The Central Processing Unit

Chapter 3
What Goes on Inside the Computer

Recall that there are four main functions of computer systems:
- Input
- Processing
- Output
- Secondary Storage

In this chapter, we will focus on the central processing unit (CPU) in more detail.

The CPU interacts closely with memory (primary storage).
Memory, however, is not part of the CPU.

The CPU consists of a variety of parts including:
- Control unit
- Arithmetic/logic unit (ALU)
- Registers

The control unit directs the other parts of the computer system to execute stored program instructions.
The control unit communicates with the ALU and memory.
The Arithmetic/Logic Unit (ALU)...

performs mathematical operations as well as logical operations.

Mathematical Operations

The ALU can perform four kinds of mathematical calculations:

- addition
- subtraction
- multiplication
- division

Logical Operations

The ALU can perform logical operations.

Logical operations can test for these conditions:

- Equal-to (=)
- Less-than (<)
- Greater-than (>)

Equal-to Condition

In a test for this condition, the ALU compares two values to determine if they are equal.

If \[ \begin{array}{c} \text{coin} \\ \text{coin} \end{array} = \begin{array}{c} \text{coin} \\ \text{coin} \end{array} \]

Then \[ \begin{array}{c} \text{coin} \\ \text{coin} \end{array} = \begin{array}{c} \text{coin} \\ \text{coin} \end{array} \]

Less-than Condition

In a test for this condition, the ALU compares values to determine if one value is less than another.

If \[ \begin{array}{c} \text{coin} \\ \text{coin} \end{array} = \begin{array}{c} \text{coin} \\ \text{coin} \end{array} \]

Then \[ \begin{array}{c} \text{coin} \\ \text{coin} \end{array} < \begin{array}{c} \text{coin} \\ \text{coin} \end{array} \]

Greater-than Condition

In a test for this condition, the ALU compares values to determine if one value is greater than another.

If \[ \begin{array}{c} \text{coin} \\ \text{coin} \end{array} = \begin{array}{c} \text{coin} \\ \text{coin} \end{array} \]

Then \[ \begin{array}{c} \text{coin} \\ \text{coin} \end{array} > \begin{array}{c} \text{coin} \\ \text{coin} \end{array} \]
Registers...

Data held temporarily in registers can be accessed at greater speeds than data stored in memory.

Memory (Primary Storage)

Memory is the part of the computer that stores data and program instructions for processing.

Memory...

is also referred to as RAM (random-access memory).

RAM is temporary, finite, and more expensive than secondary storage.

Executing Program Instructions

Before the CPU can execute a program, program instructions and data must be placed into memory from an input device or storage device.

Fetching Instructions

The control unit fetches (gets) the instruction from memory.

Executing Program Instructions

Once the necessary data and instructions are in memory, the CPU performs the following steps for each instruction:

• Fetching
• Decoding
• Executing
• Storing
Decoding Instructions

The control unit decodes the instruction and directs that the necessary data be moved from memory to the ALU.

Executing Arithmetic/Logic Operations

The ALU performs the arithmetic or logical operation on the data.

Storing Results

The ALU stores the result of its operation on the data in memory or in a register.

Executing Program Instructions

Eventually, the control unit sends the results in memory to an output device or secondary storage.

Instruction Time

The time it takes to fetch an instruction and decode it is called instruction time.

Execution Time

The time it takes to execute an ALU operation and then store the result is called execution time.
Machine Cycle

The combination of I-time and E-time is called the machine cycle.

I-time + E-time = Machine Cycle

Memory Locations and Addresses

The control unit can find data and instructions because each location in memory has an address.

Control Unit

Memory

Storage Locations

Each location in memory is identified by an address.

Each location has a unique address.

Symbolic Addresses

The choice of the location in memory is arbitrary.

Addresses can only hold one number or word.

Data Representation

The system in which all computer data is represented and manipulated is called the binary system.

Binary System

The binary system has only two digits to represent all values.

This corresponds to the two states of a computer’s electrical system—on and off.
Off/On Switches
The computer can represent data by constructing combinations of off or on switches.

Zero or One?
The binary system can also be represented by the digits zero and one.

The Bit
Each 0 or 1 in the binary system is called a bit.

The Byte
A group of 8 bits is called a byte.

One Character of Data
Each byte represents one character of data (a letter, digit, or special character).

Storing Bytes
Storage and memory capacity is expressed in the number of bytes they can hold:

- 1 kilobyte = $2^{10}$ or 1024 bytes
- 1 megabyte = $2^{20}$ or 1,048,576 bytes
- 1 gigabyte = $2^{30}$ or 1,073,741,824 bytes
Computer Word

A computer word is defined as the number of bits that constitute a common unit of data.

Computer Word Length

Word length varies by computer. For example:
8 bits = 1 byte = one word length
64 bits = 8 bytes = one word length

Coding the Computer

A code for determining which group of bits represent which characters on a keyboard is called ASCII.

ASCII

ASCII has been adopted as the standard, by the U.S. government and is found in a variety of computers.

\[ 01001010 = J \]

ASCII-8 code Keyboard character

Inside the Computer

The flat board within the personal computer that holds the computer circuitry is the motherboard.

The Motherboard

The motherboard includes:
- Microprocessor
- Memory components
The Microprocessor...

is the CPU of the personal computer.

It looks like a computer chip.

The microprocessor contains tiny electronic switches (transistors).

Microprocessor Switches

If electric current passes through a transistor, the switch is on.

If no current can pass, the switch is off.

A combination of transistors can stand for a combination of bits.

Memory...

components are also chips.

This form of semiconductor memory stores data in binary form (off/on or zero/one).

Storage can be temporary (RAM) or permanent (ROM).

RAM...

means random-access memory.

RAM chips can be accessed easily and fast—regardless of where the data is stored.

RAM is usually volatile storage in that if the power is shut off, the contents are lost.

ROM...

is read-only memory.

ROM chips contain programs and data that are permanently etched on the chip.

The contents do not disappear when the power is turned off.

Computer Speed and Power

Speed and power are determined by:

- Microprocessor speed
- Bus lines
- Cache
- Flash memory
- RISC
- Parallel processing
Microprocessor Speeds

Microprocessor speeds can be measured in a variety of ways:
- Megahertz
- MIPS
- Megaflops

Megahertz

One measure of microprocessor speed is megahertz (MHz) which is one million machine cycles per second.

MIPS

Another measure of microprocessor speed is MIPS which is one million instructions per second.

Megaflops

Megaflops, or one million floating-point operations per second, is still another measure of microprocessor speed.

Bus Lines

A bus line is a set of parallel electrical paths. A bus is like a mode of transportation for data.

Bus Width

The amount of data that can be carried at one time is bus width (wider = more data).
Cache

Cache is a relatively small block of very fast memory. The data and instructions stored in cache are those that are most recently or most frequently used. Cache speeds up the internal transfer of data and software instructions.

Flash Memory

Accessing data and instructions from within the CPU is fast but not when it comes from a secondary storage device. Flash memory is nonvolatile (like secondary storage) but allows fast access (like RAM).

RISC

New RISC computers have chips with fewer instructions as a means of making them run faster. Older, slower machines have chips (called CISC) with instructions that are seldom or never used.

Parallel Processing

Using several processors at the same time (in parallel) greatly increases processing speed. When parallel processing, the computer can be starting other tasks before the sequence of fetch-decode-execute-store is complete.

Conclusion

Regardless of the design and processing strategy of a computer, its goal is the same: to turn raw data into useful information.