MICROPROCESSOR REPORT
VOLUME 11, NUMBER 14
OCTOBER 27, 1997
THE INSIDERS' GUIDE TO MICROPROCESSOR HARDWARE

Intel, HP Make EPIC Disclosure

IA-64 Instruction Set Goes Beyond Traditional RISC, VLIW

by Linley Gwennap

This is a summary version of the in-depth article that appeared in Microprocessor Report.

Breaking out of the 1980s RISC mind set, Intel and Hewlett-Packard have designed a new instruction set, IA-64, geared toward the highly parallel processors of the next decade.IA-64 goes beyond previous CISC,RISC,and VLIW instruction sets with a new set of features that its creators call EPIC (explicitly parallel instruction computing). This strategy should give Merced, the first IA-64 chip, a leg up on its old-fashioned competitors when it debuts in 1999.

EPIC is similar in concept to VLIW in that both allow the compiler to explicitly group instructions for parallel execution. This technique eliminates much of the dependency-checking and grouping logic that consumes an increasingly large portion of advanced RISC and x86 processors. EPIC's flexible grouping mechanism solves VLIW's two fatal flaws: excessive code expansion and lack of scalability.

The new instruction set attacks other problems with current architectures. With four times as many addressable registers as a typical RISC processor, IA-64 eliminates the need for register renaming and reduces time-consuming cache accesses. When cache accesses are required,speculative loads can hide cache latency even when branches are in the way. Some of these branches can be eliminated entirely with predicated execution, reducing opportunities for onerous branch mispredictions.

Speaking at the recent Microprocessor Forum, architects Jerry Huck (HP) and John Crawford (Intel) disclosed these key features of IA-64 but did not provide a description of the new instruction set. Nonetheless, what was revealed clearly tags IA-64 as a new type of instruction set compared with today's RISC and x86 chips. RISC processor vendors can't simply retrofit these features into their existing instruction sets, forcing them to create new instruction sets or, as Intel has in the past, limp along with an inferior design.

IA-64 instructions use a unique format that allows the compiler to direct hardware execution without severely bloating the software. As Figure 1 shows, a single 128-bit "bundle" contains three IA-64 instructions along with "template" information about the bundle. The template indicates whether the instructions in the bundle can be executed in parallel or if one or more must be executed serially, due to register dependencies. The template also indicates whether the bundle can be executed in parallel with the following bundle. Bundles can be chained to create instruction groups of any length.

For a group of 12 parallel instructions, for example, one processor might take two cycles to issue them all, whereas a more advanced implementation might issue them all in one cycle. The same binary code would thus run on both processors without any modification.

This issue logic makes an IA-64 processor more complex than a pure VLIW design, but the ability to build a family of binary-compatible processors is well worth the extra logic. This logic is much less complicated than the issue logic in an out-of-order superscalar processor.

The IA-64 architects included 128 general registers and 128 floating-point registers in their design. The compiler can take advantage of this increase by performing more aggressive optimizations. For example, unrolling short loops sev-

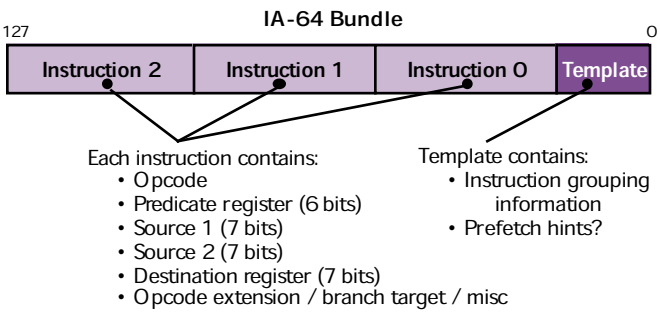

Figure 1. Three IA-64 instructions are encoded into a "bundle" along with a "template" that provides grouping information. The companies did not provide the width of the instructions or the template, or details about their contents.

eral times often increases performance, but each instantiation of the loop requires more registers to hold additional copies of the local variables. With 128 registers, the compiler can unroll loops more often while still leaving global variables in registers.

IA-64 processors will include 64 predicate registers (PR), each just one bit. Most IA-64 instructions include a predicate field; the instruction is executed only if the selected predicate register is "true." Predicates are generated by CMP instructions that compare the value of two registers (using a variety of conditions). A single CMP instruction stores the result of the comparison in one PR and automatically stores the inverse of the comparison in a second PR. This mechanism allows the processor to more efficiently handle the common IF-THEN-ELSE construction with small routines in each of the blocks.

IA-64 will allow speculative, or nonfaulting, loads. A speculative load, indicated by the .S suffix, will not trigger an exception; instead, if an exception occurs, the target register will be marked invalid. This is a fairly simple mechanism, requiring only that each register have a valid bit.

Using this mechanism, the load can be placed as early as possible in the code, as long as the address can be computed. If the data is never checked, no exception will be triggered; if an exception occurs when the data is needed, the exception will be recognized in the load's original "home block." Speculative loads thus provide the compiler with maximum flexibility to hide cache latency.

At the Forum, Intel's Fred Pollack confirmed that Merced is on target for shipments in 1999, using a 0.18-micron process; we expect it to appear in systems around the middle of that year.

Merced, however, probably won't demonstrate the full benefits of the instruction set, at least at first. For any new instruction set, compilers take some time to mature; all the simulation in the world can't match the development that can be done on dozens of real machines, and these won't be available until late next year. Thus, the initial benchmarks won't reflect the full performance of the processor.

Merced may also be hampered by the x86-compatibility logic. Even if this logic doesn't reduce clock speed or native-mode performance, its mere existence consumes die area that could have been used to enhance native performance.

Still, the advantages of IA-64 should provide a potent weapon. Pollack asserted that Merced will deliver "industry-leading performance" when it first begins shipping.

That lead is likely to grow over time as the IA-64 compilers mature and Intel squeezes more clock speed out of the Merced design. Pollack revealed that work has already started on a follow-on to Merced that will offer up to twice the performance of the initial chip in the same IC process. That chip, due in 2001, is likely to combine an even more powerful EPIC core with a new system interface that boosts performance to spectacular levels. MPR