EXHIBIT F

I hereby certify that this paper is being deposited with the United States Postal Service with sufficient postage as First Class Mail under 37 CFR 1.8 on the date indicated below and is addressed to: "Commissioner of Patents and Trademarks, Washington, DC 20231."

Signed _____
Stuart T. Auvinen, Reg. No. 36,435

Date Signed and Mailed 2/14/95

#7B
2/15
fwd

PATENT
RM-1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of)
)
Blomgren et al.)
)
Serial No. 08/179,926)
)
Filed: 1/11/94)
)
For: Dual-Instruction-Set Architecture CPU)
with Hidden Software Emulation Mode)
)
)

Examiner: V. Vu

Group Art Unit: 2315

RECEIVED
FEB 17 1995

AMENDMENT UNDER 37 C.F.R. § 1.111

Hon. Commissioner of Patents and Trademarks
Washington, DC 20231

Sir:

In response to the office action mailed 11/16/94, please amend the above-identified application as follows:

In the claims:

Please amend claims 1, 14, and 18 as follows:

1. (twice amended) A central processing unit (CPU) for processing instructions from two separate instruction sets, said CPU comprising:

first instruction decode means for decoding instructions from a first instruction set, said first instruction set having a first encoding of instructions;

second instruction decode means for decoding only a subset of instructions from a second instruction set, said second instruction set having a second encoding of instructions, said first encoding of instructions independent from said second encoding of instructions;

select means, coupled to said first instruction decode means and said second instruction decode means, for selecting said decoded instruction from either said first instruction decode means or from said second instruction decode means; and

execute means for executing decoded instructions selected by said select means,

whereby instructions from both said first instruction set and said second instruction set are executed by said CPU.

---

14. (twice amended) A method for processing instructions from two separate instruction sets on a central processing unit (CPU), said method comprising:

decoding instructions from a first instruction set with a first instruction decoder, said first instruction set having a first encoding of instructions;

decoding only a subset of instructions from a second instruction set with a second instruction decoder, said second instruction set having a second encoding of instructions, said first encoding of instructions independent from said second encoding of instructions;

selecting said decoded instruction from either said first instruction decoder or from said second instruction decoder; and

executing said decoded instruction that was selected,

whereby instructions from both said first instruction set and said second instruction set are executed by said CPU.

---

18. (twice amended) A microprocessor for executing instructions belonging to a reduced instruction set computer (RISC) instruction set and for executing instructions belonging to a complex instruction set computer (CISC) instruction set, said microprocessor comprising:

RISC instruction decode means, for decoding instructions belonging to said RISC instruction set;

CISC instruction decode means, for decoding only a subset of instructions belonging to said CISC instruction set;

mode register means for indicating a current operating mode of said microprocessor;

enable means, coupled to said RISC instruction decode means and said CISC instruction decode means for enabling said decoding of instructions belonging to said RISC instruction set or belonging to said CISC instruction set, said enable means responsive to said current operating mode of said microprocessor; and

an execution unit, coupled to said first instruction decode means and said second instruction decode means, for executing instructions belonging to said first instruction set and instructions belonging to said second instruction set,

whereby instructions from said RISC instruction set and instructions from said CISC instruction set can be executed by said execution unit.

REMARKS

Claims 1-4, 14, and 18-20 were rejected under 35 USC § 112 as being indefinite and not commensurate in scope with the disclosure. Under 35 USC § 103, claims 1-5, 14-16, and 18-20 were rejected as obvious over Portanova et al (US Pat. No. 4,992,934) in view of Onishi (U.S. Patent No. 3,764,988). Claims 6-13 and 17 were rejected under 35 USC § 103 as obvious over Portanova in view of Onishi as set forth for claims 1-5, and further in view of Bullions, III, et al (U.S. Patent No. 4,456,954). All claims were thus rejected.

Claims 1, 14, and 18 were amended to add a limitation to make the scope of the claims commensurate with the disclosure as noted by the Examiner. Applicant submits that with these amendments and the discussion below that claims 1-20 are allowable over

the cited references. Reexamination and reconsideration of the claims, as amended, is hereby requested.

Summary of Independent Claims 1, 14, 15, 18

Claim 1 recites a first and a second instruction decoder for decoding instructions from a first and a second instruction set. A select means selects either the decoded instruction from the first decoder or from the second decoder. An execute means executes the decoded instruction selected by the select means. Thus the execute means can execute both first and second instructions provided by the select means.

Independent claim 14 is directed to a method for processing instructions from two separate instruction sets. Independent claim 15 is directed to a method for processing instructions from a CISC and a RISC instruction set in which all CISC instructions are executable, either directly by the execute unit or by emulation with RISC instructions. Independent claim 18 is directed to a microprocessor for executing RISC and CISC instructions using both RISC and CISC instruction decoders and an enable means to enable one of the instruction decoders.

35 USC § 112 Rejection

Claims 1-4, 14, and 18-20 were rejected under 35 USC § 112, first paragraph, as the disclosure is enabling only for claims limited to decoder capable of directly decoding a subset and not the entire non-native instruction set.

A limitation that only a subset of the non-native instruction set is decoded was added to independent claims 1, 14, and 18. Claim 1 was amended to recite a "second instruction decode means for decoding only a subset of instructions from a second instruction set." Likewise, claims 14 and 18 were amended to recite that only a subset of the second or the CISC instruction set is decoded.

Thus with these amendments the claims recite a limitation that only a subset of the second instruction set is decoded, making the claims commensurate in scope with the specification. Thus the 35 USC § 112, first paragraph rejection has been overcome.

PRIOR ART REJECTIONS - 35 USC § 103 - PORTANOVA IN VIEW OF ONISHI

Under 35 USC § 103, claims 1-5, 14-16, and 18-20 were rejected as obvious over Portanova et al (US Pat. No. 4,992,934) in view of Onishi (U.S. Patent No. 3,764,988). For claims 1-2, Portanova teaches a system capable of executing both RISC and CISC instructions. In particularly, the system comprises a core structure of a RISC computer, and an emulation unit using RISC routine for emulating the execution of CISC instruction. The CISC emulation can be implemented with hardwired or firmware. It is noted that the hardware implementation of the CISC emulation would have required a modification to the RISC processor for providing the additional capability to decode and execute CISC instructions.

Portanova does not specifically teaches using two separate decoder units for decoding RISC and CISC instructions respectively. The use of multiple decoder units for decoding different types of instructions is however well-known in the art. The use of multiple decoders, each designated to decode certain type of instructions, is desirable because it allows a simple and efficient design of the instruction decoder. Onishi teaches a processor comprising two instruction decoders, the first decoder for decoding normal instructions and the second decoder for decoding branch instructions (see abstract). By using the second instruction decoder, the decoding sequence of a branch instruction can be reduced (see summary). It is noted that a selector is obviously needed to select decoded instructions from the first and second decoders.

Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify Portanova's system to utilize two instruction decoders as taught by Onishi for decoding RISC and CISC instructions respectively. This is because the use

of separate instruction decoder units for RISC and CISC instructions allows more simple and efficient design of the decoder units. Applicant respectfully disagrees.

Deficiencies with Portanova et al.

Portanova teaches a system that emulates CISC instructions with routines of RISC instructions. CISC instructions are never directly executed on Portanova's hardware, such as his ALU 104. CISC instructions must be replaced by RISC instructions to be executed. Only RISC instructions are ever directly executed in Portanova's hardware.

It was stated in the rejection (page 6, 1st paragraph) that "The CISC emulation can be implemented with hardwired or firmware. (see col 29, line 60 - col 30, lines 12 and figures 9-10). It is noted that the hardware implementation of the CISC emulation would have required a modification to the RISC processor for providing the additional capability to decode and execute CISC instructions."

Applicant notes that no details of a hardware implementation of CISC instructions in a RISC processor are provided in Portanova. If indeed Portanova does teach hardware execution of CISC, then more disclosure is required of Portanova. As the Examiner notes in his 35 USC § 112, first paragraph rejection, "the design of such a decoder and execution units" [capable of both RISC and CISC], "if possible, is not clearly obvious to one skilled in the art without further requiring undue experimentation because the complexity of the system with such ability would increase significantly." Certainly Portanova does not provide an enabling disclosure for a combined RISC and CISC execution unit or decoder.

A close examination of the cited Figure 11 and the context at cols. 29-30 reveals that the discussion in Portanova centers on prior-art CISC systems, not on a combined RISC and CISC processor. Indeed, Figure 11 is clearly labeled "PRIOR ART". At col 29-30, Portanova discusses the third aspect of his invention, a design methodology for

implementing a CISC (col 29, line 59-62). He then discusses current, prior-art design methods for existing CISC systems. Figure 9 is a prior-art Z8000 CISC, which is a hardwired CISC architecture. Figure 10 is a Motorola 68000 CISC, which uses firmware rather than direct hardwired control. Figure 11 is a MicroVAX (col 30, line 28). The MicroVAX uses software emulation of complex CISC instructions, but firmware of simpler CISC instructions. Figure 12 is an IBM S/370, (col 30, line 37) which uses software emulation to implement CISC instructions 326 on RISC hardware 334.

It should be noted that the Z8000, Motorola 68000, MicroVAX, and System/370 are all well-known CISC architectures. The hardware that actually executes instructions in Portanova's Figures 9-11, (302, 310, and 324) are CISC execute units that do not execute RISC instructions. Figure 12's hardwired unit 334 is a RISC unit that emulates CISC using software emulation 330, similar to Portanova's RISC that emulates CISC. Thus no hardware unit in Portanova's Figures 9-12 teach hardware that can execute both RISC and CISC.

Portanova's design methodology apparently could emulate all these CISC architectures: Z8000, 68000, VAX, S/370, implemented on his RISC processor. The RISC processor is first designed and optimized, then the CISC emulation code is written (col 30, lines 39-64). These can be two separate steps because the RISC hardware does not directly execute any CISC instructions. The fact that the RISC hardware is first designed, optimized, and fabricated before the CISC emulator is written teaches away from the invention, which executes some CISC instructions on a RISC processor. If the RISC hardware is modified for CISC hardware emulation, then CISC must be taken into account during the design of the RISC processor.

Portanova's RISC processor never executes CISC instructions, but merely emulates the CISC instruction-set architecture by replacing CISC instructions with routines of RISC

instructions. Nowhere does Portanova teach or suggest that any CISC instructions are executed on the RISC execute unit. Each CISC instruction is emulated (col 30, line 45, also col 7, lines 10-25).

Portanova thus does not teach or suggest hardware implementation of the CISC emulation. It is true that hardware execution of CISC would require a modification to the RISC processor to provide the additional capability of decoding and execution of CISC instructions. Nowhere does Portanova disclose such a modification or suggest that both CISC and RISC instructions can be executed on the same hardware. The fact that such a modification is necessary, and that Portanova does not teach or suggest such a modification is evidence that the present invention is not obvious. Certainly Portanova does not teach or suggest that a subset of the CISC instructions can be executed on the RISC hardware while the other CISC instructions are emulated.

Deficiencies of Onishi

Portanova does not specifically teach using two separate decoder units for decoding RISC and CISC instructions respectively. Onishi teaches a processor comprising two instruction decoders, the first decoder for decoding normal instructions and the second decoder for decoding branch instructions.

Onishi teaches a decoder for a single instruction set. He partitions or divides this decoder into one decoder for branch instructions, and the other decoder for all other types of instructions. This is done to reduce the decoding sequence for branch instructions, which are among the most speed-critical instructions. This makes the computer operate at higher efficiency (col 1, line 20).

Onishi Solves a Different Problem, for a Different Motivation

In contrast, the present invention uses two decoders because two separate instruction sets must be decoded. The computer does not operate at higher efficiency due to the

two decoders, and indeed may be slower because two decoders are needed rather than just one decoder. The motivation for using two decoders is to execute two instruction sets rather than just one instruction set. Thus flexibility rather than speed is the problem solved by the present invention. Solving a different problem is an indication of non-obviousness.

With two instruction decoders for two different instruction sets, the same bit pattern or opcode can be decoded into two different instructions, one for RISC and the other for CISC. Both outputs can be valid operations. Thus the present invention can output valid decoded instructions from both of the decoders, and one must be selected. Applicant's specification explains how the same opcode, 03 hex, can be two valid operations — CISC addition or RISC trap-word-immediate:

> This same opcode, 03 hex, corresponds to a completely different instruction in the RISC instruction set. In CISC 03 hex is an addition operation, while in RISC 03 hex is TWI - trap word immediate, a control transfer instruction. Thus two separate decode blocks are necessary for the two separate instruction sets. (Spec on page 25, lines 2-6)

Onishi partitions a single decoder for a single instruction set into two decoders for different types of instructions (branches) within that one instruction set. With Onishi, one of the decoder's outputs will always be invalid. The present invention uses two decoders because two separate instruction sets are decoded. Thus Onishi does not teach or suggest that a decoder for a RISC instruction set be used with a second decoder for a CISC instruction set.

As to claims 3-4, it would have been obvious to one skilled in the art to utilize an execution mode register for indicating the execution of native and non-native instructions. As to claim 5, Portanova further teaches that the CISC emulation unit can also be implemented by using both hardware and software in which some CISC instructions would be directly decoded and executed by the execution unit and the execution of the rest of CISC instructions is emulated by using RISC routines (see col 30, lines 13-28 and figure 11). Thus, it would have been an obvious engineering

design choice to one of ordinary skill in the art at the time of the invention to utilize both software and hardware implementation to emulate CISC instructions on a RISC computer. The implementation of both software and hardware approaches could have been motivated because of the combined advantages of both techniques, i.e. the simpleness and flexibility of the software emulation approach and the speed of the hardware emulation approach. Applicant respectfully disagrees.

As noted above for claims 1-4, Portanova emulates all CISC instructions. Portanova nowhere teaches or suggests that any CISC instructions be directly executed. Indeed, Portanova teaches away from hardware execution of any CISC instructions because Portanova's design methodology is to first build a pure RISC processor hardware, without regard to the CISC architecture, be it VAX, Z8000, or 68K, and then write the CISC emulation code. As discussed above, the cited figures and cols 29-30 of Portanova clearly are discussing prior-art CISC systems that can be EMULATED on his RISC hardware.

The rationale that such a major modification to the RISC hardware to support CISC hardware execution is merely an "obvious engineering design choice" indicates that undue experimentation is necessary. As the Examiner notes in his 35 USC § 112, first paragraph rejection, "the design of such a decoder and execution units" [capable of both RISC and CISC], "if possible, is not clearly obvious to one skilled in the art without further requiring undue experimentation because the complexity of the system with such ability would increase significantly." Certainly Portanova does not provide an enabling disclosure for a combined RISC and CISC execution unit or decoder. Further, nowhere does Portanova suggest or teach that only a subset of the CISC instruction set is executed in hardware.

Invention as a Whole Not Obvious from the Combination

The references are deficient because none teach or suggest decoding and directly executing two instruction sets. Beyond these deficiencies, a strained combination of references has to be made, as Onishi does not pertain to dual-instruction set processors. The present invention solves the problem of flexibility by decoding two separate instruction sets, while Onishi solves the different problem of speed of decoding branch instructions in a single instruction set.

No suggestions or motivations in these prior-art references have been cited to justify such a combination. Other combinations are possible, and this undue experimentation has been cloaked as an "engineering design choice." Something in the prior art must suggest the desirability of making the combination. *Uniroyal, Inc. v. Rudkin-Wiley Corp.*, 837 F.2d 1044, 5 USPQ2d 1434, 1438 (CAFC 1988). If the prior art provides no teaching, suggestion, or incentive supporting the combination proposed by the Examiner, then the rejection is in error and must be reversed. *In Re Bond*, 910 F.2d 831, 15 USPQ2d 1566 (CAFC 1990). The claimed invention must not be used as a blueprint.

PRIOR ART REJECTIONS - 35 USC § 103 - PORTANOVA, ONISHI, BULLIONS

Claims 6-13 and 17 were rejected under 35 U.S.C. § 103 as being unpatentable over Portanova and Onishi as set forth above for claims 1-5, 14-16 and 18-20 further in view of Bullions, III et al, (hereafter Bullions) U.S. pat. no. 4,456,954.

As to claims 6-7, 9-10 and 17, neither Portanova nor Onishi teach using a translation look aside buffer (TLB) . Bullions teaches using a TLB for translating a virtual address to a physical address for both host and guest instructions (see abstract). In particularly, a TLB is utilized to address emulation host routine for a guest instruction. Bullions

further teaches that a miss in TLB also triggers a change of execution modes, i.e. from host to guest (see summary and claims). Applicant respectfully disagrees.

The arguments presented above apply with equal force and effect to dependent claims 6-7, 9-10 and 17.

Bullions teaches a CISC system that emulates guest architectures on a native architecture. Each level of architecture is capable of using virtual addressing with dynamic address translation. (col 1, lines 8-25). All operating systems described by Bullions are well-known CISC architectures (col 1, lines 29, 50). Thus Bullions does not teach processing both RISC and CISC instructions, but merely teaches emulating "guest" CISC architectures on a native CISC machine.

Bullions uses the word "architecture" to mean something other that "instruction set". His summary and claims refer to guest "programs" but not to guest "instructions" from a different "instruction set". A guest program does not necessarily use a different instruction set. Indeed, the "plural levels of architecture" referred to "involve plural levels of address translation." (col 5, lines 24-29). These plural architectures refer to address translation architectures, not to instruction set architectures. Bullions clearly states that "different architectures may use different size addresses, e.g. one architecture may use 24 bit addresses while another architecture may use 31 bit addresses." However, all operating systems described by Bullions use the same instruction set.

Thus the execute unit that is coupled to the TLB does not execute instructions from two instruction sets, but merely executes CISC instructions. Claim 6's limitation of a TLB " coupled to the execute means" is not taught by Bullions because Bullions executes single instruction set, but merely uses different address translation architectures for native and guest programs.

Further, claim 6 recites that the TLB provide "an indication to said mode control means to change said instruction set decoded to said first instruction set when no translation is found in said TLB". Bullions does not teach that another instruction set is decoded. Bullions teaches that the architecture "mode" is changed on a TLB miss, causing a native program rather than a guest program to execute.

It is thus improper to replace Bullions's architecture or program with the word "instruction", as his programs and architectures refer to different address translation architectures and not to different instruction sets. Bullions teaches guest programs and guest architectures, but not guest instructions from a different instruction set.

Claim 17 recites translating memory references generated by CISC instructions that are directly executed, where the translation of memory references is controlled by a software translator routine comprised of RISC instructions. Bullions fails to teach that RISC instructions are used in a software translator routine while some CISC instructions are directly executed. Thus claim 17 cannot be obvious in view of Bullions and the other references.

Additional Remarks

The remarks in paragraph 16 of the second office action referring to Portanova as teaching using either software or hardware is assumed to be referring to cols 29-30, which were shown above to teach away from the invention as Portanova's design methodology first builds a RISC hardware without regard to the CISC instruction set to be emulated, and then writes the CISC emulation software. Portanova achieves faster time to market by ignoring the CISC aspects of the design until the hardware is designed.

These remarks to refer to Onishi's two decoders as "clear evidence of a system employing partially duplicated hardware resources". It was also stated that whether the emulation unit is integrated or separate is a design of choice. Applicant's invention allows both instruction sets to be executed on a single execution unit, eliminating duplicated hardware for execution. Thus duplicated hardware resources are not needed for the execute unit, but only for the decoders. This approach is not suggested in any of the cited references and is not merely a design choice of separating units, as in a coprocessor, or integrating units.

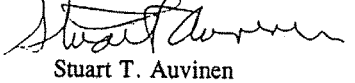
In view of the above, it is submitted that claims 1-20, as amended, are in a position for allowance. Applicant requests that the requirement for formal drawings be held in abeyance until allowance. Applicant believes that a full and complete response to the office action has been made. Reconsideration and re-examination is respectfully requested. Allowance of the claims at an early date is solicited.

If the Examiner believes that a telephone interview would expedite prosecution of this application, he is invited to telephone the undersigned at (408) 476-5506.

Stuart T. Auvinen
429 26th Avenue
Santa Cruz, CA 95062

(408) 476-5506
(408) 477-0703 Fax

Respectfully Submitted,



Stuart T. Auvinen
Agent for Applicant
Reg. No. 36,435

2315
230
11-16

6P2315

PATENT

AMENDMENT TRANSMITTAL LETTER			Docket No. RM-1
Serial No: 08/179,926	Filing Date: 1/11/94	Examiner: V. Vu	GAU: 2315
Title: Dual-Instruction-Set Architecture CPU with Hidden Software Emulation Mode			

Honorable Commissioner of Patents and Trademarks
Washington, DC 20231

RECEIVED
FEB 23 1995
GROUP 2300

Transmitted herewith is an amendment in the above-identified application.

☒ Small entity status of this application under 37 C.F.R. § 1.27 has been established by a verified statement previously submitted.

☒ No additional fee is required.

☐ The fee has been calculated as shown below:

	Claims remain. after Amend-ment		Highest No. prev. paid for	Present Extra	Rate	Add'l Fee	Rate	Add'l Fee
Total		Minus			x $11		x $22	
Indepen dent		Minus			x $37		x $74	
First Presentation of Multiple Dep. Claim					+ $115		+ $230	
					Total		Total	

☐ A check in the amount of $_____ to cover these fees is enclosed.

☐ Please charge my deposit account No. **01-2950** in the amount of $ _____ for these fees. A duplicate copy of this sheet is enclosed.

☒ The commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to deposit account No. **01-2950**. ~~A duplicate copy of this sheet is enclosed~~.

☒ Any additional filing fees required under 37 C.F.R. § 1.16.
☒ Any patent application processing fees under 37 C.F.R. § 1.17.

Respectfully Submitted,

Stuart T. Auvinen
Reg. No. 36,435
Agent for Applicant

Address all inquires to:
Stuart T. Auvinen
429 26th Ave.
Santa Cruz, CA 95062
(408) 476-5506

I hereby certify that this paper is being deposited with the United States Postal Service with sufficient postage as First Class Mail under 37 CFR 1.8 on the date indicated below and is addressed to: "Commissioner of Patents and Trademarks, Washington, DC 20231."

Signed _____
Stuart T. Auvinen, Reg. No. 36,435

Date 2/14/95