

x6818.2013

Intel Corp

THE COMPUTER HISTORY MUSEUM



1 027 7683 0



INTEL MILITARY i960™ ARCHITECTURE

**ADVANCED
AVIONICS
SOLUTIONS**

**TECHNICAL
BACKGROUND
INFORMATION**

intel®

Order Number: 271183-001

Intel's Military i960™ Architectures: Advanced Avionics Solutions

TABLE OF CONTENTS

Defining State-of-the-Art: The Military i960™ Family.....	2
Embedded Systems Requirements	2
Rigorous Requirements in Avionics	2
A Designer's Paradox	2
The i960™ Architecture Philosophy	3
CISC and RISC Synergy	3
Foresight and Extensibility	3
Architecture Levels	3
The Commercial Architectures	3
The Military Architectures	4
The i960™ Processor Implementations.....	4
Flexible Architecture - Multiple Implementations.....	5
The Military i960™ Architectures	6
The Protected Architecture	7
The Extended Architecture	7
The Military i960™ Processor Implementations	8
First-Generation Implementations	8
Second-Generation Implementations.....	9
Superscalar Primer.....	10
Performance Figures	11
The Cost of Performance Processors and a Solution.....	12
Support for the Military Architectures	12
What's Next ?	13

Defining State-of-the-Art: The Military i960™ Family

The introduction of the second generation military i960™ architectures from Intel marks the beginning of a new age in avionics systems technology. The launch coincides with the announcement of two key military airframe awards: the Air Force's Advanced Tactical Fighter (ATF) program and the Army's Light Helicopter (LH) are two highly visible programs that are leading the way in defining "state-of-the-art" avionics for the 1990s and beyond. These JIAWG¹ platforms mark the culmination of years of research and development, debate and engineering progress that have created the most advanced aircraft the world has ever seen. Intel's i960 architecture is also leading the way in avionics processing technology by meeting the rigorous requirements of avionics systems on these world-class aircraft.

Intel's first generation military version of the i960 32-bit processor architecture, the i960 MC (originally called the 80960MC), has accumulated over 110 designs in only three years - an obvious testimonial to the architecture's ability to meet the critical requirements of modern avionics designs. As the next generation of aircraft take shape, Intel's i960 architecture continues to define state-of-the-art in processors for avionics systems, as evidenced by the inclusion of the next generation military i960 processor, the superscalar² i960 MX, as the heart of the avionics systems on the ATF and LH platforms.

Embedded Systems Requirements

Intel long ago recognized the trend in embedded applications toward real-time systems that perform a variety of tasks that require ever-increasing performance while meeting the difficult constraints on space, power, size and weight. And in the avionics industry, reliability is important as well. To meet these requirements, Intel developed the 32-bit i960 architecture.

¹ JIAWG (Joint Integrated Avionics Working Group) is discussed in the sidebar titled "Industry Standardization Efforts."

² Superscalar refers to the ability to dispatch and execute multiple instructions per clock cycle through the provision of multiple processing resources.

Rigorous Requirements in Avionics

As a special subset of embedded applications, modern military and commercial avionics systems perform functions such as flight control, thrust management, navigation, stores management, real-time guidance, display control, terrain mapping, radar control and the like. Often these systems can mean the difference between mission accomplished and mission aborted - or worse. These systems have other rigorous constraints: serviceability, upgradeability and high reliability, all in a harsh environment with constraints on space, power, size and weight.

Further, the complex avionics systems must be easy to design and software tools must adequately support and utilize the processor's capabilities. Finally, the processor that forms the heart of these systems should be a widely accepted industry standard.

To further complicate system design, many military systems place special requirements on the avionics, such as fault detection, data security, multi-processing, and on-board (or on-chip) test and maintenance capabilities.

A Designer's Paradox

Designers want high performance, industry standard, highly integrated processors that are well supported and are well suited to the embedded real-time environment. They want choices that will deliver the performance and functionality they need while keeping costs within budgets and meeting environmental constraints.

Apparently a dichotomy, how can designers of embedded systems (like avionics) achieve the performance and functionality they need (which are continually rising), while making the system fit within the environmental restrictions imposed by the platform (such as a fighter plane or a missile)? Further, is the system based on an accepted industry standard that is well supported and has assured longevity - an upgrade path?

The following sections will discuss how this paradox of embedded system design requirements was approached by Intel in its design of the i960 military architectures. The discussion shows that the architecture has set a foundation for a family of products that will meet price, performance and integration demands well into the future and that will

meet the technical and design ease requirements set forth by the industry as a whole.

The i960™ Architecture Philosophy

Since inventing the world's first microprocessor in 1971, Intel has continued to lead the field of embedded computing. The i960 microprocessor family is an outgrowth of a massive research and development project initiated in the late 1970's. By the early 1980's, Intel had developed the i960 architecture -- an innovative, extensible 32-bit architecture without constraints on future performance or functionality. Samples of i960 microprocessor products were in customer hands by 1985.

CISC and RISC Synergy

Intel's i960 architecture was developed with two primary goals in mind. The first was to combine the system-level benefits of CISC (complex instruction set computer) architectures with the implementation and performance efficiencies of newer RISC (reduced instruction set computer) designs. The second goal was to define an architecture that could support a broad family of product designs and span a wide range of implementation techniques to permit the architecture to meet the diverse performance and functionality requirements of embedded systems applications.

The resulting i960 architecture is a hybrid of several design philosophies and implementation techniques. CISC architectures inspired a rich, full-featured instruction set, a range of memory-operand addressing modes, and efficient, consistent subroutine calling mechanisms for increased code density and programming flexibility. From the RISC arena came a basic memory load/store architecture, a large internal register file, and uniform instruction formats for simplified decode.

For maximum performance Intel enhanced the merged CISC/RISC architecture with several innovative features. Each implementation includes heavily overlapped instruction execution and multiple-ported register files. Register scoreboarding provides hardware-enforced pipeline interlocking to insulate the programmer from data dependency jeopardies. This scoreboarding technique is also applied to processing units to allow efficient

dispatching of instructions and effective use of available resources.

Foresight and Extensibility

From the start, the i960 architecture made provisions for increased processor parallelism, out-of-order instruction execution, and hardware support for software debugging -- facilities that Intel anticipated would become increasingly important as fabrication technology advanced and system complexity increased. The architecture also left room for new instructions and special function registers (SFRs) on future implementations so that on-chip coprocessors, peripherals and control units could be efficiently supported as extensions of the core architecture.

Intel has succeeded at meeting and exceeding the original goals by creating a single, unified microprocessor architecture that combines the best of many design principles. The i960 architecture delivers the price/performance ratio as well as the ease of design and suitability to a range of tasks that designers require.

Architecture Levels

The i960 architecture was designed from the beginning to meet the needs of a range of embedded applications. Designers are not required to design with more or less functionality than they need. To accomplish this, Intel defined four distinct levels of the i960 architecture, as depicted in Figure 1.

The Commercial Architectures

The inner ring in Figure 1 represents the *core* i960 architecture level and provides the basic integer execution environment upon which all i960 implementations are based. The next ring represents the *numerics* level of the architecture and provides full support for the IEEE 754 floating point arithmetic standard. The two inner rings together form the basis for all of Intel's commercial i960 processor offerings. The specific products based on these two levels of the architecture are discussed in the section titled "The i960™ Processor Implementations."

The Military Architectures

The third ring represents the *protected* level of the architecture. It builds on the numerics level by providing full virtual memory management and multi-

Industry Standardization Efforts

There are a number of industry standards that significantly impact the way in which military and commercial avionics systems are designed and implemented. These standards range from processor architecture to communication protocols, from backplanes to programming languages.

Of particular importance in military avionics are the JIAWG (Joint Integrated Avionics Working Group) efforts to standardize on a modular avionics environment. JIAWG is a tri-service task force assigned the task of implementing "Pave Pillar" modular avionics standards for next-generation military aircraft, including ATF and LH. JIAWG's role included defining such items as a 32-bit standard instruction set architecture, an avionics processor board form, fit and function standard (called CAP-32, for Common Avionics Processor - 32-bit) and a standard parallel intermodule backplane protocol (now known as the PI Bus standard).

Other standards initiatives are significant to the industry as well. In the programming arena, the DoD decided to standardize on the Ada programming language (now defined in MIL-STD-1815). In the commercial avionics arena, committees within ARINC (Airline Radio, Inc) are tasked with defining modular avionics standards for future commercial aircraft. The Society of Automotive Engineers (SAE) also has committees that manage communication, processor and backplane standards for military aircraft.

These industry standardization efforts are significant to both the avionics industry and to suppliers of electronics for the industry. These standards were put in place to benefit in such areas as portability of software and hardware across platforms; reduced maintenance and inventory costs; improved quality and availability of avionics components; and guaranteed minimum levels of performance and functionality. Intel has participated in the activities of JIAWG, ARINC and SAE and supports the standards they have created, both in principle and with products such as the i960 and Pi Bus architectures.

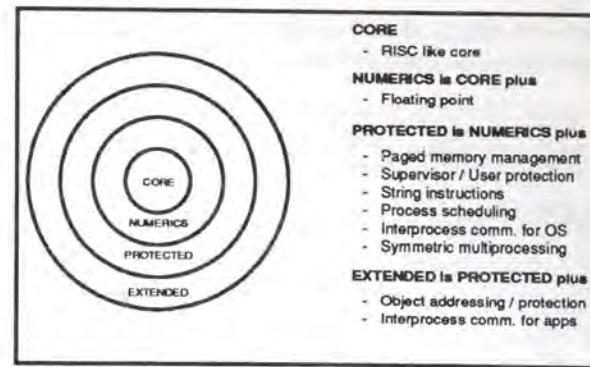


Figure 1: Intel's i960™ Architecture Levels

tasking process management capabilities to support the Ada³ tasking model. And finally, the outer ring represents the *extended* level of the i960 architecture. The extended architecture adds to the functionality of the protected architecture by adding features for very high levels of hardware-enforced data security and support for object-oriented programming in hardware. JIAWG selected the i960 extended architecture as a 32-bit ISA standard (see sidebar titled "Industry Standardization Efforts").

The outer two rings of the i960 architecture together form the architectures upon which all military implementations of the i960 family are based. The specific implementations of these two levels of the architecture are discussed in the sections titled "The Military i960™ Architectures" and "The Military i960™ Processor Implementations."

The i960™ Processor Implementations

Beyond the definition of levels within the i960 architecture discussed in the previous section, each level can be implemented in a variety of ways to provide exactly the level of functionality and performance required. The flexibility of the i960 architecture has allowed the creation of many implementations that provide solutions for a range of embedded applications. Designers can choose one part for applications where space and power constraints are severe yet good performance is still required, and another part for applications where

³ Ada is a trademark of the U.S. Department of Defense and is the standard high-level language mandated for use in government-funded military programs.

performance is the driving requirement but good integration and efficient use of space is still required.

The various implementations of the i960 architecture are the result of the definition of a flexible architecture with unlimited potential and longevity. Future implementations can incorporate increases in integration, functionality and performance, all while maintaining core ISA upward compatibility. See the sidebar titled "Architecture vs. Implementation" for more information on the difference between an architecture and an implementation.

Flexible Architecture - Multiple Implementations

There are now nine entries in the i960 microprocessor family. Rest assured that more are on the way that

Architecture vs. Implementation

It is important to distinguish between an architecture and an implementation of an architecture, particularly regarding the i960 family.

An architecture, or ISA, consists of those features that ensure commonality through generations of implementations. The concept of an architecture is perhaps best understood by considering what an architecture is not: an architecture does not define the time or speed of any operation, the internal partitioning of processing units, the electrical or physical organization of the hardware, the circuits, components or peripherals of the CPU, the manufacturing technology, the memory subsystem organization or the bus topology.

Essentially, an instruction set architecture is the programming model. It includes the instruction set, format and opcodes, addressing modes, valid data types, memory management scheme, internal interrupt handling structure and register formats and manner of use. All of these characteristics are defined by an ISA.

The differentiation between an architecture and its implementation is crucial. It permits new generations of highly functional implementations to improve on the performance and integration of prior generations while maintaining a common programming model and upward binary compatibility, as in the i960 architecture family.

will continue to meet the ever-increasing demands of embedded systems. The first three entries were introduced in April, 1988 and included the i960 KA, i960 KB and i960 MC (originally named 80960KA, 80960KB and 80960MC).

The i960 KA implements the core i960 architecture, and provides the most cost-effective, full 32-bit solution for applications requiring moderate integer performance levels. With a 25 MHz system clock, the i960 KA typically provides 20 to 25 peak native MIPS (million instructions per second) or about 10 VAX MIPS⁴. The i960 KB adds a full 80-bit IEEE 754-compatible floating-point coprocessor on-chip for greater throughput in math intensive applications. The military i960 MC includes the floating point unit of the i960 KB and adds virtual memory management and protection hardware plus new instructions to support a full real-time multitasking Ada environment for military systems. The i960 MC is fully qualified to MIL-STD-883C Level B. Further details on the i960 MC processor can be found in the sections titled "The Military i960™ Architectures" and "The Military i960™ Processor Implementations."

In September, 1989, a fourth member of the family was announced. The i960 CA is a second-generation commercial implementation of the core i960 architecture (i.e., it runs the same instruction set as the i960 KA) and was the first microprocessor ever to apply superscalar program execution techniques to break the single-instruction-per-clock (1.0 IPC) barrier. The i960 CA delivers three to five times the performance of the i960 KA and has been extended to include new instructions and SFRs to support the on-chip DMA and interrupt control units.

In September, 1990, low-cost implementations of the core and numerics levels of the architecture were introduced: The i960 SA and i960 SB exactly implement the i960 KA and i960 KB instruction sets, respectively, but provide a memory bus interface designed to reduce external system cost and complexity. The data bus on these S-series parts is 16-bits wide for connection to lower-cost 16-bit external memory systems. The parts maintain the performance advantages of their internal 32-bit architecture while preserving the system cost advantages of simpler 16-bit designs.

⁴ One VAX MIP is equivalent to the performance of a Digital Equipment Corp. VAX 11/780.

The spring of 1991 saw the introduction of three implementations of the military architectures (i.e., the protected and extended levels), bringing to nine the number of i960 architecture implementations.

First, the i960 XA, which is a first-generation implementation of the extended architecture, was announced. The i960 XA, which executes the JIAWG standard 32-bit ISA and is MIL-STD-883C Level B qualified, has actually been in existence since the i960 KA, i960 KB and i960 MC were launched in 1988.

Coincident with the ATF and LH platform awards, the superscalar i960 MX and i960 MM processor implementations of the extended and protected levels of the architecture, respectively, were announced.

The i960MM and i960MX are the first military processors ever introduced that utilize superscalar processing techniques. The i960 MM and i960 MX, capable of sustaining in excess of 25 VAX MIPS at 25 MHz, are likely the highest performance general-purpose avionics processors in the industry. Details on these three military implementations of the i960 architecture are included in the section titled "The Military i960 Architectures" and "The Military i960™ Processor Implementations."

Product	Architecture Level	Integration Level	Instruction Cache Size	Local Register Cache	RAM/Data Cache Size	External Bus Width
i960 SA	Core	32-bit core CPU	512 bytes	4 sets	N/A	32-bits addr 16-bits data
i960 SB	Numerics	32-bit CPU + 80-bit FPU	512 bytes	4 sets	N/A	32-bits addr 16-bits data
i960 KA	Core	32-bit core CPU	512 bytes	4 sets	N/A	32-bitd addr/data
i960 KB	Numerics	32-bit CPU + 80-bit FPU	512 bytes	4 sets	N/A	32-bits addr/data
i960 CA	Core	Superscalar CPU + DMA	1 Kbytes	5-15 sets	1 Kbytes data RAM	32-bits addr 8, 16 or 32 bits data
i960 MC	Protected	CPU + FPU + MMU	512 bytes	4 sets	N/A	32-bits addr/data
i960 XA	Extended	CPU + FPU + MMU + Objects	512 bytes	4 sets	N/A	32-bits addr/data
i960 MM	Protected	Superscalar CPU + FPU + MMU	2 Kbytes	8 sets	2 Kbytes data cache	32-bits addr/data 64-bits backside
i960 MX	Extended	Superscalar CPU + FPU + MMU + Objects	2 Kbytes	8 sets	2 Kbytes data cache	32-bits addr/data 64-bits backside

Table 1: The i960™ Processor Family

Table 1 summarizes all of the Intel i960 microprocessor family members.

The Military i960™ Architectures

The military levels of the i960 architecture (i.e., the protected and extended levels) provide the bases for implementations that are ideally suited to a range of military and other high-reliability applications. These applications include military and commercial avionics, missiles, radar systems and communication equipment.

A major endorsement of the i960 architecture's suitability to task came in 1989 with JIAWG's selection of the i960 extended architecture as a 32-bit ISA standard for military avionics. That endorsement was reinforced when it was announced that the winning ATF and LH proposals include the new military i960 implementations at the heart of their avionics suites. The integrity of the avionics had a tremendous impact on the credibility and successes of those proposals.

It is both the functionality designed in at the architectural level and the performance and integration designed in at the implementation level that make the military i960 architectures so suitable for applications like advanced avionics. This section discusses the architecture-level features of the military i960 architectures.

The Protected Architecture

The protected level of the i960 architecture takes the numerics level several steps further. The same extensive procedure call, fault handling and debugging mechanisms that are present in the commercial levels of the architecture are maintained in the military levels. The built-in priority interrupt controller and the full IEEE-compatible floating point support are also included in the military architectures. To that already powerful framework are added several extensions.

First, support for decimal and string data types are included in the protected architecture model. These data types and the special instructions associated with them reduce the amount of code required to handle strings or long bit patterns.

A second key addition included in the protected level is support for demand-paged, virtual-memory management, which allows each process (or task) to reach an address space of up to 2^{32} bytes (4 Gigabytes). The virtual-to-physical address translation is handled automatically.

The protected level also provides two mechanisms for protecting critical data structures or software modules which is critical to the integrity of many avionics systems. The first is the ability to use page rights to restrict access to individual pages in memory. The second mechanism is a user/supervisor protection model which provides hardware enforced protection of secure kernel procedures and data structures.

Another important extension that the protected level of the i960 architecture offers is multitasking support. Specific instructions and data structures are dedicated to provide a complete multiple process management capability, including priority-driven process scheduling, process timing and interprocess communications. These facilities map very cleanly to the Ada tasking model, an important benefit for military and commercial avionics applications that are required to develop software in the Ada programming language.

A final extension to the commercial architectures included in the protected level is support for multiprocessing. Several mechanisms are designed into the architecture that simplify the design of multiple-processor systems. Some of the support is included in the multitasking support described above, through the use of a shareable data structure that allows processes to be automatically dispatched to available processors. Beyond this, data structure entries can be locked to ensure their integrity in a multiple processor environment.

The protected level of the i960 architecture contains many useful extensions over the commercial levels of the architecture. These enhancements allow designers of complex avionics functions to easily design hardware and develop software that meet and exceed the requirements of many advanced aircraft systems.

The Extended Architecture

The extended architecture encompasses the complete functionality of the architecture as it was defined. It is a superset of the protected level and includes such architectural features as floating-point support, memory management, multitasking support, etc. However, it introduces the concept of objects (i.e., data structures) along with extremely high levels of hardware-enforced data security. Several additional instructions are included to support the object-oriented extended architecture programming model.

The extended architecture incorporates the concept of domains that encapsulate, or combine, objects with procedures that are dedicated to manipulating those objects. This encapsulation provides each domain its own private address space and prevents unauthorized manipulation of objects by procedures that should not have access to them.

An example of the protection offered by domains is helpful. Assume two procedures exist that are intended to manipulate two dynamic arrays - one procedure for each array. By encapsulating one procedure and one array in each of two distinct domains, access to an array is given only to the appropriate procedure. Imagine the scenario where one array contains an enemy aircraft's current position, and the other array contains a friendly aircraft's position. The domain protection prevents these two arrays from ever being interchanged, an error that could result in the targeting of a friendly aircraft.

While extreme, the example illustrates the utility of domains. The robust partitioning permitted by domains can be very useful in environments where a processor is tasked with many chores, such as in general-purpose avionics modules. Objects (data structures, files, etc...) are protected from corruption by unauthorized procedures. The enforcement of this partitioning is handled in the hardware with very little processor overhead. Traditional methods of such degrees of partitioning were handled in software and required significant processor resources to manage.

There are several other architecture-level features in the extended architecture. The ability to *type* objects also provides a level of security by limiting access to objects of a particular type to a *type manager* of the same type. Type managers are the interface mechanism between procedures and objects that offer further partitioning. The type manager insures that any procedure trying to access or manipulate an object has a key (called an *access descriptor*) and that it does not violate its access rights to that object (i.e., read only or read/write privilege).

An access descriptor is simply a 32-bit pointer to memory. It is identified in hardware as an *access descriptor* by the concatenation of a 33rd bit, called the *tag bit*. Access descriptors, or ADs, are unforgeable keys that can provide access to objects. ADs are distributed and managed by the type managers.

One further benefit of the extended architecture programming model is that it expands the virtual addressing range to 2^{58} bytes (256 million Gigabytes). Explicit virtual addressing permits the mapping of many processes, procedures and objects into a smaller, yet partitioned physical address space.

The extended architecture provides an unprecedented level of data security, robust partitioning, encapsulation and object-oriented addressing capability. Partitioning and security are increasingly useful and important in the realm of advanced

avionics systems where protection of critical data is crucial.

The Military i960™ Processor Implementations

The architectures define the core capabilities of the i960 family. But it takes innovative implementation techniques to deliver performance and integration to the end application. This section highlights the different implementation techniques and features of the two existing generations of i960 military processors. Figure 2 illustrates the upward compatibility from the first generation to the second and the mapping of the four military implementations to the two military architectures.

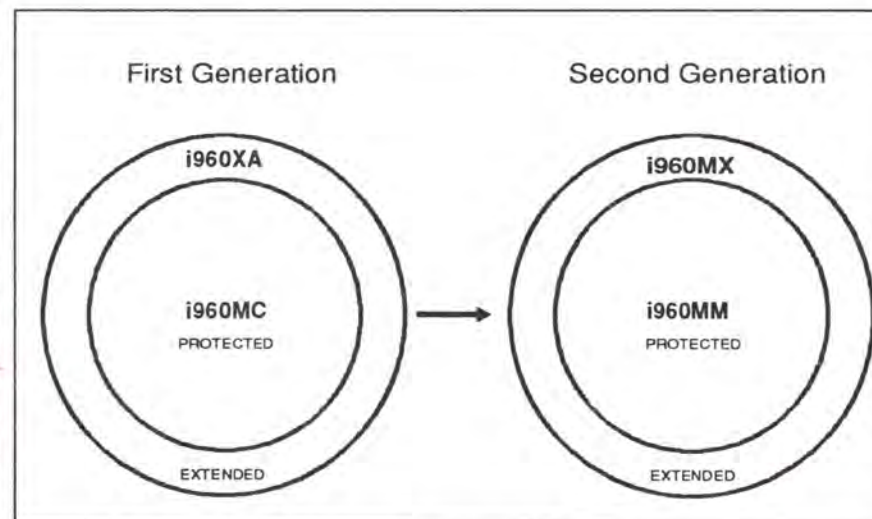


Figure 2: The Military i960™ Processor Implementations

First-Generation Implementations

The first generation of military i960 architecture implementations are incorporated in products called the i960 MC and the i960 XA. This generation of silicon was announced in 1988, along with the i960 KA and i960 KB. These two products deliver unprecedented levels of integration to avionics systems and other high-reliability embedded applications at a system price/performance ratio that rivals that of even 16-bit designs.

The only hardware difference between the two first-generation military products is that the i960 XA multiplexes a pin to implement the 33rd, or tag bit where the i960 MC does not. This necessitates supporting a 33-bit wide memory subsystem in an i960 XA application. Otherwise, both the i960 MC and i960 XA have identical implementation-specific features with the only difference being in the programming model, or ISA that they implement. These implementation-specific features, which are unique to the first-generation of silicon, are described below.

The first generation implementations are currently available in a 132-lead ceramic pin-grid array package for through-hole insertion and in a 164-lead ceramic quad flatpack package for surface mount applications. The i960 MC and i960 XA processors are manufactured using Intel's CHMOS-IV 1-micron fabrication technology.

The first entries in the military i960 family include a 512-byte instruction cache, a 48-entry translation lookaside buffer for storing frequently used virtual-to-physical address translations, sixteen global registers, four sets of sixteen local registers (to permit efficient, nested procedure calling without accessing off-chip memory), four 80-bit floating-point registers and a multiplexed 32-bit address/data bus capable of burst accesses at over 66 MBytes/second at 25 MHz clock rates. An on-chip self-test mechanism provides

coverage of over 50 percent of the 375,000 transistors to provide system-level confidence testing at start-up. Figure 3 shows a block diagram of the first-generation implementations of the military i960 architecture levels.

Second-Generation Implementations

The second generation of military i960 architecture implementations are incorporated in products called the i960 MM and the i960 MX. This generation of silicon was publicly announced in May, 1991, coincident with the ATF and LH platform awards. However, samples of the products have been in trendsetting program applications since July, 1990. The i960 MM and i960 MX are the first superscalar military microprocessors introduced.

Again, the only hardware difference between the two second-generation military products is that the i960MX multiplexes a pin to implement the 33rd, or tag bit. This necessitates supporting a 33-bit wide memory subsystem in an i960MX application. Otherwise, both the i960MM and i960MX processors have identical implementation-specific features, with their sole difference in the programming model, or ISA implemented. These implementation-specific features, which are unique to the second-generation of silicon, are described below.

The second-generation implementations are available

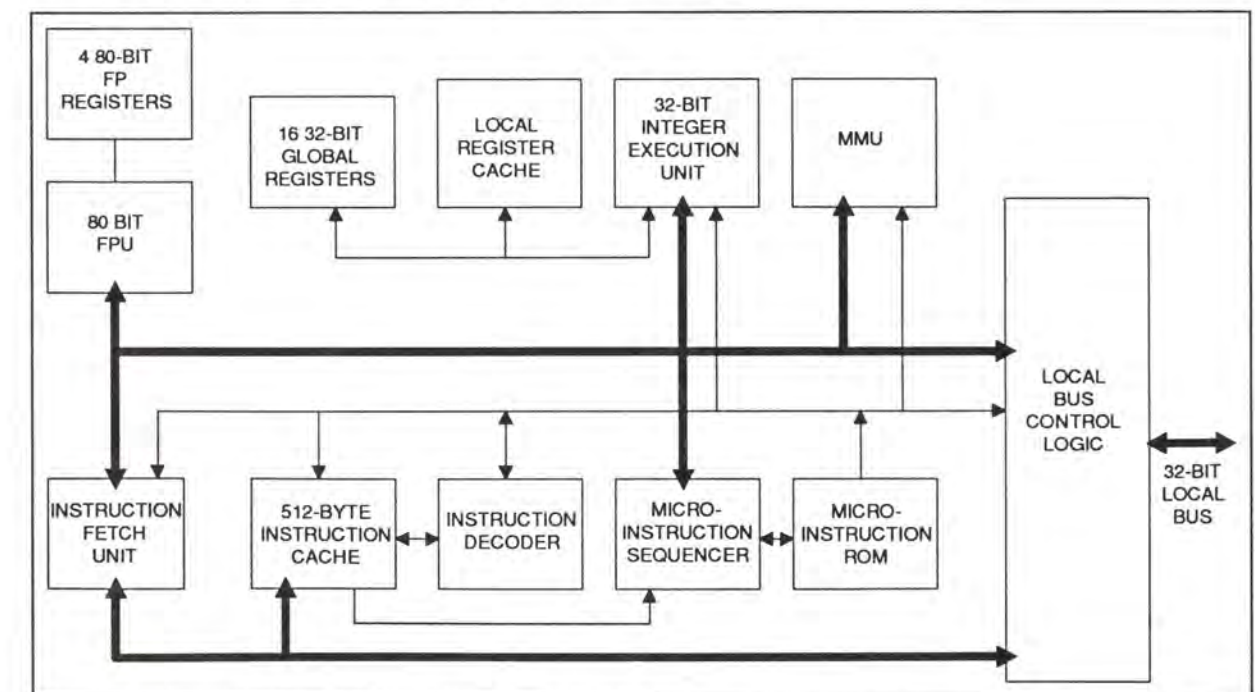


Figure 3: First-Generation i960™ Architecture Block Diagram

in a 348-lead ceramic pin-grid array package. The i960 MM and i960 MX processors are manufactured using Intel's 1-micron CHMOS-IV fabrication technology.

These entries in the military i960 family include a 2 Kbyte instruction cache, a 2 Kbyte data cache, a 64-entry translation lookaside buffer for storing frequently used virtual-to-physical address translations, sixteen global registers, eight sets of sixteen local registers (to permit efficient nested procedure calling without accessing off-chip memory), four 80-bit floating-point registers, a multiplexed 32-bit address/data bus capable of burst accesses, and an innovative, high-bandwidth (over 200 Mbytes/second at 25 MHz) backside bus to ensure ready availability of instructions and data to the processor. Five independent parallel execution units coupled with an instruction scheduler that decodes four instructions per clock cycle provide the high throughput of these superscalar processors. Built-in self-test provides approximately 85 percent coverage of the nearly 1.5 million transistors on the device for in-system confidence tests. Figure 4 shows a block diagram of the second-generation implementations of the military i960 architecture levels.

The backside bus of the i960MM and i960MX provides a high-speed channel to fast external SRAM arrays. The 64-bit bus can return two 32-bit words every clock cycle from the array. This local memory array can be configured as either second-level instruction cache, second-level data cache, a combination of the two, or as private memory. Private memory does not require a replacement or coherency algorithm as cache does. Instead, it can be loaded at startup to provide critical routines with the highest available throughput and deterministic performance. This is particularly useful for critical interrupt routines or other procedures that run frequently.

If the backside array is configured as second-level cache (supplementing the internal data and instruction caches), the processor uses the local bus in the event of a second-level cache miss. For performance-critical programs, large caches or private memory are very beneficial.

Superscalar Primer

Perhaps the most innovative and noticeable feature shown in Figure 4 is the existence of multiple, parallel execution units. Two floating point units (one for addition/subtraction, and one for multiplication/division), two integer execution units

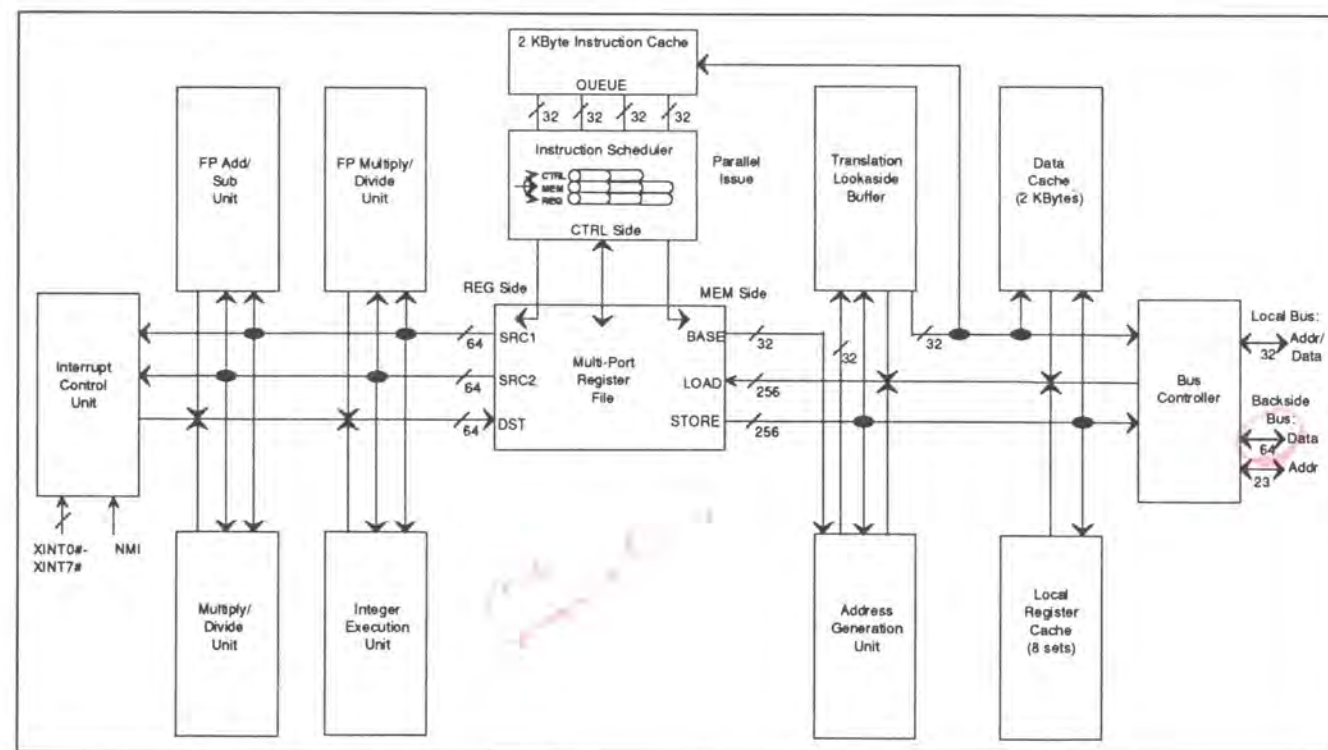


Figure 4: Second-generation Military i960™ Architecture

and a separate address calculation unit, all of which can execute in parallel, coupled with a multi-ported register file, very wide (up to 256 bits) internal data paths and an intelligent on-chip instruction scheduler combine to create one of the highest performance microprocessors available.

The existence of multiple processing resources is the essence of superscalar execution techniques. In the i960MM and i960MX processors, four instructions are fetched and decoded and up to three instructions can begin execution every clock cycle. As many as eight instructions can be in execution at a given time.

Superscalar execution techniques can reduce the cost and design burden of continually increasing the clock rates on traditional processors. Increased clock rates necessitate faster, more expensive memory and peripheral subsystems and complicate system designs. By devising a way to execute more instructions in a clock cycle, Intel has reduced this design burden while increasing processing throughput. The move to new generations of higher-performance processors does not necessitate the move to ever increasing clock frequencies.

Performance Figures

The true measure of the performance of a microprocessor is ideally measured in the end application. However, to demonstrate the capabilities of the i960 family, it is useful to study a few examples and features. The performance of the military i960 processors is an implementation-dependent characteristic. This section provides some performance figures for the military i960 processors, and discusses some of the enhancements made in the superscalar implementations.

One common operation performed in most aircraft guidance and navigation systems is the multiplication of matrices composed of real numbers. In a test of the first- and second-generation military i960 processors, two 10 x 10 matrices consisting of double-precision (64-bit) values were multiplied together. It was assumed that the matrices already existed. The i960 MC and i960 XA microprocessors performed the operation in

approximately 72,000 clock cycles, while the i960 MM and i960 MX processors required approximately 6,000 clock cycles. This difference illustrates the capability of the parallel floating-point execution units in the superscalar i960 MM and i960 MX processor implementations.

There are other benchmarks to indicate the relative floating-point throughput of a processor. The i960 MC and i960 XA processors are capable of running 5.2 MWhetstones/second (double precision) at 25 MHz, while the i960 MM and i960 MX devices execute the same benchmark at 19 MWhetstones/second. In the double-precision Linpack benchmark, the i960 MC and i960 XA execute at 1 MFLOPS (million floating point operations per second) at 25 MHz, while the superscalar versions of those architectures execute at over 10 MFLOPS at the same clock speed.

Table 2 summarizes the performance data of the military i960 processors.

Note that the improvement in performance shown above in the second generation parts is not achieved by increasing the clock frequency, but rather by improving the amount of work that the processor can do in a given clock cycle. This is a direct consequence of superscalar execution techniques.

There are numerous other indicators of throughput. In the integer-intensive Dhrystone benchmark, the first-generation military i960 processors execute at roughly 16K Dhrystones/second, while the superscalar versions can execute at over 46K Dhrystones/second. It is also common to relate general processor performance to the VAX 11-780 (which is given a rating of one VAX MIPS). The i960

Benchmark (25 MHz)	i960	
	MM/MX	MC/XA
VAX MIPS (very large programs)	25	10
VAX MIPS (small programs)	38	10
MWhetstones/sec (SP)	23	6.4
MWhetstones/sec (DP)	19	5.2
MFLOPS Linpack (DAXPY, DP)	10	1.0
MFLOPS Linpack (SAXPY, SP)	17	1.8
Dhrystones/sec (Rev 2.1)	46K	16K
10 x 10 DP Matrix Multiply (microseconds)	240	2880

Table 2: Military i960™ Processor Performance Summary

MC and i960 XA processors achieve a sustainable rate of about 10 VAX MIPS in a typical avionics application, while the i960 MM and i960 MX devices can sustain in excess of 25 VAX MIPS.

Few standard benchmarks can accurately relate the performance available in the superscalar i960 microprocessors. As an example, call and return efficiency is very high in the i960 MM and i960 MX microprocessors. A procedure call can be executed in only two clock cycles, including the switch to a new set of 16 local registers. The return from a procedure requires only three clocks. These times translate to very fast Ada context switches.

Another area where the i960 MM and i960 MX processors excel is in interrupt response. Fast interrupt response time is vital in complex embedded systems like advanced avionics. The superscalar military i960 processors respond to an external non-maskable interrupt and begin executing the first interrupt service routine instruction in only 820 nanoseconds at 25MHz.

The first generation of i960 military processors offer avionics designers unparalleled integration, ease of design, low power consumption and levels of performance. The superscalar implementations significantly improve on those throughput levels, delivering some of the highest performance products available to the industry.

The Cost of Performance Processors and a Solution

The i960 MM and i960 MX microprocessors stand alone in their ability to deliver the highest levels of performance and functionality to the avionics marketplace. Performance is not without its price however. To keep the data flowing into these instruction-hungry superscalar machines, large, fast caches are needed. Due to limitations on die size, it is necessary to supplement the on-chip caches with an external cache.

The external cache is tied to the i960 MM and i960 MX processors via an innovative, high-speed 64-bit backside bus. This bus serves to keep the on-chip caches full, and the instruction scheduler fed. This unique bus is critical to maintaining the high levels of performance attainable with the superscalar i960 MM and i960 MX microprocessors.

One of the penalties associated with the additional bus is increased pin count, which tends to reduce the high levels of integration and board space savings achieved with the first generation of military i960 processors. Therein lies the paradox: increased performance seems to necessitate decreased integration and added design complexity. Intel has a solution for the problem facing the designers of tomorrow's avionics systems: A multichip module (MCM) that integrates the processor, second-level cache and private memory, built-in test capability, and other logic into a single surface mount package.

The MCM implementation of the superscalar i960 military products provides a way to deliver what would otherwise consume most of a six-inch by six-inch SEM-E board (Standard Electronic Module, Size E) in a silicon-on-silicon module that is roughly two by three inches, yielding tremendous levels of integration and simplified design in about one-sixth the board area.

The MCM is planned for 1992 availability, to allow customers time to evaluate the architecture. In the interim, customers will be able to use a daughter-board that is functionally equivalent to the MCM. The daughter-board will consist of discrete components on a printed circuit card and will serve as an evaluation, system design and software development platform for early adopters of the i960MM and i960MX products. The daughterboard will be made available in the fall of 1991.

Support for the Military Architectures

Modern military systems are increasingly using advanced microprocessors like the i960 family members to perform an ever-expanding number of tasks. These systems must be able to take full advantage of the performance and integration features available in these complex processors. And the design of the hardware and software that comprise these systems must meet demanding schedule and quality requirements.

Furthermore, the DoD-standard Ada programming language is being mandated on virtually all government-funded military programs, and is even beginning to make its presence felt in the commercial marketplace. Indeed, future versions of commercial

jetliners will have avionics systems whose software modules are written in Ada.

The widespread need for validated Ada compilers to support modern microprocessors in both military and commercial marketplaces is now readily apparent.

Intel has long recognized the need for advanced Ada tools and has committed to ensuring their availability. The i960 architecture has been and will continue to be supported by multiple validated Ada compilers as well as a variety of other development tools that support tasks such as debugging, optimizing, simulating, disassembling and linking.

Intel is committed to ensuring that there is adequate quality Ada support for the military i960 architecture at its various levels and implementations. This development support, coupled with Intel's own applications support, development tools and microprocessor expertise will assure the existence of a quality development environment for the i960 architecture.

What's Next ?

As with all of Intel's strategic processors, the i960 architecture will continue to evolve and improve. Continuous improvements in quality and reliability, coupled with new process technologies, promise to lower costs and deliver higher-throughput processors for those applications demanding the highest performance levels.

The i960 architecture is already broadly proliferated and widely accepted. Its various levels have been realized in many different implementations, a trend that will continue as process technologies improve and transistor budgets increase.

In the military marketplace, Intel's commitment to the i960 architecture, its suitability to advanced avionics applications and strong development tools support, coupled with its status as an industry standard, has assured its acceptance and longevity well into the twenty-first century.

i960 is a trademark of Intel Corp.

*CHMOS is a patented process of Intel Corp.

Notes

102776830

Notes

Notes

UNITED STATES

Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

JAPAN

Intel Japan K.K.
5-6 Tokodai, Tsukuba-shi
Ibaraki, 300-26

FRANCE

Intel Corporation S.A.R.L.
1, Rue Edison, BP 303
78054 Saint-Quentin-en-Yvelines Cedex

UNITED KINGDOM

Intel Corporation (U.K.) Ltd.
Pipers Way
Swindon
Wiltshire, England SN3 1RJ

WEST GERMANY

Intel GmbH
Dornacher Strasse 1
8016 Feldkirchen bei Muenchen

HONG KONG

Intel Semiconductor Ltd.
10/F East Tower
Bond Center
Queensway, Central

CANADA

Intel Semiconductor of Canada, Ltd.
190 Attwell Drive, Suite 500
Rexdale, Ontario M9W 6H8