DECLARATION OF GORDON MACPHERSON

I, Gordon MacPherson, am over twenty-one (21) years of age. I have never been convicted of a felony, and I am fully competent to make this declaration. I declare the following to be true to the best of my knowledge, information and belief:

1. I am Director, Board Governance & Policy Development of The Institute of Electrical and Electronics Engineers, Incorporated ("IEEE").

2. IEEE is a neutral third party in this dispute.

3. I am not being compensated for this declaration and IEEE is only being reimbursed for the cost of the article I am certifying.

4. Among my responsibilities as Director, Board Governance & Policy Development, I act as a custodian of certain records for IEEE.

5. I make this declaration based on my personal knowledge and information contained in the business records of IEEE.

6. As part of its ordinary course of business, IEEE publishes and makes available technical articles and standards. These publications are made available for public download through the IEEE digital library, IEEE Xplore.

7. It is the regular practice of IEEE to publish articles and other writings including article abstracts and make them available to the public through IEEE Xplore. IEEE maintains copies of publications in the ordinary course of its regularly conducted activities.

8. The article below has been attached as Exhibit A to this declaration:

A.	L. Kohn; N. Margulis; "Introducing the Intel i860 64-bit microprocessor", published in Published in: IEEE Micro (Volume: 9, Issue: 4, page(s) 15 - 30), date of publication August 1989.

9. I obtained a copy of Exhibit A through IEEE Xplore, where it is maintained in the ordinary course of IEEE's business. Exhibit A is a true and correct copy of the Exhibit, as it existed on or about May 11, 2023.

10. The article and abstract from IEEE Xplore show the date of publication. IEEE Xplore populates this information using the metadata associated with the publication.

11. L. Kohn; N. Margulis; "Introducing the Intel i860 64-bit microprocessor", published in Published in: IEEE Micro (Volume: 9, Issue: 4, page(s) 15 - 30), date of publication August 1989. Copies of this publication was made available no later than the last day of the publication month. The article is currently available for public download from the IEEE digital library, IEEE Xplore.

12. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true, and further that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001.

I declare under penalty of perjury that the foregoing statements are true and correct.

Executed on: 5/11/2023

DocuSigned by:
Gordon Macpherson
E768DB210F4E4EF...

EXHIBIT A

Introducing the Intel i860 64-Bit Microprocessor

A million-transistor budget helps this RISC deliver balanced MIPS, Mflops, and graphics performance with no data bottlenecks.

Les Kohn
Neal Margulis

Intel Corp.

The single-chip i860 CPU—a 64-bit, RISC-based microprocessor—executes parallel instructions using mainframe and supercomputer architectural concepts. We designed the 1,000,000-transistor, 10 mm × 15 mm processor (see Figure 1 on the next page) for balanced integer, floating-point, and graphics performance, using the company's latest generation CAD tools and 1-micrometer semiconductor process.

To accommodate our performance goals, we divided the chip area evenly between blocks for integer operations, floating-point operations, and instruction and data cache memories. Inclusion of the RISC (reduced instruction set computing) core, floating-point units, and caches on one chip lets us design wider internal buses, eliminate interchip communication overhead, and offer higher performance. As a result, the i860 avoids off-chip delays and allows users to scale the clock beyond the current 33- and 40-MHz speeds.

We designed the i860 for performance-driven applications such as workstations, minicomputers, application accelerators for existing processors, and parallel supercomputers. The i860 CPU design began with the specification of a general-purpose RISC integer core. However, we felt it necessary to go beyond the traditional 32-bit, one-instruction-per-clock RISC processor. A 64-bit architecture provides the data and instruction bandwidth needed to support multiple operations in each clock cycle. The balanced performance between integer and floating-point computations produces the raw computing power required to support demanding applications such as modeling and simulations.

Finally, we recognized a synergistic opportunity to incorporate a 3D graphics unit that supports interactive visualization of results. The architecture of the i860 CPU provides a complete platform for software vendors developing i860 applications.

Architecture overview. The i860 CPU includes the following units on one chip (see Figure 2):

- the RISC integer core,
- a memory management unit with paging,
- a floating-point control unit,
- a floating-point adder unit,
- a floating-point multiplier unit,
- a 3D graphics unit,

0272-1732/89/0800-0015$01.0 © 1989 IEEE

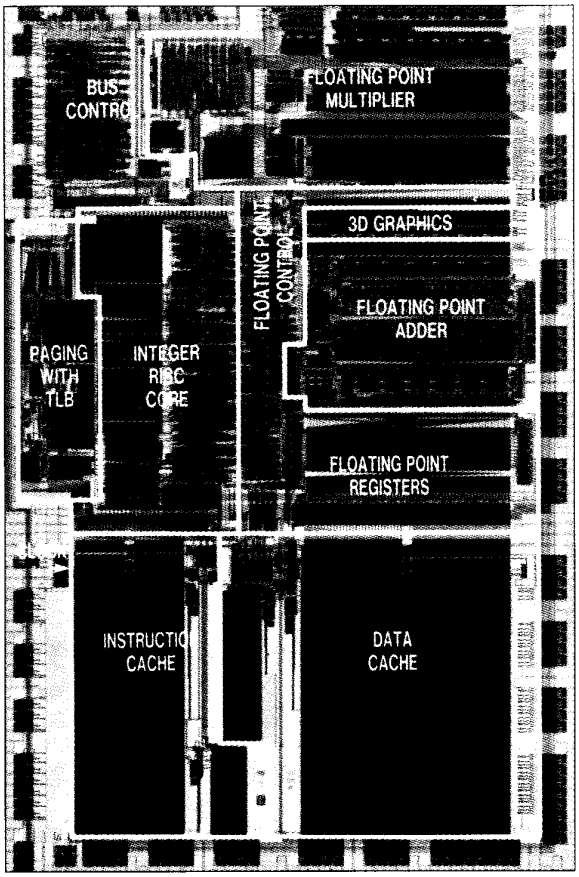

Figure 1. Die photograph of the i860 CPU.

- a 4-Kbyte instruction cache,
- an 8-Kbyte data cache, and
- a bus control unit.

Parallel execution. To support the performance available from multiple functional units, the i860 CPU issues up to three operations each clock cycle. In single-instruction mode, the processor issues either a RISC core instruction or a floating-point instruction each cycle. This mode is useful when the instruction performs scalar operations such as operating system routines.

In dual-instruction mode, the RISC core fetches two 32-bit instructions each clock cycle using the 64-bit-wide instruction cache. One 32-bit instruction moves to the RISC core, and the other moves to the floating-point section for parallel execution. This mode allows the RISC core to keep the floating-point units fed by fetching and storing information and performing loop control, while the floating-point section operates on the data.

The floating-point instructions include a set of operations that initiate both an add and a multiply. The add and multiply, combined with the integer operation, result in three operations each clock cycle. With this fine-grained parallelism, the architecture can support traditional vector processing by software libraries that implement a vector instruction set. The inner loops of the software vector routines operate up to the peak floating-point hardware rate of 80 million floating-point operations per second. Consistent with RISC philosophy, the i860 CPU achieves the performance of hardware vector instructions without the complex control logic of hardware vector instructions. The fine-grained parallelism can also be used in other parallel algorithms that cannot be vectorized.

Register and addressing model. The i860 microprocessor contains separate register files for the integer and floating-point units to support parallel execution. In addition to these register files, as can be seen in Figure 3 on page 18, are six control registers and four special-purpose registers. The RISC core contains the integer register file of thirty-two 32-bit registers, designated R0 through R31 and used for storing addresses or data. The floating-point control unit contains a separate set of thirty-two 32-bit floating-point registers designated F0 through F31. These registers can be addressed individually, as sixteen 64-bit registers, or as eight 128-bit registers. The integer registers contain three ports. Five ports in the floating-point registers allow them to be used as a data staging area for performing loads and stores in parallel with floating-point operations.

The i860 operates on standard integer and floating-point data, as well as pixel data formats for graphics operations. All operations on the integer registers execute on 32-bit data as signed or unsigned operations and additional add and subtract instructions that operate on 64-bit-long words. All 64-bit operations occur in the floating-point registers.

The i860 microprocessor supports a paged virtual address space of four gigabytes. Therefore, data and instructions can be stored anywhere in that space, and multibyte data values are addressed by specifying their lowest addressed byte. Data must be accessed on boundaries that are multiples of their size. For example, two-byte data must be aligned to an address divisible by two, four-byte data on an address divisible by four, and so on, up to 16-byte data values. Data in memory can be stored in either little-endian or big-endian format. (Little-endian format sends the least significant byte, D7-D0, first to the lowest memory address, while big-endian sends the most significant byte first.) Code is always stored in little-endian format. Support for big-endian data allows the processor to operate on data produced by a big-endian processor, without performing a lengthy data conversion.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.