Patent Watch

by Rich Belgard, Contributing Editor

This issue introduces a new column that lists a selection of recently issued U.S. patents that apply to microprocessors. This column will typically appear as one page per issue, but to kick it off we have included two pages of patents issued during the first two months of 1995.

5,394,558
Data processor having an execution unit controlled by an instruction decoder and a microprogram ROM
Issued: February 28, 1995
Inventors: Arakawa, Fumio, et al
Assignee: Hitachi
Filed: July 1, 1994
Claims: 2
A data processor in which, when two primitive instructions are decoded by instruction decoders, the two primitive instructions are executed in parallel by instruction execution units in accordance with the decoded outputs of the instruction decoders.

5,394,530
Arrangement for predicting a branch target address in the second iteration of a short loop
Issued: February 28, 1995
Inventor: Kitta, Mayumi; Yamanashi, Japan
Assignee: NEC
Filed: February 22, 1994
Claims: 6
Improved techniques for predicting a branch target address using a branch history table (BHT).

5,392,437
Method and apparatus for independently stopping and restarting functional units
Issued: February 21, 1995
Inventors: Matter, Eugene P., et al
Assignee: Intel
Filed: November 6, 1992
Claims: 32
A mechanism for powering down a function unit on an integrated circuit having multiple function units. Some of the function units are clocked independently of each other. Also included is a method and mechanism for powering down a function unit transparently and independently of the rest of the integrated circuit when the function unit is not required for use.

5,390,311
Method and apparatus for accessing microcoded instructions in a computer system
Issued: February 14, 1995
Inventors: Fu, Beatrice P., et al
Assignee: Intel
Filed: September 7, 1993
Claims: 22
A microprogrammed computer system having means for reducing entry-point table size. The invention describes a microprogrammed computer in which opcodes are grouped and each group is mapped to a common entry-point address in a microcode sequence memory. The actual microcodes for the instructions within the group are differentiated by opcode bits. Utilizing the structure of the present invention reduces the number of required inputs to a PLA or similar means for generating entry-points and further increases circuit speed and reduces entry-point table size.

5,390,309
Virtual address translation in a three-level virtual machine
Issued: February 14, 1995
Inventor: Onodera, Osamu; Hadano, Japan
Assignee: Hitachi
Filed: July 25, 1991
Claims: 11
A level-2 virtual machine is constructed under the control of a level-1 operating system (OS) operating on a real machine (level-1), and a level-3 virtual machine is constructed under the control of another operating system (OS) operating on the level-2 virtual machine. A level-3 virtual address generated in the level-3 virtual machine is translated to a level-2 virtual address, which is further translated to a level-1 virtual address. A third predetermined main storage address is added to the level-1 virtual address to generate a level-1 absolute address. The translated address is checked as to whether it is within a predetermined area on the main storage.

5,388,265
Method and apparatus for placing an integrated-circuit chip in a reduced power-consumption state
Issued: February 7, 1995
Inventor: Volk, Andrew M.
Assignee: Intel
Filed: April 26, 1993
Claims: 17
A method and apparatus for a chip to monitor its own activity and enter and exit a state of reduced power consumption. The invention includes defining a predetermined state in which the chip can power down cleanly and monitoring the chip to determine when the chip is in that predetermined state. The invention also includes a

method and apparatus for putting the chip in a state of reduced power consumption when the chip is in the predetermined state. The present invention also includes a method and apparatus for either turning off the clock-generation circuitry or leaving it on during the power-down state.

5,388,227
Transparent data bus sizing
Inventor: McFarland, Harold L.
Assignee: NexGen
Issued: February 7, 1995
Filed: October 7, 1992
Claims: 28
A bus system wherein N-bit devices attached to the lower half of a 2N-bit bus communicate with 2N-bit devices attached to the full bus. Bidirectional registered transceivers are coupled between the upper and lower halves of the bus.

5,388,226
Method and apparatus for accessing a register in a data processing system
Issued: February 7, 1995
Inventors: Gutierrez, Joseph A., et al
Assignee: Motorola
Filed: October 5, 1992
Claims: 20
A method and apparatus for accessing a register in a data-processing system. In one form, the present invention uses a storage circuit to act as a register copy of the original register. An access to the register location may actually selectively access the original register, the register copy, or both the original register and the register copy. Register-select logic is used to perform this selection function.

5,386,585
Self-timed data pipeline apparatus using asynchronous stages having toggle flip-flops
Issued: January 31, 1995
Inventor: Traylor, Roger L.
Assignee: Intel
Filed: November 18, 1993
Claims: 1
A self-timed data pipeline comprised of a plurality of pipeline stages, each incorporating at least one data latch coupled to selectively configured combinational logic. The combinational logic is selectively configured to suit the demands of the particular data pipeline and provides clocking to at least one data latch in the pipeline stage.

5,383,192
Minimizing the likelihood of slip between the instant a candidate for a break event is generated and the instant a microprocessor is instructed to perform a break, without missing breakpoints
Issued: January 17, 1995
Inventor: Alexander, James W.; Hillsboro, Oregon
Assignee: Intel
Filed: December 23, 1992
Claims : 4
An in-circuit emulator on an integrated-circuit chip having an input pin for externally triggering on-chip break mechanisms. A break logic having an arming input is connected to an instruction pointer counter (IP counter). The break logic matches the IP counter to an instruction execution address. A counter is provided that, once started, runs for a period of time and then shuts itself off, the length of the period of time being equal to the amount of time it takes for the break logic to arm after assertion of the arming input. The break-logic control starts the counter. The break-logic control includes means connected to the arming input, to the counter, to the match output, and to the abrupt break input to inhibit the assertion of the arming input by the break-logic control and assert the abrupt break input to the abrupt break logic, operative upon the condition that the match output is asserted during that period of time.

5,381,531
Data processor for selective simultaneous execution of a delay-slot instruction and a second subsequent instruction in the pair following a conditional branch instruction
Issued: January 10, 1995
Inventors: Hanawa, Makoto, et. al.
Assignee: Hitachi
Filed: July 9, 1991
Claims: 9
An instruction-fetch unit of a data processor capable of simultaneous execution of two instructions fetches a first and a second instruction from a memory in one cycle. The first and the second instruction thus fetched are set in a first and a second register before being decoded in a first and a second instruction decoder. Comparators compare data on the destination field of the first instruction with data on the source field of the second instruction. When the data is inconsistent, a parallel operation control unit permits the first and the second instruction execution unit under the first and the second instruction to execute the two instructions in response to the outputs of the comparators. When the data is consistent, the parallel operation control unit inhibits the parallel execution.

Other Issued Patents
5,386,469 *Firmware encryption for microprocessor/microcomputer*
5,381,533 *Dynamic-flow instruction-cache memory organized around trace segments independent of virtual address line* ♦

