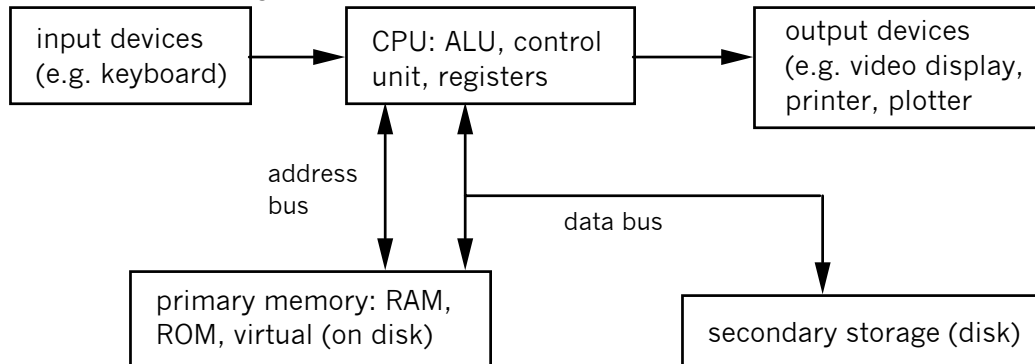


Basic design of a computer system: In a digital computer, numbers are represented by discrete numerals coded in binary arithmetic. In this system, 0s and 1s encode all data, the locations of data, and the control codes (the instructions that make up the computer program). The circuitry in the computer includes, simply put, sets of switches that can be set rapidly to on or off states (equivalent to 1s or 0s, or voltages set high or low).

A first look at the organization of a computer.



Some terminology (see the glossary for more complete definitions):

Memory: this is electronic circuitry capable of storing many bits of information.

Bit: is a single binary digit, a single 0 or a 1.

Binary numbers: this is a number system to the base 2, using only the digits 0 and 1. Counting in binary from 0 to the equivalent of 10 in decimal progresses as follows: 0,1,10,11,100,101,110,111,1000,1001,1010.

Word: a word is a packet of binary digits stored in memory. The word size (equal to the number of digits in the packet) is characteristic of a computer's complexity and 'size': the 8 bit computers of the late 1970s had an 8 bit word length.

Byte: this is the most universally used 'packet' of bits processed by computer. An 8 bit word length computer uses words 1 byte in length; a 16 bit machine word has 2 byte packets, and a 32 bit word (common PCs of today) is 4 bytes; some newer computers (the Macintosh G5) use a 64 bit word of 8 byte length.

Address: this is the word or words specifying the numeric location of some data word stored elsewhere in memory. Locations in memory have sequential addresses, from 0 to the highest allowed address (this depends on how much memory is installed in, and is addressable by, the computer). In the early 80's, the Apple IIe, a 64K machine, had possible decimal addresses from 0 to 65,535. Early Macintoshes could address 2^{24} (8 MB) memory locations; later machines were able to access 2^{32} (2 GB) memory locations.

What does a word value stored in memory "mean"? The meaning depends entirely on its context. A word can encode either (1) numeric data (perhaps the numeric code representing an alphanumeric character like "A" or a "\$" symbol), or (2) a computer instruction to be interpreted by the machine, commanding it to take some action like "ADD", or (3) the address of some other memory location being referenced.

What is a computer "instruction?" It is a numerically coded command to be interpreted by computer's electronic circuits in its "CPU".

CPU: this is the Central Processor Unit; the computing circuitry, consisting of electronic logic circuits and registers; the CPU interprets (decodes), then executes arithmetic and logic instructions. Common CPUs include the Pentium IV and the Macintosh 603.

Register: this is a set of switches within the CPU, usually of 1 or 2 word length. Registers are used to accumulate intermediate results (in the early days of computing, a processor's registers were called 'accumulators'), and to serve as a "scratch pad" for calculations in progress.

Program: this is the sequence of instructions, coded in binary, to be executed by the computer; the computer must be carefully instructed where to begin looking at the contents of memory locations and to begin interpreting the contents of those locations as instructions to be executed.

Cycle time: this is the interval required by computer to fetch information from memory or perform one defined CPU operation like obtaining the contents of a memory location: these numbers continue to decrease for modern computers. A workstation like the Compaq alpha series can perform at the rate of several Gigaflops (= billion times a second), so a 4 gigaflop machine cycle time is 0.25 nsec.

I/O bus (or buss): this is the physical wiring in the computer box that allows transfer of binary coded information to and from memory and peripheral devices. Various bus formats (e.g. the modern PC's PCI bus) are currently in use, and continue to achieve ever faster transfer speeds.