RISC vs CISC: Comparison Table

Characteristic	RISC	CISC
Instruction Complexity	Simple instructions with fixed length	Complex instructions with variable length
Instruction Count	Small number (< 100 instructions)	Large number (> 300 instructions)
Execution Time	One instruction per clock cycle	Multiple clock cycles per instruction
Instruction Format	Fixed and simple format	Multiple formats with varying complexity
Memory Access	Load-Store architecture	Any instruction can access memory
Pipeline Stages	Few pipeline stages (typically 5-7)	Many pipeline stages (typically 15-20)
Hardware Complexity	Simpler hardware design	More complex hardware design
Compiler Complexity	More complex compiler design	Simpler compiler design
Program Size	Larger program size	Smaller program size
Register Set	Large number of general-purpose registers	Fewer specialized registers
Power Consumption	Lower power consumption	Higher power consumption
Clock Speed	Generally higher clock speeds possible	Generally lower clock speeds
Examples	ARM, MIPS, SPARC, Apple M1/M2	Intel x86, AMD x86-64
Main Applications	Mobile devices, embedded systems, IoT	Desktop computers, servers, workstations
Memory Size	Requires more memory for programs	Requires less memory for programs
Code Optimization	Optimization done by compiler	Optimization done by hardware
Design Philosophy	Make the common case fast	Make the complex case manageable
Addressing Modes	Few addressing modes (2-4)	Many addressing modes (8-20)
Instruction Decoding	Simple and fast decoding	Complex decoding required
Performance	Better for parallel processing	Better for complex single instructions
Cost	Generally lower manufacturing cost	Generally higher manufacturing cost
Development Time	Shorter development cycle	Longer development cycle
Operating Temperature	Generally runs cooler	Generally runs hotter
Market Position	Dominant in mobile/embedded	Dominant in desktop/server