ADVANCE INFORMATION

MEMORY MANAGEMENT UNIT MK68451

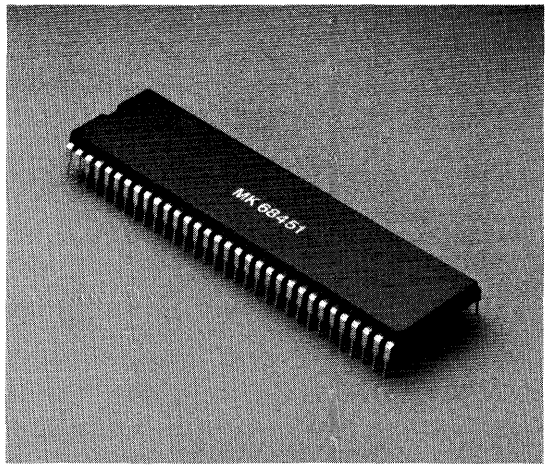
FEATURES

- ☐ Compatible with MK68000 and MK68008
- ☐ Provides virtual memory support for the MK68010
- ☐ Provides efficient memory allocation
- ☐ Seperates address spaces of system and user resources
- ☐ Provides write protection
- ☐ Supports paging and segmentation
- ☐ 32 segments of variable size with each MMU
- ☐ Multiple MMU capability to expand to any number of segments
- ☐ Allows inter-task communication through shared segments
- ☐ Quick context switching to cut operating system overhead
- ☐ Simplifies programming model of address space
- ☐ Increases system reliability
- ☐ DMA-compatible

GENERAL DESCRIPTION

The MK68451 memory management unit (MMU) provides address translation and protection for the 16 megabyte addressing range of the MK68000 MPU. Each bus master (or processor) in the MK68000 family provides a function code and an address during each bus cycle. The function code specifies an address space, and the address specifies a location within that address space. The function codes distinguish between user and supervisor spaces and, within these, between data and program spaces. This separation of address spaces provides the basis for memory management and protection by the operating system. Provision is also made for other bus masters to have separate address spaces for logical DMA.

MK68451
Figure 1



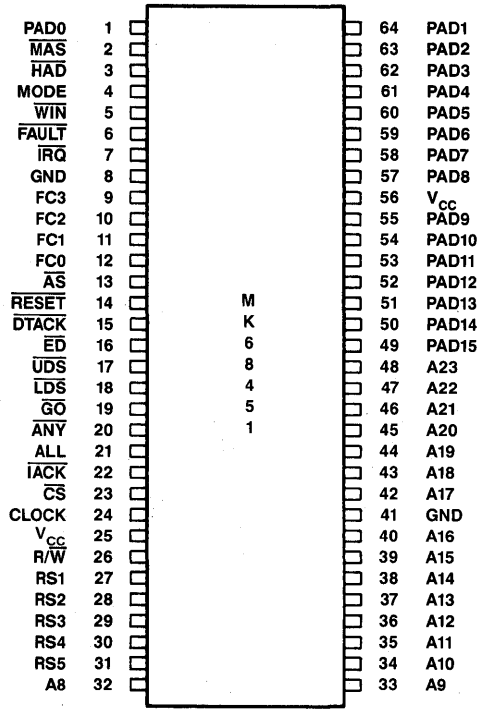
A multitasking operating system is simplified, and reliability is enhanced, through the use of the MMU.

The MK68451 memory management unit (MMU) is the basic element of a memory management mechanism (MMM) in an MK68000 family system. The operating system is responsible for insuring the proper execution of user tasks in the system environment, and memory management is basic to this responsibility. The MMM provides the operating system with the capability to allocate, control, and protect the system memory. A block diagram of a single-MMU system is shown in Figure 3.

An MMM, implemented with one or more MK68451 MMUs, can provide address translation, separation, and write protection for the system memory. The MMM can be programmed to cause an interrupt when a chosen section of memory is accessed, and can directly translate a logical address into a physical address, making it available to the MPU for use by the operating system. Using these features, the MMM can provide separation and security for user programs and allow the operating system to manage the memory in an efficient fashion for multitasking.

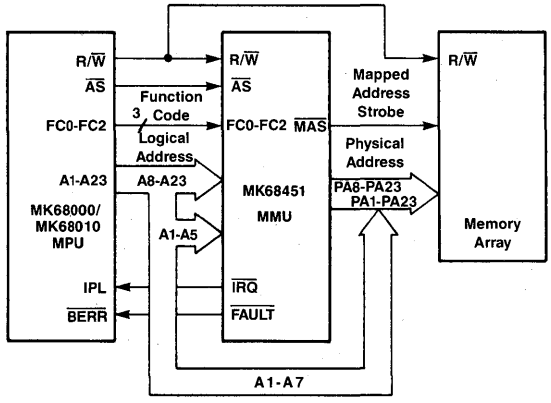
PIN ASSIGNMENT

Figure 2



SIMPLIFIED BLOCK DIAGRAM OF SINGLE-MMU SYSTEM

Figure 3



FUNCTIONAL DESCRIPTION

MEMORY SEGMENTS

The MMM partitions the logical address space into contiguous pieces called segments. Each segment is a section of the logical address space of a task which is mapped via the MMM into the physical address space. Each task may have any number of segments. Segments may be defined as user or supervisor, data-only or program-only, or program and data. They may be accessed by only one task or shared between two or more tasks. In addition, any segment can be write protected to insure system integrity. A fault (MK68000 bus error) is generated by the MMM if an undefined segment is accessed.

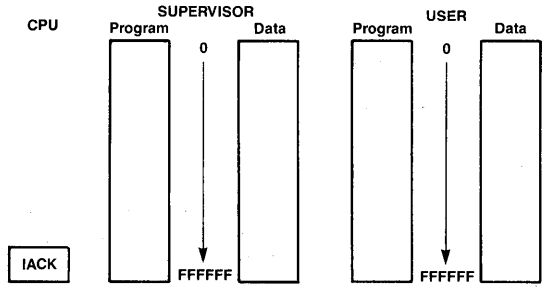
FUNCTION CODES AND ADDRESS SPACES

Each bus master in the MK68000 family provides a function code during each bus cycle to indicate the address space to be used for that cycle. The address bus then specifies a location within this address space for the operation taking place during that bus cycle.

The function codes appear on the FC0-FC2 lines of the MK68000 and divide the memory references into two logical address spaces—the supervisor and the user spaces. Each of these is further divided into program and data spaces. A separate address space is also provided for internal CPU-related activities, such as interrupt acknowledge, giving a total of five defined function codes. The address space of the MK68000 is shown in Figure 4.

ADDRESS SPACE OF MK68000

Figure 4



In addition to the 3-bit function code provided by the MK68000, the MK68451 MMU also allows a fourth bit (FC3) which provides for the possibility of another bus master in the system. In this case, FC3 would be a function of bus grant acknowledge ($\overline{BGACK}$) of the MK68000 to enable a second set of eight function codes. This raises the total number of possible function codes to 16. If there

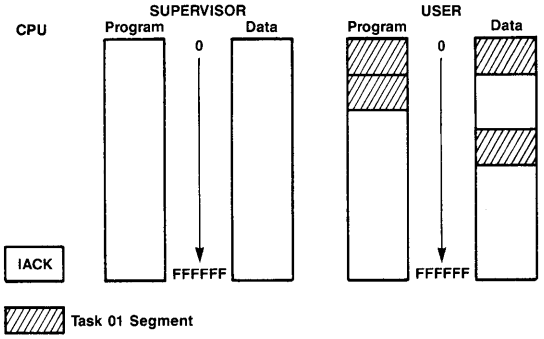
is only one bus master (the MPU), the FC3 pin on the MMU should be tied low, and only eight address spaces can then be used.

ADDRESS SPACE NUMBER

Each task in a system has an address space comprised of all the segments defined for that task. This address space is assigned a number by programming all the address space number (ASN) fields in its descriptors with the same value. This value can be considered a task number. The currently active task's number is kept in the appropriate entry(s) in the address space table (AST).

The AST is a set of MMU registers that defines which task's segments are to be used in address translation for each cycle type (supervisor program, supervisor data, etc.). The AST contains an 8-bit entry for each possible function code. Each entry is assigned an ASN (task number) and this is used to select which descriptors may be used for translation. The logical address is then translated by one of these to produce the physical address. Figure 5 is a typical memory map of a task's address space.

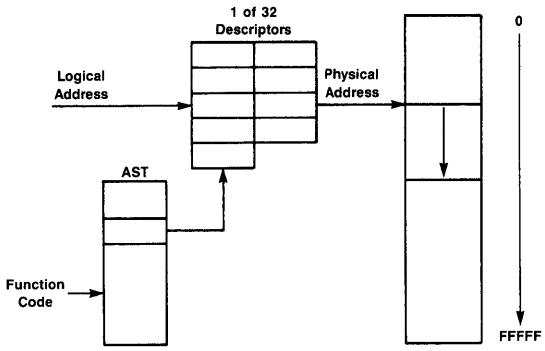
MEMORY MAP OF TYPICAL TASK ADDRESS SPACE
Figure 5



DESCRIPTORS

Address translation is done using descriptors. A descriptor is a set of six registers (nine bytes) which describe a memory segment and how that segment is to be mapped to the physical addresses. Each descriptor contains base addresses for the logical address masks. The size of the segment is then defined by "don't cares" in the masks. This method allows segment sizes from a minimum of 256 bytes to a maximum of 16 megabytes in binary increments (i.e., powers of two). This also forces both logical and physical addresses of segment boundaries to lie on a segment size boundary. That is, a segment can only start on an address which is a multiple of 2k. The segments can be defined in such a way to allow them to be logically or physically shared between tasks. Descriptor mapping is shown schematically in Figure 6.

SCHEMATIC DIAGRAM OF DESCRIPTOR MAPPING
Figure 6

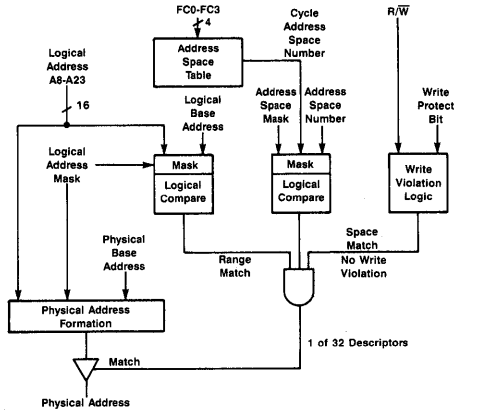


TRANSLATION

During normal translation, the MMU translates the logical address provided by the MK68000 to produce a physical address which is then presented to the memory array. This is accomplished by matching the logical address with the information in the descriptors and then mapping it into the physical address space. A block diagram of the MK68451 is shown in Figure. 7

Refer to Figure 3 for the following information. The logical address is composed of address lines A1-A23. The upper 16 bits of this address (A8-A23) are translated by the MMU and mapped into a physical address (PA8-PA23). The lower seven bits of the logical address (A1-A7) bypass the MMU and become the low-order physical address bits (PA1-PA7). In addition, the data strobes ($\overline{UDS}$ and $\overline{LDS}$) remain unmapped to become the physical data strobes for a total of eight unmapped address lines.

FUNCTIONAL BLOCK DIAGRAM
Figure 7



ORDERING INFORMATION

Part Number	Package Type	Max Clock Frequency	Temperature Range
MK68451N-8	Plastic	8.0 MHz	0° to 70°C
MK68451N-10	Plastic	10.0 MHz	0° to 70°C