1.1.2 Types of Processors

CISC & RISC

Processors can use different approaches to designing their instruction sets. Two common approaches are **Complex Instruction Set Computer (CISC)** and **Reduced Instruction Set Computer (RISC)**.

CISC processors have a large instruction set containing complex instructions. A single instruction can perform several operations, which can reduce the number of instructions needed to complete a task.

RISC processors use a smaller instruction set made up of simpler instructions. A task that requires one complex CISC instruction may require several RISC instructions. However, the simpler instructions can make the processor design less complex and can make techniques such as **pipelining** more effective.

The differences between CISC and RISC affect the **instruction set, processor design, program size and performance**.

Adding two numbers using CISC or RISC

CISC	RISC
ADD a, b	Load a from memory into register1 Load b from memory into register2 Add register1 and register2 Store Answer back into memory

CISC instructions can perform more complex operations, whereas RISC uses a larger number of simpler instructions to perform the same task

CISC & RISC Features

Feature	CISC	RISC
Instruction Set	Large, with many complex instructions.	Smaller, with simpler instructions.
Instruction Complexity	Instructions can perform complex operations.	Instructions generally perform simple operations.
FDE Cycles	An instruction may require multiple FDE cycles.	Instructions are generally simple and could be executed in a single clock cycle.
Program Size	Programs may require fewer instructions.	Programs may require more instructions.
Processor Design	More complex circuitry.	Simpler circuitry.
Pipelining	More difficult due to variable/complex instructions.	Simpler instructions can make pipelining more effective.
Typical Use	Commonly used in desktops and laptops.	Commonly used in smartphones and tablets.

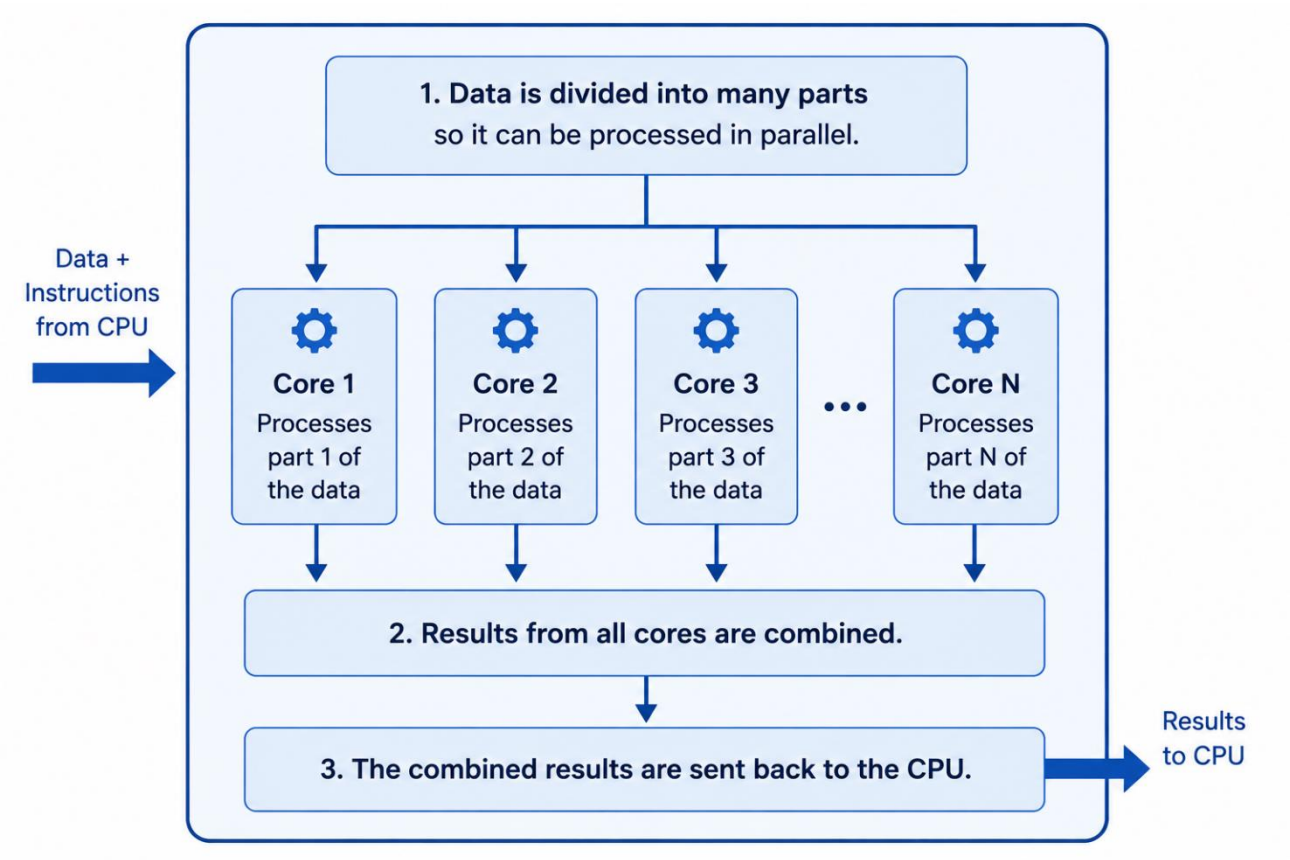
CISC vs. RISC

CISC	RISC
Programs can require less memory because a single complex instruction can perform a task that would require several RISC instructions.	Cheaper as less complex circuitry.
The compiler may need to generate fewer instructions because individual CISC instructions can perform more complex operations.	Simpler instructions can make pipelining more effective, potentially increasing instruction throughput.
Can be easier to program because individual instructions can perform more complex operations	Simpler processor circuitry can reduce manufacturing complexity and power consumption.

GPUs

A Graphics Processing Unit (GPU) is a co-processor that supplements the CPU and contains many processing units that can perform operations in parallel.

GPUs are particularly effective at processing large amounts of data in parallel, making them useful for tasks such as computer graphics, image processing and machine learning.



Multi-core and parallel systems

Multi-core systems use multiple processing cores to improve performance by distributing work across the cores, with each core capable of running its own FDE cycle.

Performance is dependent on the software being able to take advantage of parallel computing.

GPUs contain many processing units designed to perform large numbers of similar operations in parallel, whereas multi-core CPUs use a smaller number of more general-purpose processing cores.