RISC vs CISC

The purpose of both RISC and CISC architectures is to increase CPU performance, but they try to achieve that goal in different ways. Generally speaking, RISC is seen by many as an improvement over CISC. The argument for RISC over CISC is that having a less complicated set of instructions makes designing a CPU easier, cheaper and quicker.

The primary difference between RISC and CISC architecture is that RISC-based machines execute one instruction per clock cycle. In a CISC processor, each instruction performs so many actions that it takes several clock cycles to complete. In a RISC processor, every instruction also has a fixed memory size, which makes them easier to decode and execute. In a CISC machine, the instructions can be variable lengths, which increases the processing time.

At a glance, the main differences between RISC and CISC architecture are as follows:

RISC	**CISC**
Emphasis on software	Emphasis on hardware
Small number of fixed length instructions	Large number of instructions
Simple, standardised instructions	Complex, variable-length instructions
Single clock cycle instructions	Instructions can take several clock cycles
Heavy use of RAM	More efficient use of RAM
Low cycles per second with large code sizes	Small code sizes with high cycles per second