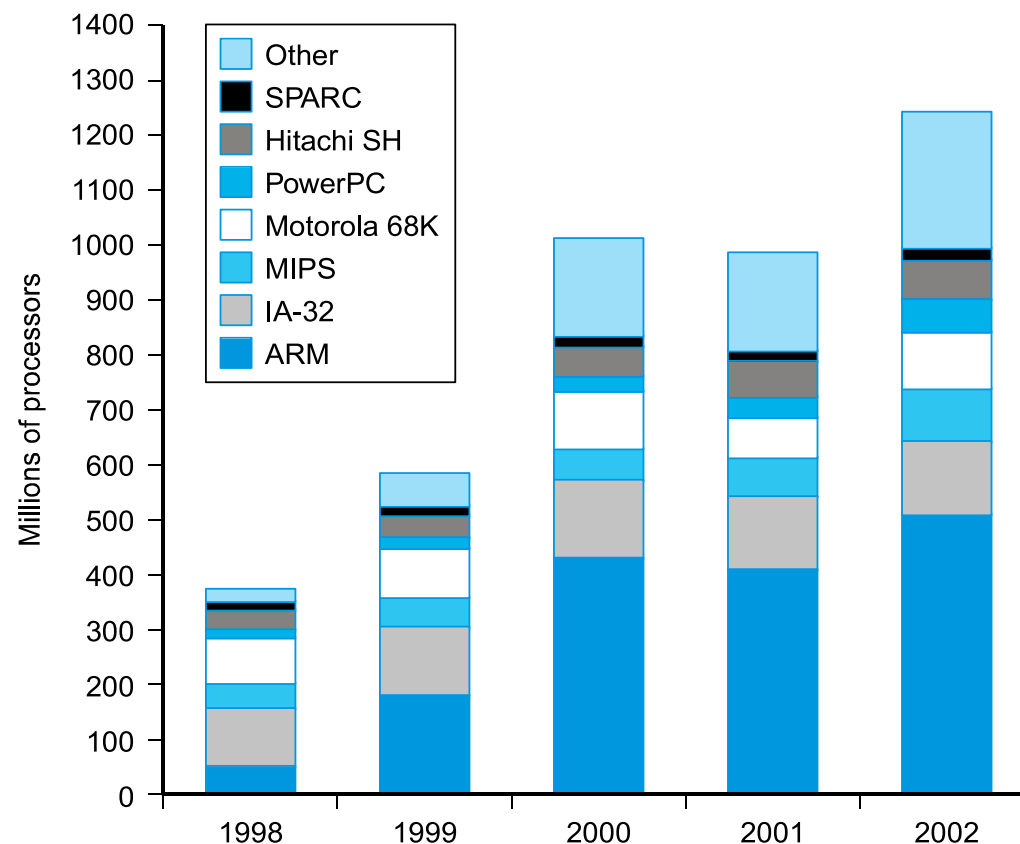


Chapter 2

Instructions:

- Language of the Machine
- *Stored program* concept: data + instructions in numbers
- We'll be working with the MIPS instruction set architecture
 - similar to other architectures developed since the 1980's
 - Almost 100 million MIPS processors manufactured in 2002
 - used by NEC, Nintendo, Cisco, Silicon Graphics, Sony, ...



RISC - Reduced Instruction Set Computer

- RISC philosophy
 - fixed instruction lengths
 - load-store instruction sets
 - limited addressing modes
 - limited operations
- MIPS, Sun SPARC, HP PA-RISC, IBM PowerPC, Intel (Compaq) Alpha, ...
- Instruction sets are measured by how well compilers use them as opposed to how well assembly language programmers use them

Design goals: speed, cost, size, power consumption, reliability, memory space

MIPS arithmetic

- All instructions have 3 operands
- Operand order is fixed (destination first)

Example:

C code: `a = b + c`

MIPS 'code': `add a, b, c`



(we'll talk about registers in a bit)

“The natural number of operands for an operation like addition is three...

requiring every instruction to have exactly three operands,

no more and no less, conforms to the philosophy of **keeping the hardware simple”**

For a given level of function, however, that system is **best** in which one can specify things with the most **simplicity** and **straightforwardness**. ...
Simplicity and straightforwardness proceed from **conceptual integrity**. ... Ease of use, then, dictates unity of design, conceptual integrity.

The Mythical Man-Month, Brooks, pg 44

MIPS Arithmetic Instructions

- MIPS assembly language arithmetic statement

```
add $t0, $s1, $s2
```

```
sub $t0, $s1, $s2
```

- Each arithmetic instruction performs only one operation
- Each arithmetic instruction fits in 32 bits and specifies exactly three operands
- Those operands are all contained in the datapath's registers (\$t0,\$s1,\$s2) – indicated by \$
 - only 32 registers provided
- Operand order is fixed (destination first)
- Each register contains 32 bits (a word)
- Design Principle: smaller is faster. Why?