

Week 2: System basics

Week 2: Processor internals

Processor internals

The most primitive piece of a computer's processing functionality is the [logic gate](http://en.wikipedia.org/wiki/Logic_gate) (http://en.wikipedia.org/wiki/Logic_gate). A logic gate is a piece of circuitry that takes in a number of inputs, and returns some form of logical result. Through a complex combination of logic gates, circuits to provide more useful operations (such as add, multiply, shift and rotate bits, etc.) can be built.

A computer processor contains the circuitry for evaluating a number of these operations, known as instructions. In order to perform a task, the processor needs to know which instruction to execute (the *operator*), and the data to operate on (the *operands*).

Computer programs (a series of instructions and data) are usually stored in memory. When running a program, the processor starts at the "top" of the program, executing instructions using a process known as the *instruction cycle*:

- Fetch: the next instruction to run is fetched from memory.
- Decode: the instruction information is decoded so the processor knows what circuitry to use.
- Execute: the instruction is run
- Store: the results of the operation

Logic units

The circuitry for performing related operations are grouped together into logic units. The most basic of these is the *Arithmetic Logic Unit*, which provides basic arithmetic capabilities. More advanced processors often have extra logic units suited to specific tasks. For example, it is normal for today's processors to have a separate *floating-point logic unit* for decimal calculations.

As part of the "decode" step of the instruction cycle, the processor chooses which logic unit (and which part of its circuitry) to activate for the current instruction.

Registers

[Registers](http://en.wikipedia.org/wiki/Processor_register) (http://en.wikipedia.org/wiki/Processor_register) are places in the processor's circuitry where it can store information about the execution currently taking place. There are different types of registers for specific purposes, including:

- **Data registers**, for temporarily storing data
- **Address registers**, for keeping track of memory addresses
- An **instruction register**, for storing the instruction currently being executed
- **Special purpose registers**, for storing internal processor information

Registers are different from normal computer memory in that they store data only for executing an instruction, and are not used as a general data store.