RISC Properties

- Instruction sets that only included simple straight forward instructions
- Simple addressing modes
- Fixed size machine instructions that can be fetched with one memory access
- Instruction pipelines that overlap the execution of instructions and complete one instruction per clock cycle

RISC Properties

- Restricting the use of memory operands to only the load and store type instructions
- Hardwired logic implementation as opposed to microprogramming
- The use of optimizing compilers to generate the code to perform more complex tasks
- May have large instruction sets but the instructions are simple in nature

RISC Disadvantages

Require multiple instructions to accomplish what CISC machines perform in a single complex instruction.

The compiler (or assembly language programmer) has to generate a sequence of simple instructions that perform the same actions of the individual CISC instructions.

Can result in code expansion which can be a problem.

RISC Disadvantages

Code expansion refers to the increase in size that you get when you take a program that had been compiled for a CISC machine and re-compile it for a RISC machine

The amount of expansion depends primarily on the compiler

higher quality compilers generate less code

the nature of the machine's instruction set has an effect

Typically, code expansion can range up to 100% or more

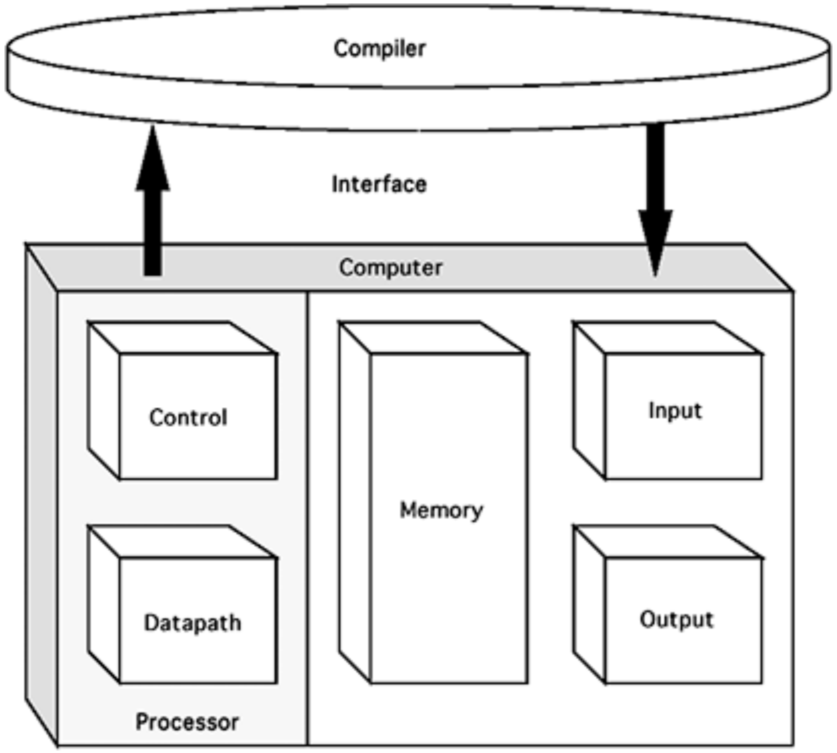
Example RISC Machines

- MIPS
- Sparc
- ARM
- PowerPC

The MIPS architecture will be explained in detail later in the course as a prime example of a RISC machine.

Moving Complexity to the Compiler

much of the complexity is handled by the compiler not the hardware



optimizations such as instruction rearrangement take care of pipeline dependencies

RISC Registers

The simpler nature of the RISC control unit leaves space (real estate) on the CPU chip

This extra space can be used to implement additional registers

RISC machines tend to have 32 or more registers

CISC machines may have 8 or fewer registers

Sparc processors have more than a hundred CPU registers organized into logical groups called register windows

JOHNS HOPKINS UNIVERSITY | Engineering for Professionals

RISC versus CISC Summary

RISC	CISC
Simple instructions	Complex instructions
Completes one instruction per cycle	Requires many cycles to complete an instruction
Load/Store Architecture (only load and store instructions can reference memory)	Most instructions can reference memory
Relies heavily on pipelining	Relies less on pipelining
Instructions executed in hardware	Microcode interprets instructions
Fixed size instructions with a small number of formats	Variable size instructions with many formats
Simpler addressing modes	Many intricate addressing modes