In More Depth: MIPS, MOPS, and Other FLOPS

One particularly misleading definition of MIPS that has been occasionally popular is *peak MIPS*. Peak MIPS is obtained by choosing an instruction mix that minimizes the CPI, even if that instruction mix is totally impractical. In practice, processors are sometimes marketed by touting the peak MIPS rating, which can distort the real picture of performance. When the Intel i860 was announced in February 1989, the product announcement used the peak performance of the processor to compare performance against other computers. The i860 was able to execute up to two floating-point operations and one integer operation per clock. With a clock rate target of 50 MHz, the i860 was claimed to offer 100 MFLOPS and 150 MOPS (millions of operations per second). The first i860-based systems (using 40-MHz parts) became available for benchmarking during the first quarter of 1991. By comparison, a MIPS computer based on a 33 MHz R3000 processor, available at about the same time, had a peak performance of about 16 MFLOPS and 33 MOPS. Although the peak performance claims might suggest that the i860-based computer was more than five times faster than the R3000-based computer, the SPEC benchmarks showed that the R3000-based computer was actually about 15% faster! Although peak MIPS is an essentially useless measure, computer manufacturers still occasionally announce products using peak MIPS as a metric, often neglecting to include the word "peak"!

One attempt to retain the use of the term *MIPS*, but to make it useful among different instruction sets, was to choose a definition of MIPS that is relative to some agreed-upon reference computer, similar to the SPEC ratio measurement. *Relative MIPS*, the term used for this measure, has been defined as follows:

$$\text{Relative MIPS} = \frac{\text{Time}_{\text{reference}}}{\text{Time}_{\text{unrated}}} \times \text{MIPS}_{\text{reference}}$$

where

$\text{Time}_{\text{reference}}$ = Execution time of a program on the reference computer

$\text{Time}_{\text{unrated}}$ = Execution time of the same program on computer to be rated

$\text{MIPS}_{\text{reference}}$ = Agreed-upon MIPS rating of the reference computer

Relative MIPS is proportional to execution time *only* for a given program and a given input. Even when these are identified, it becomes harder to find a reference computer on which to run programs as the computer ages. (In the 1980s the dominant reference computer was the VAX-11/780, which was called a 1-MIPS computer, for a reason we will discuss shortly, and is now hard to find in operation.) Moreover, should the older computer be run with the newest release of the compiler and operating system, or should the software be fixed so the reference computer does not become faster over time? There is also the temptation to generalize from a relative MIPS rating obtained using one benchmark to a general statement about relative performance, even though there can be wide variations in performance of two computers across a complete set of benchmarks.

The popularity of the VAX-11/780 made it a popular reference computer for relative MIPS, especially since relative MIPS for a 1-MIPS reference computer is easy to calculate. If a computer was five times faster than the VAX-11/780, its rating for that benchmark would be 5 relative MIPS. The 1-MIPS rating was widely believed for four years, until Joel Emer of DEC measured the VAX-11/780 under a timesharing load. Emer found that the actual VAX-11/780 MIPS rate was 0.5. Subsequent VAXes that run 3 million VAX instructions per second for some benchmarks were therefore called 6-MIPS computers because they run six times faster than the VAX-11/780.

The 1970s and 1980s marked the growth of the supercomputer industry, which was defined by high performance on floating-point-intensive programs. Average instruction time and MIPS were clearly inappropriate metrics for this industry—hence the invention of MFLOPS (millions of floating-point operations per second). Unfortunately, customers quickly forgot the program used for the rating, and marketing groups decided to start quoting peak MFLOPS in the supercomputer performance wars.

4.29 [4 hours] <§4.3> Devise a program in C or Java that determines the peak MIPS rating for a computer. Run it on two computers to calculate the peak MIPS. Now run a real C or Java program such as a compiler on the two computers. How well does peak MIPS predict performance of the real program?