Introducing the MC68020: Performance and flexibility improvement through redesigned hardware and software enhancements*

by CLARA I. SERRANO


ABSTRACT

The MC68020 microprocessor is the full 32-bit implementation of the M68000-family architecture providing 4 gigabytes of linear address space. The MC68020 is upward user object-code compatible with the MC68000, MC68008, MC68010, and MC68012. As far as the user can determine, the architecture of the MC68020 is merely an extension of earlier processors in the family. However, the MC68020 has many new features invisible to the user program that enhance speed of operation, and many new software features that provide added flexibility for the system designer and programmer. Some of the hardware enhancements include additional supervisor registers, increased parallelism, an on-chip cache, a deeper instruction pipe, 3-bus-cycle reads and writes, a barrel shifter, three 32-bit ALUs, and an asynchronous dynamic bus-sizing interface. Software enhancements include a richer instruction set with added support for high-level languages, additional internal registers, a very general coprocessor interface, additional addressing modes, more sophisticated interrupt control, added exception-processing capabilities, and added privilege levels for memory management support; this is in addition to all the virtual memory/virtual machine support available with the MC68010 and MC68012, and the reliability provided by the exception-handling structure of all the other processors in the family.

Two new M68000 coprocessors, the MC68881 Floating Point Coprocessor and the MC68851 Paged-Memory Management Unit, can be directly interfaced with the MC68020. The MC68881 carries out all internal calculations to 80 bits of precision, and conforms to the IEEE Floating Point Specification. The MC68851 provides very flexible demand-paged memory management support and performs logical-to-physical address translations in 45 nanoseconds.
INTRODUCTION

The M68000 family of microprocessors and peripherals has provided extensive processing power and flexibility to system designers working on applications ranging from low-cost microcomputer consumer products to sophisticated multi-user engineering workstations. As technology has advanced and system integration progressed, a new high-performance microprocessor market has evolved—that of the 32-bit world. Users and designers in 32-bit environments typically wish to emulate the features of mini/mainframe computers, but at microprocessor sizes and prices. Motorola’s 32-bit implementation of the M68000 architecture, the MC68020, is an attractive solution for 32-bit applications that require raw processing power plus comprehensive hardware and software interfaces.

Two very important design concepts dominated the specification of the M68000 family architecture, and now the implementation of the MC68020; the first is performance or simply execution speed; the second is generality and extensibility, providing the system designer with a general purpose architecture with which to build a sophisticated system that solves the problem at hand with minimal effort and cost. These concepts are obviously related in that the most flexible of architectures allows the designer to implement objectives efficiently.

The new features of the MC68020 are discussed and highlighted from a systems standpoint.

M68000 FAMILY OVERVIEW

The M68000 is considered to be a 16/32-bit architecture, because all microprocessors in the family have 32-bit internal registers, can perform 32-bit operations, but do not all bring the 32-bit address and data paths directly to the outside world. This family of processors consists of the following:

1. The MC68000 with 24 address lines and 16 data lines
2. The MC68008, a reduced-cost part with 20 address lines and 8 data lines
3. The MC68010 Virtual Memory/Machine processor with the same addressing range of the MC68000, but with the added registers and hardware to allow full virtual-memory support
4. The MC68012, an extension of the MC68010 with 7 additional address lines, and now
5. The MC68020, the full 32-bit implementation of the family architecture

An important architectural feature of the entire family of processors is that the address space for each CPU is linear (not segmented).

Asynchronous Bus Structure

The M68000 family is a very popular choice for system designers interfacing a CPU with various speed memory and peripheral devices because of the performance and flexibility available with an asynchronous, non-multiplexed interface to these devices. With M68000 microprocessors, the access timing of the processor is dynamically controlled on each bus cycle by the device being accessed via a handshake signal called DTACK (Data Transfer Acknowledge). Thus with the fastest memory devices, a no wait-state system can be designed such that memory reads and writes only take four clock cycles. Another portion of the bus structure dynamically gives up the external bus to other requesting bus masters (i.e., DMA Controllers, other CPUs etc.) via the bus arbitration pins BR (Bus Request), BG (Bus Grant), and BGACK (Bus Grant Acknowledge), thereby allowing maximum external bus utilization.

Exception Processing

The asynchronous bus structure also handles hardware failures and improper memory accesses in a very straightforward manner. If data is not read or written correctly to memory (for example, if error detection and correction logic finds an error, or if a page fault occurs in a virtual-memory system), the external circuitry asserts BERR (Bus Error), and the microprocessor enters exception processing to handle the error gracefully. With the MC68010, MC68012, and MC68020, the simultaneous assertion of BERR and HALT causes the processor to rerun the current cycle, providing very quick recovery from a memory error or other predefined conditions.

Interrupts

External interrupts cause control to be passed directly to an exception-handler routine. The efficient handling of interrupts makes the M68000 family the logical choice for applications requiring real-time processing. Other exceptions can be initiated by software (via TRAP instructions, for example), and by hardware (tracing, etc.) yielding more than 200 (230 for the MC68020) unique exception routines that can automatically be called by the CPU when the corresponding conditions exist.

Instruction emulation

Another kind of exception occurs when the opcode of an instruction is not recognized by the processor. In this case, the
CPU automatically passes control to an illegal instruction handler routine, providing the capability to simulate user-defined instructions in software.

Register Set

The basic user programming model for all microprocessors in the M68000 family is shown in Figure 1. This model includes 16 general-purpose address and data registers, all 32 bits wide; a condition code register with carry, overflow, zero, negative, and extend bits; and a program counter providing direct program access to the full addressing range of each of the processors. The data registers may contain byte, word, or longword operands; the address registers may contain word or longword pointers with A7 as the default user stack pointer.

HARDWARE ENHANCEMENTS

MC68000 Supervisor Registers

Each of the M68000 processors has a second set of registers that can only be accessed in the supervisor state. This privilege distinction is particularly useful in an operating system environment where the user should not have direct access to operating system handling information. In the case of the MC68020, this set of registers includes two supervisor stack pointers, the master stack pointer and the interrupt stack pointer (see Figure 2). These two stack pointers facilitate multi-tasking control by allowing each task to have its own master stack containing control information relevant to that task. For example, when an interrupt occurs, thereby signaling the need for a task switch, the interrupt handler simply loads the master stack pointer, which points to the control block of the new task, without having to transfer all of the

interrupt-related information from the previous task's context, as this information is kept on the interrupt stack (see Figure 3).

The status register for the MC68020 contains the condition code bits (present in the user model), and the interrupt mask encoded in 12, 11, and 10. The M bit determines if the master or interrupt stack is being used by the supervisor, and the S bit determines if a program is executing in the user or supervisor state. The MC68020 also has two trace bits (T1 and T0) that allow a software debugger to trace on every instruction boundary, or on changes in program flow (e.g., a BRANCH instruction).

TheVector Base Register allows the exception vector table (containing the pointers to exception handler routines) to be arbitrarily relocatable, and therefore supports multiple exception vector tables. The alternate function code registers (SFC and DFC) allow the supervisor to access any address space by explicit manipulation of the function codes. Finally, the cache control registers (CACR and CAAR) allow the supervisor to manipulate the on-chip instruction cache in software.

Enhanced Bus Structure

Although the architecture of the MC68020 appears to be an extension of the MC68010 and MC68012, it is actually a com-
Instruction Cache

The MC68020 was designed to include a very high-speed cache memory on-chip for storing the instruction stream accesses in case they are accessed again. Mainframe computers have historically implemented small high-speed memories in order to store instructions and data that are accessed frequently due to the phenomenon known as locality of reference (i.e., the looping nature of most assembly language programs). The MC68020 is the first 32-bit microprocessor to incorporate the advantages of on-chip cache memory.

As shown in Figure 5, the cache is direct-mapped (i.e., each address maps into a unique location in the cache) with A7–A2 serving as an index into the tag store. The tag consists of the upper 24 address bits, Function Code 2 (to protect supervisor programs from user accesses), and a valid bit (ensuring that a valid instruction has been loaded into that location of the cache). Thus if an instruction prefetch is initiated, and the address in the tag (selected by the index) matches with the prefetch address, a hit is said to have occurred and the instruction is then fetched from the cache. When a cache miss occurs, and the cache is enabled, the CPU continues its external memory access, and loads the new instruction into the cache for possible future reference. Data is not stored in the cache, so that other bus masters may manipulate data areas without the possibility of the CPU then having “stale data” in its cache.

The supervisor controls the cache via the two cache control registers. It can enable, clear, or freeze the cache, or clear a particular cache entry. In the case of a clear entry operation, the cache address register (CAAR) is used to specify the entry to be invalidated. An external Cache Disable pin (CDIS) operates independently of the cache enable bit in the cache control register. CDIS can disable the cache to aid in software debugging.

The MC68020 cache enhances overall system performance as follows:

1. When an instruction is found in the cache, it is accessed in only two clocks (a 33% improvement over a no wait-state external prefetch).
2. When programs are executing out of the cache, the available external bus bandwidth is increased, thereby allowing other bus masters more access to the buses. The MC68020 was designed to include a very high-speed cache memory on-chip for storing the instruction stream accesses in case they are accessed again. Mainframe computers have historically implemented small high-speed memories in order to store instructions and data that are accessed frequently due to the phenomenon known as locality of reference (i.e., the looping nature of most assembly language programs). The MC68020 is the first 32-bit microprocessor to incorporate the advantages of on-chip cache memory.

As shown in Figure 5, the cache is direct-mapped (i.e., each address maps into a unique location in the cache) with A7–A2 serving as an index into the tag store. The tag consists of the upper 24 address bits, Function Code 2 (to protect supervisor programs from user accesses), and a valid bit (ensuring that a valid instruction has been loaded into that location of the cache). Thus if an instruction prefetch is initiated, and the address in the tag (selected by the index) matches with the prefetch address, a hit is said to have occurred and the instruction is then fetched from the cache. When a cache miss occurs, and the cache is enabled, the CPU continues its external memory access, and loads the new instruction into the cache for possible future reference. Data is not stored in the cache, so that other bus masters may manipulate data areas without the possibility of the CPU then having “stale data” in its cache.

The supervisor controls the cache via the two cache control registers. It can enable, clear, or freeze the cache, or clear a particular cache entry. In the case of a clear entry operation, the cache address register (CAAR) is used to specify the entry to be invalidated. An external Cache Disable pin (CDIS) operates independently of the cache enable bit in the cache control register. CDIS can disable the cache to aid in software debugging.

The MC68020 cache enhances overall system performance as follows:

1. When an instruction is found in the cache, it is accessed in only two clocks (a 33% improvement over a no wait-state external prefetch).
2. When programs are executing out of the cache, the available external bus bandwidth is increased, thereby allowing other bus masters more access to the buses.
3. Because instruction words are specified as 16-bit quantities, and instruction prefetches are always made as long-word reads, almost every other instruction word is sure to be found in the cache.
4. Because of the parallelism designed into the MC68020, the internal cache bus is isolated from the external bus so that instructions can be fetched from the cache at the same time that external data accesses are made.

Internal Arithmetic and Control Features

The MC68020 has three 32-bit Arithmetic and Logic Units (ALUs), which may be employed independently or in unison, again adding to the performance factor. A barrel shifter was also included to allow high-speed shift and rotate operations
SOFTWARE ENHANCEMENTS

The instruction set of the MC68020 is a superset of the instructions available with the earlier M68000 processors. The software enhancements in the MC68020 emphasize further flexibility and increased support for sophisticated operating systems and structured high-level languages.

Addressing Modes

The addressing modes have been expanded to allow 32-bit base addresses, indexes, base displacements and outer displacements to be specified in a single mode, and sizing and scaling of the index register by two, four, or eight. Sophisticated memory-indirect addressing modes allow for efficient generation of addresses to satisfy most complex programming structures. Intermediate memory accesses can be made in the calculation of an effective address, and the program counter may be used as a base displacement in any addressing mode. Post-increment and pre-decrement modes are present (as with other M68000 systems), thus allowing the user to create stacks and queues anywhere in memory.

Trap Exceptions

Supervisor exception processing can be initiated in software by the use of TRAP # or TRAPcc instructions. The programmer may pass control to a supervisor handler routine with 16 TRAP # exceptions, or with 16 trap-on-condition possibilities. The format of the TRAP instructions now allows two to four bytes of additional user-specified information to be passed to the trap handler as well.

Breakpoints

A BKPT instruction causes a Breakpoint Acknowledge cycle to be initiated. This bus cycle places a user-specified value of 0–7 on A4–A2. If the cycle is terminated by DSACKs, the value on the data bus is used to replace the breakpoint opcode in the instruction stream; if terminated by BERR, an exception is taken. The breakpoint instructions allow for sophisticated debug environments and real-time tracing of cache-resident routines.

Bit Field Support

A new data type called a bit field has been defined to be used with new bit field manipulation instructions. A bit field is a string of one to 32 bits that can reside in a data register or anywhere in memory with no restrictions on boundaries (i.e.,
COPROCESSOR INTERFACE

Extending the software capabilities of the MC68020 is achieved through its powerful coprocessor interface. A general-purpose protocol was established for CPU/coprocessor communication so that designers may customize systems with any type of coprocessor.

Coprocessor instructions are characterized by a leading “1111” in the opcode, and as mentioned earlier, an internal PLA detects a coprocessor instruction very early in its decode cycle. The CPU communicates with the coprocessor by placing access information on the address bus, including a coprocessor ID on A11–A9, selecting one of eight possible coprocessors. The coprocessor responds to the CPU by means of a coprocessor register indicating that it is busy, or that it is ready for more information. The possibilities of coprocessor responses are shown in Figure 7.

Coprocessors are classified as bus masters or bus slaves. A bus master type of coprocessor (for example, a paged memory management unit) can take control of the external buses; a slave-type has no bus control hardware.

Floating Point Support

The MC68881 Floating Point Coprocessor interfaces to the MC68020 as a coprocessor, or it can interface to other CPUs in the M68000 family as a peripheral. It fully implements the IEEE Floating Point Specification (Draft 10.0), and carries out all internal calculations to 80 bits of precision.

Figure 7—Coprocessor response primitives

From the collection of the Computer History Museum (www.computerhistory.org)
Memory Management

Due to a design philosophy that emphasizes flexibility, the MC68020 does not confine the system designer to a particular memory management scheme. However, the MC68851 Paged Memory Management Unit is available for those systems requiring support for paged virtual memory. The MC68851 has an on-chip content addressable memory that stores multiple page descriptors (64), once again supporting very efficient task switching and real-time interrupt processing in systems using memory management. It also implements up to eight levels of program protection, and contains table search hardware to automatically replace cache entries on a translation miss.

SUMMARY

The performance of the MC68020 exceeds that of any other 32-bit microprocessor on the market for several reasons. The hardware design incorporates state-of-the-art features such as an on-chip cache, multiple ALUs, and execution overlap. However, software enhancements also contribute to performance by allowing flexible interfacing with the rest of the system and significant operating system support. Furthermore, the general-purpose coprocessor interface on the MC68020 provides for limitless expansion by allowing the implementation of special purpose coprocessors that communicate with the MC68020.