

Aaron Kobayashi

10 Arguments for the RISC Architecture

1. Cost Effectiveness – RISC hardware is less complex and requires less real estate therefore costing less.
2. Implementation Feasibility – The technology to support a complex processor may not currently exist and consequently increase the time it takes to get a processor to the market.
3. Design Time – Complex processors take more time to design and verify than a RISC processor.
4. Speed – A less complex chip lends itself to being “faster” because the overall propagation delays of signals will be smaller due to fewer gates in the critical path.
5. Better Use of Chip Area – A less complex chip will be allow us to use the saved transistors and “real estate” for other processor support systems such as cache.
6. Better Compilers – A reduced instruction set decreases the development time as well as the complexity of a compiler, effectively increasing the efficiency while decreasing the possibility of bugs.
7. Fewer Design Errors – A simpler processor is less likely to have hardware flaws (like the floating point problem with Intel’s Pentium chip).
8. Better ISA design – As a CISC grows in instructions, there is a tendency to create special purpose functions that are not utilized and can cause penalties the user if called at an improperly.
9. Faster Context Switching – A more simple design has less state information to save on a context switch, improving the performance of multitasking and multiprogramming operating systems/applications.
10. Power Consumption - A more simple processor can be designed with less hardware and consequently can draw less power and dissipate less heat than a more complex design.

References:

- D.A. Patterson, and D.R. Ditzel, "The Case for the Reduced Instruction Set Computer," ACM Computer Architecture News, Vol. 8, No. 6 (October 1980), pp. 25-33.