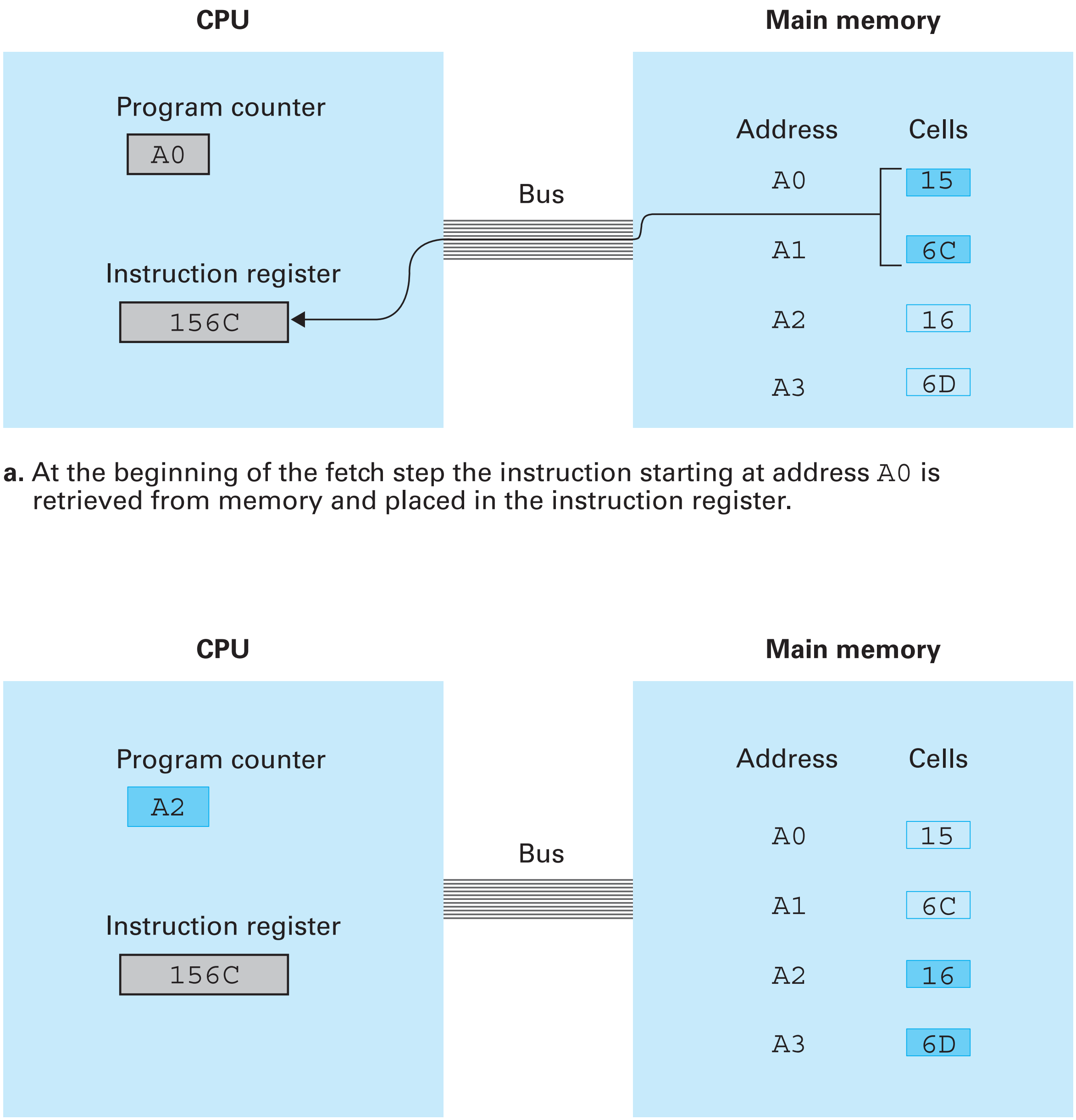
Figure 2.10 The program from Figure 2.7 stored in main memory ready for execution



At the end of the fetch step of the first machine cycle, the program counter and instruction register contain the following data:

Program Counter: A2
Instruction Register: 156C

Figure 2.11 Performing the fetch step of the machine cycle



a. At the beginning of the fetch step the instruction starting at address A0 is retrieved from memory and placed in the instruction register.

b. Then the program counter is incremented so that it points to the next instruction.

Assignment

- Read chapter 2 to end of 2.3 (page 90)
- Report a summary of your reading
- Quiz next week

Thanks!