Patent Watch

by Rich Belgard, Contributing Editor

The following U.S. Patents related to microprocessors were issued recently. Please e-mail comments to belgard@umunhum.stanford.edu.

5,448,705
RISC microprocessor architecture implementing fast trap and exception state
Issued: September 5, 1995
Inventors: Le Nguyen, et al
Assignee: Seiko Epson
Filed: May 24, 1993
Claims: 3
A method for use in a microprocessor to return execution to a main program after processing an interrupt. The method comprises fetching instructions from a main instruction stream to a main buffer section of a prefetch buffer and executing those instructions. The method also provides for returning to the main instruction stream without requiring prefetching of instructions already fetched. Similarly, the method also provides for handling of nested interrupts.

5,446,912
Partial width stalls within register alias table
Issued: August 29, 1995
Inventors: Robert Colwell, et al
Assignee: Intel
Filed: December 29, 1993
Claims: 34
A partial width stall mechanism within a register alias table (RAT) for handling partial-width data dependencies of operations issued simultaneously within a superscalar microprocessor. Operations are presented to the RAT in program order; partial-width data dependencies occur when the size of a logical source register is larger than the corresponding physical source register. At this occurrence, the data required by the logical source register does not reside in any one physical source register. Therefore, renaming of that logical register must be stalled until the data for that logical register is accumulated into one location. The data will be accumulated when the last operation to have written the physical source register is retired.

5,442,760
Decoded instruction cache architecture with each instruction field in multiple-instruction cache line directly connected to specific functional unit
Issued: August 15, 1995
Inventors: Einar Rustad, et al
Assignee: Dolphin Interconnect Solutions AS
Filed: December 22, 1993
Claims: 8
A decoded instruction cache has multiple instructions per cache line. For cache hits, the decode logic fills the cache line with instructions up to its limit. For cache misses, the cache line enables the processor to dispatch multiple instructions during one clock cycle. An important feature of the instruction cache is that it holds the target addresses for the next instructions. No separate address logic is needed to proceed in the program execution during cache hits. A conditional branch holds its alternative target address in a separate field.

5,442,757
Computer processor with distributed pipeline control that allows functional units to complete operations out of order while maintaining precise interrupts
Issued: August 15, 1995
Inventors: Harold McFarland, et al
Assignee: NexGen
Filed: March 3, 1993
Claims: 56
A pipeline control system is distributed among the functional units in a processor. A decoder issues operations, each with an associated tag, to the functional units, with up to n operations allowed to be outstanding. The units execute the operations and report termination status to the decoder but do not irrevocably change the state of the machine. Based on the termination information, the decoder retires normally terminated operations in order. If an operation terminates abnormally, the decoder instructs the units to back out of those operations later than the operation that terminated abnormally.

5,442,756
Branch prediction and resolution apparatus for a superscalar computer processor
Issued August 15, 1995
Inventors: Edward Grochowski, et al
Assignee: Intel
Filed: July 31, 1992
Claims: 15
An apparatus and method for improving the performance of superscalar pipelined computers using branch prediction and verification that the predicted branch is correct. A predicted branch may be resolved in one of two distinct pipeline stages, and a method is provided for handling branches that are resolved in either of the pipeline stages. A branch verification method is provided that verifies that the architecturally correct instructions are in the decode and execution stages. ♦

