Advanced Micro Devices

AmZ8000 Family Data Book

THE AM8000 IS BETTER

Advanced Micro Devices

AmZ8000 Family Data Book

Portions of the information in this book were originated by Zilog, Inc. and are used here with their permission.

901 Thompson Place, P.O. Box 453, Sunnyvale, California 94086
(408) 732-2400 TWX: 910-339-9280 TELEX: 34-6306

AM-PUB098

PREFACE

The present state of MOS LSI semiconductor technology has permitted powerful and complex general-purpose processors to be economically incorporated into single silicon chips. This capability ushers in a new era of system design, where for the first time low cost tools are available for solving many complex problems. Significant levels of computing power are now available inexpensively and can be used both to lower the cost of high performance systems and to improve the efficiency of programmers in their increasingly more complex tasks.

The AmZ8000 Family is the first integrated processor family to fully exploit this new era, breaking tradition with the legacy of compromised performance dictated by past manufacturing technologies. The two processors in the family incorporate many of the features heuristically evolved from both minicomputer and main-frame systems. This gives the applications programmer, the systems programmer and the system designer the power and flexibility required for today's complex systems.

This Data Book is one of a series of documents that support the AmZ8000 Family. The AmZ8000 Processor Instruction Set book (AM-PUB086) provides a complete, detailed description of all the processor instructions; the AmZ8000 Interface Manual (AM-PUB089) provides a detailed discussion of the CPU interface and its use with several support devices; AmZ8010 Memory Management introduction (AM-PUB093). Additional information is currently being prepared for publication.

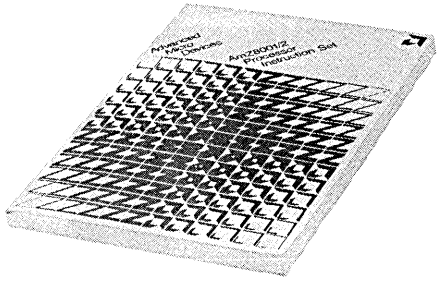
TABLE OF CONTENTS

1. Introduction to AmZ8000 Family 1-1
2. Processor Architectural Overview 2-1
3. System Architecture Overview 3-1
4. Component Data Sheets 4-1
5. Analog Interface Components 5-1
6. Memory Components 6-1
7. AmSYS™8 Development System, Software and Board Products 7-1

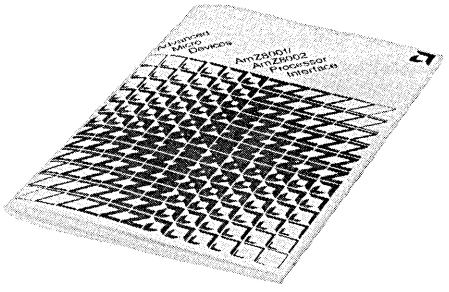
Appendices
- Advanced Commitment to Excellence (ACE) A-1
- Package Dimensions B-1
- MIL-STD-883/MIL-M-38510 ..... C-1

AmZ8000 Literature

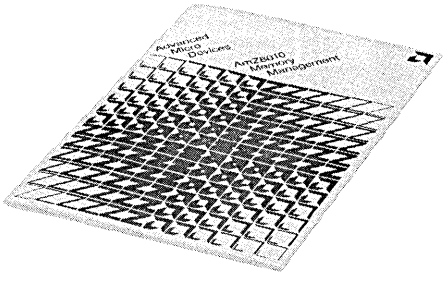
Processor Instruction Set
Defines the exact form and function of each CPU instruction. AM-PUB086



Processor Interface Manual
Describes hardware interconnections between CPU and peripherals. Describes interrupt daisy chain and multimicroprocessor systems. AM-PUB089



Memory Management
An introduction to memory management concepts and the AmZ8010 MMU device. AM-PUB093



COMPONENT PART NUMBER INDEX

AmZ8000 Processors

AmZ8001	Segmented 16-Bit Microprocessor	4-1
AmZ8002	Non-Segmented 16-Bit Microprocessor	4-22

AmZ8000 Family Peripherals

AmZ8010	Memory Management Unit	4-42
AmZ8016	Direct Memory Access Transfer Controller	4-44
AmZ8030	Serial Communication Controller	4-46
AmZ8036	Counter, Parallel I/O Interface	4-48
AmZ8038	FIFO Buffered Parallel I/O Interface	4-50
AmZ8052	CRT Controller	4-52
AmZ8060	FIFO Buffer Unit and FIO Expander	4-54
AmZ8065	Burst Error Processor	4-56
AmZ8068	Data Ciphering Processor	4-58
AmZ8073	System Timing Controller	4-60
AmZ8103/4/7/8	Octal Bidirectional Buffers	4-62, 4-68
AmZ8120	Octal D-Type Flip-Flop	4-69
AmZ8121	8-Bit Comparator	4-73
AmZ8127	System Clock Generator and Controller	4-77
AmZ8133	Octal 3-State Latches	4-78
AmZ8136	8-Bit Decoder with Control Storage	4-83
AmZ8140/44	Octal 3-State Buffers	4-87
AmZ8148	3-to-8 Decoder with Acknowledge	4-91
AmZ8160	Error Detection and Correction Unit	4-96
AmZ8161/62	Multiple Bus Buffers	4-104
AmZ8163	Timing, Refresh and EDC Controller	4-105
AmZ8164	Dynamic Memory Controller	4-106
AmZ8165/66	Octal Memory Driver	4-107
AmZ8173	Octal 3-State Latches	4-78, 4-113

LSI System Components

Am8255A	Programmable Peripheral Interface	4-114
Am9511A	Arithmetic Processor Unit	4-119
Am9512	Floating-Point Processor	4-144
Am9517A	Multimode DMA Controller	4-162
Am9519	Universal Interrupt Controller	4-176

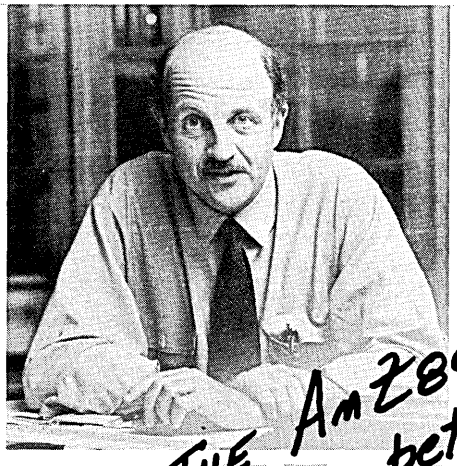
Analog Interface Components

Am6012	12-Bit Multiplying D/A Converter	5-1
Am6080	Microprocessor System Compatible 8-Bit High-Speed Multiplying D/A Converter	5-9
Am6081	Microprocessor System Compatible 8-Bit High-Speed Multiplying D/A Converter	5-11
Am6108	Microprocessor System Compatible A/D Converter	5-13

Memory Components

Am9016	16K x 1 Dynamic RAM	6-7
Am9044/9244	4K x 1 Static RAM	6-17
Am9114/9124	1K x 4 Static RAM	6-21
Am2716	2K x 8 Erasable PROM	6-25
Am9218	2K x 8 ROM	6-30
Am9232/33	4K x 8 ROM	6-33
Am27S28/29	512 x 8 PROM	6-37
Am27S30/31	512 x 8 PROM	6-42
Am27S180/181	1K x 8 PROM	6-47
Am27S184/185	2K x 8 PROM	6-52

AmZ8000 Microprocessor Family



CHAPTER 1

Introduction

INTRODUCTION

Advanced Micro Devices has undertaken a significant commitment to the world of 16-bit fixed-instruction-set processors. AMD is bringing to the market:

- A new, advanced processor architecture,
- A complete family of LSI peripheral circuits,
- A complete family of system support circuits,
- A complete family of memories and memory support circuits,
- Complete technical documentation,
- Effective development system products,
- Extensive support software.

This book describes all of these items in as much detail as is available at press time. Future editions will be propagated as new information is generated. Some of the data included here is preliminary and is intended to aid long-term planning. The factory should be contacted for the latest technical data on specific products and for the latest product availability information.

A large majority of future microprocessor applications will be serviced by a combination of single-chip microcomputer products such as the Am8048 series and by 16-bit microprocessors such as the AmZ8000. Where applications are simple enough, the 8-bit microcomputer chips will tend to be used. Increasing software costs and throughput requirements will cause the 16-bit CPUs to dominate the balance of the designs because they can answer these problems more efficiently. Conventional 8-bit microprocessors will serve a shrinking share of new designs.

In addition to significant increases in throughput that flow directly from the 16-bit structures, improved technology and more sophisticated architectures add even more performance. Software cost savings are being realized due to more powerful instruction sets, and in conjunction with sophisticated high-level languages such as PASCAL. Language compilers allow the programmers to write, debug and document programs in a shorter time span. This is vitally important for such a labor-intensive activity where costs are rapidly rising. The declining costs of technology-intensive LSI hardware can be used to improve software costs.

The AmZ8000 processors, in terms of resources, system features, instructions, interface and architecture, represents a major advance in microprocessor sophistication and system-level performance. The processors form the heart of a large family of components, systems, software, documentation and support. In addition to existing peripheral chips, a variety of new advanced peripherals has been designed to support the AmZ8001 and AmZ8002 processors. Figure 1 shows these new MOS/LSI components.

Several types of products are available for buffering, driving, latching, decoding and control functions. These are useful within the system as well as for external interface and for implementing memory subsystems. Additional specialized components have been designed for control of specific memory subsystem functions such as refreshing and error correction.

A wide variety of memory devices are available to support the AmZ8000 Family. Advanced Micro Devices manufactures many types of RAMs, ROMs, PROMs, EPROMs and FIFOs. Memories, of course, are essential elements in any processor design. Indeed, processors can be considered as tools for converting logic gates and algorithms into memory cells, thus providing user access to the excellent levels of technology available via memories.

An AmZ8000 Evaluation Board is available from Advanced Micro Computers for quick hands-on experience with the AmZ8000. It is a complete small computer with RAM, ROM and several I/O ports. Available software includes a resident monitor and a simple line-by-line assembler. The AmSYS™8/8 Microcomputer Development System supports the AmZ8000 Family as well as other microprocessors such as the Am8080, Am8085 and Z80. The system includes RAM, dual 8 inch Floppy Disk drives and several serial and parallel interfaces.

A powerful set of development software is available with the AmSYS /8 to make the complex process of product development easier and faster. The software includes a sophisticated Disk Operating System, Macro Assemblers, a linking loader, a powerful editor and debugger and a PASCAL compiler.

Advanced Micro Devices has an educational department which offers courses on the AmZ8000 microprocessor family, Am2900 bit-slice family and on related topics. Check with your AMD sales office for course outlines and schedules.

Advanced Micro Devices was conceived on the premise that there was a place in the semiconductor community for a manufacturer dedicated to excellence. This attitude is manifested in many ways throughout the structure of the company and has been maintained consistently throughout the life of AMD. In product assurance procedures, Advanced Micro Devices is unique. Only AMD processes all integrated circuits, commercial as well as military, to the demanding requirements of MIL-STD-883. The AmZ8000 microprocessors and its family of support devices are no exception; every component is 100% screened to MIL-STD-883, Method 5004, Class C.

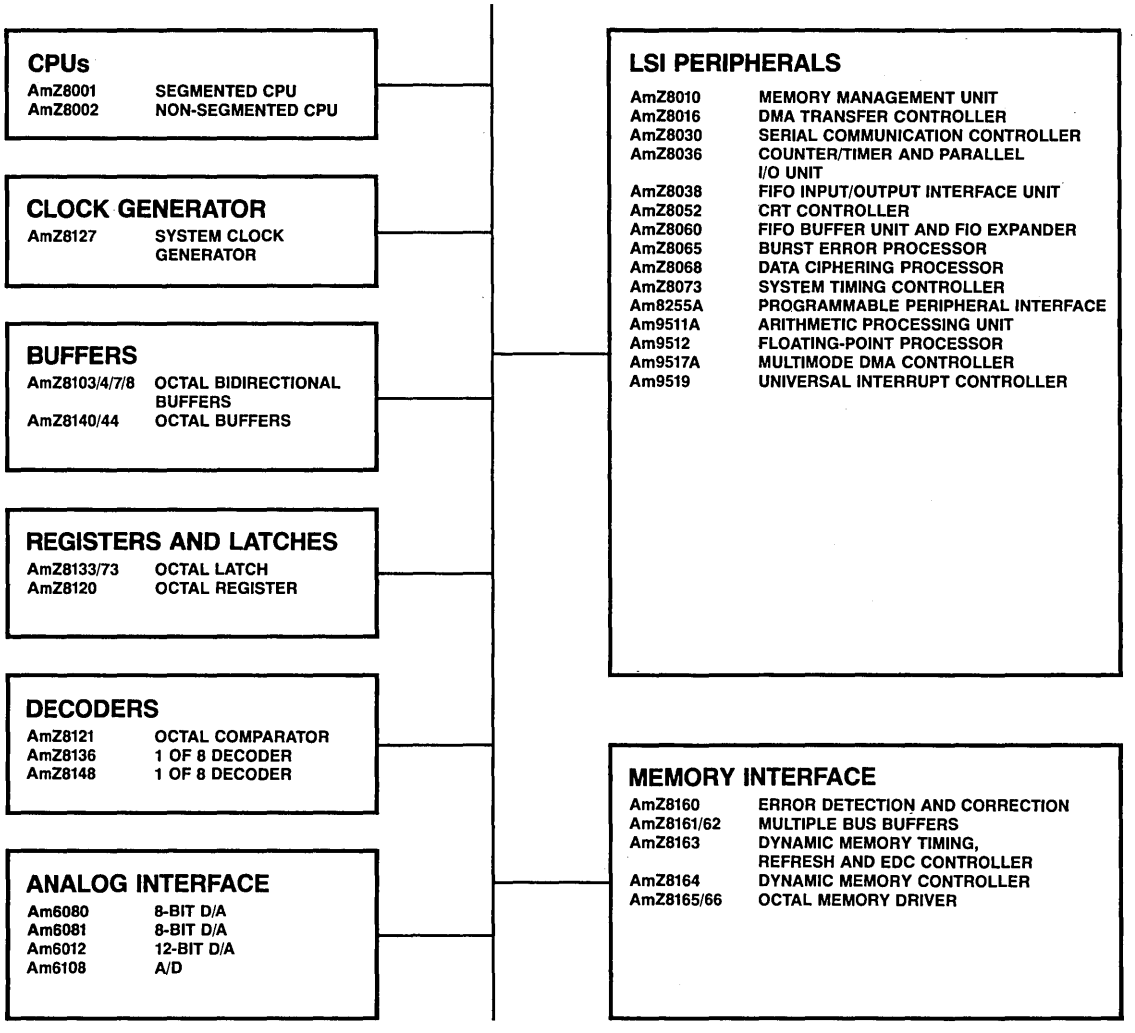


Figure 1. The AmZ8000 Family.

MOS-320

AmZ8000 Microprocessor Family

CHAPTER 2

Processor Architecture

PROCESSOR ARCHITECTURAL OVERVIEW

INTRODUCTION

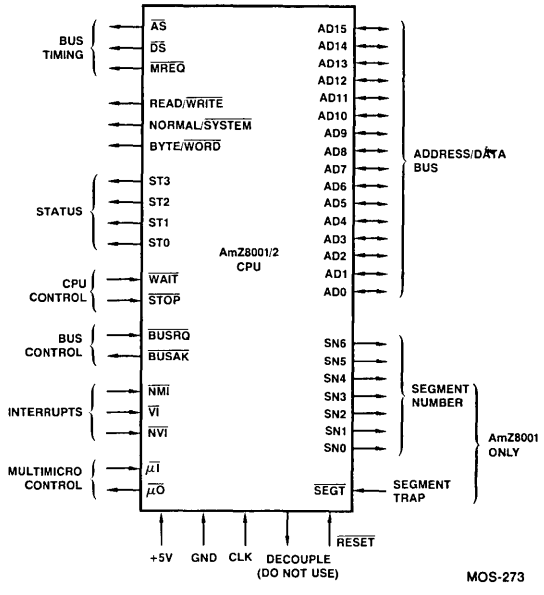
The AmZ8000 is an advanced high-end 16-bit microprocessor designed to span a wide variety of applications. Its features allow it to be used effectively in complex, high-throughput systems, yet it remains efficient for simpler systems as well. The AmZ8000 is available in two versions: the AmZ8001 48-pin segmented CPU and the AmZ8002 40-pin non-segmented CPU. The difference between the two devices is the addressing range: the AmZ8001 can directly address eight megabytes of memory per memory space and the AmZ8002 can directly address 65 kilobytes of memory per memory space. To meet the requirements of complex, memory intensive applications, the AmZ8010 Memory Management Unit offers logical-to-physical address translation and several memory protection features.

The AmZ8000 has abundant CPU resources that include numerous registers, many data element types, a large instruction set and several addressing modes. Not only are the CPU resources abundant, but they exhibit a consistency and regularity not found in previous microprocessor architectures. Regularity of register organization, data types, instructions and addressing modes greatly simplifies the programming process and reduces program length.

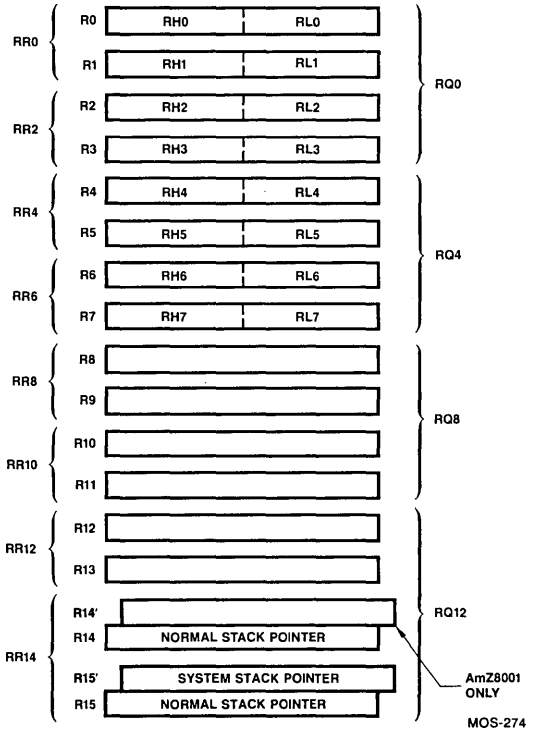
Compiler, compiler-produced, and operating system code all run efficiently on the AmZ8000. The AmZ8000 supports compilers with features such as a consistent instruction set, large address space, relocation, multiple stacks and some specific instructions (Push, Pop, Increment, Test).

Operating systems are supported by features such as system and normal modes, system and normal stacks, specific instructions (System Call, Load Program Status and privileged instructions), and by a sophisticated interrupt and trap structure. This structure includes three types of interrupts (non-maskable, non-vectored and vectored) and four types of traps (system calls, illegal instructions, privileged instructions and segment errors).

Multi-microprocessor systems are supported in software by exclusion and synchronization instructions and in hardware by the Micro In and Micro Out interface lines.



CPU Pin Configuration



General-Purpose Registers

CPU RESOURCES

Not only must the address space of an advanced architecture by large, but its CPU resources must be abundant enough for the solution of large problems.

Registers

The AmZ8000 offers sixteen 16-bit general-purpose registers in addition to special system registers. All 16 general-purpose registers may be used as accumulators and all but R0 as index registers and stack pointers. The first eight registers (R0-R7) can be used as sixteen 8-bit byte registers. For operations requiring long words (32 bits), the general purpose registers are grouped in pairs (RR0-RR14). For certain 64-bit operands (i.e., multiplication and division with long words), the register set is grouped in quadruples (RQ0-RQ12) to form 64-bit registers.

The CPU instruction set supports seven main data types: bits, BCD digits, bytes, words (16 bits), long words (32 bits) byte strings and word strings. Additionally, many other data elements such as memory addresses, I/O addresses, segment table entries and program status words may also be manipulated.

Stacks

The AmZ8000 allows data stacks to be located anywhere in memory. Push and Pop instructions allow any register (except RR0 (AmZ8001) and R0 (AmZ8002)) to be designated as data stack pointers. Call and Return instructions, as well as interrupts and traps use the (implied) linkage stack pointers. For the AmZ8001, the register pair RR14 (R14 and R15) is the linkage stack pointer, while for the AmZ8002 register R15 is the linkage stack pointer.

The CPUs operate in one of two selectable modes: System and Normal. The System mode is sometimes called a Supervisor or Privileged mode and the Normal mode is sometimes known as User or Task or Nonprivileged mode. Separation from system information is provided by a dual set of linkage stack pointers. In the AmZ8001, the register pair R14′ and R15′ will be used in the System mode as the implied stack pointer, while the AmZ8002 will use the R15′ register. Because the implied stack pointers are part of the general-purpose register group, the user can manipulate the stack pointers with any of the instructions available for register operation.

Program Status Information

This group of status registers contains the program counter (PC), the flag and control word (FCW) and the new program status area pointer (NPSAP). When an interrupt or trap occurs, the Program Status registers (PC and FCW) are saved on the system stack. An Identifier that describes the reason for interruption is also saved. The NPSAP provides the memory location for loading new information into the PC and FCW registers.

The CPU Control bits occupy the upper byte of the Flag and Control Word. The bits may be read and loaded by the privileged LDCTL instruction. The Control bits are:

- SEG Segmented Mode Enable
- S/N System or Normal Mode
- VIE Vectored Interrupt Enable
- NVIE Non-Vectored Interrupt Enable

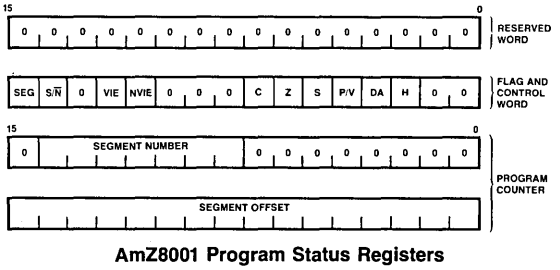
The SEG bit indicates segmented operation in the AmZ8001 if set to 1. When the SEG bit is 0, the AmZ8001 is forced into non-segmented operation and will interpret all programs as non-segmented. In this mode, the AmZ8001 executes AmZ8002 non-segmented code. The AmZ8002 SEG bit is always set to 0 and cannot be altered by the programmer.

The CPU Flags occupy the lower byte of the Flag and Control Word. The privileged instructions LDCTLB, RESFLG and SETFLG are used to load, read, set and clear the flags. The Flag bits are:

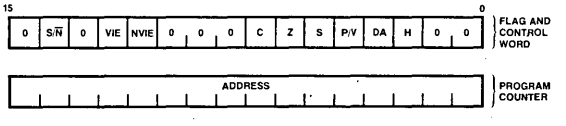
- C Carry Result
- Z Zero Result
- S Sign Result
- P/V Even Parity or Overflow
- DA Decimal Adjust
- H Half Carry

Interrupt and Trap Structure

The AmZ8000 provides a flexible and powerful interrupt and trap structure. Interrupts are asynchronous events triggered by an external device requesting service, while traps are synchronous events occurring upon the execution of certain instructions. The AmZ8000 supports three types of interrupts (non-maskable, vectored and non-vectored) and four traps (system call, unimplemented instruction, privileged instructions and segmentation trap). The vectored and non-vectored interrupts are maskable. When an interrupt or trap occurs, a 16-bit identifier (in addition to the PC and FCW registers) is pushed onto the system stack.

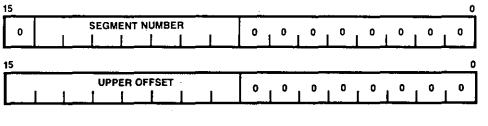


AmZ8001 Program Status Registers

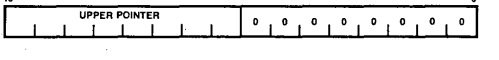


AmZ8002 Program Status Registers

MOS-275



AmZ8001 New Program Status Area Pointer

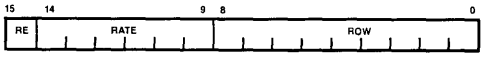


AmZ8002 New Program Status Area Pointer

The identifier contains the reason for the trap or interrupt. For internal traps, the identifier is the first word of the trapped instruction. For external traps or interrupts, the identifier is placed on the data bus by the interrupting or trapping peripheral.

Memory Refresh

The AmZ8000 CPUs contain a refresh counter for automatically refreshing dynamic memory. A 9-bit row counter can address up to 256 rows and thereby assures compatibility with the latest 64K dynamic memories. It is incremented by two each time the rate counter reaches end-of-count. The rate counter determines the time between successive refreshes and can be programmed from 1 to 64μsec assuming a 4MHz CPU clock. The refresh mechanism can be disabled under software control.



Refresh Counter MOS-276

Large Addressing Space

High-level languages, sophisticated operating systems, large data bases, large programs and decreasing memory prices are all accelerating the trend toward larger memories. The AmZ8000 processors can directly address up to eight megabytes of memory per address space. Four convenient, separate address spaces exist in both versions of the AmZ8000 processors: code and data for both the system mode and the normal mode.

Each space is addressed by a 16-bit or 23-bit address. Thus the total system addressing for a user is 32M bytes for the AmZ8001 and 256K bytes for the AmZ8002. Instructions are always addressed on word boundaries (even-numbered addresses) while data is addressed by byte, word, long word, or quadruple word addresses. A specific bit can also be addressed within a byte or word address.

The AmZ8000 also has a 16-bit I/O addressing space which is separate from the memory address space. An attractive complement of single element I/O and block I/O instructions exist for bytes and words.

Additionally, many useful memory management features, are provided by the AmZ8010 Memory Management Unit when used as a companion to the AmZ8001. These features will extend the life of the architecture by avoiding memory address limitations that have hampered microprocessors in the past.

The only drawback of the long addresses required by the large addressing space is the larger size of the instructions and the need for register pairs for some addressing modes. This problem is minimized by segmented addressing features, the use of short addresses in many situations, relative addressing ability and by the availability of a large number of general-purpose registers.

Memory Management

AmZ8001 programs can directly access the entire address space. A full address mode is available where 23 bits are set aside within the instruction for the address. The AmZ8001 also offers a mode called short offset mode in which the same address can be expressed by 16 bits in many situations where the higher-order offset bits are zeros.

Alternative methods commonly employ fixed internal registers that contain address extensions. Although these methods may use shorter instruction addresses, the byte savings are lost because many instructions are required to explicitly manage the contents of the registers. The AmZ8001 can use variations of these methods; however, it also provides direct addressing that removes the necessity for those extra instructions and unburdens the programmer from managing the register contents.

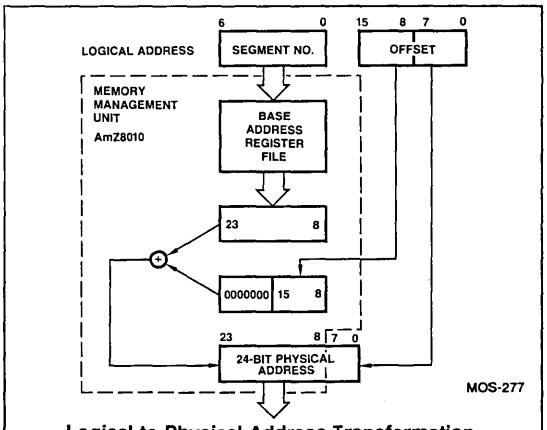
Another important feature provided to the system designer is the ability to distinguish externally between System code and Normal code, and in both cases, to distinguish between instruction space and data space. If this feature is utilized, the AmZ8001 can address up to 32 megabytes of memory. Aided by system programs, the memory management unit can help manage the large address space on behalf of the user.

Segmentation is the mechanism provided to address the large amount of memory addressable by the AmZ8001. A segmented address is made of two parts: a segment number and an offset value. The AmZ8001 can designate up to 128 segments that reference areas of memory variable in size from 256 bytes to 64 kilobytes, in increments of 256 bytes.

The only difference between running segmented or non-segmented code is the number of bytes per address and the number of registers used for full addresses. Code written for the non-segmented AmZ8002 can run in one segment of the segmented AmZ8001. Thus, full compatibility exists between the two versions.

The AmZ8010 Memory Management Unit essentially doubles the silicon area available for the processor function (and adds pins as well). Hence, it also doubles the hardware available to the designer for implementing more high-end features than otherwise would have been possible. Some of these features include variable sized segments, more sophisticated dynamic memory relocation and several types of memory protection attributes.

Addresses manipulated by the programmer are called logical addresses. The MMU translates these logical addresses to physical addresses required for accessing the memory. This address transformation makes user software addresses independent of the actual physical memory thus freeing the user from specifying where information is actually located in memory. The translation table in the Memory Management Unit associates the 7-bit segment number with the base address of the physical memory segment. The 16-bit offset portion of the logical address from the CPU is added to the physical base address to obtain the actual physical address. The system may dynamically reload translation tables as tasks are created, suspended, or changed. Memory protection features prevent illegal uses of segment, such as writing into a write-protected zone. Several Memory Management Units may be used with a single CPU.



MOS-277

Logical-to-Physical Address Transformation

Instructions

Compared to other microprocessors or to 16-bit minicomputers, the number and power of individual instructions has been greatly increased. Over 110 distinct instruction types are available with the AmZ8000. Byte, word and long-word data elements can be processed by all the core instructions. Each instruction, with few exceptions, can use most of the addressing modes. Over 410 meaningful combinations of instruction types, data elements and addressing modes are available.

The instruction set provides nine basic instruction groups: Load and Exchange, Arithmetic, Logical, Program Control, Bit Manipulation, Shift and Rotate, String Manipulation, CPU Control, and Input/Output. Instructions vary in length from one to five words depending on the operation and addressing mode. String and block move instructions are interruptable to enhance response time. Also provided are signed-multiply and signed-divide instructions implemented in hardware for both 16-bit and 32-bit values.

The instruction set provides several user-selectable addressing modes including five main modes: Register (R), Indirect Register (IR), Direct Address (DA), Indexed (X) and Immediate (IM). For certain instructions, there are other modes: Base Address (BA), Base indexed (BX), Relative Address (RA), Autoincrement and Autodecrement.

Code Density

For a given hardware technology, microprocessor speed is largely dependent on the number of executed instruction words needed for a particular function. Therefore, code density becomes an important issue in high-performance systems. The AmZ8000 offers several advantages in this respect.

The number of words required to specify frequently executed instructions has been minimized. A special group of high frequency instructions has been designed into single words. This not only improves speed, but increases code density as well.

A short offset mechanism is also designed to allow certain 23-bit addresses to be reduced to a single word. It can be automatically invoked by assemblers and compilers.

Additional large improvements in program size and speed result from the consistent and regular architecture, and the greater power of the instruction set. These factors allow fewer instructions to accomplish a given task.

Compiler Efficiency

It is tempting to adapt a computer architecture that efficiently executes a particular high-level language. Any special-purpose match between an architecture and a language is efficient for that language, but most likely inefficient for unrelated languages. Since the AmZ8000 is a general-purpose microprocessor, general language support has been provided through the inclusion of features that ease typical compilation and code-generation problems for all high-level languages.

Among these features is the regularity of the AmZ8000 addressing modes, registers and data element types. In addition, any register can be used as a data stack pointer with the Push and Pop instructions. Segmentation and relocation are useful features for high-level language procedure implementation. Procedure parameter passing is aided by these features as well as by special increment and decrement instructions which are useful in stack frame allocation and de-allocation. Base Address and Base Indexed addressing modes are also useful for stack frames. Useful testing and comparison of data, logical evaluation and initialization are made very efficient by several special instruction types. Compilers and assemblers handle character blocks quite frequently and the string manipulation instructions provide unusual efficiency compared to software simulations of these important tasks.

Operating System Support

Interrupt and task-switching features are included to improve operating system implementations. The memory-management and compiler-support features also contribute effectiveness in this environment.

The interrupt structure has three levels: non-maskable, non-vectored and vectored. When an interrupt occurs, the program status is saved on the stack with an indication of the reason for this state switching. A new program status is then loaded from memory and execution proceeds using the new Program Counter. In the case of a vectored interrupt, each vector points to a unique new Program Counter, providing direct access to particular service routines.

The System/Normal partition improves operating system integrity and organization. In the System mode, all operations are allowed; in the Normal mode, certain system instructions are prohibited. The System Call instruction allows a controlled switch of mode, and the implementation of traps enforces these restrictions. Dual linkage stack pointers also support the System/Normal partition.

Traps result in the same type of program status saving as interrupts; in both cases, the information saved is pushed on the system linkage stack and keeps the normal stack undisturbed. The Load Multiple instruction allows the contents of any group of the general registers to be saved and restored efficiently in memory. Running system programs can cause selective or general program status changes under direct software control. Finally, exclusion and serialization can be achieved with the "atomic" Test and Set instruction that synchronizes cooperating processes.

Multiprocessor Support

The AmZ8000 exclusion/serialization mechanism is designed for multimicroprocessor systems. Any CPU in a multiprocessor system can exclude all other asynchronous CPUs from any critical shared resource by using the Micro In (μI) input and Micro Out (μO) output in conjunction with several coordinating instructions.

In addition, the large address space of the AmZ8000 proves to be a beneficial feature in most multiprocessor systems.

AmZ8000
Microprocessor
Family

A wide variety of peripheral circuits are easy to use with the AmZ8000.

CHAPTER 3

System Architecture

SYSTEM ARCHITECTURE OVERVIEW

INTRODUCTION

The AmZ8000 CPU and peripheral device family support a variety of interface exchanges. The AmZ8000 processors are designed to handle five basic types of transfers: memory operations, I/O operations, resource request daisy chain, bus request daisy chain and interrupt request daisy chain. These interface exchanges share most of a common set of control signals. Most of the AmZ8000 peripherals use a similar control structure and some subset of the five basic transfer types. Some of the features of the system structure are:

- Juxtaposition and coordination of five different buses.
- Transparent bus for asynchronous peripheral operations.
- Multiplexed I/O and memory operations.
- Multiplexed Address and Data transfers.
- Full parallel 23-bit addresses.
- Preemptive interrupt daisy chains.
- Interrupt protocol allows vectors for peripheral identification.
- Resource allocation uses exclude/grant daisy chain.
- Bus requests use daisy chain for contention resolution.

Memory Operations

The AmZ8000 CPU offers a powerful combination of memory access techniques to a word organized memory. The CPU distinguishes between instructions, data and stack entries in both System and Normal modes and outputs a combination of status and bus timing signals. The status lines consist of four outputs which indicate the type of bus access; a Normal/System line to aid in selecting system or user memory space; a Byte/Word output line to allow addressing of an individual byte or a 16-bit word; and the Read/Write output indicating the direction of data flow.

Both versions of the AmZ8000 exhibit the identical set of status lines thereby allowing convenient access to separate 64K byte address spaces: system code, normal code, system data, normal data. The AmZ8001, through its additional seven segment number outputs allows extension of its address spaces to 8M bytes each. The AmZ8010 Memory Management Unit can further enhance memory space usage and allocation in the AmZ8001. The MMU offers the capability of generating a 24-bit physical address from the 23-bit logical addresses emitted by the CPU. Thus the AmZ8001 can directly address any 8M bytes within a 16M byte physical memory. Furthermore, memory address segments can be of any size that is a multiple of 256 bytes and may overlap. Each Memory Management Unit may translate up to 64 address segments. If more segments are required, several Memory Management Units may simply be paralleled.

Status Line Codes

ST3-ST0	Definition
0000	Internal operation
0001	Memory refresh
0010	I/O reference
0011	Special I/O reference
0100	Segment trap acknowledge
0101	Non-maskable interrupt acknowledge
0110	Non-vectored interrupt acknowledge
0111	Vectored interrupt acknowledge
1000	Data memory request
1001	Stack memory request
1100	Program reference, nth word
1101	Instruction fetch, first word

The AmZ8000 bus timing signals consisting of three signals: MREQ whose level indicates a memory or I/O type transfer; AS which is an address strobe (an address is indicated valid by its rising edge); and DS which is the data strobe (data read or written is also indicated valid by the rising edge of DS). A typical memory fetch cycle occurs within three CPU clock cycles thereby allowing the usage of moderate access time memories for price/performance enhancement. A WAIT bus control input is also available on the CPU thus allowing even slower memory operation.

I/O Operations

I/O operations between the AmZ8000 CPU and its peripherals are transacted on the same multiplexed address/data bus lines and are distinguished only by status outputs. (MREQ HIGH specifies an I/O operation while status lines ST0-ST3 specify either a standard I/O address space or a special I/O address space: for example, an AmZ8010 MMU.) The I/O address space can, therefore, be two separate 16-bit address spaces, in addition to the memory space, if the status lines are decoded in conjunction with MREQ signal. The large I/O address space allows great

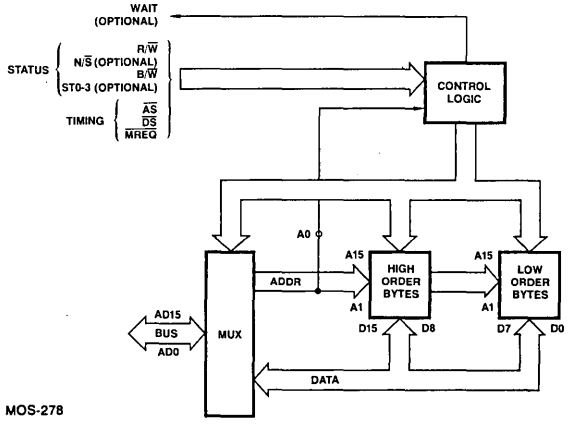


Figure 1. Typical Memory Organization.

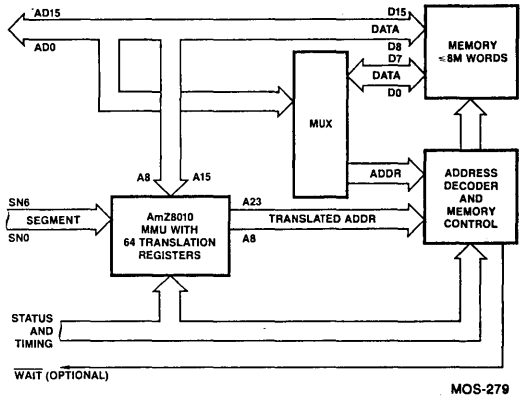


Figure 2. AmZ8010 Managed Segmented Memory.

versatility for accessing intensive register oriented peripherals: AmZ8000 peripherals typically follow the philosophy of allowing read and write operations to all their internal registers.

I/O mapped 8-bit peripherals can be located on either the upper or lower half of the address/data bus for data transfers; the AmZ8000 CPU duplicates a data byte on both halves of the bus when executing a byte write I/O operation.

A byte read I/O operation follows the same principle as byte read memory operations: Odd addresses refer to the low-order byte (AD0-AD7 bus) and even addresses to the high-order byte (AD8-AD15 bus).

Eight and 16-bit wide peripherals can be intermixed on the same bus, as the AmZ8000 provides both byte and word I/O oriented instructions, as well as Block I/O instructions for rapid data transfers (i.e., I/O peripheral register initialization).

AmZ8000 peripherals are not required to be synchronous with the CPU, because no clock is transmitted. AS and DS strobe signals provide the timing. In addition, the CPU inserts an automatic wait state whenever an I/O operation is performed. Additional wait states can be inserted by a slow peripheral by activating the AmZ8000 CPU wait status input.

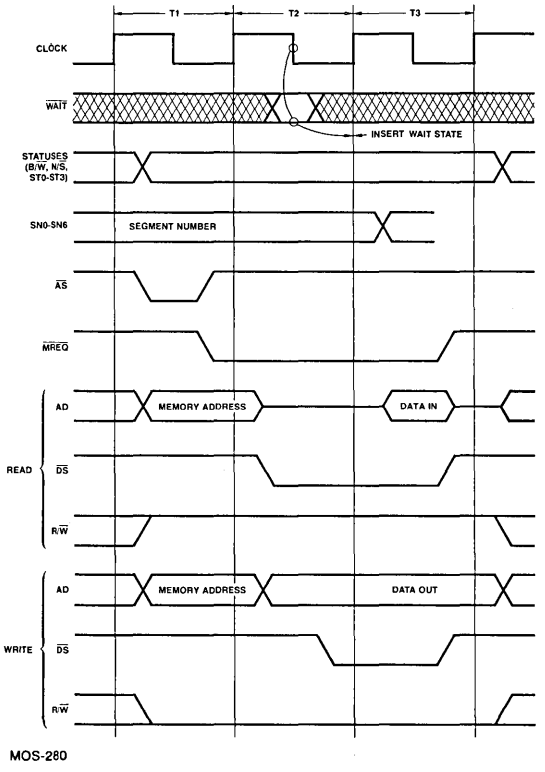


Figure 3. Memory Read and Write Timing.

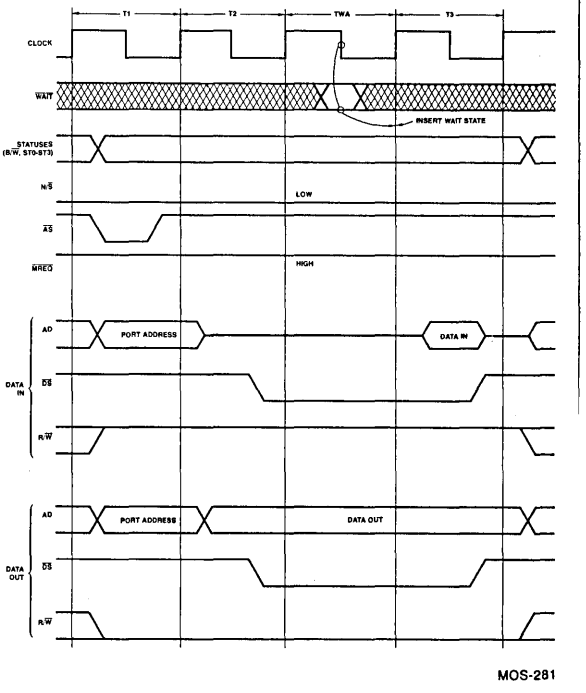


Figure 4. Input/Output Timing.

Interrupt Daisy Chain Protocol

An interrupt daisy chain protocol enables peripherals to act as slaves to the CPU. In contrast to I/O transactions which occur on the data bus in an orderly fashion; each peripheral, with interrupt capability, may attempt to use it simultaneously with another interrupting device. A daisy chain link between peripherals implements a distributed arbitration policy between interrupt requests to achieve an orderly, prioritized sharing of the CPU's resources.

The majority of the AmZ8000 family peripherals have the ability to request interrupts due to several internal reasons (for example, the AmZ8036 CIO can have three reasons to interrupt and the AmZ8030 SCC, eight). The peripheral itself contains an internal interrupt structure for prioritizing its internal interrupt request sources; for requesting service according to its priority status among other peripherals sharing the system bus; for outputting a proper vector identification, with or without additional status bits, when acknowledged by the CPU. The 8-bit vector allows efficient program transfers to the appropriate service routine without additional polling of the peripheral by the CPU. The peripheral self-vectoring and daisy chain prioritizing precludes the requirement for an additional interrupt controller device in the system.

The AmZ8000 CPU recognizes three interrupt inputs (non-maskable, vectored, and non-vectored), and a segmentation trap input (normally intended to be generated by an AmZ8010 MMU).

Interrupt inputs are asynchronous, and the interrupt acknowledge is decodable from the ST0-ST3 status lines for each interrupt type. An explicit completion code must be emitted by the CPU to terminate some peripheral's interrupt service. An appropriate I/O instruction easily accomplishes it.

To ensure data integrity and proper settling of IEO signals in the daisy chain link, the AmZ8000 CPU automatically inserts five wait states following interrupt acknowledge and before asynchronously strobing in the peripheral's identification vector ("Identifier"). The peripheral may activate the CPU's WAIT line and cause additional wait cycles, if required.

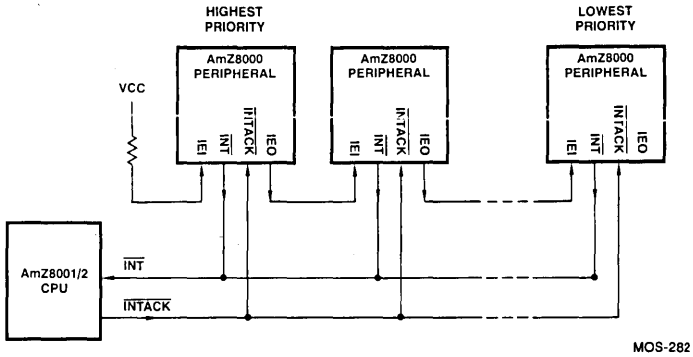


Figure 5. Interrupt Daisy Chain Connections.

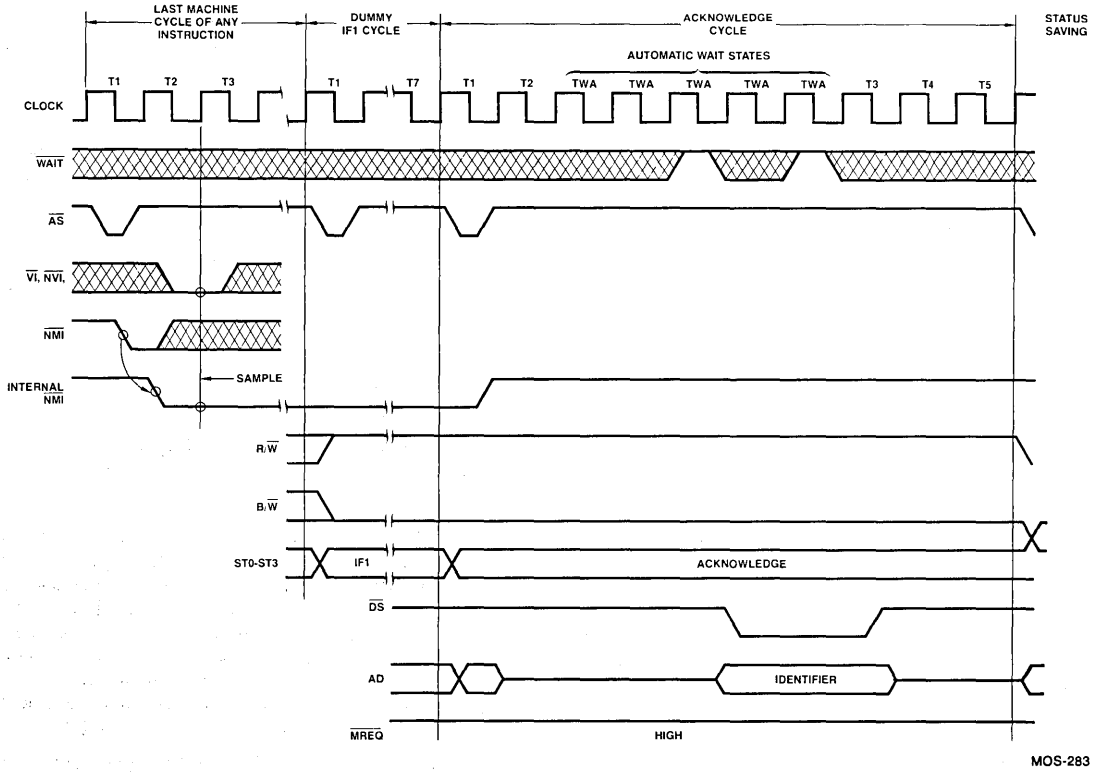


Figure 6. CPU Interrupt and Segment Trap Request Acknowledge Timing.

Peripheral Internal Interrupt Protocol

In general, AmZ8000 family peripherals have three bits for control and status of the internal interrupt logic: an Interrupt Pending bit, an Interrupt Under Service bit and an Interrupt Enable bit. For a multisourced internal interrupt structure, the three above bits are duplicated for each interrupt source, and the interrupt sources are internally prioritized. Lower priority peripherals in the daisy chain link are disabled by the status of another internal list: Disable Lower Chain bit. The peripheral will output a vector for interrupt source identification, when acknowledged by the CPU, according to the status of its No Vector bit. The 8-bit vector (usually defined by an internally programmed register) may include status information as specified by the Vector Include Status bit.

Resource Request Chain Protocol

Multimicroprocessor systems are well supported in hardware and software. Resources, like buses with their associated peripherals or common memories, are easily shared in an orderly and expedient fashion by several CPUs in a system. In general, the resource user need not be a CPU (i.e., it could be an I/O Channel Processor, an Arithmetic Processor, etc.) as long as it can implement a meaningful resource sharing protocol. This is in contrast to a bus request chain (i.e., DMA operations) in which system exists a default master and a defined bus with a tailored protocol. The AmZ8000 CPU supports resource protocol with two hardware pins: the MultiMicro Out (μO) to issue requests for the resource and the MultiMicro In (μI) to recognize or test the state of the resource. Four special instructions allow the CPU to test and request a resource, and to exclude or allow other users to the resource. These functions are particularly interesting with the eight megabyte AmZ8001 CPU address space capability for multiple microprocessor systems with large shared memory requirements.

A resource request protocol is easily implemented in hardware through simple SSI logic where each resource user is connected to the resource-sharing system by four unidirectional lines.

Unlike an interrupt or bus request chain no user is a default master and no user can therefore be preempted. The resource request protocol uses the request algorithm described below:

1. A user process checks the status of the μST (resource status) line to see if the resource is busy. (The AmZ8000 CPU would look at its μI input pin via the privileged MBIT instruction.)
2. If the μST line is active, then another user is either using the resource or is in the process of requesting it. In both cases, the resource request is terminated with the indication of a busy resource via internal flag. This implements the policy of no preemption.
3. If the μST line is not active, then the user:
   a. Activates its μRQ line (μO output on the AmZ8000) and,
   b. Waits for a finite delay (the AmZ8000 CPU MREQ instruction computes the delay by decrementing a 16-bit register). The delay is required in the case of several users requesting access to a particular resource. In this case, the daisy chain link lines (μAI, μAO) in the resource arbitration logic resolves the conflict by granting the resource to the highest priority requestor.
   c. Tests the μAI – if active, then the resource is granted (the AmZ8000 again sets an internal flag); otherwise, the request is terminated.

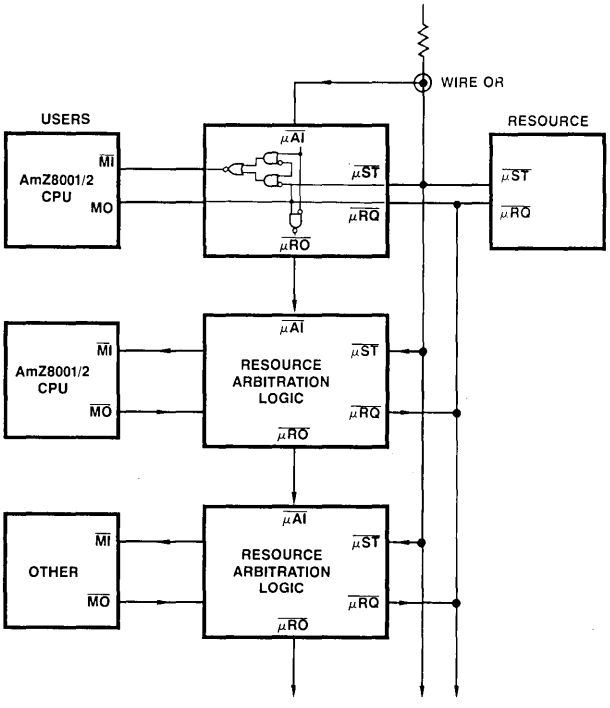


Figure 7. Multimicro Resource Daisy Sharing Chain Link. MOS-284

Bus Request Chain Protocol

Typical data bus transactions follow a master/slave protocol between the AmZ8000 CPU and its peripherals. The CPU is the default bus master and does not require arbitration to access the bus. Some complex peripherals, like the AmZ8016 DMA, may require to obtain bus mastership in order to perform more complex bus transactions than the typical peripheral.

A Bus Request protocol is easily implemented to orderly request the AmZ8000 CPU to grant its bus mastership to one of several peripherals capable of commanding the bus. A daisy chain link priority resolution among bus requestors, along with a bus request protocol, forms an efficient bus mastership transfer. As common to most microcomputers, the AmZ8000 CPU recognizes asynchronous bus requests on its BUSRQ input pin, and grants it upon completion of the current machine cycle by causing its address/data, control and status outputs to go to high impedance and its BUSAK output pin to become active.

The algorithm for bus priority and request for bus requestors is as follows:

1. The requestor looks at its BRQ line to see if it is active. The BRQ line is bidirectional.
2. If the BRQ line is active, then the requestor waits. This implements the policy of non-preemption, since another peripheral could be in process of using or requesting the bus.
3. If the BRQ line is inactive, then the requestor activates its BRQ output and waits for its BAI input to go active. Upon BAI active, the requestor deactivates its BAO output (as opposed to letting the BAI input to ripple-thru to the BAO output), and takes control of the bus.

It is easy to see how the daisy chain link among peripherals is implemented to determine peripheral bus mastership priority.

Typical System Structures

The complexity of an AmZ8000 system mainly depends on the application and the needs of the user. Interface peripheral and memory devices can vary widely upon this complexity. At various stages of complexity, different types of system interface logic must be used depending on the loading requirements. A minimum AmZ8002 system is shown in Figure 9. The AmZ8002 CPU can drive one standard TTL load. An AmZ8127 clock generator is used to provide the system clock. Since the address and data are multiplexed together, an AmZ8173 octal latch is used to latch the addresses. Status is decoded by a Am74LS139 decoder. If loading exceeds one TTL load, buffers should be used as in Figure 10. On the data bus, two AmZ8104s are used to provide 16 bidirectional data lines. While an AmZ8144 buffer is used to give extra fan-out to the control signals. Direct addressing is available up to 64K bytes.

In Figure 11, an AmZ8001 segmented CPU is shown with the AmZ8010 Memory Management Unit. The AmZ8010 extends the memory into 64 segments where each segment can vary between 256 to 64K bytes. Without the AmZ8010, the AmZ8001 CPU can only access 8M bytes directly. This example can drive only TTL load. More drive capability could be added as in Figure 12. The buffering example is the same as Figure 11 with the exception of the AmZ8010 and the AmZ8144 to buffer the segment addresses.

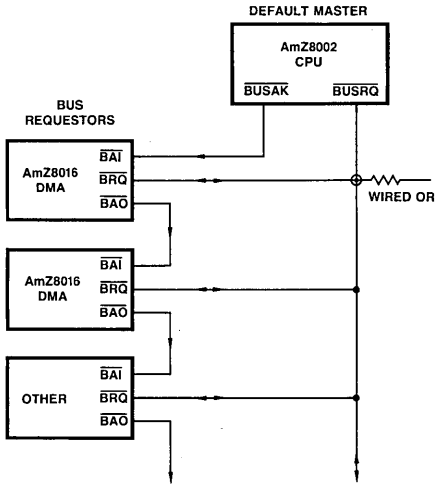


Figure 8. Bus Request Chain Link. MOS-285

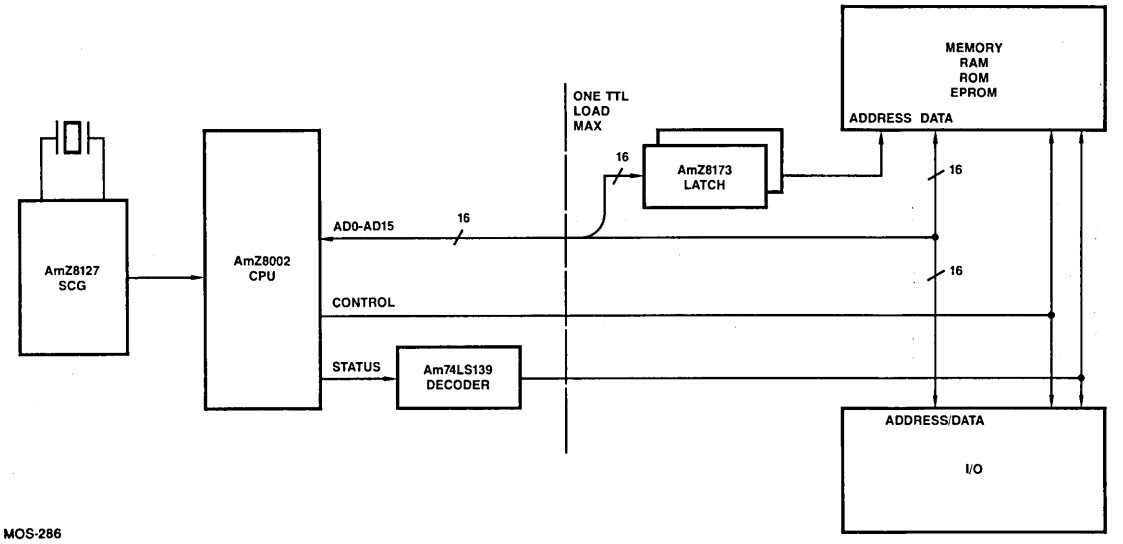


Figure 9. Simple AmZ8002 System Requires Few MSI Circuits, but has Limited Drive Capability.

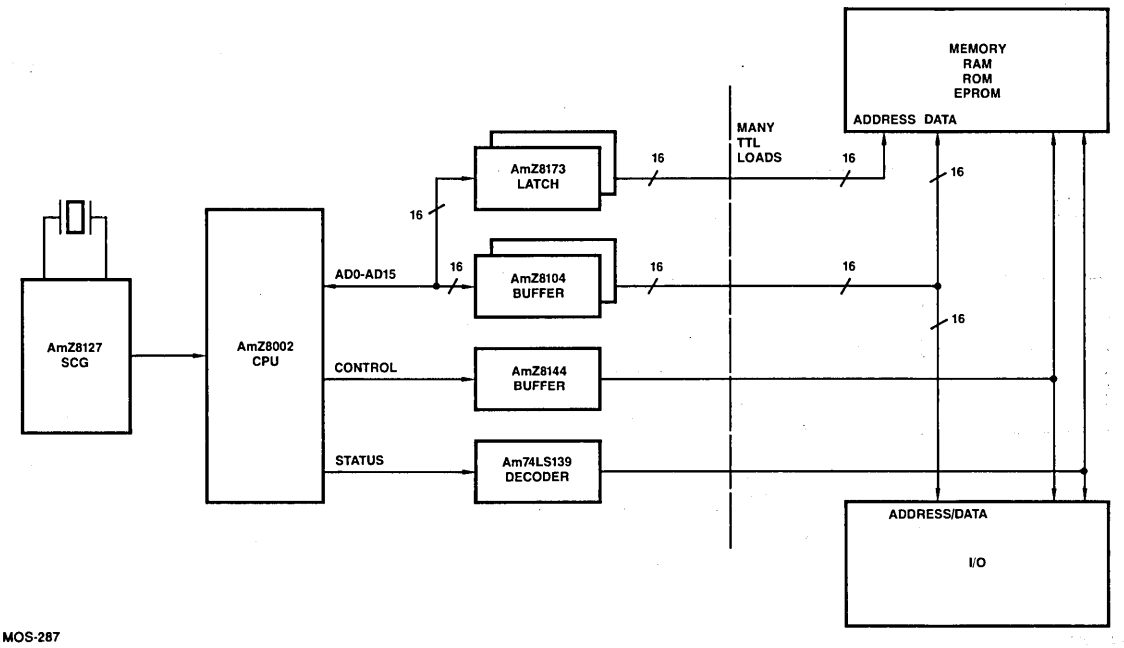


Figure 10. Fully Buffered AmZ8002 System Provides Plenty of Drive for Backplane Buses and Peripherals.

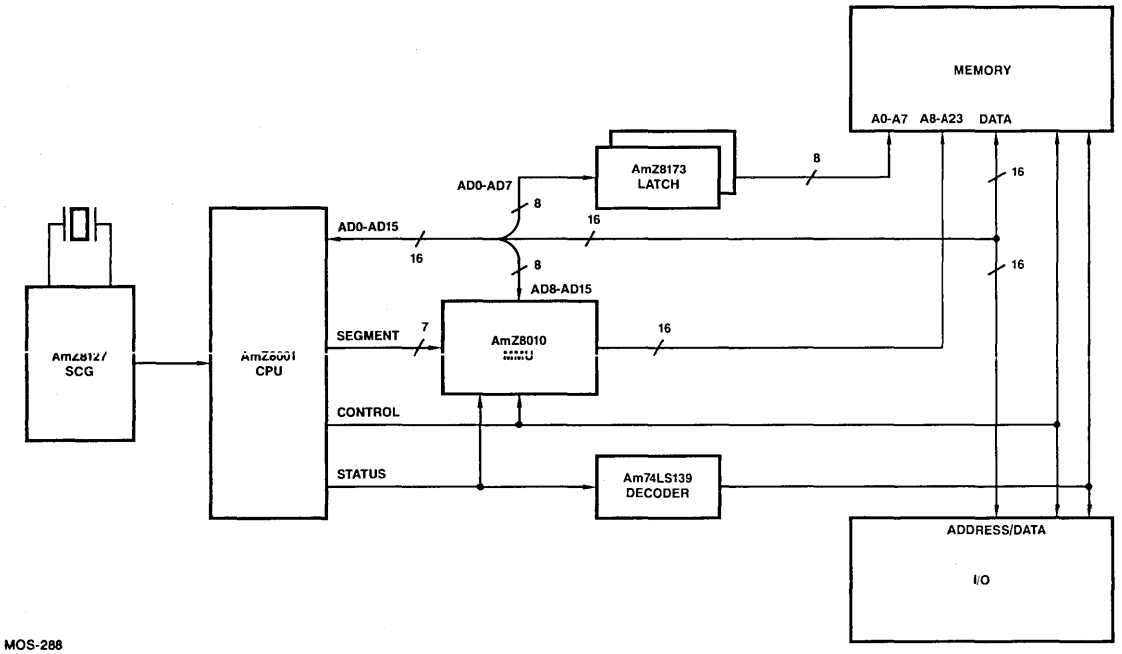


Figure 11. Simple AmZ8001 System with MMU.

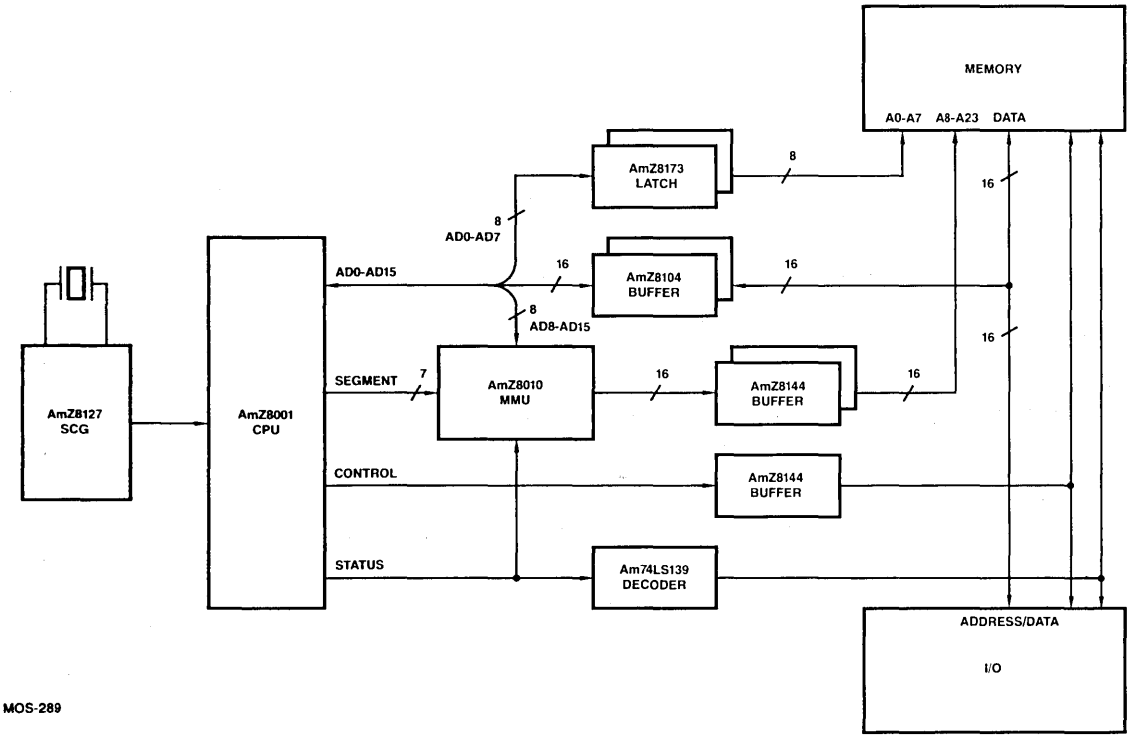
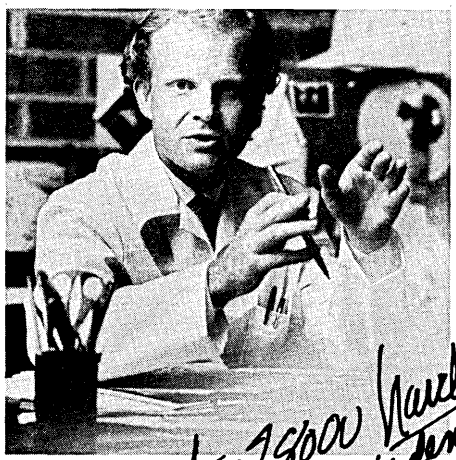


Figure 12. Fully Buffered AmZ8001 System with MMU.

AmZ8000 Microprocessor Family



CHAPTER 4

Components

AmZ8001

16-Bit Microprocessor

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

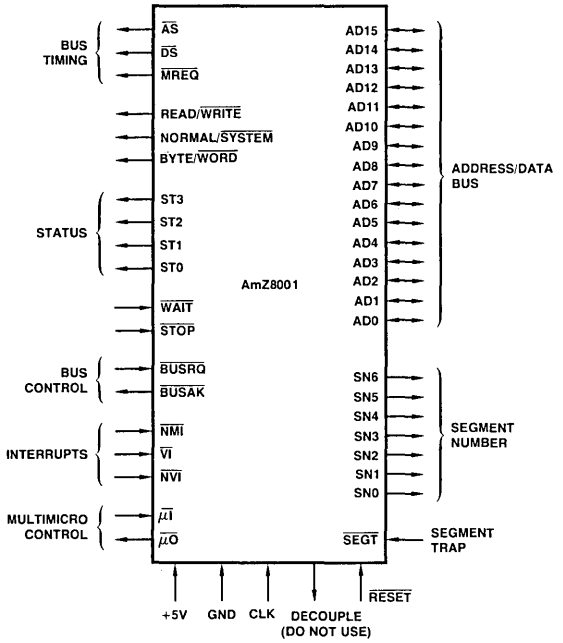
- Sixteen general purpose registers
- Direct addressing up to 8MB segmented memory
- Software compatible with AmZ8002 microprocessor
- Powerful instructions with flexible addressing modes
- Privileged/Non-Privileged mode of operation
- Sophisticated interrupt structure
- On-chip memory refresh facility
- TTL compatible inputs and outputs
- Single phase clock
- Single +5V power supply
- 48-pin package

GENERAL DESCRIPTION

The AmZ8001 is a general-purpose 16-bit CPU belonging to the AmZ8000 family of microprocessors. Its architecture is centered around sixteen 16-bit general registers. The CPU deals with 23-bit address spaces and hence can address directly 8MB of memory. The 23-bit address consists of two components: 7-bit segment number and 16-bit offset. Facilities are provided to maintain three distinct address spaces – code, data and stack. The AmZ8001 implements a powerful instruction set with flexible addressing modes. These instructions operate on several data types – bit, byte, word (16-bit), long word (32-bit), byte string and word string. The CPU can execute instructions in one of two modes – System and Normal. Sometimes these modes are also known as Privileged and Non-Privileged, respectively. The CPU also contains an on-chip memory refresh facility. The AmZ8001 is software compatible with the AmZ8002 microprocessor. The AmZ8001 is fabricated using silicon-gate N-MOS technology and is packaged in a 48-pin DIP. The AmZ8001 requires a single +5 power supply and a single phase clock for its operation.

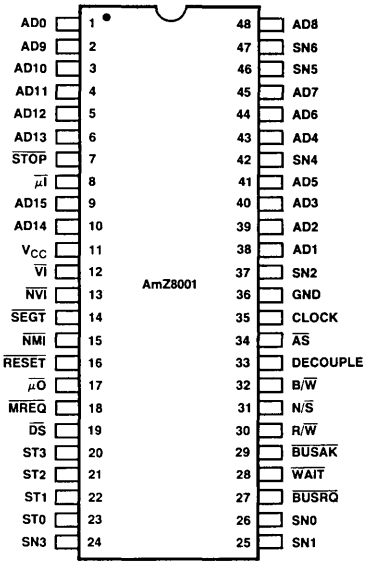
4

LOGIC SYMBOL



MOS-240

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MOS-241

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency 4MHz
Hermetic DIP	0°C ≤ T_A ≤ 70°C	AmZ8001DC

INTERFACE SIGNAL DESCRIPTION

V_{CC}: +5V Power Supply

V_{SS}: Ground

AD0-AD15: Address/Data Bus (Bidirectional, 3-State)

This 16-bit multiplexed address/data bus is used for all I/O and memory transactions. HIGH on the bus corresponds to 1 and LOW corresponds to 0. AD0 is the least significant bit position with AD15 is most significant. The $\overline{AS}$ output and $\overline{DS}$ output will indicate whether the bus is used for address offset or data. The status output lines ST0-ST3 will indicate the type of transaction; memory or I/O.

$\overline{AS}$: Address Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus contains address information. The address information is stable by the time of the LOW-to-HIGH transition of the $\overline{AS}$ output (see timing diagrams). The status outputs ST0-ST3 indicate whether the bus contains a memory address or I/O address.

$\overline{DS}$: Data Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus is being used for data transfer. The R/$\overline{W}$ output indicates the direction of data transfer – read (or in) means data into the CPU and write (or out) means data from the CPU. During a read operation, data can be gated on to the bus when $\overline{DS}$ goes LOW. A LOW-to-HIGH transition on the $\overline{DS}$ output indicates that the CPU has accepted the data (see timing diagram). During a write operation, LOW on the $\overline{DS}$ output indicates that data is setup on the bus. Data will be removed sometime after the LOW-to-HIGH transition of the $\overline{DS}$ output (see timing diagram).

R/$\overline{W}$: Read/Write (Output, 3-State)

This output indicates the direction of data flow on the AD0-AD15 bus. HIGH indicates a read operation, i.e., data into the CPU and LOW indicates a write operation, i.e., data from the CPU. This output is activated at the same time as $\overline{AS}$ going LOW and remains stable for the duration of the whole transaction (see timing diagram).

B/$\overline{W}$: Byte/Word (Output, 3-State)

This output indicates the type of data transferred on the AD0-AD15 bus. HIGH indicates byte (8-bit) and LOW indicates word (16-bit) transfer. This output is activated at the same stage as $\overline{AS}$ going LOW and remains valid for the duration of the whole transaction (see timing diagram). The address generated by the CPU is always a byte address. However, the memory is organized as 16-bit words. All instructions and word operands are word aligned and are addressed by even addresses. Thus, for all word transactions with the memory the least significant address bit will be zero. When addressing the memory for byte transactions, the least significant address bit determines which byte of the memory word is needed; even address specifies the most significant byte and odd address specifies the least significant byte. In the case of I/O transactions, the address information on the AD0-AD15 bus refers to an I/O port and B/$\overline{W}$ determines whether a data word or data byte will be transacted. During I/O byte transactions, the least significant address bit A0 determines which half of the AD0-AD15 bus will be used for the I/O transactions. The ST0-ST3 outputs will indicate whether the current transaction is for memory, normal I/O or special I/O.

ST0-ST3: Status (Outputs, 3-State)

These four outputs contain information regarding the current transaction in a coded form. The status line codes are shown in the following table:

ST3	ST2	ST1	ST0	
L	L	L	L	Internal Operation
L	L	L	H	Memory Refresh
L	L	H	L	Normal I/O Transaction
L	L	H	H	Special I/O Transaction
L	H	L	L	Segment Trap Acknowledge
L	H	L	H	Non-Maskable Interrupt Acknowledge
L	H	H	L	Non-Vectored Interrupt Acknowledge
L	H	H	H	Vectored Interrupt Acknowledge
H	L	L	L	Memory Transaction for Operand
H	L	L	H	Memory Transaction for Stack
H	L	H	L	Reserved
H	L	H	H	Reserved
H	H	L	L	Memory Transaction for Instruction Fetch (Subsequent Word)
H	H	L	H	Memory Transaction for Instruction Fetch (First Word)
H	H	H	L	Reserved
H	H	H	H	Reserved

$\overline{WAIT}$: Wait (Input)

LOW on this input indicates to the CPU that memory or I/O is not ready for the data transfer and hence the current transaction should be stretched. The $\overline{WAIT}$ input is sampled by the CPU at certain instances during the transaction (see timing diagram). If $\overline{WAIT}$ input is LOW at these instances, the CPU will go into wait state to prolong the transaction. The wait state will repeat until the $\overline{WAIT}$ input is HIGH at the sampling instant.

N/$\overline{S}$: Normal/System Mode (Output, 3-State)

HIGH on this output indicates that the CPU is operating in Normal Mode and LOW indicates operation in System Mode. This output is derived from the Flag Control Word (FCW) register. The FCW register is described under the program status information section of this document.

$\overline{MREQ}$: Memory Request (Output, 3-State)

LOW on this output indicates that a CPU transaction with memory is taking place.

$\overline{BUSRQ}$: Bus Request (Input)

LOW on this input indicates to the CPU that another device (such as DMA) is requesting to take control of the bus. The $\overline{BUSRQ}$ input can be driven LOW anytime. The CPU synchronizes this input internally. The CPU responds by activating $\overline{BUSAK}$ output LOW to indicate that the bus has been relinquished. Relinquishing the bus means that the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3, SN0-SN6 and $\overline{MREQ}$ outputs will be in the high impedance state. The requesting device should control these lines in an identical fashion to the CPU to accomplish transactions. The $\overline{BUSRQ}$ input must remain LOW as long as needed to perform all the transactions and the CPU will keep the $\overline{BUSAK}$ output LOW. After completing the transactions, the device must disable the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3, SN0-SN6 and $\overline{MREQ}$ into the high impedance state and stop driving the $\overline{BUSRQ}$ input LOW. The CPU will make $\overline{BUSAK}$ output HIGH sometime later and take back the bus control.

$\overline{BUSAK}$: Bus Acknowledge (Output)

LOW on this output indicates that the CPU has relinquished the bus in response to a bus request.

$\overline{NMI}$: Non-Maskable Interrupt (Input)

HIGH to LOW transition on this input constitutes non-maskable interrupt request. The CPU will respond with the Non-maskable Interrupt Acknowledge on the ST0-ST3 outputs and will enter an interrupt sequence. The transition on the $\overline{NMI}$ can occur anytime. Of the three kinds of interrupts available, the non-maskable interrupt has the highest priority.

$\overline{VI}$: Vectored Interrupt (Input)

LOW on this input constitutes vectored interrupt request. Vectored interrupt is next lower to the non-maskable interrupt in priority. The VIE bit in the Flag and Control Word register must be 1 for the vectored interrupt to be honored. The CPU will respond with Vectored Interrupt Acknowledge code on the ST0-ST3 outputs and will begin the interrupt sequence. The $\overline{VI}$ input can be driven LOW anytime and should be held LOW until acknowledged.

$\overline{NVI}$: Non-Vectored Interrupt (Input)

LOW on this input constitutes non-vectored interrupt request. Non-vectored has the lowest priority of the three types of interrupts. The NVIE bit in the Flag and Control Word register must be 1 for this request to be honored. The CPU will respond with Non-Vectored Interrupt Acknowledge code on the ST0-ST3 outputs and will begin the interrupt sequence. The NVI input can be driven LOW anytime and should be held LOW until acknowledged.

$\overline{\mu I}$: Micro-In (Input)

This input participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{\mu O}$: Micro-Out (Output)

This output participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{RESET}$: Reset (Input)

LOW on this input initiates a reset sequence in the CPU. See the section on Initialization for details on reset sequence.

CLK: Clock (Input)

All CPU operations are controlled from the signal fed into this input. See DC characteristics for clock voltage level requirements.

DECOUPLE: (Output)

Output from the on-chip substrate bias generator. Do not use.

$\overline{STOP}$: Stop (Input)

This active LOW input facilitates one instruction at a time operation. See the section on single stepping.

SN0-SN6: Segment Number (Outputs, 3-State)

These seven outputs contain the segment number part of a memory address. A HIGH on the output corresponds to 1 and a LOW corresponds to 0. SN0 is the least significant bit position and SN6 is the most significant bit position.

$\overline{SEGT}$: Segment Trap (Input)

LOW on this input constitutes a segment trap request. If the line is driven LOW, the CPU will respond with the Segment Trap Acknowledge code on the Status lines, and commence a trap sequence. The $\overline{SEGT}$ input may be driven LOW at any time and is customarily held LOW until acknowledged. This input has priority over the interrupts.

PROCESSOR ORGANIZATION

The following is a brief discussion of the AmZ8001 CPU. For detailed information, see the AmZ8001/AmZ8002 Processor Instruction Set Manual (Publication No. AM-PUB086).

GENERAL PURPOSE REGISTERS

The CPU is organized around sixteen 16-bit general purpose registers R0 through R15 as shown in Figure 1. For byte operations, the first eight registers (R0 through R7) can also be addressed as sixteen 8-bit registers designated as RL0, RH0 and so on to RL7 and RH7. The sixteen registers can also be grouped in pairs RR0, RR2 and so on to RR14 to form eight long word (32-bit) registers. Similarly, the sixteen registers can be grouped in quadruples RQ0, RQ4, RQ8 and RQ12 to form four 64-bit registers.

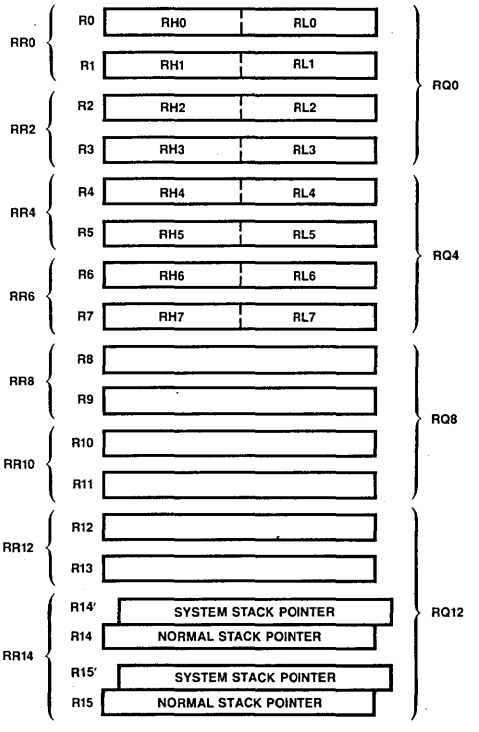


Figure 1. AmZ8001 General Registers.

MOS-242

STACK POINTER

The AmZ8001 architecture allows stacks to be maintained in memory. Any general-purpose register pair except RR0 can be used as a stack pointer in stack manipulating instructions such as PUSH and POP. The designated register pair holds a 23-bit segmented address. Certain instructions (such as subroutine call and return) make implicit use of the register pair RR14 as the stack pointer. Two implicit stacks are allowed – normal stack using RR14 as the stack pointer, and system stack using RR14' as the system stack pointer (see Figure 1). If the CPU is operating in the Normal Mode, RR14 is active, and if the CPU is in System Mode, RR14' will be used instead of RR14. The implied stack pointer is a part of the general registers and hence can be manipulated using the instructions available for register operations.

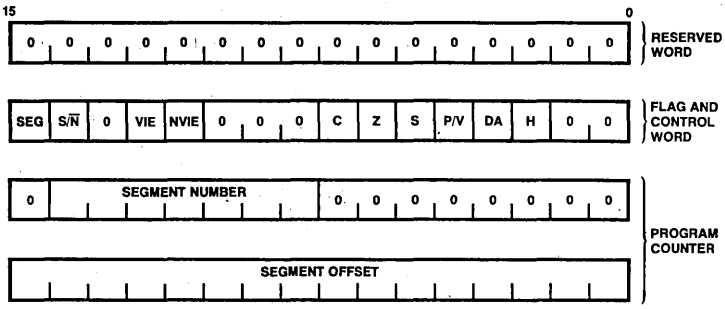


Figure 2. AmZ8001 Processor Status.

MOS-243

PROCESSOR STATUS

The CPU status consists of the 16-bit flag and control word (FCW) register, and the 23-bit program counter (see Figure 2). A reserved word is also included for future expansion. The following is a brief description of the FCW bits.

SEG: Segmented/Non-Segmented Bit. Indicates whether the AmZ8001 is running in segmented or non-segmented mode. 1 indicates segmented, 0 indicates non-segmented. See the section on non-segmented mode, elsewhere in this document.

S/$\overline{N}$: System/Normal – 1 indicates System Mode and 0 indicates Normal Mode.

VIE: Vectored Interrupt Enable – 1 indicates that Vectored Interrupt requests will be honored.

NVIE: Non-Vectored Interrupt Enable – 1 indicates that Non-vectored interrupt requests will be honored.

C: Carry – 1 indicates that a carry has occurred from the most significant bit position when performing arithmetic operations.

Z: Zero – 1 indicates that the result of an operation is zero.

S: Sign – 1 indicates that the result of an operation is negative i.e., most significant bit is one.

P/V: Parity/Overflow – 1 indicates that there was an overflow during arithmetic operations. For byte logical operations this bit indicates parity of the result.

DA: Decimal Adjust – Records byte arithmetic operations.

H: Half Carry – 1 indicates that there was a carry from the most significant bit of the lower digit during byte arithmetic.

DATA TYPES

The AmZ8001 instructions operate on bits, digits (4 bits), bytes (8 bits), words (16 bits), long words (32 bits), byte strings and word strings type operands. Bits can be set, reset or tested. Digits are used to facilitate BCD arithmetic operations. Bytes are used for characters and small integers. Words are used for integer values and addresses while long words are used for large integer values and addresses. All operands except strings can reside either in memory or general registers. Strings can reside in memory only.

INTERRUPT AND TRAP STRUCTURE

Interrupt is defined as an external asynchronous event requiring program interruption. For example, interruption is caused by a peripheral needing service. Traps are synchronous events resulting from execution of certain instructions under some defined circumstances. Both interrupts and traps are handled in a similar manner by the AmZ8001.

The AmZ8001 supports three types of interrupts in order of descending priority – non-maskable, vectored and non-vectored. The vectored and non-vectored interrupts can be disabled by appropriate control bits in the FCW. The AmZ8001 has four traps – system call, segment trap, unimplemented opcode and privileged instruction. The traps have higher priority than interrupts.

When an interrupt or trap occurs, the current program status is automatically pushed on to the system stack. The program status consists of processor status (i.e., PC and FCW) plus a 16-bit identifier. The identifier contains the reason, source and other coded information relating to the interrupt or trap.

After saving the current program status, the new processor status is automatically loaded from the new program status area located in the memory. This area is designated by the New Program Status Area Pointer (NPSAP) register. See AM-PUB086 publication for further details.

SEGMENTED ADDRESSING

The AmZ8001 can directly address up to 8MB of memory space, using a 23-bit segmented address. The memory space is divided up into 128 segments, each up to 64KB in size. The upper seven bits of address designate the segment number, and are available on the SN0-SN6 outputs during a memory transaction. See the section on memory transactions for details.

The lower sixteen bits of address designate an offset within the segment, relative to the start of the segment, and are available on AD0-AD15 during part of the memory transaction. See the section on memory transactions for details.

The segmented address may be stored as a long word in memory, or in a register pair. The segment number and offset can be manipulated separately or together, by suitable use of the instruction set.

When the segmented address is contained in code space, a short offset format may be adopted. The segmented address is stored as one word, seven bits of segment number and eight bits of offset. Figure 3 shows the format for segmented addresses.

ADDRESSING MODES

Information contained in the AmZ8001 instructions consists of the operation to be performed, the operand type and the location of the operands. Operand locations are designated by general register addresses, memory addresses or I/O addresses. The addressing mode of a given instruction defines the address space referenced and the method to compute the operand address. Addressing modes are explicitly specified or implied in an instruction. Figure 4 illustrates the eight explicit addressing modes: Register (R), Immediate (IM), Indirect Register (IR), Direct Address (DA), Indexed (X), Relative Address (RA), Base Address (BA) and Base Indexed (BX).

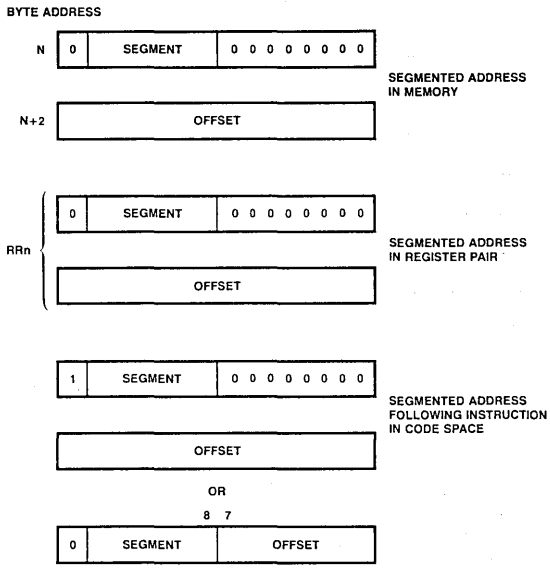


Figure 3. Segmented Address Formats.

MOS-244

When an effective segmented address is being computed according to the designated addressing mode, the segment number is not affected by any carry from the 16-bit offset.

NON-SEGMENTED MODE ON THE AmZ8001

The AmZ8001 can execute code designed to run on the non-segmented AmZ8002. This is achieved by changing the mode of execution of the AmZ8001 from segmented to non-segmented by writing a 0 to the SEG bit in the FCW. (See the section on processor status.) The change to non-segmented mode sets up a suitable environment for running non-segmented code. However, this environment only exists within the code segment that caused the change of mode from segmented to non-segmented.

SN0-SN6 will continue to indicate the code segment until a reset, interruption or return to segmented mode is encountered.

The effects of the non-segmented mode of operation on the AmZ8001 are described below.

a) The AmZ8001 will interpret instruction length as if it was a non-segmented AmZ8002.
b) The AmZ8001 will implement address computation in an identical manner to the AmZ8002.

Other CPU functions, such as interrupt and trap handling, reset and stack pointer manipulation are unaltered. These functions are characterized by the type of CPU, not by the state of the SEG bit in the FCW.

INPUT/OUTPUT

A set of I/O instructions are provided to accomplish byte or word transfers between the AmZ8001 and I/O devices. I/O devices are addressed using 16-bit I/O port addresses and I/O address space is not a part of the memory address space. Two types of I/O instructions are provided; each with its own 16-bit address space. I/O instructions include a comprehensive set of In, Out and Block transfers.

CPU TIMING

The AmZ8001 accomplishes instruction execution by stepping through a pre-determined sequence of machine cycles, such as memory read, memory write, etc. Each machine cycle requires between three and ten clock cycles. Bus Requests by DMA devices are granted at machine cycle boundaries. No machine cycle is longer than ten clock cycles; thus assuring fast response to a Bus Request (assuming no extra wait states). The start of a machine cycle is always marked by a LOW pulse on the $\overline{AS}$ output. The status output lines ST0-ST3 indicate the nature of the current cycle in a coded form.

STATUS LINE CODES

Status line coding was listed in the table shown under ST0-ST3 outputs in the Interface Signal Description. The following is a detailed description of the status codes.

Internal Operation:

This status code indicates that the AmZ8001 is going through a machine cycle for its internal operation. Figure 5 depicts an internal operation cycle. It consists of three clock periods identified as T1, T2 and T3. The $\overline{AS}$ output will be activated with a LOW pulse by the AmZ8001 to mark the start of a machine cycle. The ST0-ST3 will reflect the code for the internal operation. The $\overline{MREQ}$, $\overline{DS}$ and R/$\overline{W}$ outputs will be HIGH. The N/$\overline{S}$ and SN0-SN6 outputs will remain at the same level as in the previous machine cycle. The AmZ8001 will ignore the $\overline{WAIT}$ input during the internal operation cycle. The CPU will drive the AD0-AD15 bus with unspecified information during T1. However, the bus will go into high impedance during T2 and remain in that state for the remainder of the cycle. The B/$\overline{W}$ output is also activated by the CPU with unspecified information.

Memory Refresh:

This status code indicates that AmZ8001 is accessing the memory to refresh. The refresh cycle consists of three clock periods as depicted in Figure 6. The CPU will activate the $\overline{AS}$ output with a LOW pulse to mark the beginning of a machine cycle and ST0-ST3 outputs will reflect the refresh cycle code. The least significant 9 lines of the AD0-AD15 bus contain the refresh address. Because the memory is word organized, the AD0 will always be LOW. The most significant 7 bus lines are not specified. The $\overline{DS}$ output will remain HIGH for the entire cycle while R/$\overline{W}$, B/$\overline{W}$, SN0-SN6 and N/$\overline{S}$ outputs will remain at the same level as in the machine cycle prior to refresh. The AD0-AD15 bus will go into high impedance state during T2 period and remain there for the remainder of the cycle. The AmZ8001 will activate the $\overline{MREQ}$ output LOW during the refresh cycle. It should be noted that $\overline{WAIT}$ input is ignored by the CPU for refresh operations.

I/O Transactions:

There are two status line codes used for I/O transaction cycles. The AmZ8001 provides two separate I/O spaces and two types of instructions called Normal I/O and Special I/O. Each I/O space is addressed by a 16-bit address called port address. The timing for both types of I/O transactions is essentially identical. A typical I/O cycle consists of four clock periods T1, T2, TWA and T3 as shown in Figure 7. The TWA is the wait state; insertion of one wait state for an I/O cycle is always automatic. Additional wait cycles can be inserted by LOW on the $\overline{WAIT}$ input. The $\overline{WAIT}$ input is sampled during every TW state. If this input is LOW, one more wait state will be inserted. Insertion of wait states continues until $\overline{WAIT}$ input is HIGH. T3 state will follow the last wait state to complete the I/O cycle.

Mode	Operand Addressing			Operand Value
	In the Instruction	In a Register	In Memory	
Register		REGISTER ADDRESS → OPERAND		The content of the register.
Immediate	OPERAND			In the instruction
Indirect Register		REGISTER ADDRESS → ADDRESS	→ OPERAND	The content of the location whose address is in the register.
Direct Address	ADDRESS		→ OPERAND	The content of the location whose address is in the instruction.
Index		REGISTER ADDRESS → DISPLACEMENT BASE ADDRESS	⊕ → OPERAND	The content of the location whose address is the address in the instruction, offset by the content of the working register.
Relative Address		PC VALUE DISPLACEMENT	⊕ → OPERAND	The content of the location whose address is the content of the program counter, offset by the displacement in the instruction.
Base Address		REGISTER ADDRESS → BASE ADDRESS DISPLACEMENT	⊕ → OPERAND	The content of the location whose address is the address in the register, offset by the displacement in the instruction.
Base Index		REGISTER ADDRESS → BASE ADDRESS REGISTER ADDRESS → DISPLACEMENT	⊕ → OPERAND	The content of the location whose address is the address in the register, offset by the displacement in the register.

Figure 4. Addressing Modes.

MOS-245

During I/O cycles the ST0-ST3 outputs will reflect appropriate code depending on the type of instruction being executed (Normal I/O or Special I/O). $\overline{AS}$ output will be pulsed LOW to mark the beginning of the cycle. The CPU drives the AD0-AD15 bus with the 16-bit port address specified by the current instruction. The N/$\overline{S}$ output will be LOW indicating that CPU is operating in the system mode. It should be recalled that the N/$\overline{S}$ output is derived from the appropriate bit in the FCW register. All I/O instructions are privileged instructions and will be allowed to execute only if the FCW specifies system mode operation. The $\overline{MREQ}$ output will be HIGH. The AmZ8001 I/O instructions provide both word or byte transactions. The B/$\overline{W}$ output will be HIGH or LOW depending whether the instruction specifies a byte or word transfer. The SN0-SN6 output will remain at the same level as in the machine cycle prior to the I/O cycle.

Two kinds of I/O transfers should be considered: Data In means reading from the device and Data Out means writing into the device. For In operations, the R/$\overline{W}$ output will be HIGH. The AD0-AD15 bus will go into high impedance state during T2. During byte input instructions, the CPU reads either the even or odd half of the Data Bus dependent upon the port address. If the port address is even, the most significant half of the Data Bus is read. If the port address is odd, the least significant half of the Data Bus is read. During word input instructions, the CPU reads all 16 bits of the Data Bus. The AmZ8001 will drive the $\overline{DS}$

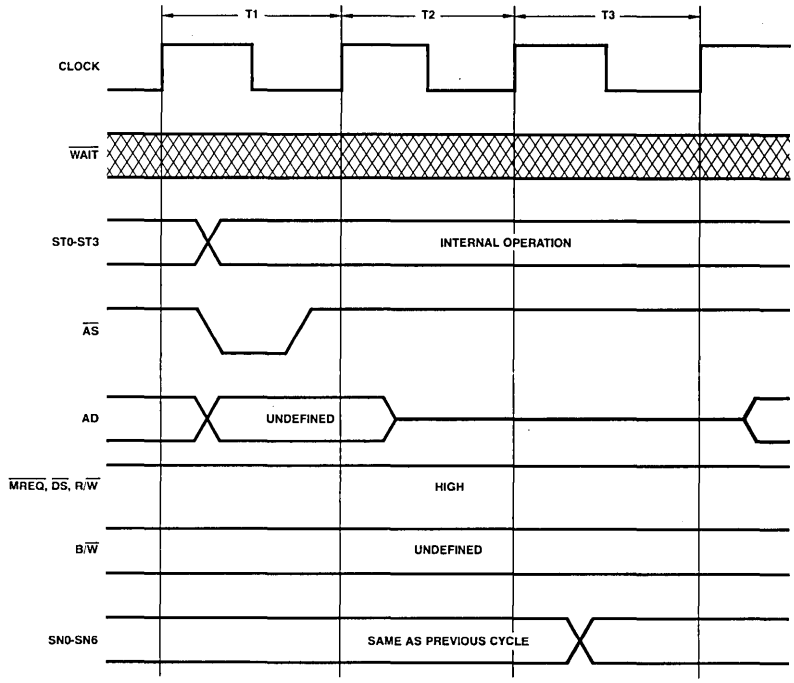


Figure 5. Internal Operation Cycle.

MOS-246

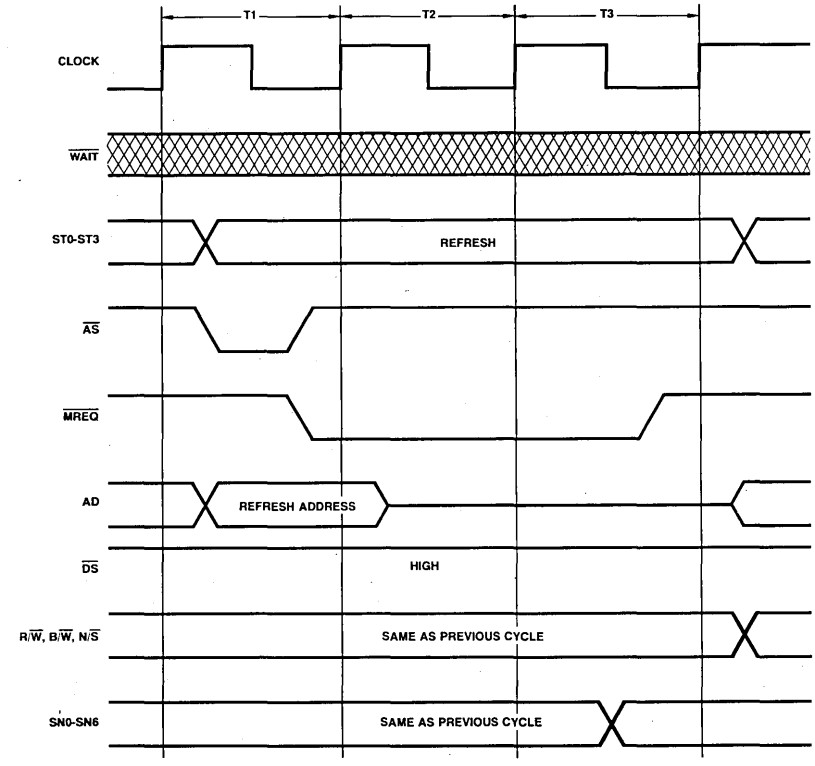


Figure 6. Refresh Cycle.

MOS-247

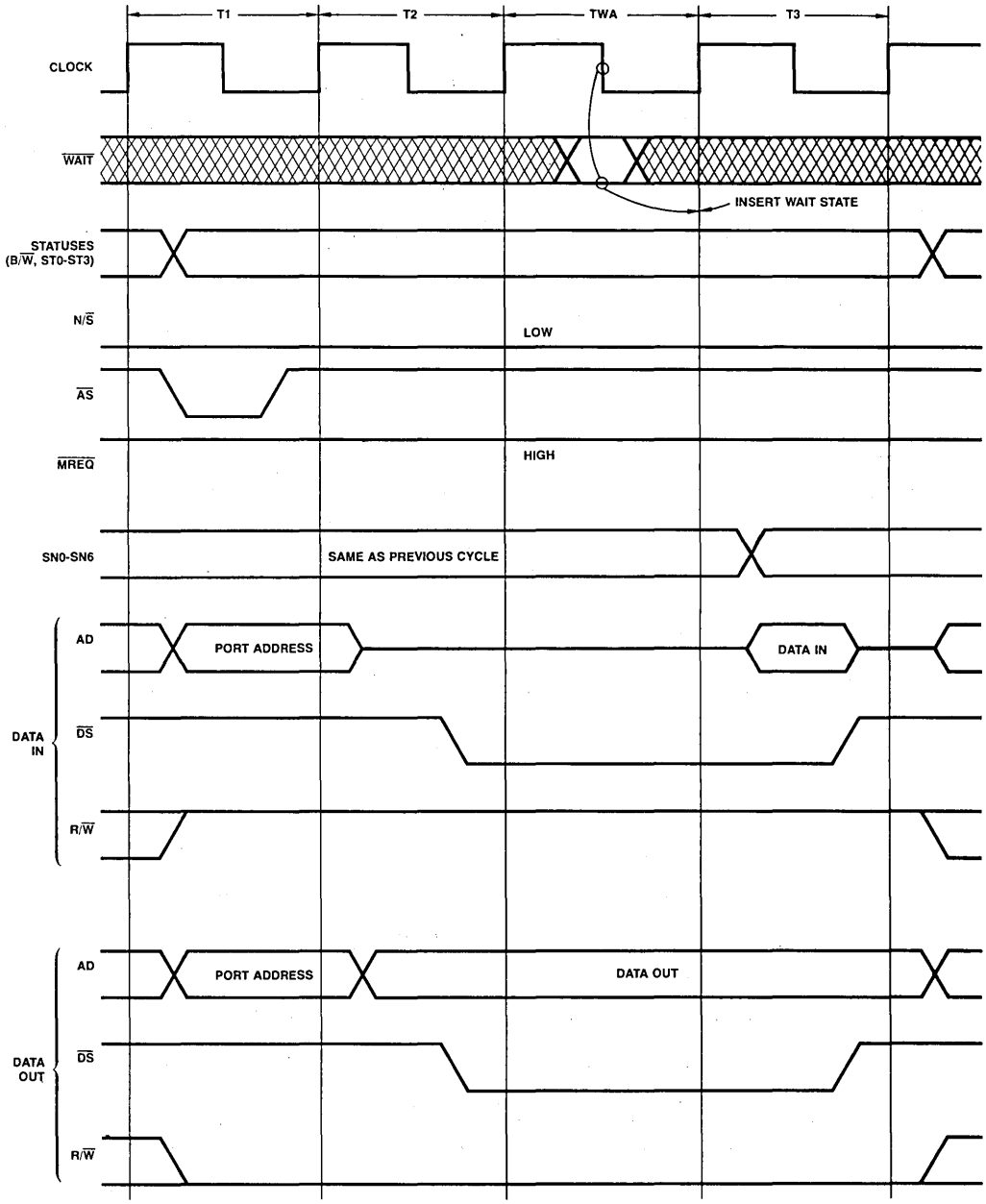


Figure 7. AmZ8001 I/O Cycle.

MOS-248

output LOW to signal to the device that data can be gated on to the bus. The CPU will accept the data during T3 and $\overline{DS}$ output will go HIGH signalling the end of an I/O transaction.

For Data Out, the R/$\overline{W}$ output will be LOW. The AmZ8001 will provide data on the AD0-AD15 bus and activates the $\overline{DS}$ output LOW during T2. During byte output instructions, the CPU duplicates the byte data onto both the high and low halves of the Data Bus and external logic, using A0, enables the appropriate byte port. During word output instructions the CPU outputs data onto all 16 bits of the Data Bus. The $\overline{DS}$ output goes HIGH during T3 and the cycle is complete.

Memory Transactions:

There are four status line codes that indicate a memory transaction:

a) Memory transaction to read or write an operand
b) Memory transaction to read from or write into the stack
c) Memory transaction to fetch the first word of an instruction (sometimes called IF1)
d) Memory transaction to fetch the subsequent word of an instruction (sometimes called IFN).

It can be appreciated that all the above transactions essentially fall into two categories: memory read and memory write. In the case of IF1 and IFN cycles, the memory will be read at the address supplied by the program counter. All AmZ8001 instructions are multiples of 16-bit words. Words are always addressed by an even address. Thus IF1 and IFN cycles involve performing a memory read for words. On the other hand, a memory transaction for operand and stack operation could be a read or write. Moreover, an operand could be a word or a byte. For stack operation involving the implied stack pointer the address will be supplied by the RR14 (or RR14'). For operand transactions, the memory address will come from several sources depending on the instruction and the addressing mode. Memory transaction cycle timing is shown in Figure 8. It typically consists of three clock periods T1, T2 and T3. Wait states (TW) can be inserted

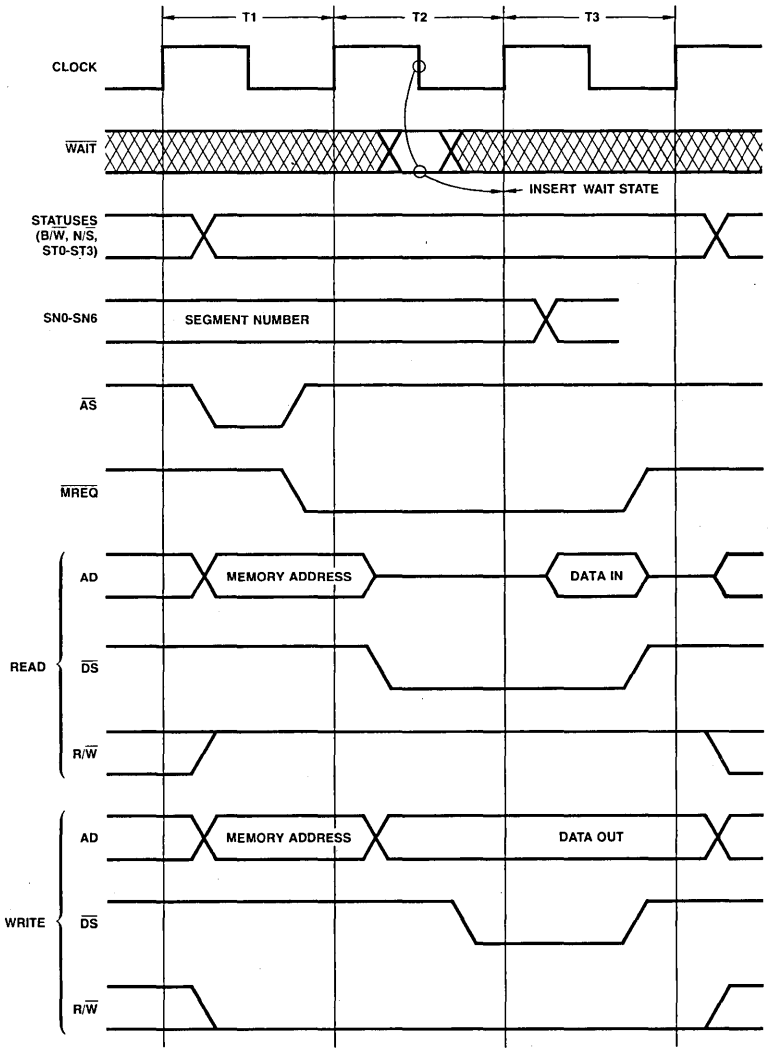


Figure 8. Memory Transactions.

MOS-249

between T2 and T3 by activating the $\overline{WAIT}$ input LOW. The $\overline{WAIT}$ input will be sampled during T2 and during every subsequent TW. The ST0-ST3 outputs will reflect the appropriate code for the current cycle early in T1 and the $\overline{AS}$ output will be pulsed LOW to mark the beginning of the cycle. The N/$\overline{S}$ output will indicate whether the normal or system address space will be used for the current cycle. As shown in the figure the $\overline{MREQ}$ output will go LOW during T1 to indicate a memory operation.

The segment number becomes valid on the segment lines one clock period before the start of the memory operation, and remains valid until the start of T3.

Consider a read operation first. The R/$\overline{W}$ output will be HIGH. The AmZ8001 will drive the AD0-AD15 with the appropriate address early in T1. During T2, the bus will go into high-impedance state and $\overline{DS}$ output will be activated LOW by the CPU. The data can be gated on to the bus when $\overline{DS}$ is LOW. During T1 the B/$\overline{W}$ will also be activated to indicate byte or word will be transacted. The AmZ8001 memory is word organized and words are addressed by even addresses. However, when addressing bytes, the memory address may be odd or even; an even address for most significant byte of a word and the next odd address for the least significant byte of that word. When reading a byte from the memory, the least significant address bit can be ignored and the whole word containing the desired byte is gated on to the bus. The CPU will pick the appropriate byte automatically. The AmZ8001 will drive the $\overline{DS}$ output HIGH indicating data acceptance.

Consider the write operation next. The R/$\overline{W}$ output will be LOW. The AmZ8001 removes the address and gates out the data to be written on the bus and activates the $\overline{DS}$ output LOW during T2. If the data to be written is a byte then the same byte will be on both halves of the bus. The $\overline{DS}$ output will go HIGH during T3 signifying completion of the cycle.

Interrupt and Segment Trap Acknowledge:

There are four status line codes devoted to interrupt and trap acknowledgement. These correspond to non-maskable, vectored and non-vectored interrupts, as well as segment trap. The $\overline{NMI}$ input of the AmZ8001 is edge detected i.e., a HIGH to LOW input level change is stored in an internal latch. Similar internal storage is not provided for the $\overline{VI}$, $\overline{NVI}$, and $\overline{SEGT}$ inputs. For $\overline{VI}$ and $\overline{NVI}$ inputs to cause an interruption, the corresponding interrupt enable bits in the FCW must be 1. For the following discussion, both the VIE and NVIE bits in the FCW are assumed to be 1.

As shown in the figure, the $\overline{VI}$, $\overline{NVI}$ and $\overline{SEGT}$ input and the internal $\overline{NMI}$ latch output are sampled during T3 of the last machine cycle of an instruction.

A LOW on these signals triggers the corresponding interrupt acknowledge sequence described on the following page. The AmZ8001 executes a dummy IF1 cycle prior to entering the actual acknowledge cycle (see memory transactions for IF1 cycle description).

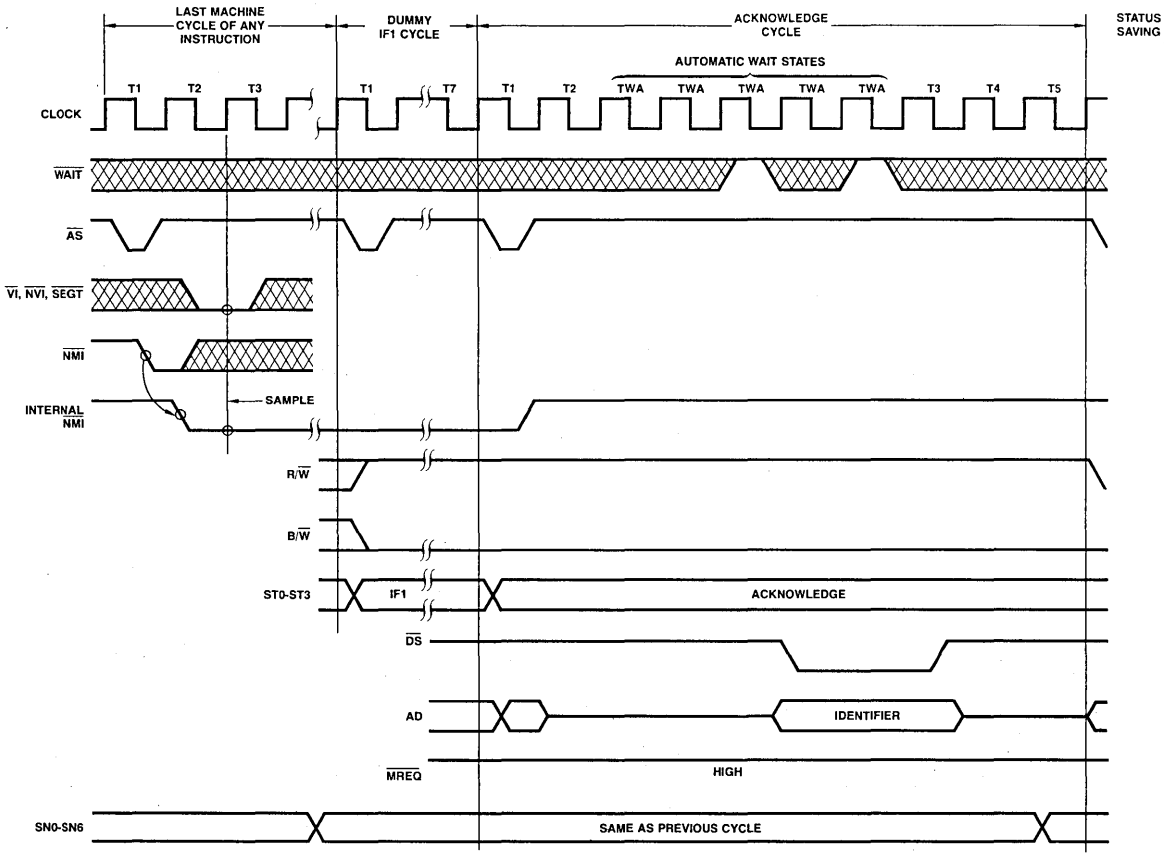


Figure 9. Interrupt Acknowledge Cycle. MOS-250

During this dummy IF1 cycle, the program counter is not updated; instead the implied system stack pointer (RR14') will be decremented. Following the dummy IF1 cycle is the actual interrupt/trap acknowledge cycle.

The interrupt acknowledge cycle typically consists of 10 clock periods; T1 through T5 and five automatic TW (wait states). As usual, the $\overline{AS}$ output will be pulsed LOW during T1 to mark the beginning of a cycle. The ST0-ST3 outputs will reflect the appropriate interrupt acknowledge code, the $\overline{MREQ}$ output will be HIGH, the N/$\overline{S}$ output remains the same as in the preceding cycle, the R/$\overline{W}$ output will be HIGH and the B/$\overline{W}$ output will be LOW. The AmZ8001 will drive the AD0-AD15 bus with unspecified information during T1 and the bus will go into the high impedance state during T2. Three TWA states will automatically follow T2. The $\overline{WAIT}$ input will be sampled during the third TWA state.

If LOW, an extra TW state will be inserted and the $\overline{WAIT}$ will be sampled again during TW. Such insertion of TW states continues until the $\overline{WAIT}$ input is HIGH. After the last TW state, the $\overline{DS}$ output will go LOW and two more automatic wait states (TWA) follow. The interrupting device can gate up to a 16-bit identifier on to the bus when the $\overline{DS}$ output is LOW. The $\overline{WAIT}$ input will be sampled again during the last TWA state. If the $\overline{WAIT}$ input is LOW one TW state will be inserted and the $\overline{WAIT}$ will be sampled during TW. Such TW insertion continues until the $\overline{WAIT}$ input is HIGH. After completing the last TW state T3 will be entered and the $\overline{DS}$ output will go HIGH. The interrupting device should remove the identifier and cease driving the bus. T4 and T5 states will follow T3 to complete the cycle. Following the interrupt acknowledge cycle will be memory transaction cycles to save the status on the stack. Note that the N/$\overline{S}$ output will be automatically LOW during status saving. The SN0-SN6 outputs are undefined during the acknowledge cycle.

The internal NMI latch will be reset to the initial state at $\overline{AS}$ going HIGH in the interrupt acknowledge cycle. The $\overline{VI}$, $\overline{NVI}$ and $\overline{SEGT}$ input should be kept LOW until this time also.

STATUS SAVING SEQUENCE:

The machine cycles following the interrupt acknowledge cycle push the old status information on the system stack in the following order: program counter, the flag and control word and the interrupt/trap identifier. Subsequent machine cycles fetch the new program status from the new program status area, and then branch to the interrupt/service routine.

BUS REQUEST/BUS ACKNOWLEDGE TIMING:

A LOW on the $\overline{BUSRQ}$ input is an indication to the AmZ8001 that another device (such as DMA) is requesting control of the bus. The $\overline{BUSRQ}$ input is synchronized internally at T1 of any machine cycle. (See next paragraph for exception.) The $\overline{BUSAK}$ will go LOW after the last clock period of the machine cycle. The LOW on the $\overline{BUSAK}$ output indicates acknowledgement. When $\overline{BUSAK}$ is LOW the following outputs will go into the high impedance state; AD0-AD15, $\overline{AS}$, $\overline{DS}$, $\overline{MREQ}$, ST0-ST3, B/$\overline{W}$, R/$\overline{W}$, SN0-SN6 and N/S. The BUSRQ must be held LOW until all transactions are completed. When $\overline{BUSRQ}$ goes HIGH, it is synchronized internally, the $\overline{BUSAK}$ output will go HIGH and normal CPU operation will resume. Figure 10 illustrates the $\overline{BUSRQ}$/$\overline{BUSAK}$ timing.

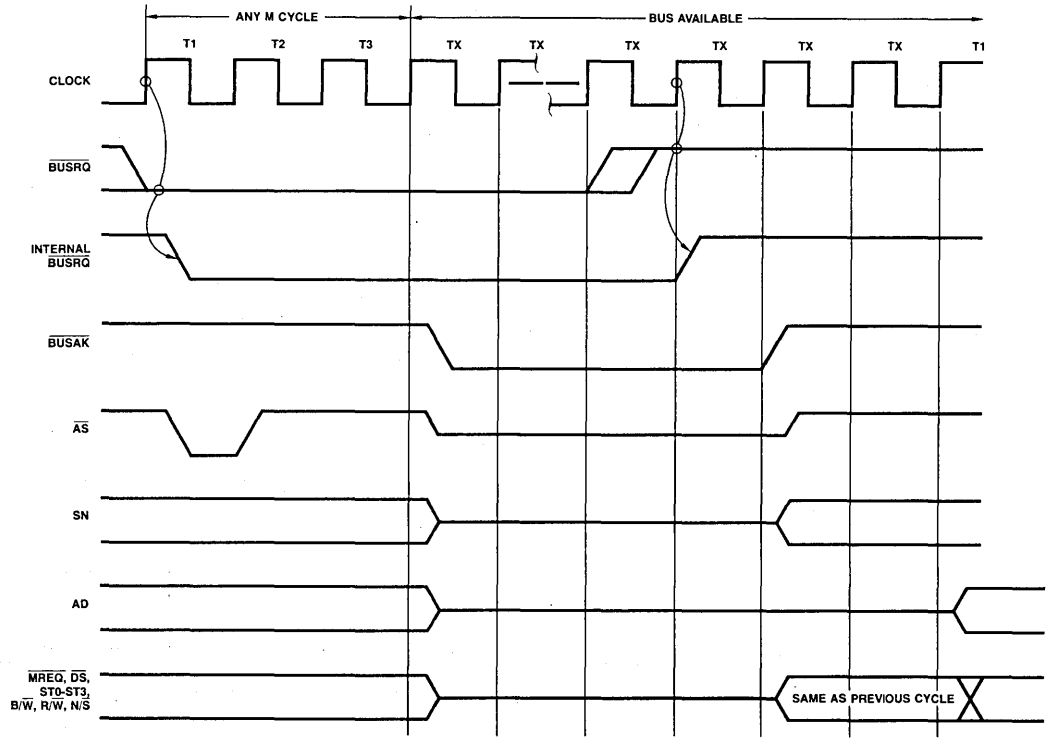


Figure 10. Bus Request/Acknowledge Cycle.

MOS-251

It was mentioned that $\overline{BUSRQ}$ will be honored during any machine cycle with one exception. This exception is during the execution of TSET/TSETB instructions. $\overline{BUSRQ}$ will not be honored once execution of these instructions has started.

SINGLE STEPPING

The $\overline{STOP}$ input of the AmZ8001 facilitates one instruction at a time or single step operation. Figure 11 illustrates $\overline{STOP}$ input timing. The $\overline{STOP}$ input is sampled on the HIGH to LOW transition of the clock input that immediately precedes an IF1 cycle. If the $\overline{STOP}$ is found LOW, AmZ8001 introduces a memory refresh cycle after T3. Moreover, $\overline{STOP}$ input will be sampled again at T3 in the refresh cycle. If $\overline{STOP}$ is LOW one more refresh cycle will follow the previous refresh cycle. The $\overline{STOP}$ will be sampled during T3 of the refresh cycle also. One additional refresh cycle will be added every time $\overline{STOP}$ input is sampled LOW. After completing the last refresh cycle which will occur after $\overline{STOP}$ is HIGH, the CPU will insert two dummy states T4 and T5 to complete the IF1 cycle and resume its normal operations for executing the instruction. See appropriate sections on memory transactions and memory refresh. It should be noted that refresh cycles will occur even if the refresh facility is disabled during single stepping.

MULTIMICROPROCESSOR FACILITIES

The AmZ8001 is provided with hardware and software facilities to support multiple microprocessor systems. The $\overline{\mu O}$ and $\overline{\mu I}$ signals of the AmZ8001 are used in conjunction with the MBIT, MREQ, MRES and MSET instructions for this purpose. The $\overline{\mu O}$ output can be activated LOW by using appropriate instruction to signal a request from the AmZ8001 for a resource. The $\overline{\mu I}$ input is tested by the AmZ8001 before activating the $\overline{\mu O}$ output. LOW at the $\overline{\mu I}$ input indicates that the resource is busy. The AmZ8001 can examine the $\overline{\mu I}$ input after activating the $\overline{\mu O}$ output LOW. The $\overline{\mu I}$ will be tested again to see if the requested resource became available. For detailed information on the Multimicroprocessor facilities, AmZ8001/AmZ8002 Processor Interface Manual (Publication No. AM-PUB089) should be consulted.

INITIALIZATION

A LOW on the $\overline{Reset}$ input starts the CPU initialization. The initialization sequence is shown in Figure 12. Within five clock periods after the HIGH to LOW level change of the $\overline{Reset}$ input the following will occur:

a) AD0-AD15 bus will be in the HIGH impedance state
b) $\overline{AS}$, $\overline{DS}$, $\overline{MREQ}$, $\overline{BUSAK}$ and $\overline{\mu O}$ outputs will be HIGH
c) ST0-ST3 outputs will be LOW
d) Refresh will be disabled
e) R/$\overline{W}$, B/$\overline{W}$ and N/$\overline{S}$ outputs are not affected. For a power on reset the state of these outputs is not specified.
f) SN0-SN6 outputs will be LOW.

After the $\overline{Reset}$ input returns HIGH and remains HIGH for three clock periods, three 16-bit memory read operations will be performed as follows from segment C. Note that the N/$\overline{S}$ output will be LOW and ST0-ST3 outputs will reflect IFN code.

a) The contents of the memory location 0002 will be read. This information will be loaded into the FCW of the AmZ8001.
b) The contents of the memory location 0004 will be read. This information will be loaded into the program counter segment number.
c) The contents of the memory location 0006 will be read. This information will be loaded into the program counter offset.

This completes initialization sequence and an IF1 cycle will follow to fetch the first instruction to begin program execution. See the section on memory transactions for timing.

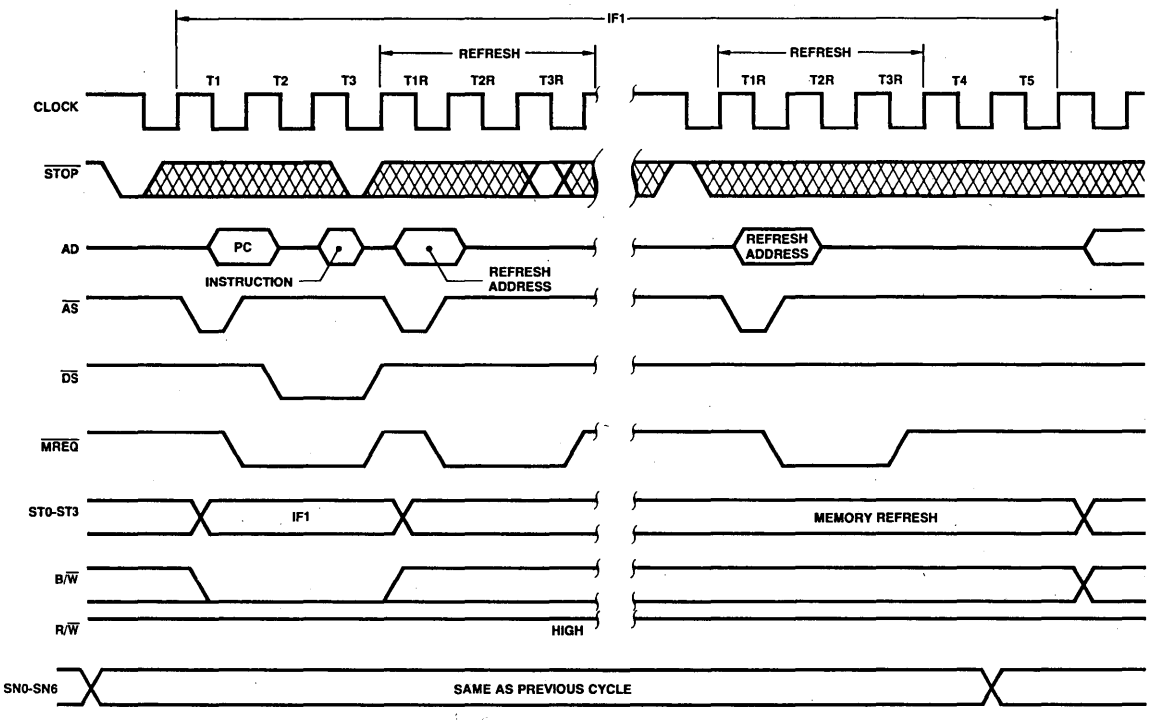


Figure 11. Single Step Timing.

MOS-252

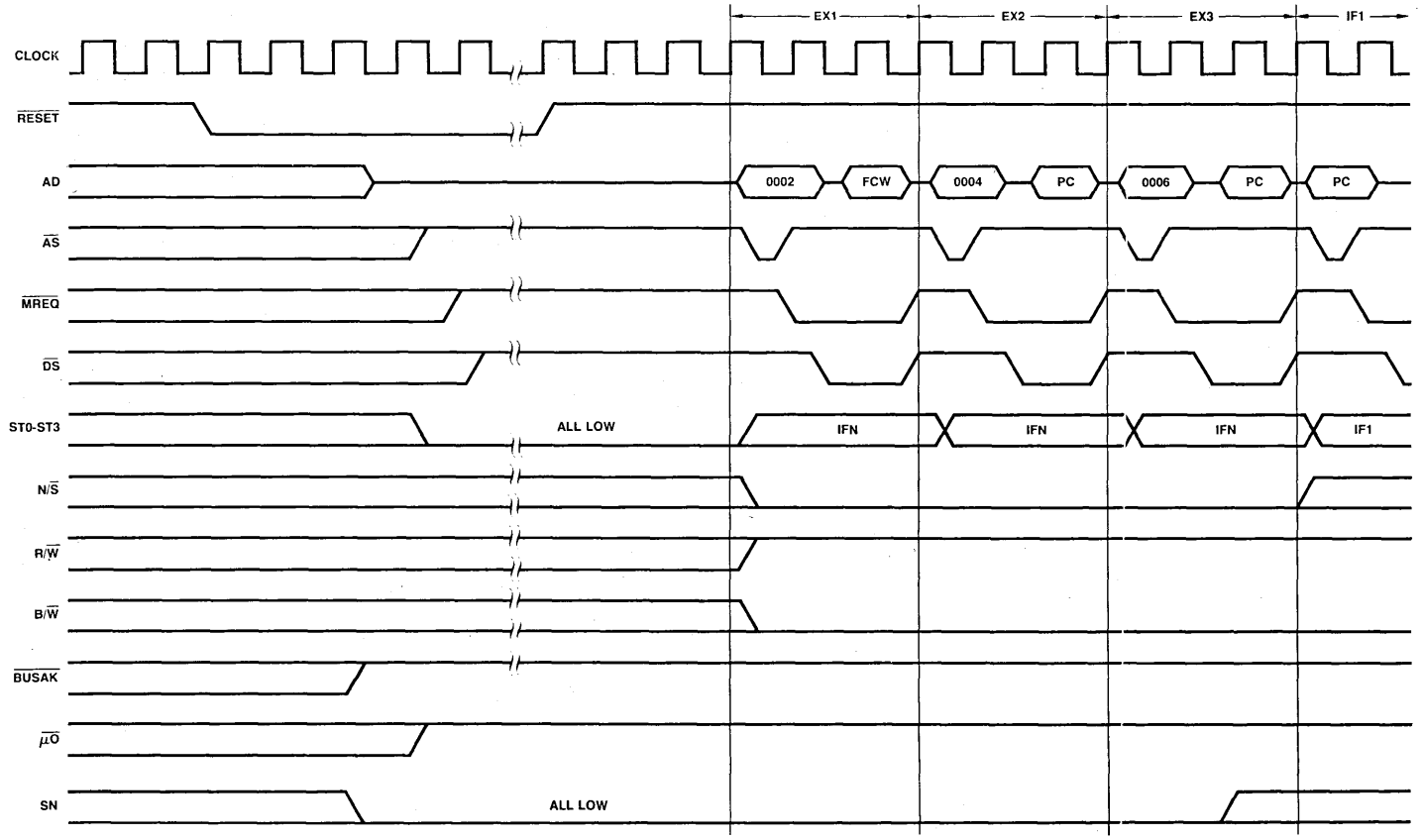


Figure 12. Reset Sequence.

AmZ8001 INSTRUCTION SET

LOAD AND EXCHANGE

Mnemonics	Operands	Addr. Modes	Operation
CLR CLRB	dst	R IR DA X	Clear dst ← 0
EX EXB	R, src	R IR DA X	Exchange R ← src
LD LDB LDL	R, src	R IM IM IR DA X BA BX	Load into Register R ← src
LD LDB LDL	dst, R	IR DA X BA BX	Load into Memory (Store) dst ← R
LD LDB	dst, IM	IR DA X	Load Immediate into Memory dst ← IM
LDA	R, src	DA X BA BX	Load Address R ← source address
LDAR	R, src	RA	Load Address Relative R ← source address
LDK	R, src	IM	Load Constant R ← n (n = 0 . . . 15)
LDM	R, src, n	IR DA X	Load Multiple R ← src (n consecutive words) (n = 1 . . . 16)
LDM	dst, R, n	IR DA X	Load Multiple (Store Multiple) dst ← R (n consecutive words) (n = 1 . . . 16)
LDR LDRB LDRL	R, src	RA	Load Relative R ← src (range −32768 . . . +32767)
LDR LDRB LDRL	dst, R	RA	Load Relative (Store Relative) dst ← R (range −32768 . . . +32767)
POP POPL	dst, R	R IR DA X	Pop dst ← IR Autoincrement contents of R
PUSH PUSHL	IR, src	R IM IR DA X	Push Autodecrement contents of R IR ← src

ARITHMETIC

Mnemonics	Operands	Addr. Modes	Operation
ADC ADCB	R, src	R	Add with Carry R ← R + src + carry
ADD ADDB ADDL	R, src	R IM IR DA X	Add R ← R + src
CP CPB CPL	R, src	R IM IR DA X	Compare with Register R − src
CP CPB	dst, IM	IR DA X	Compare with Immediate dst − IM
DAB	dst	R	Decimal Adjust
DEC DECB	dst, n	R IR DA X	Decrement by n dst ← dst − n (n = 1 . . . 16)
DIV DIVL	R, src	R IM IR DA X	Divide (signed) Word: $R_{n+1} \leftarrow R_{n,n+1} \div$ src $R_n \leftarrow$ remainder Long Word: $R_{n+2,n+3} \leftarrow R_{n...n+3} \div$ src $R_{n,n+1} \leftarrow$ remainder
EXTS EXTSB EXTSL	dst	R	Extend Sign Extend sign of low order half of st through high order half of dst
INC INCB	dst, n	R IR DA X	Increment by n dst ← dst + n (n = 1 . . . 16)
MULT MULTL	R, src	R IM IR DA X	Multiply (signed) Word: $R_{n,n+1} \leftarrow R_{n+1} \cdot$ src Long Word: $R_{n...n+3} \leftarrow R_{n+2,n+3} \cdot$ src *Plus seven cycles for each 1 in the multiplicand
NEG NEGB	dst	R IR DA X	Negate dst ← 0 − dst
SBC SBCB	R, src	R	Subtract with Carry R ← R − src − carry
SUB SUBB SUBL	R,src	R IM IR DA X	Subtract R ← R − src

LOGICAL

Mnemonics	Operands	Addr. Modes	Operation
AND ANDB	R, src	R IM IR DA X	AND R ← R AND src
COM COMB	dst	R IM IR DA X	Complement dst ← NOT dst
OR ORB	R, src	R IM IR DA X	OR R ← R OR src
TEST TESTB TESTL	dst	R IR DA X	TEST dst OR 0
TCC TCCB	cc, dst	R	Test Condition Code Set LSB if cc is true
XOR XORB	R, src	R IM IR DA X	Exclusive OR R ← R XOR src

PROGRAM CONTROL

Mnemonics	Operands	Addr. Modes	Operation
CALL	dst	IR DA X	Call Subroutine Autodecrement SP @ SP ← PC PC ← dst
CALR	dst	RA	Call Relative Autodecrement SP @ SP ← PC PC ← PC + dst (range −4094 to +4096)
DJNZ DBJNZ	R, dst	RA	Decrement and Jump if Non-Zero R ← R − 1 IF R = 0: PC ← PC + dst (range −254 to 0)
IRET*	–	–	Interrupt Return PS ← @ SP Autoincrement SP
JP	cc, dst	IR IR DA X	Jump Conditional If cc is true: PC ← dst
JR	cc, dst	RA	Jump Conditional Relative If cc is true: PC ← PC + dst (range −256 to +254)
RET	cc	–	Return Conditional If cc is true: PC ← @ SP Autodecrement SP
SC	src	IM	System Call Autodecrement SP @ SP ← old PS Push instruction PS ← System Call PS

*Privileged instructions. Executed in system mode only.

BIT MANIPULATION

Mnemonics	Operand	Addr. Modes	Operation
BIT BITB	dst, b	R IR DA X	Test Bit Static Z flag ← NOT dst bit specified by b
BIT BITB	dst, R	R	Test Bit Dynamic Z flag ← NOT dst bit specified by contents of R
RES RESB	dst, b	R IR DA X	Reset Bit Static Reset dst bit specified by b
RES RESB	dst, R	R	Reset Bit Dynamic Reset dst bit specified by contents of R
SET SETB	dst, b	R IR DA X	Set Bit Static Set dst bit specified by b
SET SETB	dst, R	R	Set Bit Dynamic Set dst bit specified by contents of R
TSET TSETB	dst	R IR DA X	Test and Set S flag ← MSB of dst dst ← all 1s

ROTATE AND SHIFT

Mnemonics	Operand	Addr. Modes	Operation
RLDB	R, src	R	Rotate Digit Left
RRDB	R, src	R	Rotate Digit Right
RL RLB	dst, n	R R	Rotate Left by n bits (n = 1, 2)
RLC RLCB	dst, n	R R	Rotate Left through Carry by n bits (n = 1, 2)
RR RRB	dst, n	R R	Rotate Right by n bits (n = 1, 2)
RRC RRCB	dst, n	R R	Rotate Right through Carry by n bits (n = 1, 2)
SDA SDAB SDAL	dst, R	R	Shift Dynamic Arithmetic Shift dst left or right by contents of R
SDL SDLB SDLL	dst, R	R	Shift Dynamic Logical Shift dst left or right by contents of R
SLA SLAB SLAL	dst, n	R	Shift Left Arithmetic by n bits
SLL SLLB SLLL	dst, n	R	Shift Left Logical by n bits
SRA SRAB SRAL	dst, n	R	Shift Right Arithmetic by n bits
SRL SRLB SRLL	dst, n	R	Shift Right Logical by n bits

BLOCK TRANSFER AND STRING MANIPULATION

Mnemonics	Operands	Addr. Modes	Operation
CPD CPDB	R_X, src, R_Y, cc	IR	Compare and Decrement R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$
CPDR CPDRB	R_X, src, R_Y, cc	IR	Compare, Decrement and Repeat R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPI CPIB	R_X, src, R_Y, cc	IR	Compare and Increment R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$
CPIR CPIRB	R_X, src, R_Y, cc	IR	Compare, Increment and Repeat R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPSD CPSDB	dst, src, R, cc	IR	Compare String and Decrement dst − src Autodecrement dst and src addresses R ← R − 1
CPSDR CPSDRB	dst, src, R, cc	IR	Compare String, Decr. and Repeat dst − src Autodecrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
CPSI CPSIB	dst, src, R, cc	IR	Compare String and Increment dst − src Autoincrement dst and src addresses R ← R − 1
CPSIR CPSIRB	dst, src, R, cc	IR	Compare String, Incr. and Repeat dst − src Autoincrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
LDD LDDB	dst, src, R	IR	Load and Decrement dst ← src Autodecrement dst and src addresses R ← R − 1
LDDR LDDRB	dst, src, R	IR	Load, Decrement and Repeat dst ← src Autodecrement dst and src addresses R ← R − 1 Repeat until R = 0

BLOCK TRANSFER AND STRING MANIPULATION (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
LDI LDIB	dst, src, R	IR	Load and Increment dst ← src Autoincrement dst and src addresses R ← R − 1
LDIR LDIRB	dst, src, R	IR	Load, Increment and Repeat dst ← src Autoincrement dst and src addresses R ← R − 1 Repeat until R = 0
TRDB	dst, src, R	IR	Translate and Decrement dst ← src (dst) Autodecrement dst address R ← R − 1
TRDRB	dst, src, R	IR	Translate, Decrement and Repeat dst ← src (dst) Autodecrement dst address R ← R − 1 Repeat until R = 0
TRIB	dst, src, R	IR	Translate and Increment dst ← src (dst) Autoincrement dst address R ← R − 1
TRIRB	dst, src, R	IR	Translate, Increment and Repeat dst ← src (dst) Autoincrement dst address R ← R − 1 Repeat until R = 0
TRTDB	src 1, src 2, R	IR	Translate and Test, Decrement RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1
TRTDRB	src 1, src 2, R	IR	Translate and Test, Decrement and Repeat RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0
TRTIB	src 1, src 2, R	IR	Translate and Test, Increment RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1
TRTIRB	src 1, src 2, R	IR	Translate and Test, Increment and Repeat RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0

INPUT/OUTPUT

Mnemonics	Operands	Addr. Modes	Operation
IN* INB*	R, src	IR DA	Input R ← src
IND* INDB*	dst, src, R	IR	Input and Decrement dst ← src Autodecrement dst address R ← R − 1
INDR* INDRB*	dst, src, R	IR	Input, Decrement and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
INI* INIB*	dst, src, R	IR	Input and Increment dst ← src Autoincrement dst address R ← R − 1
INIR* INIRB*	dst, src, R	IR	Input, Increment and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0
OUT* OUTB*	dst, R	IR DA	Output dst ← R
OUTD* OUTDB*	dst, src, R	IR	Output and Decrement dst ← src Autodecrement src address R ← R − 1
OTDR* OTDRB*	dst, src, R	IR	Output, Decrement and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
OUTI* OUTIB*	dst, src, R	IR	Output and Increment dst ← src Autoincrement src address R ← R − 1
OTIR* OTIRB*	dst, src, R	IR	Ouput, Increment and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0
SIN* SINB*	R, src	DA	Special Input R ← src
SIND* SINDB*	dst, src, R	IR	Special Input and Decrement dst ← src Autodecrement dst address R ← R − 1
SINDR* SINDRB*	dst, src, R	IR	Special Input, Decr. and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
SINI* SINIB*	dst, src, R	IR	Special Input and Increment dst ← src Autoincrement dst address R ← R − 1
SINIR* SINIRB*	dst, src, R	IR	Special Input, Incr. and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0

INPUT/OUTPUT (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
SOUT* SOUTB*	dst, src	DA	Special Output dst ← src
SOUTD* SOUTDB*	dst, src, R	IR	Special Output and Decrement dst ← src Autodecrement src address R ← R − 1
SOTDR* SOTDRB*	dst, src, R	IR	Special Output, Decr. and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
SOUTI* SOUTIB*	dst, src, R	IR	Special Output and Increment dst ← src Autoincrement src address R ← R − 1
SOTIR* SOTIRB*	dst, src, R	R	Special Output, Incr. and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0

CPU CONTROL

Mnemonics	Operands	Addr. Modes	Operation
COMFLG	flags	–	Complement Flag (Any combination of C, Z, S, P/V)
DI*	int	–	Disable Interrupt (Any combination of NVI, VI)
EI*	int	–	Enable Interrupt (Any combination of NVI, VI)
HALT*	–	–	HALT
LDCTL*	CTLR, src	R	Load into Control Register CTLR ← src
LDCTL*	dst, CTLR	R	Load from Control Register dst ← CTLR
LDCTLB	FLGR, src	R	Load into Flag Byte Register FLGR ← src
LDCTLB	dst, FLGR	R	Load from Flag Byte Register dst ← FLGR
LDPS*	src	IR DA X	Load Program Status PS ← src
MBIT*	–	–	Test Multi-Micro Bit Set S if $\overline{\mu I}$ is High; reset S if $\overline{\mu I}$ is Low.
MREQ*	dst	R	Multi-Micro Request
MRES*	–	–	Multi-Micro Reset
MSET*	–	–	Multi-Micro Set
NOP	–	–	No Operation
RESFLG	flag	–	Reset Flag (Any combination of C, Z, S, P/V)
SETFLG	flag	–	Set Flag (Any combination of C, Z, S, P/V)

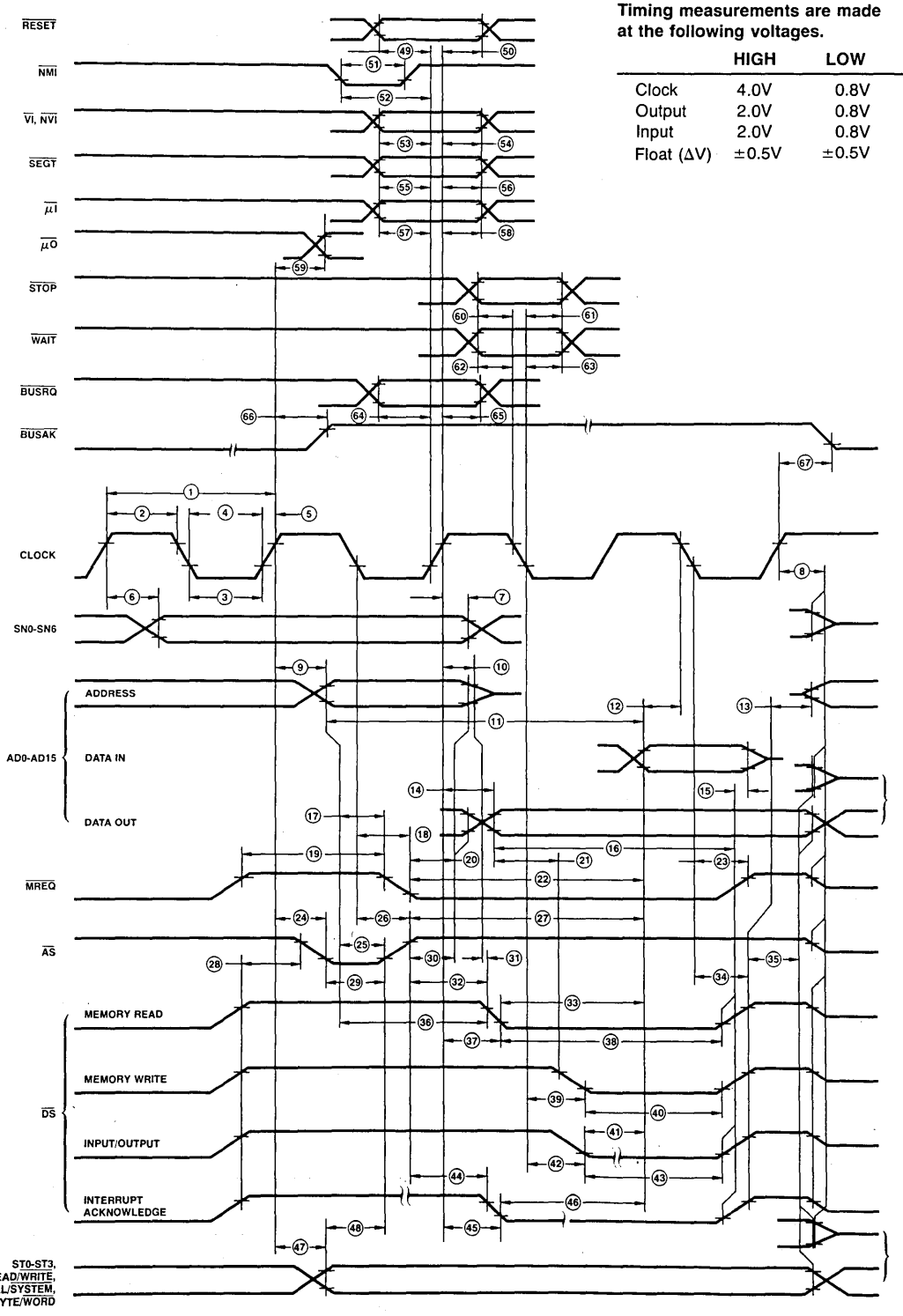
*Privileged instructions. Executed in system mode only.

4

AC TIMING DIAGRAM

Timing measurements are made at the following voltages.

	HIGH	LOW
Clock	4.0V	0.8V
Output	2.0V	0.8V
Input	2.0V	0.8V
Float (ΔV)	±0.5V	±0.5V



This composite timing diagram does not show actual timing sequences. Refer to this diagram only for the detailed timing relationships of individual edges. Use the preceding illustrations as an explanation of the various timing sequences.

MOS-255

SWITCHING CHARACTERISTICS over operating range

AmZ8001DC

Number	Parameter	Description	Min	Max	Units
1	TcC	Clock Cycle Time	250	2000	ns
2	TwCh	Clock Width (High)	105	2000	ns
3	TwCl	Clock Width (Low)	105	2000	ns
4	TfC	Clock Fall Time		20	ns
5	TrC	Clock Rise Time		20	ns
6	TdC(SNv)	Clock ↑ to Segment Number Valid (50pF Load)		130	ns
7	TdC(SNn)	Clock ↑ to Segment Number Not Valid	20		ns
8	TdC(Bz)	Clock ↑ to Bus Float		65	ns
9	TdC(A)	Clock ↑ to Address Valid		100	ns
10	TdC(Az)	Clock ↑ to Address Float		65	ns
11	TdA(DI)	Address Valid to Data In Required Valid	400		ns
12	TsDI(C)	Data In to Clock ↓ Set-up Time	70		ns
13	TdDS(A)	$\overline{DS}$ ↑ to Address Active	80		ns
14	TdC(DO)	Clock ↑ to Data Out Valid		100	ns
15	ThDI(DS)	Data In to $\overline{DS}$ ↑ Hold Time	0		ns
16	TdDO(DS)	Data Out Valid to $\overline{DS}$ ↑ Delay	230		ns
17	TdA(MR)	Address Valid to $\overline{MREQ}$ ↓ Delay	55		ns
18	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↓ Delay		80	ns
19	TwMRh	$\overline{MREQ}$ Width (High)	190		ns
20	TdMR(A)	$\overline{MREQ}$ ↓ to Address Not Active	70		ns
21	TdDO(DSW)	Data Out Valid to DS ↓ (Write) Delay	55		ns
22	TdMR(DI)	$\overline{MREQ}$ ↓ to Data In Required Valid	330		ns
23	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↑ Delay		80	ns
24	TdC(ASf)	Clock ↑ to $\overline{AS}$ ↓ Delay		80	ns
25	TdA(AS)	Address Valid to $\overline{AS}$ ↑ Delay	55		ns
26	TdC(ASr)	Clock ↓ to $\overline{AS}$ ↑ Delay		90	ns
27	TdAS(DI)	$\overline{AS}$ ↑ to Data In Required Valid	290		ns
28	TdDS(AS)	$\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay	70		ns
29	TwAS	$\overline{AS}$ Width (Low)	80		ns
30	TdAS(A)	$\overline{AS}$ ↑ to Address Not Active Delay	60		ns
31	TdAz(DSR)	Address Float to $\overline{DS}$ (Read) ↓ Delay	0		ns
32	TdAS(DSR)	$\overline{AS}$ ↑ to $\overline{DS}$ (Read) ↓ Delay	70		ns
33	TdDSR(DI)	$\overline{DS}$ (Read) ↓ to Data In Required Valid	155		ns
34	TdC(DSr)	Clock ↓ to $\overline{DS}$ ↑ Delay		70	ns
35	TdDS(DO)	$\overline{DS}$ ↑ to Data Out and STATUS Not Valid	80		ns
36	TdA(DSR)	Address Valid to $\overline{DS}$ (Read) ↓ Delay	120		ns
37	TdC(DSR)	Clock ↑ to $\overline{DS}$ (Read) ↓ Delay		120	ns
38	TwDSR	$\overline{DS}$ (Read) Width (Low)	275		ns
39	TdC(DSW)	Clock ↓ to $\overline{DS}$ (Write) ↓ Delay		95	ns
40	TwDSW	$\overline{DS}$ (Write) Width (Low)	160		ns
41	TdDSI(DI)	$\overline{DS}$ (Input) ↓ to Data In Required Valid	315		ns
42	TdC(DSf)	Clock ↓ to $\overline{DS}$ (I/O) ↓ Delay		120	ns
43	TwDS	$\overline{DS}$ (I/O) Width (Low)	400		ns
44	TdAS(DSA)	$\overline{AS}$ ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay	960		ns
45	TdC(DSA)	Clock ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay		120	ns
46	TdDSA(DI)	$\overline{DS}$ (Acknowledge) ↓ to Data In Required Delay	420		ns
47	TdC(S)	Clock ↑ to Status Valid Delay		110	ns
48	TdS(AS)	Status Valid to $\overline{AS}$ ↑ Delay	40		ns

SWITCHING CHARACTERISTICS (Cont.)

AmZ8001DC

Number	Parameter	Description	Min	Max	Units
49	TsR(C)	$\overline{\text{RESET}}$ to Clock ↑ Set-up Time	180		ns
50	ThR(C)	$\overline{\text{RESET}}$ to Clock ↑ Hold Time	0		ns
51	TwNMI	$\overline{\text{NMI}}$ Width (Low)	100		ns
52	TsNMI(C)	$\overline{\text{NMI}}$ to Clock ↑ Set-up Time	140		ns
53	TsVI(C)	$\overline{\text{VI}}$, $\overline{\text{NVI}}$ to Clock ↑ Set-up Time	110		ns
54	ThVI(C)	$\overline{\text{VI}}$, $\overline{\text{NVI}}$ to Clock ↑ Hold Time	0		ns
55	TsSGT(C)	$\overline{\text{SEGT}}$ to Clock ↑ Set-up Time	70		ns
56	ThSGT(C)	$\overline{\text{SEGT}}$ to Clock ↑ Hold Time	0		ns
57	TsμI(C)	μI to Clock ↑ Set-up Time	180		ns
58	ThμI(C)	μI to Clock ↑ Hold Time	0		ns
59	TdC(μO)	Clock ↑ to μO Delay		120	ns
60	TsSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Set-up Time	140		ns
61	ThSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Hold Time	0		ns
62	TsWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Set-up Time	70		ns
63	ThWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Hold Time	0		ns
64	TsBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Set-up Time	90		ns
65	ThBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Hold Time	0		ns
66	TdC(BAKr)	Clock ↑ to $\overline{\text{BUSAK}}$ ↑ Delay		100	ns
67	TdC(BAKf)	Clock ↑ to $\overline{\text{BUSAK}}$ ↓ Delay		100	ns

MAXIMUM RATINGS above which useful life may be impaired.

Voltages on all inputs and outputs with respect to GND	−0.3V to +7.0V
Operating Ambient Temperature	0 to +70°C
Storage Temperature	−65 to +150°C

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

ELECTRICAL CHARACTERISTICS over operating range (Note 1)

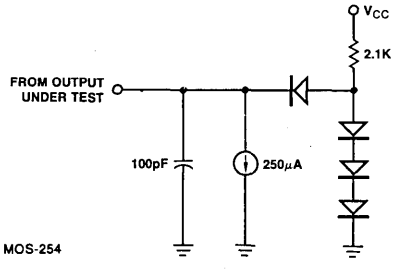
Parameter	Description	Test Conditions	Min	Max	Units
V_{CH}	Clock Input High Voltage	Driven by External Clock Generator	$V_{CC}-0.4$	$V_{CC}+0.3$	Volts
V_{CL}	Clock Input Low Voltage	Driven by External Clock Generator	−0.3	0.45	Volts
V_{IH}	Input High Voltage		2.0	$V_{CC}+0.3$	Volts
V_{IL}	Input Low Voltage		−0.3	0.8	Volts
V_{OH}	Output High Voltage	$I_{OH} = -250\mu A$	2.4		Volts
V_{OL}	Output Low Voltage	$I_{OL} = +2.0mA$		0.4	Volts
I_{IL}	Input Leakage	$0.4 \leq V_{IN} \leq +2.4V$		±10	μA
I_{OL}	Output Leakage	$0.4 \leq V_{OUT} \leq +2.4V$		±10	μA
I_{CC}	V_{CC} Supply Current			300	mA

4

Standard Test Conditions

The characteristics below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND. Positive current flows into the referenced pin. Standard conditions are as follows:

$+4.75V \leq V_{CC} \leq +5.25V$
GND = 0V
$0°C \leq T_A \leq +70°C$



All AC parameters assume a load capacitance of 100pF max, except for parameter 6, TdC(SNv) (50pF max). Timing references between two output signals assume a load difference of 50pF max.

For more information, refer to these AMD publications:

Processor Instruction Set (AM-PUB086).
Describes each instruction in detail. 250 pp.

Processor Interface Manual (AM-PUB089).
Describes hardware interfacing for interrupts and I/O, including the Am9511A Arithmetic Processor, the Am9517A DMA Controller, and the Am9519 Interrupt Controller. 81 pp.

AmZ8002

16-Bit Microprocessor

PRELIMINARY DATA

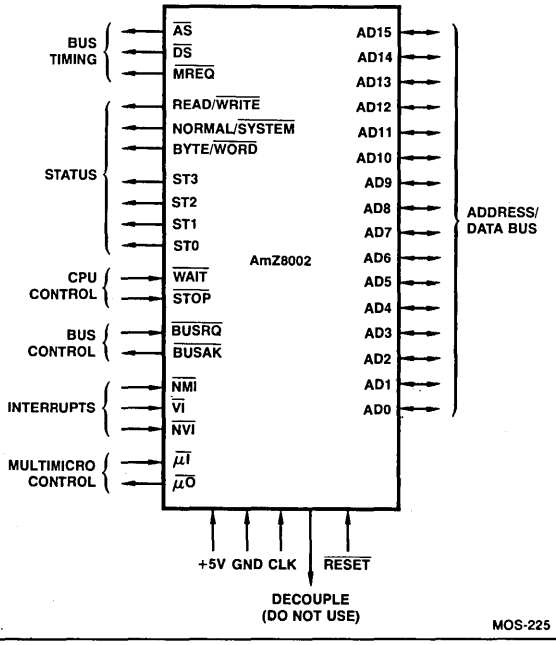
DISTINCTIVE CHARACTERISTICS

- Sixteen general purpose registers
- Direct addressing up to 64KB memory
- Software compatible with AmZ8001 microprocessor
- Powerful instructions with flexible addressing modes
- Privileged/Non-Privileged mode of operation
- Sophisticated interrupt structure
- On-chip memory refresh facility
- TTL compatible inputs and outputs
- Single phase clock
- Single +5V power supply
- 40-pin package

GENERAL DESCRIPTION

The AmZ8002 is a general-purpose 16-bit CPU belonging to the AmZ8000 family of microprocessors. Its architecture is centered around sixteen 16-bit general registers. The CPU deals with 16-bit address spaces and hence can address directly 64 Kilobytes of memory. Facilities are provided to maintain three distinct address spaces – code, data and stack. The AmZ8002 implements a powerful instruction set with flexible addressing modes. These instructions operate on several data types – bit, byte, word (16-bit), long word (32-bit), byte string and word string. The CPU can execute instructions in one of two modes – System and Normal. Sometimes these modes are also known as Privileged and Non-Privileged, respectively. The CPU also contains an on-chip memory refresh facility. The AmZ8002 is software compatible with the AmZ8001 microprocessor. The AmZ8002 is fabricated using silicon-gate N-MOS technology and is packaged in a 40-pin DIP. The AmZ8002 requires a single +5 power supply and a single phase clock for its operation.

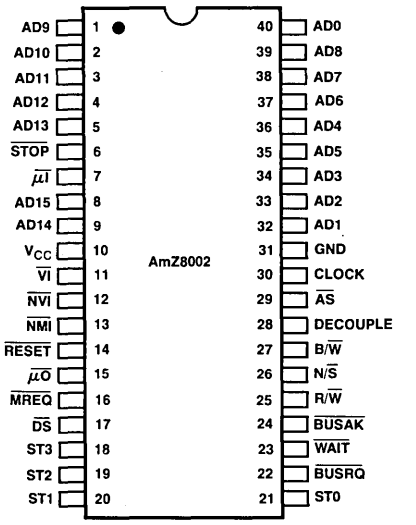
LOGIC SYMBOL



MOS-225

CONNECTION DIAGRAM

Top View



Note: Pin 1 is marked for orientation.

MOS-226

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency
		4MHz
Hermetic DIP	0°C ≤ T_A ≤ 70°C	AmZ8002DC

INTERFACE SIGNAL DESCRIPTION

V_{CC}: +5V Power Supply

V_{SS}: Ground

AD0-AD15: Address/Data Bus (Bidirectional, 3-State)

This 16-bit multiplexed address/data bus is used for all I/O and memory transactions. HIGH on the bus corresponds to 1 and LOW corresponds to 0. AD0 is the least significant bit position with AD15 the most significant. The $\overline{AS}$ output and $\overline{DS}$ output will indicate whether the bus is used for address or data. The status output lines ST0-ST3 will indicate the type of transaction; memory or I/O.

$\overline{AS}$: Address Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus contains address information. The address information is stable by the time of the LOW-to-HIGH transition of the $\overline{AS}$ output (see timing diagrams). The status outputs ST0-ST3 indicate whether the bus contains a memory address or I/O address.

$\overline{DS}$: Data Strobe (Output, 3-State)

LOW on this output indicates that the AD0-AD15 bus is being used for data transfer. The R/$\overline{W}$ output indicates the direction of data transfer – read (or in) means data into the CPU and write (or out) means data from the CPU. During a read operation, data can be gated on to the bus when $\overline{DS}$ goes LOW. A LOW-to-HIGH transition on the $\overline{DS}$ output indicates that the CPU has accepted the data (see timing diagram). During a write operation, LOW on the $\overline{DS}$ output indicates that data is setup on the bus. Data will be removed sometime after the LOW-to-HIGH transition of the $\overline{DS}$ output (see timing diagram).

R/$\overline{W}$: Read/Write (Output, 3-State)

This output indicates the direction of data flow on the AD0-AD15 bus. HIGH indicates a read operation, i.e., data into the CPU and LOW indicates a write operation, i.e., data from the CPU. This output is activated at the same time as $\overline{AS}$ going LOW and remains stable for the duration of the whole transaction (see timing diagram).

B/$\overline{W}$: Byte/Word (Output, 3-State)

This output indicates the type of data transferred on the AD0-AD15 bus. HIGH indicates byte (8-bit) and LOW indicates word (16-bit) transfer. This output is activated at the same stage as $\overline{AS}$ going LOW and remains valid for the duration of the whole transaction (see timing diagram). The address generated by the CPU is always a byte address. However, the memory is organized as 16-bit words. All instructions and word operands are word aligned and are addressed by even addresses. Thus, for all word transactions with the memory the least significant address bit will be zero. When addressing the memory for byte transactions, the least significant address bit determines which byte of the memory word is needed; even address specifies the most significant byte and odd address specifies the least significant byte. In the case of I/O transactions, the address information on the AD0-AD15 bus refers to an I/O port and B/$\overline{W}$ determines whether a data word or data byte will be transacted. During I/O byte transactions, the least significant address bit A0 determines which half of the AD0-AD15 bus will be used for the I/O transactions. The ST0-ST3 outputs will indicate whether the current transaction is for memory, normal I/O or special I/O.

ST0-ST3: Status (Outputs, 3-State)

These four outputs contain information regarding the current transaction in a coded form. The status line codes are shown in the following table:

ST3	ST2	ST1	ST0	
L	L	L	L	Internal Operation
L	L	L	H	Memory Refresh
L	L	H	L	Normal I/O Transaction
L	L	H	H	Special I/O Transaction
L	H	L	L	Reserved
L	H	L	H	Non-Maskable Interrupt Acknowledge
L	H	H	L	Non-Vectored Interrupt Acknowledge
L	H	H	H	Vectored Interrupt Acknowledge
H	L	L	L	Memory Transaction for Operand
H	L	L	H	Memory Transaction for Stack
H	L	H	L	Reserved
H	L	H	H	Reserved
H	H	L	L	Memory Transaction for Instruction Fetch (Subsequent Word)
H	H	L	H	Memory Transaction for Instruction Fetch (First Word)
H	H	H	L	Reserved
H	H	H	H	Reserved

$\overline{WAIT}$: Wait (Input)

LOW on this input indicates to the CPU that memory or I/O is not ready for the data transfer and hence the current transaction should be stretched. The $\overline{WAIT}$ input is sampled by the CPU at certain instances during the transaction (see timing diagram). If $\overline{WAIT}$ input is LOW at these instances, the CPU will go into wait state to prolong the transaction. The wait state will repeat until the $\overline{WAIT}$ input is HIGH at the sampling instant.

N/$\overline{S}$: Normal/System Mode (Output, 3-State)

HIGH on this output indicates that the CPU is operating in Normal Mode and LOW indicates operation in System Mode. This output is derived from the Flag Control Word (FCW) register. The FCW register is described under the program status information section of this document.

$\overline{MREQ}$: Memory Request (Output, 3-State)

LOW on this output indicates that a CPU transaction with memory is taking place.

$\overline{BUSRQ}$: Bus Request (Input)

LOW on this input indicates to the CPU that another device (such as DMA) is requesting to take control of the bus. The $\overline{BUSRQ}$ input can be driven LOW anytime. The CPU synchronizes this input internally. The CPU responds by activating $\overline{BUSAK}$ output LOW to indicate that the bus has been relinquished. Relinquishing the bus means that the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3 and $\overline{MREQ}$ outputs will be in the high impedance state. The requesting device should control these lines in an identical fashion to the CPU to accomplish transactions. The $\overline{BUSRQ}$ input must remain LOW as long as needed to perform all the transactions and the CPU will keep the $\overline{BUSAK}$ output LOW. After completing the transactions, the device must disable the AD0-AD15, $\overline{AS}$, $\overline{DS}$, B/$\overline{W}$, R/$\overline{W}$, N/$\overline{S}$, ST0-ST3 and $\overline{MREQ}$ into the high impedance state and stop driving the $\overline{BUSRQ}$ input LOW. The CPU will make $\overline{BUSAK}$ output HIGH sometime later and take back the bus control.

$\overline{BUSAK}$: Bus Acknowledge (Output)

LOW on this output indicates that the CPU has relinquished the bus in response to a bus request.

$\overline{NMI}$: Non-Maskable Interrupt (Input)

HIGH-to-LOW transition on this input constitutes non-maskable interrupt request. The CPU will respond with the non-maskable Interrupt Acknowledge on the ST0-ST3 outputs and will enter an interrupt sequence. The transition on the $\overline{NMI}$ can occur anytime. Of the three kinds of interrupts available, the non-maskable interrupt has the highest priority.

$\overline{VI}$: Vectored Interrupt (Input)

LOW on this input constitutes vectored interrupt request. Vectored interrupt is next lower to the non-maskable interrupt in priority. The VIE bit in the Flag and Control Word register must be 1 for the vectored interrupt to be honored. The CPU will respond with Vectored Interrupt Acknowledge code on the ST0-ST3 outputs and will begin the interrupt sequence. The $\overline{VI}$ input can be driven LOW any time and should be held LOW until acknowledged.

$\overline{NVI}$: Non-Vectored Interrupt (Input)

LOW on this input constitutes non-vectored interrupt request. Non-vectored has the lowest priority of the three types of interrupts. The NVIE bit in the Flag and Control Word register must be 1 for this request to be honored. The CPU will respond with Non-Vectored Interrupt Acknowledge code on the ST0-ST3 outputs and will begin the interrupt sequence. The $\overline{NVI}$ input can be driven LOW anytime and should be held LOW until acknowledged.

$\overline{\mu I}$: Micro-In (Input)

This input participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{\mu O}$: Micro-Out (Output)

This output participates in the resource request daisy chain. See the section on multimicroprocessor support facilities in this document.

$\overline{RESET}$: Reset (Input)

LOW on this input initiates a reset sequence in the CPU. See the section on Initialization for details on reset sequence.

CLK: Clock (Input)

All CPU operations are controlled from the signal fed into this input. See DC Characteristics for clock voltage level requirements.

DECOUPLE: (Output)

Output from the on-chip substrate bias generator. Do not use.

$\overline{STOP}$: Stop (Input)

This active LOW input facilitates one instruction at a time operation. See the section on single stepping.

PROCESSOR ORGANIZATION

The following is a brief discussion of the AmZ8002 CPU. For detailed information, see the AmZ8001/AmZ8002 Processor Instruction Set Manual (Publication No. AM-PUB086).

General Purpose Registers

The CPU is organized around sixteen 16-bit general purpose registers R0 through R15 as shown in Figure 1. For byte operations, the first eight registers (R0 through R7) can also be addressed as sixteen 8-bit registers designated as RL0, RH0 and so on to RL7 and RH7. The sixteen registers can also be grouped in pairs RR0, RR2 and so on to RR14 to form eight long word (32-bit) registers. Similarly, the sixteen registers can be grouped in quadruples RQ0, RQ4, RQ8 and RQ12 to form four 64-bit registers.

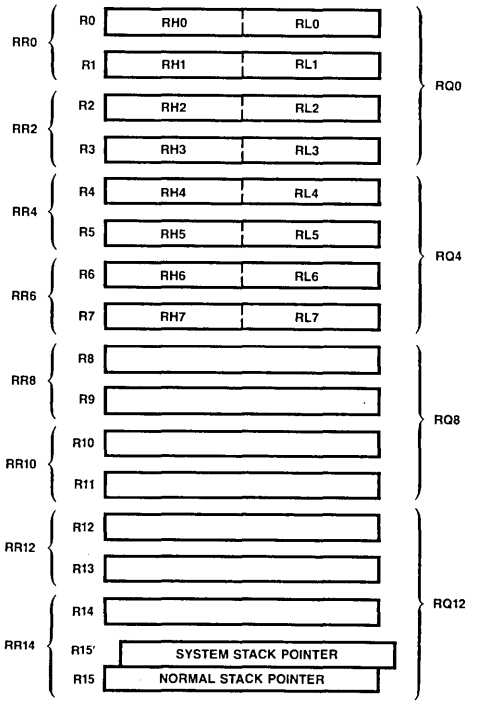


Figure 1. AmZ8002 General Registers.

MOS-227

STACK POINTER

The AmZ8002 architecture allows stacks to be maintained in the memory. Any general purpose register except R0 can be used as a stack pointer in stack manipulating instructions such as PUSH and POP. However, certain instructions such as subroutine call and return make implicit use of the register R15 as the stack pointer. Two implicit stacks are maintained – normal stack using R15 as the stack pointer and system stack using R15' as the system stack pointer (see Figure 1). If the CPU is operating in the Normal Mode, R15 is active, and if the CPU is in System Mode R15' will be used instead of R15. The implied stack pointer is a part of the general registers and hence can be manipulated using the instructions available for register operations.

PROCESSOR STATUS

The CPU status consists of the 16-bit Program Counter (PC) and the 16-bit Flag and Control Word (FCW) register (see Figure 2). The following is a brief description of the FCW bits.

S/$\overline{N}$: System/Normal – 1 indicates System Mode and 0 indicates Normal Mode.

VIE: Vectored Interrupt Enable – 1 indicates that Vectored Interrupt requests will be honored.

NVIE: Non-Vectored Interrupt Enable – 1 indicates that non-vectored interrupt requests will be honored.

C: Carry – 1 indicates that a carry has occurred from the most significant bit position when performing arithmetic operations.

Z: Zero – 1 indicates that the result of an operation is zero.

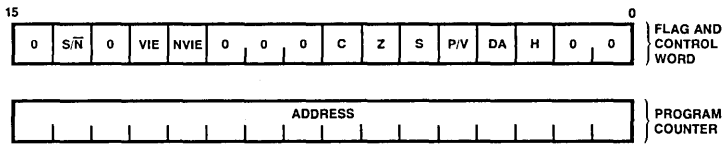


Figure 2. AmZ8002 Processor Status.

MOS-228

S: Sign – 1 indicates that the result of an operation is negative i.e., most significant bit is one.

P/V: Parity/Overflow – 1 indicates that there was an overflow during arithmetic operations. For logical operations this bit indicates parity of the result.

DA: Decimal Adjust – Records byte arithmetic operations.

H: Half Carry – 1 indicates that there was a carry from the most significant bit of the lower digit during byte arithmetic.

DATA TYPES

The AmZ8002 instructions operate on bits, digits (4 bits), bytes (8 bits), words (16 bits), long words (32 bits), byte strings and word strings type operands. Bits can be set, reset or tested. Digits are used to facilitate BCD arithmetic operations. Bytes are used for characters and small integers. Words are used for integer values and addresses while long words are used for large integer values. All operands except strings can reside either in memory or general registers. Strings can reside in memory only.

INTERRUPT AND TRAP STRUCTURE

Interrupt is defined as an external asynchronous event requiring program interruption. For example, interruption is caused by a peripheral needing service. Traps are synchronous events resulting from execution of certain instructions under some defined circumstances. Both interrupts and traps are handled in a similar manner by the AmZ8002.

The AmZ8002 supports three types of interrupts in order of descending priority – non-maskable, vectored and non-vectored. The vectored and non-vectored interrupts can be disabled by appropriate control bits in the FCW. The AmZ8002 has three traps – system call, unimplemented opcode and privileged instruction. The traps have higher priority than interrupts.

When an interrupt or trap occurs, the current program status is automatically pushed on to the system stack. The program status consists of processor status (i.e., PC and FCW) plus a 16-bit identifier. The identifier contains the reason, source and other coded information relating to the interrupt or trap.

After saving the current program status, the new processor status is automatically loaded from the new program status area located in the memory. This area is designated by the New Program Status Area Pointer (NPSAP) register. See AM-PUB086 publication for further details.

ADDRESSING MODES

Information contained in the AmZ8002 instructions consists of the operation to be performed, the operand type and the location of the operands. Operand locations are designated by general register addresses, memory addresses or I/O addresses. The addressing mode of a given instruction defines the address space referenced and the method to compute the operand address. Addressing modes are explicitly specified or implied in an instruction. Figure 3 illustrates the eight explicit addressing modes: Register (R), Immediate (IM), Indirect Register (IR), Direct Address (DA), Indexed (X), Relative Address (RA), Base Address (BA) and Base Indexed (BX).

INPUT/OUTPUT

A set of I/O instructions are provided to accomplish byte or word transfers between the AmZ8002 and I/O devices. I/O devices are addressed using 16-bit I/O port addresses and I/O address space is not a part of the memory address space. Two types of I/O instructions are provided; each with its own 16-bit address space. I/O instructions include a comprehensive set of In, Out and Block transfers.

CPU TIMING

The AmZ8002 accomplishes instruction execution by stepping through a pre-determined sequence of machine cycles, such as memory read, memory write, etc. Each machine cycle requires between three and ten clock cycles. Bus Requests by DMA devices are granted at machine cycle boundaries. No machine cycle is longer than ten clock cycles; thus assuring fast response to a Bus Request (assuming no extra wait states). The start of a machine cycle is always marked by a LOW pulse on the $\overline{AS}$ output. The status output lines ST0-ST3 indicate the nature of the current cycle in a coded form.

STATUS LINE CODES

Status line coding was listed in the table shown under ST0-ST3 outputs in the Interface Signal Description. The following is a detailed description of the status codes.

Internal Operation:

This status code indicates that the AmZ8002 is going through a machine cycle for its internal operation. Figure 4 depicts an internal operation cycle. It consists of three clock periods identified as T1, T2 and T3. The $\overline{AS}$ output will be activated with a LOW pulse by the AmZ8002 to mark the start of a machine cycle. The ST0-ST3 will reflect the code for the internal operation. The $\overline{MREQ}$, $\overline{DS}$ and R/$\overline{W}$ outputs will be HIGH. The N/$\overline{S}$ output will remain at the same level as in the previous machine cycle. The AmZ8002 will ignore the $\overline{WAIT}$ input during the internal operation cycle. The CPU will drive the AD0-AD15 bus with unspecified information during T1. However, the bus will go into high impedance during T2 and remain in that state for the remainder of the cycle. The B/$\overline{W}$ output is also activated by the CPU with unspecified information.

Memory Refresh:

This status code indicates that AmZ8002 is accessing the memory to refresh. The refresh cycle consists of three clock periods as depicted in Figure 5. The CPU will activate the $\overline{AS}$ output with a LOW pulse to mark the beginning of a machine cycle and ST0-ST3 outputs will reflect the refresh cycle code. The least significant 9 lines of the AD0-AD15 bus contain the refresh address. Because the memory is word organized, the AD0 will always be LOW. The most significant 7 bus lines are not specified. The $\overline{DS}$ output will remain HIGH for the entire cycle while R/$\overline{W}$, B/$\overline{W}$ and N/$\overline{S}$ outputs will remain at the same level as in the machine cycle prior to refresh. The AD0-AD15 bus will go into high impedance state during T2 period and remain there for

Mode	Operand Addressing			Operand Value
	In the Instruction	In a Register	In Memory	
Register		REGISTER ADDRESS → OPERAND		The content of the register.
Immediate	OPERAND			In the instruction
Indirect Register		REGISTER ADDRESS → ADDRESS	→ OPERAND	The content of the location whose address is in the register.
Direct Address			ADDRESS → OPERAND	The content of the location whose address is in the instruction.
Index		REGISTER ADDRESS, BASE ADDRESS; DISPLACEMENT	→ OPERAND	The content of the location whose address is the address in the instruction, offset by the content of the working register.
Relative Address		DISPLACEMENT; PC VALUE	→ OPERAND	The content of the location whose address is the content of the program counter, offset by the displacement in the instruction.
Base Address		REGISTER ADDRESS, DISPLACEMENT; BASE ADDRESS	→ OPERAND	The content of the location whose address is the address in the register, offset by the displacement in the instruction.
Base Index		REGISTER ADDRESS, REGISTER ADDRESS; BASE ADDRESS, DISPLACEMENT	→ OPERAND	The content of the location whose address is the address in the register, offset by the displacement in the register.

Figure 3. Addressing Modes.

MOS-229

the remainder of the cycle. The AmZ8002 will activate the $\overline{MREQ}$ output LOW during the refresh cycle. It should be noted that $\overline{WAIT}$ input is ignored by the CPU for refresh operations.

I/O Transactions:

There are two status line codes used for I/O transaction cycles. The AmZ8002 provides two separate I/O spaces and two types of instructions called Normal I/O and Special I/O. Each I/O space is addressed by a 16-bit address called port address. The timing for both types of I/O transactions is essentially identical. A typical I/O cycle consists of four clock periods T1, T2, TWA and T3 as shown in Figure 6. The TWA is the wait state; insertion of one wait state for an I/O cycle is always automatic. Additional wait cycles can be inserted by LOW on the $\overline{WAIT}$ input. The $\overline{WAIT}$ input is sampled during every TW state. If this input is LOW, one more wait state will be inserted. Insertion of wait states continues until $\overline{WAIT}$ input is HIGH. T3 state will follow the last wait state to complete the I/O cycle.

During I/O cycles the ST0-ST3 outputs will reflect appropriate code depending on the type of instruction being executed (Normal I/O or Special I/O). $\overline{AS}$ output will be pulsed LOW to mark the beginning of the cycle. The CPU drives the AD0-AD15 bus with the 16-bit port address specified by the current instruction. The N/$\overline{S}$ output will be LOW indicating that CPU is operating in the system mode. It should be recalled that the N/$\overline{S}$ output is derived from the appropriate bit in the FCW register. All I/O instructions are privileged instructions and will be allowed to

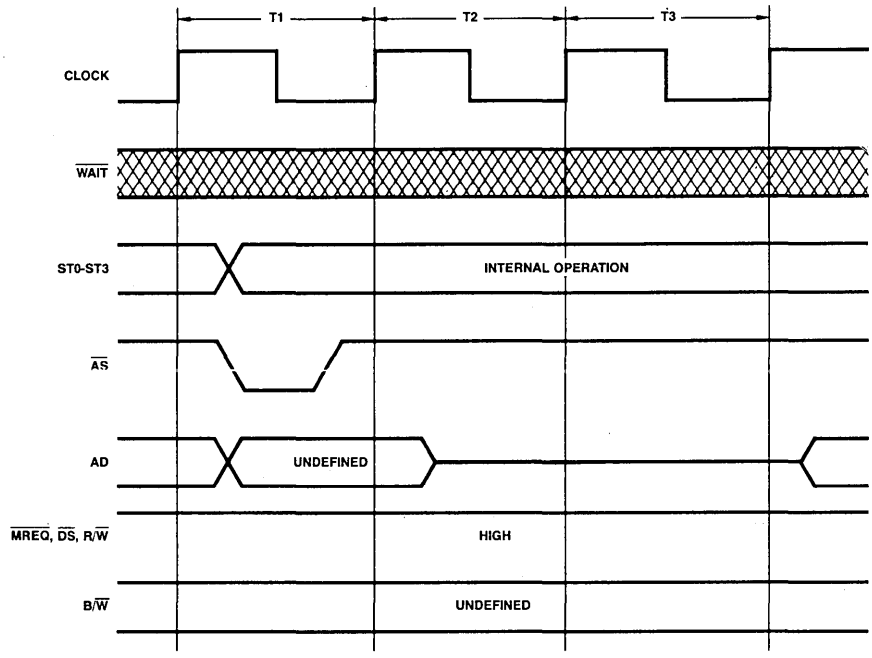


Figure 4. Internal Operation Cycle.

MOS-230

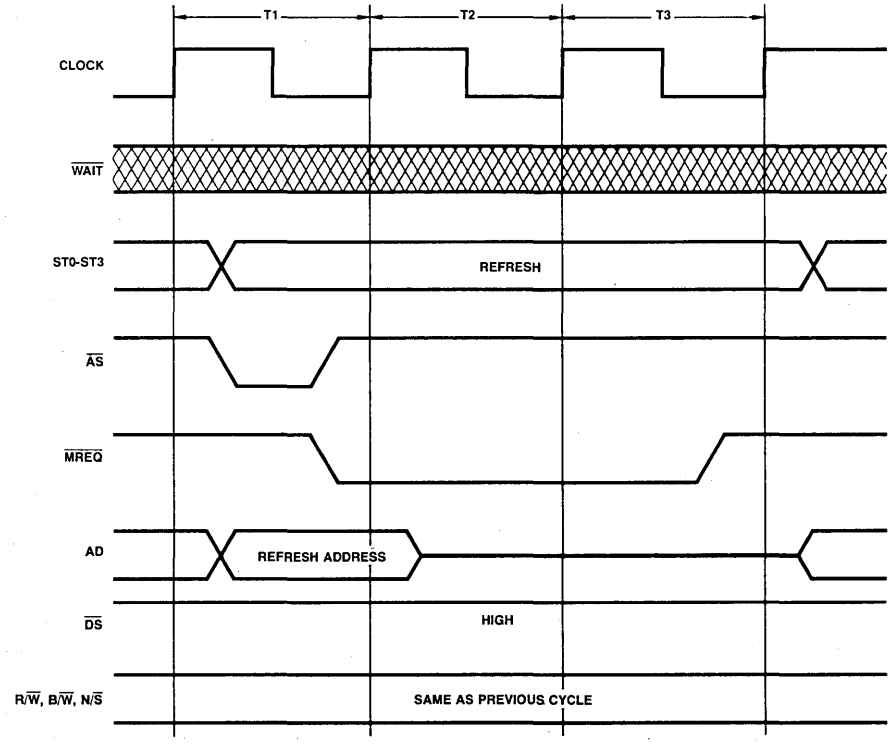


Figure 5. Refresh Cycle.

MOS-231

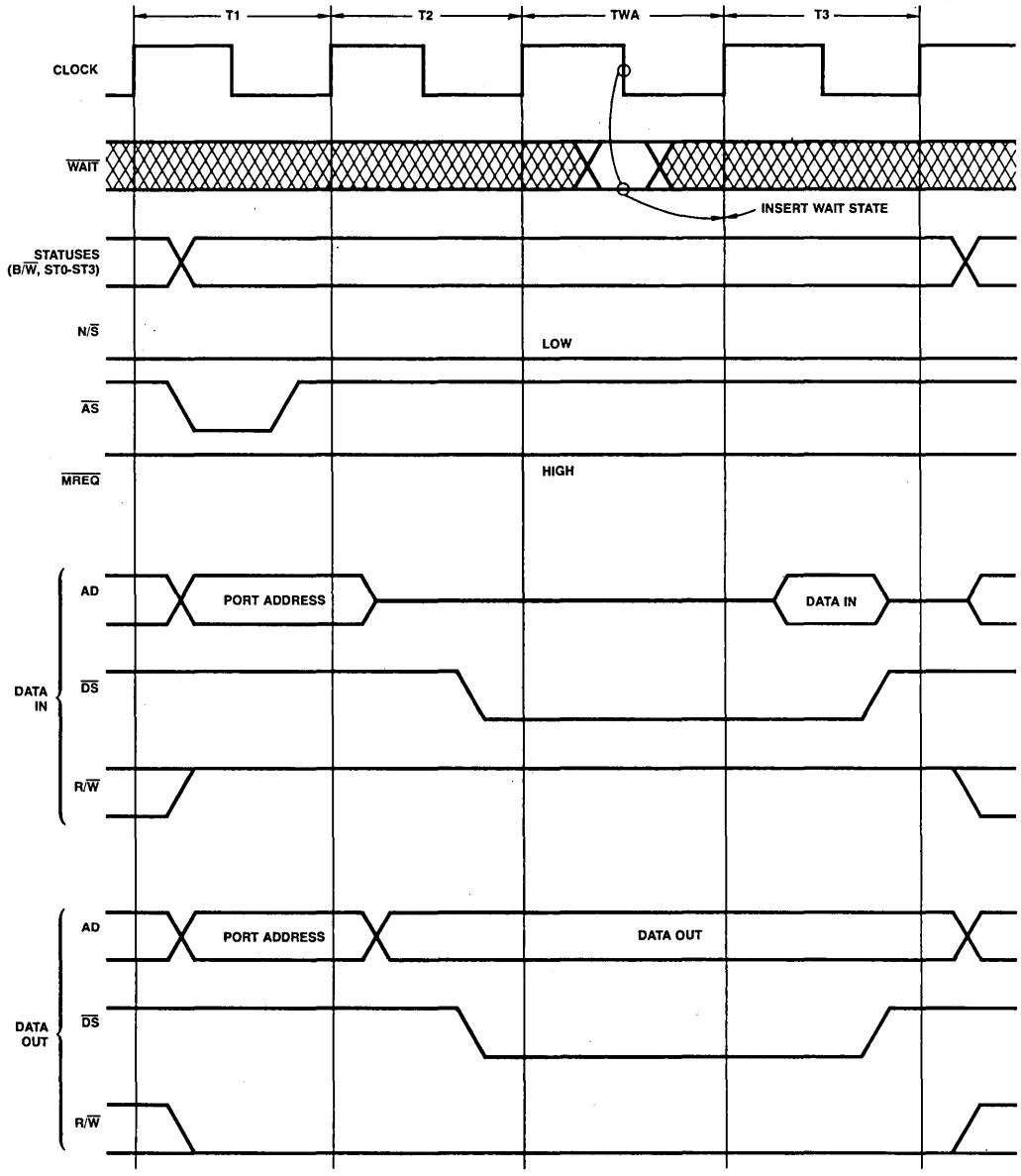


Figure 6. AmZ8002 I/O Cycle.

MOS-232

execute only if the FCW specifies system mode operation. The MREQ output will be HIGH. The AmZ8002 I/O instructions provide both word or byte transactions. The B/W output will be HIGH or LOW depending whether the instruction specifies a byte or word transfer.

Two kinds of I/O transfers should be considered: Data In means reading from the device and Data Out means writing into the device. For In operations, the R/W output will be HIGH. The AD0-AD15 bus will go into high impedance state during T2. During byte input instructions, the CPU reads either the even or odd half of the Data Bus, dependent upon the port address. If the port address is even, the most significant half of the Data Bus is read. If the port address is odd, the least significant half of the Data Bus is read. During word input instructions, the CPU reads all 16 bits of the Data Bus. The AmZ8002 will drive the DS output LOW to signal to the device that data can be gated on to the bus. The CPU will accept the data during T3 and DS output will go HIGH signalling the end of an I/O transaction.

For Data Out, the R/W output will be LOW. The AmZ8002 will provide data on the AD0-AD15 bus and activates the DS output LOW during T2. During byte output instructions, the CPU duplicates the byte data onto both the high and low halves of the Data Bus and external logic, using A0, enables the appropriate byte port. During word output instructions the CPU outputs data onto all 16 bits of the Data Bus. The DS output goes HIGH during T3 and the cycle is complete.

Memory Transactions:

There are four status line codes that indicate a memory transaction:

a) Memory transaction to read or write an operand
b) Memory transaction to read from or write into the stack
c) Memory transaction to fetch the first word of an instruction (sometimes called IF1)
d) Memory transaction to fetch the subsequent word of an instruction (sometimes called IFN).

It can be appreciated that all the above transactions essentially fall into two categories: memory read and memory write. In the case of IF1 and IFN cycles, the memory will be read at the address supplied by the program counter. All AmZ8002 instructions are multiples of 16-bit words. Words are always addressed by an even address. Thus IF1 and IFN cycles involve performing a memory read for words. On the other hand, a memory transaction for operand and stack operation could be a read or write. Moreover, an operand could be a word or a byte. For stack operation involving the implied stack pointer the address will be supplied by the R15 (or R15'). For operand transactions, the memory address will come from several sources depending on the instruction and the addressing mode. Memory transaction cycle timing is shown in Figure 7. It typically consists of three clock periods T1, T2 and T3. Wait states (TW) can be inserted between T2 and T3 by activating the $\overline{WAIT}$ input LOW. The $\overline{WAIT}$ input will be sampled during T2 and during every subsequent TW. The ST0-ST3 outputs will reflect the appropriate code for the current cycle early in T1 and the $\overline{AS}$ output will be pulsed LOW to mark the beginning of the cycle. The N/$\overline{S}$ output will indicate whether the normal or system address space will be used for the current cycle. As shown in the figure the $\overline{MREQ}$ output will go LOW during T1 to indicate a memory operation.

Consider a read operation first. The R/$\overline{W}$ output will be HIGH. The AmZ8002 will drive the AD0-AD15 with the appropriate address early in T1. During T2, the bus will go into high impedance state and $\overline{DS}$ output will be activated LOW by the CPU. The data can be gated on to the bus when $\overline{DS}$ is LOW. During T1 the B/$\overline{W}$ will also be activated to indicate byte or word will be transacted. The AmZ8002 memory is word organized and words are addressed by even addresses. However, when addressing bytes, the memory address may be odd or even; an even address for

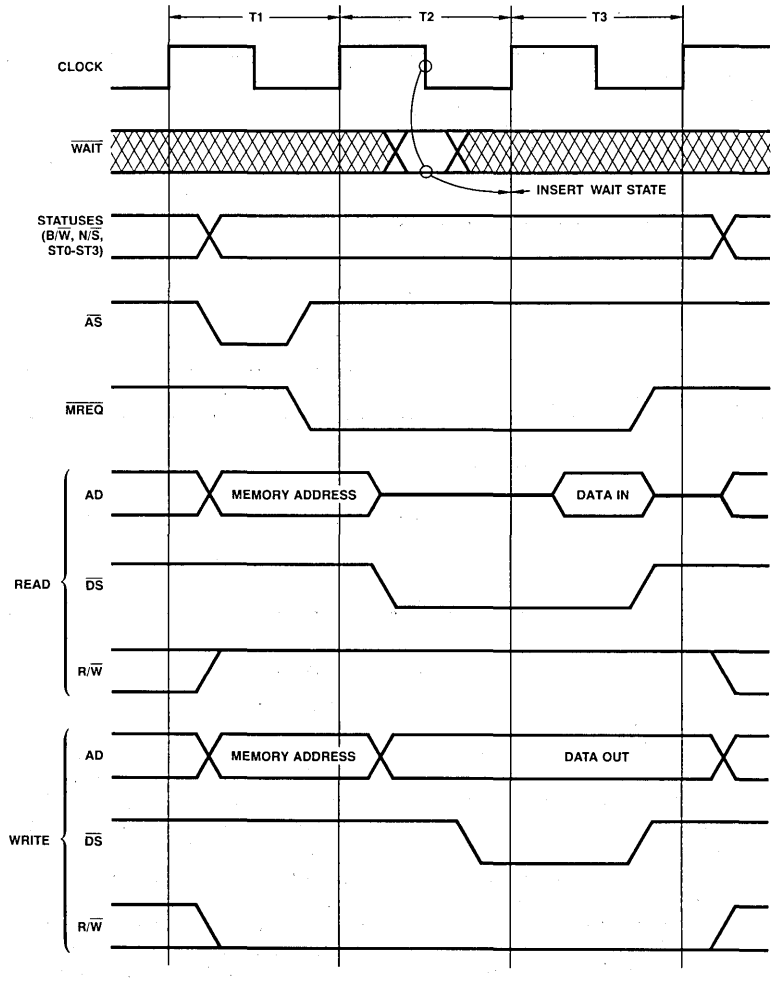


Figure 7. Memory Transactions.

MOS-233

most significant byte of a word and the next odd address for the least significant byte of that word. When reading a byte from the memory, the least significant address bit can be ignored and the whole word containing the desired byte is gated on to the bus. The CPU will pick the appropriate byte automatically. The AmZ8002 will drive the $\overline{DS}$ output HIGH indicating data acceptance.

Consider the write operation next. The R/$\overline{W}$ output will be LOW. The AmZ8002 removes the address and gates out the data to be written on the bus and activates the $\overline{DS}$ output LOW during T2. If the data to be written is a byte then the same byte will be on both halves of the bus. The $\overline{DS}$ output will go HIGH during T3 signifying completion of the cycle.

Interrupt Acknowledge:

There are three status line codes devoted to interrupt acknowledgement. These correspond to non-maskable, vectored and non-vectored interrupts. The Interrupt Acknowledge cycle is illustrated in Figure 8. The $\overline{NMI}$ input of the AmZ8002 is edge detected i.e., a HIGH to LOW input level change is stored in an internal latch. Similar internal storage is not provided for the $\overline{VI}$ and $\overline{NVI}$ inputs. For $\overline{VI}$ and $\overline{NVI}$ inputs to cause an interruption, the corresponding interrupt enable bits in the FCW must be 1. For the following discussion, both the VIE and NVIE bits in the FCW are assumed to be 1.

As shown in the figure, the $\overline{VI}$ input, $\overline{NVI}$ input and the internal $\overline{NMI}$ latch output are sampled during T3 of the last machine cycle of an instruction.

A LOW on these signals triggers the corresponding interrupt acknowledge sequence described below. The AmZ8002 executes a dummy IF1 cycle prior to entering the actual acknowledge cycle (see memory transactions for IF1 cycle description). During this dummy IF1 cycle, the program counter is not updated; instead the implied system stack pointer (R15') will be decremented. Following the dummy IF1 cycle is the actual interrupt acknowledge cycle.

The interrupt acknowledge cycle typically consists of 10 clock periods; T1 through T5 and five automatic TW (wait) states. As usual, the $\overline{AS}$ output will be pulsed LOW during T1 to mark the beginning of a cycle. The ST0-ST3 outputs will reflect the appropriate interrupt acknowledge code, the $\overline{MREQ}$ output will be HIGH, the N/$\overline{S}$ output remains the same as in the preceding cycle, the R/$\overline{W}$ output will be HIGH and the B/W output will be LOW. The AmZ8002 will drive the AD0-AD15 bus with unspecified information during T1 and the bus will go into the high impedance state during T2. Three TWA states will automatically follow T2. The WAIT input will be sampled during the third TWA state.

If LOW, an extra TW state will be inserted and the $\overline{WAIT}$ will be sampled again during TW. Such insertion of TW states continues until the $\overline{WAIT}$ input is HIGH. After the last TW state, the $\overline{DS}$ output will go LOW and two more automatic wait states follow. The interrupting device can gate up to a 16-bit identifier on to the bus when the $\overline{DS}$ output is LOW. The $\overline{WAIT}$ input will be sampled again during the last TWA state. If the $\overline{WAIT}$ input is LOW one TW state will be inserted and the $\overline{WAIT}$ will be sampled during TW. Such TW insertion continues until the $\overline{WAIT}$

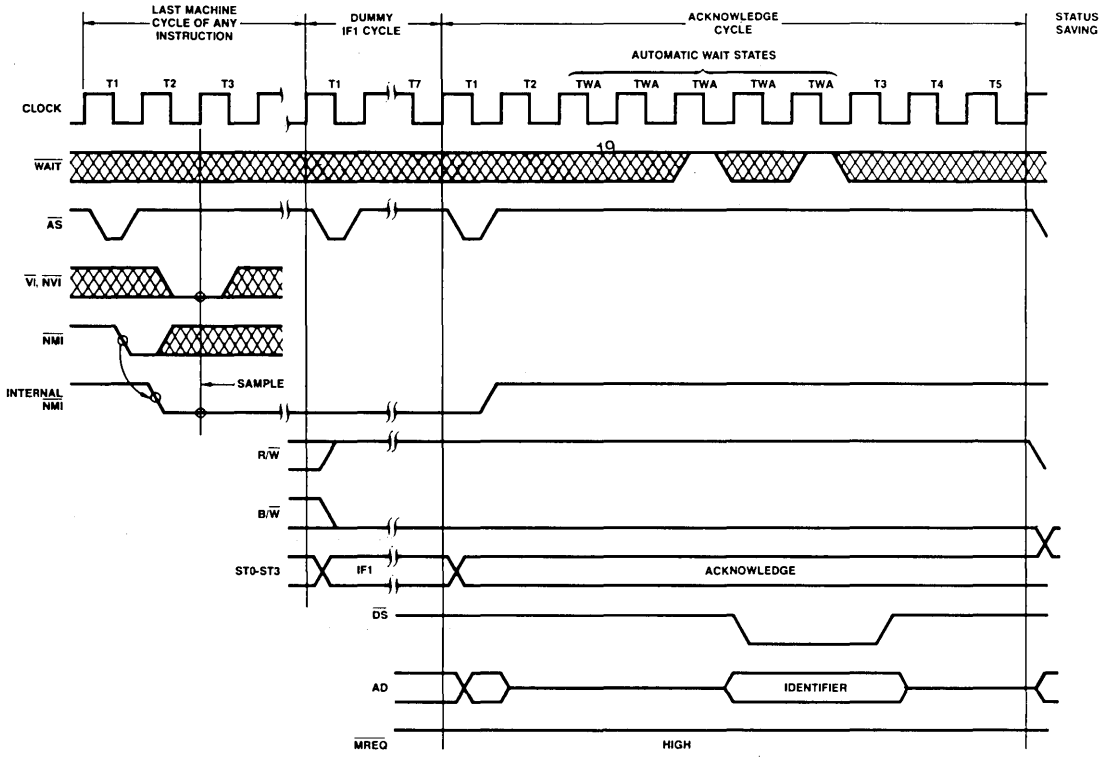


Figure 8. Interrupt Acknowledge Cycle.

MOS-234

input is HIGH. After completing the last TW state T3 will be entered and the $\overline{DS}$ output will go HIGH. The interrupting device should remove the identifier and cease driving the bus. T4 and T5 states will follow T3 to complete the cycle. Following the interrupt acknowledge cycle will be memory transaction cycles to save the status on the stack. Note that the N/$\overline{S}$ output will be automatically LOW during status saving.

The internal $\overline{NMI}$ latch will be reset to the initial state at $\overline{AS}$ going HIGH in the interrupt acknowledge cycle. The $\overline{VI}$ and $\overline{NVI}$ inputs should be kept LOW until this time also.

Status Saving Sequence:

The machine cycles following the interrupt acknowledge cycle push the old status information on the system stack in the following order: the 16-bit program counter; the flag and control word; and finally the interrupt/trap indentifier. Subsequent machine cycles fetch the new program status from the new program status area, and then branch to the interrupt/service routine.

BUS REQUEST/BUS ACKNOWLEDGE TIMING:

A LOW on the $\overline{BUSRQ}$ input is an indication to the AmZ8002 that another device (such as DMA) is requesting control of the bus. The $\overline{BUSRQ}$ input is synchronized internally at T1 of any machine cycle. (See below for exception.) The $\overline{BUSAK}$ will go LOW after the last clock period of the machine cycle. The LOW on the $\overline{BUSAK}$ output indicates acknowledgement. When $\overline{BUSAK}$ is LOW the following outputs will go into the high impedance state; AD0-AD15, $\overline{AS}$, $\overline{DS}$, $\overline{MREQ}$, ST0-ST3, B/$\overline{W}$, R/$\overline{W}$ and N/$\overline{S}$. The $\overline{BUSRQ}$ must be held LOW until all transactions are completed. When $\overline{BUSRQ}$ goes HIGH, it is synchronized internally, the $\overline{BUSAK}$ output will go HIGH and normal CPU operation will resume. Figure 9 illustrates the $\overline{BUSRQ}$/$\overline{BUSAK}$ timing.

It was mentioned that $\overline{BUSRQ}$ will be honored during any machine cycle with one exception. This exception is during the execution of TSET/TSETB instructions. $\overline{BUSRQ}$ will not be honored once execution of these instructions has started.

SINGLE STEPPING

The $\overline{STOP}$ input of the AmZ8002 facilitates one instruction at a time or single step operation. Figure 10 illustrates $\overline{STOP}$ input timing. The $\overline{STOP}$ input is sampled on the HIGH to LOW transition of the clock input that immediately precedes an IF1 cycle. If the $\overline{STOP}$ is found LOW, AmZ8002 introduces a memory refresh cycle after T3. Moreover, $\overline{STOP}$ input will be sampled again at T3. If $\overline{STOP}$ is LOW one more refresh cycle will follow the previous refresh cycle. The $\overline{STOP}$ will be sampled during T3 of the refresh cycle also. One additional refresh cycle will be added every time $\overline{STOP}$ input is sampled LOW. After completing the last refresh cycle which will occur after $\overline{STOP}$ is HIGH, the CPU will insert two dummy states T4 and T5 to complete the IF1 cycle and resume its normal operations for executing the instruction. See appropriate sections on memory transactions and memory refresh.

It should be noted that refresh cycles will occur in the stop mode even if the refresh facility is disabled in the refresh register.

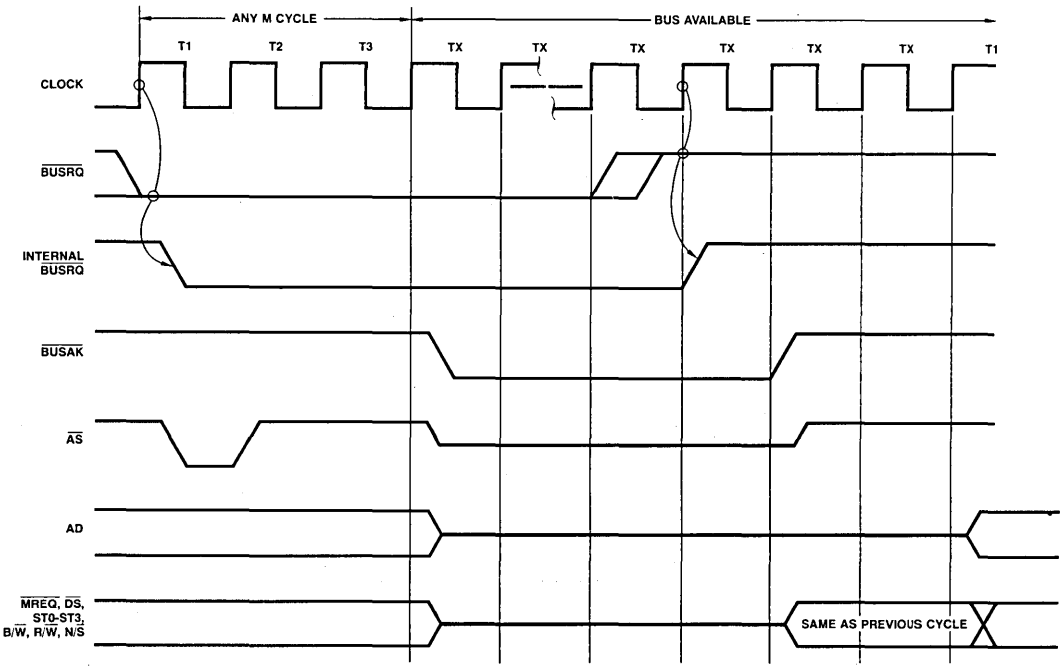


Figure 9. Bus Request/Acknowledge Cycle.

MOS-235

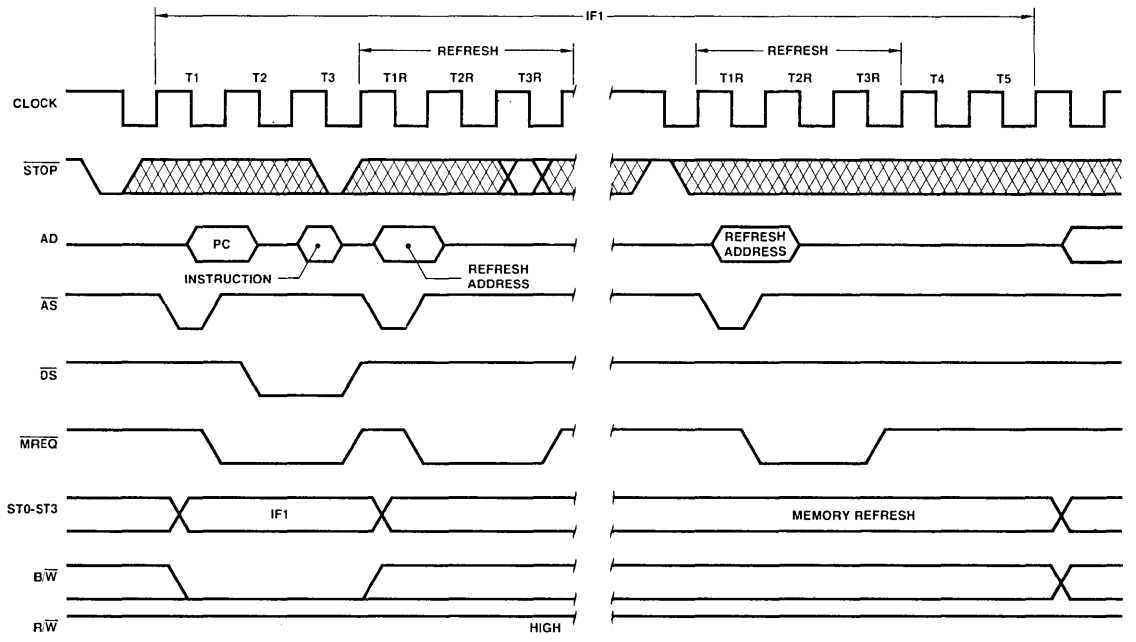


Figure 10. Single Step Timing.

MOS-236

MULTIMICROPROCESSOR FACILITIES

The AmZ8002 is provided with hardware and software facilities to support multiple microprocessor systems. The $\overline{\mu O}$ and $\overline{\mu I}$ signals of the AmZ8002 are used in conjunction with the MBIT, MREQ, MRES and MSET instructions for this purpose. The $\overline{\mu O}$ output can be activated LOW by using an appropriate instruction to signal a request from the AmZ8002 for a resource. The $\overline{\mu I}$ input is tested by the AmZ8002 before activating the $\overline{\mu O}$ output. LOW at the $\overline{\mu I}$ input at this time indicates that the resource is busy. The AmZ8002 can examine the $\overline{\mu I}$ input after activating the $\overline{\mu O}$ output LOW. The $\overline{\mu I}$ will be tested again to see if the requested resource became available. For detailed information on the Multimicroprocessor facilities the AmZ8001/AmZ8002 Processor Interface Manual (Publication No. AM-PUB089) should be consulted.

INITIALIZATION

A LOW on the $\overline{\text{Reset}}$ input starts the CPU initialization. The initialization sequence is shown in Figure 11. Within five clock periods after the HIGH to LOW level change of the $\overline{\text{Reset}}$ input the following will occur:

a) AD0-AD15 bus will be in the HIGH impedance state
b) $\overline{\text{AS}}$, $\overline{\text{DS}}$, $\overline{\text{MREQ}}$, $\overline{\text{BUSAK}}$ and $\overline{\mu O}$ outputs will be HIGH
c) ST0-ST3 outputs will be LOW
d) Refresh will be disabled
e) R/$\overline{\text{W}}$, B/$\overline{\text{W}}$ and N/$\overline{\text{S}}$ outputs are not affected. For a power on reset the state of these outputs is not specified.

After the $\overline{\text{Reset}}$ input returns HIGH and remains HIGH for three clock periods, two 16-bit memory read operations will be performed as follows. Note that the N/$\overline{\text{S}}$ output will be LOW and ST0-ST3 outputs will reflect IFN code.

a) The contents of the memory location 0002 will be read. This information will be loaded into the FCW of the AmZ8002.
b) The contents of the memory location 0004 will be read. This information will be loaded into the AmZ8002 program counter.

This completes initialization sequence and an IF1 cycle will follow to fetch the first instruction to begin program execution. See the section on memory transactions for timing.

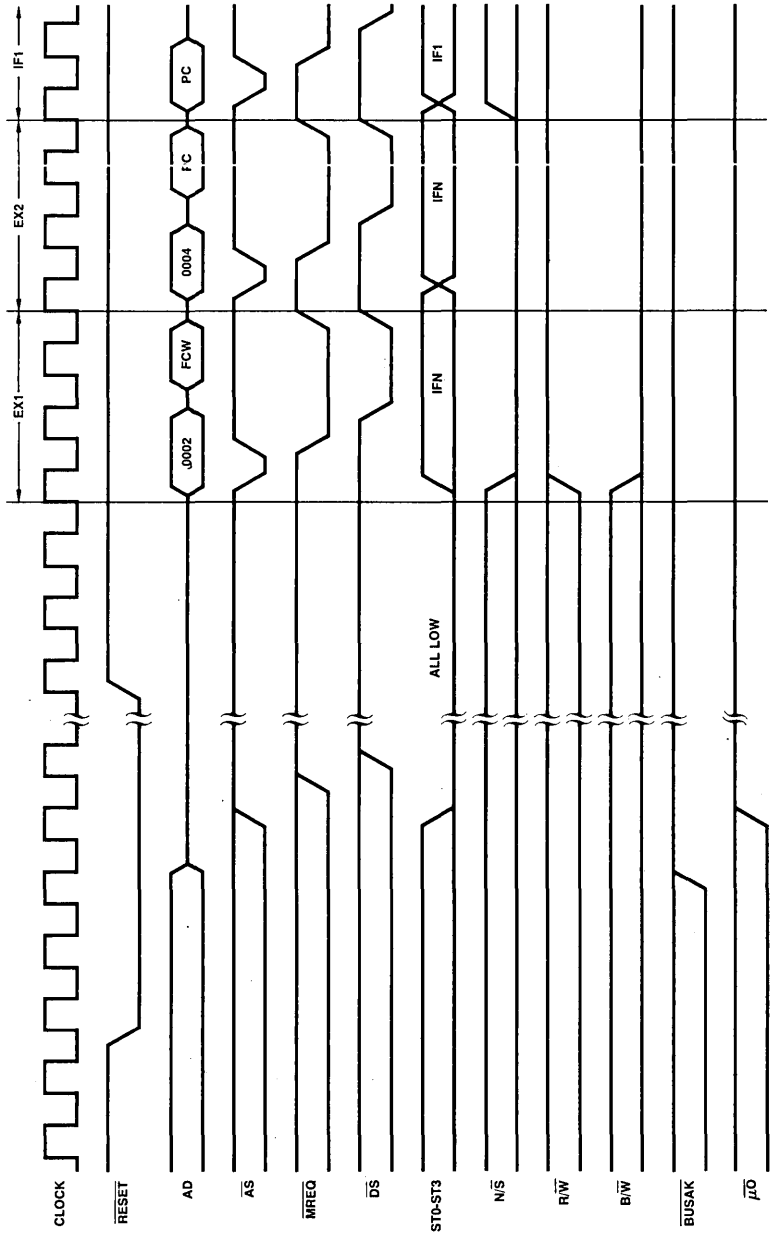


Figure 11. Reset Sequence.

AmZ8002 INSTRUCTION SET

LOAD AND EXCHANGE

Mnemonics	Operands	Addr. Modes	Operation
CLR CLRB	dst	R IR DA X	Clear dst ← 0
EX EXB	R, src	R IR DA X	Exchange R ← src
LD LDB LDL	R, src	R IM IM IR DA X BA BX	Load into Register R ← src
LD LDB LDL	dst, R	IR DA X BA BX	Load into Memory (Store) dst ← R
LD LDB	dst, IM	IR DA X	Load Immediate into Memory dst ← IM
LDA	R, src	DA X BA BX	Load Address R ← source address
LDAR	R, src	RA	Load Address Relative R ← source address
LDK	R, src	IM	Load Constant R ← n (n = 0 . . . 15)
LDM	R, src, n	IR DA X	Load Multiple R ← src (n consecutive words) (n = 1 . . . 16)
LDM	dst, R, n	IR DA X	Load Multiple (Store Multiple) dst ← R (n consecutive words) (n = 1 . . . 16)
LDR LDRB LDRL	R, src	RA	Load Relative R ← src (range −32768 . . . +32767)
LDR LDRB LDRL	dst, R	RA	Load Relative (Store Relative) dst ← R (range −32768 . . . +32767)
POP POPL	dst, R	R IR DA X	Pop dst ← IR Autoincrement contents of R
PUSH PUSHL	IR, src	R IM IR DA X	Push Autodecrement contents of R IR ← src

ARITHMETIC

Mnemonics	Operands	Addr. Modes	Operation
ADC ADCB	R, src	R	Add with Carry R ← R + src + carry
ADD ADDB ADDL	R, src	R IM IR DA X	Add R ← R + src
CP CPB CPL	R, src	R IM IR DA X	Compare with Register R − src
CP CPB	dst, IM	IR DA X	Compare with Immediate dst − IM
DAB	dst	R	Decimal Adjust
DEC DECB	dst, n	R IR DA X	Decrement by n dst ← dst − n (n = 1 . . . 16)
DIV DIVL	R, src	R IM IR DA X	Divide (signed) Word: $R_{n+1} \leftarrow R_{n,n+1} \div$ src $R_n \leftarrow$ remainder Long Word: $R_{n+2,n+3} \leftarrow R_{n...n+3} \div$ src $R_{n,n+1} \leftarrow$ remainder
EXTS EXTSB EXTSL	dst	R	Extend Sign Extend sign of low order half of st through high order half of dst
INC INCB	dst, n	R IR DA X	Increment by n dst ← dst + n (n = 1 . . . 16)
MULT MULTL	R, src	R IM IR DA X	Multiply (signed) Word: $R_{n,n+1} \leftarrow R_{n+1} \cdot$ src Long Word: $R_{n...n+3} \leftarrow R_{n+2,n+3} \cdot$ src *Plus seven cycles for each 1 in the multiplicand
NEG NEGB	dst	R IR DA X	Negate dst ← 0 − dst
SBC SBCB	R, src	R	Subtract with Carry R ← R − src − carry
SUB SUBB SUBL	R,src	R IM IR DA X	Subtract R ← R − src

LOGICAL

Mnemonics	Operands	Addr. Modes	Operation
AND ANDB	R, src	R IM IR DA X	AND R ← R AND src
COM COMB	dst	R IM IR DA X	Complement dst ← NOT dst
OR ORB	R, src	R IM IR DA X	OR R ← R OR src
TEST TESTB TESTL	dst	R IR DA X	TEST dst OR 0
TCC TCCB	cc, dst	R	Test Condition Code Set LSB if cc is true
XOR XORB	R, src	R IM IR DA X	Exclusive OR R ← R XOR src

PROGRAM CONTROL

Mnemonics	Operands	Addr. Modes	Operation
CALL	dst	IR DA X	Call Subroutine Autodecrement SP @ SP ← PC PC ← dst
CALR	dst	RA	Call Relative Autodecrement SP @ SP ← PC PC ← PC + dst (range −4094 to +4096)
DJNZ DBJNZ	R, dst	RA	Decrement and Jump if Non-Zero R ← R − 1 IF R = 0: PC ← PC + dst (range −254 to 0)
IRET*	–	–	Interrupt Return PS ← @ SP Autoincrement SP
JP	cc, dst	IR IR DA X	Jump Conditional If cc is true: PC ← dst
JR	cc, dst	RA	Jump Conditional Relative If cc is true: PC ← PC + dst (range −256 to +254)
RET	cc	–	Return Conditional If cc is true: PC ← @ SP Autodecrement SP
SC	src	IM	System Call Autodecrement SP @ SP ← old PS Push instruction PS ← System Call PS

*Privileged instructions. Executed in system mode only.

BIT MANIPULATION

Mnemonics	Operand	Addr. Modes	Operation
BIT BITB	dst, b	R IR DA X	Test Bit Static Z flag ← NOT dst bit specified by b
BIT BITB	dst, R	R	Test Bit Dynamic Z flag ← NOT dst bit specified by contents of R
RES RESB	dst, b	R IR DA X	Reset Bit Static Reset dst bit specified by b
RES RESB	dst, R	R	Reset Bit Dynamic Reset dst bit specified by contents of R
SET SETB	dst, b	R IR DA X	Set Bit Static Set dst bit specified by b
SET SETB	dst, R	R	Set Bit Dynamic Set dst bit specified by contents of R
TSET TSETB	dst	R IR DA X	Test and Set S flag ← MSB of dst dst ← all 1s

ROTATE AND SHIFT

Mnemonics	Operand	Addr. Modes	Operation
RLDB	R, src	R	Rotate Digit Left
RRDB	R, src	R	Rotate Digit Right
RL RLB	dst, n	R R	Rotate Left by n bits (n = 1, 2)
RLC RLCB	dst, n	R R	Rotate Left through Carry by n bits (n = 1, 2)
RR RRB	dst, n	R R	Rotate Right by n bits (n = 1, 2)
RRC RRCB	dst, n	R R	Rotate Right through Carry by n bits (n = 1, 2)
SDA SDAB SDAL	dst, R	R	Shift Dynamic Arithmetic Shift dst left or right by contents of R
SDL SDLB SDLL	dst, R	R	Shift Dynamic Logical Shift dst left or right by contents of R
SLA SLAB SLAL	dst, n	R	Shift Left Arithmetic by n bits
SLL SLLB SLLL	dst, n	R	Shift Left Logical by n bits
SRA SRAB SRAL	dst, n	R	Shift Right Arithmetic by n bits
SRL SRLB SRLL	dst, n	R	Shift Right Logical by n bits

4

BLOCK TRANSFER AND STRING MANIPULATION

Mnemonics	Operands	Addr. Modes	Operation
CPD CPDB	R_X, src, R_Y, cc	IR	Compare and Decrement R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$
CPDR CPDRB	R_X, src, R_Y, cc	IR	Compare, Decrement and Repeat R_X − src Autodecrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPI CPIB	R_X, src, R_Y, cc	IR	Compare and Increment R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$
CPIR CPIRB	R_X, src, R_Y, cc	IR	Compare, Increment and Repeat R_X − src Autoincrement src address $R_Y \leftarrow R_Y - 1$ Repeat until cc is true or $R_Y = 0$
CPSD CPSDB	dst, src, R, cc	IR	Compare String and Decrement dst − src Autodecrement dst and src addresses R ← R − 1
CPSDR CPSDRB	dst, src, R, cc	IR	Compare String, Decr. and Repeat dst − src Autodecrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
CPSI CPSIB	dst, src, R, cc	IR	Compare String and Increment dst − src Autoincrement dst and src addresses R ← R − 1
CPSIR CPSIRB	dst, src, R, cc	IR	Compare String, Incr. and Repeat dst − src Autoincrement dst and src addresses R ← R − 1 Repeat until cc is true or R = 0
LDD LDDB	dst, src, R	IR	Load and Decrement dst ← src Autodecrement dst and src addresses R ← R − 1
LDDR LDDRB	dst, src, R	IR	Load, Decrement and Repeat dst ← src Autodecrement dst and src addresses R ← R − 1 Repeat until R = 0

BLOCK TRANSFER AND STRING MANIPULATION (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
LDI LDIB	dst, src, R	IR	Load and Increment dst ← src Autoincrement dst and src addresses R ← R − 1
LDIR LDIRB	dst, src, R	IR	Load, Increment and Repeat dst ← src Autoincrement dst and src addresses R ← R − 1 Repeat until R = 0
TRDB	dst, src, R	IR	Translate and Decrement dst ← src (dst) Autodecrement dst address R ← R − 1
TRDRB	dst, src, R	IR	Translate, Decrement and Repeat dst ← src (dst) Autodecrement dst address R ← R − 1 Repeat until R = 0
TRIB	dst, src, R	IR	Translate and Increment dst ← src (dst) Autoincrement dst address R ← R − 1
TRIRB	dst, src, R	IR	Translate, Increment and Repeat dst ← src (dst) Autoincrement dst address R ← R − 1 Repeat until R = 0
TRTDB	src 1, src 2, R	IR	Translate and Test, Decrement RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1
TRTDRB	src 1, src 2, R	IR	Translate and Test, Decrement and Repeat RH1 ← src 2 (src 1) Autodecrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0
TRTIB	src 1, src 2, R	IR	Translate and Test, Increment RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1
TRTIRB	src 1, src 2, R	IR	Translate and Test, Increment and Repeat RH1 ← src 2 (src 1) Autoincrement src 1 address R ← R − 1 Repeat until R = 0 or RH1 = 0

INPUT/OUTPUT

Mnemonics	Operands	Addr. Modes	Operation
IN* INB*	R, src	IR DA	Input R ← src
IND* INDB*	dst, src, R	IR	Input and Decrement dst ← src Autodecrement dst address R ← R − 1
INDR* INDRB*	dst, src, R	IR	Input, Decrement and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
INI* INIB*	dst, src, R	IR	Input and Increment dst ← src Autoincrement dst address R ← R − 1
INIR* INIRB*	dst, src, R	IR	Input, Increment and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0
OUT* OUTB*	dst, R	IR DA	Output dst ← R
OUTD* OUTDB*	dst, src, R	IR	Output and Decrement dst ← src Autodecrement src address R ← R − 1
OTDR* OTDRB*	dst, src, R	IR	Output, Decrement and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
OUTI* OUTIB*	dst, src, R	IR	Output and Increment dst ← src Autoincrement src address R ← R − 1
OTIR* OTIRB*	dst, src, R	IR	Ouput, Increment and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0
SIN* SINB*	R, src	DA	Special Input R ← src
SIND* SINDB*	dst, src, R	IR	Special Input and Decrement dst ← src Autodecrement dst address R ← R − 1
SINDR* SINDRB*	dst, src, R	IR	Special Input, Decr. and Repeat dst ← src Autodecrement dst address R ← R − 1 Repeat until R = 0
SINI* SINIB*	dst, src, R	IR	Special Input and Increment dst ← src Autoincrement dst address R ← R − 1
SINIR* SINIRB*	dst, src, R	IR	Special Input, Incr. and Repeat dst ← src Autoincrement dst address R ← R − 1 Repeat until R = 0

INPUT/OUTPUT (Cont.)

Mnemonics	Operands	Addr. Modes	Operation
SOUT* SOUTB*	dst, src	DA	Special Output dst ← src
SOUTD* SOUTDB*	dst, src, R	IR	Special Output and Decrement dst ← src Autodecrement src address R ← R − 1
SOTDR* SOTDRB*	dst, src, R	IR	Special Output, Decr. and Repeat dst ← src Autodecrement src address R ← R − 1 Repeat until R = 0
SOUTI* SOUTIB*	dst, src, R	IR	Special Output and Increment dst ← src Autoincrement src address R ← R − 1
SOTIR* SOTIRB*	dst, src, R	R	Special Output, Incr. and Repeat dst ← src Autoincrement src address R ← R − 1 Repeat until R = 0

CPU CONTROL

Mnemonics	Operands	Addr. Modes	Operation
COMFLG	flags	–	Complement Flag (Any combination of C, Z, S, P/V)
DI*	int	–	Disable Interrupt (Any combination of NVI, VI)
EI*	int	–	Enable Interrupt (Any combination of NVI, VI)
HALT*	–	–	HALT
LDCTL*	CTLR, src	R	Load into Control Register CTLR ← src
LDCTL*	dst, CTLR	R	Load from Control Register dst ← CTLR
LDCTLB	FLGR, src	R	Load into Flag Byte Register FLGR ← src
LDCTLB	dst, FLGR	R	Load from Flag Byte Register dst ← FLGR
LDPS*	src	IR DA X	Load Program Status PS ← src
MBIT*	–	–	Test Multi-Micro Bit Set S if $\overline{\mu I}$ is High; reset S if $\overline{\mu I}$ is Low.
MREQ*	dst	R	Multi-Micro Request
MRES*	–	–	Multi-Micro Reset
MSET*	–	–	Multi-Micro Set
NOP	–	–	No Operation
RESFLG	flag	–	Reset Flag (Any combination of C, Z, S, P/V)
SETFLG	flag	–	Set Flag (Any combination of C, Z, S, P/V)

*Privileged instructions. Executed in system mode only.

MAXIMUM RATINGS above which useful life may be impaired

Voltages on all inputs and outputs with respect to GND	−0.3 to +7.0V
Ambient Temperature under bias	0 to 70°C
Storage Temperature	−65 to +150°C

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

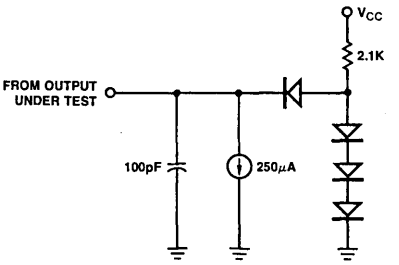
ELECTRICAL CHARACTERISTICS over operating range (Note 1)

			AmZ8002DC		
Parameter	**Description**	**Test Conditions**	**Min**	**Max**	**Units**
V_{CH}	Clock Input High Voltage	Driven by External Clock Generator	$V_{CC}-0.4$	$V_{CC}+0.3$	Volts
V_{CL}	Clock Input Low Voltage	Driven by External Clock Generator	−0.3	0.45	Volts
V_{IH}	Input High Voltage		2.0	$V_{CC}+0.3$	Volts
V_{IL}	Input Low Voltage		−0.3	0.8	Volts
V_{OH}	Output High Voltage	$I_{OH} = -250\mu A$	2.4		Volts
V_{OL}	Output Low Voltage	$I_{OL} = +2.0mA$		0.4	Volts
I_{IL}	Input Leakage	$0.4 \leq V_{IN} \leq +2.4V$		±10	μA
I_{OL}	Output Leakage	$0.4 \leq V_{OUT} \leq +2.4V$		±10	μA
I_{CC}	V_{CC} Supply Current			300	mA

Standard Test Conditions

The characteristics below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to GND. Positive current flows into the referenced pin. Standard conditions are as follows:

$+4.75V \leq V_{CC} \leq +5.25V$
GND = 0V
$0°C \leq T_A \leq +70°C$



MOS-238

All AC parameters assume a load capacitance of 100pF max. Timing references between two output signals assume a load difference of 50pF max.

SWITCHING CHARACTERISTICS over operating range

Number	Parameter	Description	AmZ8002DC Min	AmZ8002DC Max	Units
1	TcC	Clock Cycle Time	250	2000	ns
2	TwCh	Clock Width (High)	105	2000	ns
3	TwCl	Clock Width (Low)	105	2000	ns
4	TfC	Clock Fall Time		20	ns
5	TrC	Clock Rise Time		20	ns
6					
7					
8	TdC(Bz)	Clock ↑ to Bus Float		65	ns
9	TdC(A)	Clock ↑ to Address Valid		100	ns
10	TdC(Az)	Clock ↑ to Address Float		65	ns
11	TdA(DI)	Address Valid to Data In Required Valid	400		ns
12	TsDI(C)	Data In to Clock ↓ Set-up Time	70		ns
13	TdDS(A)	$\overline{DS}$ ↑ to Address Active	80		ns
14	TdC(DO)	Clock ↑ to Data Out Valid		100	ns
15	ThDI(DS)	Data In to $\overline{DS}$ ↑ Hold Time	0		ns
16	TdDO(DS)	Data Out Valid to $\overline{DS}$ ↑ Delay	230		ns
17	TdA(MR)	Address Valid to $\overline{MREQ}$ ↓ Delay	55		ns
18	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↓ Delay		80	ns
19	TwMRh	$\overline{MREQ}$ Width (High)	190		ns
20	TdMR(A)	$\overline{MREQ}$ ↓ to Address Not Active	70		ns
21	TdDO(DSW)	Data Out Valid to $\overline{DS}$ ↓ (Write) Delay	55		ns
22	TdMR(DI)	$\overline{MREQ}$ ↓ to Data In Required Valid	330		ns
23	TdC(MR)	Clock ↓ to $\overline{MREQ}$ ↑ Delay		80	ns
24	TdC(ASf)	Clock ↑ to $\overline{AS}$ ↓ Delay		80	ns
25	TdA(AS)	Address Valid to $\overline{AS}$ ↑ Delay	55		ns
26	TdC(ASr)	Clock ↓ to $\overline{AS}$ ↑ Delay		90	ns
27	TdAS(DI)	$\overline{AS}$ ↑ to Data In Required Valid	290		ns
28	TdDS(AS)	$\overline{DS}$ ↑ to $\overline{AS}$ ↓ Delay	70		ns
29	TwAS	$\overline{AS}$ Width (Low)	80		ns
30	TdAS(A)	$\overline{AS}$ ↑ to Address Not Active Delay	60		ns
31	TdAz(DSR)	Address Float to $\overline{DS}$ (Read) ↓ Delay	0		ns
32	TdAS(DSR)	$\overline{AS}$ ↑ to $\overline{DS}$ (Read) ↓ Delay	70		ns
33	TdDSR(DI)	$\overline{DS}$ (Read) ↓ to Data In Required Valid	155		ns
34	TdC(DSr)	Clock ↓ to $\overline{DS}$ ↑ Delay		70	ns
35	TdDS(DO)	$\overline{DS}$ ↑ to Data Out and STATUS Not Valid	80		ns
36	TdA(DSR)	Address Valid to $\overline{DS}$ (Read) ↓ Delay	120		ns
37	TdC(DSR)	Clock ↑ to $\overline{DS}$ (Read) ↓ Delay		120	ns
38	TwDSR	$\overline{DS}$ (Read) Width (Low)	275		ns
39	TdC(DSW)	Clock ↓ to $\overline{DS}$ (Write) ↓ Delay		95	ns
40	TwDSW	$\overline{DS}$ (Write) Width (Low)	160		ns
41	TdDSI(DI)	$\overline{DS}$ (Input) ↓ to Data In Required Valid	315		ns
42	TdC(DSf)	Clock ↓ to $\overline{DS}$ (I/O) ↓ Delay		120	ns
43	TwDS	$\overline{DS}$ (I/O) Width (Low)	400		ns
44	TdAS(DSA)	$\overline{AS}$ ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay	960		ns
45	TdC(DSA)	Clock ↑ to $\overline{DS}$ (Acknowledge) ↓ Delay		120	ns
46	TdDSA(DI)	$\overline{DS}$ (Acknowledge) ↓ to Data In Required Delay	420		ns
47	TdC(S)	Clock ↑ to Status Valid Delay		110	ns
48	TdS(AS)	Status Valid to $\overline{AS}$ ↑ Delay	40		ns

SWITCHING CHARACTERISTICS (Cont.)

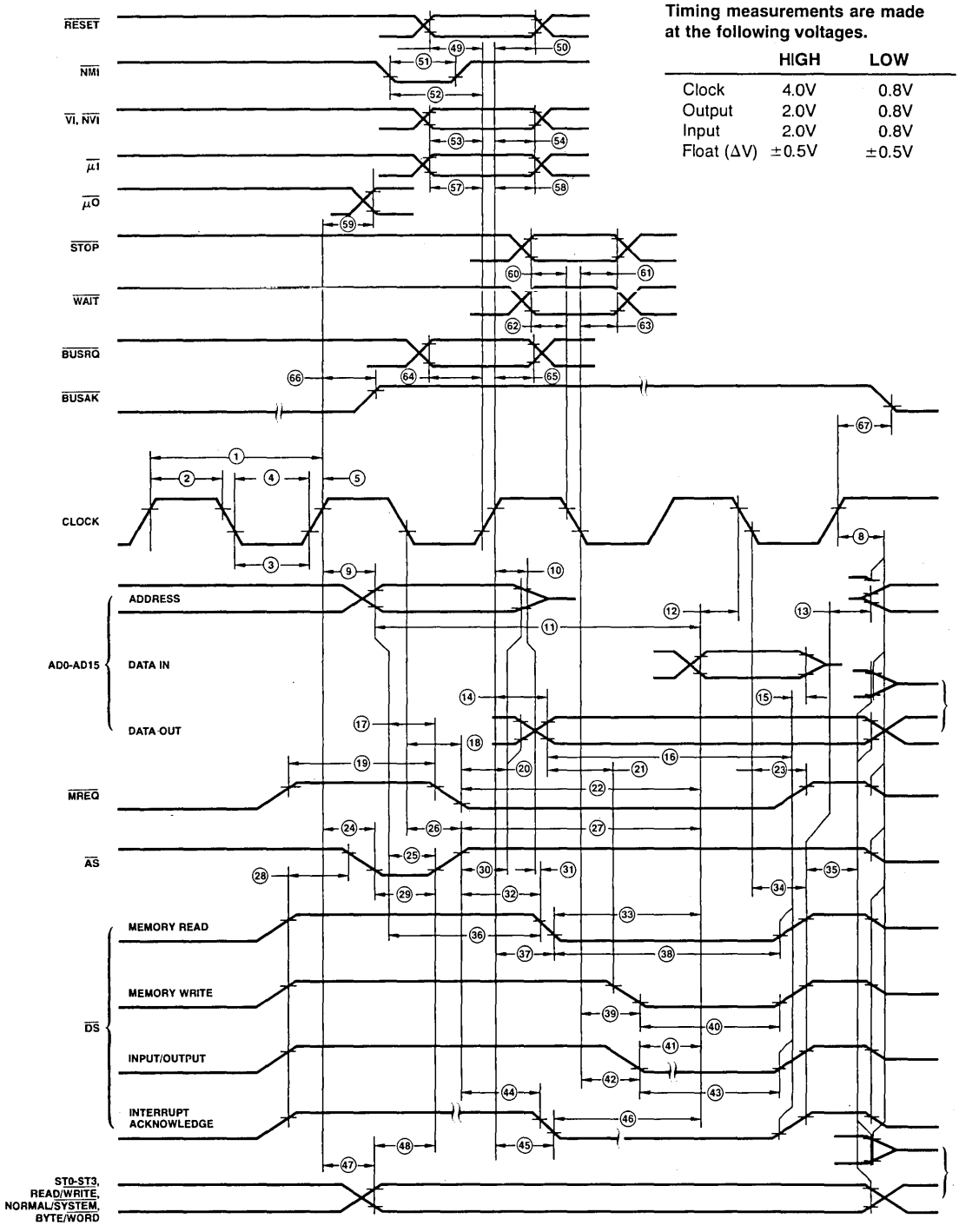
Number	Parameter	Description	AmZ8002DC Min	Max	Units
49	TsR(C)	$\overline{\text{RESET}}$ to Clock ↑ Set-up Time	180		ns
50	ThR(C)	$\overline{\text{RESET}}$ to Clock ↑ Hold Time	0		ns
51	TwNMI	$\overline{\text{NMI}}$ Width (Low)	100		ns
52	TsNMI(C)	$\overline{\text{NMI}}$ to Clock ↑ Set-up Time	140		ns
53	TsVI(C)	$\overline{\text{VI}}$, $\overline{\text{NVI}}$ to Clock ↑ Set-up Time	110		ns
54	ThVI(C)	$\overline{\text{VI}}$, $\overline{\text{NVI}}$ to Clock ↑ Hold Time	0		ns
55					
56					
57	TsμI(C)	$\overline{\mu\text{I}}$ to Clock ↑ Set-up Time	180		ns
58	ThμI(C)	$\overline{\mu\text{I}}$ to Clock ↑ Hold Time	0		ns
59	TdC(μo)	Clock ↑ to $\overline{\mu\text{O}}$ Delay		120	ns
60	TsSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Set-up Time	140		ns
61	ThSTP(C)	$\overline{\text{STOP}}$ to Clock ↓ Hold Time	0		ns
62	TsWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Set-up Time	70		ns
63	ThWT(C)	$\overline{\text{WAIT}}$ to Clock ↓ Hold Time	0		ns
64	TsBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Set-up Time	90		ns
65	ThBRQ(C)	$\overline{\text{BUSRQ}}$ to Clock ↑ Hold Time	0		ns
66	TdC(BAKr)	Clock ↑ to $\overline{\text{BUSAK}}$ ↑ Delay		100	ns
67	TdC(BAKf)	Clock ↑ to $\overline{\text{BUSAK}}$ ↓ Delay		100	ns

For more information, refer to these AMD publications:

Processor Instruction Set (AM-PUB086).
Describes each instruction in detail. 250 pp.

Processor Interface Manual (AM-PUB089).
Describes hardware interfacing for interrupts and I/O, including the Am9511A Arithmetic Processor, the Am9517A DMA Controller, and the Am9519 Interrupt Controller. 81 pp.

AC TIMING DIAGRAM



Timing measurements are made at the following voltages.

	HIGH	LOW
Clock	4.0V	0.8V
Output	2.0V	0.8V
Input	2.0V	0.8V
Float (ΔV)	±0.5V	±0.5V

This composite timing diagram does not show actual timing sequences. Refer to this diagram only for the detailed timing relationships of individual edges. Use the preceding illustrations as an explanation of the various timing sequences.

MOS-239

AmZ8010
Memory Management Unit

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

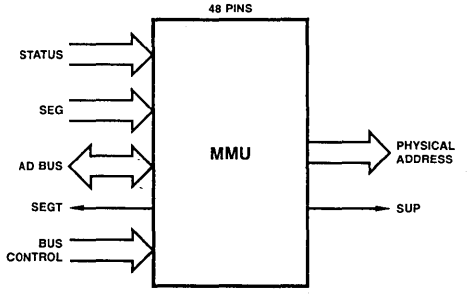
- Logical to Physical address translation
- Dynamic segment relocation
- 64 segments per MMU
- Segment sizes individually assignable from 256 to 64K bytes
- Size limit protection
- Segment protection attributes for system security
  - System only
  - Execute only
  - Read only
  - CPU only
  - Segment invalid
- Segment history aids paging
  - Segment referenced
  - Segment changed

GENERAL DESCRIPTION

The AmZ8010 Memory Management Unit (MMU) is a high-performance, LSI product which adds sophisticated address translation and memory protection capabilities to AmZ8001 CPU systems. Addresses output by the CPU consist of a 7-bit segment number and a 16-bit offset. The MMU uses the segment number to index an address translation table. The offset is added to the segment base to form the physical address. A separate table allows the user to individually program each segment size from 256 to 64K bytes.

The MMU also contains a table of access attributes which are individually programmable for each segment. Attributes provided are Read Only, System Mode Only, Invalid Segment, Execute Only, and CPU Only (Exclude DMA). A trap is issued to the CPU and writes to memory are suppressed if an access is attempted which is prohibited by the attributes or which falls outside of the programmed segment size.

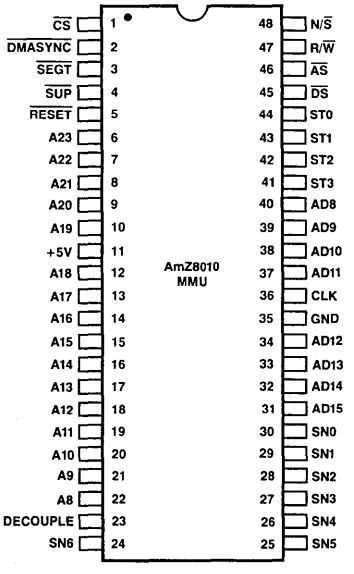
INTERFACE FLOW



MOS-290

For more information see "The AmZ8010 Memory Management Unit" AM-PUB093.

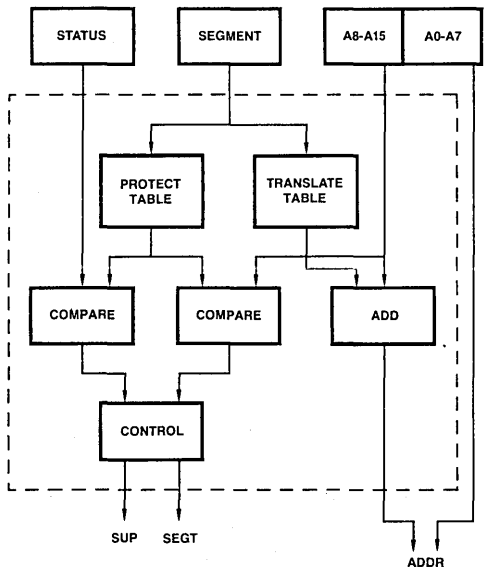
CONNECTION DIAGRAM
Top View



MOS-291

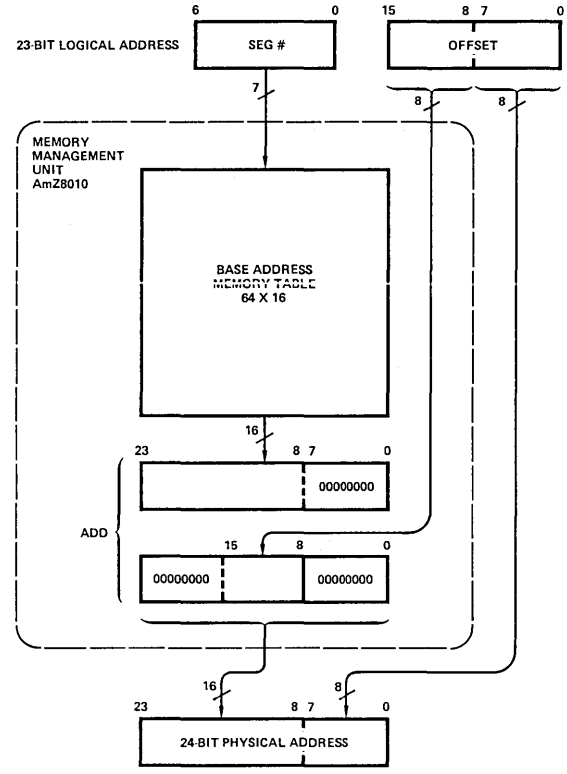
Note: Pin 1 is marked for orientation.

BLOCK DIAGRAM

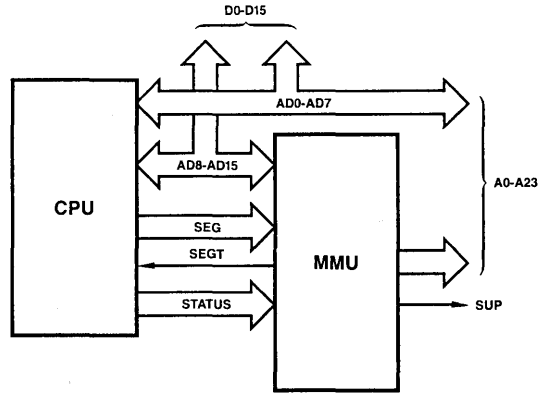


MOS-292

ADDRESS TRANSLATION MECHANISM



MOS-293



CPU TO MMU BLOCK INTERFACE

MOS-294

AmZ8016
DMA Transfer Controller

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

- Two independent high-speed DMA channels
- Memory/Peripheral transfers up to 2M bytes/sec
- Memory/Memory transfers up to 1.3M bytes/sec
- Fully compatible with AmZ8000 systems
- Full 8M byte logical addressing range
- Supports both logical and physical addressing
- Control parameters self-loaded from memory
- Successive transfer operations automatically chained without CPU intervention
- Base registers for efficient repetitive operations
- Programmable data matching with masking
- Vectored interrupts facilitate control coordination with CPU
- 48-pin DIP package
- +5V power supply

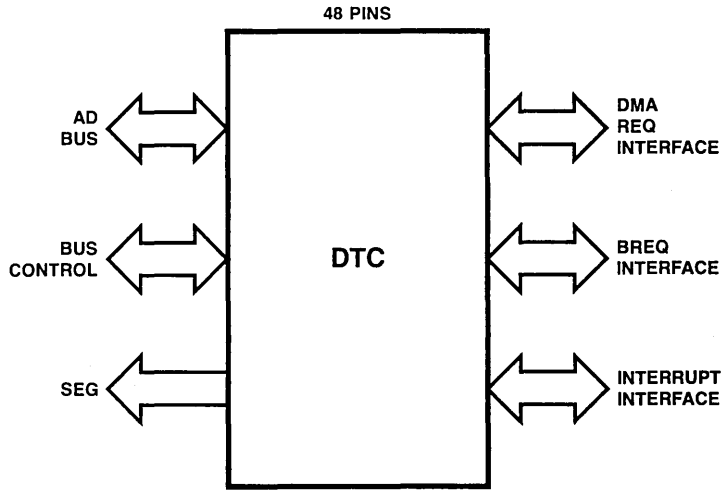
GENERAL DESCRIPTION

The AmZ8016 Direct Memory Access Transfer Controller (DTC) is a high-performance, LSI peripheral support circuit. With full AmZ8000 family bus compatibility, the DTC has been designed for ease of use in a variety of environments.

The AmZ8016 supports high-speed transfers between any mix of word/byte and memory/IO sources. To support variable length data blocks, the DTC permits search and transfer-and-search operations. Either 8-bit or 16-bit patterns may be used with masking, to permit searching for control codes or other specialized bit patterns. Communications between the DTC and the CPU is enhanced by an interrupt interface.

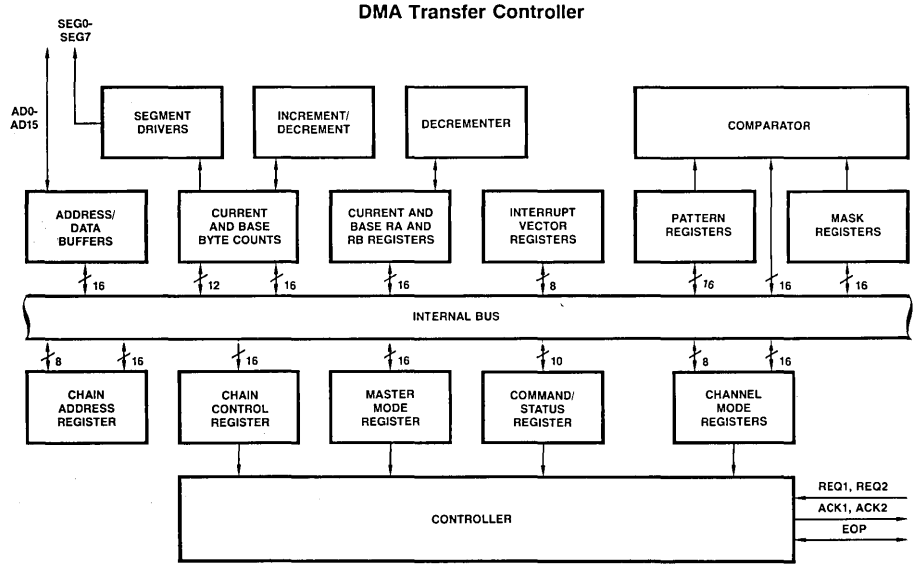
Included on the DTC are base registers permitting automatic reloading of the source, destination and byte count registers without bus overhead. Moreover, each channel contains circuitry capable of reloading any of the channel's registers from a control block in memory. Control blocks can be chained together. Since the DTC may be programmed to generate either physical or logical addresses it may be used either with or without an AmZ8010 Memory Management Unit and may be physically positioned with the peripheral (distributed DMA) or with the CPU (central DMA).

INTERFACE FLOW



MOS-295

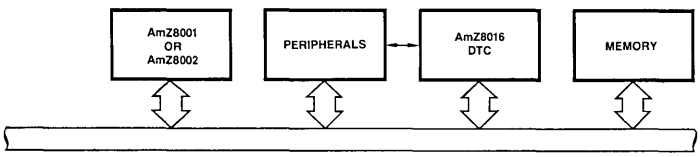
BLOCK DIAGRAM
DMA Transfer Controller



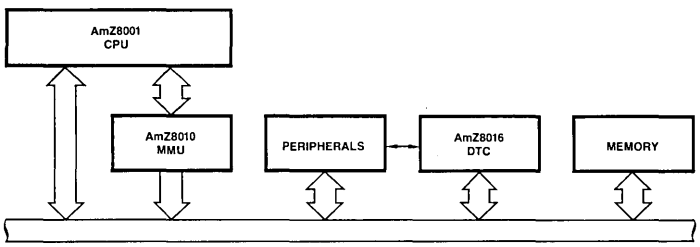
MOS-296

4

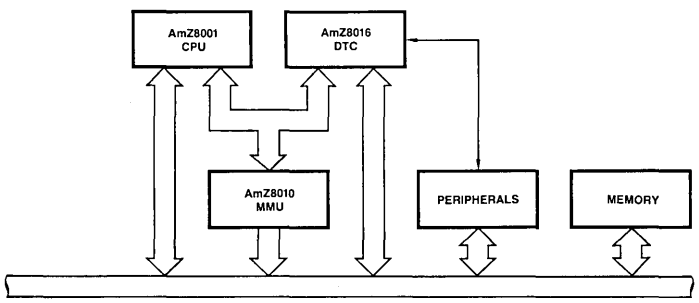
DTC CONFIGURATION OPTIONS



a) Unsegmented System Configuration



b) Segmented System with Physical DMA Addressing



c) Segmented System with Logical Addressing

MOS-297

AmZ8030

Serial Communications Controller

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

- Two independent serial full-duplex channels
- Buffered receiver and transmitter data registers
- All popular data formats
  - Asynchronous
  - Synchronous Byte Oriented protocols
  - Synchronous Bit Oriented protocols
- Data rates up to 880K bits/sec
- CRC-16 or CCITT block frame check
- Internal baud rate generation
- Separate modem controls for both channels
- Interrupt interface with programmable interrupt vectors
- 40-pin DIP package
- +5V power supply

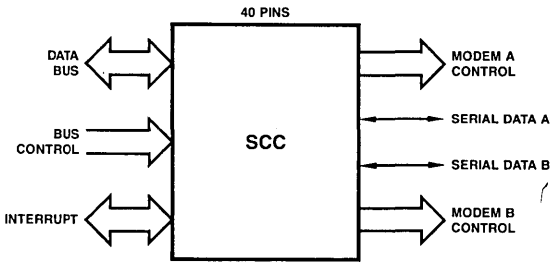
GENERAL DESCRIPTION

The AmZ8030 is a high-performance dual-channel multifunction serial communication controller (SCC). It is capable of handling serial-to-parallel and parallel-to-serial conversions for asynchronous modes, synchronous byte-oriented protocols such as IBM bisync, and synchronous bit-oriented protocols such as HDLC and SDLC. It generates and checks CRC codes in any synchronous mode.

It features two independent sets of four modem control signals with quadruply buffered receiver data and doubly buffered transmitter data for more tolerant host processor or DMA data transfers. Each channel includes baud rate generation.

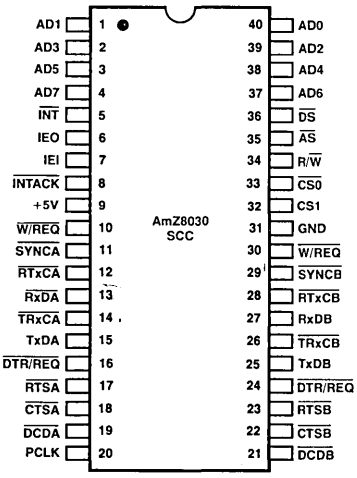
Its flexible daisy chain priority interrupt structure allows it to output an interrupt vector pointing to the correct routine by providing prioritized status information for the transmitter, receiver and External/Status interrupts. It can also be used in a polled environment or can output interrupt without vector.

INTERFACE FLOW



MOS-298

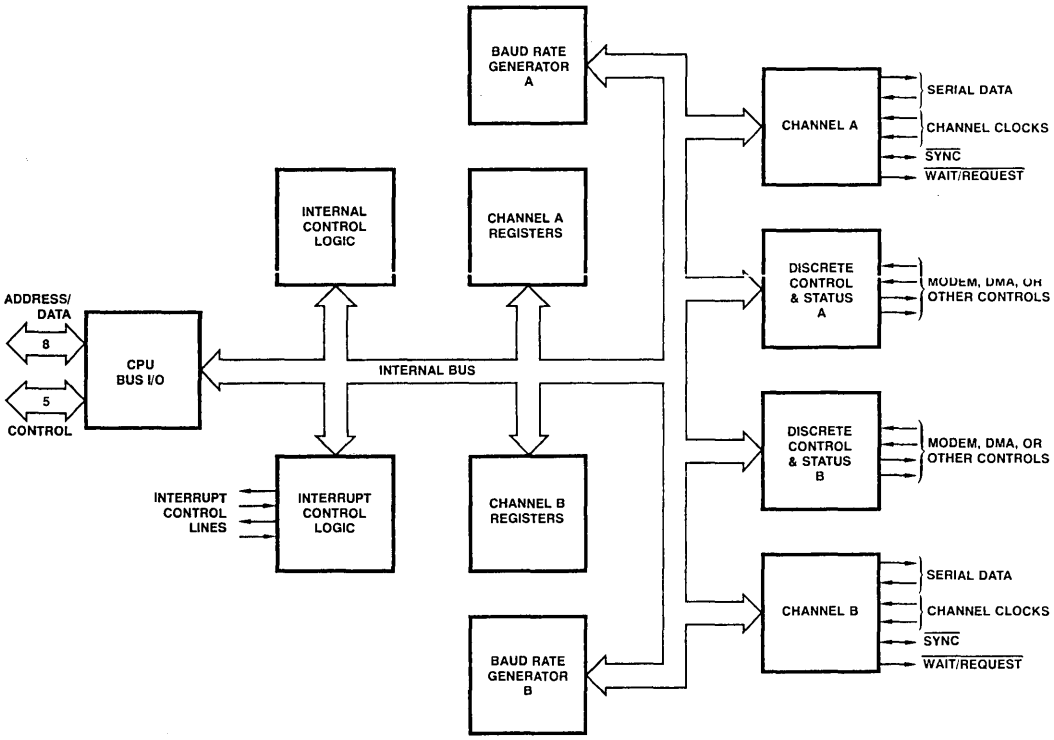
CONNECTION DIAGRAM
Top View



MOS-299

Note: Pin 1 is marked for orientation.

BLOCK DIAGRAM



MOS-300

4

AmZ8036

Counter/Timer and Parallel I/O Unit

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

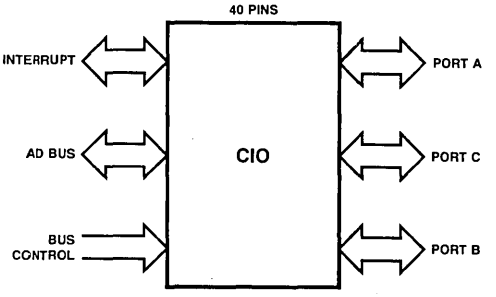
- Two general purpose 8-bit I/O ports
- Special purpose 4-bit control port
- Three independent 16-bit counter/timers
- Data Pattern matching with masking
- Interrupt control and interface
- Bit interface option
- One's catching input option
- Open drain output option
- Interlocked, pulsed, strobed or 3-wire handshaking
- 40-pin DIP package
- +5V power supply

GENERAL DESCRIPTION

The AmZ8036 counter/timer and parallel I/O is a general purpose peripheral device. It can perform as an interrupt controller as well as a flexible parallel data interface with various handshaking modes. Two double-buffered general purpose 8-bit ports can be used independently or linked to form a 16-bit port. The port bits can also operate individually with each bit direction individually programmable. A 4-bit port can be used as I/O or to provide handshake controls. Data pattern matching with individually maskable bits is available for each 8-bit port.

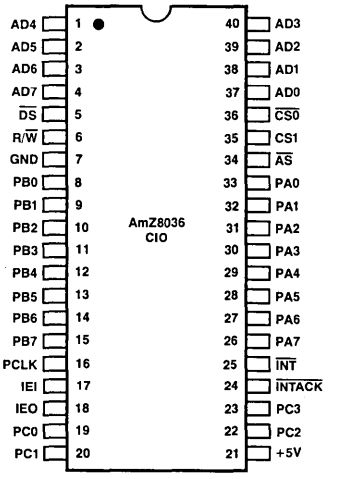
Three independent 16-bit counters can operate in single or continous cycles. Two may be linked together internally. The counters can count internal clocks or can count external events.

INTERFACE FLOW



MOS-301

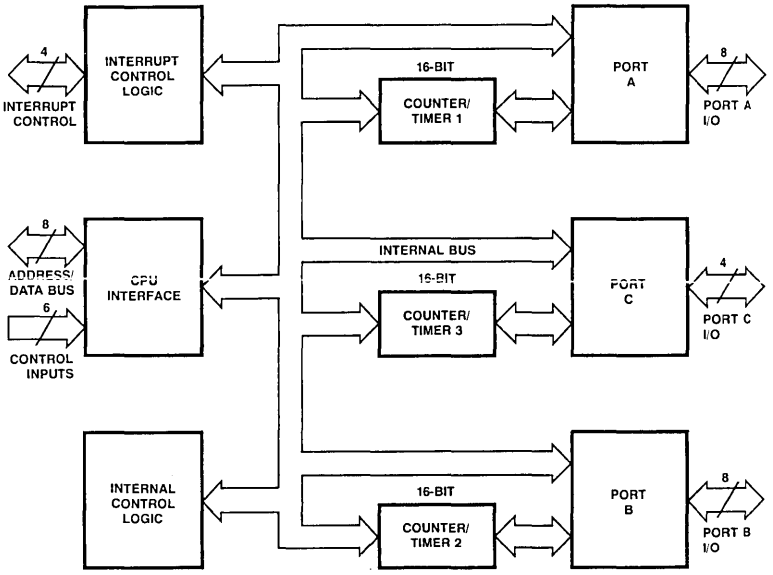
CONNECTION DIAGRAM
Top View



MOS-302

Note: Pin 1 is marked for orientation.

BLOCK DIAGRAM



MOS-303

4

AmZ8038

FIFO Input/Output Interface Unit

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

- General purpose FIFO buffered 8-bit I/O port
- 128 x 8 internal bidirectional FIFO
- Expandable in width and depth
- Buffers CPU to CPU and CPU to peripheral
- Simultaneous asynchronous read and write operations
- Interrupt Daisy Chain Interface
- Eight prioritized vectored interrupt sources
- General purpose or CPU-Bus interface
- Flyby DMA interface
- Pattern matching with mask
- IEEE-488 or Interlocked handshake post
- 40-pin DIP package
- +5V power supply

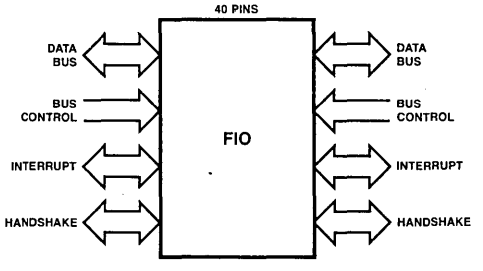
GENERAL DESCRIPTION

The AmZ8038 FIFO I/O (FIO) is a general-purpose 8-bit I/O port with a 128 x 8 bidirectional FIFO buffer memory between the port and the CPU bus. It functions as a buffered "elastic" general purpose interface. It has an 8-bit Master side which controls direction of data transfer, and 8-bit slave side which follows the data direction. It is capable of simultaneous, asynchronous, independent read and write operations.

The AmZ8038 has several control lines that may be used to interface to a DMA device or to synchronize the servicing processor. Two handshake lines allow direct interfacing to other peripheral devices. Buffer status (FULL and EMPTY) is also available as separate pins. Byte pattern matching with individual bit masking is available to generate an interrupt or to disable data loading. A daisy chain interrupt structure with eight prioritized Vectors is available for status information to the host processor.

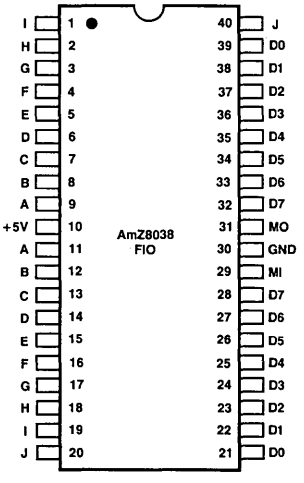
Many types of handshaking interfaces are available including Interlocked and IEEE-488.

INTERFACE FLOW



MOS-304

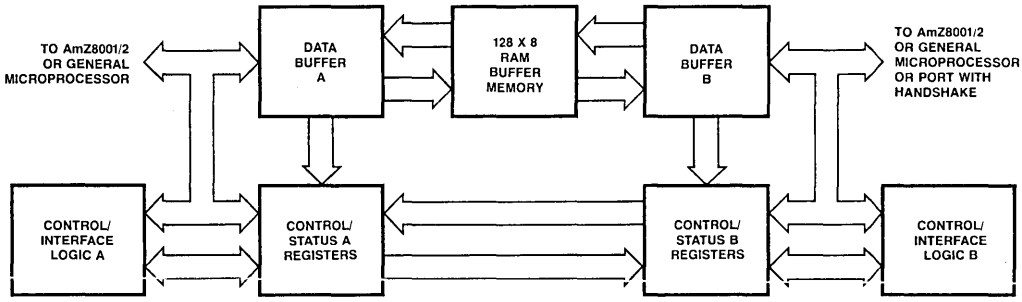
CONNECTION DIAGRAM
Top View

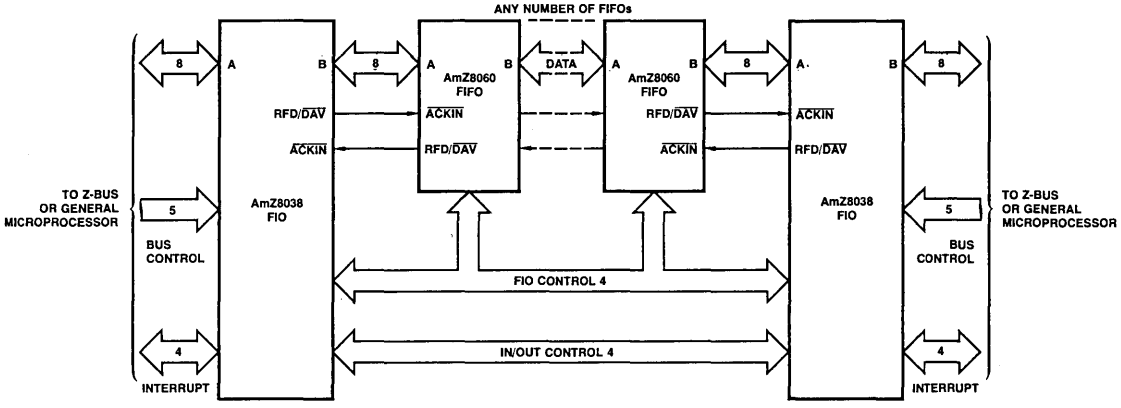


MOS-305

Note: Pin 1 is marked for orientation.

BLOCK DIAGRAM





MOS-306

AmZ8052
CRT Controller

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

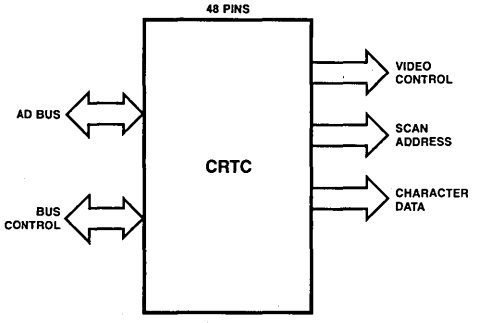
- 16-bit Address/Data port for host processor
- Multiple character attributes on a per-character basis
- Redefinable row attributes
- Linked list buffer addressing
- External source vertical frame rate sync input
- Flexible vertical and horizontal sync and blank control
- Programmable vertical and horizontal split-screen capability
- 5-bit Character Generator row addressing
- Super/subscript capability
- Multiple cursors
- Variable scans per row
- 48-pin package
- Various blink rates and duty cycles for characters and cursor
- Variable character clock frequency
- +5V power supply

GENERAL DESCRIPTION

The AmZ8052 CRTC is a new generation raster scan alpha-numeric display controller. It features host processor access to the display memory without contention; linked list addressing of character data; double character per cell display with variable vertical position; vertical retrace synchronization to an external source; and a variable character clock frequency input for character justification or character font mix. Scan lines per character are variable on row-by-row basis. Blinking rates are individually selectable for cursors and characters.

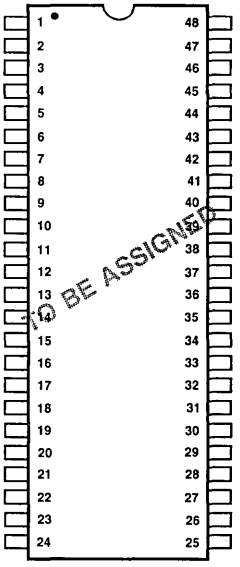
Row attributes define the address of the character data, scan line count of the current row, vertical shift options per character row and vertical split screen location. Character attributes are stored on a character-by-character basis.

INTERFACE FLOW



MOS-307

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MOS-308

4

AmZ8060

FIFO Buffer Unit and FIO Expander

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

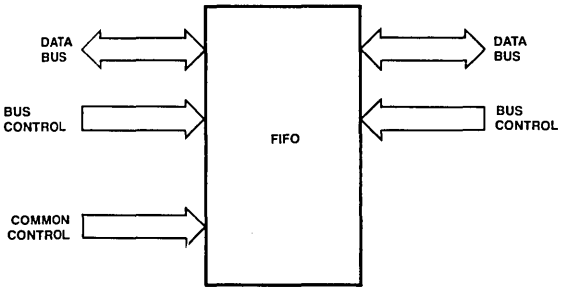
- General purpose 8-bit FIFO
- 128 x 8 bidirectional FIFO
- Expandable in width and depth
- Extends depth capability of AmZ8038 FIO
- Asynchronous
- +5V power supply

GENERAL DESCRIPTION

The AmZ8030 FIFO is a 128 x 8 bidirectional buffer. It is designed to operate as a stand-alone FIFO or to be used to expand the AmZ8033 buffer depth. The FIFO buffers can be cascaded without limit by daisy-chaining the RFD/DAV and ACKIN signals. The structure of the AmZ8038 FIFOs are similar to the Am2812. The FIFO is capable of simultaneous, asynchronous, independent read and write operations.

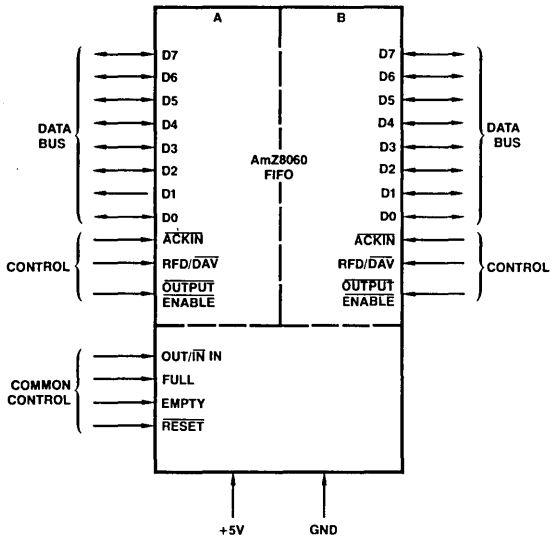
The AmZ8038 has control lines that may be used to interface to other devices or to synchronize the servicing processor. Two handshake lines allow direct interfacing to other peripheral devices. Buffer status (FULL and EMPTY) is also available as separate pins.

INTERFACE FLOW



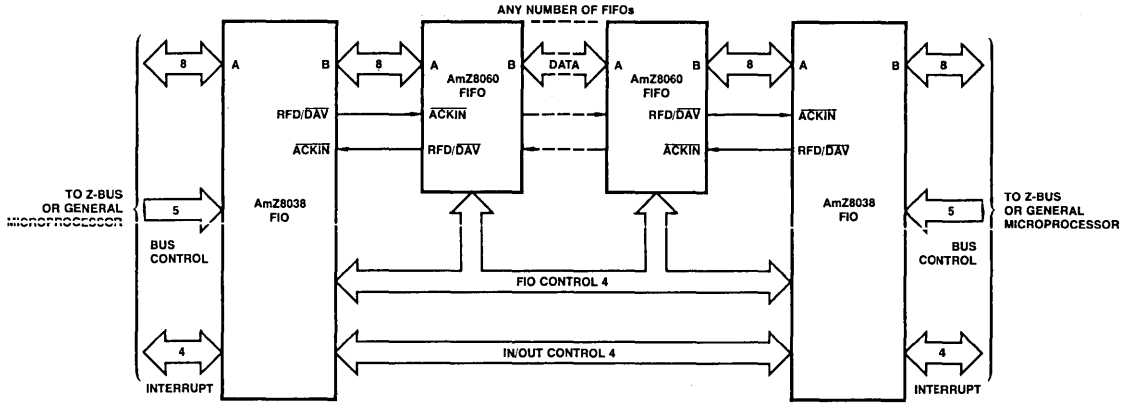
MOS-309

CONNECTION DIAGRAM
Top View



MOS-310

Note: Pin 1 is marked for orientation.



EXTENDING THE FIOs WITH FIFOs

MOS-311

AmZ8065

Burst Error Processor

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

- Burst error detection and correction
- Four selectable industry-standard polynomials including 56 and 48-bit IBM versions
- Serial data rates up to 20M bits/sec
- 12-bit error pattern output
- High-speed correction facilities based on Chinese Remainder theorem
- Single burst error correction
- Selectable error check modes in write mode
- 8-bit data port
- 40-pin DIP package
- −5V power supply

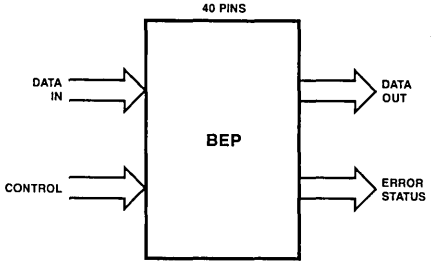
GENERAL DESCRIPTION

The AmZ8065 Burst Error Processor (BEP) is a peripheral interface circuit for serial or parallel data error detection and correction. It is used in applications such as high performance disk systems. Four different generator polynomials are internally encoded to satisfy a broad range of applications.

Data is entered in 8-bit (byte) parallel format and check bits are provided in the same parallel format.

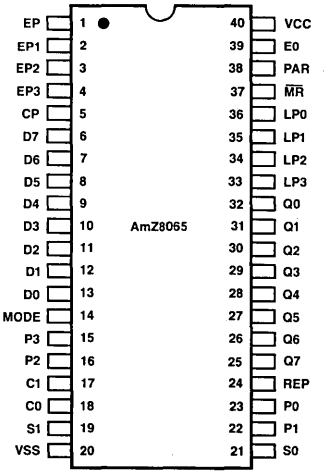
Write data is entered on the fly into the AmZ8065 while blocks are written to the associated disk and check bits are extracted following the last data byte. A Read Normal mode performs extraction of the error pattern and location while a Read High Speed mode allows direct division of data by the factors of the generator polynomial. A Divide mode generates output check bits and validates data. The Compute mode initiates a data correction process by locating and outputting the error pattern for correction.

INTERFACE FLOW



MOS-312

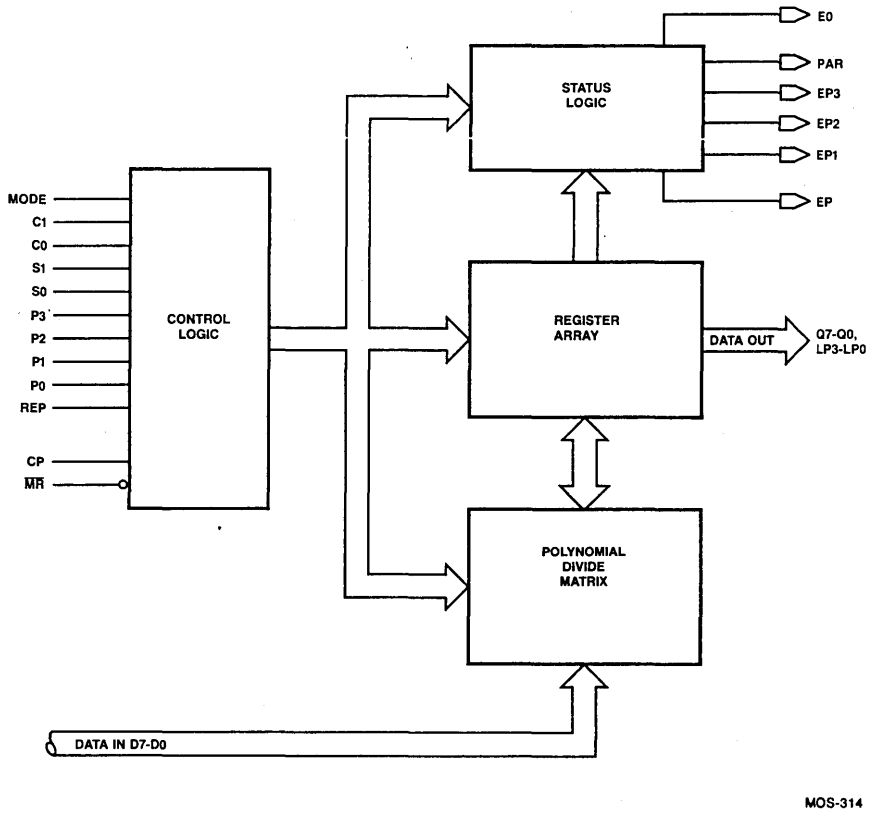
CONNECTION DIAGRAM
Top View



MOS-313

Note: Pin 1 is marked for orientation.

BLOCK DIAGRAM



MOS-314

AmZ8068

Data Ciphering Processor

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

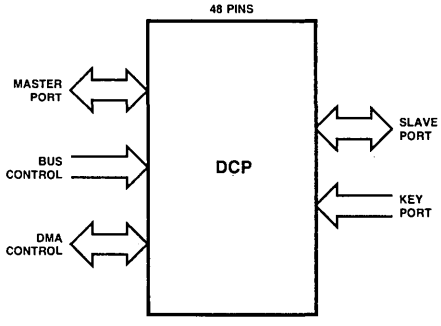
- Standard encryption and decryption algorithms
- Throughput rates greater than 1 megabyte per second
- Supports three standard ciphering options
  - Electronic Code Book
  - Cipher Feedback
  - Chain Block
- Master, Encrypt and Decrypt key registers
- Key parity check
- Separate key port
- Session keys and Initializing Vectors may be entered encrypted or clear
- Master data port for bidirectional bus operation
- Slave data port for pipelined operation
- 40-pin DIP package
- +5V power supply

GENERAL DESCRIPTION

The AmZ8068 Data Ciphering Processor is an N-channel silicon gate LSI product containing the circuitry necessary to encrypt and decrypt data using the National Bureau of Standards encryption algorithm. It is designed to be used in a variety of environments including dedicated controllers, communication concentrators, terminals and peripheral task processors in general processor systems. The DCP provides a high throughput rate using Cipher Feedback, Electronic Code Book or Chain Block Cipher operating modes. Separate ports are provided for key input, clear data and enciphered data to enhance security.

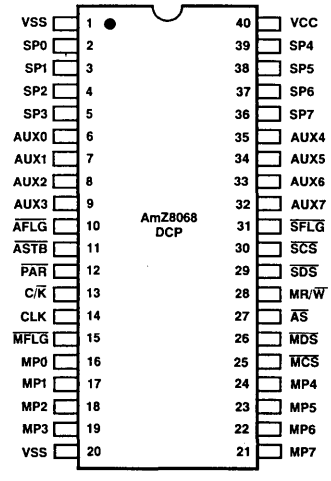
The system communicates with the Am8068 using commands entered in the master port and through auxiliary control lines. Once set up, data can flow through the DCP at high speeds because input, output and ciphering activities are all performed concurrently. External DMA control can easily be used to enhance throughput in some system configurations. This device is designed to interface directly to the AmZ8000 CPU bus and, with a minimum of external logic, to the 2900, 8080, 8085 and 8048 families of processors.

INTERFACE FLOW



MOS-315

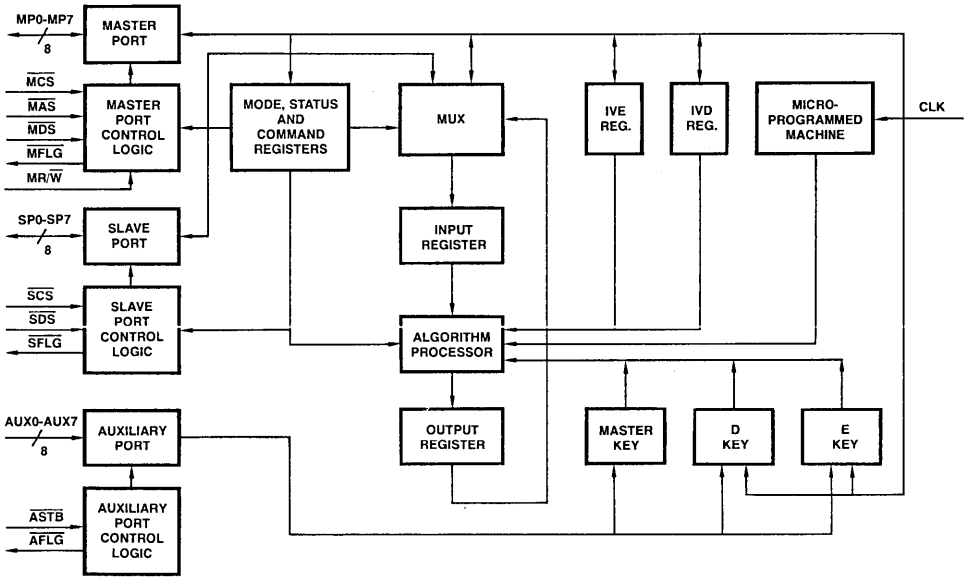
CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MOS-326

BLOCK DIAGRAM



MOS-316

4

AmZ8073
System Timing Controller

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

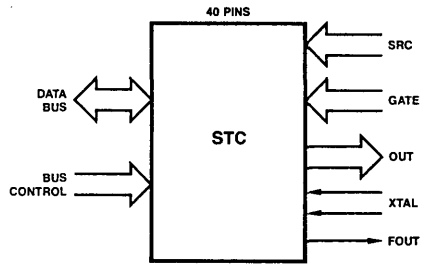
- Five independent 16-bit counters
- High speed counting rates
- Up/down and binary/BCD counting
- Internal oscillator frequency source
- Tapped frequency scaler
- Programmable frequency output
- 8-bit or 16-bit bus interface
- Time-of-day option
- Alarm comparators on counters 1 and 2
- Complex duty cycle outputs
- One-shot or continuous outputs
- Programmable count/gate source selection
- Programmable input and output polarities
- Programmable gating functions
- Retriggering capability
- +5 volt power supply
- Standard 40-pin package
- 100% MIL-STD-883 reliability assurance testing

GENERAL DESCRIPTION

The AmZ8073 System Timing Controller is an LSI circuit designed to service many types of counting, sequencing and timing applications. It provides the capability for programmable frequency synthesis, high resolution programmable duty cycle waveforms, retriggerable digital one-shots, time-of-day clocking, coincidence alarms, complex pulse generation, high resolution baud rate generation, frequency shift keying, stop-watching timing, event count accumulation, waveform analysis and many more. A variety of programmable operating modes and control features allow the AmZ8073 to be personalized for particular applications as well as dynamically reconfigured under program control.

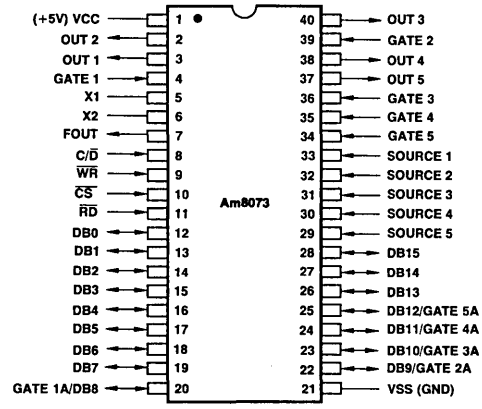
The STC includes five general-purpose 16-bit counters. A variety of internal frequency sources and external pins may be selected as inputs for individual counters with software selectable active-high or active-low input polarity. Both hardware and software gating of each counter is available. Three-state outputs for each counter provide pulses or levels and can be active-high or active-low. The counters can be programmed to count up or down in either binary or BCD. The host processor may read an accumulated count at any time without disturbing the counting process. Any of the counters may be internally concatenated to form any effective counter length up to 80 bits.

INTERFACE FLOW



MOS-317

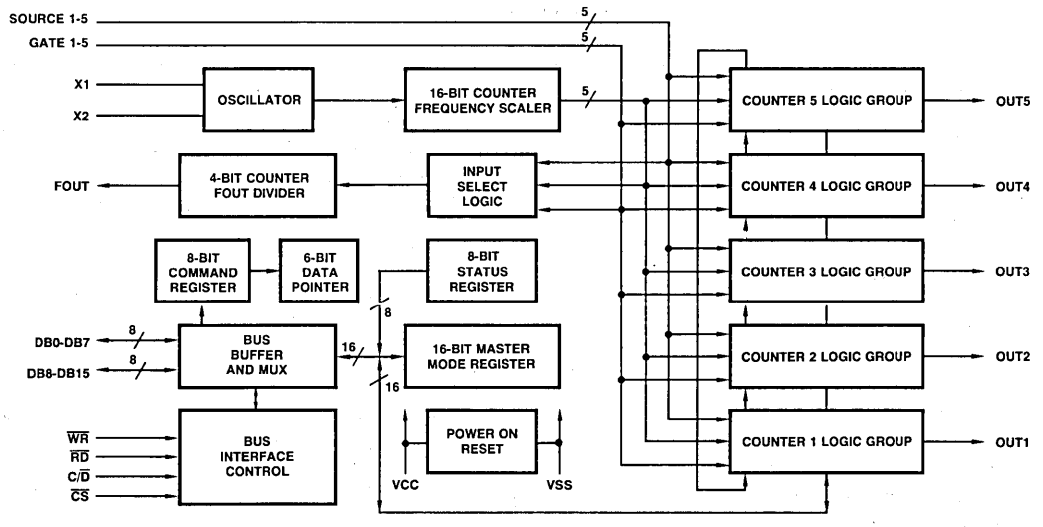
CONNECTION DIAMOND
Top View



MOS-318

Note: Pin 1 is marked for orientation.

FUNCTIONAL BLOCK DIAGRAM



MOS-319

4

AmZ8103 • AmZ8104

Octal Three-State Bidirectional Bus Transceivers

DISTINCTIVE CHARACTERISTICS

- 8-bit bidirectional data flow reduces system package count
- 3-state inputs/outputs for interfacing with bus-oriented systems
- PNP inputs reduce input loading
- VCC – 1.15V VOH interfaces with TTL, MOS and CMOS
- 48mA, 300pF bus drive capability
- AmZ8103 inverting transceivers
- AmZ8104 non-inverting transceivers
- Transmit/$\overline{\text{Receive}}$ and Chip Disable simplify control logic
- 20-pin ceramic and molded DIP package
- Low power – 8mA per bidirectional bit
- Advanced Schottky processing
- Bus port stays in hi-impedance state during power up/down
- 100% product assurance screening to MIL-STD-883 requirements

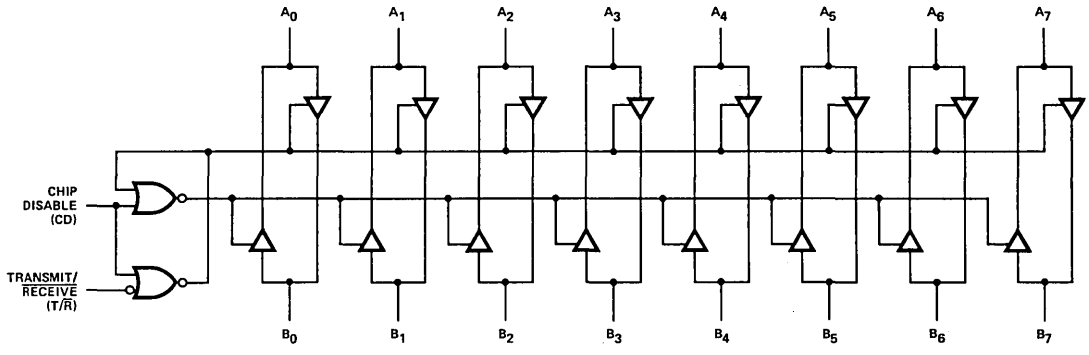
FUNCTIONAL DESCRIPTION

The AmZ8103 and AmZ8104 are 8-bit 3-state Schottky transceivers. They provide bidirectional drive for bus-oriented microprocessor and digital communications systems. Straight through bidirectional transceivers are featured, with 24mA drive capability on the A ports and 48mA bus drive capability on the B ports. PNP inputs are incorporated to reduce input loading.

One input, Transmit/$\overline{\text{Receive}}$ determines the direction of logic signals through the bidirectional transceiver. The Chip Disable input disables both A and B ports by placing them in a 3-state condition. Chip Disable is functionally the same as an active LOW chip select.

The output high voltage (VOH) is specified at VCC – 1.15V minimum to allow interfacing with MOS, CMOS, TTL, ROM, RAM, or microprocessors.

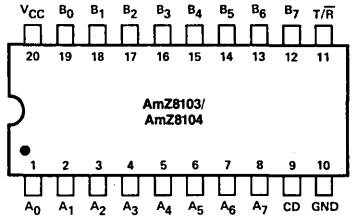
AmZ8104 LOGIC DIAGRAM



AmZ8103 has inverting transceivers.

BLI-216

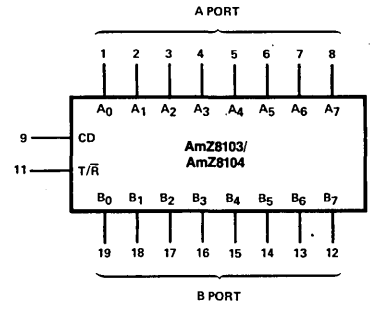
CONNECTION DIAGRAM Top View



BLI-169

Note: Pin 1 is marked for orientation.

LOGIC SYMBOL



BLI-170

VCC = Pin 20
GND = Pin 10

AmZ8103 • AmZ8104

ABSOLUTE MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Supply Voltage	7.0V
Input Voltage	5.5V
Output Voltage	5.5V
Lead Temperature (Soldering, 10 seconds)	300°C

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Noted:

MIL T_A = −55 to +125°C VCC MIN = 4.5V VCC MAX = 5.5V
COM'L T_A = 0 to 70°C VCC MIN = 4.75V VCC MAX = 5.25V

DC ELECTRICAL CHARACTERISTICS over operating temperature range

Parameters	Description		Test Conditions		Min	Typ (Note 1)	Max	Units
			A PORT (A0-A7)					
VIH	Logical "1" Input Voltage		CD = VIL MAX, T/R = 2.0V		2.0			Volts
VIL	Logical "0" Input Voltage		CD = VIL MAX, T/$\overline{R}$ = 2.0V	COM'L			0.8	Volts
				MIL			0.7	
VOH	Logical "1" Output Voltage		CD = VIL MAX, T/$\overline{R}$ = 0.8V	IOH = −0.4mA	VCC−1.15	VCC−0.7		Volts
				IOH = −3.0mA	2.7	3.95		
VOL	Logical "0" Output Voltage		CD = VIL MAX, T/$\overline{R}$ = 0.8V	IOL = 12mA		0.3	0.4	Volts
				COM'L, IOL = 24mA		0.35	0.50	
IOS	Output Short Circuit Current		CD = VIL MAX, T/$\overline{R}$ = 0.8V, VO = 0V, VCC = MAX, Note 2		−10	−38	−75	mA
IIH	Logical "1" Input Current		CD = VIL MAX, T/$\overline{R}$ = 2.0V, VI = 2.7V			0.1	80	μA
II	Input Current at Maximum Input Voltage		CD = 2.0V, VCC = MAX, VI = VCC MAX				1	mA
IIL	Logical "0" Input Current		CD = VIL MAX, T/$\overline{R}$ = 2.0V, VI = 0.4V			−70	−200	μA
VC	Input Clamp Voltage		CD = 2.0V, IIN = −12mA			−0.7	−1.5	Volts
IOD	Output/Input 3-State Current		CD = 2.0V	VO = 0.4V			−200	μA
				VO = 4.0V			80	
			B PORT (B0-B7)					
VIH	Logical "1" Input Voltage		CD = VIL MAX, T/$\overline{R}$ = VIL MAX		2.0			Volts
VIL	Logical "0" Input Voltage		CD = VIL MAX, T/R = VIL MAX	COM'L			0.8	Volts
				MIL			0.7	
VOH	Logical "1" Output Voltage		CD = VIL MAX, T/$\overline{R}$ = 2.0V	IOH = −0.4mA	VCC−1.15	VCC−0.8		Volts
				IOH = −5mA	2.7	3.9		
				IOH = −10mA	2.4	3.6		
VOL	Logical "0" Output Voltage		CD = VIL MAX, T/$\overline{R}$ = 2.0V	IOL = 20mA		0.3	0.4	Volts
				IOL = 48mA		.4	0.5	
IOS	Output Short Circuit Current		CD = VIL MAX, T/$\overline{R}$ = 2.0V, VO = 0V, VCC = MAX, Note 2		−25	−50	−150	mA
IIH	Logical "1" Input Current		CD = VIL MAX, T/$\overline{R}$ = VIL MAX, VI = 2.7V			0.1	80	μA
II	Input Current at Maximum Input Voltage		CD = 2.0V, VCC = MAX, VI = VCC MAX				1	mA
IIL	Logical "0" Input Current		CD = VIL MAX, T/$\overline{R}$ = VIL MAX, VI = 0.4V			−70	−200	μA
VC	Input Clamp Voltage		CD = 2.0V, IIN = −12mA			−0.7	−1.5	Volts
IOD	Output/Input 3-State Current		CD = 2.0V	VO = 0.4V			−200	μA
				VO = 4.0V			200	
			CONTROL INPUTS CD, T/$\overline{R}$					
VIH	Logical "1" Input Voltage				2.0			Volts
VIL	Logical "0" Input Voltage			COM'L			0.8	Volts
				MIL			0.7	
IIH	Logical "1" Input Current		VI = 2.7V			0.5	20	μA
II	Input Current at Maximum Input Voltage		VCC = MAX, VI = VCC MAX				1.0	mA
IIL	Logical "0" Input Current		VI = 0.4V	T/$\overline{R}$		−0.1	−.25	mA
				CD		−0.1	−0.25	
VC	Input Clamp Voltage		IIN = −12mA			−0.8	−1.5	Volts
			POWER SUPPLY CURRENT					
ICC	Power Supply Current	AmZ8103	CD =, VI = 2.0V, VCC = MAX			70	100	mA
			CD = 0.4V, VINA = T/$\overline{R}$ = 2V, VCC = MAX			100	150	
		AmZ8104	CD = 2.0V, VI = 0.4V, VCC = MAX			70	100	
			CD = VINA = 0.4V, T/$\overline{R}$ = 2V, VCC = MAX			90	140	

4

AmZ8103

AC ELECTRICAL CHARACTERISTICS (VCC = 5.0V, T_A = 25°C)

Parameters	Description	Test Conditions	Min	Typ (Note 1)	Max	Units
A PORT DATA/MODE SPECIFICATIONS						
tPDHLA	Propagation Delay to a Logical "0" from B Port to A Port	CD = 0.4V, T/$\overline{R}$ = 0.4V (Figure 1) R1 = 1k, R2 = 5k, C1 = 30pF		8	12	ns
tPDLHA	Propagation Delay to a Logical "1" from B Port to A Port	CD = 0.4V, T/$\overline{R}$ = 0.4V (Figure 1) R1 = 1k, R2 = 5k, C1 = 30pF		11	16	ns
tPLZA	Propagation Delay from a Logical "0" to 3-State from CD to A Port	B0 to B7 = 2.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 1, R5 = 1k, C4 = 15pF		10	15	ns
tPHZA	Propagation Delay from a Logical "1" to 3-State from CD to A Port	B0 to B7 = 0.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 0, R5 = 1k, C4 = 15pF		8	15	ns
tPZLA	Propagation Delay from 3-State to a Logical "0" from CD to A Port	B0 to B7 = 2.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 1, R5 = 1k, C4 = 30pF		20	30	ns
tPZHA	Propagation Delay from 3-State to a Logical "1" from CD to A Port	B0 to B7 = 0.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 0, R5 = 5k, C4 = 30pF		19	30	ns
B PORT DATA/MODE SPECIFICATIONS						
tPDHLB	Propagation Delay to a Logical "0" from A Port to B Port	CD = 0.4V, T/$\overline{R}$ = 2.4V (Figure 1) R1 = 100Ω, R2 = 1k, C1 = 300pF R1 = 667Ω, R2 = 5k, C1 = 45pF		12 7	18 12	ns
tPDLHB	Propagation Delay to a Logical "1" from A Port to B Port	CD = 0.4V, T/$\overline{R}$ = 2.4V (Figure 1) R1 = 100Ω, R2 = 1k, C1 = 300pF R1 = 667Ω, R2 = 5k, C1 = 45pF		15 9	20 14	ns
tPLZB	Propagation Delay from a Logical "0" to 3-State from CD to B Port	A0 to A7 = 2.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 1, R5 = 1k, C4 = 15pF		13	18	ns
tPHZB	Propagation Delay from A Logical "1" to 3-State from CD to B Port	A0 to A7 = 0.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 0, R5 = 1k, C4 = 15pF		8	15	ns
tPZLB	Propagation Delay from 3-State to a Logical "0" from CD to B Port	A0 to A7 = 2.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 1, R5 = 100Ω, C4 = 300pF S3 = 1, R5 = 667Ω, C4 = 45pF		25 16	35 25	ns
tPZHB	Propagation Delay from 3-State to a Logical "1" from CD to B Port	A0 to A7 = 0.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 0, R5 = 1k, C4 = 300pF S3 = 0, R5 = 5kΩ, C4 = 45pF		22 14	35 25	ns
TRANSMIT RECEIVE MODE SPECIFICATIONS						
tTRL	Propagation Delay from Transmit Mode to Receive a Logical "0," T/R to A Port	CD = 0.4V (Figure 2) S1 = 1, R4 = 100Ω, C3 = 5pF S2 = 1, R3 = 1k, C2 = 30pF		23	35	ns
tTRH	Propagation Delay from Transmit Mode to Receive a Logical "1," T/R to A Port	CD = 0.4V (Figure 2) S1 = 0, R4 = 100Ω, C3 = 5pF S2 = 0, R3 = 5k, C2 = 30pF		22	35	ns
tRTL	Propagation Delay from Receive Mode to Transmit a Logical "0," T/R to B Port	CD = 0.4V (Figure 2) S1 = 1, R4 = 100Ω, C3 = 300pF S2 = 1, R3 = 300Ω, C2 = 5pF		26	35	ns
tRTH	Propagation Delay from Receive Mode to Transmit a Logical "1," T/R to B Port	CD = 0.4V (Figure 2) S1 = 0, R4 = 1k, C3 = 300pF S2 = 0, R3 = 300Ω, C2 = 5pF		27	35	ns

Notes: 1. All typical values given are for VCC = 5.0V and T_A = 25°C.
2. Only one output at a time should be shorted.

FUNCTIONAL TABLE

Inputs	Conditions		
Chip Disable	0	0	1
Transmit/$\overline{\text{Receive}}$	0	1	X
A Port	Out	In	HI-Z
B Port	In	Out	HI-Z

AmZ8104

AC ELECTRICAL CHARACTERISTICS (VCC = 5.0V, T_A = 25°C)

Parameters	Description	Test Conditions	Min	Typ (Note 1)	Max	Units
A PORT DATA/MODE SPECIFICATIONS						
tPDHLA	Propagation Delay to a Logical "0" from B Port to A Port	CD = 0.4V, T/$\overline{R}$ = 0.4V (Figure 1) R1 = 1k, R2 = 5k, C1 = 30pF		14	18	ns
tPDLHA	Propagation Delay to a Logical "1" from B Port to A Port	CD =0.4V, T/$\overline{R}$ = 0.4V (Figure 1) R1 = 1k, R2 = 5k, C1 = 30pF		13	18	ns
tPLZA	Propagation Delay from a Logical "0" to 3-State from CD to A Port	B0 to B7 = 0.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 1, R5 = 1k, C4 = 15pF		11	15	ns
tPHZA	Propagation Delay from a Logical "1" to 3-State from CD to A Port	B0 to B7 = 2.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 0, R5 = 1k, C4 = 15pF		8	15	ns
tPZLA	Propagation Delay from 3-State to a Logical "0" from CD to A Port	B0 to B7 = 0.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 1, R5 = 1k, C4 = 30pF		27	35	ns
tPZHA	Propagation Delay from 3-State to a Logical "1" from CD to A Port	B0 to B7 = 2.4V, T/$\overline{R}$ = 0.4V (Figure 3) S3 = 0, R5 = 5k, C4 = 30pF		19	25	ns
B PORT DATA/MODE SPECIFICATIONS						
tPDHLB	Propagation Delay to Logical "0" from A Port to B Port	CD = 0.4V, T/$\overline{R}$ = 2.4V (Figure 1) R1 = 100Ω, R2 = 1k, C1 = 300pF		18	23	ns
		R1 = 667Ω, R2 = 5k, C1 = 45pF		11	18	
tPDLHB	Propagation Delay to Logical "1" from A Port to B Port	CD = 0.4V, T/$\overline{R}$ = 2.4V (Figure 1) R1 = 100Ω, R2 = 1k, C1 = 300pF		16	23	ns
		R1 = 667Ω, R2 = 5k, C1 = 45pF		11	18	
tPLZB	Propagation Delay from a Logical "0" to 3-State from CD to B Port	A0 to A7 = 0.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 1, R5 = 1k, C4 = 15pF		13	18	ns
tPHZB	Propagation Delay from a Logical "1" to 3-State from CD to B Port	A0 to A7 = 2.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 0, R5 = 1k, C4 = 15pF		8	15	ns
tPZLB	Propagation Delay from 3-State to a Logical "0" from CD to B Port	A0 to A7 = 0.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 1, R5 = 100Ω, C4 = 300pF		32	40	ns
		S3 = 1, R5 = 667Ω, C4 = 45pF		16	22	
tPZHB	Propagation Delay from 3-State to a Logical "1" from CD to B Port	A0 to A7 = 2.4V, T/$\overline{R}$ = 2.4V (Figure 3) S3 = 0, R5 = 1k, C4 = 300pF		26	35	ns
		S3 = 0, R5 = 5kΩ, C4 = 45pF		14	22	
TRANSMIT RECEIVE MODE SPECIFICATIONS						
tTRL	Propagation Delay from Transmit Mode to Receive a Logical "0," T/$\overline{R}$ to A Port	CD = 0.4V (Figure 2) S1 = 0, R4 = 100Ω, C3 = 5pF S2 = 1, R3 = 1k, C2 = 30pF		30	40	ns
tTRH	Propagation Delay from Transmit Mode to Receive a Logical "1," T/$\overline{R}$ to A Port	CD = 0.4V (Figure 2) S1 = 1, R4 = 100Ω, C3 =5pF S2 = 0, R3 = 5k, C2 = 30pF		28	40	ns
tRTL	Propagation Delay from Receive Mode to Transmit a Logical "0," T/$\overline{R}$ to B Port	CD = 0.4V (Figure 2) S1 = 1, R4 = 100Ω, C3 = 300pF S2 = 0, R3 = 300Ω, C2 = 5pF		31	40	ns
tRTH	Propagation Delay from Receive Mode to Transmit a Logical "1," T/$\overline{R}$ to B Port	CD = 0.4V (Figure 2) S1 = 0, R4 = 1k, C3 = 300pF S2 = 1, R3 = 300Ω, C2 = 5pF		28	40	ns

Notes: 1. All typical values given are for VCC = 5.0V and T_A = 25°C.
2. Only one output at a time should be shorted.

DEFINITION OF FUNCTIONAL TERMS

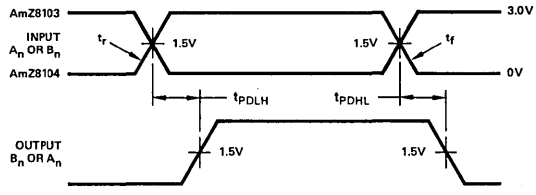
A0-A7 A port inputs/outputs are receiver output drivers when T/$\overline{R}$ is LOW and are transmit inputs when T/$\overline{R}$ is HIGH.

B0-B7 B port inputs/outputs are transmit output drivers when T/$\overline{R}$ is HIGH and receiver inputs when T/$\overline{R}$ is LOW.

CD Chip Disable forces all output drivers into 3-state when HIGH (same function as active LOW chip select, $\overline{CS}$).

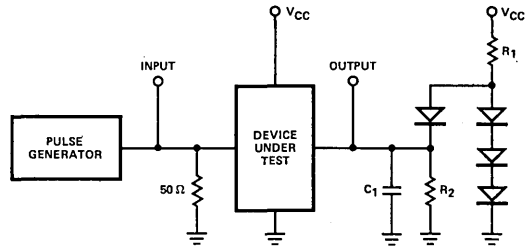
T/$\overline{R}$ Transmit/$\overline{Receive}$ direction control determines whether A port or B port drivers are in 3-state. With T/$\overline{R}$ HIGH A port is the input and B port is the output. With T/$\overline{R}$ LOW A port is the output and B port is the input.

SWITCHING TIME WAVEFORMS AND AC TEST CIRCUITS



$t_r = t_f < 10ns$
10% to 90%

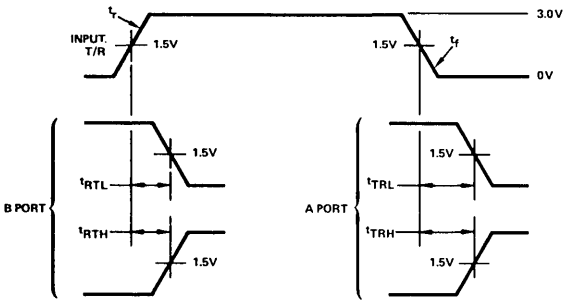
BLI-171



Note: C_1 includes test fixture capacitance.

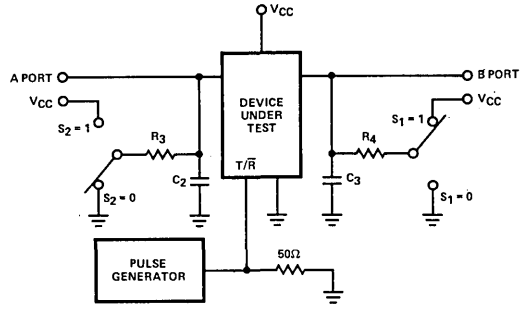
BLI-172

Figure 1. Propagation Delay from A Port to B Port or from B Port to A Port.



$t_r = t_f < 10ns$
10% to 90%

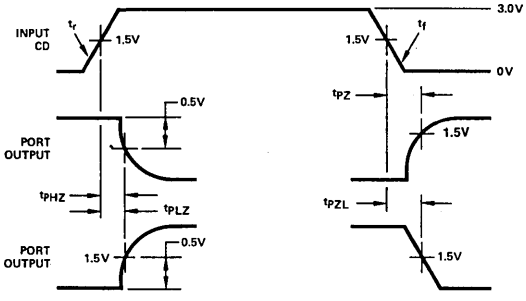
BLI-173



Note: C_2 and C_3 include test fixture capacitance.

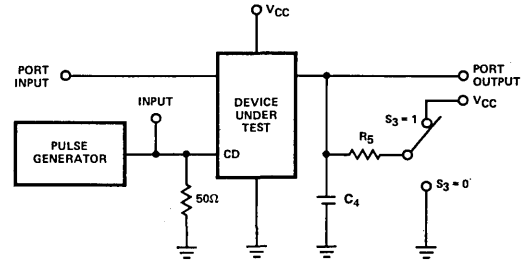
BLI-174

Figure 2. Propagation Delay from T/$\overline{R}$ to A Port or B Port.



$t_r = t_f < 10ns$
10% to 90%

BLI-175



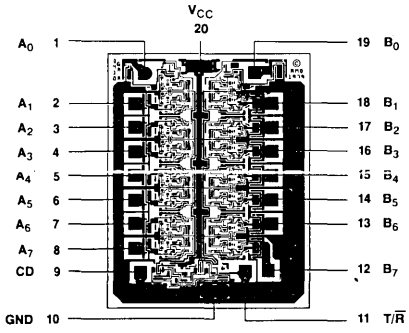
Note: C_4 includes test fixture capacitance.
Port input is in a fixed logical condition.

BLI-176

Figure 3. Propagation Delay from CD to A Port or B Port.

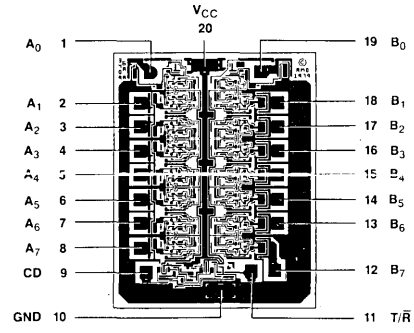
Metallization and Pad Layouts

AmZ8103



DIE SIZE .069" X .089"

AmZ8104



DIE SIZE .069" X .089"

4

ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range and screening level.

AmZ8103 Order Number	AmZ8104 Order Number	Package Type (Note 1)	Operating Range (Note 2)	Screening Level (Note 3)
AMZ8103DC	AMZ8104DC	D-20	C	C-1
AMZ8103DCB	AMZ8104DCB	D-20	C	B-1
AMZ8103DM	AMZ8104DM	D-20	M	C-3
AMZ8103DMB	AMZ8104DMB	D-20	M	B-3
AMZ8103PC	AMZ8104PC	P-20	C	C-1
AMZ8103PCB	AMZ8104PCB	P-20	C	C-1

Notes: 1. P = Molded DIP, D = Hermetic DIP, F = Flatpak. Number following letter is number of leads.
2. C = 0 to 70°C, VCC = 4.75V to 5.25V, M = −55 to +125°C, VCC = 4.50V to 5.50V.
3. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.

AmZ8107 • AmZ8108

Octal Three-State Bidirectional Bus Transceivers

DISTINCTIVE CHARACTERISTICS

- 8-bit bidirectional data flow reduces system package count
- 3-state inputs/outputs for interfacing with bus-oriented systems
- PNP inputs reduce input loading
- VCC – 1.15V VOH interfaces with TTL, MOS, and CMOS
- 48mA, 300pF bus drive capability
- AmZ8107 has inverting tranceivers
- AmZ8108 has non-inverting transceivers
- Separate $\overline{\text{TRANSMIT}}$ and $\overline{\text{RECEIVE}}$ Enables
- 20 pin ceramic and molded DIP package
- Low power – 8mA per bidirectional bit
- Advanced Schottky processing
- Bus port stays in hi-impedance state during power up/down
- 100% product assurance screening to MIL-STD-883 requirements

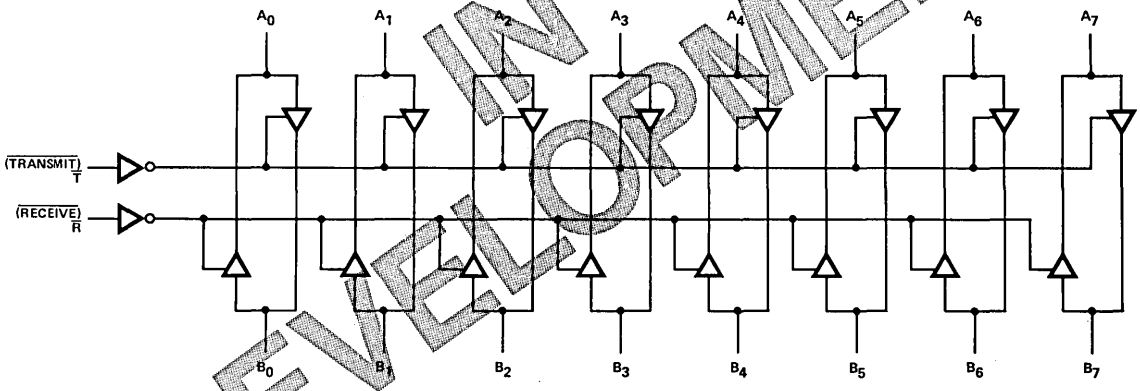
GENERAL DESCRIPTION

The AmZ8107 and AmZ8108 are 8-bit, 3-state Schottky transceivers. They provide bidirectional drive for bus-oriented microprocessor and digital communications systems. Straight through bidirectional transceivers are featured, with 24mA drive capability on the A ports and 48mA bus drive capability on the B ports. PNP inputs are incorporated to reduce input loading.

Separate $\overline{\text{TRANSMIT}}$ and $\overline{\text{RECEIVE}}$ Enables are provided for microprocessor system with separated read and write control bus lines.

The output high voltage (VOH) is specified at VCC – 1.15V minimum to allow interfacing with MOS, CMOS, TTL, ROM, RAM, or microprocessors.

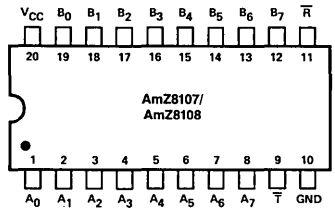
AmZ8108 LOGIC DIAGRAM



AmZ8107 has inverting transceivers

BLI-177

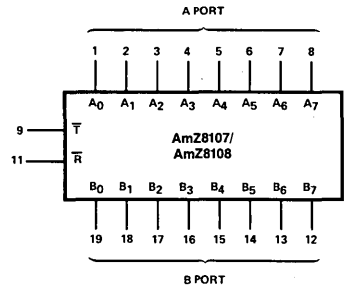
CONNECTION DIAGRAM Top View



Note: Pin 1 is marked for orientation.
AmZ8107 is inverting from Ai to Bi

BLI-178

LOGIC SYMBOL



VCC = Pin 20
GND = Pin 10

BLI-179

AmZ8120

Octal D-Type Flip-Flop with Clear, Clock Enable and 3-State Control

DISTINCTIVE CHARACTERISTICS

- Buffered common clock enable input
- Buffered common asynchronous clear input
- Three-state outputs
- 8-bit, high-speed parallel register with positive edge-triggered, D-type flip-flops
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

The AmZ8120 is an 8-bit register built using advanced Low-Power Schottky technology. The register consists of eight D-type flip-flops with a buffered common clock, a buffered common clock enable, a buffered asynchronous clear input, and three-state outputs.

When the clear input is LOW, the internal flip-flops of the register are reset to logic 0 (LOW), independent of all other inputs. When the clear input is HIGH, the register operates in the normal fashion.

When the three-state output enable ($\overline{OE}$) input is LOW, the Y outputs are enabled and appear as normal TTL outputs. When the output enable ($\overline{OE}$) input is HIGH, the Y outputs are in the high impedance (three-state) condition. This does not affect the internal state of the flip-flop Q output.

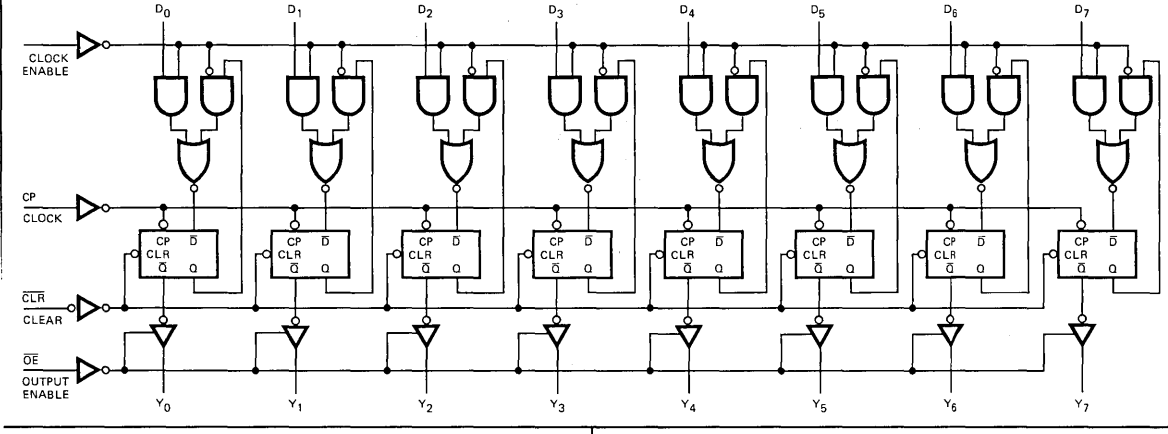
The clock enable input ($\overline{E}$) is used to selectively load data into the register. When the $\overline{E}$ input is HIGH, the register will retain its current data. When the $\overline{E}$ is LOW, new data is entered into the register on the LOW-to-HIGH transition of the clock input.

This device is packaged in a slim 24-pin package (0.3 inch row spacing).

4

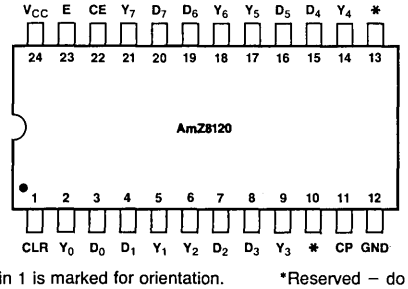
BLI-180

LOGIC DIAGRAM



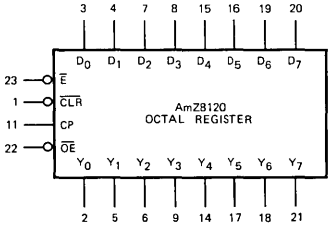
BLI-181

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation. *Reserved – do not use.

LOGIC SYMBOL



V_{CC} = Pin 24
GND = Pin 12

BLI-182

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L	T_A = 0°C to +70°C	V_{CC} = 5.0V ±5%	MIN. = 4.75V	MAX. = 5.25V
MIL	T_A = −55°C to +125°C	V_{CC} = 5.0V ±10%	MIN. = 4.50V	MAX. = 5.50V

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)		Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN., V_{IN} = V_{IH} or V_{IL}	MIL, I_{OH} = −1.0mA	2.4	3.4		Volts
			COM'L, I_{OH} = −2.6mA	2.4	3.4		
V_{OL}	Output LOW Voltage	V_{CC} = MIN., V_{IN} = V_{IH} or V_{IL}	I_{OL} = 4.0mA			0.4	Volts
			I_{OL} = 8.0mA			0.45	
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs	MIL			0.7	Volts
			COM'L			0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA				−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.4V				−0.36	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V				20	µA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 7.0V				0.1	mA
I_O	Off-State (High-Impedance) Output Current	V_{CC} = MAX.	V_O = 0.4V			−20	µA
			V_O = 2.4V			20	
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX.		−15		−85	mA
I_{CC}	Power Supply Current (Note 4)	V_{CC} = MAX.			24	37	mA

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. All outputs open, $\overline{E}$ = GND, DI inputs = CLR = $\overline{OE}$ = 4.5V. Apply momentary ground, then 4.5V to clock input.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to +V_{CC} max.
DC Input Voltage	−0.5V to +7.0V
DC Output Current, Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

SWITCHING CHARACTERISTICS

(T_A = +25°C, V_{CC} = 5.0V)

Parameters	Description		Min.	Typ.	Max.	Units	Test Conditions
t_{PLH}	Clock to Y_i ($\overline{OE}$ LOW)			18	27	ns	C_L = 15pF, R_L = 2.0kΩ
t_{PHL}				24	36		
t_{PHL}	Clear to Y			22	35	ns	
t_s	Data (D_i)		10	3		ns	
t_h	Data (D_i)		10	3		ns	
t_s	Enable ($\overline{E}$)	Active	15	10		ns	
		Inactive	20	12			
t_h	Enable ($\overline{E}$)		0	0		ns	
t_s	Clear Recovery (In-Active) to Clock		11	7		ns	
t_{pw}	Clock	HIGH	20	14		ns	
		LOW	25	13			
t_{pw}	Clear		20	13		ns	
t_{ZH}	$\overline{OE}$ to Y_i			9	13	ns	
t_{ZL}				14	21		
t_{HZ}	$\overline{OE}$ to Y_i			20	30	ns	C_L = 5.0pF, R_L = 2.0kΩ
t_{LZ}				24	36		
f_{max}	Maximum Clock Frequency (Note 1)			40		MHz	

Note 1. Per industry convention, f_{max} is the worst case value of the maximum device operating frequency with no constraints on t_r, t_f, pulse width or duty cycle.

4

SWITCHING CHARACTERISTICS OVER OPERATING RANGE*

Parameters	Description		COM'L T_A = 0°C to +70°C, V_{CC} = 5.0V ± 5% Min.	COM'L Max.	MIL T_A = −55°C to +125°C, V_{CC} = 5.0V ± 10% Min.	MIL Max.	Units	Test Conditions
t_{PLH}	Clock to Y_i ($\overline{OE}$ LOW)			33		39	ns	C_L = 50pF, R_L = 2.0kΩ
t_{PHL}				45		54		
t_{PHL}	Clear to Y			43		51	ns	
t_s	Data (D_i)		12		15		ns	
t_h	Data (D_i)		12		15		ns	
t_s	Enable ($\overline{E}$)	Active	17		20		ns	
		Inactive	20		23			
t_h	Enable ($\overline{E}$)		0		0		ns	
t_s	Clear Recovery (In-Active) to Clock		13		15		ns	
t_{pw}	Clock	HIGH	25		30		ns	
		LOW	30		35			
t_{pw}	Clear		22		25		ns	
t_{ZH}	$\overline{OE}$ to Y_i			19		25	ns	
t_{ZL}				30		39		
t_{HZ}	$\overline{OE}$ to Y_i			35		40	ns	C_L = 5.0 pF, R_L = 2.0 kΩ
t_{LZ}				39		42		
f_{max}	Maximum Clock Frequency (Note 1)		25		20		MHz	

*AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.

DEFINITION OF FUNCTIONAL TERMS

D_i The D flip-flop data inputs.

$\overline{CLR}$ When the clear input is LOW, the Q_i outputs are LOW, regardless of the other inputs. When the clear input is HIGH, data can be entered into the register.

CP Clock Pulse for the Register; enters data into the register on the LOW-to-HIGH transition.

Y_i The register three-state outputs.

$\overline{E}$ Clock Enable, When the clock enable is LOW, data on the D_i input is transferred to the Q_i output on the LOW-to-HIGH clock transition. When the clock enable is HIGH, the Q_i outputs do not change state, regardless of the data or clock input transitions.

$\overline{OE}$ Output Control. When the $\overline{OE}$ input is HIGH, the Y_i outputs are in the high impedance state. When the $\overline{OE}$ input is LOW, the TRUE register data is present at the Y_i outputs.

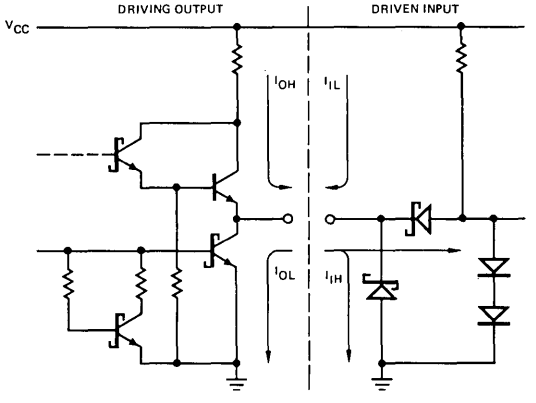
FUNCTION TABLE

	Inputs					Internal	Outputs
Function	$\overline{OE}$	$\overline{CLR}$	$\overline{E}$	D_i	CP	Q_i	Y_i
Hi-Z	H	X	X	X	X	X	Z
Clear	H	L	X	X	X	L	Z
	L	L	X	X	X	L	L
Hold	H	H	H	X	X	NC	Z
	L	H	H	X	X	NC	NC
Load	H	H	L	L	↑	L	Z
	H	H	L	H	↑	H	Z
	L	H	L	L	↑	L	L
	L	H	L	H	↑	H	H

H = HIGH
L = LOW
X = Don't Care

NC = No Change
↑ = LOW-to-HIGH Transition
Z = High Impedance

LOW-POWER SCHOTTKY INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



Note: Actual current flow direction shown.

BLI-183

AmZ8121

Eight-Bit Equal-To Comparator

DISTINCTIVE CHARACTERISTICS

- 8-bit byte oriented equal-to comparator
- Cascadable using $\overline{E}_{IN}$
- High-speed, Low-Power Schottky technology
- t_{pd} A⊙B to $\overline{E}_{OUT}$ in 9ns
- Standard 20-pin package
- 100% product assurance screening to MIL-STD-883 requirements

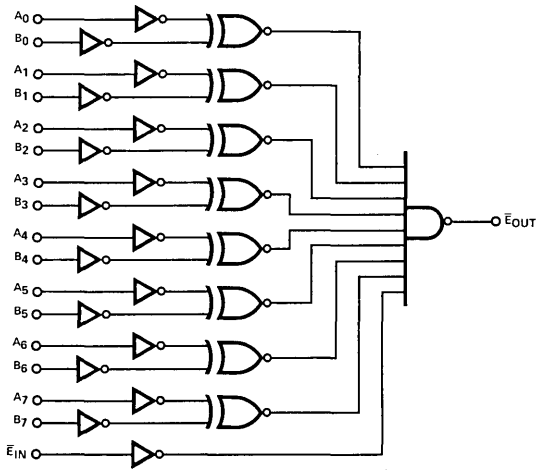
FUNCTIONAL DESCRIPTION

The AmZ8121 is an 8-bit "equal to" comparator capable to comparing two 8-bit words for "equal to" with provision for expansion or external enabling. The matching of the two 8-bit inputs plus a logic LOW on the $\overline{E}_{IN}$ produces an active LOW on the output $\overline{E}_{OUT}$.

The logic expression for the device can be expressed as: $\overline{E}_{OUT} = \overline{(A_0 \odot B_0)\ (A_1 \odot B_1)\ (A_2 \odot B_2)\ (A_3 \odot B_3)\ (A_4 \odot B_4)\ (A_5 \odot B_5)\ (A_7 \odot B_7)\ E_{IN}}$. It is obvious that the expression is valid where $A_0 - A_7$ and $B_0 - B_7$ are expressed as either assertions or negations. This is also true for pair of terms i.e. A_0 can be compared with B_0 at the same time $\overline{A}_1$ is compared with $\overline{B}_1$. It is only essential that the polarity of the paired terms be maintained.

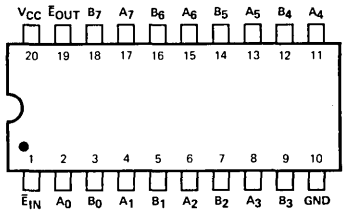
4

LOGIC DIAGRAM



BLI-184

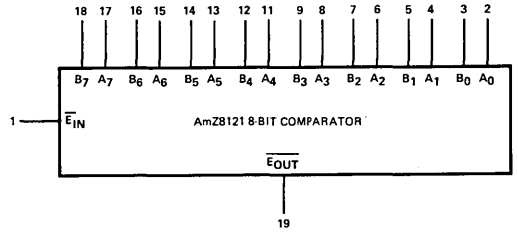
CONNECTION DIAGRAM



Note: Pin 1 is marked for orientation

BLI-185

LOGIC SYMBOL



V_{CC} = Pin 20
GND = Pin 10

BLI-186

AmZ8121
ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L $T_A = 0°C$ to $+70°C$ $V_{CC} = 5.0V \pm 5\%$ MIN. = 4.75 V MAX. = 5.25 V
MIL $T_A = -55°C$ to $+125°C$ $V_{CC} = 5.0V \pm 10\%$ MIN. = 4.50 V MAX. = 5.50 V

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)		Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN., $V_{IN} = V_{IH}$ or V_{IL}, $I_{OH} = -440\mu A$	MIL	2.5			Volts
			COM'L	2.7			
V_{OL}	Output LOW Voltage	V_{CC} = MIN., $V_{IN} = V_{IH}$ or V_{IL}, $I_{OL} = 4.0mA$				0.4	Volts
			$I_{OL} = 8.0mA$			0.45	
			$I_{OL} = 12mA$			0.5	
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs	MIL			0.7	Volts
			COM'L			0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., $I_{IN} = -18mA$				−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., $V_{IN} = 0.4V$	A_i, B_i			−0.36	mA
			$\overline{E}$			−0.72	
I_{IH}	Input HIGH Current	V_{CC} = MAX., $V_{IN} = 2.7V$	A_i, B_i			20	μA
			$\overline{E}$			40	
I_I	Input HIGH Current	V_{CC} = MAX., $V_{IN} = 7.0V$	A_i, B_i			0.1	mA
			$\overline{E}$			0.2	
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX.		−15		−85	mA
I_{CC}	Power Supply Current (Note 4)	V_{CC} = MAX.			27	40	mA

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at $V_{CC} = 5.0V$, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. $\overline{E}$ = GND, all other inputs and outputs open.

AmZ8121
MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to $+V_{CC}$ max
DC Input Voltage	−0.5V to +7.0V
DC Output Current, Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

SWITCHING CHARACTERISTICS
(T_A = +25°C, V_{CC} = 5.0V)

Parameters	Description	Min.	Typ.	Max.	Units	Test Conditions
t_{PLH}	A_i or B_i to $\overline{Equal}$		9	15	ns	C_L = 15pF, R_L = 2.0kΩ
t_{PHL}			9	15		
t_{PLH}	$\overline{E}$ to $\overline{Equal}$		5	7	ns	
t_{PHL}			6	8		

SWITCHING CHARACTERISTICS OVER OPERATING RANGE*

Parameters	Description	COM'L T_A = 0°C to +70°C, V_{CC} = 5.0V ±5% Min.	COM'L Max.	MIL T_A = −55°C to +125°C, V_{CC} = 5.0V ±10% Min.	MIL Max.	Units	Test Conditions
t_{PLH}	A_i or B_i to $\overline{Equal}$ Output		20		22	ns	C_L = 50pF, R_L = 2.0kΩ
t_{PHL}			19		21		
t_{PLH}	$\overline{E}$ to $\overline{Equal}$ Output		10.5		12	ns	
t_{PHL}			12.5		15		

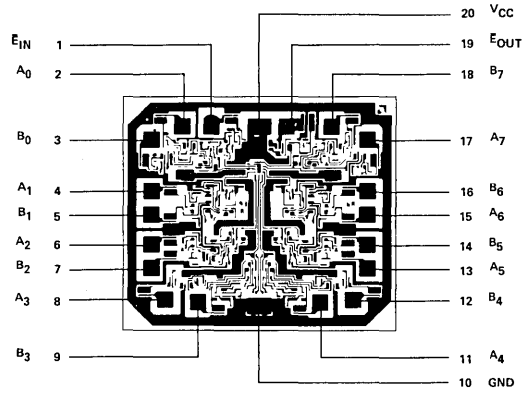
*AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.

4

DEFINITION OF FUNCTIONAL TERMS

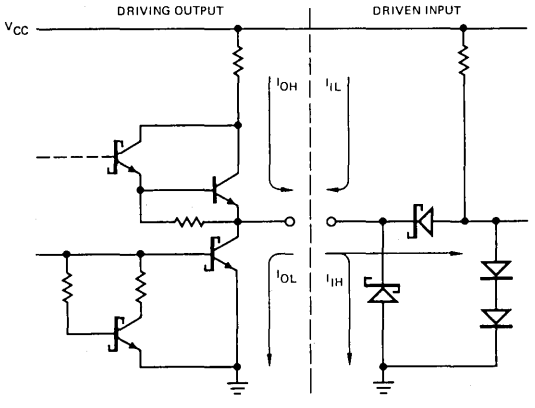
A_0–A_7	A input to comparator
B_0–B_7	B input to comparator
$\overline{E}_{IN}$	Enable active LOW
$\overline{E}_{OUT}$	$\overline{EQUAL}$ output active LOW

METALLIZATION AND PAD LAYOUT



DIE SIZE 0.063" x 0.074"

LOW-POWER SCHOTTKY INPUT/OUTPUT CURRENT INTERFACE CONDITIONS

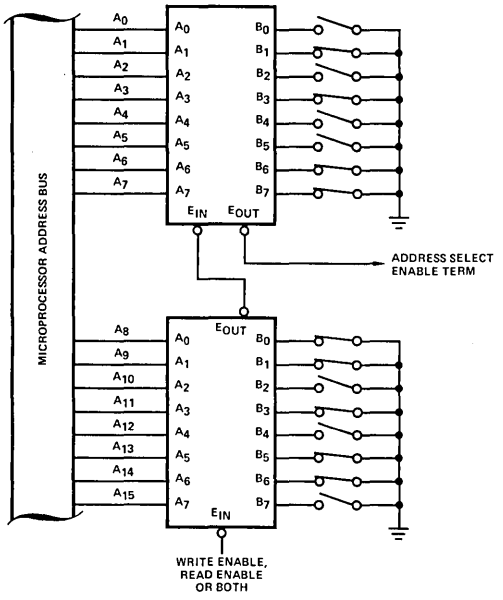


Note: Actual current flow direction shown.

BLI-187

APPLICATIONS

MICROPROCESSOR ENABLE CONTROLLED, SELECTABLE, ADDRESS DECODER

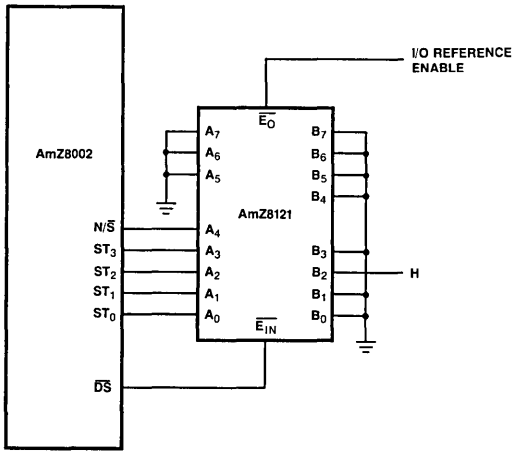


MAX. $\overline{\text{ENABLE}}$ (HIGH-to-LOW) DELAY OVER 16-BITS
(Commercial Range)

t_{PHL}	A_i or B_i to $\overline{E}_{OUT}$	19ns
t_{PHL}	$\overline{E}_{IN}$ to $\overline{E}_{OUT}$	12.5ns
Total		31.5ns

BLI-188

STATUS LINE DECODING



BLI-189

AmZ8127

AmZ8000 Clock Generator and Controller

Advanced Information

DISTINCTIVE CHARACTERISTICS

- **High-drive high-level clock output**
  Special output provides clock signal matched to requirements of AmZ8000 CPU, MMU and DMA devices.
- **Four TTL-level clocks**
  Generates synchronized TTL compatible clocks at 16MHz, 2MHz and 1MHz to drive memory circuits and LSI peripheral devices. An additional TTL clock is synchronized with the high-level clock for registers, latches and other peripherals.
- **Synchronized WAIT state and time-out controls**
  On-chip logic generates WAIT signal under control of Halt, Single-step and Ready signals. Automatic time-out of peripheral wait requests.

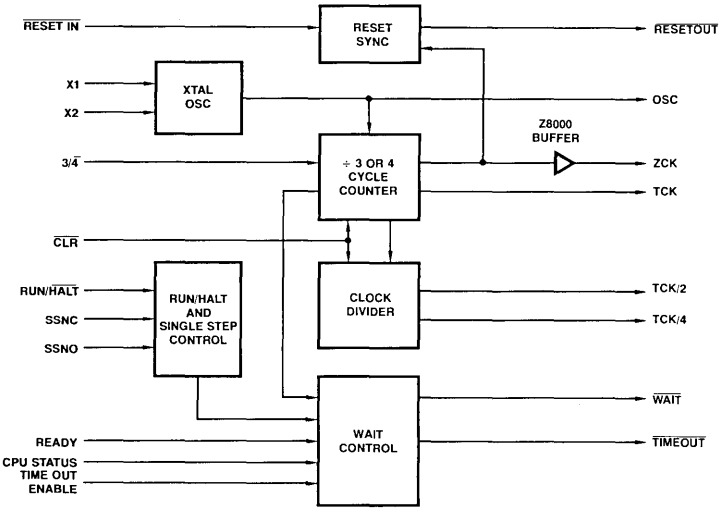
FUNCTIONAL DESCRIPTION

The AmZ8127 Clock Generator and Controller provides the clock oscillator, frequency dividers and clock drivers for the complete array of AmZ8000 CPUs, peripherals and memory system configurations. In addition to the special 4MHz output driver for the AmZ8001 and AmZ8002 CPUs, a standard buffered TTL 16MHz oscillator output is provided for dynamic memory timing and control. The AmZ8127 forms an integral part of the dynamic memory support chip set including the AmZ8163 EDC and Refresh Controller, AmZ8164 Dynamic Memory Controller, AmZ8160 Error Detection and Correction Unit and AmZ8161/AmZ8162 EDC Bus Buffers. The oscillator is designed to operate with a 16MHz crystal or with external 16MHz drive. The AmZ8127 uses an internal divide-by-4 to provide 4MHz clock drive to the AmZ8001/AmZ8002 CPU. Additional dividers generate synchronous buffered 2MHz and 1MHz clock outputs for use by peripheral devices. The clock divider counters are clearable to allow synchronizing the multiple clock outputs.

The controller functions include $\overline{\text{RESET}}$, RUN/$\overline{\text{HALT}}$, SINGLE-STEP, READY and a READY TIMEOUT counter which limits a peripheral's wait request to 16 clock cycles. The CPU's $\overline{\text{WAIT}}$ input is controlled by RUN/$\overline{\text{HALT}}$, SINGLE-STEP and READY. A $\overline{\text{HALT}}$ command to the AmZ8127 drives the $\overline{\text{WAIT}}$ output LOW causing the CPU to add wait states (TW to TW). The READY input is used by peripherals to request wait states. The active HIGH input TIMEOUT ENABLE is used to force TIMEOUT and $\overline{\text{WAIT}}$ to HIGH 16 clock cycles after a peripheral has requested a wait but fails to release the request. The CPU status lines ST1, ST2 and ST3 are decoded in the AmZ8127 to disable the TIME-OUT counter during CPU "Internal Operations" and during refresh.

BLOCK DIAGRAM

CLOCK GENERATOR



BLI-168

AmZ8133 • AmZ8173

Octal Latches with Three-State Outputs

DISTINCTIVE CHARACTERISTICS

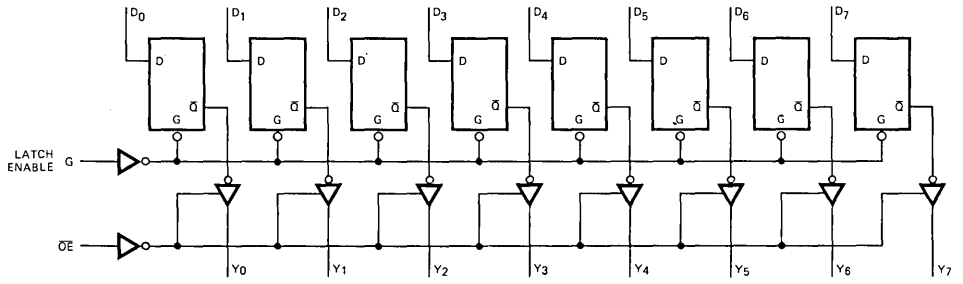
- 18ns max data in to data out
- Non-inverting AmZ8173, inverting AmZ8133
- Three-state outputs interface directly with bus organized systems
- Hysteresis on latch enable input for improved noise margin
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

The AmZ8133 and AmZ8173 are octal latches with three-state outputs for bus organized system applications. The latches appear to be transparent to the data (data changes asynchronously) when latch enable, G, is HIGH. When G is LOW, the data that meets the set-up times is latched. Data appears on the bus when the output enable, $\overline{OE}$, is LOW. When $\overline{OE}$ is HIGH the bus output is in the high-impedance state.

The AmZ8173 has non-inverted data inputs while the AmZ8133 is inverting.

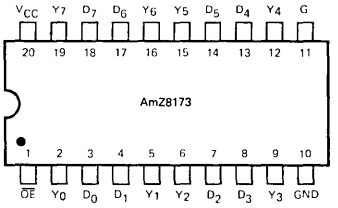
LOGIC DIAGRAM
AmZ8173

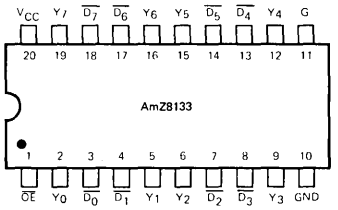


Inputs D_0 through D_7 are inverted on the AmZ8133.

BLI-041

CONNECTION DIAGRAMS
Top Views

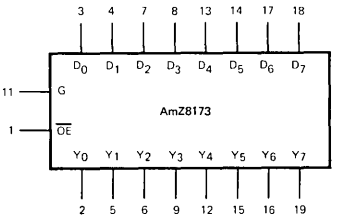




Note: Pin 1 is marked for orientation.

BLI-042

LOGIC SYMBOL



BLI-043

V_{CC} = Pin 20
GND = Pin 10

Inputs D_0 through D_7 are inverted on the AmZ8133.

AmZ8133 • AmZ8173
ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L $T_A = 0°C$ to $+70°C$ $V_{CC} = 5.0V \pm 5\%$ MIN. = 4.75 V MAX. = 5.25 V
MIL $T_A = -55°C$ to $+125°C$ $V_{CC} = 5.0V \pm 10\%$ MIN. = 4.50 V MAX. = 5.50 V

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)			Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN., $V_{IN} = V_{IH}$ or V_{IL}	$I_{OH} = -1.0mA$	MIL	2.4	3.4		Volts
			$I_{OH} = -2.6mA$	COM'L	2.4	3.4		
V_{OL}	Output LOW Voltage	V_{CC} = MIN., $V_{IN} = V_{IH}$ or V_{IL}	$I_{OL} = 12mA$				0.4	Volts
			$I_{OL} = 24mA$				0.5	
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs		MIL			0.7	Volts
				COM'L			0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., $I_{IN} = -18mA$					−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., $V_{IN} = 0.4V$					−0.4	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., $V_{IN} = 2.7V$					20	μA
I_I	Input HIGH Current	V_{CC} = MAX., $V_{IN} = 7.0V$					0.1	mA
I_{OZ}	Off-State (High-Impedance) Output Current	V_{CC} = MAX.	$V_O = 0.4V$				−20	μA
			$V_O = 2.4V$				20	
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX.			−30		−85	mA
I_{CC}	Power Supply Current (Note 4)	V_{CC} = MAX.				24	40	mA

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at $V_{CC} = 5.0V$, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. Inputs grounded; outputs open.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential Continuous	−0.5 V to +7.0 V
DC Voltage Applied to Outputs for High Output State	−0.5 V to $+V_{CC}$ max.
DC Input Voltage	−0.5 V to +7.0 V
DC Output Current, Into Outputs	30 mA
DC Input Current	−30 mA to +5.0 mA

AmZ8133
SWITCHING CHARACTERISTICS
(T_A = +25°C, V_{CC} = 5.0V)

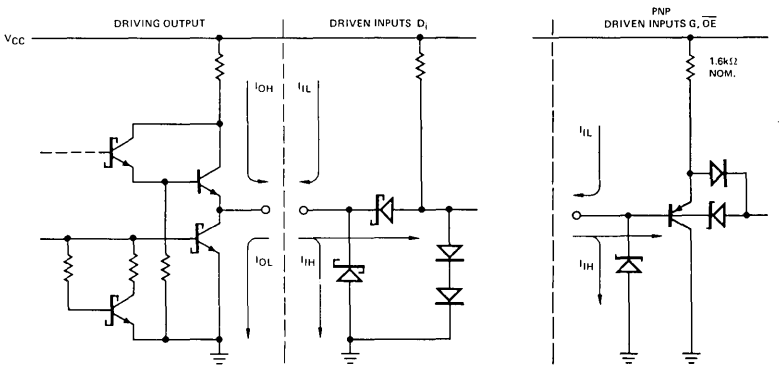
Parameters	Description	Min	Typ	Max	Units	Test Conditions
t_{PLH}	Enable to Output		20	30	ns	C_L = 45pF, R_L = 667Ω
t_{PHL}			18	30		
t_{PLH}	Data Input to Output		13	20	ns	
t_{PHL}			15	23		
t_s(H)	HIGH Data to Enable	3			ns	
t_s(L)	LOW Data to Enable	0				
t_h(H)	HIGH Data to Enable	13			ns	
t_h(L)	LOW Data to Enable	7				
t_{pw}	Enable Pulse Width	15			ns	
t_{ZH}	$\overline{OE}$ to Y_i			28	ns	
t_{ZL}				36		
t_{HZ}	$\overline{OE}$ to Y_i			20	ns	C_L = 5pF, R_L = 667Ω
t_{LZ}				25		

AmZ8133
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	COM'L T_A = 0°C to +70°C, V_{CC} = 5.0V ±5% Min.	COM'L Max.	MIL T_A = −55°C to +125°C, V_{CC} = 5.0V ±10% Min.	MIL Max.	Units	Test Conditions
t_{PLH}	Enable to Output		35		40	ns	C_L = 45pF, R_L = 667Ω
t_{PHL}			35		40		
t_{PLH}	Data Input to Output		20		21	ns	
t_{PHL}			25		30		
t_s(H)	HIGH Data to Enable	5		5		ns	
t_s(L)	LOW Data to Enable	0		0			
t_h(H)	HIGH Data to Enable	14		15		ns	
t_h(L)	LOW Data to Enable	9		10			
t_{pw}	Enable Pulse Width	17		20		ns	
t_{ZH}	$\overline{OE}$ to Y_i		28		28	ns	
t_{ZL}			36		36		
t_{HZ}	$\overline{OE}$ to Y_i		33		36	ns	C_L = 5pF, R_L = 667Ω
t_{LZ}			33		36		

*AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.

LOW-POWER SCHOTTKY INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



Note: Actual current flow direction shown.

BLI-044

AmZ8173
SWITCHING CHARACTERISTICS
(T_A = 25°C, V_{CC} = 5.0V)

Parameters	Description	Min	Typ	Max	Units	Test Conditions
t_{PLH}	Enable to Output		20	30	ns	C_L = 45pF R_L = 667Ω
t_{PHL}			18	30		
t_{PLH}	Data Input to Output		10	18	ns	
t_{PHL}			12	18		
t_s(H)	HIGH Data to Enable	0			ns	
t_s(L)	LOW Data to Enable	0				
t_h(H)	HIGH Data to Enable	10			ns	
t_h(L)	LOW Data to Enable	10				
t_{pw}	Enable Pulse Width	15			ns	
t_{ZH}	$\overline{OE}$ to Y_i			28	ns	
t_{ZL}				36		
t_{HZ}	$\overline{OE}$ to Y_i			20	ns	C_L = 5pF R_L = 667Ω
t_{LZ}				25		

4

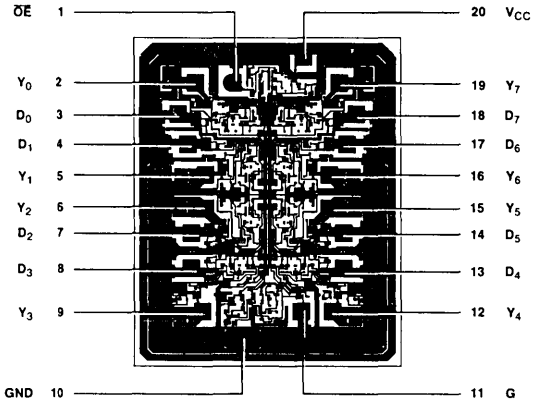
AmZ8173
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	COM'L T_A = 0°C to +70°C V_{CC} = 5.0V ±5% Min.	 Max.	MIL T_A = −55°C to +125°C V_{CC} = 5.0V ±10% Min.	 Max.	Units	Test Conditions
t_{PLH}	Enable to Output		35		40	ns	C_L = 45pF R_L = 667Ω
t_{PHL}			35		40		
t_{PLH}	Data Input to Output		19		20	ns	
t_{PHL}			20		25		
t_s(H)	HIGH Data to Enable	0		0		ns	
t_s(L)	LOW Data to Enable	0		0			
t_h(H)	HIGH Data to Enable	11		12		ns	
t_h(L)	LOW Data to Enable	15		17			
t_{pw}	Enable Pulse Width	17		20		ns	
t_{ZH}	$\overline{OE}$ to Y_i		28		28	ns	
t_{ZL}			36		36		
t_{HZ}	$\overline{OE}$ to Y_i		33		36	ns	C_L = 5pF R_L = 667Ω
t_{LZ}			33		36		

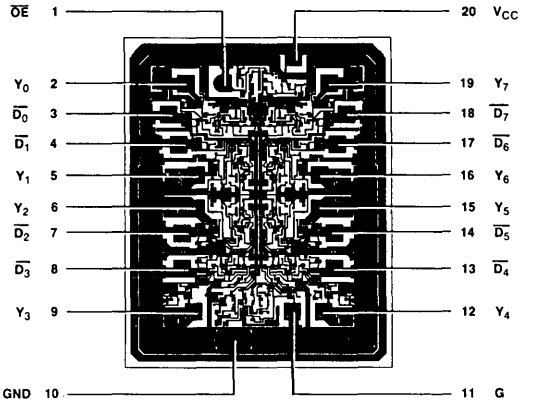
*AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.

Metallization and Pad Layouts

AmZ8173



AmZ8133



DIE SIZE 0.073" X 0.089"

FUNCTION TABLES

AmZ8173

Inputs			Internal	Outputs	Function
$\overline{OE}$	G	D_i	Q_i	Y_i	
H	X	X	X	Z	Hi-Z
L	H	L	H	L	Transparent
L	H	H	L	H	
L	L	X	NC	NC	Latched

AmZ8133

Inputs			Internal	Outputs	Function
$\overline{OE}$	G	$\overline{D_i}$	Q_i	Y_i	
H	X	X	X	Z	Hi-Z
L	H	L	H	H	Transparent
L	H	H	L	L	
L	L	X	NC	NC	Latched

H = HIGH
L = LOW
X = Don't Care

NC = No Change
Z = High Impedance

DEFINITION OF FUNCTIONAL TERMS

AmZ8173

D_i The latch data inputs.

G The latch enable input. The latches are transparent when G is HIGH. Input data is latched on the HIGH-to-LOW transition.

Y_i The three-state latch outputs.

$\overline{OE}$ The output enable control. When $\overline{OE}$ is LOW, the outputs Y_i are enabled. When $\overline{OE}$ is HIGH, the outputs Y_i are in the high-impedance (off) state.

AmZ8133

$\overline{D_i}$ The latch inverting data inputs.

G The latch enable input. The latches are transparent when G is HIGH. Input data is latched on the HIGH-to-LOW transition.

Y_i The three-state latch outputs.

$\overline{OE}$ The output enable control. When $\overline{OE}$ is LOW, the inverted outputs Y_i are enabled. When $\overline{OE}$ is HIGH, the outputs Y_i are in the high-impedance (off) state.

AmZ8136

Eight-Bit Decoder With Control Storage

DISTINCTIVE CHARACTERISTICS

- 8-bit decoder/demultiplexer with control storage
- 3-state outputs
- Common clock enable
- Common clear
- Polarity control
- Advanced Low Power Schottky Process
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

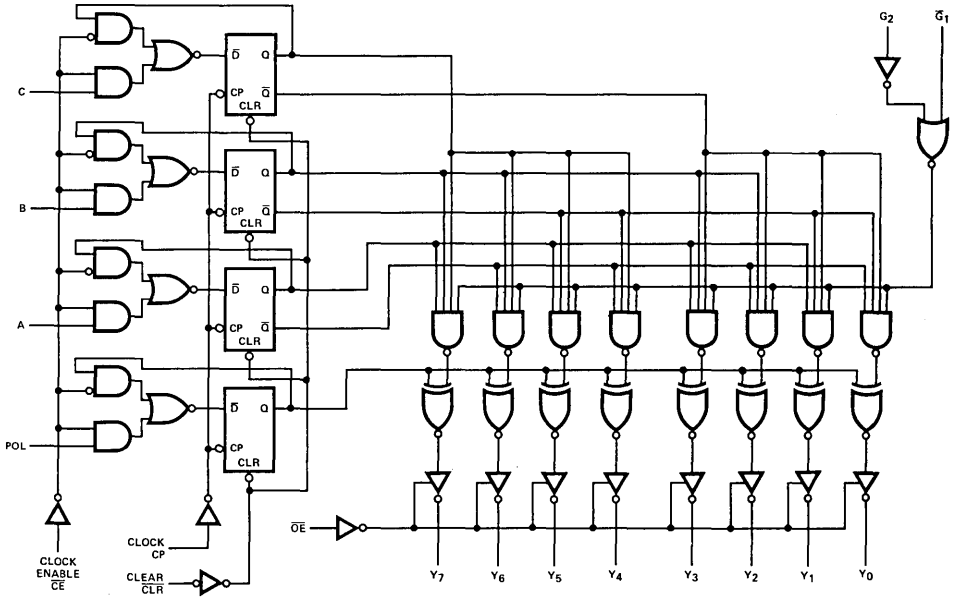
The AmZ8136 is an eight-bit decoder with control storage. It provides a conventional 8-bit decoder function with two enable inputs which may also be used for data input. This can be used to implement a demultiplexer function. In addition, the "exclusive-OR" gates provide polarity control of the selected output. The 3-state outputs are enabled by an active LOW input on the output enable, $\overline{OE}$.

The three control bits representing the output selection and the single bit polarity control are stored in "D" type flip-flops. These flip-flops have Clear, Clock, and Clock Enable functions provided. The $\overline{G}_1$ and G_2 inputs provide either polarity for input control or data.

4

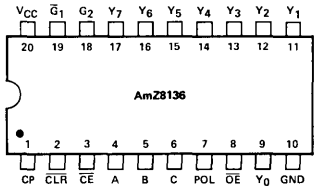
LOGIC DIAGRAM

8-Bit Decoder/Demultiplexer with Control Storage



BLI-190

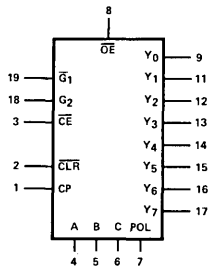
CONNECTION DIAGRAM



Note: Pin 1 is marked for orientation.

BLI-191

LOGIC SYMBOL



BLI-192

V_{CC} = 20
GND = 10

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L	$T_A = 0°C$ to $+70°C$	$V_{CC} = 5.0V \pm 5\%$	MIN. = 4.75 V	MAX. = 5.25 V
MIL	$T_A = -55°C$ to $+125°C$	$V_{CC} = 5.0V \pm 10\%$	MIN. = 4.50 V	MAX. = 5.50 V

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)			Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN., $V_{IN} = V_{IH}$ or V_{IL}	$I_{OH} = -2.6mA$, COM'L		2.4	3.2		Volts
			$I_{OH} = -1.0mA$, MIL		2.4	3.4		
V_{OL}	Output LOW Voltage	V_{CC} = MIN., $V_{IN} = V_{IH}$ or V_{IL}	$I_{OL} = 24mA$, COM'L			0.4	0.5	Volts
			$I_{OL} = 12mA$, MIL			0.35	0.4	
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs		MIL			0.7	Volts
				COM'L			0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., $I_{IN} = -18mA$					−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., $V_{IN} = 0.4V$					−0.4	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., $V_{IN} = 2.7V$					20	μA
I_I	Input HIGH Current	V_{CC} = MAX., $V_{IN} = 7.0V$					0.1	mA
I_O	Off-State (High-Impedance) Output Current	V_{CC} = MAX.	$V_O = 0.4V$				−20	μA
			$V_O = 2.4V$				20	
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX.			−15		−85	mA
I_{CC}	Power Supply Current (Note 4)	V_{CC} = MAX.				37	56	mA

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at $V_{CC} = 5.0V$, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. Test Conditions: A = B = C = $\overline{G_1}$ = G_2 = $\overline{OE}$ = $\overline{CE}$ = GND; CLK = $\overline{CLR}$ = POL = 4.5V.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to +V_{CC} max.
DC Input Voltage	−0.5V to +7.0V
DC Output Current, Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

SWITCHING CHARACTERISTICS

(T_A = +25°C, V_{CC} = 5.0V)

Parameters	Description		Min.	Typ.	Max.	Units	Test Conditions
t_{PLH}	$\overline{G_1}$ to Y_0 – Y_7			17	25	ns	C_L = 45pF, R_L = 667Ω
t_{PHL}				23	34		
t_{PLH}	G_2 to Y_0 – Y_7			20	30	ns	
t_{PHL}				26	39		
t_{PLH}	CP to Y_0 – Y_7			24	36	ns	
t_{PHL}				30	45		
t_{PLH}	$\overline{CLR}$ to Y_0 – Y_7			24	36	ns	
t_{PHL}				31	46		
t_S	$\overline{CE}$ to CP		25			ns	
t_h			0				
t_S	A, B, C, POL to CP		15			ns	
t_h			0				
t_{HZ}	$\overline{OE}$ to Y_0 – Y_7			9	14	ns	C_L = 5pF, R_L = 667Ω
t_{LZ}				11	17		
t_{ZH}	$\overline{OE}$ to Y_0 – Y_7			15	22	ns	C_L = 45pF, R_L = 667Ω
t_{ZL}				16	24		
t_S	Set-up Time, Clear Recovery to CP		20			ns	
t_{pw}	Pulse Width	Clock	15			ns	
		Clear	15				

4

SWITCHING CHARACTERISTICS OVER OPERATING RANGE*

Parameters	Description		COM'L T_A = 0°C to +70°C, V_{CC} = 5.0V ±5% Min.	COM'L Max.	MIL T_A = −55°C to +125°C, V_{CC} = 5.0V ±10% Min.	MIL Max.	Units	Test Conditions
t_{PLH}	$\overline{G_1}$ to Y_0 – Y_7			29		31	ns	C_L = 45pF, R_L = 667Ω
t_{PHL}				39		42		
t_{PLH}	G_2 to Y_0 – Y_7			34		37	ns	
t_{PHL}				44		48		
t_{PLH}	CP to Y_0 – Y_7			40		42	ns	
t_{PHL}				51		55		
t_{PLH}	$\overline{CLR}$ to Y_0 – Y_7			47		54	ns	
t_{PHL}				58		66		
t_S	$\overline{CE}$ to CP		27		30		ns	
t_h			0		0			
t_S	A, B, C, POL to CP		17		20		ns	
t_h			0		0			
t_{HZ}	$\overline{OE}$ to Y_0 – Y_7			17		18	ns	C_L = 5.0pF, R_L = 667Ω
t_{LZ}				27		34		
t_{ZH}	$\overline{OE}$ to Y_0 – Y_7			25		27	ns	C_L = 5.0pF, R_L = 667Ω
t_{ZL}				28		30		
t_S	Set-up Time, Clear Recovery to CP		23		25		ns	
t_{pw}	Pulse Width	Clock	17		20		ns	
		Clear	15		15			

*AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.

FUNCTION TABLE

Mode	Inputs									Internal Registers				Three-State Outputs							
	C	B	A	POL	$\overline{CE}$	$\overline{CLR}$	G*	$\overline{OE}$	CP	Q_C	Q_B	Q_A	Q_{POL}	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
Clear	X	X	X	X	X	L	L	L	X	L	L	L	L	H	H	H	H	H	H	H	H
	X	X	X	X	X	L	H	L	X	L	L	L	L	L	H	H	H	H	H	H	H
Hold	X	X	X	X	H	H	NC	L	↑	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC	NC
Select	L	L	L	H	L	H	H	L	↑	L	L	L	H	H	L	L	L	L	L	L	L
	L	L	H	H	L	H	H	L	↑	L	L	H	H	L	H	L	L	L	L	L	L
	L	H	L	H	L	H	H	L	↑	L	H	L	H	L	L	H	L	L	L	L	L
	L	H	H	H	L	H	H	L	↑	L	H	H	H	L	L	L	H	L	L	L	L
	H	L	L	H	L	H	H	L	↑	H	L	L	H	L	L	L	L	H	L	L	L
	H	L	H	H	L	H	H	L	↑	H	L	H	H	L	L	L	L	L	H	L	L
	H	H	L	H	L	H	H	L	↑	H	H	L	H	L	L	L	L	L	L	H	L
	H	H	H	H	L	H	H	L	↑	H	H	H	H	L	L	L	L	L	L	L	H
	L	L	L	L	L	H	H	L	↑	L	L	L	L	L	H	H	H	H	H	H	H
	L	L	H	L	L	H	H	L	↑	L	L	H	L	H	L	H	H	H	H	H	H
	L	H	L	L	L	H	H	L	↑	L	H	L	L	H	H	L	H	H	H	H	H
	L	H	H	L	L	H	H	L	↑	L	H	H	L	H	H	H	L	H	H	H	H
	H	L	L	L	L	H	H	L	↑	H	L	L	L	H	H	H	H	L	H	H	H
	H	L	H	L	L	H	H	L	↑	H	L	H	L	H	H	H	H	H	L	H	H
	H	H	L	L	L	H	H	L	↑	H	H	L	L	H	H	H	H	H	H	L	H
	H	H	H	L	L	H	H	L	↑	H	H	H	L	H	H	H	H	H	H	H	L
	X	X	X	H	L	H	L	L	↑	X	X	X	H	L	L	L	L	L	L	L	L
	X	X	X	L	L	H	L	L	↑	X	X	X	L	H	H	H	H	H	H	H	H
Output Disable	X	X	X	X	X	X	X	H	X	NC	NC	NC	NC	Z	Z	Z	Z	Z	Z	Z	Z

*

$\overline{G_1}$	G_2	G
L	L	L
L	H	H
H	L	L
H	H	L

NC = No Change X = Don't Care Z = High-Impedance ↑ = Low-to-High Transition

DEFINITION OF TERMS

$\overline{CLR}$ CLEAR – When the CLEAR input is LOW, the control register outputs (Q_A, Q_B, Q_C, Q_{POL}) are set LOW regardless of any other inputs.

CP CLOCK – Enters data into the control register on the LOW-to-HIGH transition.

$\overline{CE}$ CLOCK ENABLE – Allows data to enter the control register when $\overline{CE}$ is LOW. When $\overline{CE}$ is HIGH, the Q_i outputs do not change state, regardless of data or clock input transitions.

A,B,C Inputs to the control register which are entered on the LOW-to-HIGH clock transition if $\overline{CE}$ is LOW.

POL Input to the control register bit used for determining the polarity of the selected output.

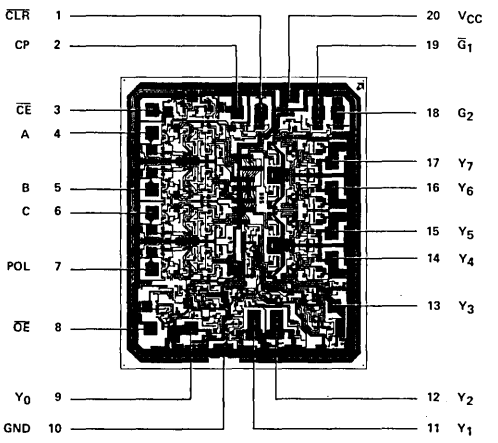
$\overline{G}_1$ Active LOW part of the expression $G = G_1G_2$ [or $G = (G_1)\,G_2$] where G is either data input for the selected Y_n or is used as an input enable.

G_2 Active HIGH part of the expression $G = G_1G_2$.

Y_n The three-state outputs. When active ($\overline{OE}$ = LOW), one of eight outputs is selected by the code stored in the control register, with the polarity of all eight determined by the bit stored in the POL flip-flop of the control register. The selected output can further be controlled by G according to the expression $Y_{SELECTED} = \overline{G \oplus Q_{POL}}$.

$\overline{OE}$ OUTPUT ENABLE. When $\overline{OE}$ is HIGH the Y_n outputs are in the high impedance state; when $\overline{OE}$ is LOW the Y_n's are in their active state as determined by the other control logic. The $\overline{OE}$ input affects the Y_n output buffers only and has no effect on the control register or any other logic.

METALLIZATION AND PAD LAYOUT



DIE SIZE 0.084" X 0.099"

AmZ8140 • AmZ8144

Octal Three-State Buffers

DISTINCTIVE CHARACTERISTICS

- Three-state outputs drive bus lines directly
- Hysteresis at inputs inproves noise margin
- PNP inputs reduce DC loading on bus lines
- Data-to-output propagation delay times – 16ns MAX
- Enable-to-output – 20ns MAX
- 48mA output current
- 20-pin hermetic and molded DIP packages
- 100% product assurance testing to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

The AmZ8140 and AmZ8144 are octal buffers fabricated using advanced low-power Schottky technology. The 20-pin package provides improved printed circuit board density for use in memory address and clock driver applications.

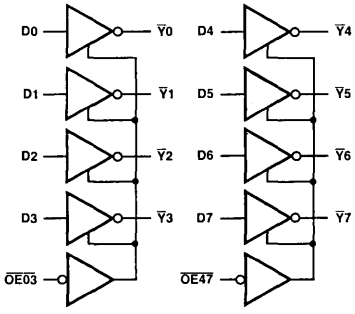
Three-state outputs are provided to drive bus lines directly. The AmZ8140 and AmZ8144 are specified at 48mA and 24mA output sink current. Four buffers are enabled from one common line and the other four from a second enable line. The AmZ8140 and AmZ8144 enables are of similar polarity for use as a unidirectional buffer in which both halves are enabled simultaneously.

Improved noise rejection and high fan-out are provided by input hysteresis and low current PNP inputs.

4

LOGIC DIAGRAMS

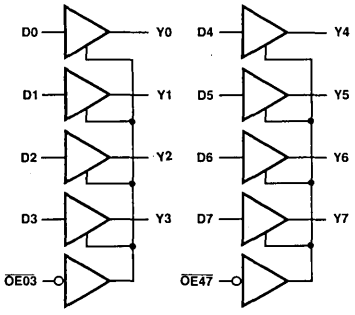
AmZ8140



INPUTS		OUTPUT
$\overline{OE}$	D	Y
H	X	Z
L	H	L
L	L	H

BLI-193

AmZ8144

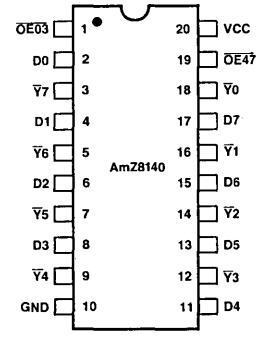


INPUTS		OUTPUT
$\overline{OE}$	D_i	Y
H	X	Z
L	H	H
L	L	L

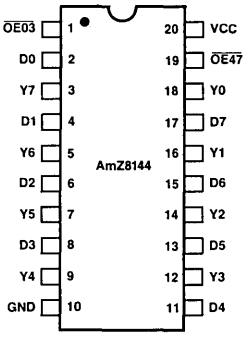
BLI-194

Note: All devices have input hysteresis.

CONNECTION DIAGRAMS
Top Views



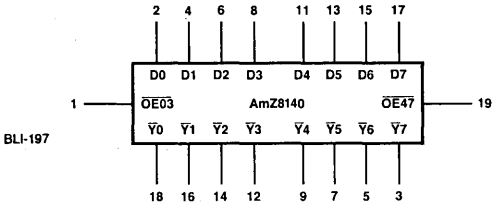
BLI-195



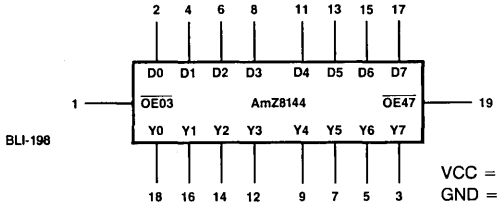
BLI-196

Note: Pin 1 is marked for orientation.

LOGIC SYMBOLS



BLI-197



BLI-198

VCC = Pin 20
GND = Pin 10

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L	T_A = 0 to 70°C	VCC = 5.0V ±5%	(MIN = 4.75V MAX = 5.25V)
MIL	T_A = −55 to +125°C	VCC = 5.0V ±10%	(MIN = 4.50V MAX = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description		Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Units
VOH	High-Level Output Voltage		VCC = MIN, VIH = 2.0V IOH = −3.0mA, VIL = VIL MAX		2.4	3.4		Volts
			VCC = MIN, VIL = 0.5V	MIL, IOH = −12mA	2.0			
				COM'L, IOH = −15mA	2.0			
VOL	Low-Level Output Voltage		VCC = MIN	All IOL = 12mA		0.25	0.4	Volts
				All IOL = 24mA		0.35	0.5	
				COM'L, IOL = 48mA			0.55	
VIH	High-Level Input Voltage		Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
VIL	Low-Level Input Voltage	COM'L					0.8	Volts
		MIL					0.7	
VIK	Input Clamp Voltage		VCC = MIN, II = −18mA				−1.5	Volts
	Hysteresis (VT+ − VT−)		VCC = MIN		0.2	0.4		Volts
IOZH	Off-State Output Current, High-Level Voltage Applied		VCC = MAX VIH = 2.0V VIL = VIL MAX	VO = 2.7V			20	μA
IOZL	Off-State Output Current, Low-Level Voltage Applied			VO = 0.4V			−20	
II	Input Current at Maximum Input Voltage		VCC = MAX, VI = 7.0V				0.1	mA
IIH	High-Level Input Current, Any Input		VCC = MAX, VIH = 2.7V				20	μA
IIL	Low-Level Input Current		VCC = MAX, VIL = 0.4V				−200	μA
ISC	Short Circuit Output Current (Note 3)		VCC = MAX		−50		−225	mA
ICC	Supply Current VCC = MAX Outputs Open	AmZ8140	All Outputs HIGH			13	23	mA
			All Outputs LOW			26	44	
			Outputs at Hi-Z			29	50	
ICC		AmZ8144	All Outputs HIGH			13	23	mA
			All Outputs LOW			27	46	
			Outputs at Hi-Z			32	54	

Notes: 1. For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
2. All typical values are VCC = 5.0V, T_A = 25°C.
3. Not more than one output should be shorted at a time, and duration of the short-circuits should not exceed one second.

MAXIMUM RATINGS above which the useful life may be impaired

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential	−0.5V to +7.0V
DC Voltage Applied to Outputs for HIGH Output State	−0.5V to +V_{CC} max.
DC Input Voltage	−0.5V to +7.0V
DC Output Current	150mA
DC Input Current	−30mA to +5.0mA

AmZ8140 • AmZ8144
SWITCHING CHARACTERISTICS
(T_A = +25°C, VCC = 5.0V)

Parameters	Description	AmZ8140 Min	AmZ8140 Typ	AmZ8140 Max	AmZ8144 Min	AmZ8144 Typ	AmZ8144 Max	Units	Test Conditions (Notes 1-5)
tPLH	Propagation Delay Time, Low-to-High-Level Output		6	10		9	13	ns	CL = 45pF RL = 667Ω
tPHL	Propagation Delay Time, High-to-Low-Level Output		9	13		11	16	ns	
tPZL	Output Enable Time to Low Level		13	20		13	20	ns	
tPZH	Output Enable Time to High Level		8	14		8	14	ns	
tPLZ	Output Disable Time from Low Level		13	20		13	20	ns	CL = 5.0pF RL = 667Ω
tPHZ	Output Disable Time from High Level		12	18		12	18	ns	

AmZ8140
SWITCHING CHARACTERISTICS
OVER OPERATING RANGE*

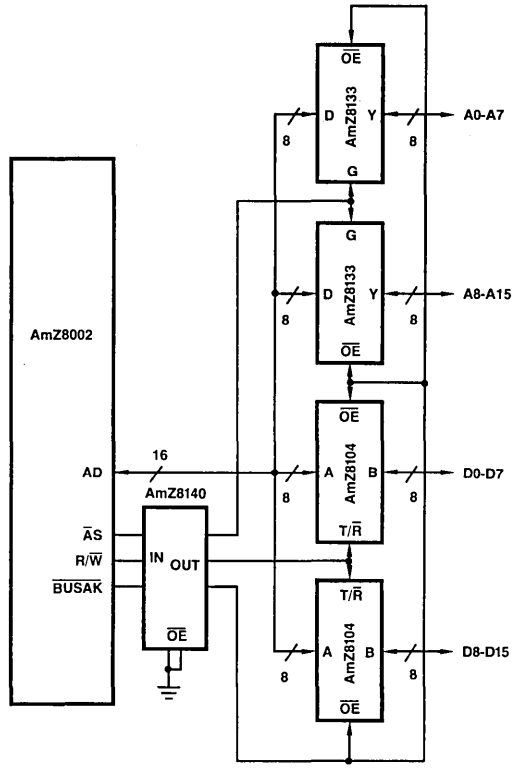
Parameters	Description	COM'L T_A = 0 to 70°C VCC = 5.0V ±5% Min	COM'L Max	MIL T_A = −55 to +125°C VCC = 5.0V ±10% Min	MIL Max	Units	Test Conditions
tPLH	Propagation Delay Time, Low-to-High-Level Output		13		15	ns	CL = 45pF RL = 667Ω
tPHL	Propagation Delay Time, High-to-Low-Level Output		15		18	ns	
tPZL	Output Enable Time to Low Level		25		30	ns	
tPZH	Output Enable Time to High Level		18		21	ns	
tPLZ	Output Disable Time from Low Level		25		30	ns	CL = 5.0pF RL = 667Ω
tPHZ	Output Disable Time from High Level		21		25	ns	

AmZ8144
SWITCHING CHARACTERISTICS
OVER OPERATING RANGE*

Parameters	Description	COM'L T_A = 0 to 70°C VCC = 5.0V ±5% Min	COM'L Max	MIL T_A = −55 to +125°C VCC = 5.0V ±10% Min	MIL Max	Units	Test Conditions
tPLH	Propagation Delay Time, Low-to-High-Level Output		15		16	ns	CL = 45pF RL = 667Ω
tPHL	Propagation Delay Time, High-to-Low-Level Output		18		20	ns	
tPZL	Output Enable Time to Low Level		25		30	ns	
tPZH	Output Enable Time to High Level		18		21	ns	
tPLZ	Output Disable Time from Low Level		25		30	ns	CL = 5.0pF RL = 667Ω
tPHZ	Output Disable Time from High Level		21		25	ns	

*AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.

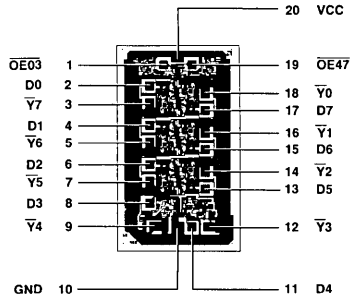
APPLICATION



BLI-199

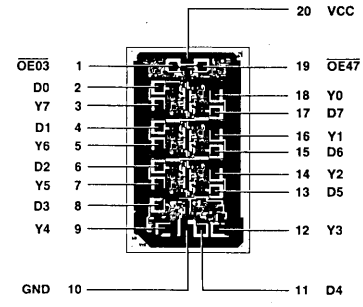
METALLIZATION AND PAD LAYOUTS

AmZ8140



DIE SIZE 0.060" X 0.103"

AmZ8144



DIE SIZE 0.060" X 0.103"

AmZ8148

Chip Select Address Decoder With Acknowledge

DISTINCTIVE CHARACTERISTICS

- One-of-Eight Decoder provides eight chip select outputs
- Acknowledge output responds to enables and acknowledge input command
- Open-collector Acknowledge output for wired-OR application
- Inverting and non-inverting enable inputs for upper address decoding
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

The AmZ8148 Address Decoder combines a three-line to eight-line decoder with four qualifying enable inputs (two active HIGH and two active LOW) and the acknowledge output required for "ready" or "wait state" control of all popular MOS microprocessors.

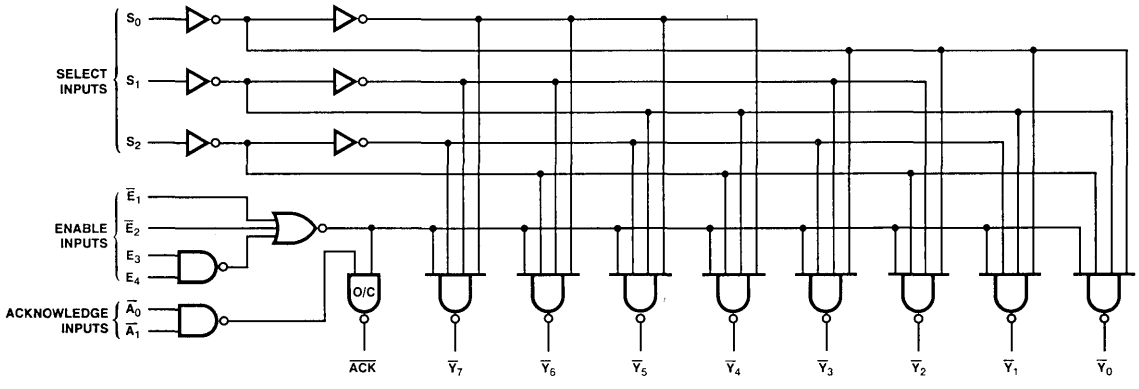
The acknowledge output, ACK, is active LOW and responds to the combination of all enables and an acknowledge active, input command.

The eight chip select outputs are individually active LOW in response to the combination of all enables active and the corresponding 3-bit input code at the S inputs.

The AmZ8148 is intended for chip select decoding in small, medium or large systems where multiple chip selects must be generated and address space must be allocated conservatively.

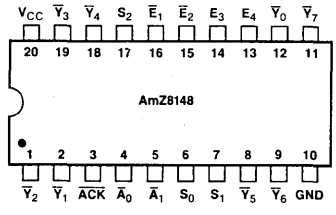
4

LOGIC DIAGRAM



BLI-045

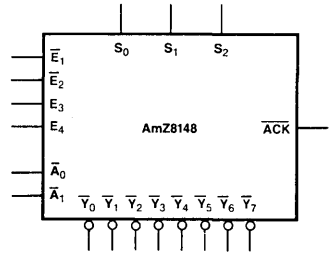
CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BLI-046

LOGIC SYMBOL



BLI-047

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L T_A = 0°C to +70°C V_{CC} = 5.0V ±5% (MIN. = 4.75V MAX. = 5.25V)
MIL T_A = −55°C to +125°C V_{CC} = 5.0V ±10% (MIN. = 4.50V MAX. = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description	Test Conditions (Note 1)		Min.	Typ. (Note 2)	Max.	Units
V_{OH}	Output HIGH Voltage	V_{CC} = MIN. V_{IN} = V_{IH} or V_{IL}	I_{OH} = −440μA	2.4	3.4		Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN. V_{IN} = V_{IH} or V_{IL}	I_{OL} = 4.0mA			0.4	Volts
			I_{OL} = 8.0mA			0.45	
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs	MIL			0.7	Volts
			COM'L			0.8	
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA				−1.5	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.4V				−0.36	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V				20	μA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 7.0V				0.1	mA
I_{SC}	Output Short Circuit Current (Note 3)	V_{CC} = MAX.		−15		−85	mA
I_{CC}	Power Supply Current (Note 4)	V_{CC} = MAX.			15	20	mA

Notes: 1. For conditions shown as MIN. or MAX., use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.
4. TEST CONDITIONS: S_0 = S_1 = S_2 = $\overline{E}_1$ = $\overline{E}_2$ = GND: $\overline{A}_0$ = $\overline{A}_1$ = E_3 = E_4 = 4.5V.

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to +V_{CC} max.
DC Input Voltage	−0.5V to +7.0V
DC Output Current Into Outputs	30mA
DC Input Current	−30mA to +5.0mA

SWITCHING CHARACTERISTICS

(T_A = +25°C, V_{CC} = 5.0V)

Parameters	Description	Min.	Typ.	Max.	Units	Test Conditions
t_{PLH}	S_i to $\overline{Y_i}$ (Three Level Delay)		14	20	ns	C_L = 15pF R_L = 2.0kΩ
t_{PHL}			19	27	ns	
t_{PLH}	S_i to Y_i (Two Level Delay)		13	18	ns	
t_{PHL}			15	21	ns	
t_{PLH}	$\overline{E_1}$, $\overline{E_2}$ to $\overline{Y_i}$		13	18	ns	
t_{PHL}			16	23	ns	
t_{PLH}	E_3, E_4 to $\overline{Y_i}$		15	21	ns	
t_{PHL}			19	27	ns	
t_{PLH}	$\overline{A_i}$ to ACK		25	35	ns	
t_{PHL}			16	22	ns	
t_{PLH}	$\overline{E_1}$, $\overline{E_2}$ to $\overline{ACK}$		29	40	ns	
t_{PHL}			25	35	ns	
t_{PLH}	E_3, E_4 to $\overline{ACK}$		29	40	ns	
t_{PHL}			25	35	ns	

4

SWITCHING CHARACTERISTICS OVER OPERATING RANGE

		COM'L		MIL			
		T_A = 0°C to +70°C V_{CC} = 5.0V ±5%		T_A = −55°C to +125°C V_{CC} = 5.0V ±10%			
Parameters	Description	Min.	Max.	Min.	Max.	Units	Test Conditions
t_{PLH}	S_i to Y_i (Three Level Delay)		27		30	ns	C_L = 50pF R_L = 2.0kΩ
t_{PHL}			34		36	ns	
t_{PLH}	S_i to Y_i (Two Level Delay)		23		25	ns	
t_{PHL}			28		31	ns	
t_{PLH}	$\overline{E_1}$, $\overline{E_2}$ to $\overline{Y_i}$		23		25	ns	
t_{PHL}			29		31	ns	
t_{PLH}	E_3, E_4 to $\overline{Y_i}$		27		28	ns	
t_{PHL}			34		36	ns	
t_{PLH}	$\overline{A_i}$ to ACK		45		45	ns	
t_{PHL}			31		35	ns	
t_{PLH}	$\overline{E_1}$, $\overline{E_2}$ to $\overline{ACK}$		45		45	ns	
t_{PHL}			39		40	ns	
t_{PLH}	E_3, E_4 to $\overline{ACK}$		45		45	ns	
t_{PHL}			39		40	ns	

DEFINITION OF FUNCTIONAL TERMS

S_0, S_1, S_2 Three-line to eight-line chip select decoder inputs.

$\overline{E}_1$, $\overline{E}_2$ The active LOW enable inputs. A HIGH on either the $\overline{E}_1$ or $\overline{E}_2$ input forces all decoded functions to be disabled, and forces $\overline{ACK}$ HIGH.

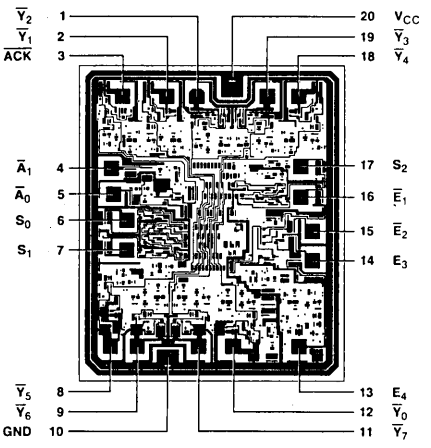
E_3, E_4 The active HIGH enable inputs. A LOW on either E_3 or E_4 inputs forces all the decoded functions to be inhibited, and forces $\overline{ACK}$ HIGH.

A_0, A_1 The acknowledge inputs, A_0 and A_1, are active LOW inputs used as conditions for an active LOW output at the acknowledge, $\overline{ACK}$, output.

$\overline{ACK}$ The acknowledge output, $\overline{ACK}$, is an active LOW output used to signal the microprocessor that specific devices have been selected. $\overline{ACK}$ goes LOW only when $\overline{E}_1$ and $\overline{E}_2$ are LOW, E_3 and E_4 are HIGH and $\overline{A}_0$ or $\overline{A}_1$ is LOW.

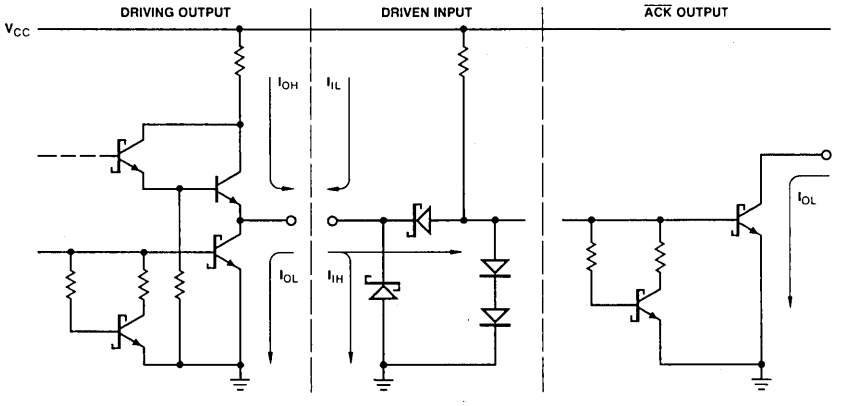
$\overline{Y}_i$ The eight active LOW chip select outputs.

METALLIZATION AND PAD LAYOUT



DIE SIZE: 0.081" X 0.096"

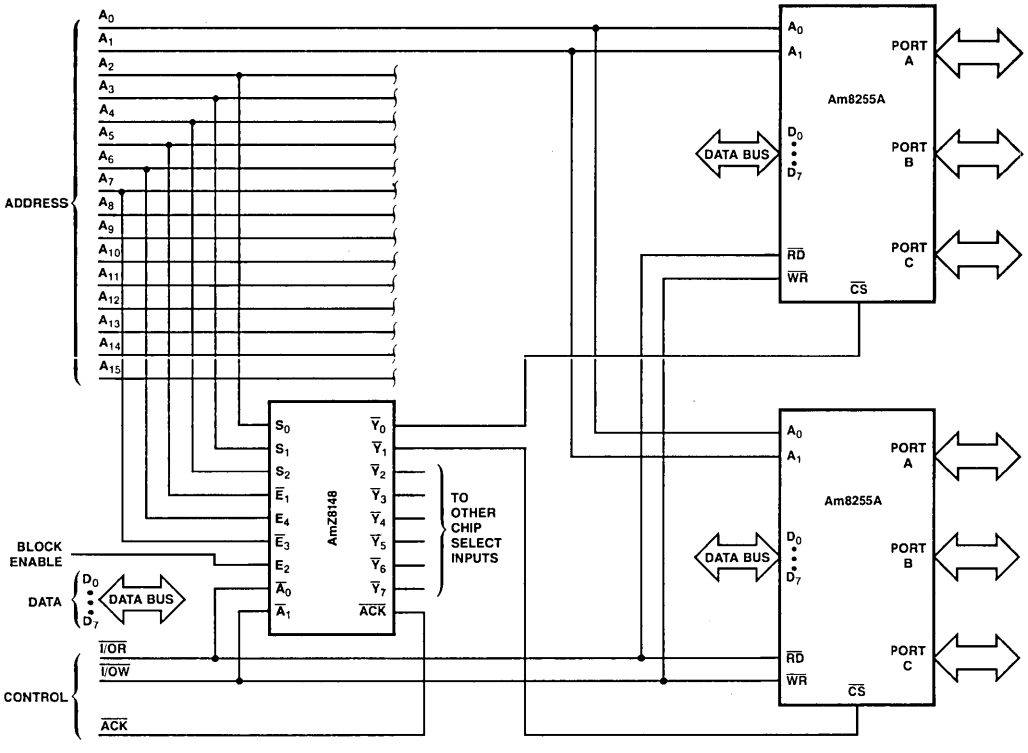
LOW-POWER SCHOTTKY INPUT/OUTPUT CURRENT INTERFACE CONDITIONS



Note: Actual current flow direction shown.

BLI-04

APPLICATION DIAGRAM



BLI-049

FUNCTION TABLES

CHIP SELECT OUTPUTS Y_i

S_2	S_1	S_0	$\overline{E}_1$	$\overline{E}_2$	E_3	E_4	$\overline{Y}_0$	$\overline{Y}_1$	$\overline{Y}_2$	$\overline{Y}_3$	$\overline{Y}_4$	$\overline{Y}_5$	$\overline{Y}_6$	$\overline{Y}_7$
L	L	L	L	L	H	H	L	H	H	H	H	H	H	H
L	L	H	L	L	H	H	H	L	H	H	H	H	H	H
L	H	L	L	L	H	H	H	H	L	H	H	H	H	H
L	H	H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	L	L	L	H	H	H	H	H	H	L	H	H	H
H	L	H	L	L	H	H	H	H	H	H	H	L	H	H
H	H	L	L	L	H	H	H	H	H	H	H	H	L	H
H	H	H	L	L	H	H	H	H	H	H	H	H	H	L
X	X	X	H	X	X	X	H	H	H	H	H	H	H	H
X	X	X	X	H	X	X	H	H	H	H	H	H	H	H
X	X	X	X	X	L	X	H	H	H	H	H	H	H	H
X	X	X	X	X	X	L	H	H	H	H	H	H	H	H

ACKNOWLEDGE OUTPUT $\overline{ACK}$

$\overline{E}_1$	$\overline{E}_2$	E_3	E_4	$\overline{A}_0$	$\overline{A}_1$	$\overline{ACK}$
H	X	X	X	X	X	H
X	H	X	X	X	X	H
X	X	L	X	X	X	H
X	X	X	L	X	X	H
L	L	H	H	L	X	L
L	L	H	H	X	L	L

AmZ8160

Cascadable 16-Bit Error Detection and Correction Unit

ADVANCED DATA

DISTINCTIVE CHARACTERISTICS

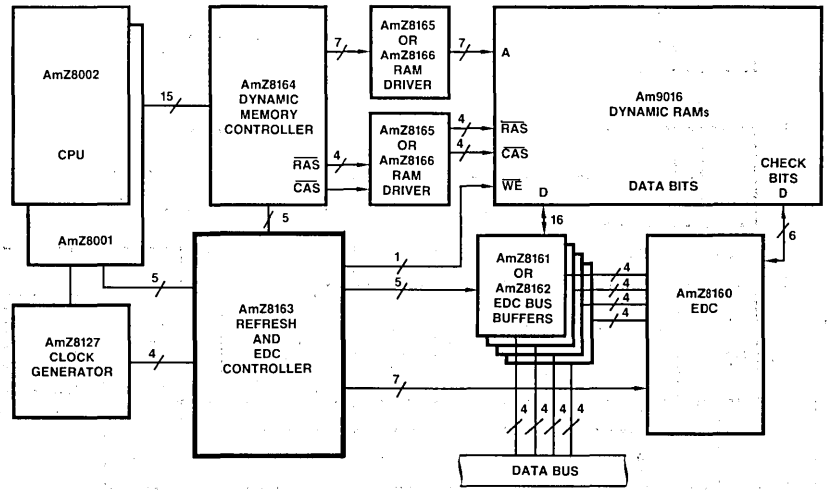
- Modified Hamming Code
  Detects multiple errors and corrects single bit errors in a parallel data word. Ideal for use in dynamic memory systems.
- Syndromes provided
  The AmZ8160 makes available the syndrome bits when an error occurs so the location of memory faults can be logged.
- Microprocessor compatible
  The AmZ8160 is designed to work with AmZ8000 microprocessor systems.
- Advanced circuit and process technologies
  Newest bipolar LSI techniques provide very high performance.
  Data-in to error detection typically 30ns
  Data-in to corrected data out typically 50ns
- Built-in Diagnostics
  Extra logic on the chip provides diagnostic functions to be used during device test and for system diagnostics.

GENERAL DESCRIPTION

The AmZ8160 Error Detection and Correction Unit (EDC) contains the logic necessary to generate 6 check bits on a 16-bit data field according to a modified Hamming Code, and to correct the data word when check bits are supplied. Operating on data read from memory, the AmZ8160 will correct any single bit error and will detect all double on some triple bit errors. The AmZ8160 is expandable to operate on 32-bit words (7 check bits) and 64-bit words (8 check bits). In all configurations, the device makes the error syndrome available on separate outputs for data logging.

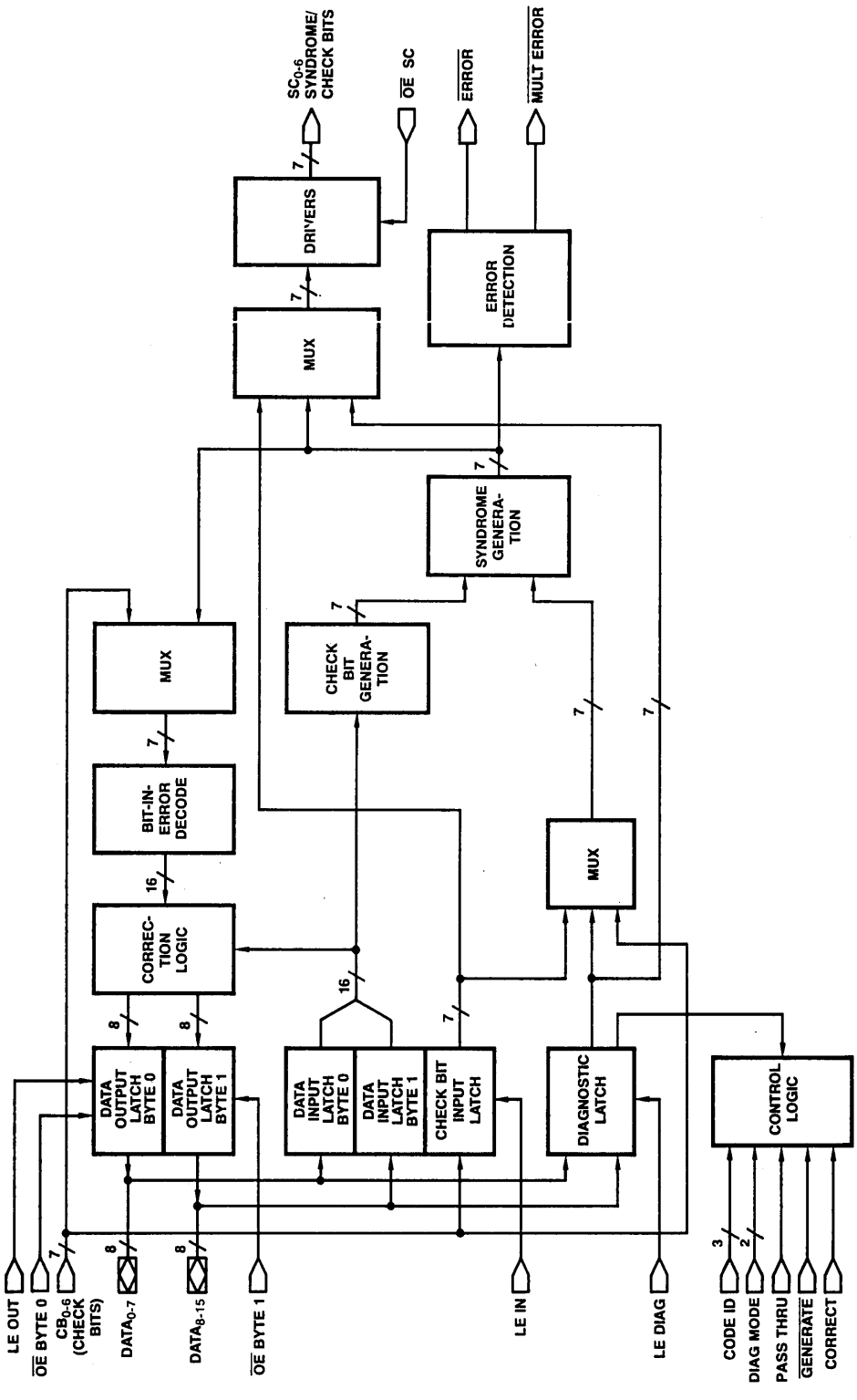
The AmZ8160 also features two diagnostic modes, in which diagnostic data can be forced into portions of the chip to simplify device testing and to execute system diagnostic functions. The product is supplied in a 48 lead hermetic DIP package.

SYSTEM EXAMPLE



BLI-200

BLOCK DIAGRAM



EDC Architecture

The EDC Unit is a powerful 16-bit cascadable slice used for check bit generation, error detection, error correction and diagnostics.

As shown in the block diagram, the device consists of the following:

- Data Input Latch
- Check Bit Input Latch
- Check Bit Generation Logic
- Syndrome Generation Logic
- Error Detection Logic
- Error Correction Logic
- Data Output Latch
- Diagnostic Latch
- Control Logic

Data Input Latch

16 bits of data are loaded from the bidirectional DATA lines under control of the Latch Enable input, LE IN. Depending on the control mode the input data is either used for check bit generation or error detection/correction.

Check Bit Input Latch

Seven check bits are loaded under control of LE IN. Check bits are used in the Error Detection and Error Correction modes.

Check Bit Generation Logic

This block generates the appropriate check bits for the 16 bits of data in the Data Input Latch. The check bits are generated according to a modified Hamming code.

Syndrome Generation Logic

In both Error Detection and Error Correction modes, this logic block compares the check bits read from memory against a newly generated set of check bits produced for the data read in from memory. If both sets of check bits match, then there are no errors. If there is a mismatch, then one or more of the data or check bits is in error.

The syndrome bits are produced by an exclusive-OR of the two sets of check bits. If the two sets of check bits are identical (meaning there are no errors) the syndrome bits will be all zeroes. If there are errors, the syndrome bits can be decoded to determine the number of errors and the bit-in-error.

Error Detection Logic

This section decodes the syndrome bits generated by the Syndrome Generation Logic. If there are no errors in either the input data or check bits, the $\overline{\text{ERROR}}$ and $\overline{\text{MULTI ERROR}}$ outputs remain HIGH. If one or more errors are detected, $\overline{\text{ERROR}}$ goes LOW. If two or more errors are detected, both $\overline{\text{ERROR}}$ and $\overline{\text{MULTI ERROR}}$ go LOW.

Error Correction Logic

For single errors, the Error Correction Logic complements (corrects) the single data bit in error. This corrected data is loadable into the Data Output Latch, which can then be read onto the bidirectional data lines. If the single error is one of the check bits, the correction logic does not place corrected check bits on the syndrome/check bit outputs. If the corrected check bits are needed the EDC must be switched to Generate Mode.

Data Output Latch

The Data Output Latch is used for storing the result of an error correction operation. The latch is loaded from the correction logic under control of the Data Output Latch Enable, LE OUT. The Data Output Latch may also be loaded directly from the Data Input Latch under control of the PASS THRU control input.

The Data Output Latch is split into two 8-bit (byte) latches which may be enabled independently for reading onto the bidirectional data lines.

Diagnostic Latch

This is a 16-bit latch loadable from the bidirectional data lines under control of the Diagnostic Latch Enable, LE DIAG. The Diagnostic Latch contains check bit information in one byte and control information in the other byte. The Diagnostic Latch is used for driving the device when in Internal Control Mode, or for supplying check bits when in one of the Diagnostic Modes.

Control Logic

The control logic determines the specific mode the device operates in. Normally the control logic is driven by external control inputs. However, in Internal Control Mode, the control signals are instead read from the Diagnostic Latch.

PIN DEFINITIONS

$DATA_{0-15}$ 16 bidirectional data lines. They provide input to the Data Input Latch and Diagnostic Latch, and receive output from the Data Output Latch. $DATA_0$ is the least significant bit; $DATA_{15}$ the most significant.

CB_{0-6} Seven Check Bit input lines. The check bit lines are used to input check bits for error detection. Also used to input syndrome bits for error correction in 32 and 64-bit configurations.

LE IN Latch Enable – Data Input Latch. Controls latching of the input data. When HIGH the Data Input Latch and Check Bit Input Latch follow the input data and input check bits. When LOW, the Data Input Latch and Check Bit Input Latch are latched to their previous state.

$\overline{\text{GENERATE}}$ Generate Check Bits input. When this input is LOW the EDC is in the Check Bit Generate Mode. When HIGH the EDC is in the Detect Mode or Correct Mode.

In the Generate Mode the circuit generates the check bits or partial check bits specific to the data in the Data Input Latch. The generated check bits are placed on the SC outputs.

In the Detect or Correct Modes the EDC detects single and multiple errors, and generates syndrome bits based upon the contents of the Data Input Latch and Check Bit Input Latch. In Correct Mode, single-bit errors are also automatically corrected – corrected data is placed at the inputs of the Data Output Latch. The syndrome result is placed on the SC outputs and indicates in a coded form the number of errors and the bit-in-error.

SC_{0-6} Syndrome/Check Bit outputs. These seven lines hold the check/partial-check bits when the EDC is in Generate Mode, and will hold the syndrome/partial syndrome bits when the device is in Detect or Correct Modes. These are 3-state outputs.

$\overline{\text{OE SC}}$ Output Enable – Syndrome/Check Bits. When LOW, the 3-state output lines SC_{0-6} are enabled. When HIGH, the SC outputs are in the high impedance state.

$\overline{\text{ERROR}}$ Error Detected output. When the EDC is in Detect or Correct Mode, this output will go LOW if one or more syndrome bits are asserted, meaning there are one or more bit errors in the data or check bits. If no syndrome bits are asserted, there are no errors detected and the output will be HIGH. In Generate Mode, $\overline{\text{ERROR}}$ is forced HIGH. (In a 64-bit configuration, $\overline{\text{ERROR}}$ must be externally implemented.)

$\overline{\text{MULTI ERROR}}$ Multiple Errors Detected output. When the EDC is in Detect or Correct Mode, this output if LOW indicates that there are two or more bit errors that have been detected. If HIGH this indicates that either one or no errors have been detected. In Generate Mode, $\overline{\text{MULTI ERROR}}$ is forced HIGH. (In a 64-bit configuration, $\overline{\text{MULTI ERROR}}$ must be externally implemented.)

CORRECT Correct input. When HIGH this signal allows the correction network to correct any single-bit error in the Data Input Latch (by complementing the bit-in-error) before putting it onto the Data Output Latch. When LOW the EDC will drive data directly from the Data Input Latch to the Data Output Latch without correction.

LE OUT Latch Enable – Data Output Latch. Controls the latching of the Data Output Latch. When LOW the Data Output Latch is latched to its previous state. When HIGH the Data Output Latch follows the output of the Data Input Latch as modified by the correction logic network. In Correct Mode, single-bit errors are corrected by the network before loading into the Data Output Latch. In Detect Mode, the contents of the Data Input Latch are passed through the correction network unchanged into the Data Output Latch. The inputs to the Data Output Latch are unspecified if the EDC is in Generate Mode.

$\overline{\text{OE}}$ BYTE 0, $\overline{\text{OE}}$ BYTE 1 Output Enable – Bytes 0 and 1, Data Output Latch. These lines control the 3-state outputs for each of the two bytes of the Data Output Latch. When LOW these lines enable the Data Output Latch and when HIGH these lines force the Data Output Latch into the high impedance state. The two enable lines can be separately activated to enable only one byte of the Data Output Latch at a time.

PASS THRU Pass Thru input. This line when HIGH forces the contents of the Check Bit Input Latch onto the Syndrome/Check Bit outputs (SC_{0-6}) and the unmodified contents of the Data Input Latch onto the inputs of the Data Output Latch.

DIAG $MODE_{0-1}$ Diagnostic Mode Select. These two lines control the initialization and diagnostic operation of the EDC.

CODE ID_{0-2} Code Identification inputs. These three bits identify the size of the total data word to be processed and which 16-bit slice of larger data words a particular EDC is processing. The three allowable data word sizes are 16, 32 and 64 bits and their respective modified Hamming codes are designated 16/22, 32/39 and 64/72. Special CODE ID input 001 (ID_2, ID_1, ID_0) is also used to instruct the EDC that the signals CODE ID_{0-2}, DIAG $MODE_{0-1}$, CORRECT and PASS THRU are to be taken from the Diagnostic Latch, rather than from the input control lines.

LE DIAG Latch Enable – Diagnostic Latch. When HIGH the Diagnostic Latch follows the 16-bit data on the input lines. When LOW the outputs of the Diagnostic Latch are latched to their previous states. The Diagnostic Latch holds diagnostic check bits, and internal control signals for CODE ID_{0-2}, DIAG $MODE_{0-1}$, CORRECT and PASS THRU.

4

FUNCTIONAL DESCRIPTION

The EDC contains the logic necessary to generate check bits on a 16-bit data field according to a modified Hamming code. Operating on data read from memory, the EDC will correct any single-bit error, and will detect all double and some triple-bit errors. It may be configured to operate on 16-bit data words (with 6 check bits), 32-bit data words (with 7 check bits) and 64-bit data words (with 8 check bits). In all configurations, the device makes the error syndrome bits available on separate outputs for error data logging.

Code and Byte Specification

The EDC may be configured in several different ways and operates differently in each configuration. It is necessary to indicate to the device what size data word is involved and which bytes of the data word it is processing. This is done with input lines CODE ID_{0-2}, as shown in Table I. The three modified Hamming codes referred to in Table I are:

- 16/22 code – 16 data bits
  – 6 check bits
  – 22 bits in total.
- 32/39 code – 32 data bits
  – 7 check bits
  – 39 bits in total.
- 64/72 code – 64 data bits
  – 8 check bits
  – 72 bits in total.

CODE ID input 001 (ID_2, ID_1, ID_0) is a special code used to operate the device in Internal Control Mode (described later in this section).

TABLE I. HAMMING CODE AND SLICE IDENTIFICATION.

CODE ID_2	CODE ID_1	CODE ID_0	Hamming Code and Slice Selected
0	0	0	Code 16/22
0	0	1	Internal Control Mode
0	1	0	Code 32/39, Bytes 0 and 1
0	1	1	Code 32/39, Bytes 2 and 3
1	0	0	Code 64/72, Bytes 0 and 1
1	0	1	Code 64/72, Bytes 2 and 3
1	1	0	Code 64/72, Bytes 4 and 5
1	1	1	Code 64/72, Bytes 6 and 7

Control Mode Selection

The device control lines are $\overline{\text{GENERATE}}$, CORRECT, PASS THRU, DIAG $MODE_{0-1}$ and CODE ID_{0-2}. Table II indicates the control modes selected by various combinations of the control line inputs.

Diagnostics

Table III shows specifically how DIAG $MODE_{0-1}$ select between normal operation, initialization and one of two diagnostic modes.

The Diagnostic Modes allow the user to operate the EDC under software control in order to verify proper functioning of the device.

Check and Syndrome Bit Labeling

The check bits generated in the EDC are designated as follows:

- 16-bit configuration – CX C0, C1, C2, C4, C8;
- 32-bit configuration – CX, C0, C1, C2, C4, C8, C16;
- 64-bit configuration – CX, C0, C1, C2, C4, C8, C16, C32.

Syndrome bits are similarly labeled SX through S32. There are only 6 syndrome bits in the 16-bit configuration, 7 for 32 bits and 8 syndrome bits in the 64-bit configuration.

FUNCTIONAL DESCRIPTION – 16-BIT DATA WORD CONFIGURATION

The 16-bit format consists of 16 data bits, 6 check bits and is referred to as 16/22 code (see Figure 1).

The 16-bit configuration is shown in Figure 2.

Generate Mode

In this mode check bits will be generated that correspond to the contents of the Data Input Latch. The check bits generated are placed on the outputs SC_{0-5} (SC_6 is unspecified for 16-bit operation).

Check bits are generated according to a modified Hamming code. Details of the code for check bit generation are contained in Table IV. Each check bit is generated as either an XOR or XNOR of eight of the 16 data bits as indicated in the table. The XOR function results in an even parity check bit, the XNOR is an odd parity check bit.

Detect Mode

In this mode the device examines the contents of the Data Input Latch against the Check Bit Input Latch, and will detect all single-bit errors, all double-bit errors and some triple-bit errors. If one or more errors are detected, $\overline{\text{ERROR}}$ goes LOW. If two or more errors are detected, $\overline{\text{MULTI ERROR}}$ goes LOW. Both error indicators are HIGH if there are no errors.

Also available on device outputs SC_{0-5} are the syndrome bits generated by the error detection step. The syndrome bits may be decoded to determine if a bit error was detected and, for single-bit errors, which of the data or check bits is in error. Table V gives the chart for decoding the syndrome bits generated by the 16-bit configuration (as an example, if the syndrome bits SX/S0/S1/S2/S4/S8 were 101001 this would be decoded to indicate that there is a single-bit error at data bit 9). If no error is detected the syndrome bits will all be zeroes.

In Detect Mode, the contents of the Data Input Latch are driven directly to the inputs of the Data Output Latch without correction.

Correct Mode

In this mode, the EDC functions the same as in Detect Mode except that the correction network is allowed to correct (complement) any single-bit error of the Data Input Latch before putting it onto the inputs of the Data Output Latch. If multiple errors are detected, the output of the correction network is unspecified. If the single-bit error is a check bit there is no automatic correction. If check bit correction is desired, this can be done by placing the device in Generate Mode to produce a correct check bit sequence for the data in the Data Input Latch.

Pass Thru Mode

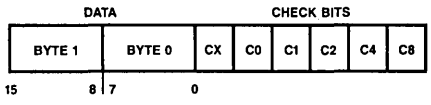
In this mode, the unmodified contents of the Data Input Latch are placed on the inputs of the Data Output Latch and the contents of the Check Bit Input Latch are placed on outputs SC_{0-5}.

TABLE II. EDC CONTROL MODE SELECTION.

$\overline{\text{GENERATE}}$	CORRECT	PASS THRU	DIAG MODE$_{0-1}$ (DM_1, DM_0)	CODE ID$_{0-2}$ (ID_2, ID_1, ID_0)	Control Mode Selected
LOW	LOW	LOW	00	Not 001	Generate
LOW	LOW	LOW	01	Not 001	Generate Using Diagnostic Latch
LOW	LOW	LOW	10	Not 001	Generate
LOW	LOW	LOW	11	Not 001	Initialize
LOW	LOW	HIGH	00	Not 001	Pass Thru
LOW	LOW	HIGH	01	Not 001	Pass Thru
LOW	LOW	HIGH	10	Not 001	Pass Thru
LOW	LOW	HIGH	11	Not 001	Undefined
LOW	HIGH	LOW	00	Not 001	Generate
LOW	HIGH	LOW	01	Not 001	Generate Using Diagnostic Latch
LOW	HIGH	LOW	10	Not 001	Generate
LOW	HIGH	LOW	11	Not 001	Initialize
LOW	HIGH	HIGH	00	Not 001	Pass Thru
LOW	HIGH	HIGH	01	Not 001	Pass Thru
LOW	HIGH	HIGH	10	Not 001	Pass Thru
LOW	HIGH	HIGH	11	Not 001	Undefined
HIGH	LOW	LOW	00	Not 001	Detect
HIGH	LOW	LOW	01	Not 001	Detect
HIGH	LOW	LOW	10	Not 001	Detect Using Diagnostic Latch
HIGH	LOW	LOW	11	Not 001	Initialize
HIGH	LOW	HIGH	00	Not 001	Pass Thru
HIGH	LOW	HIGH	01	Not 001	Pass Thru
HIGH	LOW	HIGH	10	Not 001	Pass Thru
HIGH	LOW	HIGH	11	Not 001	Undefined
HIGH	HIGH	LOW	00	Not 001	Correct
HIGH	HIGH	LOW	01	Not 001	Correct
HIGH	HIGH	LOW	10	Not 001	Correct Using Diagnostic Latch
HIGH	HIGH	LOW	11	Not 001	Initialize
HIGH	HIGH	HIGH	00	Not 001	Pass Thru
HIGH	HIGH	HIGH	01	Not 001	Pass Thru
HIGH	HIGH	HIGH	10	Not 001	Pass Thru
HIGH	HIGH	HIGH	11	Not 001	Undefined
Any	Any	Any	Any	001	Internal Control Using Diagnostic Latch

TABLE III. DIAGNOSTIC MODE CONTROL.

DIAG MODE$_1$	DIAG MODE$_0$	Diagnostic Mode Selected
0	0	**Non-diagnostic mode.** The EDC functions normally in all modes.
0	1	**Diagnostic Mode A.** The contents of the Diagnostic Latch are substituted for the normally generated check bits when in the Generate Mode. The EDC functions normally in the Detect or Correct modes.
1	0	**Diagnostic Mode B.** In the Detect or Correct Mode, the contents of the Diagnostic Latch are substituted for the check bits normally read from the Check Bit Input Latch. The EDC functions normally in the Generate Mode.
1	1	**Initialize.** The inputs of the Data Output Latch are forced to zeroes and the check bits generated correspond to the all-zero data.



Uses Modified Hamming Code 16/22
- 16 data bits
- 6 check bits
- 22 bits in total

BLI-202

Figure 1. 16 Bit Data Format.

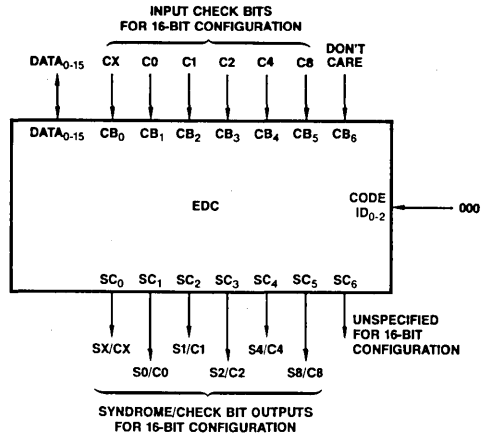


Figure 2. 16 Bit Configuration.

BLI-203

TABLE IV. 16-BIT MODIFIED HAMMING CODE – CHECK BIT ENCODE CHART.

Generated Check Bits	Parity	Participating Data Bits														
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CX	Even (XOR)		X	X	X		X			X	X		X			X	
C0	Even (XOR)	X	X	X		X		X		X		X		X			
C1	Odd (XNOR)	X			X	X			X		X	X			X		X
C2	Odd (XNOR)	X	X				X	X	X				X	X	X		
C4	Even (XOR)			X	X	X	X	X	X							X	X
C8	Even (XOR)									X	X	X	X	X	X	X	X

The check bit is generated as either an XOR or XNOR of the eight data bits noted by an "X" in the table.

TABLE V. SYNDROME DECODE TO BIT-IN-ERROR.

Syndrome Bits										
SX	S0	S1	S8	0	1	0	1	0	1	0	1
			S4	0	0	1	1	0	0	1	1
			S2	0	0	0	0	1	1	1	1
0	0	0		*	C8	C4	T	C2	T	T	M
0	0	1		C1	T	T	15	T	13	7	T
0	1	0		C0	T	T	M	T	12	6	T
0	1	1		T	10	4	T	0	T	T	M
1	0	0		CX	T	T	14	T	11	5	T
1	0	1		T	9	3	T	M	T	T	M
1	1	0		T	8	2	T	1	T	T	M
1	1	1		M	T	T	M	T	M	M	T

* – no errors detected
Number – the location of the single bit-in-error
T – two errors detected
M – three or more errors detected

TABLE VI. DIAGNOSTIC LATCH LOADING – 16-BIT FORMAT.

Data Bit	Internal Function
0	Diagnostic Check Bit X
1	Diagnostic Check Bit 0
2	Diagnostic Check Bit 1
3	Diagnostic Check Bit 2
4	Diagnostic Check Bit 4
5	Diagnostic Check Bit 8
6, 7	Don't Care
8	CODE ID 0
9	CODE ID 1
10	CODE ID 2
11	DIAG MODE 0
12	DIAG MODE 1
13	CORRECT
14	PASS THRU
15	Don't Care

Diagnostic Latch

The Diagnostic Latch serves both for diagnostic uses and internal control uses. It is loaded from the DATA lines under the control of LE DIAG. Table VI shows the loading definitions for the DATA lines.

Generate Using Diagnostic Latch (Diagnostic Mode A)

Detect Using Diagnostic Latch (Diagnostic Mode B)

Correct Using Diagnostic Latch (Diagnostic Mode B)

These are special diagnostic modes selected by DIAG $MODE_{0-1}$ where either normal check bit inputs or outputs are substituted for by check bits loaded into the Diagnostic Latch. See Table III for details.

Internal Control Mode

This mode is selected by CODE ID_{0-2} input 001 (ID_2, ID_1, ID_0).

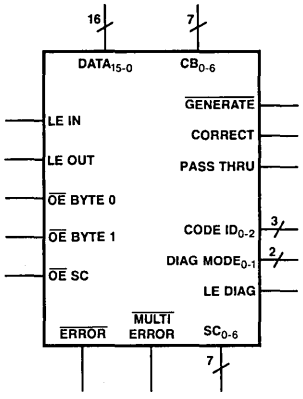
When in Internal Control Mode, the EDC takes the CODE ID_{0-2}, DIAG $MODE_{0-1}$, CORRECT and PASS THRU control signals from the internal Diagnostic Latch rather than from the external input lines.

Table VI gives the format for loading the Diagnostic Latch.

32 and 64-Bit Operation

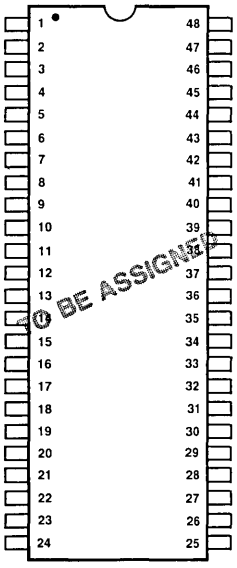
The EDC can easily be cascaded to operate on 32 and 64-bit data words. Since this is unlikely to occur in a Z8000 system, it is not discussed in this data sheet. Interested users should refer to the Am2960 data sheet for more information.

LOGIC SYMBOL



BLI-204

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BLI-205

4

AmZ8161 • AmZ8162

4-Bit Error Correction Multiple Bus Buffers

IN DEVELOPMENT

DISTINCTIVE CHARACTERISTICS

- Quad high-speed LSI bus-transceiver
- Provides complete data path interface between the AmZ8160 Error Detection and Correction Unit, the system data bus and dynamic RAM memory
- 3-state 24mA output to data bus
- 3-state data output to memory
- Inverting data bus for AmZ8161 and non-inverting for AmZ8162
- Data bus latches allow operation with multiplexed buses
- Advanced Low-Power Schottky processing
- 100% product assurance screening to MIL-STD-883 requirements

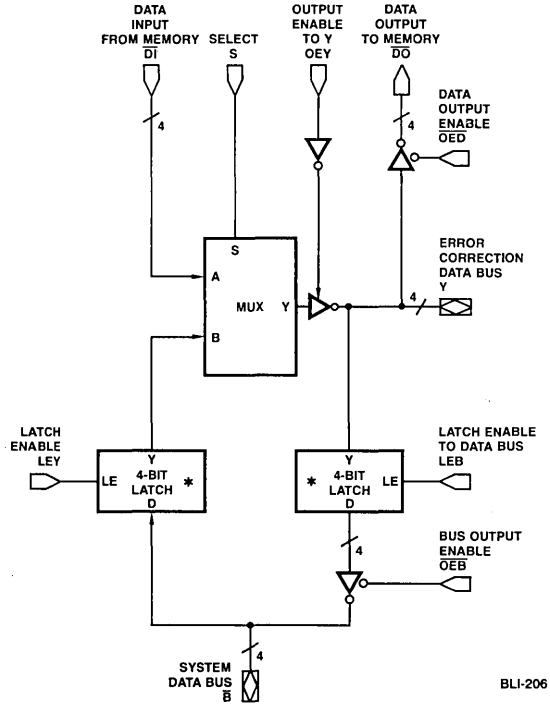
FUNCTIONAL DESCRIPTION

The AmZ8161 and AmZ8162 are high-performance, low-power Schottky multiple bus buffers that provide the complete data path interface between the AmZ8160 Error Detection and Correction Unit, dynamic RAM memory and the AmZ8000 system data bus. The AmZ8161 provides an inverting data path between the data bus ($\overline{B_i}$) and the AmZ8160 error correction data input (Y_i) and the AmZ8162 provides a non-inverting configuration (B_i to Y_i). Both devices provide inverting data paths between the AmZ8160 and memory data bus thereby optimizing internal data path speeds.

The AmZ8161 and AmZ8162 are 4-bit devices. Four devices are used to interface each 16-bit AmZ8160 Error Detection and Correction Unit with dynamic memory. The system can easily be expanded to 32 or more bits for wider memory applications. The 4-bit configuration allows enabling the appropriate devices two-at-a-time for intermixed word or byte, read and write in 16-bit systems with error correction.

Data latches between the error correction data bus and the system data bus facilitate the addition of error corrected memory in synchronous data bus systems. They also provide a data holding capability during single-step system operation.

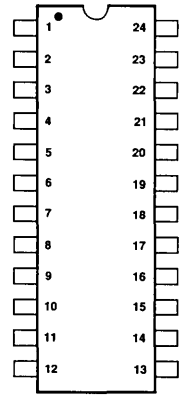
AmZ8161 LOGIC DIAGRAM



*AmZ8162 is the same function but non-inverting between the Y bus to the system data bus, B. This is done by making both latches inverting.

CONNECTION DIAGRAM

Top View



24 pin slim (0.3")

BLI-207

Note: Pin 1 is marked for orientation.

AmZ8163

Dynamic Memory Timing, Refresh and EDC Controller

DISTINCTIVE CHARACTERISTICS

- Complete AmZ8000 CPU to dynamic RAM contol interface
- RAS/CAS Sequencer to eliminate delay lines
- Memory request/refresh arbitration
- Controls for Word/Byte read or write
- Complete EDC data path and mode controls
- Refresh interval timer independent of CPU
- Refresh control during Single Step or Halt modes
- EDC error flag latches for error logging under software control
- Also, complete control for 8-Bit MOS μP

FUNCTIONAL DESCRIPTION

The AmZ8163 is a high speed bus interface controller forming an integral part of the AmZ8000 memory support chip set using dynamic MOS RAMs with Error Detection and Correction (EDC). The complete chip set includes the AmZ8127 Clock Generator and Controller, the AmZ8164 Dynamic Memory Controller, the AmZ8161/2 EDC Bus Buffers, the AmZ8160 EDC Unit and optional AmZ8165/6 RAM Drivers.

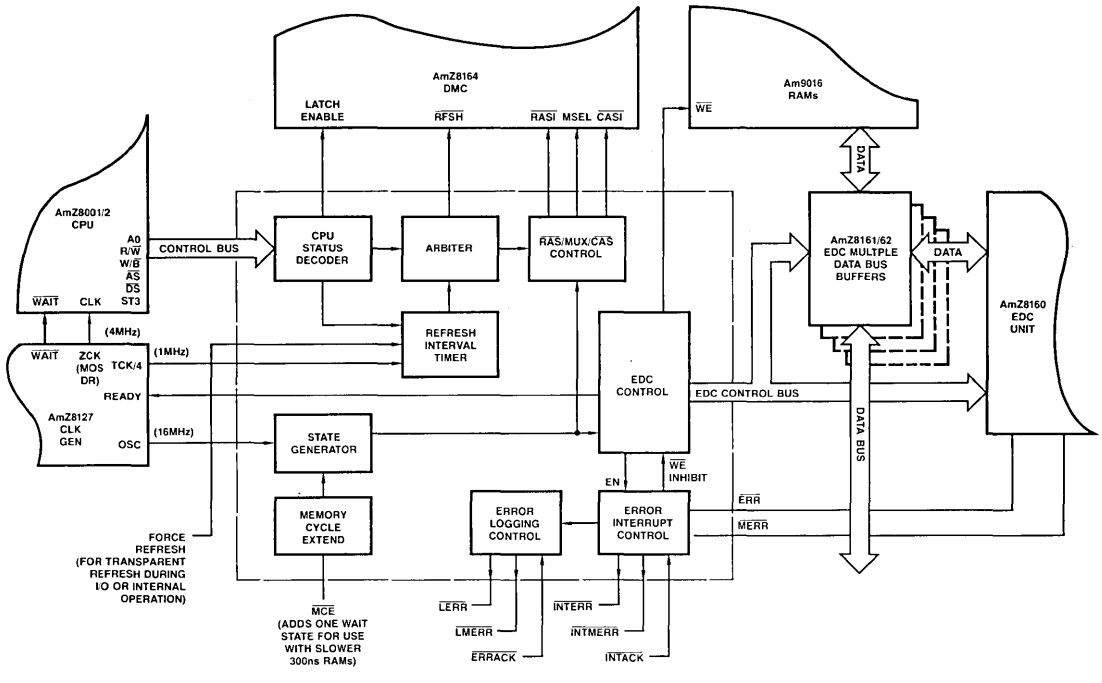
The AmZ8163 provides all of the control interface functions including RAS/Address MUX/CAS timing (without delay lines), refresh timing, memory request/refresh arbitration and all EDC enables and controls. The enable controls are configured for both word and byte operations including the data controls for byte write with error correction. The AmZ8163 generates bus and operating mode controls for the AmZ8160 EDC Unit.

The AmZ8163 uses the AmZ8127 16MHz (4 x CLK) output to generate RAS/Address MUX/CAS timing. An internal refresh interval timer generates the memory refresh request independent of the CPU to guarantee the proper refresh timing under all combinations of CPU and DMA memory requests.

4

FUNCTION DIAGRAM

REFRESH AND EDC CONTROLLER



BLI-167

AmZ8164

Dynamic Memory Controller

DISTINCTIVE CHARACTERISTICS

- Dynamic Memory Controller for 16K and 64K MOS dynamic RAMs
- 8-Bit Refresh Counter for refresh address generation, has clear input and terminal count output
- Refresh Counter Terminal count selectable at 256 or 128
- Latch Input RAS Decoder provides 4 RAS outputs, all active during refresh
- Dual 8-Bit Address Latches plus separate $\overline{RAS}$ Decoder Latches
- Grouping functions on a common chip minimizes speed differential/skew between address, $\overline{RAS}$ and $\overline{CAS}$ outputs
- 3-Port 8-Bit Address Multiplexer with Schottky speed
- Burst mode, distributed refresh or transparent refresh mode determined by user
- Non-inverting address, $\overline{RAS}$ and $\overline{CAS}$ paths
- 100% product assurance screening to MIL-STD-883 requirements

FUNCTIONAL DESCRIPTION

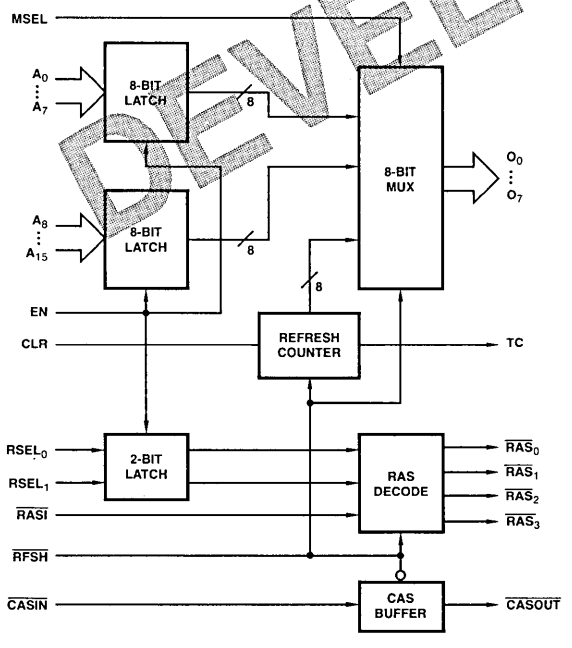
The AmZ8164 Dynamic Memory Controller replaces several MSI devices by grouping several unique functions. Two 8-bit latches capture and hold the memory address from the AmZ8000 multiplexed data and address bus. These latches and a clearable, 8-bit refresh counter feed into an 8-bit, 3-input, Schottky speed MUX, for output to the dynamic RAM address lines. The device is also compatible with Am8085 or any CPU interfacing with dynamic RAMs.

The same silicon chip also includes a special RAS decoder and CAS buffer. Placing these functions on the same chip minimizes the time skew between output functions which would otherwise be separate MSI chips, and therefore, allows a faster memory cycle time by the amount of skew eliminated.

Pulsing the active LOW refresh line, $\overline{RFSH}$, switches the MUX to the counter output, inhibits $\overline{CAS}$, and forces all four $\overline{RAS}$ decoder outputs active simultaneously. The counter is advanced at the end of the refresh cycle – the LOW-to-HIGH transition of $\overline{RFSH}$. Various refresh modes can be accommodated – for 16K or 64K RAMs and for a wide variety of processor configurations.

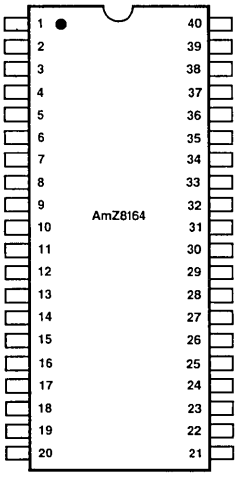
A_{15} is a dual function input which controls the refresh counter's range. For 64K RAMs it is an address input. For 16K RAMs it can be pulled to +12V through 1KΩ to terminate the refresh count at 128 instead of 256 if this is needed.

LOGIC DIAGRAM



BLI-208

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BLI-209

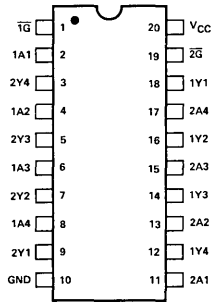
AmZ8165 • AmZ8166

Octal Dynamic Memory Drivers with Three-State Outputs

DISTINCTIVE CHARACTERISTICS

- **Controlled rise and fall characteristics**
  Internal resistors provide symmetrical drive to HIGH and LOW states, eliminating need for external series resistor.
- **Output swings designed to drive 16K and 64K RAMs**
  V_{OH} guaranteed at V_{CC} – 1.15V. Undershoot going LOW guaranteed at less than 0.5V.
- **Large capacitive drive capability**
  35mA min source or sink current at 2.0V. Propagation delays specified for 50pF and 500pF loads.
- **Pin-compatible with 'S240 and 'S244**
  Non-inverting AmZ8166 replaces 74S244; inverting AmZ8165 replaces 74S240. Faster than 'S240/244 under equivalent load.
- **No-glitch outputs**
  Outputs forced into OFF state during power up and down.

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BLI-210

FUNCTIONAL DESCRIPTION

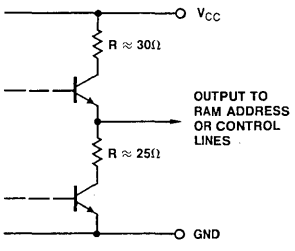
The AmZ8165 and AmZ8166 are designed and specified to drive the capacitive input characteristics of the address and control lines of MOS dynamic RAMs. The unique design of the lower output driver includes a collector resistor to control undershoot on the HIGH-to-LOW transition. The upper output driver pulls up to V_{CC} – 1.15V to be compatible with MOS memory and is designed to have a rise time symmetrical with the lower output's controlled fall time. This allows optimization of Dynamic RAM performance.

The AmZ8165 and AmZ8166 are pin-compatible with the popular 'S240 and 'S244 with identical 3-state output enable controls. The AmZ8165 has inverting drivers and the AmZ8166 has non-inverting drivers.

The inclusion of an internal resistor in the lower output driver eliminates the requirement for an external series resistor, therefore reducing package count and the board area required. The internal resistor controls the output fall and undershoot without slowing the output rise.

These devices are designed for use with the AmZ8164 Dynamic Memory Controller where large dynamic memories with highly capacitive input lines require additional buffering. Driving eight address lines or four $\overline{RAS}$ and four $\overline{CAS}$ lines with drivers on the same silicon chip also provides a significant performance advantage by minimizing skew between drivers. Each device has specified skew between drivers to improve the memory access worst case timing over the min and max t_{PD} difference of unspecified devices.

TYPICAL OUTPUT DRIVER

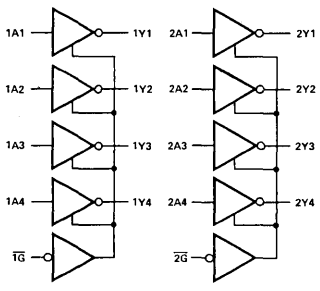


BLI-211

LOGIC DIAGRAMS

BLI-212

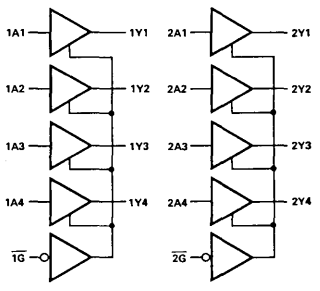
AmZ8165



Inputs		Outputs
$\overline{G}$	A	Y
H	X	Z
L	H	L
L	L	H

BLI-213

AmZ8166



Inputs		Outputs
$\overline{G}$	A	Y
H	X	Z
L	L	L
L	H	H

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential Continuous	−0.5 to +7.0V
DC Voltage Applied to Outputs for High Output State	−0.5V to +V_{CC} max
DC Input Voltage	−0.5 to +7.0V
DC Output Current, into Outputs	30mA
DC Input Current	−30 to +5.0mA

ELECTRICAL CHARACTERISTICS

The Following Conditions Apply Unless Otherwise Specified:

COM'L T_A = 0 to 70°C V_{CC} = 5.0V ±10% (MIN = 4.50V MAX = 5.50V)
MIL T_A = −55 to +125°C V_{CC} = 5.0V ±10% (MIN = 4.50V MAX = 5.50V)

DC CHARACTERISTICS OVER OPERATING RANGE

Parameters	Description		Test Conditions (Note 1)		Min	Typ (Note 2)	Max	Units
V_{OH}	Output High Voltage		V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OH} = −1mA	V_{CC}−1.15	V_{CC}−0.7V		Volts
V_{OL}	Output LOW Voltage		V_{CC} = MIN, V_{IN} = V_{IH} or V_{IL}	I_{OL} = 1mA			0.5	Volts
				I_{OL} = 12mA			0.8	
V_{IH}	Input HIGH Level		Guaranteed input logical HIGH voltage for all inputs		2.0			Volts
V_{IL}	Input LOW Level		Guaranteed input logical LOW voltage for all inputs				0.8	Volts
V_I	Input Clamp Voltage		V_{CC} = MIN, I_{IN} = −18mA				−1.2	Volts
I_{IL}	Input LOW Current		V_{CC} = MAX, V_{IN} = 0.4V				−200	μA
I_{IH}	Input HIGH Current		V_{CC} = MAX, V_{IN} = 2.7V				20	μA
I_I	Input HIGH Current		V_{CC} = MAX, V_{IN} = 7.0V				0.1	mA
I_{OZH}	Off-State Current		V_O = 2.7V				100	μA
I_{OZL}	Off-State Current		V_O = 0.4V				−200	μA
I_{OL}	Output Sink Current		V_{OL} = 2.0V		35			mA
I_{OH}	Output Source Current		V_{OH} = 2.0V		−35			mA
I_{SC}	Output Short Circuit Current (Note 3)		V_{CC} = MAX		−60 (see I_{OH})		−200	mA
I_{CC}	Supply Current	AmZ8165	All Outputs HIGH	V_{CC} = MAX Outputs Open		24	50	mA
			All Outputs LOW			86	125	
			All Outputs Hi-Z			86	125	
		AmZ8166	All Outputs HIGH	V_{CC} = MAX Outputs Open		53	75	
			All Outputs LOW			92	130	
			All Outputs Hi-Z			116	150	

Notes: 1. For conditions shown as MIN or MAX, use the appropriate value specified under Electrical Characteristics for the applicable device type.
2. Typical limits are at V_{CC} = 5.0V, 25°C ambient and maximum loading.
3. Not more than one output should be shorted at a time. Duration of the short circuit test should not exceed one second.

AmZ8165 • AmZ8166
SWITCHING CHARACTERISTICS
(T_A = +25°C, VCC = 5.0V)

Parameters	Description	Test Conditions		Min	Typ	Max	Units
t_{PLH}	Propagation Delay Time from LOW-to-HIGH Output	Figure 1 Test Circuit Figure 3 Voltage Levels and Waveforms	C_L = 0pF		6	(Note 4)	ns
			C_L = 50pF	6	9	15	
			C_L = 500pF	15	22	35	
t_{PHL}	Propagation Delay Time from HIGH-to-LOW Output		C_L = 0pF		4	(Note 4)	ns
			C_L = 50pF	6	12	20	
			C_L = 500pF	20	30	45	
t_{PLZ}	Output Disable Time from LOW, HIGH	Figures 2 and 4, S = 1			13	20	ns
t_{PHZ}		Figures 2 and 4, S = 2			8	12	
t_{PZL}	Output Enable Time from LOW, HIGH	Figures 2 and 4, S = 1			13	20	ns
t_{PZH}		Figures 2 and 4, S = 2			13	20	
t_{SKEW}	Output-to-Output Skew	Figures 1 and 3, C_L = 50pF			±0.5	±3.0 (Note 5)	ns
V_{ONP}	Output Voltage Undershoot	Figures 1 and 3, C_L = 50pF			0	−0.5	Volts

4

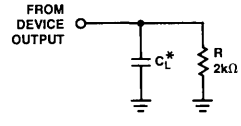
SWITCHING CHARACTERISTICS
OVER OPERATING RANGE (Note 6)

Parameters	Description	Test Conditions		COM'L T_A = 0 to 70°C V_{CC} = 5.0V ±10% Min	COM'L Max	MIL T_A = −55 to +125°C V_{CC} = 5.0V ±10% Min	MIL Max	Units
t_{PLH}	Propagation Delay Time LOW-to-HIGH Output	Figures 1 and 3	C_L = 50pF	4	20	4	20	ns
			C_L = 500pF	13	40	13	40	
t_{PHL}	Propagation Delay Time HIGH-to-LOW Output	Figures 1 and 3	C_L = 50pF	4	24	4	24	ns
			C_L = 500pF	17	50	17	50	
t_{PLZ}	Output Disable Time from LOW, HIGH	Figures 2 and 4	S = 1		24		24	ns
t_{PHZ}			S = 2		16		16	
t_{PZL}	Output Enable Time from LOW, HIGH	Figures 2 and 4	S = 1		28		28	ns
t_{PZH}			S = 2		28		28	
V_{ONP}	Output Voltage Undershoot	Figures 1 and 3, C_L = 50pF			−0.5		−0.5	Volts

Notes: 4. Typical time shown for reference only – not tested.
5. Time Skew specification is guaranteed by design but not tested.
6. AC performance over the operating temperature range is guaranteed by testing defined in Group A, Subgroup 9.

SWITCHING TEST CIRCUITS

BLI-214



*t_{pd} specified at C = 50 and 500pF.

Figure 1. Capacitive Load Switching.

BLI-215

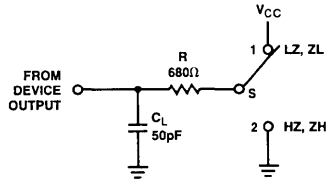
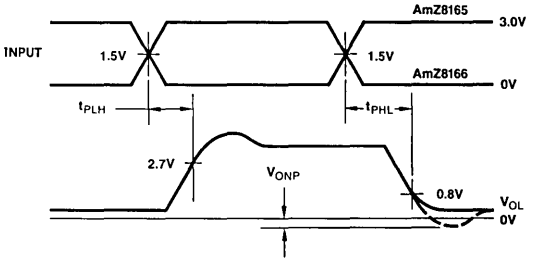


Figure 2. Three-State Enable/Disable.

TYPICAL SWITCHING CHARACTERISTICS

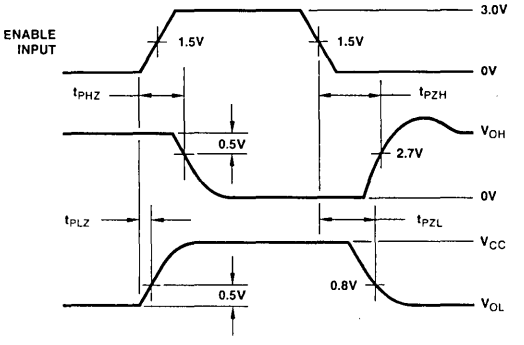
VOLTAGE WAVEFORMS



$t_r = t_f = 2.5ns$
$f = 2.5MHz$
$t_{pw} = 200ns$

BLI-217

Figure 3. Output Drive Levels.



$t_r = t_f = 2.5ns$
$f = 1MHz$
$t_{pw} = 800ns$

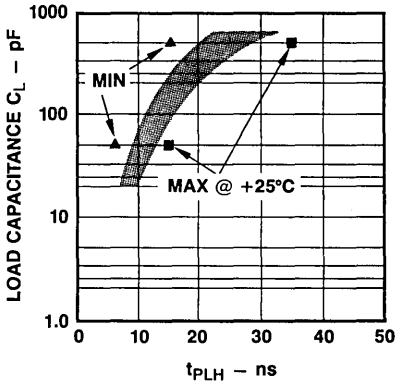
BLI-218

Figure 4. Three-State Control Levels.

The RAM Driver symmetrical output design offers significant improvement over a standard Schottky output by providing a balanced drive output impedance (≈33Ω both HIGH and LOW), and by pulling up to MOS V_{OH} levels (V_{CC} – 1.15V). External resistors, not required with the RAM Driver, protect standard Schottky drivers from error causing undershoot but also slow the output rise by adding to the internal R.

The RAM Driver is optimized to drive LOW at maximum speed based on safe undershoot control and to drive HIGH with a symmetrical speed characteristic. This is an optimum approach because the dominant RAM loading characteristic is input capacitance.

The curves shown below provide performance characteristics typical of both the inverting (AmZ8165) and non-inverting (AmZ8166) RAM Drivers.



BLI-219

Figure 5. t_{PLH} vs. C_L.

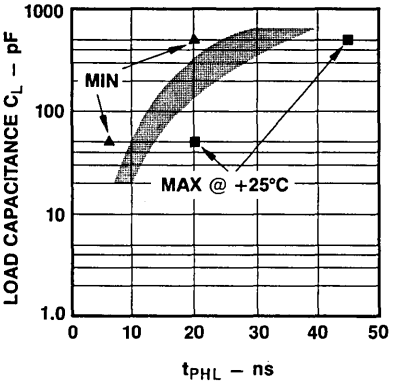
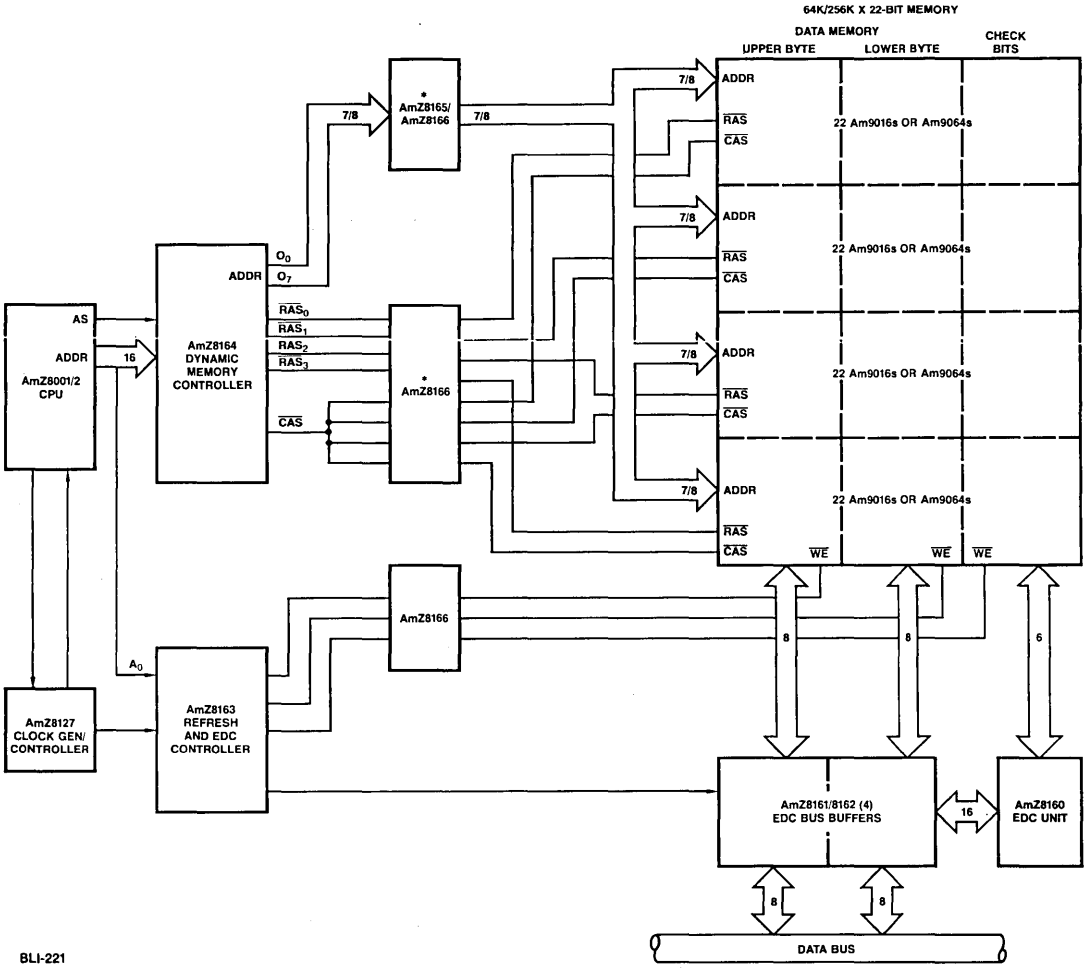


Figure 6. t_{PHL} vs. C_L.

BLI-220

The curves above depict the typical t_{PLH} and t_{PHL} for the RAM Driver outputs as a function of load capacitance. The minimums and maximums are shown for worst case design. The typical band is provided as a guide for intermediate capacitive loads.

APPLICATION

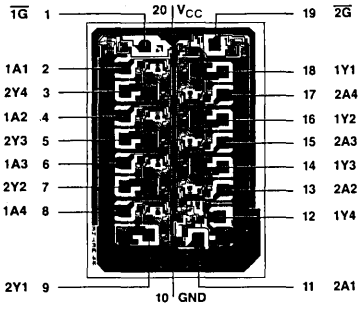


*Address and $\overline{RAS}$/$\overline{CAS}$ drivers each drive 22 RAM inputs at each output. Timing skew is minimized by using one device for address lines and one device for $\overline{RAS}$/$\overline{CAS}$, spreading the $\overline{CAS}$ loading over four drivers to equalize the capacitive load on each driver.

DYNAMIC MEMORY CONTROL WITH ERROR DETECTION AND CORRECTION

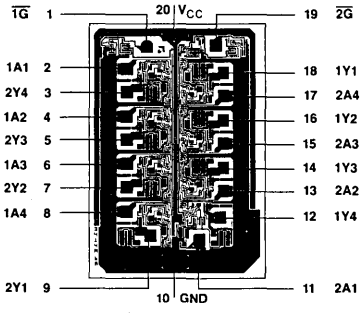
Metallization and Pad Layouts

AmZ8165



DIE SIZE 0.094" X 0.060"

AmZ8166



DIE SIZE 0.094" X 0.060"

ORDERING INFORMATION

Order the part number according to the table below to obtain the desired package, temperature range, and screening level.

AmZ8165 Order Number	AmZ8166 Order Number	Package Type	Temperature Range	Screening Level
AMZ8165PC	AMZ8166PC	P-20	C	C-1
AMZ8165DC	AmZ8166DC	D-20	C	C-1
AMZ8165DM	AMZ8166DM	D-20	M	C-3
AMZ8165XC	AMZ8166XC	Dice	C	Visual Inspection to MIL-STD-883 Method 2010B.
AMZ8165XM	AMZ8166XM	Dice	M	

Notes: 1. P = Molded DIP, D = Hermetic DIP. Number following letter is number of leads.
2. C = 0 to 70°C, V_{CC} = 4.50 to 5.50V, M = −55 to +125°C, V_{CC} = 4.50 to 5.50V.
3. Levels C-1 and C-3 conform to MIL-STD-883, Class C. Level B-3 conforms to MIL-STD-883, Class B.

AmZ8173

Octal Latch with Three-State Outputs

See AmZ8133 • AmZ8173 data sheet, listed under AmZ8133.

Am8255A/8255A-5

Programmable Peripheral Interface

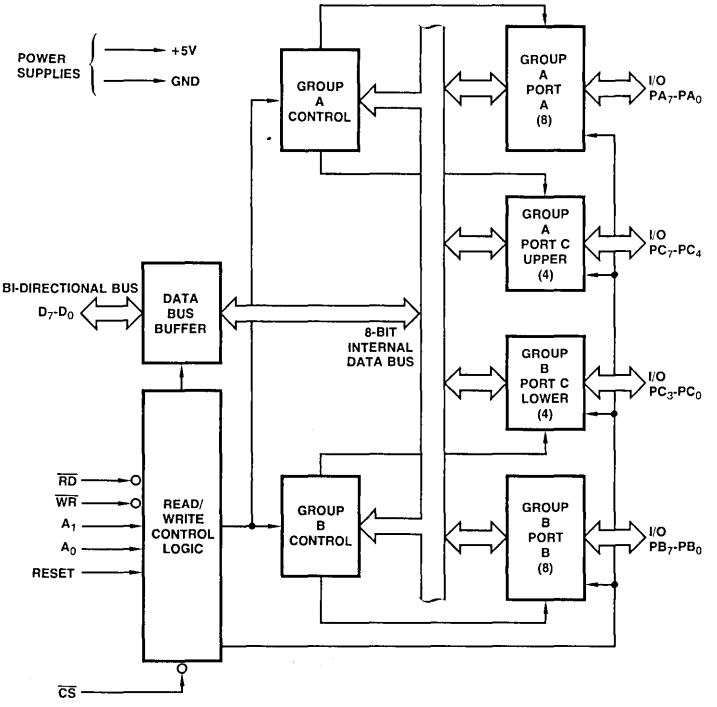
DISTINCTIVE CHARACTERISTICS

- Direct bit set/reset capability easing control application interface
- Reduces system package count
- Improved DC driving capability
- 24 programmable I/O pins
- Completely TTL compatible
- Fully compatible with 8080A and 8085A microprocessor families
- Improved timing characteristics
- Military version available

GENERAL DESCRIPTION

The Am8255A is a general purpose programmable I/O device designed for use with Am8080A and Am8085A microprocessors. It has 24 I/O pins which may be individually programmed in two groups of twelve and used in three major modes of operation. In the first mode, each group of twelve I/O pins may be programmed in sets of 4 and 8 to be input or output. In Mode 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining four pins three are used for handshaking and interrupt control signals. The third mode of operation (Mode 2) is a bi-directional bus mode which uses eight lines for a bi-directional bus, and five lines, borrowing one from the other group, for handshaking.

Am8255A BLOCK DIAGRAM

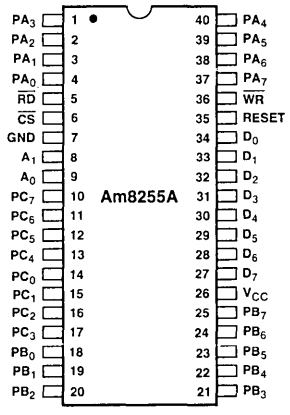


MOS-107

ORDERING INFORMATION

Package Type	Ambient Temperature Specification	Order Numbers	
Hermetic DIP	$0°C \leq T_A \leq +70°C$	AM8255ADC	AM8255A-5DC
Molded DIP		AM8255APC	AM8255A-5PC
Hermetic DIP	$-55°C \leq T_A \leq +125°C$	AM9555ADM/M8255A	

CONNECTION DIAGRAM – Top View



MOS-108

PIN NAMES

D_7-D_0	Data Bus (Bi-Directional)
Reset	Reset Input
$\overline{CS}$	Chip Select
$\overline{RD}$	Read Input
$\overline{WR}$	Write Input
A_0, A_1	Port Address
PA_7-PA_0	Port A (Bit)
PB_7-PB_0	Port B (Bit)
PC_7-PC_0	Port C (Bit)
V_{CC}	+5 Volts
GND	0 Volts

MAXIMUM RATINGS above which useful life may be impaired

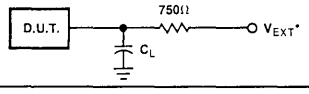
Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
V_{CC} with Respect to V_{SS}	−0.5V to +7.0V
All Signal Voltages with Respect to V_{SS}	−0.5V to +7.0V
Power Dissipation	1.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

CAPACITANCE T_A = 25°C; V_{CC} = GND = 0V

Parameter	Description	Test Conditions	Min.	Typ.	Max.	Unit
C_{IN}	Input Capacitance	fc = 1MHz			10	pF
$C_{I/O}$	I/O Capacitance	Unmeasured pins returned to GND			20	pF

TEST LOAD CIRCUIT (FOR DATA BUS)



*V_{EXT} is set at various voltages during testing to guarantee the specification.

OPERATING RANGE

Part Number	T_A	V_{CC}	V_{SS}
Am8255ACC/PC	0°C to +70°C	+5V ±5%	0V
Am9555ADM	−55°C to +125°C	+5V ±10%	0V

DC CHARACTERISTICS Over Operating Range

Parameter	Description	Test Conditions	Min.	Typ.	Max.	Units
V_{IL}	Input Low Voltage		−0.5		0.8	Volts
V_{IH}	Input High Voltage		2.0		V_{CC}	Volts
V_{OL}(DB)	Output Low Voltage (Data Bus)	I_{OL} = 2.5mA			0.45	Volts
V_{OL}(PER)	Output Low Voltage (Peripheral Port)	I_{OL} = 1.7mA			0.45	Volts
V_{OH}(DB)	Output High Voltage (Data Bus)	I_{OH} = −400μA	2.4			Volts
V_{OH}(PER)	Output High Voltage (Peripheral Port)	I_{OH} = −200μA	2.4			Volts
I_{DAR} (Note 1)	Darlington Drive Current	R_{EXT} = 750Ω; V_{EXT} = 1.5V	−1.0		−4.0	mA
I_{CC}	Power Supply Current				120	mA
I_{IL}	Input Load Current	V_{IN} = V_{CC} to 0V			±10	μA
I_{OFL}	Output Float Leakage	V_{OUT} = V_{CC} to 0V			±10	μA

Note 1: Available on any 8 pins from Port B and C.

AC CHARACTERISTICS Over Operating Range

BUS PARAMETERS:

Read:

Parameter	Description	Am8255A Min.	Am8255A Max.	Am8255A-5 Min.	Am8255A-5 Max.	Am9555ADM Min.	Am9555ADM Max.	Unit
t_{AR}	Address Stable Before READ	0		0		0		ns
t_{RA}	Address Stable After READ	0		0		0		ns
t_{RR}	READ Pulse Width	300		300		300		ns
t_{RD}	Data Valid From READ (Note 1)		250		200		250	ns
t_{DF}	Data Float After READ	10	150	10	100	10	150	ns
t_{RV}	Time Between READs and/or WRITEs	850		850		850		ns

Write:

Parameter	Description	Am8255A Min.	Am8255A Max.	Am8255A-5 Min.	Am8255A-5 Max.	Am9555ADM Min.	Am9555ADM Max.	Unit
t_{AW}	Address Stable Before WRITE	0		0		0		ns
t_{WA}	Address Stable After WRITE	20		20		20		ns
t_{WW}	WRITE Pulse Width	400		300		400		ns
t_{DW}	Data Valid to WRITE (T.E.)	100		100		100		ns
t_{WD}	Data Valid After WRITE	30		30		30		ns

Other Timings:

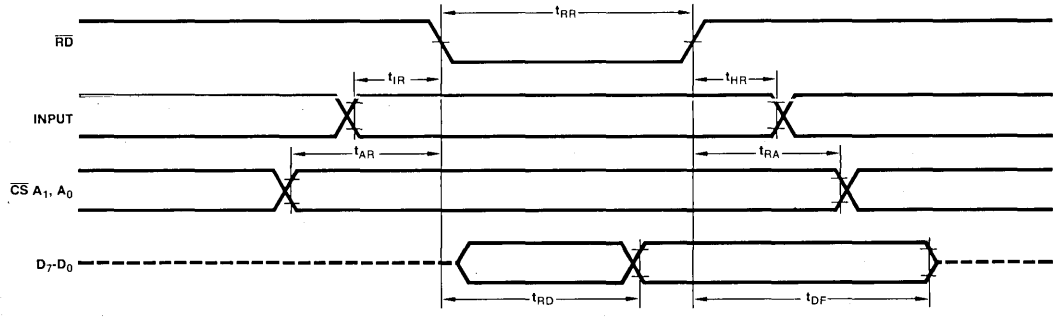
Parameter	Description	Am8255A Min.	Am8255A Max.	Am8255A-5 Min.	Am8255A-5 Max.	Am9555ADM Min.	Am9555ADM Max.	Unit
t_{WB}	WR = 1 to Output (Note 1)		350		350		350	ns
t_{IR}	Peripheral Data Before RD	0		0		0		ns
t_{HR}	Peripheral Data After RD	0		0		0		ns
t_{AK}	ACK Pulse Width	300		300		300		ns
t_{ST}	STB Pulse Width	500		500		500		ns
t_{PS}	Per. Data Before T.E. of STB	0		0		0		ns
t_{PH}	Per. Data After T.E. of STB	180		180		180		ns
t_{AD}	ACK = 0 to Output (Note 1)		300		300		300	ns
t_{KD}	ACK = 1 to Output Float	20	250	20	250	20	250	ns
t_{WOB}	WR = 1 to OBF = 0 (Note 1)		650		650		650	ns
t_{AOB}	ACK = 0 to OBF = 1 (Note 1)		350		350		350	ns
t_{SIB}	STB = 0 to IBF = 1 (Note 1)		300		300		300	ns
t_{RIB}	RD = 1 to IBF = 0 (Note 1)		300		300		300	ns
t_{RIT}	RD = 0 to INTR = 0 (Note 1)		400		400		400	ns
t_{SIT}	STB = 1 to INTR = 1 (Note 1)		300		300		300	ns
t_{AIT}	ACK = 1 to INTR = 1 (Note 1)		350		350		350	ns
t_{WIT}	WR = 0 to INTR = 0 (Note 1)		850		850		850	ns

Notes: 1. Test Conditions: Am8255A/Am9555ADM: C_L = 100pF; Am8255A-5: C_L = 150pF.
2. Period of Reset pulse must be at least 50μs during or after power on. Subsequent Reset pulse can be 500ns min.

WAVEFORMS

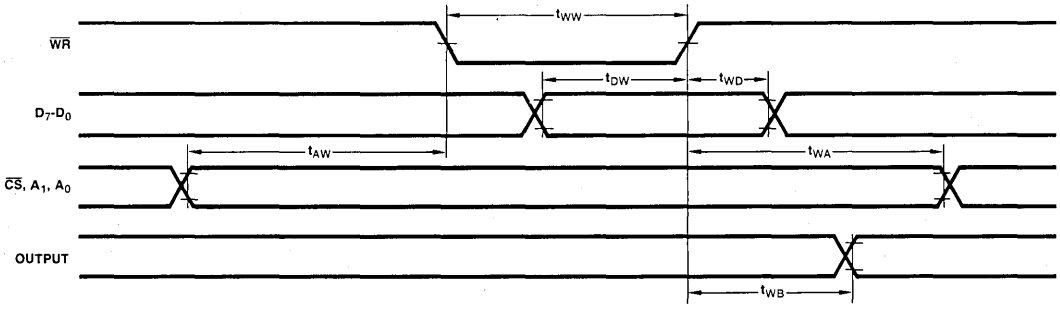
Input Waveforms For A.C. Tests

MOS-109



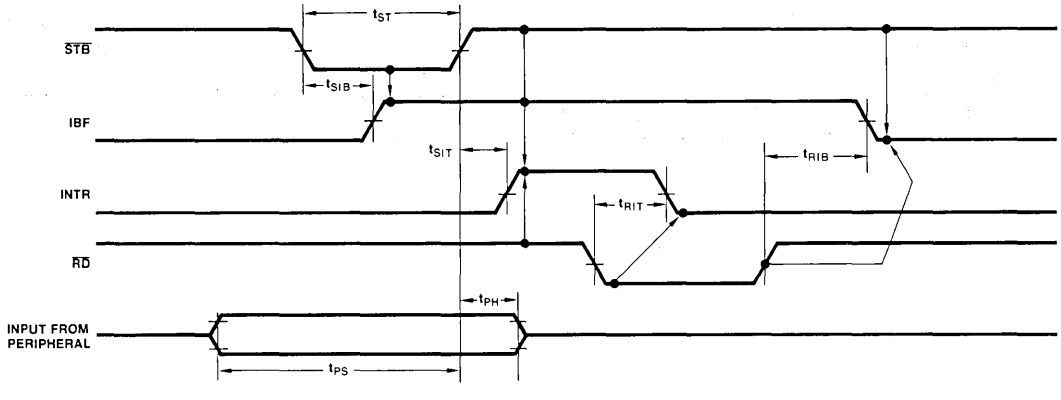
Mode 0 (Basic Input)

MOS-110



Mode 0 (Basic Output)

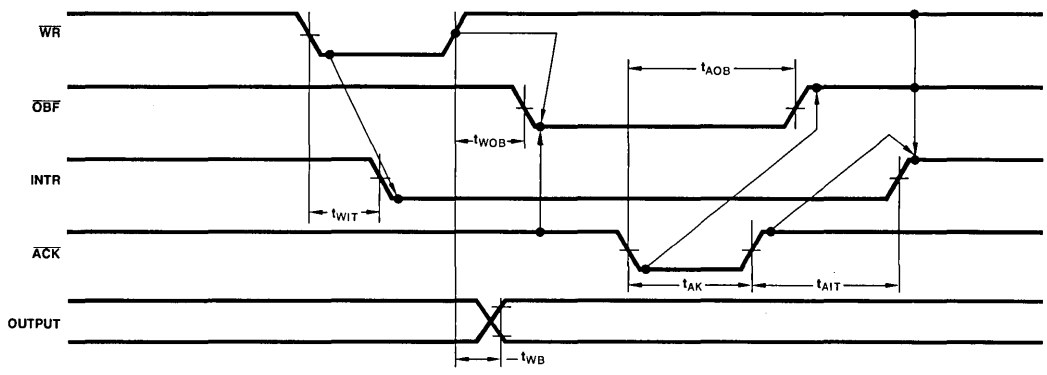
MOS-111



Mode 1 (Strobed Input)

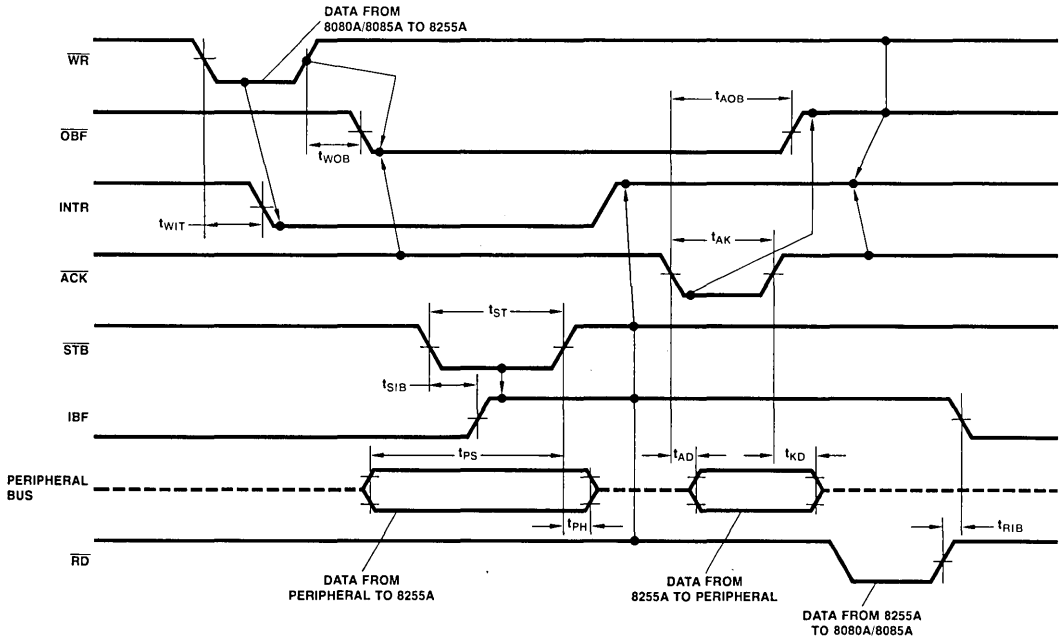
MOS-112

WAVEFORMS (Cont.)



Mode 1 (Strobed Output)

MOS-113



Note: Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$ and $\overline{STB}$ occurs before $\overline{RD}$ is permissible.
(INTR = IBF • $\overline{MASK}$ • $\overline{STB}$ • $\overline{RD}$ + $\overline{OBF}$ • $\overline{MASK}$ • $\overline{ACK}$ • $\overline{WR}$)

Mode 2 (Bi-directional)

MOS-114

Am9511A
Arithmetic Processor

DISTINCTIVE CHARACTERISTICS

- Replaces Am9511
- Fixed point 16 and 32 bit operations
- Floating point 32 bit operations
- Binary data formats
- Add, Subtract, Multiply and Divide
- Trigonometric and inverse trigonometric functions
- Square roots, logarithms, exponentiation
- Float to fixed and fixed to float conversions
- Stack-oriented operand storage
- DMA or programmed I/O data transfers
- End signal simplifies concurrent processing
- Synchronous/Asynchronous operations
- General purpose 8-bit data bus interface
- Standard 24 pin package
- +12 volt and +5 volt power supplies
- Advanced N-channel silicon gate MOS technology
- 100% MIL-STD-883 reliability assurance testing

GENERAL DESCRIPTION

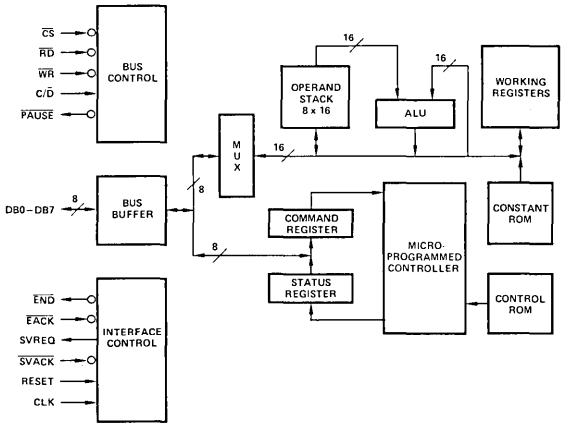
The Am9511A Arithmetic Processing Unit (APU) is a monolithic MOS/LSI device that provides high performance fixed and floating point arithmetic and a variety of floating point trigonometric and mathematical operations. It may be used to enhance the computational capability of a wide variety of processor-oriented systems.

All transfers, including operand, result, status and command information, take place over an 8-bit bidirectional data bus. Operands are pushed onto an internal stack and a command is issued to perform operations on the data in the stack. Results are then available to be retrieved from the stack, or additional commands may be entered.

Transfers to and from the APU may be handled by the associated processor using conventional programmed I/O, or may be handled by a direct memory access controller for improved performance. Upon completion of each command, the APU issues an end of execution signal that may be used as an interrupt by the CPU to help coordinate program execution.

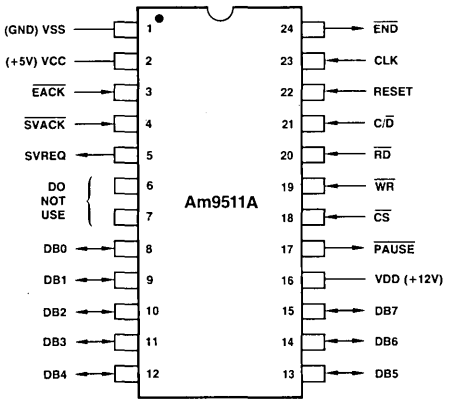
4

BLOCK DIAGRAM



MOS-046

CONNECTION DIAGRAM
Top View



Pin 1 is marked for orientation.

MOS-047

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency	
		2MHz	3MHz
Hermetic DIP	0°C ≤ T_A ≤ +70°C	Am9511ADC	Am9511A-1DC
	−55°C ≤ T_A ≤ +125°C	Am9511ADM	Am9511A-1DM

INTERFACE SIGNAL DESCRIPTION

VCC: +5V Power Supply
VDD: +12V Power Supply
VSS: Ground

CLK (Clock, Input)

An external timing source connected to the CLK input provides the necessary clocking. The CLK input can be asynchronous to the $\overline{RD}$ and $\overline{WR}$ control signals.

RESET (Reset, Input)

A HIGH on this input causes initialization. Reset terminates any operation in progress, and clears the status register to zero. The internal stack pointer is initialized and the contents of the stack may be affected but the command register is not affected by the reset operation. After a reset the $\overline{END}$ output, and the SVREQ output will be LOW. For proper initialization, the RESET input must be HIGH for at least five CLK periods following stable power supply voltages and stable clock.

C/$\overline{D}$ (Command/Data Select, Input)

The C/$\overline{D}$ input together with the $\overline{RD}$ and $\overline{WR}$ inputs determines the type of transfer to be performed on the data bus as follows:

C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	Function
L	H	L	Push data byte into the stack
L	L	H	Pop data byte from the stack
H	H	L	Enter command byte from the data bus
H	L	H	Read Status
X	L	L	Undefined

L = LOW
H = HIGH
X = DON'T CARE

$\overline{END}$ (End of Execution, Output)

A LOW on this output indicates that execution of the current command is complete. This output will be cleared HIGH by activating the $\overline{EACK}$ input LOW or performing any read or write operation or device initialization using the RESET. If $\overline{EACK}$ is tied LOW, the $\overline{END}$ output will be a pulse (see $\overline{EACK}$ description). This is an open drain output and requires a pull up to +5V.

Reading the status register while a command execution is in progress is allowed. However any read or write operation clears the flip-flop that generates the $\overline{END}$ output. Thus such continuous reading could conflict with internal logic setting the $\overline{END}$ flip-flop at the completion of command execution.

$\overline{EACK}$ (End Acknowledge, Output)

This input when LOW makes the $\overline{END}$ output go LOW. As mentioned earlier HIGH on the $\overline{END}$ output signals completion of a command execution. The $\overline{END}$ output signal is derived from an internal flip-flop which is clocked at the completion of a command. This flip-flop is clocked to the reset state when $\overline{EACK}$ is LOW. Consequently, if the $\overline{EACK}$ is tied LOW, the $\overline{END}$ output will be a pulse that is approximately one CLK period wide.

SVREQ (Service Request, Output)

A HIGH on this output indicates completion of a command. In this sense this output is same as the $\overline{END}$ output. However, whether the SVREQ output will go HIGH at the completion of a command or not is determined by a service request bit in the command register. This bit must be 1 for SVREQ to go HIGH. The SVREQ can be cleared (i.e., go LOW) by activating the $\overline{SVACK}$ input LOW or initializing the device using the RESET. Also, the SVREQ will be automatically cleared after completion of any command that has the service request bit as 0.

$\overline{SVACK}$ (Service Acknowledge, Input)

A LOW on this input activates the reset input of the flip-flop generating the SVREQ output. If the $\overline{SVACK}$ input is permanently tied LOW, it will conflict with the internal setting of the flip-flop to generate the SVREQ output. Thus the SVREQ indication cannot be relied upon if the $\overline{SVACK}$ is tied LOW.

DB0-DB7 (Bidirectional Data Bus, Input/Output)

These eight bidirectional lines are used to transfer command, status and operand information between the device and the host processor. DB0 is the least significant and DB7 is the most significant bit position. HIGH on the data bus line corresponds to 1 and LOW corresponds to 0.

When pushing operands on the stack using the data bus, the least significant byte must be pushed first and most significant byte last. When popping the stack to read the result of an operation, the most significant byte will be available on the data bus first and the least significant byte will be the last. Moreover, for pushing operands and popping results, the number of transactions must be equal to the proper number of bytes appropriate for the chosen format. Otherwise, the internal byte pointer will not be aligned properly. The Am9511A single precision format requires 2 bytes, double precision and floating-point formats require 4 bytes.

$\overline{CS}$ (Chip Select, Input)

This input must be LOW to accomplish any read or write operation to the Am9511A.

To perform a write operation data is presented on DB0 through DB7 lines, C/$\overline{D}$ is driven to an appropriate level and the $\overline{CS}$ input is made LOW. However, actual writing into the Am9511A cannot start until $\overline{WR}$ is made LOW. After initiating the write operation by a $\overline{WR}$ HIGH to LOW transition, the $\overline{PAUSE}$ output will go LOW momentarily (TPPWW).

The $\overline{WR}$ input can go HIGH after $\overline{PAUSE}$ goes HIGH. The data lines, C/$\overline{D}$ input and the $\overline{CS}$ input can change when appropriate hold time requirements are satisfied. See write timing diagram for details.

To perform a read operation an appropriate logic level is established on the C/$\overline{D}$ input and $\overline{CS}$ is made LOW. The Read operation does not start until the $\overline{RD}$ input goes LOW. $\overline{PAUSE}$ will go LOW for a period of TPPWR. When $\overline{PAUSE}$ goes back HIGH again, it indicates that read operation is complete and the required information is available on the DB0 through DB7 lines. This information will remain on the data lines as long as $\overline{RD}$ input is LOW. The $\overline{RD}$ input can return HIGH anytime after $\overline{PAUSE}$ goes HIGH. The $\overline{CS}$ input and C/$\overline{D}$ inputs can change anytime after $\overline{RD}$ returns HIGH. See read timing diagram for details. The $\overline{CS}$ must have a HIGH to LOW transition for every READ or WRITE operation.

$\overline{RD}$ (Read, Input)

A LOW on this input is used to read information from an internal location and gate that information on to the data bus. The $\overline{CS}$ input must be LOW to accomplish the read operation. The C/$\overline{D}$ input determines what internal location is of interest. See C/$\overline{D}$, $\overline{CS}$ input descriptions and read timing diagram for details. If the END output was LOW, performing any read operation will make the $\overline{END}$ output go HIGH after the HIGH to LOW transition of the $\overline{RD}$ input (assuming $\overline{CS}$ is LOW).

$\overline{WR}$ (Write, Input)

A LOW on this input is used to transfer information from the data bus into an internal location. The $\overline{CS}$ must be LOW to accomplish the write operation. The C/$\overline{D}$ determines which internal location is to be written. See C/$\overline{D}$, $\overline{CS}$ input descriptions and write timing diagram for details.

If the $\overline{END}$ output was LOW, performing any write operation will make the $\overline{END}$ output go HIGH after the LOW to HIGH transition of the $\overline{WR}$ input (assuming $\overline{CS}$ is LOW).

$\overline{PAUSE}$ (Pause, Output)

This output is a handshake signal used while performing read or write transactions with the Am9511A. A LOW at this output indicates that the Am9511A has not yet completed its information transfer with the host over the data bus. During a read operation, after $\overline{CS}$ went LOW, the $\overline{PAUSE}$ will become LOW shortly (TRP) after $\overline{RD}$ goes LOW. $\overline{PAUSE}$ will return high only after the data bus contains valid output data. The $\overline{CS}$ and $\overline{RD}$ should remain LOW when $\overline{PAUSE}$ is LOW. The RD may go high anytime after $\overline{PAUSE}$ goes HIGH. During a write operation, after $\overline{CS}$ went LOW, the $\overline{PAUSE}$ will be LOW for a very short duration (TPPWN) after $\overline{WR}$ goes LOW. Since the minimum of TPPWW is 0, the $\overline{PAUSE}$ may not go LOW at all for fast devices. $\overline{WR}$ may go HIGH anytime after $\overline{PAUSE}$ goes HIGH.

FUNCTIONAL DESCRIPTION

Major functional units of the Am9511A are shown in the block diagram. The Am9511A employs a microprogram controlled stack oriented architecture with 16-bit wide data paths.

The Arithmetic Logic Unit (ALU) receives one of its operands from the Operand Stack. This stack is an 8-word by 16-bit 2-port memory with last in-first out (LIFO) attributes. The second operand to the ALU is supplied by the internal 16-bit bus. In addition to supplying the second operand, this bidirectional bus also carries the results from the output of the ALU when required. Writing into the Operand Stack takes place from this internal 16-bit bus when required. Also connected to this bus are the Constant ROM and Working Registers. The ROM provides the required constants to perform the mathematical operations (Chebyshev Algorithms) while the Working Registers provide storage for the intermediate values during command execution.

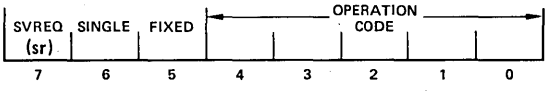
Communication between the external world and the Am9511A takes place on eight bidirectional input/output lines DB0 through DB7 (Data Bus). These signals are gated to the internal eight-bit bus through appropriate interface and buffer circuitry. Multiplexing facilities exist for bidirectional communication between the internal eight and sixteen-bit buses. The Status Register and Command Register are also accessible via the eight-bit bus.

The Am9511A operations are controlled by the microprogram contained in the Control ROM. The Program Counter supplies the microprogram addresses and can be partially loaded from the Command Register. Associated with the Program Counter is the Subroutine Stack where return addresses are held during subroutine calls in the microprogram. The Microinstruction Register holds the current microinstruction being executed. This register facilitates pipelined microprogram execution. The Instruction Decode logic generates various internal control signals needed for the Am9511A operation.

The Interface Control logic receives several external inputs and provides handshake related outputs to facilitate interfacing the Am9511A to microprocessors.

COMMAND FORMAT

Each command entered into the Am9511A consists of a single 8-bit byte having the format illustrated below:



Bits 0-4 select the operation to be performed as shown in the table. Bits 5-6 select the data format for the operation. If bit 5 is a 1, a fixed point data format is specified. If bit 5 is a 0, floating point format is specified. Bit 6 selects the precision of the data to be operated on by fixed point commands (if bit 5 = 0, bit 6 must be 0). If bit 6 is a 1, single-precision (16-bit) operands are indicated; if bit 6 is a 0, double-precision (32-bit) operands are indicated. Results are undefined for all illegal combinations of bits in the command byte. Bit 7 indicates whether a service request is to be issued after the command is executed. If bit 7 is a 1, the service request output (SVREQ) will go high at the conclusion of the command and will remain high until reset by a low level on the service acknowledge pin ($\overline{SVACK}$) or until completion of execution of a succeeding command where bit 7 is 0. Each command issued to the Am9511A requests post execution service based upon the state of bit 7 in the command byte. When bit 7 is a 0, SVREQ remains low.

COMMAND SUMMARY

Command Code								Command Mnemonic	Command Description
7	6	5	4	3	2	1	0		
FIXED-POINT 16-BIT									
sr	1	1	0	1	1	0	0	SADD	Add TOS to NOS. Result to NOS. Pop Stack.
sr	1	1	0	1	1	0	1	SSUB	Subtract TOS from NOS. Result to NOS. Pop Stack.
sr	1	1	0	1	1	1	0	SMUL	Multiply NOS by TOS. Lower half of result to NOS. Pop Stack.
sr	1	1	1	0	1	1	0	SMUU	Multiply NOS by TOS. Upper half of result to NOS. Pop Stack.
sr	1	1	0	1	1	1	1	SDIV	Divide NOS by TOS. Result to NOS. Pop Stack.
FIXED-POINT 32-BIT									
sr	0	1	0	1	1	0	0	DADD	Add TOS to NOS. Result to NOS. Pop Stack.
sr	0	1	0	1	1	0	1	DSUB	Subtract TOS from NOS. Result to NOS. Pop Stack.
sr	0	1	0	1	1	1	0	DMUL	Multiply NOS by TOS. Lower half of result to NOS. Pop Stack.
sr	0	1	1	0	1	1	0	DMUU	Multiply NOS by TOS. Upper half of result to NOS. Pop Stack.
sr	0	1	0	1	1	1	1	DDIV	Divide NOS by TOS. Result to NOS. Pop Stack.
FLOATING-POINT 32-BIT									
sr	0	0	1	0	0	0	0	FADD	Add TOS to NOS. Result to NOS. Pop Stack.
sr	0	0	1	0	0	0	1	FSUB	Subtract TOS from NOS. Result to NOS. Pop Stack.
sr	0	0	1	0	0	1	0	FMUL	Multiply NOS by TOS. Result to NOS. Pop Stack.
sr	0	0	1	0	0	1	1	FDIV	Divide NOS by TOS. Result to NOS. Pop Stack.
DERIVED FLOATING-POINT FUNCTIONS									
sr	0	0	0	0	0	0	1	SQRT	Square Root of TOS. Result in TOS.
sr	0	0	0	0	0	1	0	SIN	Sine of TOS. Result in TOS.
sr	0	0	0	0	0	1	1	COS	Cosine of TOS. Result in TOS.
sr	0	0	0	0	1	0	0	TAN	Tangent of TOS. Result in TOS.
sr	0	0	0	0	1	0	1	ASIN	Inverse Sine of TOS. Result in TOS.
sr	0	0	0	0	1	1	0	ACOS	Inverse Cosine of TOS. Result in TOS.
sr	0	0	0	0	1	1	1	ATAN	Inverse Tangent of TOS. Result in TOS.
sr	0	0	0	1	0	0	0	LOG	Common Logarithm (base 10) of TOS. Result in TOS.
sr	0	0	0	1	0	0	1	LN	Natural Logarithm (base e) of TOS. Result in TOS.
sr	0	0	0	1	0	1	0	EXP	Exponential (e^x) of TOS. Result in TOS.
sr	0	0	0	1	0	1	1	PWR	NOS raised to the power in TOS. Result in NOS. Pop Stack.
DATA MANIPULATION COMMANDS									
sr	0	0	0	0	0	0	0	NOP	No Operation
sr	0	0	1	1	1	1	1	FIXS	Convert TOS from floating point to 16-bit fixed point format.
sr	0	0	1	1	1	1	0	FIXD	Convert TOS from floating point to 32-bit fixed point format.
sr	0	0	1	1	1	0	1	FLTS	Convert TOS from 16-bit fixed point to floating point format.
sr	0	0	1	1	1	0	0	FLTD	Convert TOS from 32-bit fixed point to floating point format.
sr	1	1	1	0	1	0	0	CHSS	Change sign of 16-bit fixed point operand on TOS.
sr	0	1	1	0	1	0	0	CHSD	Change sign of 32-bit fixed point operand on TOS.
sr	0	0	1	0	1	0	1	CHSF	Change sign of floating point operand on TOS.
sr	1	1	1	0	1	1	1	PTOS	Push 16-bit fixed point operand on TOS to NOS (Copy)
sr	0	1	1	0	1	1	1	PTOD	Push 32-bit fixed point operand on TOS to NOS. (Copy)
sr	0	0	1	0	1	1	1	PTOF	Push floating point operand on TOS to NOS. (Copy)
sr	1	1	1	1	0	0	0	POPS	Pop 16-bit fixed point operand from TOS. NOS becomes TOS.
sr	0	1	1	1	0	0	0	POPD	Pop 32-bit fixed point operand from TOS. NOS becomes TOS.
sr	0	0	1	1	0	0	0	POPF	Pop floating point operand from TOS. NOS becomes TOS.
sr	1	1	1	1	0	0	1	XCHS	Exchange 16-bit fixed point operands TOS and NOS.
sr	0	1	1	1	0	0	1	XCHD	Exchange 32-bit fixed point operands TOS and NOS.
sr	0	0	1	1	0	0	1	XCHF	Exchange floating point operands TOS and NOS.
sr	0	0	1	1	0	1	0	PUPI	Push floating point constant "π" onto TOS. Previous TOS becomes NOS.

NOTES:

1. TOS means Top of Stack. NOS means Next on Stack.
2. AMD Application Brief "Algorithm Details for the Am9511A APU" provides detailed descriptions of each command function, including data ranges, accuracies, stack configurations, etc.
3. Many commands destroy one stack location (bottom of stack) during development of the result. The derived functions may destroy several stack locations. See Application Brief for details.
4. The trigonometric functions handle angles in radians, not degrees.
5. No remainder is available for the fixed-point divide functions.
6. Results will be undefined for any combination of command coding bits not specified in this table.

COMMAND INITIATION

After properly positioning the required operands on the stack, a command may be issued. The procedure for initiating a command execution is as follows:

1. Enter the appropriate command on the DB0-DB7 lines.
2. Establish HIGH on the C/$\overline{D}$ input.
3. Establish LOW on the $\overline{CS}$ input.
4. Establish LOW on the $\overline{WR}$ input after an appropriate set up time (see timing diagrams).
5. Sometime after the HIGH to LOW level transition of $\overline{WR}$ input, the $\overline{PAUSE}$ output will become LOW. After a delay of TPPWW, it will go HIGH to acknowledge the write operation. The WR input can return to HIGH anytime after $\overline{PAUSE}$ going HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs are allowed to change after the hold time requirements are satisfied (see timing diagram).

An attempt to issue a new command while the current command execution is in progress is allowed. Under these circumstances, the $\overline{PAUSE}$ output will not go HIGH until the current command execution is completed.

OPERAND ENTRY

The Am9511A commands operate on the operands located at the TOS and NOS and results are returned to the stack at NOS and then popped to TOS. The operands required for the Am9511A are one of three formats – single precision fixed-point (2 bytes), double precision fixed-point (4 bytes) or floating-point (4 bytes). The result of an operation has the same format as the operands except for float to fix or fix to float commands.

Operands are always entered into the stack least significant byte first and most significant byte last. The following procedure must be followed to enter operands onto the stack:

1. The lower significant operand byte is established on the DB0-DB7 lines.
2. A LOW is established on the C/$\overline{D}$ input to specify that data is to be entered into the stack.
3. The $\overline{CS}$ input is made LOW.
4. After appropriate set up time (see timing diagrams), the $\overline{WR}$ input is made LOW. The $\overline{PAUSE}$ output will become LOW.
5. Sometime after this event, the $\overline{PAUSE}$ will return HIGH to indicate that the write operation has been acknowledged.
6. Anytime after the $\overline{PAUSE}$ output goes HIGH the $\overline{WR}$ input can be made HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagrams).

The above procedure must be repeated until all bytes of the operand are pushed into the stack. It should be noted that for single precision fixed-point operands 2 bytes should be pushed and 4 bytes must be pushed for double precision fixed-point or floating-point. Not pushing all the bytes of a quantity will result in byte pointer misalignment.

The Am9511A stack can accommodate 8 single precision fixed-point quantities or 4 double precision fixed-point or floating-point quantities. Pushing more quantities than the capacity of the stack will result in loss of data which is usual with any LIFO stack.

DATA REMOVAL

Result from an operation will be available at the TOS. Results can be transferred from the stack to the data bus by reading the stack. When the stack is popped for results, the most significant byte is available first and the least significant byte last. A result is always of the same precision as the operands that produced it except for format conversion commands. Thus when the result is taken from the stack, the total number of bytes popped out should be appropriate with the precision – single precision results are 2 bytes and double precision and floating-point results are 4 bytes. The following procedure must be used for reading the result from the stack:

1. A LOW is established on the C/$\overline{D}$ input.
2. The $\overline{CS}$ input is made LOW.
3. After appropriate set up time (see timing diagrams), the $\overline{RD}$ input is made LOW. The $\overline{PAUSE}$ will become LOW.
4. Sometime after this, $\overline{PAUSE}$ will return HIGH indicating that the data is available on the DB0-DB7 lines. This data will remain on the DB0-DB7 lines as long as the $\overline{RD}$ input remains LOW.
5. Anytime after $\overline{PAUSE}$ goes HIGH, the $\overline{RD}$ input can return HIGH to complete transaction.
6. The $\overline{CS}$ and C/$\overline{D}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagram).
7. Repeat this procedure until all bytes appropriate for the precision of the result are popped out.

Reading of the stack does not alter its data; it only adjusts the byte pointer. If more data is popped than the capacity of the stack, the internal byte pointer will wrap around and older data will be read again, consistent with the LIFO stack.

STATUS READ

The Am9511A status register can be read without any regard to whether a command is in progress or not. The only implication that has to be considered is the effect this might have on the END output discussed in the signal descriptions.

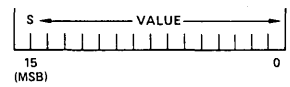
The following procedure must be followed to accomplish status register reading.

1. Establish HIGH on the C/$\overline{D}$ input.
2. Establish LOW on the $\overline{CS}$ input.
3. After appropriate set up time (see timing diagram) $\overline{RD}$ input is made LOW. The $\overline{PAUSE}$ will become LOW.
4. Sometime after the HIGH to LOW transition of $\overline{RD}$ input, the $\overline{PAUSE}$ will become HIGH indicating that status register contents are available on the DB0-DB7 lines. The status data will remain on DB0-DB7 as long as $\overline{RD}$ input is LOW.
5. The $\overline{RD}$ input can be returned HIGH anytime after $\overline{PAUSE}$ goes HIGH.
6. The C/$\overline{D}$ input and $\overline{CS}$ input can change after satisfying appropriate hold time requirements (see timing diagram).

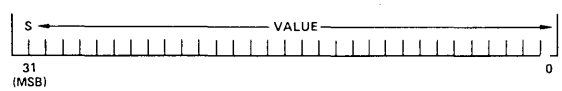
DATA FORMATS

The Am9511A Arithmetic Processing Unit handles operands in both fixed-point and floating-point formats. Fixed-point operands may be represented in either single (16-bit operands) or double precision (32-bit operands), and are always represented as binary, two's complement values.

16-BIT FIXED-POINT FORMAT



32-BIT FIXED-POINT FORMAT



The sign (positive or negative) of the operand is located in the most significant bit (MSB). Positive values are represented by a sign bit of zero (S = 0). Negative values are represented by the two's complement of the corresponding positive value with a sign bit equal to 1 (S = 1). The range of values that may be accommodated by each of these formats is −32,768 to +32,767 for single precision and −2,147,483,648 to +2,147,483,647 for double precision.

Floating point binary values are represented in a format that permits arithmetic to be performed in a fashion analogous to operations with decimal values expressed in scientific notation.

$$(5.83 \times 10^{2})(8.16 \times 10^{1}) = (4.75728 \times 10^{4})$$

In the decimal system, data may be expressed as values between 0 and 10 times 10 raised to a power that effectively shifts the implied decimal point right or left the number of places necessary to express the result in conventional form (e.g., 47,572.8). The value-portion of the data is called the mantissa. The exponent may be either negative or positive.

The concept of floating point notation has both a gain and a loss associated with it. The gain is the ability to represent the significant digits of data with values spanning a large dynamic range limited only by the capacity of the exponent field. For example, in decimal notation if the exponent field is two digits wide, and the mantissa is five digits, a range of values (positive or negative) from 1.0000×10^{-99} to $9.9999 \times 10^{+99}$ can be accommodated. The loss is that only the significant digits of the value can be represented. Thus there is no distinction in this representation between the values 123451 and 123452, for example, since each would be expressed as: 1.2345×10^{5}. The sixth digit has been discarded. In most applications where the dynamic range of values to be represented is large, the loss of significance, and hence accuracy of results, is a minor consideration. For greater precision a fixed point format could be chosen, although with a loss of potential dynamic range.

The Am9511 is a binary arithmetic processor and requires that floating point data be represented by a fractional mantissa value between .5 and 1 multiplied by 2 raised to an appropriate power. This is expressed as follows:

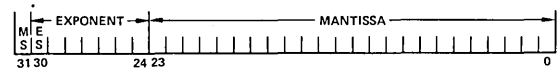
$$\text{value} = \text{mantissa} \times 2^{\text{exponent}}$$

For example, the value 100.5 expressed in this form is 0.11001001×2^{7}. The decimal equivalent of this value may be computed by summing the components (powers of two) of the mantissa and then multiplying by the exponent as shown below:

$$\begin{aligned}\text{value} &= (2^{-1} + 2^{-2} + 2^{-5} + 2^{-8}) \times 2^{7} \\ &= (0.5 + 0.25 + 0.03125 + 0.00290625) \times 128 \\ &= 0.78515625 \times 128 \\ &= 100.5\end{aligned}$$

FLOATING POINT FORMAT

The format for floating-point values in the Am9511A is given below. The mantissa is expressed as a 24-bit (fractional) value; the exponent is expressed as an unbiased two's complement 7-bit value having a range of −64 to +63. The most significant bit is the sign of the mantissa (0 = positive, 1 = negative), for a total of 32 bits. The binary point is assumed to be to the left of the most significant mantissa bit (bit 23). All floating-point data values must be normalized. Bit 23 must be equal to 1, except for the value zero, which is represented by all zeros.



The range of values that can be represented in this format is $\pm(2.7 \times 10^{-20}$ to $9.2 \times 10^{18})$ and zero.

STATUS REGISTER

The Am9511A contains an eight bit status register with the following bit assignments:

BUSY	SIGN	ZERO	ERROR CODE				CARRY
7	6	5	4	3	2	1	0

BUSY: Indicates that Am9511A is currently executing a command (1 = Busy).

SIGN: Indicates that the value on the top of stack is negative (1 = Negative).

ZERO: Indicates that the value on the top of stack is zero (1 = Value is zero).

ERROR CODE: This field contains an indication of the validity of the result of the last operation. The error codes are:

- 0000 – No error
- 1000 – Divide by zero
- 0100 – Square root or log of negative number
- 1100 – Argument of inverse sine, cosine, or e^{x} too large
- XX10 – Underflow
- XX01 – Overflow

CARRY: Previous operation resulted in carry or borrow from most significant bit. (1 = Carry/Borrow, 0 = No Carry/No Borrow)

If the BUSY bit in the status register is a one, the other status bits are not defined; if zero, indicating not busy, the operation is complete and the other status bits are defined as given above.

Table 1.

Command Mnemonic	Hex Code (sr = 1)	Hex Code (sr = 0)	Execution Cycles	Summary Description
16-BIT FIXED-POINT OPERATIONS				
SADD	EC	6C	16-18	Add TOS to NOS. Result to NOS. Pop Stack.
SSUB	ED	6D	30-32	Subtract TOS from NOS. Result to NOS. Pop Stack.
SMUL	EE	6E	84-94	Multiply NOS by TOS. Lower result to NOS. Pop Stack.
SMUU	F6	76	80-98	Multiply NOS by TOS. Upper result to NOS. Pop Stack.
SDIV	EF	6F	84-94	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FIXED-POINT OPERATIONS				
DADD	AC	2C	20-22	Add TOS to NOS. Result to NOS. Pop Stack.
DSUB	AD	2D	38-40	Subtract TOS from NOS. Result to NOS. Pop Stack.
DMUL	AE	2E	194-210	Multiply NOS by TOS. Lower result to NOS. Pop Stack.
DMUU	B6	36	182-218	Multiply NOS by TOS. Upper result to NOS. Pop Stack.
DDIV	AF	2F	196-210	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FLOATING-POINT PRIMARY OPERATIONS				
FADD	90	10	54-368	Add TOS to NOS. Result to NOS. Pop Stack.
FSUB	91	11	70-370	Subtract TOS from NOS. Result to NOS. Pop Stack.
FMUL	92	12	146-168	Multiply NOS by TOS. Result to NOS. Pop Stack.
FDIV	93	13	154-184	Divide NOS by TOS. Result to NOS. Pop Stack.
32-BIT FLOATING-POINT DERIVED OPERATIONS				
SQRT	81	01	782-870	Square Root of TOS. Result to TOS.
SIN	82	02	3796-4808	Sine of TOS. Result to TOS.
COS	83	03	3840-4878	Cosine of TOS. Result to TOS.
TAN	84	04	4894-5886	Tangent of TOS. Result to TOS.
ASIN	85	05	6230-7938	Inverse Sine of TOS. Result to TOS.
ACOS	86	06	6304-8284	Inverse Cosine of TOS. Result to TOS.
ATAN	87	07	4992-6536	Inverse Tangent of TOS. Result to TOS.
LOG	88	08	4474-7132	Common Logarithm of TOS. Result to TOS.
LN	89	09	4298-6956	Natural Logarithm of TOS. Result to TOS.
EXP	8A	0A	3794-4878	e raised to power in TOS. Result to TOS.
PWR	8B	0B	8290-12032	NOS raised to power in TOS. Result to NOS. Pop Stack.
DATA AND STACK MANIPULATION OPERATIONS				
NOP	80	00	4	No Operation. Clear or set SVREQ.
FIXS	9F	1F	90-214	Convert TOS from floating point format to fixed point format.
FIXD	9E	1E	90-336	
FLTS	9D	1D	62-156	Convert TOS from fixed point format to floating point format.
FLTD	9C	1C	56-342	
CHSS	F4	74	22-24	Change sign of fixed point operand on TOS.
CHSD	B4	34	26-28	
CHSF	95	15	16-20	Change sign of floating point operand on TOS.
PTOS	F7	77	16	Push stack. Duplicate NOS in TOS.
PTOD	B7	37	20	
PTOF	97	17	20	
POPS	F8	78	10	Pop stack. Old NOS becomes new TOS. Old TOS rotates to bottom.
POPD	B8	38	12	
POPF	98	18	12	
XCHS	F9	79	18	Exchange TOS and NOS.
XCHD	B9	39	26	
XCHF	99	19	26	
PUPI	9A	1A	16	Push floating point constant π onto TOS. Previous TOS becomes NOS.

4

COMMAND DESCRIPTIONS

This section contains detailed descriptions of the APU commands. They are arranged in alphabetical order by command mnemonic. In the descriptions, TOS means Top Of Stack and NOS means Next On Stack.

All derived functions except Square Root use Chebyshev polynomial approximating algorithms. This approach is used to help minimize the internal microprogram, to minimize the maximum error values and to provide a relatively even distribution of errors over the data range. The basic arithmetic operations are used by the derived functions to compute the various Chebyshev terms. The basic operations may produce error codes in the status register as a result.

Execution times are listed in terms of clock cycles and may be converted into time values by multiplying by the clock period used. For example, an execution time of 44 clock cycles when running at a 3MHz rate translates to 14 microseconds (44 x 32μs = 14μs). Variations in execution cycles reflect the data dependency of the algorithms.

In some operations exponent overflow or underflow may be possible. When this occurs, the exponent returned in the result will be 128 greater or smaller than its true value.

Many of the functions use portions of the data stack as scratch storage during development of the results. Thus previous values in those stack locations will be lost. Scratch locations destroyed are listed in the command descriptions and shown with the crossed-out locations in the Stack Contents After diagram.

Table 1 is a summary of all the Am9511A commands. It shows the hex codes for each command, the mnemonic abbreviation, a brief description and the execution time in clock cycles. The commands are grouped by functional classes.

The command mnemonics in alphabetical order are shown below in Table 2.

Table 2.

Command Mnemonics in Alphabetical Order.

ACOS	ARCCOSINE
ASIN	ARCSINE
ATAN	ARCTANGENT
CHSD	CHANGE SIGN DOUBLE
CHSF	CHANGE SIGN FLOATING
CHSS	CHANGE SIGN SINGLE
COS	COSINE
DADD	DOUBLE ADD
DDIV	DOUBLE DIVIDE
DMUL	DOUBLE MULTIPLY LOWER
DMUU	DOUBLE MULTIPLY UPPER
DSUB	DOUBLE SUBTRACT
EXP	EXPONENTIATION (e^x)
FADD	FLOATING ADD
FDIV	FLOATING DIVIDE
FIXD	FIX DOUBLE
FIXS	FIX SINGLE
FLTD	FLOAT DOUBLE
FLTS	FLOAT SINGLE
FMUL	FLOATING MULTIPLY
FSUB	FLOATING SUBTRACT
LOG	COMMON LOGARITHM
LN	NATURAL LOGARITHM
NOP	NO OPERATION
POPD	POP STACK DOUBLE
POPF	POP STACK FLOATING
POPS	POP STACK SINGLE
PTOD	PUSH STACK DOUBLE
PTOF	PUSH STACK FLOATING
PTOS	PUSH STACK SINGLE
PUPI	PUSH π
PWR	POWER (X^Y)
SADD	SINGLE ADD
SDIV	SINGLE DIVIDE
SIN	SINE
SMUL	SINGLE MULTIPLY LOWER
SMUU	SINGLE MULTIPLY UPPER
SQRT	SQUARE ROOT
SSUB	SINGLE SUBTRACT
TAN	TANGENT
XCHD	EXCHANGE OPERANDS DOUBLE
XCHF	EXCHANGE OPERANDS FLOATING
XCHS	EXCHANGE OPERANDS SINGLE

ACOS

32-BIT FLOATING-POINT INVERSE COSINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	1	0

Hex Coding: 86 with sr = 1
06 with sr = 0

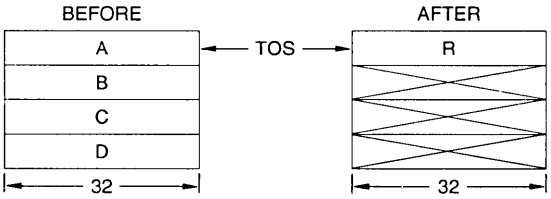
Execution Time: 6304 to 8284 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point inverse cosine of A. The result R is a value in radians between 0 and π. Initial operands A, B, C and D are lost.
ACOS will accept all input data values within the range of -1.0 to $+1.0$. Values outside this range will return an error code of 1100 in the status register.

Accuracy: ACOS exhibits a maximum relative error of 2.0 x 10^{-7} over the valid input data range.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



ATAN

32-BIT FLOATING-POINT INVERSE TANGENT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	1	1

Hex Coding: 87 with sr = 1
07 with sr = 0

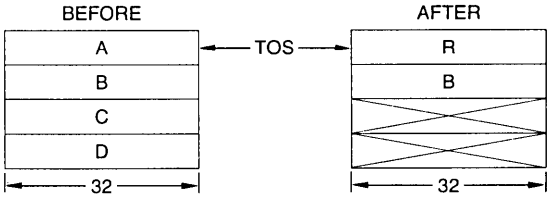
Execution Time: 4992 to 6536 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point inverse tangent of A. The result R is a value in radians between $-\pi/2$ and $+\pi/2$. Initial operands A, C and D are lost. Operand B is unchanged.
ATAN will accept all input data values that can be represented in the floating point format.

Accuracy: ATAN exhibits a maximum relative error of 3.0 x 10^{-7} over the input data range.

Status Affected: Sign, Zero

STACK CONTENTS



4

ASIN

32-BIT FLOATING-POINT INVERSE SINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	0	1

Hex Coding: 85 with sr = 1
05 with sr = 0

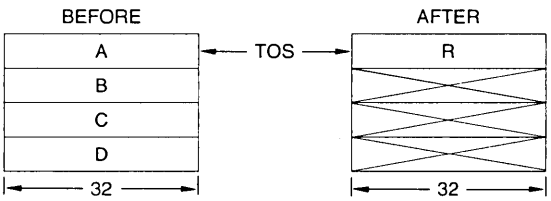
Execution Time: 6230 to 7938 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point inverse sine of A. The result R is a value in radians between $-\pi/2$ and $+\pi/2$. Initial operands A, B, C and D are lost.
ASIN will accept all input data values within the range of -1.0 to $+1.0$. Values outside this range will return an error code of 1100 in the status register.

Accuracy: ASIN exhibits a maximum relative error of 4.0 x 10^{-7} over the valid input data range.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



CHSD

32-BIT FIXED-POINT SIGN CHANGE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	0	1	0	0

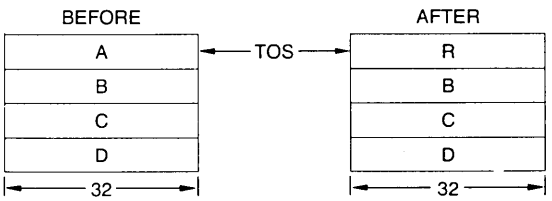
Hex Coding: B4 with sr = 1
34 with sr = 0

Execution Time: 26 to 28 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is subtracted from zero. The result R replaces A at the TOS. Other entries in the stack are not disturbed.
Overflow status will be set and the TOS will be returned unchanged when A is input as the most negative value possible in the format since no positive equivalent exists.

Status Affected: Sign, Zero, Error Field (overflow)

STACK CONTENTS



CHSF

32-BIT FLOATING-POINT SIGN CHANGE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	1	0	1

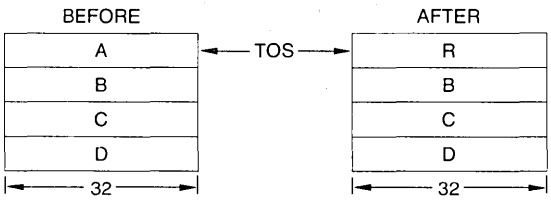
Hex Coding: 95 with sr = 1
15 with sr = 0

Execution Time: 16 to 20 clock cycles

Description:
The sign of the mantissa of the 32-bit floating-point operand A at the TOS is inverted. The result R replaces A at the TOS. Other stack entries are unchanged.
If A is input as zero (mantissa MSB = 0), no change is made.

Status Affected: Sign, Zero

STACK CONTENTS



CHSS

16-BIT FIXED-POINT SIGN CHANGE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	0	1	0	0

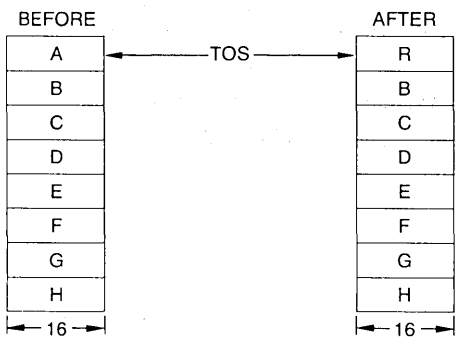
Hex Coding: F4 with sr = 1
74 with sr = 0

Execution Time: 22 to 24 clock cycles

Description:
16-bit fixed-point two's complement integer operand A at the TOS is subtracted from zero. The result R replaces A at the TOS. All other operands are unchanged.
Overflow status will be set and the TOS will be returned unchanged when A is input as the most negative value possible in the format since no positive equivalent exists.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



COS

32-BIT FLOATING-POINT COSINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	1	1

Hex Coding: 83 with sr = 1
03 with sr = 0

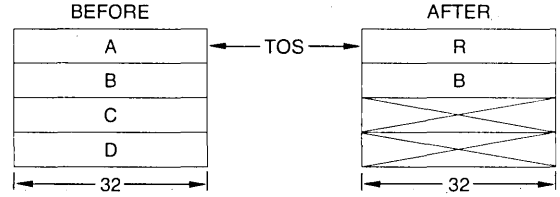
Execution Time: 3840 to 4878 clock cycles

Description:
The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point cosine of A. A is assumed to be in radians. Operands A, C and D are lost. B is unchanged.
The COS function can accept any input data value that can be represented in the data format. All input values are range reduced to fall within an interval of $-\pi/2$ to $+\pi/2$ radians.

Accuracy: COS exhibits a maximum relative error of 5.0×10^{-7} for all input data values in the range of -2π to $+2\pi$ radians.

Status Affected: Sign, Zero

STACK CONTENTS



DADD

32-BIT FIXED-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	0	0

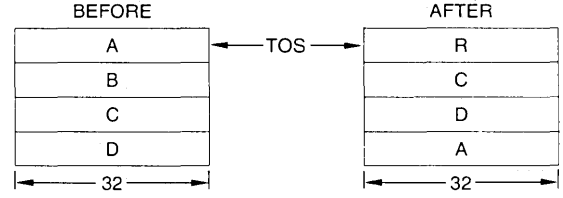
Hex Coding: AC with sr = 1
2C with sr = 0

Execution Time: 20 to 22 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is added to the 32-bit fixed-point two's complement integer operand B at the NOS. The result R replaces operand B and the Stack is moved up so that R occupies the TOS. Operand B is lost. Operands A, C and D are unchanged. If the addition generates a carry it is reported in the status register.
If the result is too large to be represented by the data format, the least significant 32 bits of the result are returned and overflow status is reported.

Status Affected: Sign, Zero, Carry, Error Field

STACK CONTENTS



DDIV

32-BIT FIXED-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	1	1

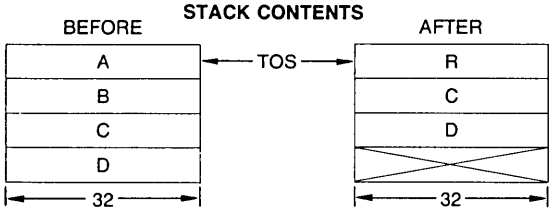
Hex Coding: AF with sr = 1
2F with sr = 0

Execution Time: 196 to 210 clock cycles when A ≠ 0
18 clock cycles when A = 0.

Description:
The 32-bit fixed-point two's complement integer operand B at NOS is divided by the 32-bit fixed-point two's complement integer operand A at the TOS. The 32-bit integer quotient R replaces B and the stack is moved up so that R occupies the TOS. No remainder is generated. Operands A and B are lost. Operands C and D are unchanged.
If A is zero, R is set equal to B and the divide-by-zero error status will be reported. If either A or B is the most negative value possible in the format, R will be meaningless and the overflow error status will be reported.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



DMUL

32-BIT FIXED-POINT MULTIPLY, LOWER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	1	0

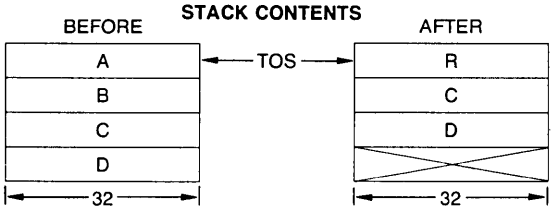
Hex Coding: AE with sr = 1
2E with sr = 0

Execution Time: 194 to 210 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 32-bit fixed-point two's complement integer operand B at the NOS. The 32-bit least significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The most significant half of the product is lost. Operands A and B are lost. Operands C and D are unchanged.
The overflow status bit is set if the discarded upper half was non-zero. If either A or B is the most negative value that can be represented in the format, that value is returned as R and the overflow status is set.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



DMUU

32-BIT FIXED-POINT MULTIPLY, UPPER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	0	1	1	0

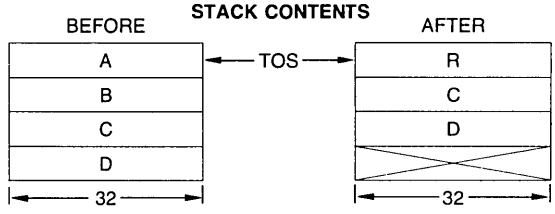
Hex Coding: B6 with sr = 1
36 with sr = 0

Execution Time: 182 to 218 clock cycles

Description:
The 32-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 32-bit fixed-point two's complement integer operand B at the NOS. The 32-bit most significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The least significant half of the product is lost. Operands A and B are lost. Operands C and D are unchanged.
If A or B was the most negative value possible in the format, overflow status is set and R is meaningless.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



DSUB

32-BIT FIXED-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	0	1	1	0	1

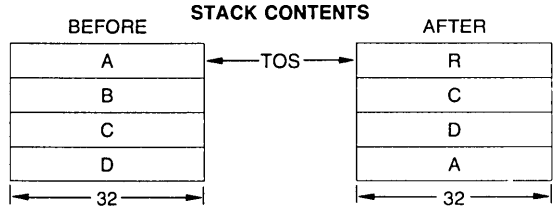
Hex Coding: AD with sr = 1
2D with sr = 0

Execution Time: 38 to 40 clock cycles

Description:
The 32-bit fixed-point two's complement operand A at the TOS is subtracted from the 32-bit fixed-point two's complement operand B at the NOS. The difference R replaces operand B and the stack is moved up so that R occupies the TOS. Operand B is lost. Operands A, C and D are unchanged.
If the subtraction generates a borrow it is reported in the carry status bit. If A is the most negative value that can be represented in the format the overflow status is set. If the result cannot be represented in the data format range, the overflow bit is set and the 32 least significant bits of the result are returned as R.

Status Affected: Sign, Zero, Carry, Overflow

STACK CONTENTS



4

EXP

32-BIT FLOATING-POINT e^X

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	1	0

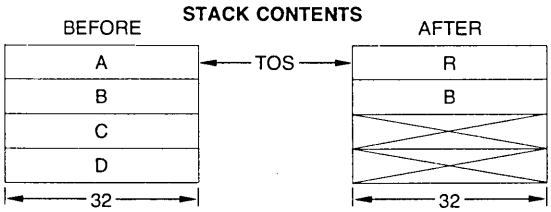
Hex Coding: 8A with sr = 1
0A with sr = 0

Execution Time: 3794 to 4878 clock cycles for $|A| \leq 1.0 \times 2^5$
34 clock cycles for $|A| > 1.0 \times 2^5$

Description:
The base of natural logarithms, e, is raised to an exponent value specified by the 32-bit floating-point operand A at the TOS. The result R of e^A replaces A. Operands A, C and D are lost. Operand B is unchanged.
EXP accepts all input data values within the range of $-1.0 \times 2^{+5}$ to $+1.0 \times 2^{+5}$. Input values outside this range will return a code of 1100 in the error field of the status register.

Accuracy: EXP exhibits a maximum relative error of 5.0×10^{-7} over the valid input data range.

Status Affected: Sign, Zero, Error Field



FADD

32-BIT FLOATING-POINT ADD

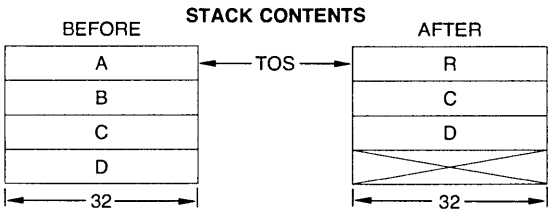
	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	0	0

Hex Coding: 90 with sr = 1
10 with sr = 0

Execution Time: 54 to 368 clock cycles for A ≠ 0
24 clock cycles for A = 0

Description:
32-bit floating-point operand A at the TOS is added to 32-bit floating-point operand B at the NOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.
Exponent alignment before the addition and normalization of the result accounts for the variation in execution time. Exponent overflow and underflow are reported in the status register, in which case the mantissa is correct and the exponent is offset by 128.

Status Affected: Sign, Zero, Error Field



FDIV

32-BIT FLOATING-POINT DIVIDE

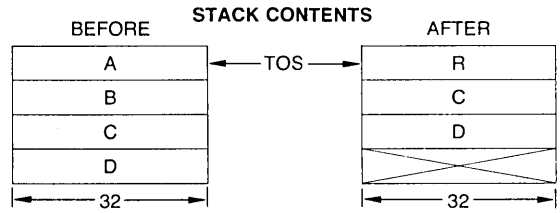
	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	1	1

Hex Coding: 93 with sr = 1
13 with sr = 0

Execution Time: 154 to 184 clock cycles for A ≠ 0
22 clock cycles for A = 0

Description:
32-bit floating-point operand B at NOS is divided by 32-bit floating-point operand A at the TOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.
If operand A is zero, R is set equal to B and the divide-by-zero error is reported in the status register. Exponent overflow or underflow is reported in the status register, in which case the mantissa portion of the result is correct and the exponent portion is offset by 128.

Status Affected: Sign, Zero, Error Field



FIXD

32-BIT FLOATING-POINT TO 32-BIT FIXED-POINT CONVERSION

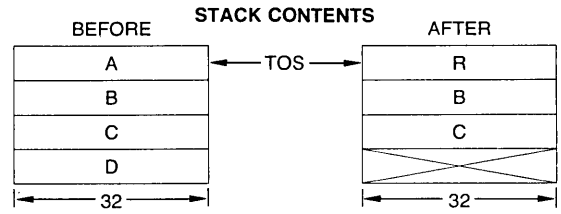
	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	1	0

Hex Coding: 9E with sr = 1
1E with sr = 0

Execution Time: 90 to 336 clock cycles

Description:
32-bit floating-point operand A at the TOS is converted to a 32-bit fixed-point two's complement integer. The result R replaces A. Operands A and D are lost. Operands B and C are unchanged.
If the integer portion of A is larger than 31 bits when converted, the overflow status will be set and A will not be changed. Operand D, however, will still be lost.

Status Affected: Sign, Zero Overflow



FIXS

32-BIT FLOATING-POINT TO 16-BIT FIXED-POINT CONVERSION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	1	1

Hex Coding: 9F with sr = 1
1F with sr = 0

Execution Time: 90 to 214 clock cycles

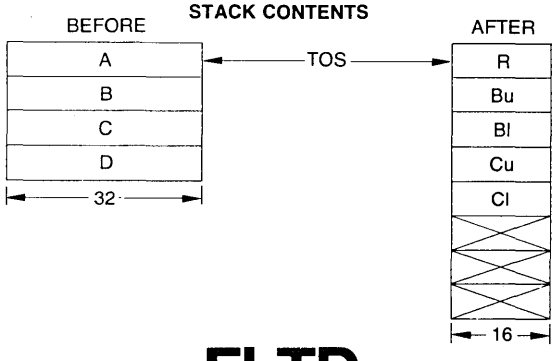
Description:

32-bit floating-point operand A at the TOS is converted to a 16-bit fixed-point two's complement integer. The result R replaces the lower half of A and the stack is moved up by two bytes so that R occupies the TOS. Operands A and D are lost. Operands B and C are unchanged, but appear as upper (u) and lower (l) halves on the 16-bit wide stack if they are 32-bit operands.

If the integer portion of A is larger than 15 bits when converted, the overflow status will be set and A will not be changed. Operand D, however, will still be lost.

Status Affected: Sign, Zero, Overflow

STACK CONTENTS



FLTD

32-BIT FIXED-POINT TO 32-BIT FLOATING-POINT CONVERSION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	0	0

Hex Coding: 9C with sr = 1
1C with sr = 0

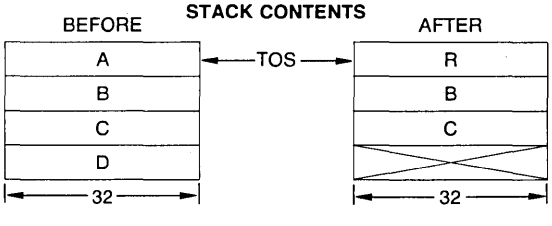
Execution Time: 56 to 342 clock cycles

Description:

32-bit fixed-point two's complement integer operand A at the TOS is converted to a 32-bit floating-point number. The result R replaces A at the TOS. Operands A and D are lost. Operands B and C are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



FLTS

16-BIT FIXED-POINT TO 32-BIT FLOATING-POINT CONVERSION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	1	0	1

Hex Coding: 9D with sr = 1
1D with sr = 0

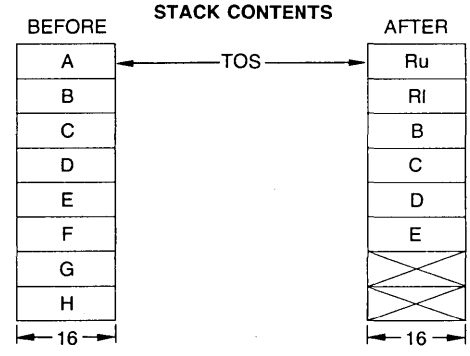
Execution Time: 62 to 156 clock cycles

Description:

16-bit fixed-point two's complement integer A at the TOS is converted to a 32-bit floating-point number. The lower half of the result R (Rl) replaces A, the upper half (Ru) replaces H and the stack is moved down so that Ru occupies the TOS. Operands A, F, G and H are lost. Operands B, C, D and E are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



FMUL

32-BIT FLOATING-POINT MULTIPLY

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	1	0

Hex Coding: 92 with sr = 1
12 with sr = 0

Execution Time: 146 to 168 clock cycles

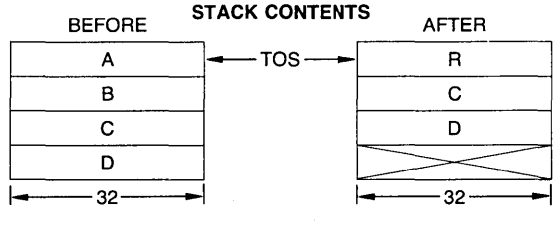
Description:

32-bit floating-point operand A at the TOS is multiplied by the 32-bit floating-point operand B at the NOS. The normalized result R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.

Exponent overflow or underflow is reported in the status register, in which case the mantissa portion of the result is correct and the exponent portion is offset by 128.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



4

FSUB

32-BIT FLOATING-POINT SUBTRACTION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	0	0	1

Hex Coding: 91 with sr = 1
11 with sr = 0

Execution Time: 70 to 370 clock cycles for A ≠ 0
26 clock cycles for A = 0

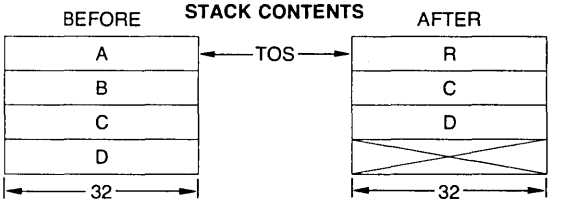
Description:

32-bit floating-point operand A at the TOS is subtracted from 32-bit floating-point operand B at the NOS. The normalized difference R replaces B and the stack is moved up so that R occupies the TOS. Operands A and B are lost. Operands C and D are unchanged.

Exponent alignment before the subtraction and normalization of the result account for the variation in execution time.

Exponent overflow or underflow is reported in the status register in which case the mantissa portion of the result is correct and the exponent portion is offset by 128.

Status Affected: Sign, Zero, Error Field (overflow)



LOG

32-BIT FLOATING-POINT COMMON LOGARITHM

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	0	0

Hex Coding: 88 with sr = 1
08 with sr = 0

Execution Time: 4474 to 7132 clock cycles for A > 0
20 clock cycles for A ≤ 0

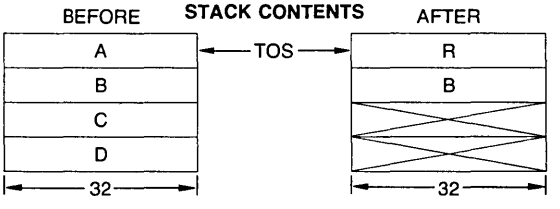
Description:

The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point common logarithm (base 10) of A. Operands A, C and D are lost. Operand B is unchanged.

The LOG function accepts any positive input data value that can be represented by the data format. If LOG of a non-positive value is attempted an error status of 0100 is returned.

Accuracy: LOG exhibits a maximum absolute error of 2.0×10^{-7} for the input range from 0.1 to 10, and a maximum relative error of 2.0×10^{-7} for positive values less than 0.1 or greater than 10.

Status Affected: Sign, Zero, Error Field



LN

32-BIT FLOATING-POINT NATURAL LOGARITHM

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	0	1

Hex Coding: 89 with sr = 1
09 with sr = 0

Execution Time: 4298 to 6956 clock cycles for A > 0
20 clock cycles for A≤ 0

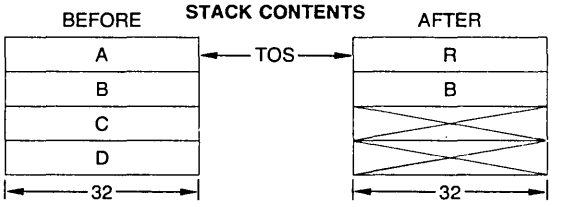
Description:

The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point natural logarithm (base e) of A. Operands A, C and D are lost. Operand B is unchanged.

The LN function accepts all positive input data values that can be represented by the data format. If LN of a non-positive number is attempted an error status of 0100 is returned.

Accuracy: LN exhibits a maximum absolute error of 2×10^{-7} for the input range from e^{-1} to e, and a maximum relative error of 2.0×10^{-7} for positive values less than e^{-1} or greater than e.

Status Affected: Sign, Zero, Error Field



NOP

NO OPERATION

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	0	0

Hex Coding: 80 with sr = 1
00 with sr = 0

Execution Time: 4 clock cycles

Description:

The NOP command performs no internal data manipulations. It may be used to set or clear the service request interface line without changing the contents of the stack.

Status Affected: The status byte is cleared to all zeroes.

POPD

32-BIT STACK POP

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	1	0	0	0

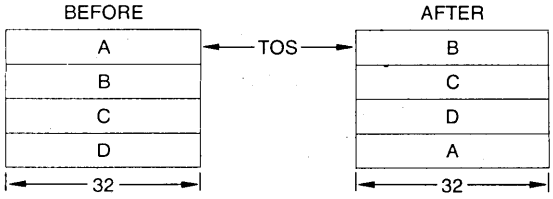
Hex Coding: B8 with sr = 1
38 with sr = 0

Execution Time: 12 clock cycles

Description:
The 32-bit stack is moved up so that the old NOS becomes the new TOS. The previous TOS rotates to the bottom of the stack. All operand values are unchanged. POPD and POPF execute the same operation.

Status Affected: Sign, Zero

STACK CONTENTS



POPS

16-BIT STACK POP

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	1	0	0	0

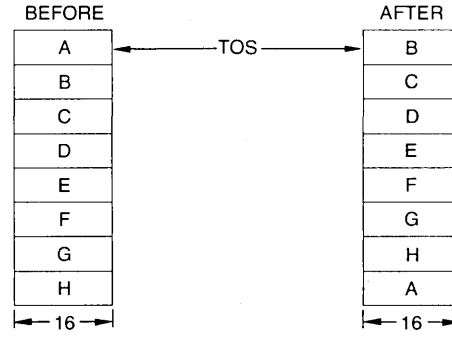
Hex Coding: F8 with sr = 1
78 with sr = 0

Execution Time: 10 clock cycles

Description:
The 16-bit stack is moved up so that the old NOS becomes the new TOS. The previous TOS rotates to the bottom of the stack. All operand values are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



POPF

32-BIT STACK POP

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	0	0	0

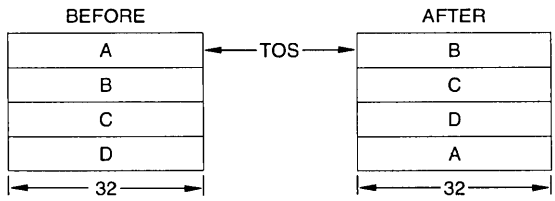
Hex Coding: 98 with sr = 1
18 with sr = 0

Execution Time: 12 clock cycles

Description:
The 32-bit stack is moved up so that the old NOS becomes the new TOS. The old TOS rotates to the bottom of the stack. All operand values are unchanged. POPF and POPD execute the same operation.

Status Affected: Sign, Zero

STACK CONTENTS



PTOD

PUSH 32-BIT TOS ONTO STACK

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	0	1	1	1

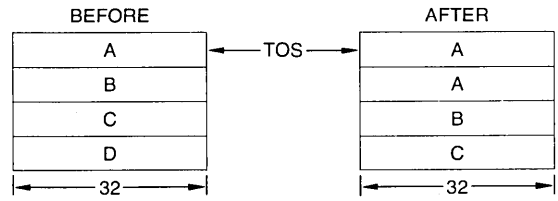
Hex Coding: B7 with sr = 1
37 with sr = 0

Execution Time: 20 clock cycles

Description:
The 32-bit stack is moved down and the previous TOS is copied into the new TOS location. Operand D is lost. All other operand values are unchanged. PTOD and PTOF execute the same operation.

Status Affected: Sign, Zero

STACK CONTENTS



4

PTOF

PUSH 32-BIT TOS ONTO STACK

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	0	1	1	1

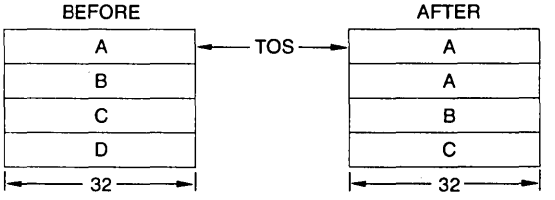
Hex Coding: 97 with sr = 1
17 with sr = 0

Execution Time: 20 clock cycles

Description:
The 32-bit stack is moved down and the previous TOS is copied into the new TOS location. Operand D is lost. All other operand values are unchanged. PTOF and PTOD execute the same operation.

Status Affected: Sign, Zero

STACK CONTENTS



PTOS

PUSH 16-BIT TOS ONTO STACK

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	0	1	1	1

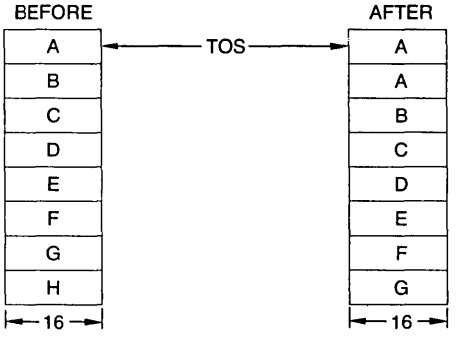
Hex Coding: F7 with sr = 1
77 with sr = 0

Execution Time: 16 clock cycles

Description:
The 16-bit stack is moved down and the previous TOS is copied into the new TOS location. Operand H is lost and all other operand values are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



PUPI

PUSH 32-BIT FLOATING-POINT π

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	0	1	0

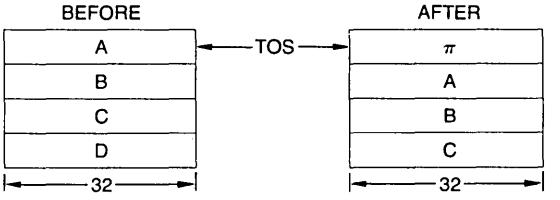
Hex Coding: 9A with sr = 1
1A with sr = 0

Execution Time: 16 clock cycles

Description:
The 32-bit stack is moved down so that the previous TOS occupies the new NOS location. 32-bit floating-point constant π is entered into the new TOS location. Operand D is lost. Operands A, B and C are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



PWR

32-BIT FLOATING-POINT X^Y

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	1	0	1	1

Hex Coding: 8B with sr = 1
0B with sr = 0

Execution Time: 8290 to 12032 clock cycles

Description:

32-bit floating-point operand B at the NOS is raised to the power specified by the 32-bit floating-point operand A at the TOS. The result R of B^A replaces B and the stack is moved up so that R occupies the TOS. Operands A, B, and D are lost. Operand C is unchanged.

The PWR function accepts all input data values that can be represented in the data format for operand A and all positive values for operand B. If operand B is non-positive an error status of 0100 will be returned. The EXP and LN functions are used to implement PWR using the relationship $B^A = EXP\,[A(LN\ B)]$. Thus if the term $[A(LN\ B)]$ is outside the range of $-1.0 \times 2^{+5}$ to $+1.0 \times 2^{+5}$ an error status of 1100 will be returned. Underflow and overflow conditions can occur.

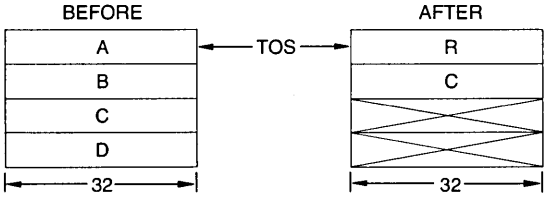
Accuracy: The error performance for PWR is a function of the LN and EXP performance as expressed by:

$|(\text{Relative Error})_{PWR}| = |(\text{Relative Error})_{EXP} + |A(\text{Absolute Error})_{LN}|$

The maximum relative error for PWR occurs when A is at its maximum value while $[A(LN\ B)]$ is near 1.0×2^5 and the EXP error is also at its maximum. For most practical applications the relative error for PWR will be less than 7.0×10^{-7}

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



SADD

16-BIT FIXED-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	0	0

Hex Coding: EC with sr = 1
6C with sr = 0

Execution Time: 16 to 18 clock cycles

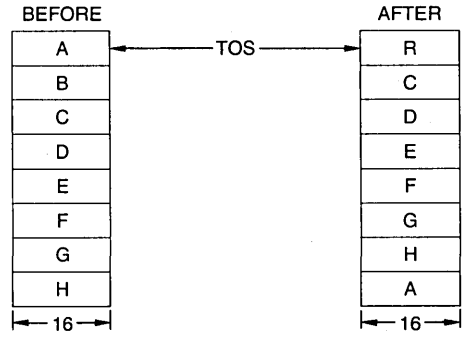
Description:

16-bit fixed-point two's complement integer operand A at the TOS is added to 16-bit fixed-point two's complement integer operand B at the NOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operand B is lost. All other operands are unchanged.

If the addition generates a carry bit it is reported in the status register. If an overflow occurs it is reported in the status register and the 16 least significant bits of the result are returned.

Status Affected: Sign, Zero, Carry, Error Field

STACK CONTENTS



4

SDIV

16-BIT FIXED-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	1	1

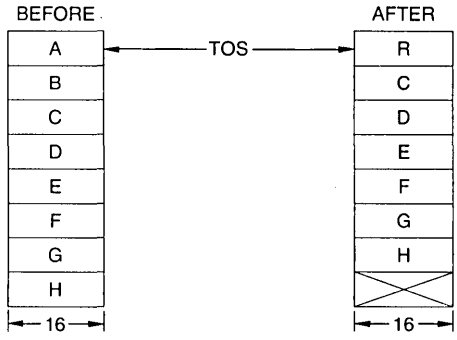
Hex Coding: EF with sr = 1
6F with sr = 0

Execution Time: 84 to 94 clock cycles for A ≠ 0
14 clock cycles for A = 0

Description:
16-bit fixed-point two's complement integer operand B at the NOS is divided by 16-bit fixed-point two's complement integer operand A at the TOS. The 16-bit integer quotient R replaces B and the stack is moved up so that R occupies the TOS. No remainder is generated. Operands A and B are lost. All other operands are unchanged.
If A is zero, R will be set equal to B and the divide-by-zero error status will be reported.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



SIN

32-BIT FLOATING-POINT SINE

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	1	0

Hex Coding: 82 with sr = 1
02 with sr = 0

Execution Time: 3796 to 4808 clock cycles for $|A| > 2^{-12}$ radians
30 clock cycles for $|A| \leq 2^{-12}$ radians

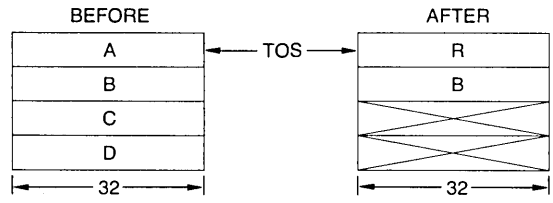
Description:
The 32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point sine of A. A is assumed to be in radians. Operands A, C and D are lost. Operand B is unchanged.

The SIN function will accept any input data value that can be represented by the data format. All input values are range reduced to fall within the interval $-\pi/2$ to $+\pi/2$ radians.

Accuracy: SIN exhibits a maximum relative error of 5.0 x 10^{-7} for input values in the range of -2π to $+2\pi$ radians.

Status Affected: Sign, Zero

STACK CONTENTS



SMUL

16-BIT FIXED-POINT MULTIPLY, LOWER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	1	0

Hex Coding: EE with sr = 1
6E with sr = 0

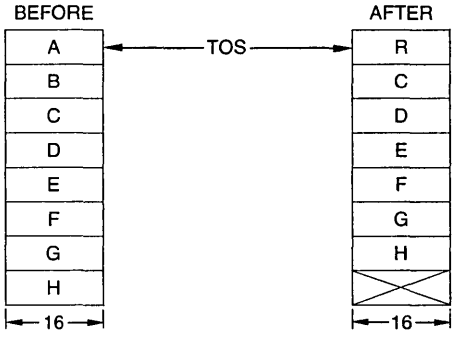
Execution Time: 84 to 94 clock cycles

Description:

16-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 16-bit fixed-point two's complement integer operand B at the NOS. The 16-bit least significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The most significant half of the product is lost. Operands A and B are lost. All other operands are unchanged. The overflow status bit is set if the discarded upper half was non-zero. If either A or B is the most negative value that can be represented in the format, that value is returned as R and the overflow status is set.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



SMUU

16-BIT FIXED-POINT MULTIPLY, UPPER

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	0	1	1	0

Hex Coding: F6 with sr = 1
76 with sr = 0

Execution Time: 80 to 98 clock cycles

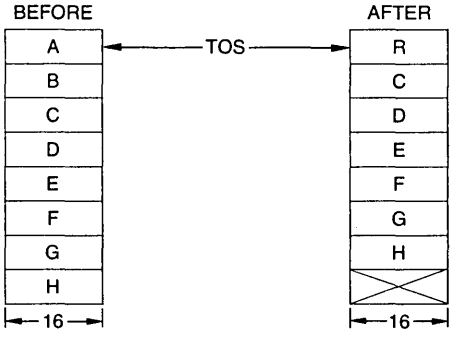
Description:

16-bit fixed-point two's complement integer operand A at the TOS is multiplied by the 16-bit fixed-point two's complement integer operand B at the NOS. The 16-bit most significant half of the product R replaces B and the stack is moved up so that R occupies the TOS. The least significant half of the product is lost. Operands A and B are lost. All other operands are unchanged.

If either A or B is the most negative value that can be represented in the format, that value is returned as R and the overflow status is set.

Status Affected: Sign, Zero, Error Field

STACK CONTENTS



4

SQRT

32-BIT FLOATING-POINT SQUARE ROOT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	0	0	1

Hex Coding: 81 with sr = 1
01 with sr = 0

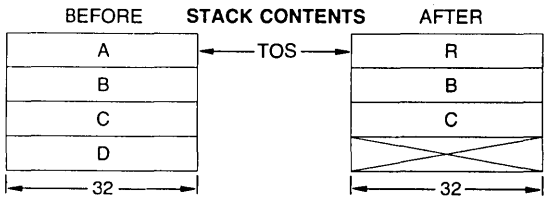
Execution Time: 782 to 870 clock cycles

Description:

32-bit floating-point operand A at the TOS is replaced by R, the 32-bit floating-point square root of A. Operands A and D are lost. Operands B and C are not changed.

SQRT will accept any non-negative input data value that can be represented by the data format. If A is negative an error code of 0100 will be returned in the status register.

Status Affected: Sign, Zero, Error Field



SSUB

16-BIT FIXED-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	0	1	1	0	1

Hex Coding: ED with sr = 1
6D with sr = 0

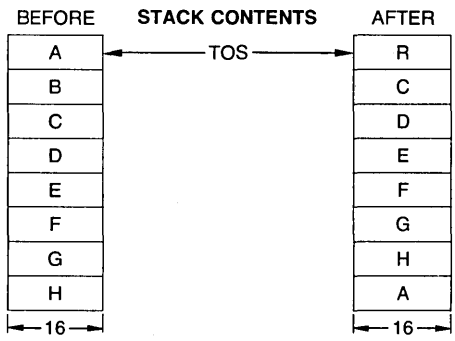
Execution Time: 30 to 32 clock cycles

Description:

16-bit fixed-point two's complement integer operand A at the TOS is subtracted from 16-bit fixed-point two's complement integer operand B at the NOS. The result R replaces B and the stack is moved up so that R occupies the TOS. Operand B is lost. All other operands are unchanged.

If the subtraction generates a borrow it is reported in the carry status bit. If A is the most negative value that can be represented in the format the overflow status is set. If the result cannot be represented in the format range, the overflow status is set and the 16 least significant bits of the result are returned as R.

Status Affected: Sign, Zero, Carry, Error Field



TAN

32-BIT FLOATING-POINT TANGENT

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	0	0	1	0	0

Hex Coding: 84 with sr = 1
04 with sr = 0

Execution Time: 4894 to 5886 clock cycles for $|A| \geq 2^{-12}$ radians
30 clock cycles for $|A| \leq 2^{-12}$ radians

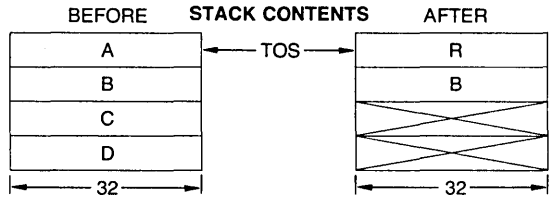
Description:

The 32-bit floating-point operand A at the TOS is replaced by the 32-bit floating-point tangent of A. Operand A is assumed to be in radians. A, C and D are lost. B is unchanged.

The TAN function will accept any input data value that can be represented in the data format. All input data values are range-reduced to fall within $-\pi/4$ to $+\pi/4$ radians. TAN is unbounded for input values near odd multiples of $\pi/2$ and in such cases the overflow bit is set in the status register. For angles smaller than 2^{-12} radians, TAN returns A as the tangent of A.

Accuracy: TAN exhibits a maximum relative error of 5.0×10^{-7} for input data values in the range of -2π to $+2\pi$ radians except for data values near odd multiples of $\pi/2$.

Status Affected: Sign, Zero, Error Field (overflow)



XCHD

EXCHANGE 32-BIT STACK OPERANDS

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	1	1	1	0	0	1

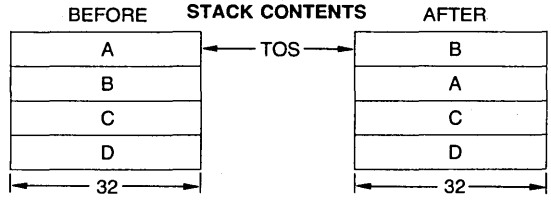
Hex Coding: B9 with sr = 1
39 with sr = 0

Execution Time: 26 clock cycles

Description:

32-bit operand A at the TOS and 32-bit operand B at the NOS are exchanged. After execution, B is at the TOS and A is at the NOS. All operands are unchanged. XCHD and XCHF execute the same operation.

Status Affected: Sign, Zero



XCHF

EXCHANGE 32-BIT STACK OPERANDS

	7	6	5	4	3	2	1	0
Binary Coding:	sr	0	0	1	1	0	0	1

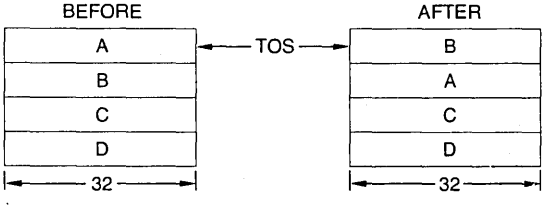
Hex Coding: 99 with sr = 1
19 with sr = 0

Execution Time: 26 clock cycles

Description:
32-bit operand A at the TOS and 32-bit operand B at the NOS are exchanged. After execution, B is at the TOS and A is at the NOS. All operands are unchanged. XCHD and XCHF execute the same operation.

Status Affected: Sign, Zero

STACK CONTENTS



XCHS

EXCHANGE 16-BIT STACK OPERANDS

	7	6	5	4	3	2	1	0
Binary Coding:	sr	1	1	1	1	0	0	1

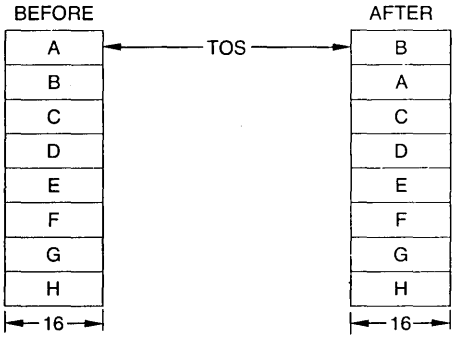
Hex Coding: F9 with sr = 1
79 with sr = 0

Execution Time: 18 clock cycles

Description:
16-bit operand A at the TOS and 16-bit operand B at the NOS are exchanged. After execution, B is at the TOS and A is at the NOS. All operand values are unchanged.

Status Affected: Sign, Zero

STACK CONTENTS



MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VDD with Respect to VSS	−0.5V to +15.0V
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	2.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VSS	VCC	VDD
Am9511ADC	$0°C \leq T_A \leq +70°C$	0V	+5.0V ±5%	+12V ±5%
Am9511ADM	$-55°C \leq T_A \leq +125°C$	0V	+5.0V ±10%	+12V ±10%

ELECTRICAL CHARACTERISTICS Over Operating Range (Note 1)

Parameters	Description	Test Conditions	Min.	Typ.	Max.	Units
VOH	Output HIGH Voltage	IOH = −200μA	3.7			Volts
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage		2.0		VCC	Volts
VIL	Input LOW Voltage		−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VI ≤ VCC			±10	μA
IOZ	Data Bus Leakage	VO = 0.4V			10	μA
		VO = VCC			10	
ICC	VCC Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
IDD	VDD Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
CO	Output Capacitance	fc = 1.0MHz, Inputs = 0V		8	10	pF
CI	Input Capacitance			5	8	pF
CIO	I/O Capacitance			10	12	pF

SWITCHING CHARACTERISTICS over operating range (Notes 2, 3)

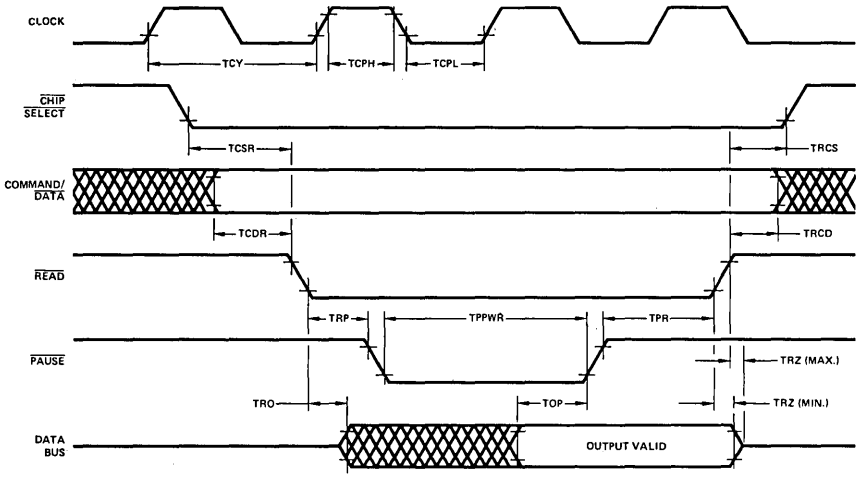
Parameters	Description		Am9511A Min.	Am9511A Max.	Am9511A-1 Min.	Am9511A-1 Max.	Units
TAPW	$\overline{EACK}$ LOW Pulse Width		100		75		ns
TCDR	C/$\overline{D}$ to $\overline{RD}$ LOW Set up Time		0		0		ns
TCDW	C/$\overline{D}$ to $\overline{WR}$ LOW Set up Time		0		0		ns
TCPH	Clock Pulse HIGH Width		200		140		ns
TCPL	Clock Pulse LOW Width		240		160		ns
TCSR	$\overline{CS}$ LOW to $\overline{RD}$ LOW Set up Time		0		0		ns
TCSW	$\overline{CS}$ LOW to $\overline{WR}$ LOW Set up Time		0		0		ns
TCY	Clock Period		480	5000	320	3300	ns
TDW	Data Bus Stable to $\overline{WR}$ HIGH Set up Time			150		100 (Note 9)	ns
TEAE	$\overline{EACK}$ LOW to $\overline{END}$ HIGH Delay			200		175	ns
TEPW	$\overline{END}$ LOW Pulse Width (Note 4)		400		300		ns
TOP	Data Bus Output Valid to $\overline{PAUSE}$ HIGH Delay		0		0		ns
TPPWR	$\overline{PAUSE}$ LOW Pulse Width Read (Note 5)	Data	3.5TCY+50	5.5TCY+300	3.5TCY+50	5.5TCY+200	ns
		Status	1.5TCY+50	3.5TCY+300	1.5TCY+50	3.5TCY+200	
TPPWW	$\overline{PAUSE}$ LOW Pulse Width Write (Note 8)			50		50	ns
TPR	$\overline{PAUSE}$ HIGH to $\overline{RD}$ HIGH Hold Time		0		0		ns
TPW	$\overline{PAUSE}$ HIGH to $\overline{WR}$ HIGH Hold Time		0		0		ns
TRCD	$\overline{RD}$ HIGH to C/$\overline{D}$ Hold Time		0		0		ns
TRCS	$\overline{RD}$ HIGH to $\overline{CS}$ HIGH Hold Time		0		0		ns
TRO	$\overline{RD}$ LOW to Data Bus ON Delay		50		50		ns
TRP	$\overline{RD}$ LOW to $\overline{PAUSE}$ LOW Delay (Note 6)			150		100 (Note 9)	ns
TRZ	$\overline{RD}$ HIGH to Data Bus OFF Delay		50	200	50	150	ns
TSAPW	$\overline{SVACK}$ LOW Pulse Width		100		75		ns
TSAR	$\overline{SVACK}$ LOW to SVREQ LOW Delay			300		200	ns
TWCD	$\overline{WR}$ HIGH to C/$\overline{D}$ Hold Time		60		30		ns
TWCS	$\overline{WR}$ HIGH to $\overline{CS}$ HIGH Hold Time		60		30		ns
TWD	$\overline{WR}$ HIGH to Data Bus Hold Time		20		20		ns
TWI	Write Inactive Time (Note 8)	Command	3TCY		3TCY		ns
		Data	4TCY		4TCY		
TWP	$\overline{WR}$ LOW to $\overline{PAUSE}$ LOW Delay (Note 6)			150		100 (Note 9)	ns

NOTES

1. Typical values are for T_A = 25°C, nominal supply voltages and nominal processing parameters.
2. Switching parameters are listed in alphabetical order.
3. Test conditions assume transition times of 20ns or less, output loading of one TTL gate plus 100pF and timing reference levels of 0.8V and 2.0V.
4. $\overline{END}$ low pulse width is specified for $\overline{EACK}$ tied to VSS. Otherwise TEAE applies.
5. Minimum values shown assume no previously entered command is being executed for the data access. If a previously entered command is being executed, $\overline{PAUSE}$ LOW Pulse Width is the time to complete execution plus the time shown. Status may be read at any time without exceeding the time shown.
6. $\overline{PAUSE}$ is pulled low for both command and data operations.
7. TEX is the execution time of the current command (see the Command Execution Times table).
8. $\overline{PAUSE}$ low pulse width is less than 50ns when writing into the data port or the control port as long as the duty cycle requirement (TWI) is observed and no previous command is being executed. TWI may be safely violated as long as the extended TPPWW that results is observed. If a previously entered command is being executed, $\overline{PAUSE}$ LOW Pulse Width is the time to complete execution plus the time shown.
9. 150ns for Am9511A-1DM.

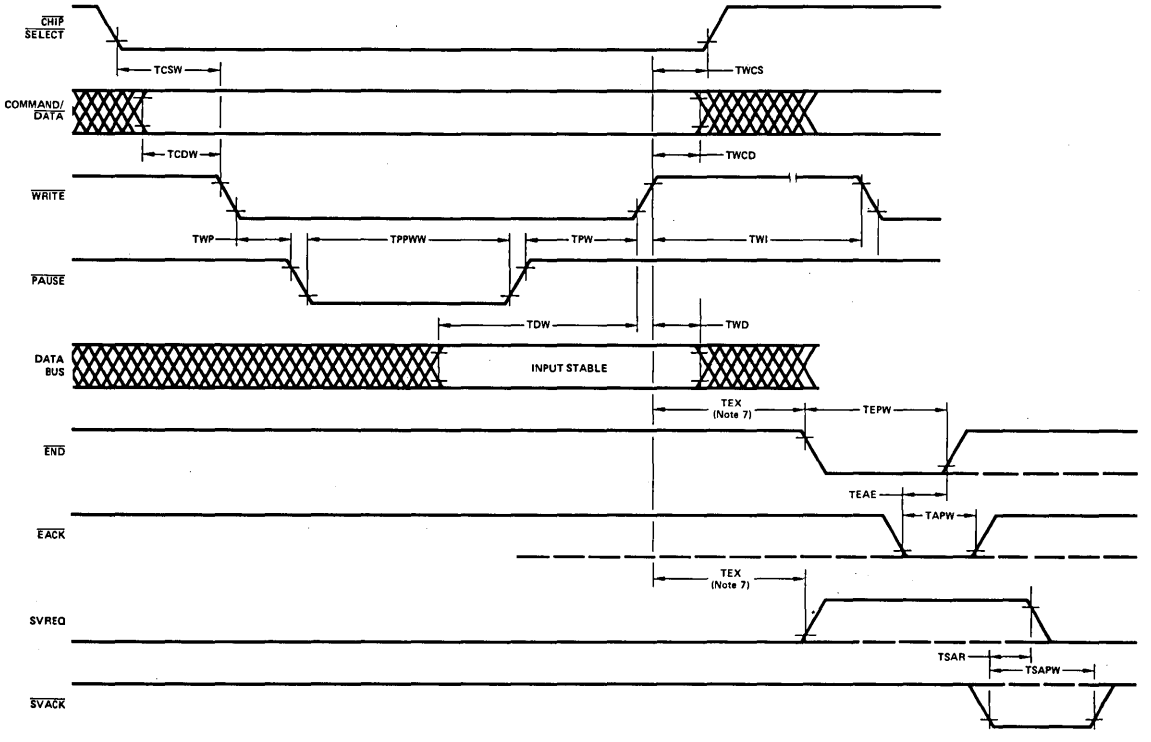
SWITCHING WAVEFORMS

READ OPERATIONS



MOS-048

WRITE OPERATIONS



MOS-049

APPLICATION INFORMATION

The diagram in Figure 2 shows the interface connections for the Am9511A APU with operand transfers handled by an Am9517 DMA controller, and CPU coordination handled by an Am9519 Interrupt Controller. The APU interrupts the CPU to indicate that a command has been completed. When the performance enhancements provided by the DMA and Interrupt operations are not required, the APU interface can be simplified as shown in Figure 1. The Am9511A APU is designed with a general purpose 8-bit data bus and interface control so that it can be conveniently used with any general 8-bit processor.

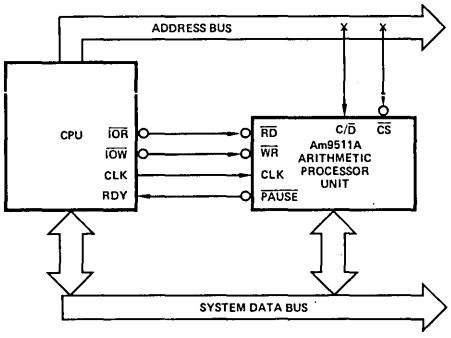


Figure 1. Am9511A Minimum Configuration Example.

MOS-050

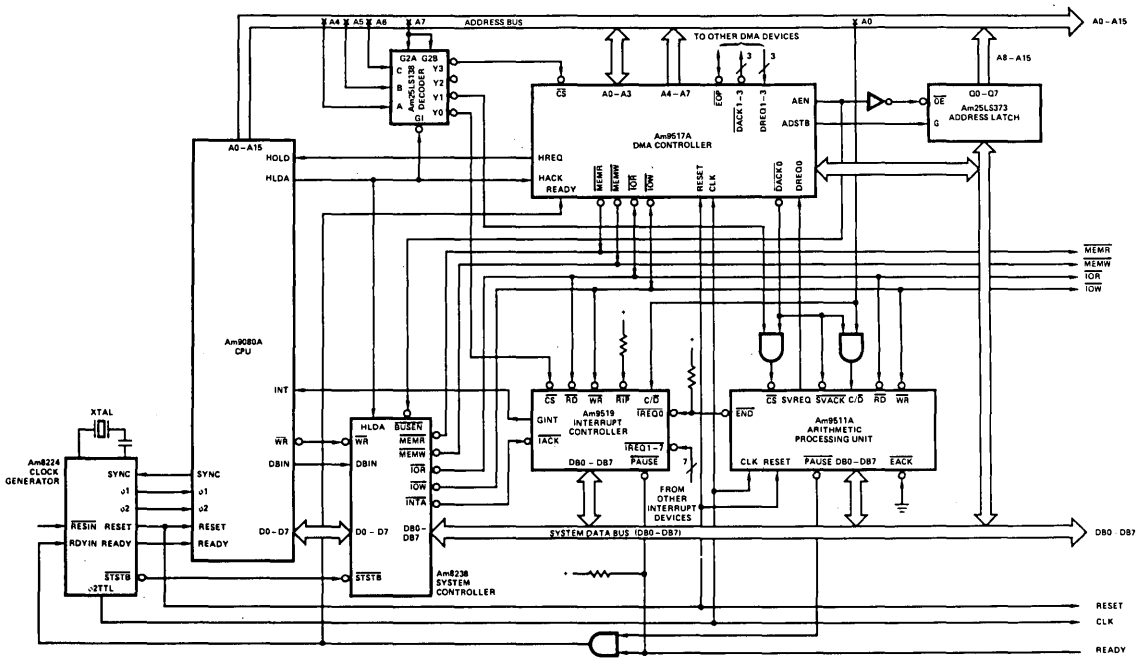


Figure 2. Am9511A High Performance Configuration Example.

MOS-051

Am9512

Floating Point Processor

DISTINCTIVE CHARACTERISTICS

- Single (32-bit) and double (64-bit) precision capability
- Add, subtract, multiply and divide functions
- Compatible with proposed IEEE format
- Easy interfacing to most microprocessors
- 8-bit data bus
- Standard 24-pin package
- 12V and 5V power supplies
- Stack oriented operand storage
- Direct memory access or programmed I/O Data Transfers
- End of execution signal
- Error interrupt

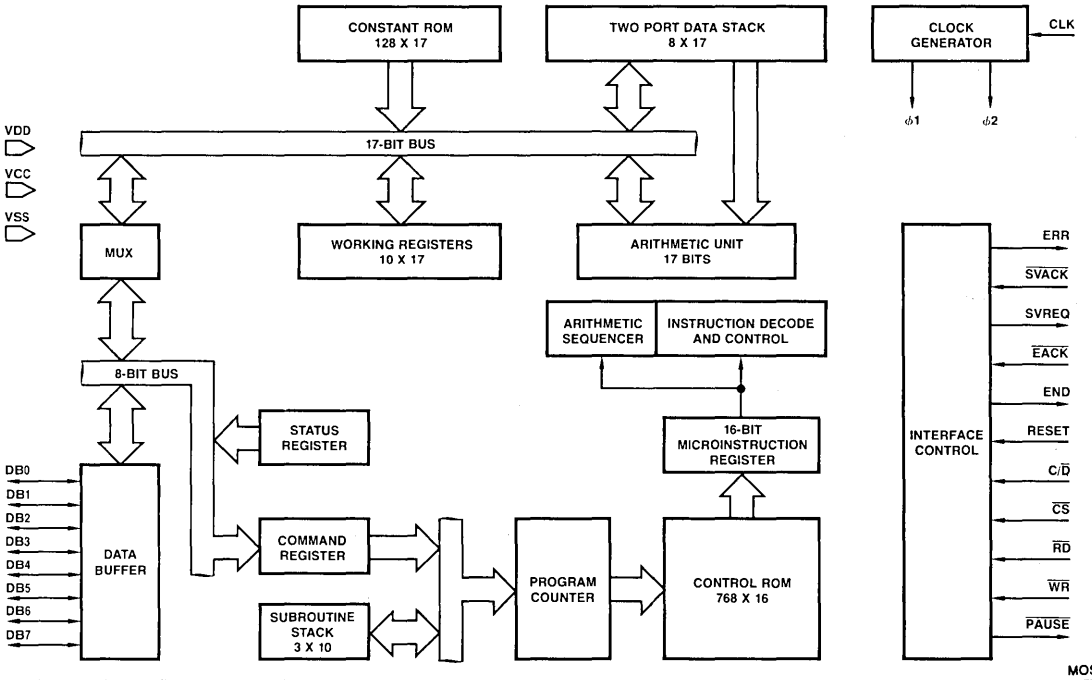
GENERAL DESCRIPTION

The Am9512 is a high performance floating-point processor unit (FPU). It provides single precision (32-bit) and double precision (64-bit) add, subtract, multiply and divide operations. It can be easily interfaced to enhance the computational capabilities of the host microprocessor.

The operand, result, status and command information transfers take place over an 8-bit bidirectional data bus. Operands are pushed onto an internal stack by the host processor and a command is issued to perform an operation on the data stack. The results of this operation are available to the host processor by popping the stack.

Information transfers between the Am9512 and the host processor can be handled by using programmed I/O or direct memory access techniques. After completing an operation, the Am9512 activates an "end of execution" signal that can be used to interrupt the host processor.

BLOCK DIAGRAM

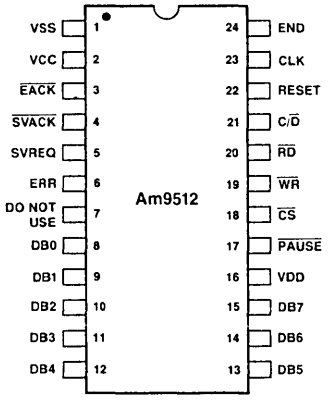


MOS-203

ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency	
		2MHz	3MHz
Hermetic DIP	$0°C \leq T_A \leq 70°C$	AM9512DC	AM9512-1DC
	$-55°C \leq T_A \leq +125°C$	AM9512DM	AM9512-1DM

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

MOS-204

INTERFACE SIGNAL DESCRIPTION

VCC: +5V Power Supply

VDD: +12V Power Supply

VSS: Ground

CLK (Clock, Input)

An external timing source connected to the CLK input provides the necessary clocking.

RESET (Reset, Input)

A HIGH on this input causes initialization. Reset terminates any operation in progress, and clears the status register to zero. The internal stack pointer is initialized and the contents of the stack may be affected. After a reset the END output, the ERR output and the SVREQ output will be LOW. For proper initialization, RESET must be HIGH for at least five CLK periods following stable power supply voltages and stable clock.

C/$\overline{D}$ (Command/Data Select, Input)

The C/$\overline{D}$ input together with the $\overline{RD}$ and $\overline{WR}$ inputs determines the type of transfer to be performed on the data bus as follows:

C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	Function
L	H	L	Push data byte into the stack
L	L	H	Pop data byte from the stack
H	H	L	Enter command
H	L	H	Read Status
X	L	L	Undefined

L = LOW
H = HIGH
X = DON'T CARE

END (End of Execution, Output)

A HIGH on this output indicates that execution of the current command is complete. This output will be cleared LOW by activating the $\overline{EACK}$ input LOW or performing any read or write operation or device initialization using the RESET. If $\overline{EACK}$ is tied LOW, the END output will be a pulse (see $\overline{EACK}$ description).

Reading the status register while a command execution is in progress is allowed. However any read or write operation clears the flip-flop that generates the END output. Thus such continuous reading could conflict with internal logic setting of the END flip-flop at the end of command execution.

$\overline{EACK}$ (End Acknowledge, Input)

This input when LOW makes the END output go LOW. As mentioned earlier HIGH on the END output signals completion of a command execution. The END signal is derived from an internal flip-flop which is clocked at the completion of a command. This flip-flop is clocked to the reset state when $\overline{EACK}$ is LOW. Consequently, if $\overline{EACK}$ is tied LOW, the END output will be a pulse that is approximately one CLK period wide.

SVREQ (Service Request, Output)

A HIGH on this output indicates completion of a command. In this sense this output is the same as the END output. However, the SVREQ output will go HIGH at the completion of a command. This bit must be 1 for SVREQ to go HIGH. The SVREQ can be cleared (i.e., go LOW) by activating the $\overline{SVACK}$ input LOW or initializing the device using the RESET. Also, the SVREQ will be automatically cleared after completion of any command that has the service request bit as 0.

$\overline{SVACK}$ (Service Acknowledge, Input)

A LOW on this input clears SVREQ. If the $\overline{SVACK}$ input is permanently tied LOW, it will conflict with the internal setting of the SVREQ output. Thus the SVREQ indication cannot be relied upon if the $\overline{SVACK}$ is tied LOW.

DB0-DB7 (Data Bus, Input/Output)

These eight bidirectional lines are used to transfer command, status and operand information between the device and the host processor. DB0 is the least significant and DB7 is the most significant bit position. HIGH on a data bus line corresponds to 1 and LOW corresponds to 0.

When pushing operands on the stack using the data bus, the least significant byte must be pushed first and most significant byte last. When popping the stack to read the result of an operation, the most significant byte will be available on the data bus first and the least significant byte will be the last. Moreover, for pushing operands and popping results, the number of transactions must be equal to the proper number of bytes appropriate for the chosen format. Otherwise, the internal byte pointer will not be aligned properly. The Am9512 single precision format requires 4 bytes and double precision format requires 8 bytes.

ERR (Error, Output)

This output goes HIGH to indicate that the current command execution resulted in an error condition. The error conditions are: attempt to divide by zero, exponent overflow and exponent underflow. The ERR output is cleared LOW on read status register operation or upon RESET.

The ERR output is derived from the error bits in the status register. These error bits will be updated internally at an appropriate time during a command execution. Thus ERR output going HIGH may not correspond with the completion of a command. Reading of the status register can be performed while a command execution is in progress. However it should be noted that reading the status register clears the ERR output. Thus reading the status register while a command execution in progress may result in an internal conflict with the ERR output.

$\overline{CS}$ (Chip Select, Input)

This input must be LOW to accomplish any read or write operation to the Am9512.

To perform a write operation, appropriate data is presented on DB0 through DB7 lines, appropriate logic level on the C/$\overline{D}$ input and the $\overline{CS}$ input is made LOW. Whenever $\overline{WR}$ and $\overline{RD}$ inputs are both HIGH and $\overline{CS}$ is LOW, $\overline{PAUSE}$ goes LOW. However actual writing into the Am9512 cannot start until $\overline{WR}$ is made LOW. After initiating the write operation by the HIGH to LOW transition on the $\overline{WR}$ input, the $\overline{PAUSE}$ output will go HIGH indicating the write operation has been acknowledged. The $\overline{WR}$ input can go HIGH after $\overline{PAUSE}$ goes HIGH. The data lines, C/$\overline{D}$ input and the $\overline{CS}$ input can change when appropriate hold time requirements are satisfied. See write timing diagram for details.

To perform a read operation an appropriate logic level is established on the C/$\overline{D}$ input and $\overline{CS}$ is made LOW. The $\overline{PAUSE}$ output goes LOW because $\overline{WR}$ and $\overline{RD}$ inputs are HIGH. The read operation does not start until the $\overline{RD}$ input goes LOW. $\overline{PAUSE}$ will go HIGH indicating that read operation is complete and the required information is available on the DB0 through DB7 lines. This information will remain on the data lines as long as $\overline{RD}$ is LOW. The $\overline{RD}$ input can return HIGH anytime after $\overline{PAUSE}$ goes HIGH. The $\overline{CS}$ input and C/$\overline{D}$ input can change anytime after $\overline{RD}$ returns HIGH. See read timing diagram for details. If the $\overline{CS}$ is tied LOW permanently, $\overline{PAUSE}$ will remain LOW until the next Am9512 read or write access.

$\overline{RD}$ (Read, Input)

A LOW on this input is used to read information from an internal location and gate that information onto the data bus. The $\overline{CS}$ input must be LOW to accomplish the read operation. The C/$\overline{D}$ input determines what internal location is of interest. See C/$\overline{D}$, $\overline{CS}$ input descriptions and read timing diagram for details. If the END output was HIGH, performing any read operation will make the END output go LOW after the HIGH to LOW transition of the $\overline{RD}$ input (assuming $\overline{CS}$ is LOW). If the ERR output was HIGH performing a status register read operation will make the ERR output LOW. This will happen after the HIGH to LOW transition of the RD input (assuming $\overline{CS}$ is LOW).

$\overline{WR}$ (Write, Input)

A LOW on this input is used to transfer information from the data bus into an internal location. The $\overline{CS}$ must be LOW to accomplish the write operation. The C/$\overline{D}$ determines which internal location is to be written. See C/$\overline{D}$, $\overline{CS}$ input descriptions and write timing diagram for details.

If the END output was HIGH, performing any write operation will make the END output go LOW after the LOW to HIGH transition of the $\overline{WR}$ input (assuming $\overline{CS}$ is LOW).

$\overline{PAUSE}$ (Pause, Output)

This output is a handshake signal used while performing read or write transactions with the Am9512. If the $\overline{WR}$ and $\overline{RD}$ inputs are both HIGH, the $\overline{PAUSE}$ output goes LOW with the $\overline{CS}$ input in anticipation of a transaction. If $\overline{WR}$ goes LOW to initiate a write transaction with proper signals established on the DB0-DB7, C/$\overline{D}$ inputs, the $\overline{PAUSE}$ will return HIGH indicating that the write operation has been accomplished. The $\overline{WR}$ can be made HIGH after this event. On the other hand, if a read operation is desired, the $\overline{RD}$ input is made LOW after activating $\overline{CS}$ LOW and establishing proper C/$\overline{D}$ input. (The $\overline{PAUSE}$ will go LOW in response to $\overline{CS}$ going LOW.) The $\overline{PAUSE}$ will return HIGH indicating completion of read. The $\overline{RD}$ can return HIGH after this event. It should be noted that a read or write operation can be initiated without any regard to whether a command execution is in progress or not. Proper device operation is assured by obeying the $\overline{PAUSE}$ output indication as described.

FUNCTIONAL DESCRIPTION

Major functional units of the Am9512 are shown in the block diagram. The Am9512 employs a microprogram controlled stack oriented architecture with 17-bit wide data paths.

The Arithmetic Unit receives one of its operands from the Operand Stack. This stack is an eight word by 17-bit two port memory with last in – first out (LIFO) attributes. The second operand to the Arithmetic Unit is supplied by the internal 17-bit bus. In addition to supplying the second operand, this bidirectional bus also carries the results from the output of the Arithmetic Unit when required. Writing into the Operand Stack takes place from this internal 17-bit bus when required. Also connected to this bus are the Constant ROM and Working Registers. The ROM provides the required constants to perform the mathematical operations while the Working Registers provide storage for the intermediate values during command execution.

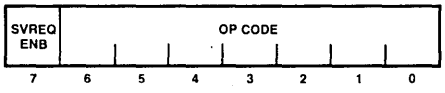
Communication between the external world and the Am9512 takes place on eight bidirectional input/output lines, DB0 through DB7 (Data Bus). These signals are gated to the internal 8-bit bus through appropriate interface and buffer circuitry. Multiplexing facilities exist for bidirectional communication between the internal eight and 17-bit buses. The Status Register and Command Register are also located on the 8-bit bus.

The Am9512 operations are controlled by the microprogram contained in the Control ROM. The Program Counter supplies the microprogram addresses and can be partially loaded from the Command Register. Associated with the Program Counter is the Subroutine Stack where return addresses are held during subroutine calls in the microprogram. The Microinstruction Register holds the current microinstruction being executed. The register facilitates pipelined microprogram execution. The Instruction Decode logic generates various internal control signals needed for the Am9512 operation.

The Interface Control logic receives several external inputs and provides handshake related outputs to facilitate interfacing the Am9512 to microprocessors.

COMMAND FORMAT

The Operation of the Am9512 is controlled from the host processor by issuing instructions called commands. The command format is shown below:



The command consists of 8 bits; the least significant 7 bits specify the operation to be performed as detailed in the accompanying table. The most significant bit is the Service Request Enable bit. This bit must be a 1 if SVREQ is to go high at end of executing a command.

The Am9512 commands fall into three categories: Single precision arithmetic, double precision arithmetic and data manipulation. There are four arithmetic operations that can be performed with single precision (32-bit), or double precision (64-bit) floating-point numbers: add, subtract, multiply and divide. These operations require two operands. The Am9512 assumes that these operands are located in the internal stack as Top of Stack (TOS) and Next on Stack (NOS). The result will always be returned to the previous NOS which becomes the new TOS. Results from an operation are of the same precision and format as the operands. The results will be rounded to preserve the accuracy. The actual data formats and rounding procedures are described in a later section. In addition to the arithmetic operations, the Am9512 implements eight data manipulating operations. These include changing the sign of a double or single precision operand located in TOS, exchanging single precision operands located at TOS and NOS, as well as copying and popping single or double precision operands. See also the sections on status register and operand formats.

The Execution times of the Am9512 commands are all data dependent. Table 2 shows one example of each command execution time:

Table 1. Command Decoding Table.

Command Bits								Mnemonic	Description
7	6	5	4	3	2	1	0		
X	0	0	0	0	0	0	1	SADD	Add TOS to NOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	0	1	0	SSUB	Subtract TOS from NOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	0	1	1	SMUL	Multiply NOS by TOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	1	0	0	SDIV	Divide NOS by TOS Single Precision and result to NOS. Pop stack.
X	0	0	0	0	1	0	1	CHSS	Change sign of TOS Single Precision operand.
X	0	0	0	0	1	1	0	PTOS	Push Single Precision operand on TOS to NOS.
X	0	0	0	0	1	1	1	POPS	Pop Single Precision operand from TOS. NOS becomes TOS.
X	0	0	0	1	0	0	0	XCHS	Exchange TOS with NOS Single Precision.
X	0	1	0	1	1	0	1	CHSD	Change sign of TOS Double Precision operand.
X	0	1	0	1	1	1	0	PTOD	Push Double Precision operand on TOS to NOS.
X	0	1	0	1	1	1	1	POPD	Pop Double Precision operand from TOS. NOS becomes TOS.
X	0	0	0	0	0	0	0	CLR	CLR status.
X	0	1	0	1	0	0	1	DADD	Add TOS to NOS Double Precision and result to NOS. Pop stack.
X	0	1	0	1	0	1	0	DSUB	Subtract TOS from NOS Double Precision and result to NOS. Pop stack.
X	0	1	0	1	0	1	1	DMUL	Multiply NOS by TOS Double Precision and result to NOS. Pop stack.
X	0	1	0	1	1	0	0	DDIV	Divide NOS by TOS Double Precision and result to NOS. Pop Stack.

Notes: X = Don't Care
Operation for bit combinations not listed above is undefined.

Table 2. Execution Times.

Command	TOS	NOS	Result	Clock periods
SADD	3F800000	3F800000	40000000	58
SSUB	3F800000	3F800000	00000000	56
SMUL	40400000	3FC00000	40900000	198
SDIV	3F800000	40000000	3F000000	228
CHSS	3F800000	–	BF800000	10
PTOS	3F800000	–	–	16
POPS	3F800000	–	–	14
XCHS	3F800000	4000000	–	26
CHSD	3FF0000000000000	–	BFF0000000000000	24
PTOD	3FF0000000000000	–	–	40
POPD	3FF0000000000000	–	–	26
CLR	3FF0000000000000	–	–	4
DADD	3FF00000A0000000	8000000000000000	3FF00000A0000000	578
DSUB	3FF00000A0000000	8000000000000000	3FF00000A0000000	578
DMUL	BFF8000000000000	3FF8000000000000	C002000000000000	1748
DDIV	BFF8000000000000	3FF8000000000000	BFF0000000000000	4560

Note: TOS, NOS and Result are in hexadecimal; Clock period is in decimal.

COMMAND INITIATION

After properly positioning the required operands in the stack, a command may be issued. The procedure for initiating a command execution is as follows:

1. Establish appropriate command on the DB0-DB7 lines.
2. Establish HIGH on the C/$\overline{D}$ input.
3. Establish LOW on the $\overline{CS}$ input. Whenever $\overline{WR}$ and $\overline{RD}$ inputs are HIGH the $\overline{PAUSE}$ output follows the $\overline{CS}$ input. Hence $\overline{PAUSE}$ will become LOW.
4. Establish LOW on the $\overline{WR}$ input after an appropriate set up time (see timing diagrams).
5. Sometime after the HIGH to LOW level transition of $\overline{WR}$ input, the $\overline{PAUSE}$ output will become HIGH to acknowledge the write operation. The $\overline{WR}$ input can return to HIGH anytime after $\overline{PAUSE}$ goes HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs are allowed to change after the hold time requirements are satisfied (see timing diagram).

An attempt to issue a new command while the current command execution is in progress is allowed. Under these circumstances, the $\overline{PAUSE}$ output will not go HIGH until the current command execution is completed.

OPERAND ENTRY

The Am9512 commands operate on the operands located at the TOS and NOS and results are returned to the stack at NOS and then popped to TOS. The operands required for the Am9512 are one of two formats – single precision floating-point (4 bytes) or double precision floating-point (8 bytes). The result of an operation has the same format as the operands. In other words, operations using single precision quantities always result in a single precision result while operations involving double precision quantities will result in double precision result.

Operands are always entered into the stack least significant byte first and most significant byte last. The following procedure must be followed to enter operands into the stack:

1. The lower significant operand byte is established on the DB0-DB7 lines.
2. A LOW is established on the C/$\overline{D}$ input to specify that data is to be entered into the stack.
3. The $\overline{CS}$ input is made LOW. Whenever the $\overline{WR}$ and $\overline{RD}$ inputs are HIGH, the $\overline{PAUSE}$ output will follow the $\overline{CS}$ input. Thus $\overline{PAUSE}$ output will become LOW.
4. After appropriate set up time (see timing diagrams), the $\overline{WR}$ input is made LOW.
5. Sometime after this event, $\overline{PAUSE}$ will return HIGH to indicate that the write operation has been acknowledged.
6. Anytime after the $\overline{PAUSE}$ output goes HIGH the $\overline{WR}$ input can be made HIGH. The DB0-DB7, C/$\overline{D}$ and $\overline{CS}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagrams).

The above procedure must be repeated until all bytes of the operand are pushed into the stack. It should be noted that for single precision operands 4 bytes should be pushed and 8 bytes must be pushed for double precision. Not pushing all the bytes of a quantity will result in byte pointer misalignment.

The Am9512 stack can accommodate 4 single precision quantities or 2 double precision quantities. Pushing more quantities than the capacity of the stack will result in loss of data which is usual with any LIFO stack.

REMOVING THE RESULTS

Result from an operation will be available at the TOS. Results can be transferred from the stack to the data bus by reading the stack.

When the stack is popped for results, the most significant byte is available first and the least significant byte last. A result is always of the same precision as the operands that produced it. Thus when the result is taken from the stack, the total number of bytes popped out should be appropriate with the precision – single precision results are 4 bytes and double precision results are 8 bytes. The following prodedure must be used for reading the result from the stack:

1. A LOW is established on the C/$\overline{D}$ input.
2. The $\overline{CS}$ input is made LOW. When $\overline{WR}$ and $\overline{RD}$ inputs are both HIGH, the $\overline{PAUSE}$ output follows the $\overline{CS}$ input, thus $\overline{PAUSE}$ will be LOW.
3. After appropriate set up time (see timing diagrams), the $\overline{RD}$ input is made LOW.
4. Sometime after this, $\overline{PAUSE}$ will return HIGH indicating that the data is available on the DB0-DB7 lines. This data will remain on the DB0-DB7 lines as long as the $\overline{RD}$ input remains LOW.
5. Anytime after $\overline{PAUSE}$ goes HIGH, the $\overline{RD}$ input can return HIGH to complete transaction.
6. The $\overline{CS}$ and C/$\overline{D}$ inputs can change after appropriate hold time requirements are satisfied (see timing diagram).
7. Repeat this procedure until all bytes appropriate for the precision of the result are popped out.

Reading of the stack does not alter its data; it only adjusts the byte pointer. If more data is popped than the capacity of the stack, the internal byte pointer will wrap around and older data will be read again, consistent with the LIFO stack.

READING STATUS REGISTER

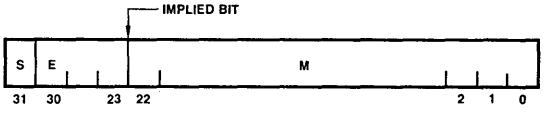
The Am9512 status register can be read without any regard to whether a command is in progress or not. The only implication that has to be considered is the effect this might have on the END and ERR outputs discussed in the signal descriptions.

The following procedure must be followed to accomplish status register reading.

1. Establish HIGH on the C/$\overline{D}$ input.
2. Establish LOW on the $\overline{CS}$ input. Whenever $\overline{WR}$ and $\overline{RD}$ inputs are HIGH, $\overline{PAUSE}$ will follow the $\overline{CS}$ input. Thus, $\overline{PAUSE}$ will go LOW.
3. After appropriate set up time (see timing diagram) $\overline{RD}$ is made LOW.
4. Sometime after the HIGH to LOW transition of $\overline{RD}$, $\overline{PAUSE}$ will become HIGH indicating that status register contents are available on the DB0-DB7 lines. These lines will contain this information as long as $\overline{RD}$ is LOW.
5. The $\overline{RD}$ input can be returned HIGH anytime after $\overline{PAUSE}$ goes HIGH.
6. The C/$\overline{D}$ input and $\overline{CS}$ input can change after satisfying appropriate hold time requirements (see timing diagram).

DATA FORMATS

The Am9512 handles floating-point quantities in two different formats – single precision and double precision. The single precision quantities are 32-bits long as shown below.



Bit 31:

S = Sign of the mantissa. 1 represents negative and 0 represents positive.

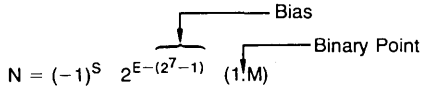
Bits 23-30

E = These 8-bits represent a biased exponent. The bias is $2^7 - 1 = 127$

Bits 0-22

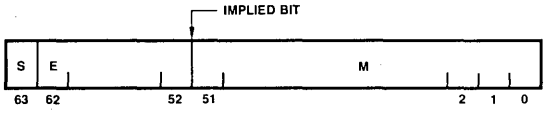
M = 23-bit mantissa. Together with the sign bit, the mantissa represents a signed fraction in sign-magitude notation. There is an implied 1 beyond the most significant bit (bit 22) of the mantissa. In other words, the mantissa is assumed to be a 24-bit normalized quantity and the most significant bit which will always be 1 due to normalization is implied. The Am9512 restores this implied bit internally before performing arithmetic; normalizes the result and strips the implied bit before returning the results to the external data bus. The binary point is between the implied bit and bit 22 of the mantissa.

The quantity N represented by the above notation is



Provided E ≠ 0 or all 1's.

A double precision quantity consists of the mantissa sign bit(s), an 11 bit biased exponent (E), and a 52-bit mantissa (M). The bias for double precision quantities is $2^{10} - 1$. The double precision format is illustrated below.



Bit 63:

S = Sign of the mantissa. 1 represents negative and 0 represents positive.

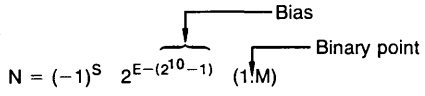
Bits 52-62

E = These 11 bits represent a biased exponent. The bias is $2^{10} - 1 = 1023$.

Bit 0-51

M = 52-bit mantissa. Together with the sign bit, the mantissa represents a signed fraction in sign-magnitude notation. There is an implied 1 beyond the most significant bit (bit 51) of the mantissa. In other words, the mantissa is assumed to a 53-bit normalized quantity and the most significant bit, which will always be a 1 due to normalization, is implied. The Am9512 restores this implied bit internally before performing arithmetic; normalizes the result and strips the implied bit before returning the result to the external data bus. The binary point is between the implied bit and bit 51 of the mantissa.

The quantity N represented by the above notation is



Provided E ≠ 0 or all 1's.

STATUS REGISTER

The Am9512 contains an 8-bit status register with the following format.

BUSY	SIGN S	ZERO Z	RESERVED	DIVIDE EXCEPTION D	EXPONENT UNDERFLOW U	EXPONENT OVERFLOW V	RESERVED
7	6	5	4	3	2	1	0

Bit 0 and bit 4 are reserved. Occurrence of exponent oerflow (V), exponent underflow (U) and divide exception (D) are indicated by bits 1, 2 and 3 respectively. An attempt to divide by zero is the only divide exception. Bits 5 and 6 represent a zero result and the sign of a result respectively. Bit 7 (Busy) of the status register indicates if the Am9512 is currently busy executing a command. All the bits are initialized to zero upon reset. Also, executing a CLR (Clear Status) command will result in all zero status register bits. A zero in Bit 7 indicates that the Am9512 is not busy and a new command may be initiated. As soon as a new command is issued, Bit 7 becomes 1 to indicate the device is busy and remains 1 until the command execution is complete, at which time it will become 0. As soon as a new command is issued, status register bits 0, 1, 2, 3, 4, 5 and 6 are cleared to zero. The status bits will be set as required during the command execution. Hence, as long as bit 7 is 1, the remainder of the status register bit indications should not be relied upon unless the ERR occurs. The following is a detailed status bit description.

Bit 0 Reserved

Bit 1 Exponent overflow (V): When 1, this bit indicates that exponent overflow has occurred. Cleared to zero otherwise.

Bit 2 Exponent Underflow (U): When 1, this bit indicates that exponent underflow has occurred. Cleared to zero otherwise.

Bit 3 Divide Exception (D): When 1, this bit indicates that an attempt to divide by zero is made. Cleared to zero otherwise.

Bit 4 Reserved

Bit 5 Zero (Z): When 1, this bit indicates that the result returned to TOS after a command is all zeros. Cleared to zero otherwise.

Bit 6 Sign (S): When 1, this bit indicates that the result returned to TOS is negative. Cleared to zero otherwise.

Bit 7 Busy: When 1, this bit indicates the Am9512 is in the process of executing a command. It will become zero after the command execution is complete.

All other status register bits are valid when the Busy bit is zero.

ALGORITHMS OF FLOATING-POINT ARITHMETIC

1. Floating Point to Decimal Conversion

   As an introduction to floating-point arithmetic, a brief description of the Decimal equivalent of the Am9512 floating-point format should help the reader to understand and verify the validity of the arithmetic operations. The Am9512 single precision format is used for the following discussions. With a minor modification of the field lengths, the discussion would also apply to the double precision format.

   There are three parts in a floating point number:

   a. The sign – the sign applies to the sign of the number. Zero means the number is positive or zero. One means the number is negative.

b. The exponent – the exponent represents the magnitude of the number. The Am9512 single precision format has an excess 127_{10} notation which means the code representation is 127_{10} higher than the actual value. The following are a few examples of actual versus coded exponent.

Actual	Coded
$+127_{10}$	$+254_{10}$
0	127_{10}
-126_{10}	$+1_{10}$

c. The mantissa – the mantissa is a 23-bit value with the binary point to the left of the most significant bit. There is a hidden 1 to the left of the binary point so the mantissa is always less than 2 and greater than or equal to 1.

To find the Decimal equivalent of the floating point number, the mantissa is multiplied by 2 to the power of the actual exponent. The number is negated if the sign bit = 1. The following are two examples of conversion:

Example 1

Floating Point No. = 0 10000011 10000000000000000000000B

Sign — Exponent — Mantissa

Coded Exponent = 10000011B
Actual Exponent = 10000011B – 01111111B = 00000100B = 4_{10}
Mantissa = 1.1 10000000000000000000000B
= 1 + 1/2 + 1/4 = 1.75_{10}
Decimal No. = 2^4 x 1.75 = 16 x 1.75 = 28_{10}

Example 2

Floating Point No. = 1 01111010 01100000000000000000000B

Sign — Exponent — Mantissa

Code Exponent = 01111010B
Actual Exponent = 01111010B – 01111111B = 11111011B = -5_{10}
Mantissa = 1.0 11000000000000000000000B
= 1 + 1/4 + 1/8 = 1.375_{10}
Decimal No. = -2^{-5} x 1.375 = $-.04296875_{10}$

2. Unpacking of the Floating-Point Numbers
The Am9512 unpacks the floating point number into three parts before any of the arithmetic operation. The number is divided into three parts as described in Section 1. The sign and exponent are copied from the original number as 1 and 8-bit numbers respectively. The mantissa is stored as a 24-bit number. The least significant 23 bits are copied from the original number and the MSB is set to 1. The binary point is assumed to the right of the MSB.

The abbreviations listed below are used in the following sections of algorithm description:

SIGN – Sign of Result
EXP – Exponent of Result
MAN – Mantissa of Result
SIGN (TOS) – Sign of Top of Stack
EXP (TOS) – Exponent of Top of Stack
MAN (TOS) – Mantissa of Top of Stack
SIGN (NOS) – Sign of Next on Stack
EXP (NOS) – Exponent of Next on Stack
MAN (NOS) – Mantissa of Next on Stack

3. Floating-Point Add/Subtract
The floating-point add and subtract essentially use the same algorithm. The only difference is that floating-point subtract changes the sign of the floating-point number at top of stack and then performs the floating-point add.

The following is a step by step description of a floating-point add algorithm (Figure 1):

a. Unpack TOS and NOS.
b. The exponent of TOS is compared to the exponent of NOS.
c. If the exponents are equal, go to step f.
d. Right shift the mantissa of the number with the smaller exponent.
e. Increment the smaller exponent and go to step b.
f. Set sign of result to sign of larger number.
g. Set exponent of result to exponent of larger number.
h. If sign of the two numbers are not equal, go to m.
i. Add Mantissas.
j. Right shift resultant mantissa by 1 and increment exponent of result by 1.
k. If MSB of exponent changes from 1 to 0 as a result of the increment, set overflow status.
l. Round if necessary and exit.
m. Subtract smaller mantissa from larger mantissa.
n. Left shift mantissa and decrement exponent of result.
o. If MSB of exponent changes from 0 to 1 as a result of the decrement, set underflow status and exit.
p. If the MSB of the resultant mantissa = 0, go to n.
q. Round if necessary and exit.

4. Floating-Point Multiply
Floating-point multiply basically involves the addition of the exponents and multiplication of the mantissas. The following is a step by step description of a floating multiplication algorithm (Figure 2):

a. Check if TOS or NOS = 0.
b. If either TOS or NOS = 0, Set result to 0 and exit.
c. Unpack TOS and NOS.
d. Convert EXP (TOS) and EXP (NOS) to unbiased form.
EXP (TOS) = EXP (TOS) -127_{10}
EXP (NOS) = EXP (NOS) -127_{10}
e. Add exponents.
EXP = EXP (TOS) + EXP (NOS)
f. If MSB of EXP (TOS) = MSB of EXP (NOS) = 0 and MSB of EXP = 1, then set overflow status and exit.
g. If MSB of EXP (TOS) = MSB of EXP (NOS) = 1 and MSB of EXP = 0, then set underflow status and exit.
h. Convert Exponent back to biased form.
EXP = EXP + 127_{10}
i. If sign of TOS = sign of NOS, set sign of result to 0, else set sign of result to 1.
j. Multiply mantissa.
k. If MSB of resultant = 1, right shift mantissa by 1 and increment exponent of resultant.
l. If MSB of exponent changes from 1 to 0 as a result of the increment, set overflow status.
m. Round if necessary and exit.

5. Floating-Point Divide
The floating-point divide basically involves the subtraction of exponents and the division of mantissas. The following is a step by step description of a division algorithm (Figure 3).

a. If TOS = 0, set divide exception error and exit.
b. If NOS = 0, set result to 0 and exit.
c. Unpack TOS and NOS.
d. Convert EXP (TOS) and EXP (NOS) to unbiased form.
EXP (TOS) = EXP (TOS) – 127_{10}
EXP (NOS) = EXP (NOS) – 127_{10}
e. Subtract exponent of TOS from exponent of NOS.
EXP = EXP (NOS) – EXP (TOS)
f. If MSB of EXP (NOS) = 0, MSB of EXP (TOS) = 1 and MSB of EXP = 1, then set overflow status and exit.
g. If MSB of EXP (NOS) = 1, MSB of EXP (TOS) = 0, and MSB of EXP = 0, then set underflow status and exit.

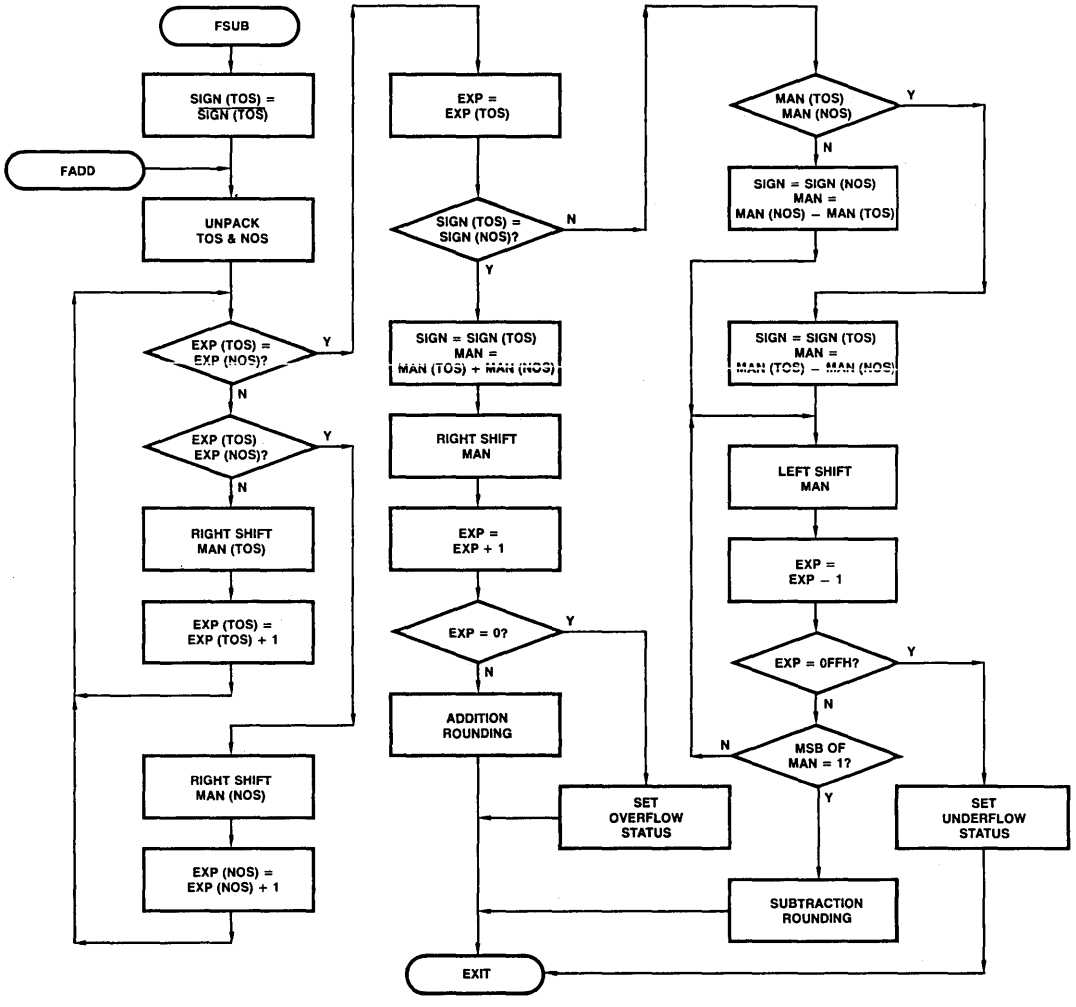


Figure 1. Conceptual Floating-Point Addition/Subtraction.

MOS-205

h. Add bias to exponent of result.
   $EXP = EXP + 127_{10}$
i. If sign of TOS = sign of NOS, set sign of result to 0, else set sign of result to 1.
j. Divide mantissa of NOS by mantissa of TOS.
k. If MSB = 0, left shift mantissa and decrement exponent of resultant, else go to n.
l. If MSB of exponent changes from 0 to 1 as a result of the decrement, set underflow status.
m. Go to k.
n. Round if necessary and exit.

The algorithms described above provide the user a means of verifying the validity of the result. They do not necessarily reflect the exact internal sequence of the Am9512.

6. Rounding

The Am9512 adopts a rounding algorithm that is consistent with the Intel® standard for floating-point arithmetic. The following description is an excerpt from the paper published in proceedings of Compsac 77, November 1977, pp. 107-112 by Dr. John F. Palmer of Intel Corporation.

The method used for doing the rounding during floating-point arithmetic is known as "Round to Even", i.e., if the resultant number is exactly halfway between two floating point numbers, the number is rounded to the nearest floating-point number whose LSB of the mantissa is 0. In order to simplify the explanation, the algorithms will be illustrated with 4-bit arithmetic. The existence of an accumulator will be assumed as shown:

OF	B1	B2	B3	B4	G	R	ST

The bit labels denote:

OF – The overflow bit
B1-B4 – The 4 mantissa bits
G – The Guard bit
R – The Rounding bit
ST – The "Sticky" bit

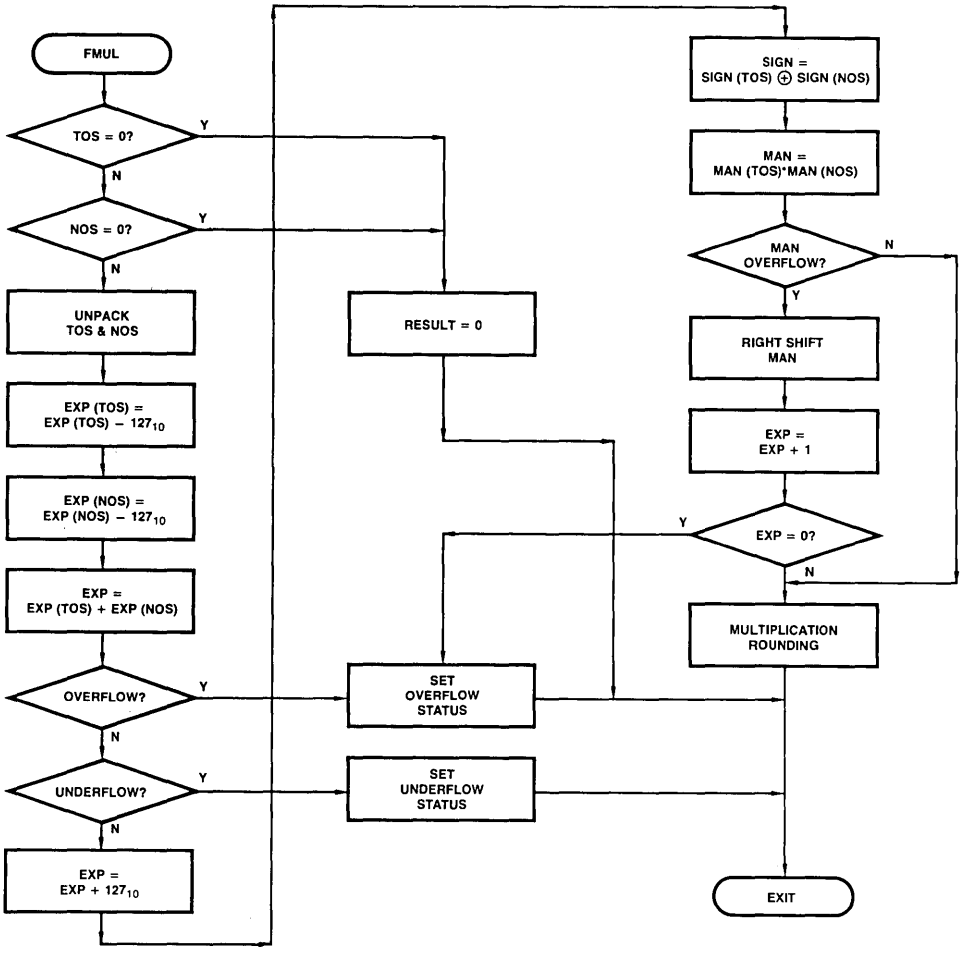


Figure 2. Conceptual Floating-Point Multiplication.

MOS-206

The Sticky bit is set to one if any ones are shifted right of the rounding bit in the process of denormalization. If the Sticky bit becomes set, it remains set throughout the operation. All shifting in the Accumulator involves the OF, G, R and ST bits. The ST bit is not affected by left shifts but, zeros are introduced into OF by right shifts.

Rounding during addition of magnitudes – add 1 to the G position, then if G=R=ST=0, set B4 to 0 ("Rounding to Even").

Rounding during subtraction of magnitudes – if more than one left shift was performed, no rounding is needed, otherwise round the same way as addition of magnitudes.

Rounding during multiplication – let the normalized double length product be:

B1	B2	B3	B4	B5	B6	B7	B8

Then G=B5, R=B6, ST=B7 V B8. The rounding is then performed as in addition of magnitudes.

Rounding during division – let the first six bits of the normalized quotient be

B1	B2	B3	B4	B5	B6

Then G=B5, R=B6, ST=0 if and only if remainder = 0. The rounding is then performed as in addition of magnitudes.

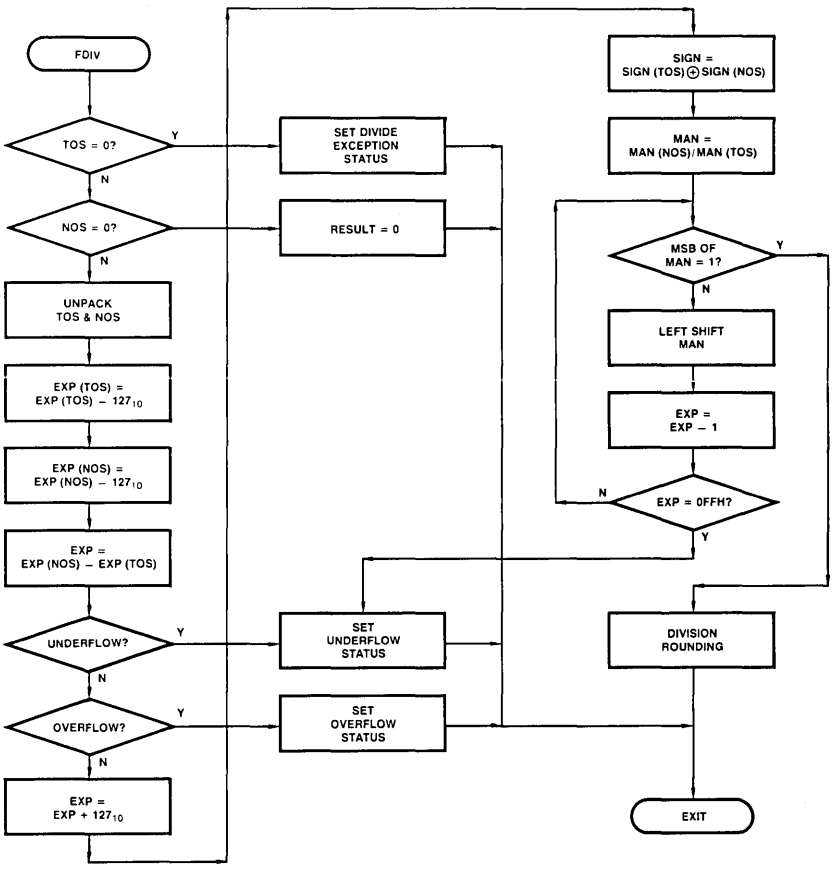


Figure 3. Conceptual Floating-Point Division.

MOS-207

4

CHSD

CHANGE SIGN DOUBLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	1	0	1

Hex Coding: AD IF SRE = 1
2D IF SRE = 0

Execution Time: See Table 2

Description:

The sign of the double precision TOS operand A is complemented. The double precision result R is returned to TOS. If the double precision operand A is zero, then the sign is not affected. The status bit S and Z indicate the sign of the result and if the result is zero. The status bits U, V and D are always cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS

BEFORE		AFTER
A	TOS	R
B	NOS	B

CHSS

CHANGE SIGN SINGLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	0	1

Hex Coding: 85 IF SRE = 1
05 IF SRE = 0

Execution Time: See Table 2

Description:

The sign of the single precision operand A at TOS is complemented. The single precision result R is returned to TOS. If the exponent field of A is zero, all bits of R will be zeros. The status bits S and Z indicate the sign of the result and if the result is zero. The status bits U, V and D are cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS

BEFORE		AFTER
A	← TOS →	R
B	← NOS →	B
C		C
D		D

CLR

CLEAR STATUS

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	0	0

Hex Coding: 80 IF SRE = 1
00 IF SRE = 0

Execution Time: 4 clock cycles

Description:
The status bits S, Z, D, U, V are cleared to zero. The stack is not affected. This essentially is a no operation command as far as operands are concerned.

Status Affected: S, Z, D, U, V always zero.

DADD

DOUBLE PRECISION FLOATING-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	0	0	1

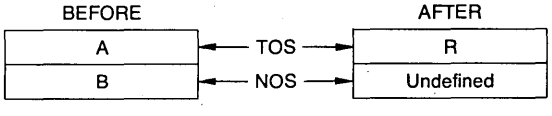
Hex Coding: A9 IF SRE = 1
29 IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand A from TOS is added to the double precision operand B from NOS. The result is rounded to obtain the final double precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



DSUB

DOUBLE PRECISION FLOATING-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	0	1	0

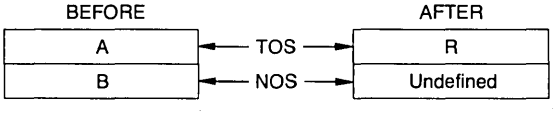
Hex Coding: AA IF SRE = 1
2A IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand A at TOS is subtracted from the double precision operand B at NOS. The result is rounded to obtain the final double precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



DMUL

DOUBLE PRECISION FLOATING-POINT MULTIPLY

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	0	1	1

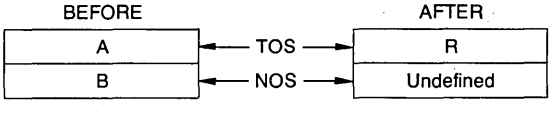
Hex Coding: AB IF SRE = 1
2B IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand A from TOS is multiplied by the double precision operand B from NOS. The result is rounded to obtain the final double precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



DDIV

DOUBLE PRECISION FLOATING-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Code:	SRE	0	1	0	1	1	0	0

Hex Coding: AC IF SRE = 1
2C IF SRE = 0

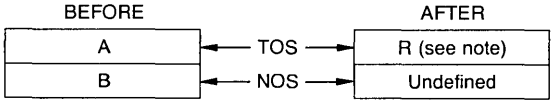
Execution Time: See Table 2

Description:

The double precision operand B from NOS is divided by the double precision operand A from TOS. The result (quotient) is rounded to obtain the final double precision result R which is returned to TOS. The status bits, S, Z, D, U and V are affected to report sign of the result, if the result is zero, attempt to divide by zero, exponent underflow and exponent overflow respectively.

Status Affected: S, Z, D, U, V

STACK CONTENT



Note: If A is zero, then R = B (Divide exception).

SADD

SINGLE PRECISION FLOATING-POINT ADD

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	0	1

Hex Coding: 81 IF SRE = 1
01 IF SRE = 0

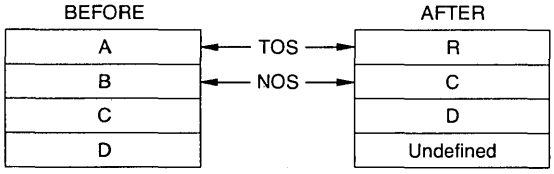
Execution Time: See Table 2

Description:

The single precision operand A from TOS is added to the single precision operand B from NOS. The result is rounded to obtain the final single precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report the sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENT



SSUB

SINGLE PRECISION FLOATING-POINT SUBTRACT

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	1	0

Hex Coding: 82 IF SRE = 1
02 IF SRE = 0

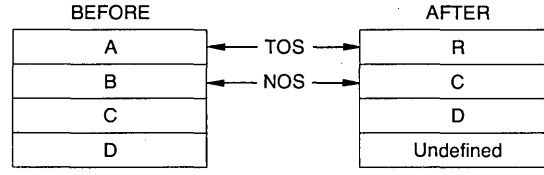
Execution Time: See Table 2

Description:

The single precision operand A at TOS is subtracted from the single precision operand B at NOS. The result is rounded to obtain the final single precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report the sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



SMUL

SINGLE PRECISION FLOATING-POINT MULTIPLY

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	0	1	1

Hex Coding: 83 IF SRE = 1
03 IF SRE = 0

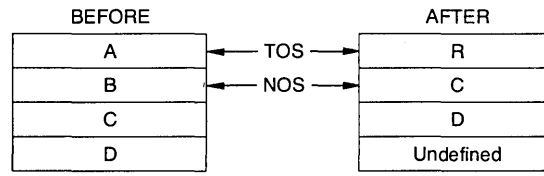
Execution Time: See Table 2

Description:

The single precision operand A from TOS is multiplied by the single precision operand B from NOS. The result is rounded to obtain the final single precision result R which is returned to TOS. The status bits S, Z, U and V are affected to report the sign of the result, if the result is zero, exponent underflow and exponent overflow respectively. The status bit D will be cleared to zero.

Status Affected: S, Z, U, V. (D always zero.)

STACK CONTENTS



4

SDIV

SINGLE PRECISION FLOATING-POINT DIVIDE

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	0	0

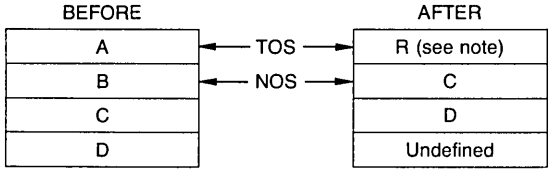
Hex Coding: 84 IF SRE = 1
04 IF SRE = 0

Execution Time: See Table 2

Description:
The single precision operand B from NOS is divided by the single precision operand A from TOS. The result (quotient) is rounded to obtain the final result R which is returned to TOS. The status bits S, Z, D, U and V are affected to report the sign of the result, if the result is zero, attempt to divide by zero, exponent underflow and exponent overflow respectively.

Status Affected: S, Z, D, U, V

STACK CONTENTS



Note: If exponent field of A is zero then R = B (Divide exception).

POPS

POP STACK SINGLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	1	1

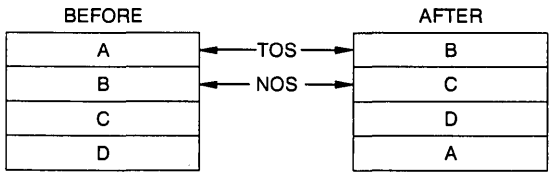
Hex Coding: 87 IF SRE = 1
07 IF SRE = 0

Execution Time: See Table 2

Description:
The single precision operand A is popped from the stack. The internal stack control mechanism is such that A will be written at the bottom of the stack. The status bits S and Z are affected to report the sign of the new operand at TOS and if it is zero, respectively. The status bits U, V and D will be cleared to zero. Note that only the exponent field of the new TOS is checked for zero, if it is zero status bit Z will set to 1.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS



PTOD

PUSH STACK DOUBLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	1	0	1	1	1	0

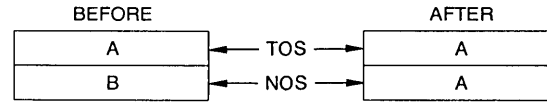
Hex Coding: AE IF SRE = 1
2E IF SRE = 0

Execution Time: See Table 2

Description:
The double precision operand A from the TOS is pushed back on to the stack. This is effectively a duplication of A into two consecutive stack locations. The status S and Z are affected to report sign of the new TOS and if the new TOS is zero respectively. The status bits U, V and D will be cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS



PTOS

PUSH STACK SINGLE PRECISION

	7	6	5	4	3	2	1	0
Binary Coding:	SRE	0	0	0	0	1	1	0

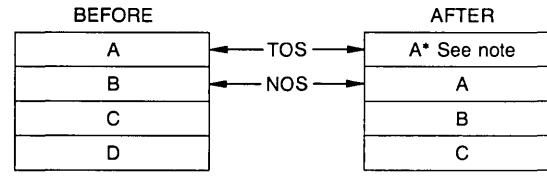
Hex Coding: 86 IF SRE = 1
06 IF SRE = 0

Execution Time: See Table 2

Description:
This instruction effectively pushes the single precision operand from TOS on to the stack. This amounts to duplicating the operand at two locations in the stack. However, if the operand at TOS prior to the PTOS command has only its exponent field as zero, the new content of the TOS will all be zeroes. The contents of NOS will be an exact copy of the old TOS. The status bits S and Z are affected to report the sign of the new TOS and if the content of TOS is zero, respectively. The status bits U, V and D will be cleared to zero.

Status Affected: S, Z. (U, V, D always zero.)

STACK CONTENTS



Note: A* = A if Exponent field of A is not zero.
A* = 0 if Exponent field of A is zero.

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VDD with Respect to VSS	−0.5V to +15.0V
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	2.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VSS	VCC	VDD
Am9512DC	$0°C \leq T_A \leq +70°C$	0V	+5.0V ±5%	+12V ±5%
Am9512DM	$-55°C \leq T_A \leq +125°C$	0V	+5.0V ±10%	+12V ±10%

4

ELECTRICAL CHARACTERISTICS Over Operating Range (Note 1)

Parameters	Description	Test Conditions	Min.	Typ.	Max.	Units
VOH	Output HIGH Voltage	IOH = −200μA	3.7			Volts
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage		2.0		VCC	Volts
VIL	Input LOW Voltage		−0.5		0.8	Volts
IIX	Input Load Current	$VSS \leq VI \leq VCC$			±10	μA
IOZ	Data Bus Leakage	VO = 0.4V			10	μA
		VO = VCC			10	
ICC	VCC Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
IDD	VDD Supply Current	T_A = +25°C		50	90	mA
		T_A = 0°C			95	
		T_A = −55°C			100	
CO	Output Capacitance	fc = 1.0MHz, Inputs = 0V		8	10	pF
CI	Input Capacitance			5	8	pF
CIO	I/O Capacitance			10	12	pF

SWITCHING CHARACTERISTICS

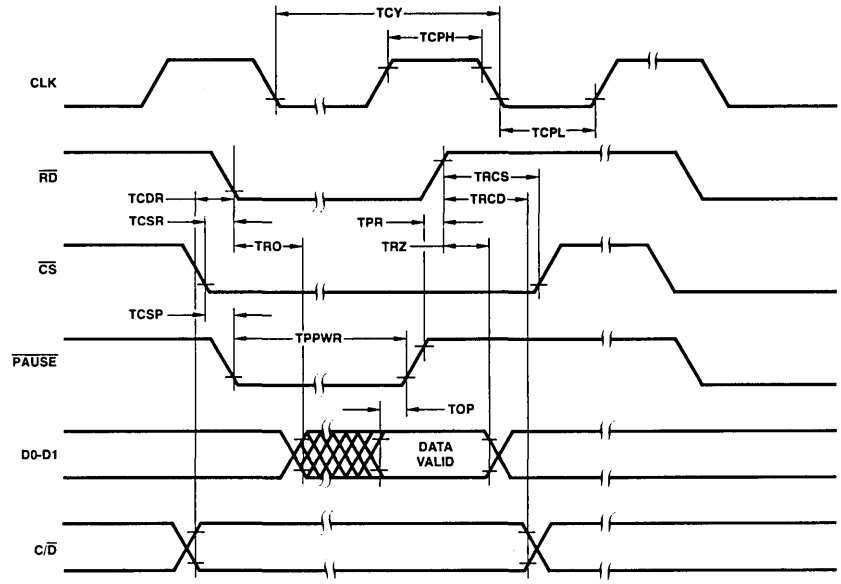
Parameters	Description		Am9512DC Min	Am9512DC Max	Am9512-1DC Min	Am9512-1DC Max	Units
TAPW	$\overline{EACK}$ LOW Pulse Width		100		75		ns
TCDR	C/$\overline{D}$ to $\overline{RD}$ LOW Set-up Time		0		0		ns
TCDW	C/$\overline{D}$ to $\overline{WR}$ LOW Set-up Time		0		0		ns
TCPH	Clock Pulse HIGH Width		200	500	140	500	ns
TCPL	Clock Pulse LOW Width		240		160		ns
TCSP	$\overline{CS}$ LOW to $\overline{PAUSE}$ LOW Delay (Note 5)		150		100		ns
TCSR	$\overline{CS}$ to $\overline{RD}$ LOW Set-up Time		0		0		ns
TCSW	$\overline{CS}$ LOW to $\overline{WR}$ LOW Set-up Time		0		0		ns
TCY	Clock Period		480	5000	320	2000	ns
TDW	Data Valid to $\overline{WR}$ HIGH Delay			150		100	ns
TEAE	$\overline{EACK}$ LOW to END LOW Delay			200		175	ns
TEHPHR	END HIGH to $\overline{PAUSE}$ HIGH Data Read when Busy			5.5TCY+300		5.5TCY+200	ns
TEHPHW	END HIGH to $\overline{PAUSE}$ HIGH Write when Busy			200		175	ns
TEPW	END HIGH Pulse Width		400		300		ns
TEX	Execution Time		See Table 2				ns
TOP	Data Bus Output Valid to $\overline{PAUSE}$ HIGH Delay		0		0		ns
TPPWR	$\overline{PAUSE}$ LOW Pulse Width Read	Data	3.5TCY+50	5.5TCY+300	3.5TCY+50	5.5TCY+200	ns
		Status	1.5TCY+50	3.5TCY+300	1.5TCY+50	3.5TCY+200	
TPPWRB	END HIGH to $\overline{PAUSE}$ HIGH Read when Busy	Data	See Table 2				ns
		Status	1.5TCY+50	3.5TCY+300	1.5TCY+50	3.5TCY+200	
TPPWW	$\overline{PAUSE}$ LOW Pulse Width Write when Not Busy			TCSW+50		TCSW+50	ns
TPPWWB	$\overline{PAUSE}$ LOW Pulse Width Write when Busy		See Table 2				ns
TPR	$\overline{PAUSE}$ HIGH to Read HIGH Hold Time		0		0		ns
TPW	$\overline{PAUSE}$ HIGH to Write HIGH Hold Time		0		0		ns
TRCD	$\overline{RD}$ HIGH to C/$\overline{D}$ Hold Time		0		0		ns
TRCS	$\overline{RD}$ HIGH to $\overline{CS}$ HIGH Hold Time		0		0		ns
TRO	$\overline{RD}$ LOW to Data Bus On Delay		50		50		ns
TRZ	$\overline{RD}$ HIGH to Data Bus Off Delay		50	200	50	150	ns
TSAPW	SVACK LOW Pulse Width		100		75		ns
TSAR	SVACK LOW to SVREQ LOW Delay			300		200	ns
TWCD	$\overline{WR}$ HIGH to C/D Hold Time		60		30		ns
TWCS	$\overline{WR}$ HIGH to CS HIGH Hold Time		60		30		ns
TWD	$\overline{WR}$ HIGH to Data Bus Hold Time		20		20		ns

NOTES:

1. Typical values are for T_A = 25°C, nominal supply voltages and nominal processing parameters.
2. Switching parameters are listed in alphabetical order.
3. Test conditions assume transition times of 20ns or less, output loading of one TTL gate plus 100pF and timing reference levels of 0.8V and 2.0V.
4. END HIGH pulse width is specified for $\overline{EACK}$ tied to VSS. Otherwise TEAE applies.
5. $\overline{PAUSE}$ is pulled low for both command and data operations.
6. TEX is the execution time of the current command (see the Command Execution Times table).

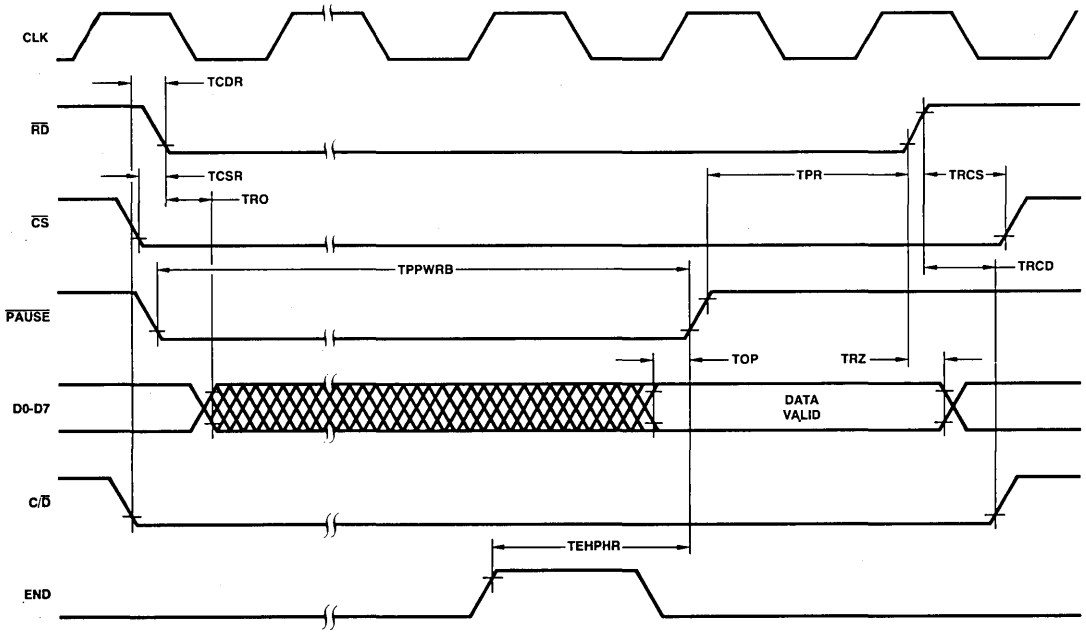
TIMING DIAGRAMS

READ OPERATION



MOS-208

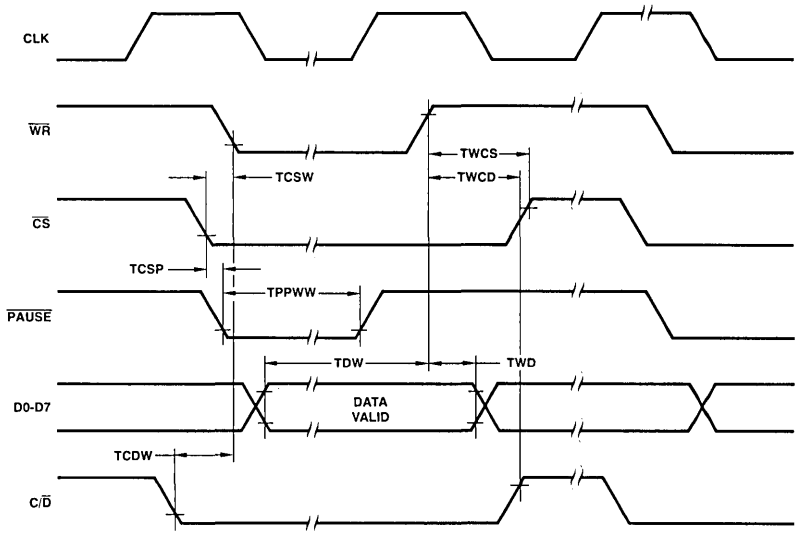
OPERAND READ WHEN Am9512 IS BUSY



MOS-209

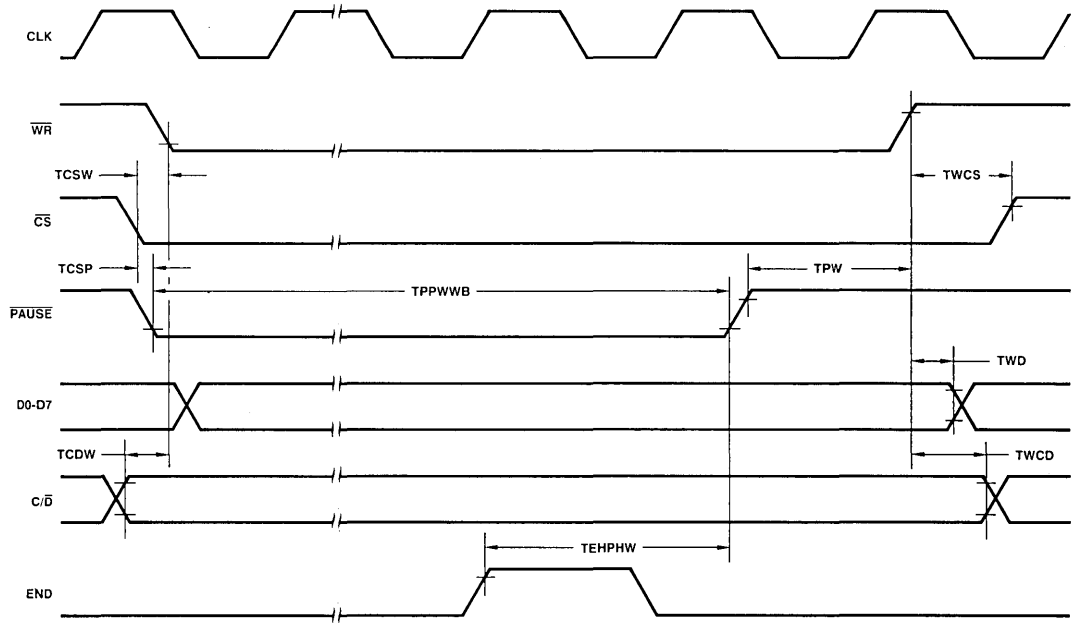
TIMING DIAGRAMS (Cont.)

OPERAND ENTRY



MOS-210

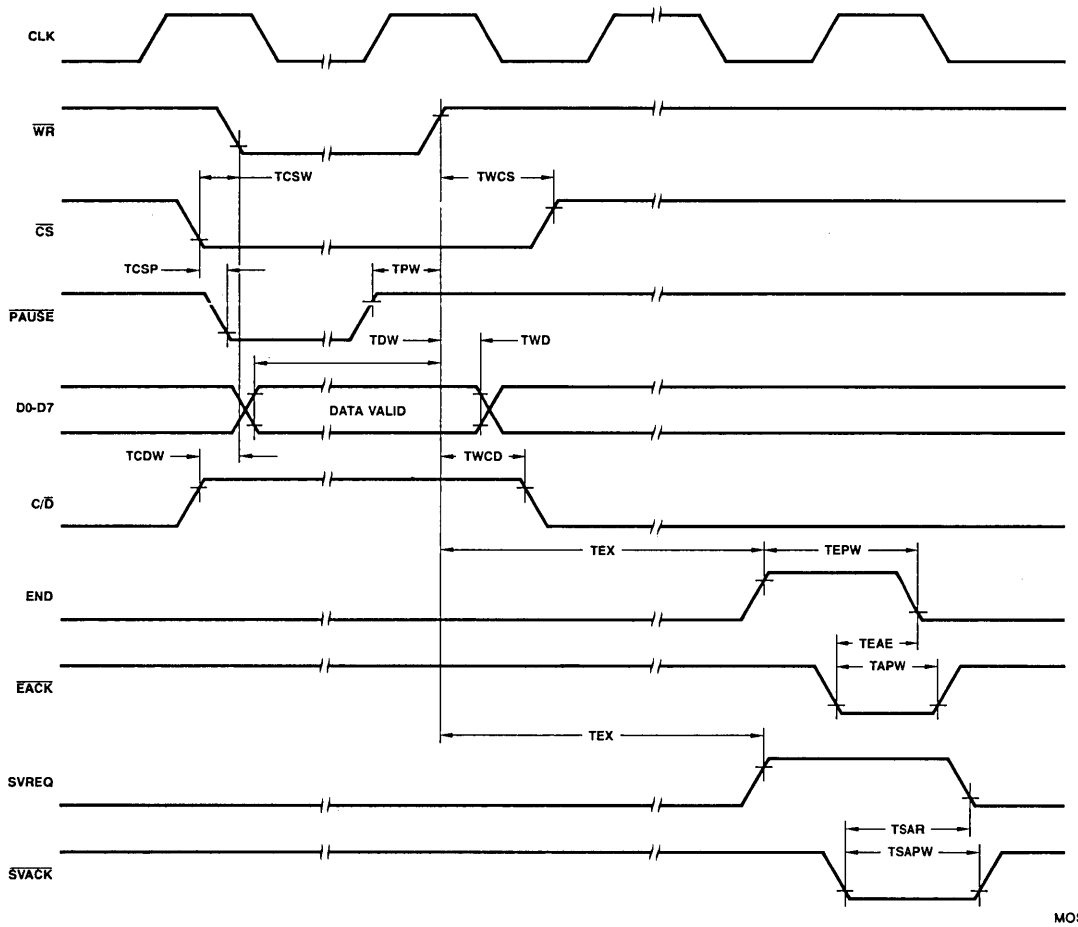
COMMAND OR DATA WRITE WHEN Am9512 IS BUSY



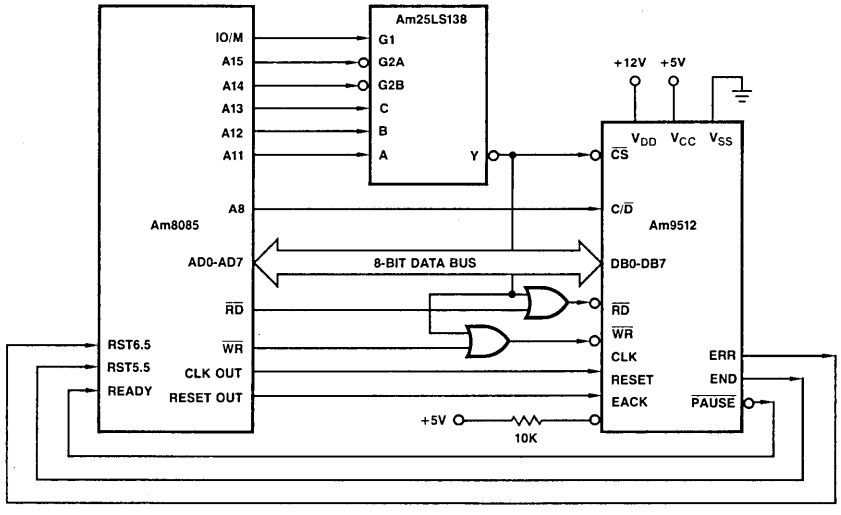
MOS-211

TIMING DIAGRAMS (Cont.)

COMMAND INITIATION



MOS-212



MOS-213

Figure 1. Am9512 to Am8085 Interface.

Am9517A

Multimode DMA Controller

DISTINCTIVE CHARACTERISTICS

- Four independent DMA channels, each with separate registers for Mode Control, Current Address, Base Address, Current Word Count and Base Word Count.
- Transfer modes: Block, Demand, Single Word, Cascade
- Independent autoinitialization of all channels
- Memory-to-memory transfers
- Memory block initialization
- Address increment or decrement
- Master system disable
- Enable/disable control of individual DMA requests
- Directly expandable to any number of channels
- End of Process input for terminating transfers
- Software DMA requests
- Independent polarity control for DREQ and DACK signals
- Compressed timing option speeds transfers – up to 2M words/second
- +5 volt power supply
- Advanced N-channel silicon gate MOS technology
- 40 pin Hermetic DIP package
- 100% MIL-STD-883 reliability assurance testing

GENERAL DESCRIPTION

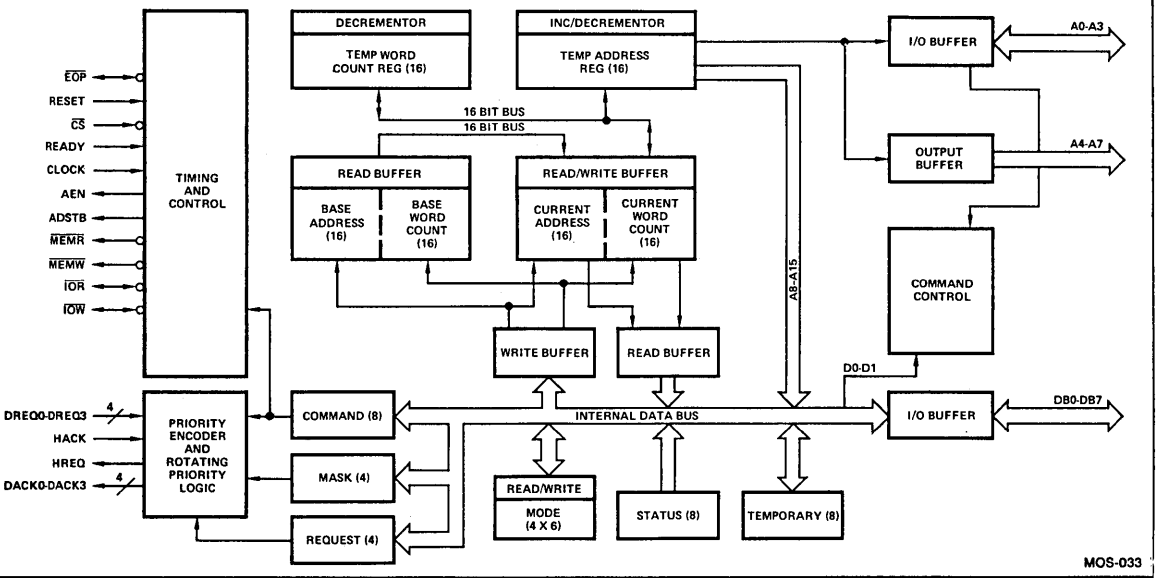
The Am9517A Multimode Direct Memory Access (DMA) Controller is a peripheral interface circuit for microprocessor systems. It is designed to improve system performance by allowing external devices to directly transfer information to or from the system memory. Memory-to-memory transfer capability is also provided. The Am9517A offers a wide variety of programmable control features to enhance data throughput and system optimization and to allow dynamic reconfiguration under program control.

The Am9517A is designed to be used in conjunction with an external 8-bit address register such as the Am74LS373. It contains four independent channels and may be expanded to any number of channels by cascading additional controller chips.

The three basic transfer modes allow programmability of the types of DMA service by the user. Each channel can be individually programmed to Autoinitialize to its original condition following an End of Process ($\overline{EOP}$).

Each channel has a full 64K address and word count capability. An external $\overline{EOP}$ signal can terminate a DMA or memory-to-memory transfer. This is useful for block search or compare operations using external comparators or for intelligent peripherals to abort erroneous services.

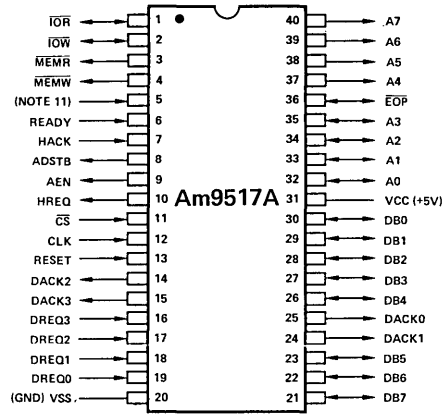
BLOCK DIAGRAM



ORDERING INFORMATION

Package Type	Ambient Temperature	Maximum Clock Frequency	
		3MHz	4MHz
Hermetic DIP/ Molded DIP	$0°C \leq T_A \leq +70°C$	AM9517ADC/PC AM9517A-1DC/PC	AM9517A-4DC/PC
Hermetic DIP	$-55°C \leq T_A \leq +125°C$	AM9517ADM	

CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

Figure 1. MOS-034

INTERFACE SIGNAL DESCRIPTION

VCC: +5 Volt Supply
VSS: Ground

CLK (Clock, Input)

This input controls the internal operations of the Am9517A and its rate of data transfers. The input may be driven at up to 3MHz for the standard Am9517A and up to 4MHz for the Am9517A-4.

$\overline{CS}$ (Chip Select, Input)

Chip Select is an active low input used to select the Am9517A as an I/O device during an I/O Read or I/O Write by the host CPU. This allows CPU communication on the data bus. During multiple transfers to or from the Am9517A by the host CPU, $\overline{CS}$ may be held low providing $\overline{IOR}$ or $\overline{IOW}$ is toggled following each transfer.

RESET (Reset, Input)

Reset is an asynchronous active high input which clears the Command, Status, Request and Temporary registers. It also clears the first/last flip/flop and sets the Mask register. Following a Reset the device is in the Idle cycle.

READY (Ready, Input)

Ready is an input used to extend the memory read and write pulses from the Am9517A to accommodate slow memories or I/O peripheral devices.

HACK (Hold Acknowledge, Input)

The active high Hold Acknowledge from the CPU indicates that control of the system buses has been relinquished.

DREQ0-DREQ3 (DMA Request, Input)

The DMA Request lines are individual asynchronous channel request inputs used by peripheral circuits to obtain DMA service. In Fixed Priority, DREQ0 has the highest priority and DREQ3 has the lowest priority. Polarity of DREQ is programmable. Reset initializes these lines to active high.

DB0-DB7 (Data Bus, Input/Output)

The Data Bus lines are bidirectional three-state signals connected to the system data bus. The outputs are enabled during the I/O Read by the host CPU, permitting the CPU to examine the contents of an Address register, the Status register, the Temporary register or a Word Count register. The Data Bus is enabled to input data during a host CPU I/O write, allowing the CPU to program the Am9517A control registers. During DMA cycles the most significant eight bits of the address are output onto the data bus to be strobed into an external latch by ADSTB. In memory-to-memory operations data from the source memory location comes into the Am9517A's Temporary register on the read-from-memory half of the operation. On the write-to-memory half of the operation, the data bus outputs the Temporary register data into the destination memory location.

$\overline{IOR}$ (I/O Read, Input/Output)

I/O Read is a bidirectional active low three-state line. In the Idle cycle, it is an input control signal used by the CPU to read the control registers. In the Active cycle, it is an output control signal used by the Am9517A to access data from a peripheral during a DMA Write transfer.

$\overline{IOW}$ (I/O Write, Input/Output)

I/O Write is a bidirectional active low three-state line. In the Idle cycle it is an input control signal used by the CPU to load information into the Am9517A. In the Active cycle it is an output control signal used by the Am9517A to load data to the peripheral during a DMA Read transfer.

Write operations by the CPU to the Am9517A require a rising $\overline{WR}$ edge following each data byte transfer. It is not sufficient to hold the $\overline{IOW}$ pin low and toggle $\overline{CS}$.

$\overline{EOP}$ (End of Process, Input/Output)

$\overline{EOP}$ is an active low bidirectional open-drain signal providing information concerning the completion of DMA service. When a channel's Word Count goes to zero, the Am9517A pulses $\overline{EOP}$ low to provide the peripheral with a completion signal. $\overline{EOP}$ may also be pulled low by the peripheral to cause premature completion. The reception of $\overline{EOP}$, either internal or external, causes the currently active channel to terminate the service, to set its TC bit in the Status register and to reset its request bit. If Autoinitialization is selected for the channel, the current registers will be updated from the base registers. Otherwise the channel's mask bit will be set and the register contents will remain unaltered.

During memory-to-memory transfers, $\overline{EOP}$ will be output when the TC for channel 1 occurs. $\overline{EOP}$ always applies to the channel with an active DACK; external $\overline{EOP}$S are disregarded in DACK0-DACK3 are all inactive.

Because $\overline{EOP}$ is an open-drain signal, an external pullup resistor is required. Values of 3.3K or 4.7K are recommended; the $\overline{EOP}$ pin can not sink the current passed by a 1K pullup.

A0-A3 (Address, Input/Output)

The four least significant address lines are bidirectional 3-state signals. During DMA Idle cycles they are inputs and allow the host CPU to load or read control registers. When the DMA is active, they are outputs and provide the lower 4-bits of the output address.

A4-A7 (Address, Output)

The four most significant address lines are three-state outputs and provide four bits of address. These lines are enabled only during DMA service.

HREQ (Hold Request, Output)

The Hold Request to the CPU is used by the DMA to request control of the system bus. Software requests or unmasked DREQs cause the Am9517A to issue HREQ.

DACK0-DACK3 (DMA Acknowledge, Output)

The DMA Acknowledge lines indicate that a channel is active. In many systems they will be used to select a peripheral. Only one DACK will be active at a time and none will be active unless the DMA is in control of the bus. The polarity of these lines is programmable. Reset initializes them to active-low.

AEN (Address Enable, Output)

Address Enable is an active high signal used to disable the system bus during DMA cycles to enable the output of the external latch which holds the upper byte of the address. Note that during DMA transfers HACK and AEN should be used to deselect all other I/O peripherals which may erroneously be accessed as programmed I/O during the DMA operation. The Am9517A automatically deselects itself by disabling the $\overline{CS}$ input during DMA transfers.

ADSTB (Address Strobe, Output)

The active high Address Strobe is used to strobe the upper address byte from DB0-DB7 into an external latch.

$\overline{MEMR}$ (Memory Read, Output)

The Memory Read signal is an active low three-state output used to access data from the selected memory location during a memory-to-peripheral or a memory-to-memory transfer.

Name	Size	Number
Base Address Registers	16 bits	4
Base Word Count Registers	16 bits	4
Current Address Registers	16 bits	4
Current Word Count Registers	16 bits	4
Temporary Address Register	16 bits	1
Temporary Word Count Register	16 bits	1
Status Register	8 bits	1
Command Register	8 bits	1
Temporary Register	8 bits	1
Mode Registers	6 bits	4
Mask Register	4 bits	1
Request Register	4 bits	1

Figure 2. Am9517A Internal Registers.

$\overline{MEMW}$ (Memory Write, Output)

The Memory Write signal is an active low three-state output used to write data to the selected memory location during a peripheral-to-memory or a memory-to-memory transfer.

FUNCTIONAL DESCRIPTION

The Am9517A block diagram includes the major logic blocks and all of the internal registers. The data interconnection paths are also shown. Not shown are the various control signals between the blocks. The Am9517A contains 344 bits of internal memory in the form of registers. Figure 2 lists these registers by name and shows the size of each. A detailed description of the registers and their functions can be found under Register Description.

The Am9517A contains three basic blocks of control logic. The Timing Control block generates internal timing and external control signals for the Am9517A. The Program Command Control block decodes the various commands given to the Am9517A by the microprocessor prior to servicing a DMA Request. It also decodes each channel's Mode Control word. The Priority Encoder block resolves priority contention among DMA channels requesting service simultaneously.

The Timing Control block derives internal timing from the clock input. In Am9080A systems this input will usually be the $\phi 2$ TTL clock from an Am8224. However, any appropriate system clock will suffice.

DMA Operation

The Am9517A is designed to operate in two major cycles. These are called Idle and Active cycles. Each device cycle is made up of a number of states. The Am9517A can assume seven separate states, each composed of one full clock period. State 1 (S1) is the inactive state. It is entered when the Am9517A has no valid DMA requests pending. While in S1, the DMA controller is inactive but may be in the Program Condition, being programmed by the processor. State 0 (S0) is the first state of a DMA service. The Am9517A has requested a hold but the processor has not yet returned an acknowledge. An acknowledge from the CPU will signal that transfers may begin. S1, S2, S3 and S4 are the working states of the DMA service. If more time is needed to complete a transfer than is available with normal timing, wait states (SW) can be inserted before S4 by the use of the Ready line on the Am9517A.

Memory-to-memory transfers require a read-from and a write-to-memory to complete each transfer. The states, which resemble the normal working states, use two digit numbers for identification. Eight states are required for each complete transfer. The first four states (S11, S12, S13, S14) are used for the read-from-memory half and the last four states (S21, S22, S23 and S24) for the write-to-memory half of the transfer. The Temporary Data register is used for intermediate storage of the memory byte.

IDLE Cycle

When no channel is requesting service, the Am9517A will enter the Idle cycle and perform "S1" states. In this cycle the Am9517A will sample the DREQ lines every clock cycle to determine if any channel is requesting a DMA service. The device will also sample $\overline{CS}$, looking for an attempt by the microprocessor to write or read the internal registers of the Am9517A. When $\overline{CS}$ is low and HACK is low the Am9517A enters the Program Condition. The CPU can now establish, change or inspect the internal definition of the part by reading from or writing to the internal registers. Address lines A0-A3 are inputs to the device and select which registers will be read or written. The $\overline{IOR}$ and $\overline{IOW}$ lines are used to select and time reads or writes. Due to the

number and size of the internal registers, an internal flip/flop is used to generate an additional bit of address. This bit is used to determine the upper or lower byte of the 16-bit Address and Word Count registers. The flip/flop is reset by Master Clear or Reset. A separate software command can also reset this flip/flop.

Special software commands can be executed by the Am9517A in the Program Condition. These commands are decoded as sets of addresses when both $\overline{CS}$ and $\overline{IOW}$ are active and do not make use of the data bus. Functions include Clear First/Last Flip/Flop and Master Clear.

ACTIVE CYCLE

When the Am9517A is in the Idle cycle and a channel requests a DMA service, the device will output a HREQ to the microprocessor and enter the Active cycle. It is in this cycle that the DMA service will take place, in one of four modes:

Single Transfer Mode: In Single Transfer mode, the Am9517A will make a one-byte transfer during each HREQ/HACK handshake. When DREQ goes active, HREQ will go active. After the CPU responds by driving HACK active, a one-byte transfer will take place. Following the transfer, HREQ will go inactive, the word count will be decremented and the address will be either incremented or decremented. When the word count goes to zero a Terminal Count (TC) will cause an Autoinitialize if the channel has been programmed to do so.

To perform a single transfer, DREQ must be held active only until the corresponding DACK goes active. If DREQ is held continuously active, HREQ will go inactive following each transfer and then will go active again and a new one-byte transfer will be made following each rising edge of HACK. In 8080A/9080A systems this will ensure one full machine cycle of execution between DMA transfers. Details of timing between the Am9517A and other bus control protocols will depend upon the characteristics of the microprocessor involved.

Block Transfer Mode: In Block Transfer mode, the Am9517A will continue making transfers until a TC (caused by the word count going to zero) or an external End of Process ($\overline{EOP}$) is encountered. DREQ need be held active only until DACK becomes active. An autoinitialize will occur at the end of the service if the channel has been programmed for it.

Demand Transfer Mode: In Demand Transfer mode the device will continue making transfers until a TC or external $\overline{EOP}$ is encountered or until DREQ goes inactive. Thus, the device requesting service may discontinue transfers by bringing DREQ inactive. Service may be resumed by asserting an active DREQ once again. During the time between services when the microprocessor is allowed to operate, the intermediate values of address and word count may be read from the Am9517A Current Address and Current Word Count registers. Autoinitialization will only occur following a TC or $\overline{EOP}$ at the end of service. Following Autoinitialization, an active-going DREQ edge is required to initiate a new DMA service.

Cascade Mode: This mode is used to cascade more than one Am9517A together for simple system expansion. The HREQ and HACK signals from the additional Am9517A are connected to the DREQ and DACK signals of a channel of the initial Am9517A. This allows the DMA requests of the additional device to propagate through the priority network circuitry of the preceding device. The priority chain is preserved and the new device must wait for its turn to acknowledge requests. Since the cascade channel in the initial device is used only for prioritizing the additional device, it does not output any address or control signals of its own. These would conflict with the outputs of the active channel in the added device. The Am9517A will respond to DREQ with DACK but all other outputs except HREQ will be disabled.

Figure 3 shows two additional devices cascaded into an initial device using two of the previous channels. This forms a two level DMA system. More Am9517As could be added at the second level by using the remaining channels of the first level. Additional devices can also be added by cascading into the channels of the second level devices forming a third level.

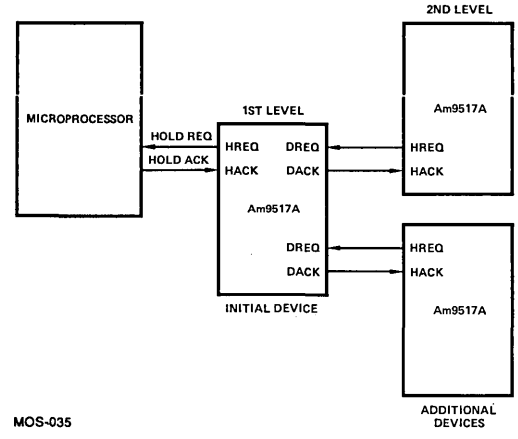


Figure 3. Cascaded Am9517As.

TRANSFER TYPES

Each of the three active transfer modes can perform three different types of transfers. These are Read, Write and Verify. Write transfers move data from an I/O device to the memory by activating $\overline{IOR}$ and $\overline{MEMW}$. Read transfers move data from memory to an I/O device by activating $\overline{MEMR}$ and $\overline{IOW}$. Verify transfers are pseudo transfers; the Am9517A operates as in Read or Write transfers generating addresses, responding to $\overline{EOP}$, etc., however, the memory and I/O control lines remain inactive.

Memory-to-Memory: The Am9517A includes a block move capability that allows blocks of data to be moved from one memory address space to another. When Bit C0 in the Command register is set to a logical 1, channels 0 and 1 will operate as memory-to-memory transfer channels. Channel 0 forms the source address and channel 1 forms the destination address. The channel 1 word count is used. A memory-to-memory transfer is initiated by setting a software DMA request for channel 0. Block Transfer Mode should be used for memory-to-memory. When channel 0 is programmed for a fixed source address, a single source word may be written into a block of memory.

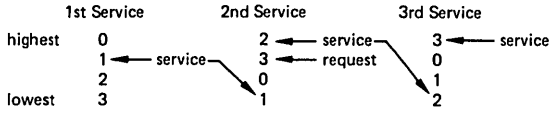
When setting up the Am9517A for memory-to-memory operation, it is suggested that both channels 0 and 1 be masked out. Further, the channel 0 word count should be initialized to the same value used in channel 1. No DACK outputs will be active during memory-to-memory transfers.

The Am9517A will respond to external $\overline{EOP}$ signals during memory-to-memory transfers. Data comparators in block search schemes may use this input to terminate the service when a match is found. The timing of memory-to-memory transfers may be found in Timing Diagram 4.

Autointialize: By programming a bit in the Mode register a channel may be set up for an Autoinitialize operation. During Autoinitialization, the original values of the Current Address and Current Word Count registers are automatically restored from the Base Address and Base Word Count registers of that channel following $\overline{EOP}$. The base registers are loaded simultaneously with the current registers by the microprocessor and remain unchanged throughout the DMA service. The mask bit is not set by $\overline{EOP}$ when the channel is in Autoinitialize. Following Autoinitialize the channel is ready to repeat its service without CPU intervention.

Priority: The Am9517A has two types of priority encoding available as software selectable options. The first is Fixed Priority which fixes the channels in priority order based upon the descending value of their number. The channel with the lowest priority is 3 followed by 2, 1 and the highest priority channel, 0.

The second scheme is Rotating Priority. The last channel to get service becomes the lowest priority channel with the others rotating accordingly. With Rotating Priority in a single chip DMA system, any device requesting service is guaranteed to be recognized after no more than three higher priority services have occurred. This prevents any one channel from monopolizing the system.



The priority encoder selects the highest priority channel requesting service on each active-going HACK edge. Once a channel is started, its operation will not be suspended if a request is received by a higher priority channel. The high priority channel will only gain control after the lower priority channel releases HREQ. When control is passed from one channel to another, the CPU will always gain bus control. This ensures generation of rising HACK edge to be used to initiate selection of the new highest-priority requesting channel.

Compressed Timing: In order to achieve even greater throughput where system characteristics permit, the Am9517A can compress the transfer time to two clock cycles. From Timing Diagram 3 it can be seen that state S3 is used to extend the access time of the read pulse. By removing state S3 the read pulse width is made equal to the write pulse width and a transfer consists only of state S2 to change the address and state S4 to perform the read/write. S1 states will still occur when A8-A15 need updating (see Address Generation). Timing for compressed transfers is found in Timing Diagram 6.

Address Generation: In order to reduce pin count, the Am9517A multiplexes the eight higher order address bits on the data lines. State S1 is used to output the higher order address bits to an external latch from which they may be placed on the address bus. The falling edge of Address Strobe (ADSTB) is used to load these bits from the data lines to the latch. Address Enable (AEN) is used to enable the bits onto the address bus through a 3-state enable. The lower order address bits are output by the Am9517A directly. Lines A0-A7 should be connected to the address bus. Timing Diagram 3 shows the time relationships between CLK, AEN, ADSTB, DB0-DB7 and A0-A7.

During Block and Demand Transfer mode services which include multiple transfers, the addresses generated will be sequential. For many transfers the data held in the external address latch will remain the same. This data need only change when a carry or borrow from A7 to A8 takes place in the normal sequence of addresses. To save time and speed transfers, the Am9517A executes S1 states only when updating of A8-A15 in the latch is necessary. This means for long services, S1 states may occur only once every 256 transfers, a savings of 255 clock cycles for each 256 transfers.

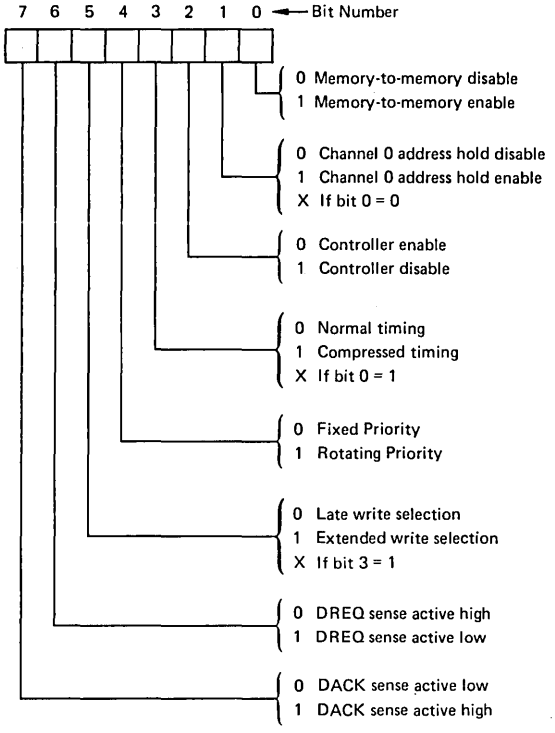
REGISTER DESCRIPTION

Current Address Register: Each channel has a 16-bit Current Address register. This register holds the value of the address used during DMA transfers. The address is automatically incremented or decremented after each transfer and the intermediate values of the address are stored in the Current Address register during the transfer. This register is written or read by the microprocessor in successive 8-bit bytes. It may also be reinitialized by an Autoinitialize back to its original value. Autoinitialization takes place only after an $\overline{EOP}$.

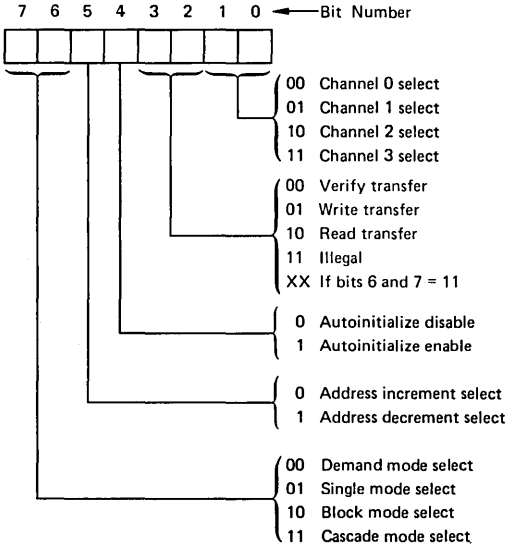
Current Word Count Register: Each channel has a 16-bit Current Word Count register. This register should be programmed with, and will return on a CPU read, a value one less than the number of words to be transferred. The word count is decremented after each transfer. The intermediate value of the word count is stored in the register during the transfer. When the value in the register goes to zero, a TC will be generated. This register is loaded or read in successive 8-bit bytes by the microprocessor in the Program Condition. Following the end of a DMA service it may also be reinitialized by an Autoinitialize back to its original value. Autoinitialize can occur only when an $\overline{EOP}$ occurs. Note that the contents of the Word Count register will be FFFF (hex) following on internally generated $\overline{EOP}$.

Base Address and Base Word Count Registers: Each channel has a pair of Base Address and Base Word Count registers. These 16-bit registers store the original values of their associated current registers. During Autoinitialize these values are used to restore the current registers to their original values. The base registers are written simultaneously with their corresponding current register in 8-bit bytes during DMA programming by the microprocessor. Accordingly, writing to these registers when intermediate values are in the Current registers will overwrite the intermediate values. The Base registers cannot be read by the microprocessor.

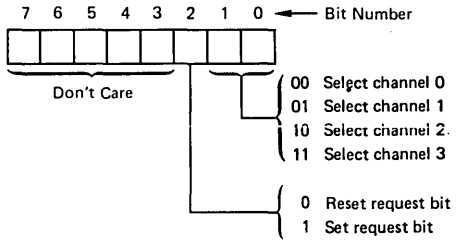
Command Register: This 8-bit register controls the operation of the Am9517A. It is programmed by the microprocessor in the Program Condition and is cleared by Reset. The following table lists the function of the command bits. See Figure 4 for address coding.



Mode Register: Each channel has a 6-bit Mode register associated with it. When the register is being written to by the microprocessor in the Program Condition, bits 0 and 1 determine which channel Mode register it to be written.

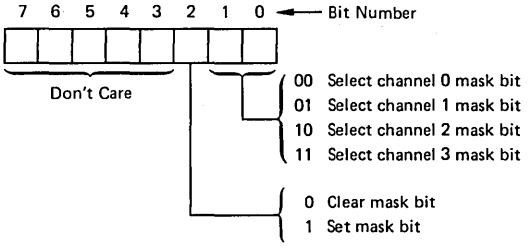


Request Register: The Am9517A can respond to requests for DMA service which are initiated by software as well as by a DREQ. Each channel has a request bit associated with it in the 4-bit Request register. These are nonmaskable and subject to prioritization by the Priority Encoder network. Each register bit is set or reset separately under software control or is cleared upon generation of a TC or external EOP. The entire register is cleared by a Reset. To set or reset a bit, the software loads the proper form of the data word. See Figure 4 for address coding.

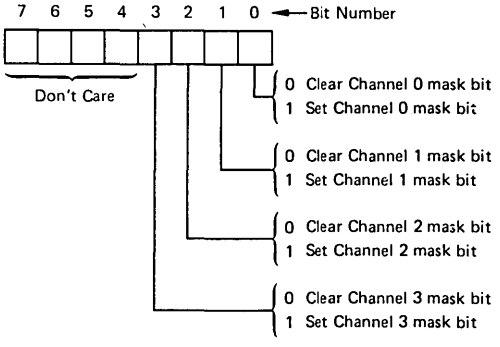


Software requests will be serviced only if the channel is in Block mode. When initiating a memory-to-memory transfer, the software request for channel 0 should be set.

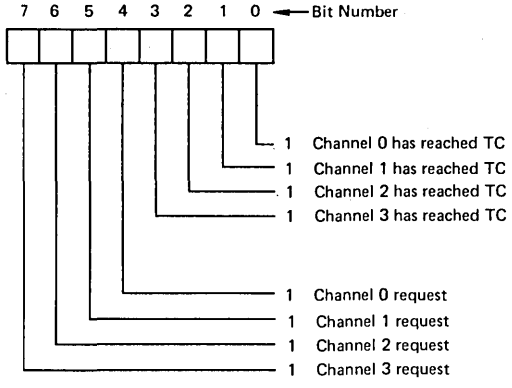
Mask Register: Each channel has associated with it a mask bit which can be set to disable the incoming DREQ. Each mask bit is set when its associated channel produces an EOP if the channel is not programmed for Autoinitialize. Each bit of the 4-bit Mask register may also be set or cleared separately under software control. The entire register is also set by a Reset. This disables all DMA requests until a clear Mask register instruction allows them to occur. The instruction to separately set or clear the mask bits is similar in form to that used with the Request register. See Figure 4 for instruction addressing.



All four bits of the Mask Register may also be written with a single command.



Status Register: The Status registers may be read out of the Am9517A by the microprocessor. It indicates which channels have reached a terminal count and which channels have pending DMA requests. Bits 0-3 are set each time a TC is reached by that channel, including after each Autoinitialization. These bits are cleared by Reset and each Status Read. Bits 4-7 are set whenever their corresponding channel is requesting service.



Temporary Register: The Temporary register is used to hold data during memory-to-memory transfers. Following the completion of the transfers, the last word moved can be read by the microprocessor in the Program Condition. The Temporary register always contains the last byte transferred in the previous memory-to-memory operation, unless cleared by a Reset.

Software Commands: There are two special software commands which can be executed in the Program Condition. They do not depend on any specific bit pattern on the data bus. The two software commands are:

Clear First/Last Flip/Flop: This command may be issued prior to writing or reading Am9517A address or word count information. This initializes the flip/flop to a known state so that subsequent accesses to register contents by the microprocessor will address lower and upper bytes in the correct sequence.

Master Clear: This software instruction has the same effect as the hardware Reset. The Command, Status, Request, Temporary and Internal First/Last Flip/Flop registers are cleared and the Mask register is set. The Am9517A will enter the Idle cycle.

Figure 4 lists the address codes for the software commands.

Interface Signals						
A3	A2	A1	A0	$\overline{IOR}$	$\overline{IOW}$	Operation
1	0	0	0	0	1	Read Status Register
1	0	0	0	1	0	Write Command Register
1	0	0	1	0	1	Illegal
1	0	0	1	1	0	Write Request Register
1	0	1	0	0	1	Illegal
1	0	1	0	1	0	Write Single Mask Register Bit
1	0	1	1	0	1	Illegal
1	0	1	1	1	0	Write Mode Register
1	1	0	0	0	1	Illegal
1	1	0	0	1	0	Clear Byte Pointer Flip/Flop
1	1	0	1	0	1	Read Temporary Register
1	1	0	1	1	0	Master Clear
1	1	1	0	0	1	Illegal
1	1	1	0	1	0	Illegal
1	1	1	1	0	1	Illegal
1	1	1	1	1	0	Write All Mask Register Bits

Figure 4. Register and Function Addressing.

Channel	Register	Operation	Signals							Internal Flip/Flop	Data Bus DB0–DB7
			$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0		
0	Base & Current Address	Write	0	1	0	0	0	0	0	0	A0-A7
			0	1	0	0	0	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	0	0	0	A0-A7
			0	0	1	0	0	0	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	0	0	1	0	W0-W7
			0	1	0	0	0	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	0	1	0	W0-W7
			0	0	1	0	0	0	1	1	W8-W15
1	Base & Current Address	Write	0	1	0	0	0	1	0	0	A0-A7
			0	1	0	0	0	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	1	0	0	A0-A7
			0	0	1	0	0	1	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	0	1	1	0	W0-W7
			0	1	0	0	0	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	1	1	0	W0-W7
			0	0	1	0	0	1	1	1	W8-W15
2	Base & Current Address	Write	0	1	0	0	1	0	0	0	A0-A7
			0	1	0	0	1	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	0	0	0	A0-A7
			0	0	1	0	1	0	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	1	0	1	0	W0-W7
			0	1	0	0	1	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	0	1	0	W0-W7
			0	0	1	0	1	0	1	1	W8-W15
3	Base & Current Address	Write	0	1	0	0	1	1	0	0	A0-A7
			0	1	0	0	1	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	1	0	0	A0-A7
			0	0	1	0	1	1	0	1	A8-A15
	Base & Current Word Count	Write	0	1	0	0	1	1	1	0	W0-W7
			0	1	0	0	1	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	1	1	0	W0-W7
			0	0	1	0	1	1	1	1	W8-W15

Figure 5. Word Count and Address Register Command Codes.

MAXIMUM RATINGS above which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.5W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	T_A	VCC
Am9517ADC/PC	0°C to +70°C	5.0V ±5%
Am9517A-1DC/PC	0°C to +70°C	5.0V ±5%
Am9517A-4DC/PC	0°C to +70°C	5.0V ±5%
Am9517ADM	−55°C to +125°C	5.0V ±10%

ELECTRICAL CHARACTERISTICS over operating range (Note 1)

Parameter	Description	Test Conditions	Min	Typ	Max	Unit
VOH	Output HIGH Voltage	IOH = −200μA	2.4			Volts
		IOH = −100μA, (HREQ Only)	3.3			
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage		2.0		VCC+0.5	Volts
VIL	Input LOW Voltage		−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VI ≤ VCC	−10		+10	μA
IOZ	Output Leakage Current	VCC ≤ VO ≤ VSS+.40	−10		+10	μA
ICC	VCC Supply Current	T_A = +25°C		65	130	mA
		T_A = 0°C		75	150	
		T_A = −55°C			175	
CO	Output Capacitance	fc = 1.0MHz, Inputs = 0V		4	8	pF
CI	Input Capacitance			8	15	pF
CIO	I/O Capacitance			10	18	pF

NOTES:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. Input timing parameters assume transition times of 20ns or less. Waveform measurement points for both input and output signals are 2.0V for High and 0.8V for Low, unless otherwise noted.
3. Output loading is 1 Standard TTL gate plus 50pF capacitance unless noted otherwise.
4. The new $\overline{IOW}$ or $\overline{MEMW}$ pulse width for normal write will be TCY-100ns and for extended write will be 2TCY-100ns. The net $\overline{IOR}$ or $\overline{MEMR}$ pulse width for normal read will be 2TCY-50ns and for compressed read will be TCY-50ns.
5. TDQ is specified for two different output HIGH levels. TDQ1 is measured at 2.0V. TDQ2 is measured at 3.3V. The value for TDQ2 assumes an external 3.3kΩ pull-up resistor connected from HREQ to VCC.
6. DREQ should be held active until DACK is returned.
7. DREQ and DACK signals may be active high or active low. Timing diagrams assume the active high mode.
8. Output loading on the data bus is 1 Standard TTL gate plus 15pF for the minimum value and 1 Standard TTL gate plus 100pF for the maximum value.
9. Successive read and/or write operations by the external processor to program or examine the controller must be timed to allow at least 600ns for the Am9517A or Am9517A-1 and at least 450ns for the Am9517A-4 as recovery time between active read or write pulses.
10. Parameters are listed in alphabetical order.
11. Pin 5 is an input that should always be at a logic high level. An internal pull-up resistor will establish a logic high when the pin is left floating. Alternatively, pin 5 may be tied to VCC.
12. Signals $\overline{READ}$ and $\overline{WRITE}$ refer to $\overline{IOR}$ and $\overline{MEMW}$ respectively for peripheral-to-memory DMA operations and to $\overline{MEMR}$ and $\overline{IOW}$ respectively for memory-to-peripheral DMA operations.
13. If N wait states are added during the write-to-memory half of a memory-to-memory transfer, this parameter will increase by N (TCY).

SWITCHING CHARACTERISTICS

ACTIVE CYCLE (Notes 2, 3, 10, 11 and 12)

		Am9517A		Am9517A-1		Am9517A-4		
Parameter	Description	Min	Max	Min	Max	Min	Max	Unit
TAEL	AEN HIGH from CLK LOW (S1) Delay Time		300		300		225	ns
TAET	AEN LOW from CLK HIGH (S1) Delay Time		200		200		150	ns
TAFAB	ADR Active to Float Delay from CLK HIGH		150		150		120	ns
TAFC	$\overline{\text{READ}}$ or $\overline{\text{WRITE}}$ Float from CLK HIGH		150		150		120	ns
TAFDB	DB Active to Float Delay from CLK HIGH		250		250		190	ns
TAHR	ADR from $\overline{\text{READ}}$ HIGH Hold Time	TCY-100		TCY-100		TCY-100		ns
TAHS	DB from ADSTB LOW Hold Time	50		50		40		ns
TAHW	ADR from $\overline{\text{WRITE}}$ HIGH Hold Time	TCY-50		TCY-50		TCY-50		ns
TAK	DACK Valid from CLK LOW Delay Time		280		280		220	ns
	$\overline{\text{EOP}}$ HIGH from CLK HIGH Delay Time		250		250		190	ns
	$\overline{\text{EOP}}$ LOW to CLK HIGH Delay Time		250		250		190	ns
TASM	ADR Stable from CLK HIGH		250		250		190	ns
TASS	DB to ADSTB LOW Setup Time	100		100		100		ns
TCH	Clock High Time (Transitions ≤ 10ns)	120		120		100		ns
TCL	Clock Low Time (Transitions ≤ 10ns)	150		150		110		ns
TCY	CLK Cycle Time	320		320		250		ns
TDCL	CLK HIGH to $\overline{\text{READ}}$ or $\overline{\text{WRITE}}$ LOW Delay (Note 4)		270		270		200	ns
TDCTR	$\overline{\text{READ}}$ HIGH from CLK HIGH (S4) Delay Time (Note 4)		270		270		210	ns
TDCTW	$\overline{\text{WRITE}}$ HIGH from CLK HIGH (S4) Delay Time (Note 4)		200		200		150	ns
TDQ1	HREQ Valid from CLK HIGH Delay Time (Note 5)		160		160		120	ns
TDQ2			250		250		190	ns
TEPS	$\overline{\text{EOP}}$ LOW from CLK LOW Setup Time	60		60		45		ns
TEPW	$\overline{\text{EOP}}$ Pulse Width	300		300		225		ns
TFAAB	ADR Float to Active Delay from CLK HIGH		250		250		190	ns
TFAC	$\overline{\text{READ}}$ or $\overline{\text{WRITE}}$ Active from CLK HIGH		200		200		150	ns
TFADB	DB Float to Active Delay from CLK HIGH		300		300		225	ns
THS	HACK valid to CLK HIGH Setup Time	100		100		75		ns
TIDH	Input Data from $\overline{\text{MEMR}}$ HIGH Hold Time	0		0		0		ns
TIDS	Input Data to $\overline{\text{MEMR}}$ HIGH Setup Time	250		250		190		ns
TODH	Output Data from $\overline{\text{MEMW}}$ HIGH Hold Time	20		20		20		ns
TODV	Output Data Valid to $\overline{\text{MEMW}}$ HIGH (Note 13)	200		200		125		ns
TQS	DREQ to CLK LOW (S1, S4) Setup Time	120		120		90		ns
TRH	CLK to READY LOW Hold Time	20		20		20		ns
TRS	READY to CLK LOW Setup Time	100		100		60		ns
TSTL	ADSTB HIGH from CLK HIGH Delay Time		200		200		150	ns
TSTT	ADSTB LOW from CLK HIGH Delay Time		140		140		110	ns

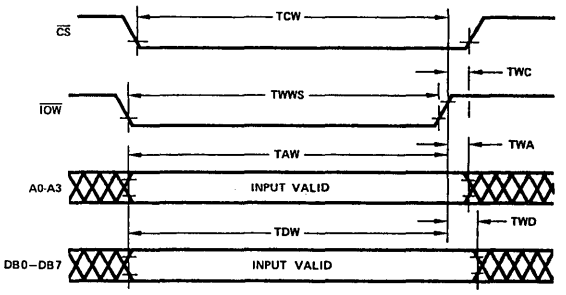
SWITCHING CHARACTERISTICS

PROGRAM CONDITION (IDLE CYCLE)

(Notes 2, 3, 10, 11 and 12)

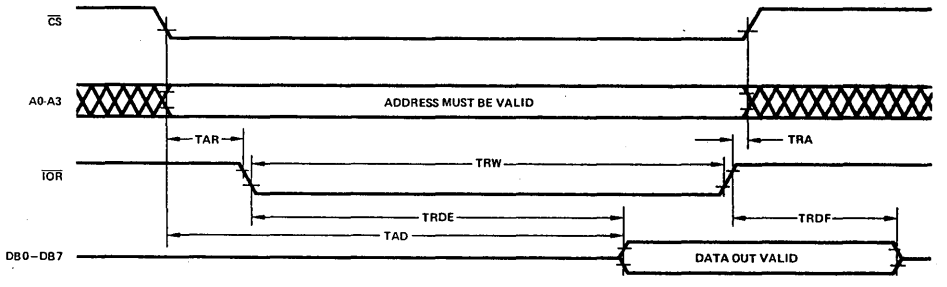
Parameter	Description	Am9517A Min.	Am9517A Max.	Am9517A-1 Min.	Am9517A-1 Max.	Am9517A-4 Min.	Am9517A-4 Max.	Unit
TAR	ADR Valid or $\overline{CS}$ LOW to $\overline{READ}$ LOW	50		50		50		ns
TAW	ADR Valid to $\overline{WRITE}$ HIGH Setup Time	200		200		150		ns
TCW	$\overline{CS}$ LOW to $\overline{WRITE}$ HIGH Setup Time	200		200		150		ns
TDW	Data Valid to $\overline{WRITE}$ HIGH Setup Time	200		200		150		ns
TRA	ADR or $\overline{CS}$ Hold from $\overline{READ}$ HIGH	0		0		0		ns
TRDE	Data Access from $\overline{READ}$ LOW (Note 8)		300		200		200	ns
TDRF	DB Float Delay from $\overline{READ}$ HIGH	20	150	20	100	20	100	ns
TRSTD	Power Supply HIGH to RESET LOW Setup Time	500		500		500		μs
TRSTS	RESET to First $\overline{IOWR}$	2		2		2		TCY
TRSTW	RESET Pulse Width	300		300		300		ns
TRW	$\overline{READ}$ Width	300		300		250		ns
TWA	ADR from $\overline{WRITE}$ HIGH Hold Time	20		20		20		ns
TWC	$\overline{CS}$ HIGH from $\overline{WRITE}$ HIGH Hold Time	20		20		20		ns
TWD	Data from $\overline{WRITE}$ HIGH Hold Time	30		30		30		ns
TWWS	Write Width	200		200		200		ns
TAD	Data Access from ADR Valid, $\overline{CS}$ LOW		350		300		300	ns

SWITCHING WAVEFORMS



Timing Diagram 1. Program Condition Write Timing (Note 9).

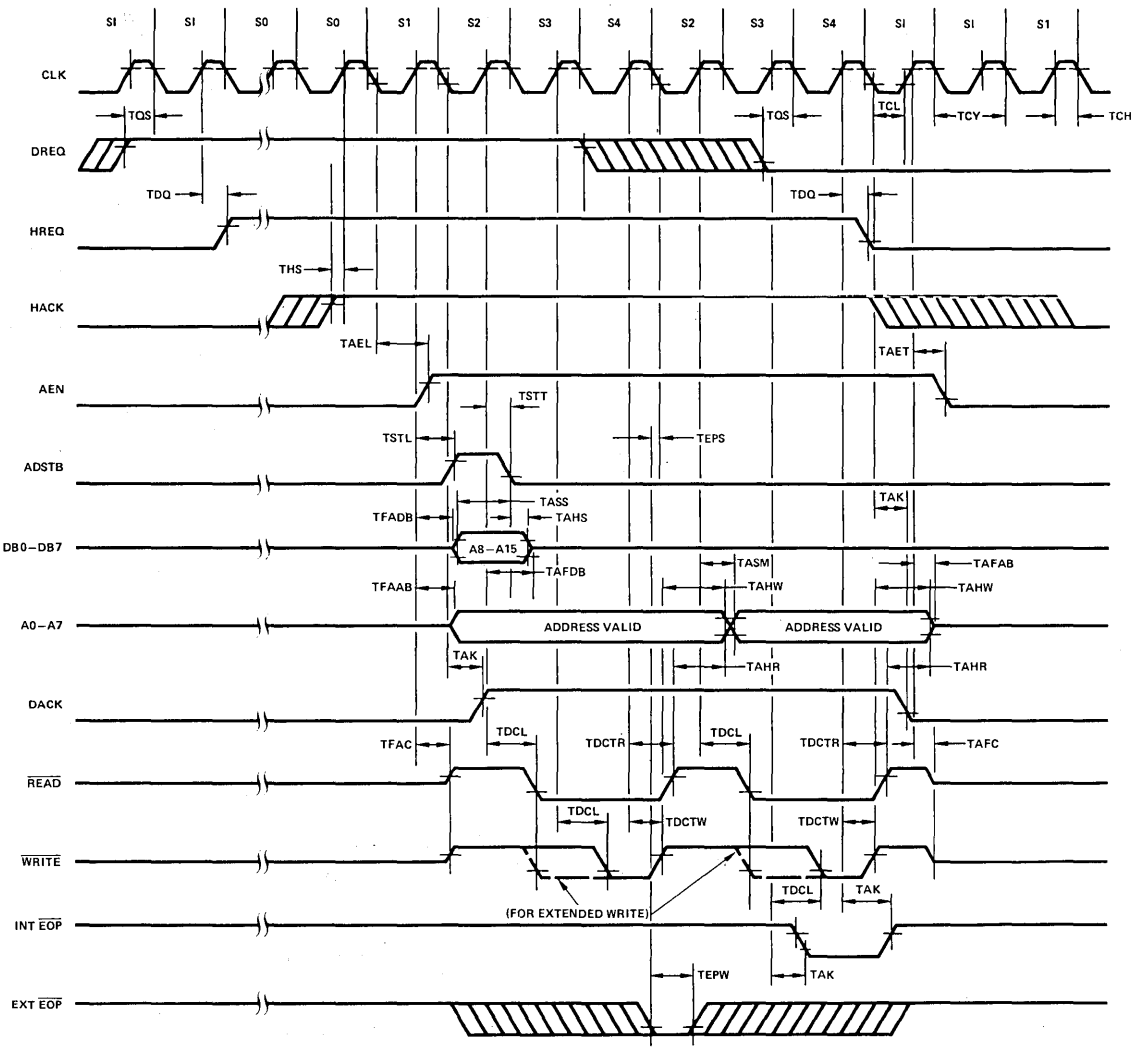
MOS-036



Timing Diagram 2. Program Condition Read Cycle (Note 9).

MOS-037

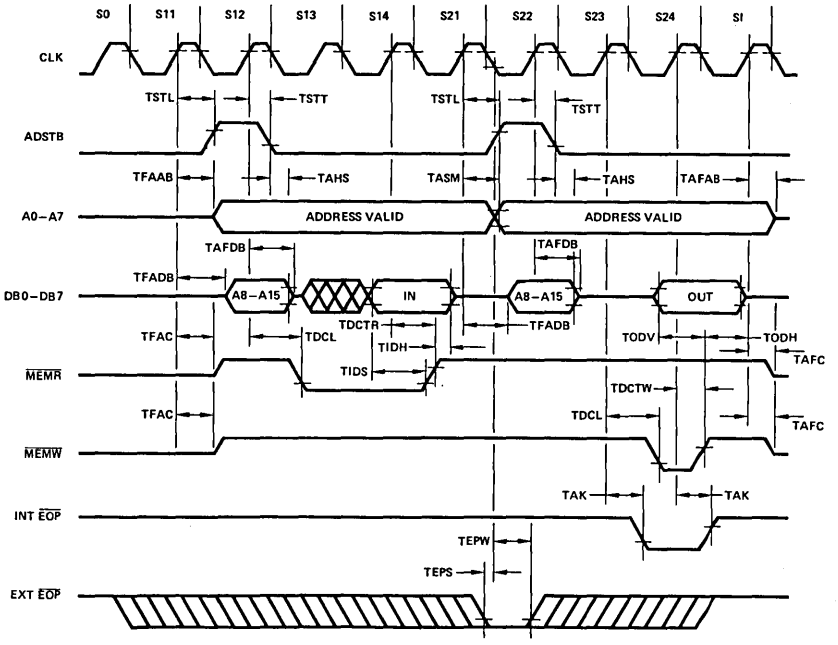
SWITCHING WAVEFORMS (Cont.)



Timing Diagram 3. Active Cycle Timing Diagram.

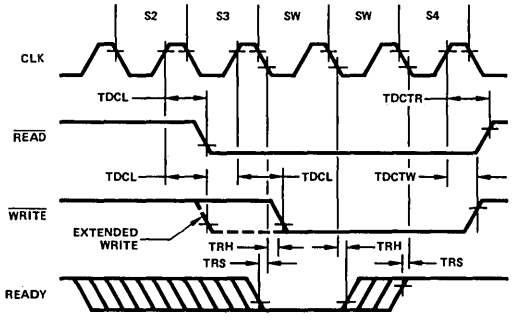
MOS-038

SWITCHING WAVEFORMS (Cont.)



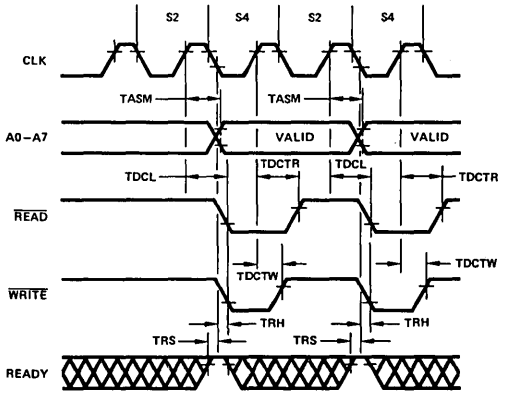
Timing Diagram 4. Memory-to-Memory.

MOS-039



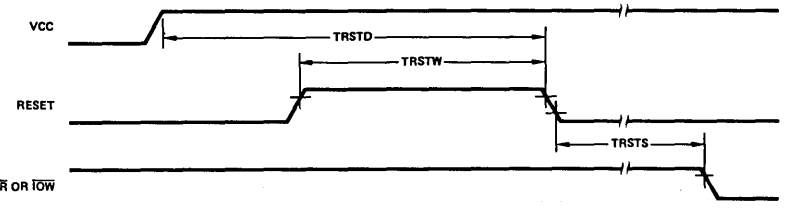
Timing Diagram 5. Ready Timing.

MOS-040



Timing Diagram 6. Compressed Timing.

MOS-041



Timing Diagram 7. Reset Timing.

MOS-042

APPLICATION INFORMATION

Figure 6 shows a convenient method for configuring a DMA system with the Am9517A Controller and a microprocessor system. The Multimode DMA Controller issues a Hold Request to the processor whenever there is at least one valid DMA Request from a peripheral device. When the processor replies with a Hold Acknowledge signal, the Am9517A takes control of the Address Bus, the Data Bus and the Control Bus. The address for the first transfer operation comes out in two bytes – the least significant eight bits on the eight Address outputs and the most significant eight bits on the Data Bus. The contents of the Data Bus are then latched into the Am74LS373 register to complete the full 16 bits of the Address Bus. The Am74LS373 is a high speed, low power, 8-bit, 3-state register in a 20-pin package. After the initial transfer takes place, the register is updated only after a carry or borrow is generated in the least significant address byte. Four DMA channels are provided when one Am9517A is used.

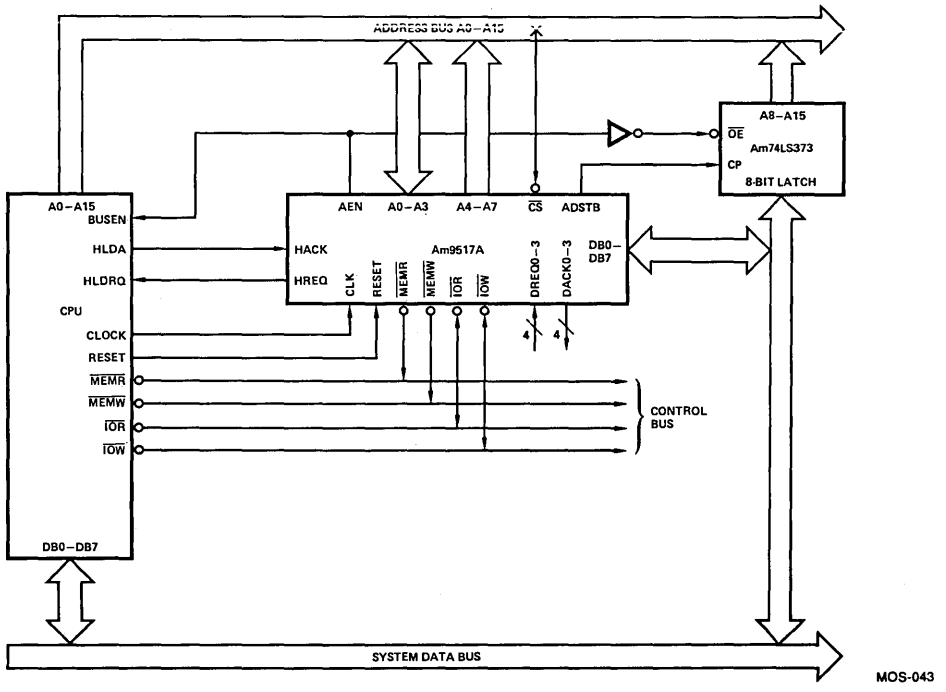
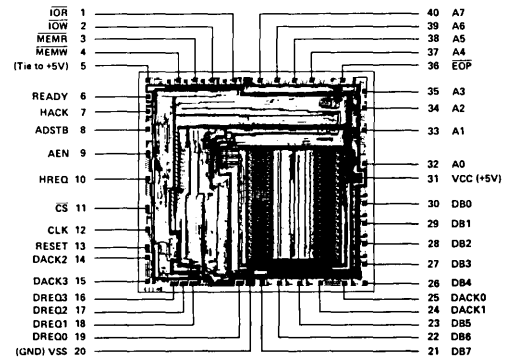


Figure 6. Basic DMA Configuration.

Metallization and Pad Layout



DIE SIZE 0.198" X 0.210"

Am9519

Universal Interrupt Controller

DISTINCTIVE CHARACTERISTICS

- Eight individually maskable interrupt inputs
- Software interrupt request capability
- Fully programmable 1, 2, 3 or 4 byte responses
- Unlimited daisy-chain expansion capability
- Fixed or rotating priority resolution
- Common vector option
- Polled mode option
- Optional automatic clearing of acknowledged interrupts
- Bit set/reset capability for Mask register
- Master Mask bit disables all interrupts
- Pulse-catching interrupt input circuitry
- Polarity control of interrupt inputs and output
- Various timing options including 8085A compatible Am9519-1
- Single +5V supply
- 100% MIL-STD-883 reliability assurance testing

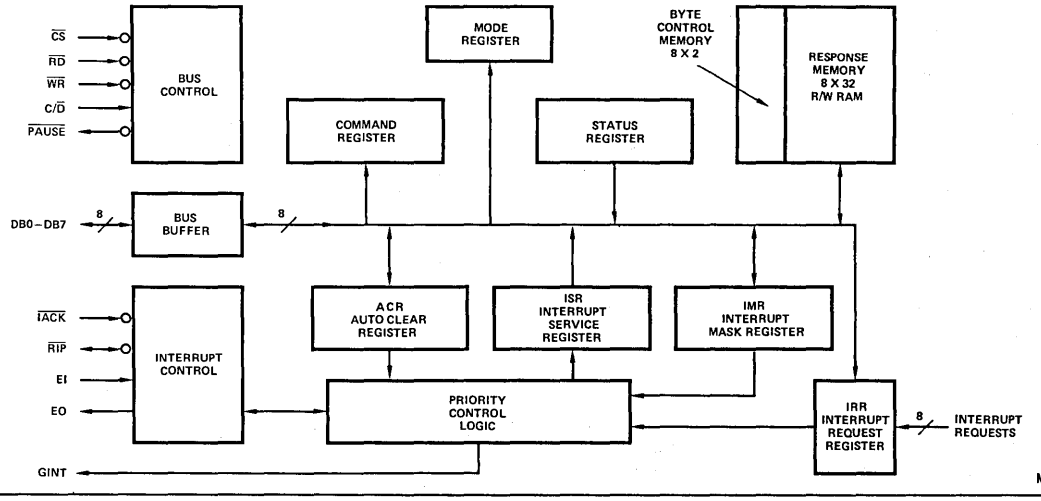
GENERAL DESCRIPTION

The Am9519 Universal Interrupt Controller is a processor support circuit that provides a powerful interrupt structure to increase the efficiency and versatility of microcomputer-based systems. A single Am9519 manages up to eight maskable interrupt request inputs, resolves priorities and supplies up to four bytes of fully programmable response for each interrupt. It uses a simple expansion structure that allows many units to be cascaded for control of large numbers of interrupts. Several programmable control features are provided to enhance system flexibility and optimization.

The Universal Interrupt Controller is designed with a general purpose interface to facilitate its use with a wide range of digital systems, including most popular 8-bit microprocessors. Since the response bytes are fully programmable, any instruction or vectoring protocol appropriate for the host processor may be used.

When the Am9519 controller receives an unmasked Interrupt Request, it issues a Group Interrupt output to the CPU. When the interrupt is acknowledged, the controller outputs the one-to-four byte response associated with the highest priority unmasked interrupt request. The ability of the CPU to set interrupt requests under software control permits hardware prioritization of software tasks and aids system diagnostic and maintenance procedures.

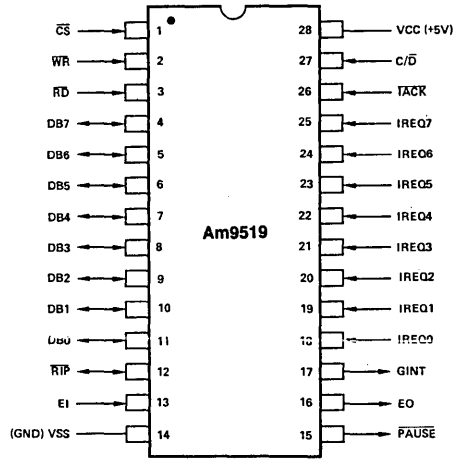
BLOCK DIAGRAM



ORDERING INFORMATION

Package Type	Ambient Temperature	Timing Options	
		Am9519	Am9519-1
Hermetic DIP	$0°C \leq T_A \leq +70°C$	AM9519DC	AM9519-1DC
	$-55°C \leq T_A \leq +125°C$	AM9519DM	
Molded DIP	$0°C \leq T_A \leq +70°C$	AM9519PC	AM9519-1PC

CONNECTION DIAGRAM



Top View

Pin 1 is marked for orientation.

MOS-019

INTERFACE SIGNAL DESCRIPTION

VCC: +5 Volt Power Supply
VSS: Ground

DB0 – DB7 (Data Bus, Input/Output)

The eight bidirectional data bus signals are used to transfer information between the Am9519 and the system data bus. The direction of transfer is controlled by the $\overline{IACK}$, $\overline{WR}$ and $\overline{RD}$ input signals. Programming and control information are written into the device; status and response data are output by it.

$\overline{CS}$ (Chip Select, Input)

The active low Chip Select input enables read and write operations on the data bus. Interrupt acknowledge responses are not conditioned by $\overline{CS}$.

$\overline{RD}$ (Read, Input)

The active low Read signal is conditioned by $\overline{CS}$ and indicates that information is to be transferred from the Am9519 to the data bus.

$\overline{WR}$ (Write, Input)

The active low Write signal is conditioned by $\overline{CS}$ and indicates that data bus information is to be transferred from the data bus to a location within the Am9519.

C/$\overline{D}$ (Control/Data, Input)

The C/$\overline{D}$ control signal selects source and destination locations for data bus read and write operations. Data read or write transfers are made to or from preselected internal registers or memory locations. Control write operations load the command register and control read operations output the status register.

IREQ0 – IREQ7 (Interrupt Request, Input)

The Interrupt Request signals are used by external devices to indicate that service by the host CPU is desired. IREQ inputs are accepted asynchronously and they may be programmed for either a high-to-low or low-to-high edge transition. Active inputs are latched internally in the Interrupt Request Register. After the IRR bit is cleared, an IREQ transition of the programmed polarity must occur to initiate another request.

$\overline{RIP}$ (Response In Process, Input/Output)

Response In Process is a bidirectional signal used when two or more Am9519 circuits are cascaded. It permits multibyte response transfers to be completed without interference from higher priority interrupts. An Am9519 that is responding to an acknowledged interrupt will treat $\overline{RIP}$ as an output and hold it low until the acknowledge response is finished. An Am9519 without an acknowledged interrupt will treat $\overline{RIP}$ as an input and will ignore $\overline{IACK}$ pulses as long as $\overline{RIP}$ is low. The $\overline{RIP}$ output is open drain and requires an external pullup resistor to VCC.

$\overline{IACK}$ (Interrupt Acknowledge, Input)

The active low Interrupt Acknowledge line indicates that the external system is asking for interrupt response information. Depending on the programmed state of the Am9519, it will accept 1, 2, 3 or 4 $\overline{IACK}$ pulses; one response byte is transferred per pulse. The first $\overline{IACK}$ pulse causes selection of the highest priority unmasked pending interrupt request and generates a $\overline{RIP}$ output signal.

$\overline{PAUSE}$ (Pause, Output)

The active-low Pause signal is used to coordinate interrupt responses with data bus and control timing. Pause goes low when the first $\overline{IACK}$ is received and remains low until $\overline{RIP}$ goes low. The external system can use Pause to stretch the acknowledge cycle and allow the control timing to automatically adjust to the actual priority resolution delays in the interrupt system. Second, third and fourth response bytes do not cause Pause to go low. Pause is an open drain output and requires an external pullup resistor to VCC.

EO (Enable Out, Output)

The active high EO signal is used to implement daisy-chained cascading of several Am9519 circuits. EO is connected to the EI input of the next lower priority chip. On receipt of an interrupt acknowledge, each EO will go inactive until it has been determined that no valid interrupt request is pending on that chip. If an active request is present, EO remains low. EO is also held low when the master mask bit is active, thus disabling all lower priority chips.

EI (Enable In, Input)

The active high EI signal is used to implement daisy-chained cascading of several Am9519 circuits. EI is connected to EO of the next higher priority chip. It may also be used as a hardware disable input for the interrupt system. When EI is low $\overline{IACK}$ inputs are ignored. EI is internally pulled up to VCC so that no external pullup is needed when EI is not used.

GINT (Group Interrupt, Output)

The Group Interrupt output signal indicates that at least one unmasked interrupt request is pending. It may be programmed for active high or active low polarity. When active low, the output is open drain and requires an external pull up resistor to VCC.

REGISTER DESCRIPTION

Interrupt Request Register (IRR): The 8-bit IRR is used to store pending interrupt requests. A bit in the IRR is set whenever the corresponding IREQ input goes active. Bits may also be set under program control from the CPU, thus permitting software generated interrupts. IRR bits may be cleared under program control. An IRR bit is automatically cleared when its interrupt is acknowledged. All IRR bits are cleared by a reset function.

Interrupt Service Register (ISR): The 8-bit ISR contains one bit for each IREQ input. It is used to indicate that a pending interrupt has been acknowledged and to mask all lower priority interrupts. When a bit is set by the acknowledge logic in the ISR, the corresponding IRR bit is cleared. If an acknowledged interrupt is not programmed to be automatically cleared, its ISR bit must be cleared by the CPU under program control when it is desired to permit interrupts from lower priority devices. When the interrupt is programmed for automatic clearing, the ISR bit is automatically reset during the acknowledge sequence. All ISR bits are cleared by a reset function.

Interrupt Mask Register (IMR): The 8-bit IMR is used to enable or disable the individual interrupt inputs. The IMR bits correspond to the IREQ inputs and all eight may be loaded, set or cleared in parallel under program control. In addition, individual IMR bits may be set or cleared by the CPU. A reset function will set all eight mask bits, disabling all requests. A mask bit that is set does not disable the IRR, and an IREQ that arrives while a corresponding mask bit is set will cause an interrupt later when the mask bit is cleared. Only unmasked interrupt inputs can generate a Group Interrupt output.

Response Memory: An 8 x 32 read/write response memory is included in the Am9519. It is used to store up to four bytes of response information for each of the eight interrupt request inputs. All bits in the memory are programmable, allowing any desired vector, opcode, instruction or other data to be entered. The Am9519 transfers the interrupt response information for the highest priority unmasked interrupt from the memory to the data bus when the $\overline{IACK}$ input is active.

Auto Clear Register: The 8-bit Auto Clear register contains one bit for each IREQ input and specifies the operating mode for each of the ISR bits. When an auto clear bit is off, the corresponding ISR bit is set when that interrupt is acknowledged and is cleared by software command. When an auto clear bit is on, the corresponding ISR bit is cleared by the hardware before the end of the acknowledge sequence. A reset function clears all auto clear bits.

Status Register: The 8-bit Status register contains information concerning the internal state of the chip. It is especially useful when operating in the polled mode in order to identify interrupting devices. Figure 1 shows the status register bit assignments. The polarity of the GINT bit 7 is not affected by the GINT polarity control (Mode bit 3). The Status register is read by executing a read operation ($\overline{CS}$ = 0, $\overline{RD}$ = 0) with the control location selected (C/$\overline{D}$ = 1).

Mode Register: The 8-bit Mode register controls the operating options of the Am9519. Figure 2 shows the bit assignments for the Mode register. The five low order mode bits (0 through 4) are loaded in parallel by command. Bits 5, 6 and 7 are controlled by separate commands. (See Figure 4.) The Mode register cannot be read out directly to the data bus, but Mode bits 0, 2 and 7 are available as part of the Status register.

Command Register: The 8-bit Command register stores the last command entered. Depending upon the command opcode, it may initiate internal actions or precondition the part for subsequent data bus transfers. The Command register is loaded by executing a write operation ($\overline{WR}$ = 0) with the control location selected (C/$\overline{D}$ = 1), as shown in Figure 3.

Byte Count Register: The length in bytes of the response associated with each interrupt is independently programmed so that different interrupts may have different length responses. The byte count for each response is stored in eight 2-bit Byte Count registers. For a given interrupt the Am9519 will expect to receive a number of $\overline{IACK}$ pulses that equals the corresponding byte count, and will hold $\overline{RIP}$ low until the count is satisfied.

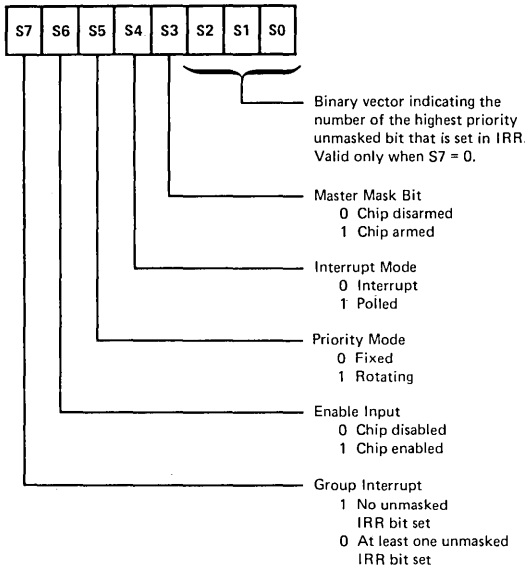


Figure 1. Status Register Bit Assignments. MOS-025

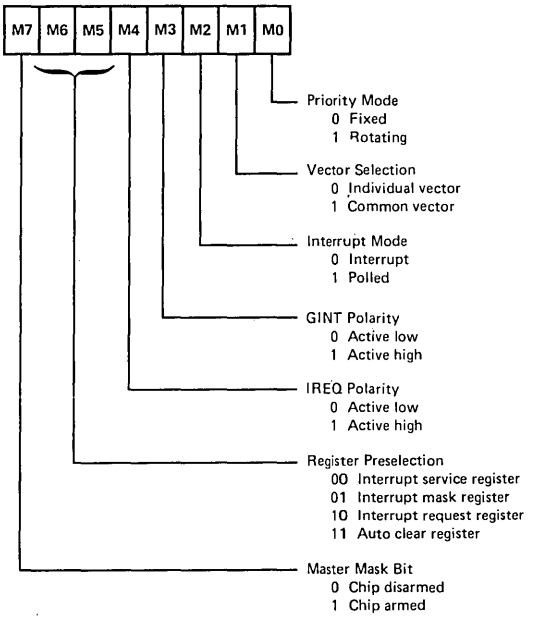


Figure 2. Mode Register Bit Assignments. MOS-026

4

FUNCTIONAL DESCRIPTION

Interrupts are used to improve system throughput and response time by eliminating heavy dependence on software polling procedures. Interrupts allow external devices to asynchronously modify the instruction sequence of a program being executed. In systems with multiple interrupts, vectoring can further improve performance by allowing direct identification of the interrupting device and its associated service routine. The Am9519 Universal Interrupt Controller contains, on one chip, all of the circuitry necessary to detect, prioritize and manage eight vectored interrupts. It includes many options and operating modes that permit the design of sophisticated interrupt systems.

Reset

The reset function is accomplished by software command or automatically during power-up. The reset command may be issued by the CPU at any time. Internal power up circuitry is triggered when VCC reaches a predetermined threshold, causing a brief internal reset pulse. In both cases, the resulting internal state of the machine is that all registers are cleared except the Mask register which is set. Thus no Group Interrupt will be generated and no interrupt requests will be recognized. The response memory and Byte Count registers are not affected by reset. Their contents after power-up are unpredictable and must be established by the host CPU during initialization.

Operating Sequence

A brief description of a typical sequence of events in an operating interrupt system will illustrate the general interactions among the host CPU, the interrupt controller and the interrupting peripheral.

1. The Am9519 controller is initialized by the CPU in order to customize its configuration and operation for the application at hand. Both the controller and the CPU are then enabled to accept interrupts.
2. One (or more) of the interrupt request inputs to the controller becomes active indicating that peripheral equipment is asking for service. The controller asynchronously accepts and latches the request(s).
3. If the request is masked, no further action takes place. If the request is not masked, a Group Interrupt output is generated by the controller.
4. The GINT signal is recognized by the CPU which normally will complete the execution of the current instruction, insert an interrupt acknowledge sequence into its instruction execution stream, and disable its internal interrupt structure. The controller expects to receive one or more $\overline{IACK}$ signals from the CPU during the acknowledge sequence.
5. When the controller receives the $\overline{IACK}$ signal, it brings $\overline{PAUSE}$ low and selects the highest priority unmasked pending request. When selection is complete, the $\overline{RIP}$ output is brought low and the first byte in the response memory associated with the selected request is output on the data bus. $\overline{PAUSE}$ stays low until RIP goes low. $\overline{RIP}$ stays low until the last byte of the response has been transferred.
6. During the acknowledge sequence, the IRR bit corresponding to the selected request is automatically cleared, and the corresponding ISR bit is set. When the ISR bit is set, the Group Interrupt output is disabled until a higher priority request arrives or the ISR bit is cleared. The ISR bit will be cleared by either hardware or software.
7. If a higher priority request arrives while the current request is being serviced, GINT will be output by the controller, but will be recognized and acknowledged only if the CPU has its interrupt input enabled. If acknowledged, the corresponding higher priority ISR bit will be set and the requests nested.

Information Transfers

Figure 3 shows the control signal configurations for all information transfer operations between the Am9519 and the data bus. The following conventions are assumed: $\overline{RD}$ and $\overline{WR}$ active are mutually exclusive; $\overline{RD}$, $\overline{WR}$ and C/$\overline{D}$ have no meaning unless $\overline{CS}$ is low; active $\overline{IACK}$ pulses occur only when $\overline{CS}$ is high.

For reading, the Status register is selected directly by the C/$\overline{D}$ control input. Other internal registers are read by preselecting the desired register with mode bits 5 and 6, and then executing a data read. The response memory can be read only with $\overline{IACK}$ pulses. For writing, the Command register is selected directly by the C/$\overline{D}$ control input. The Mask and Auto Clear registers are loaded following specific commands to that effect. To load each level of the response memory, the response preselect command is issued to select the desired level. An appropriate number of data write operations are then executed to load that level.

CONTROL INPUT					DATA BUS OPERATION
$\overline{CS}$	C/$\overline{D}$	$\overline{RD}$	$\overline{WR}$	$\overline{IACK}$	
0	0	0	1	1	Transfer contents of preselected data register to data bus
0	0	1	0	1	Transfer contents of data bus to preselected data register
0	1	0	1	1	Transfer contents of status register to data bus
0	1	1	0	1	Transfer contents of data bus to command register
1	X	X	X	0	Transfer contents of selected response memory location to data bus
1	X	X	X	1	No information transferred

Figure 3. Summary of Data Bus Transfers.

The Pause output may be used by the host CPU to ensure that proper timing relationships are maintained with the Am9519 when $\overline{IACK}$ is active. The $\overline{IACK}$ pulse width required depends on several variables, including: operating temperature, internal logic delays, number of interrupt controllers chained together, and the priority level of the interrupt being acknowledged. When delays in these variables combine to delay selection of a request following the falling edge of the first $\overline{IACK}$, the Pause output may be used to extend the $\overline{IACK}$ pulse, if necessary. Pause will remain low until a request has been selected, as indicated by the falling edge of $\overline{RIP}$. Typically, the internal interrupt selection process is quite fast, especially for systems with a single Am9519, and Pause will consequently remain low for only a very brief interval and will not cause extension of the $\overline{IACK}$ timing.

Operating Options

The Mode register specifies the various combinations of operating options that may be selected by the CPU. It is cleared by power-up or by a reset command.

Mode bit 0 specifies the rotating/fixed priority mode (see Figure 2). In the fixed mode, priority is assigned to the request inputs based upon their physical location at the chip interface, with IREQ0 the highest and IREQ7 the lowest. In the rotating mode, relative priority is the same as for the fixed mode and the most recently serviced request is assigned the lowest priority. In the fixed mode, a lower priority request might never receive service if enough higher priority requests are active. In the rotating mode, any request will receive service within a maximum of seven other service cycles no matter what pattern the request inputs follow.

Mode bit 1 selects the individual/common vector option. Individual vectoring provides a unique location in the response memory for each interrupt request. The common vector option always supplies the response associated with IREQ0 no matter which request is being acknowledged.

Mode bit 2 specifies interrupt or polled operation. In the polled mode the Group Interrupt output is disabled. The CPU may read the Status register to determine if a request is pending. Since $\overline{IACK}$ pulses are not normally supplied in polled mode, the IRR bit is not automatically cleared, but may be cleared by command. With no $\overline{IACK}$ input the ISR and the response memory are not used. An Am9519 in the polled mode has EI connected to EO so that in multichip interrupt systems the polled chip is functionally removed from the priority hierarchy.

Mode bit 3 specifies the sense of the GINT output. When active high polarity is selected the output is a two-state configuration. For active low polarity, the output is open drain and requires an external pull-up resistor to provide the high logic level. The open drain output allows wired-or configurations with other similar output signals.

Mode bit 4 specifies the sense of the IREQ inputs. When active low polarity is selected, the IRR responds to falling edges on the request inputs. When active high is selected, the IRR responds to rising edges.

Mode bits 5 and 6 specify the register that will be read on subsequent data read operations (C/$\overline{D}$ = 0, $\overline{RD}$ = 0). This preselection remains valid until changed by a reset or a command.

Mode bit 7 is the master mask bit that disables all request inputs. It is used to disable all interrupts without modifying the IMR so that the previous IMR contents are valid when interrupts are re-enabled. When the master mask bit is low, it causes the EO line to remain disabled (low). Thus, for multiple-chip interrupt systems, one master mask bit can disable the whole interrupt structure. Alternatively, portions of the structure may be disabled. The state of the master mask bit is available as bit S3 of the Status register.

Programming

After reset, the Am9519 must be initialized by the CPU in order to perform useful work. At a minimum, the master mask bit and at least one of the IMR bits should be enabled. If vectoring is to be used, the response memory must be loaded; if not, the mode must be changed to a non-vectored configuration. Normally, the first step will be to modify the Mode register and the Auto clear register in order to establish the configuraton desired for the application. Then the response memory and byte count will be loaded for those request levels that will be in use. Finally, the master mask bit and at least portions of the IMR will be enabled to allow interrupt processing to proceed.

Commands

The host CPU configures, changes and inspects the internal condition of the Am9519 using the set of commands shown in Figure 4. An "X" entry in the table indicates a "don't care" state. All commands are entered by directly loading the Command register as shown in Figure 3 ($C/\overline{D}$ = 1, $\overline{WR}$ = 0). Figure 5 shows the coding assignments for the Byte Count registers. A detailed description of each command is contained in the Am9519 Application Note AMPUB071.

BY1	BY0	COUNT
0	0	1
0	1	2
1	0	3
1	1	4

Figure 5. Byte Count Coding.

COMMAND CODE								COMMAND DESCRIPTION
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	Reset
0	0	0	1	0	X	X	X	Clear all IRR and all IMR bits
0	0	0	1	1	B2	B1	B0	Clear IRR and IMR bit specified by B2, B1, B0
0	0	1	0	0	X	X	X	Clear all IMR bits
0	0	1	0	1	B2	B1	B0	Clear IMR bit specified by B2, B1, B0
0	0	1	1	0	X	X	X	Set all IMR bits
0	0	1	1	1	B2	B1	B0	Set IMR bit specified by B2, B1, B0
0	1	0	0	0	X	X	X	Clear all IRR bits
0	1	0	0	1	B2	B1	B0	Clear IRR bit specified by B2, B1, B0
0	1	0	1	0	X	X	X	Set all IRR bits
0	1	0	1	1	B2	B1	B0	Set IRR bit specified by B2, B1, B0
0	1	1	0	X	X	X	X	Clear highest priority ISR bit
0	1	1	1	0	X	X	X	Clear all ISR bits
0	1	1	1	1	B2	B1	B0	Clear ISR bit specified by B2, B1, B0
1	0	0	M4	M3	M2	M1	M0	Load Mode register bits 0–4 with specified pattern
1	0	1	0	M6	M5	0	0	Load Mode register bits 5, 6 with specified pattern
1	0	1	0	M6	M5	0	1	Load Mode register bits 5, 6 and set mode bit 7
1	0	1	0	M6	M5	1	0	Load Mode register bits 5, 6 and clear mode bit 7
1	0	1	1	X	X	X	X	Preselect IMR for subsequent loading from data bus
1	1	0	0	X	X	X	X	Preselect Auto Clear register for subsequent loading from data bus
1	1	1	BY1	BY0	L2	L1	L0	Load BY1, BY0 into byte count register and preselect response memory level specified by L2, L1, L0 for subsequent loading from data bus

Figure 4. Am9519 Command Summary.

MAXIMUM RATINGS above which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.5W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VCC	VSS
Am9519DC/CC Am9519-1DC	$0°C \leq T_A \leq +70°C$	+5.0V ±5%	0V
Am9519DM	$-55°C \leq T_A \leq +125°C$	+5.0V ±10%	0V

ELECTRICAL CHARACTERISTICS Over Operating Range (Note 1)

Parameter	Description	Test Conditions		Min.	Typ.	Max.	Unit
VOH	Output High Voltage (Note 12)	IOH = −200μA		2.4			Volts
		IOH = −100μA (EO only)		2.4			
VOL	Output Low Voltage	IOL = 3.2mA				0.4	Volts
		IOL = 1.0mA (EO only)				0.4	
VIH	Input High Voltage			2.0		VCC	Volts
VIL	Input Low Voltage			−0.5		0.8	Volts
IIX	Input Load Current	$VSS \leq VIN \leq VCC$	EI Input	−60		10	μA
			Other Inputs	−10		10	
IOZ	Output Leakage Current	$VSS \leq VOUT \leq VCC$, Output off		−10		10	μA
ICC	VCC Supply Current	T_A = +25°C			80	125	mA
		T_A = 0°C			100	145	
CO	Output Capacitance	fc = 1.0MHz				15	pF
CI	Input Capacitance	T_A = 25°C				10	
CIO	I/O Capacitance	All pins at 0V				20	

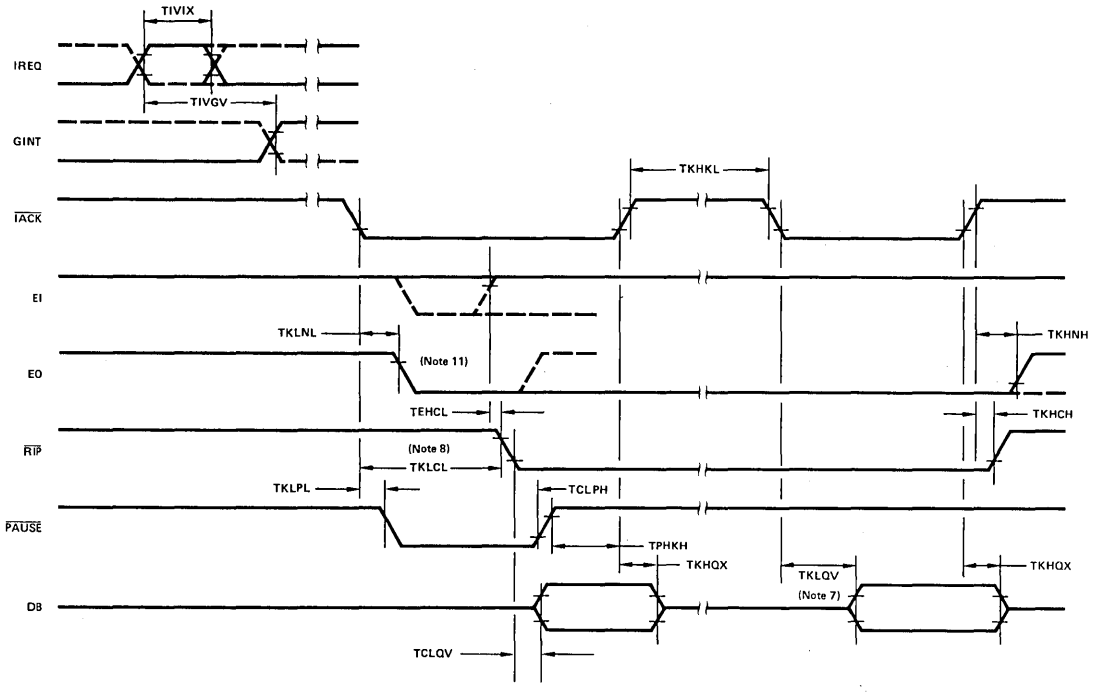
SWITCHING CHARACTERISTICS Over Operating Range (Notes 2, 3, 4, 5)

Parameters	Description	Am9519 Min.	Am9519 Max.	Am9519-1 Min.	Am9519-1 Max.	Units
TAVRL	C/$\overline{D}$ Valid and $\overline{CS}$ LOW to Read LOW	0		0		ns
TAVWL	C/$\overline{D}$ Valid and $\overline{CS}$ LOW to Write LOW	0		0		ns
TCLPH	$\overline{RIP}$ LOW to $\overline{PAUSE}$ HIGH (Note 6)	75	375	75	375	ns
TCLQV	$\overline{RIP}$ LOW to Data Out Valid (Note 7)		50		40	ns
TDVWH	Data In Valid to Write HIGH	250		200		ns
TEHCL	Enable in HIGH to $\overline{RIP}$ LOW (Notes 8, 9)	30	300	30	300	ns
TIVGV	Interrupt Request Valid to Group Interrupt Valid		800		650	ns
TIVIX	Interrupt Request Valid to Interrupt Request Don't Care (IREQ Pulse Duration)	250		250		ns
TKHCH	$\overline{IACK}$ HIGH to $\overline{RIP}$ HIGH (Note 8)		450		350	ns
TKHKL	$\overline{IACK}$ HIGH to $\overline{IACK}$ LOW ($\overline{IACK}$ Recovery)	500		500		ns
TKHNH	$\overline{IACK}$ HIGH to EO HIGH (Notes 10, 11)		975		750	ns
TKHQX	$\overline{IACK}$ HIGH to Data Out Invalid	20	200	20	100	ns
TKLCL	$\overline{IACK}$ LOW to $\overline{RIP}$ LOW (Note 8)	75	600	75	450	ns
TKLNL	$\overline{IACK}$ LOW to EO LOW (Notes 10, 11)		125		100	ns
TKLPL	$\overline{IACK}$ LOW to $\overline{PAUSE}$ LOW	25	175	25	125	ns
TKLQV	$\overline{IACK}$ LOW to Data Out Valid (Note 7)	25	300	25	200	ns
TPHKH	$\overline{PAUSE}$ HIGH to $\overline{IACK}$ HIGH	0		0		ns
TRHAX	Read HIGH to C/$\overline{D}$ and $\overline{CS}$ Don't Care	0		0		ns
TRHQX	Read HIGH to Data Out Invalid	20	200	20	100	ns
TRLQV	Read LOW to Data Out Valid		300		200	ns
TRLQX	Read LOW to Data Out Unknown	50		50		ns
TRLRH	Read LOW to Read HIGH ($\overline{RD}$ Pulse Duration)	300		250		ns
TWHAX	Write HIGH to C/$\overline{D}$ and $\overline{CS}$ Don't Care	0		0		ns
TWHDX	Write HIGH to Data In Don't Care	0		0		ns
TWHRW	Write HIGH to Read or Write LOW (Write Recovery)	600		400		ns
TWLWH	Write LOW to Write HIGH ($\overline{WR}$ Pulse Duration)	300		250		ns

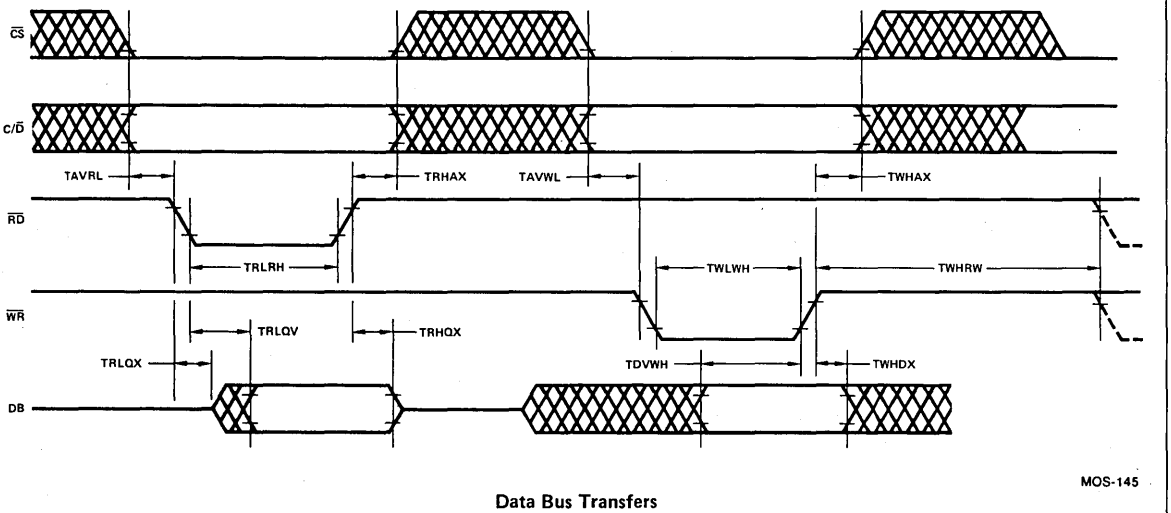
NOTES:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. Test conditions assume transition times of 20ns or less, timing reference levels of 0.8V and 2.0V and output loading of one TTL gate plus 100pF, unless otherwise noted.
3. Transition abbreviations used for the switching parameter symbols include: H = High, L = Low, V = Valid, X = unknown or don't care, Z = high impedance.
4. Signal abbreviations used for the switching parameter symbols include: R = Read, W = Write, Q = Data Out, D = Data In, A = Address ($\overline{CS}$ and C/$\overline{D}$), K = Interrupt Acknowledge, N = Enable Out, E = Enable In, P = Pause, C = $\overline{RIP}$.
5. Switching parameters are listed in alphabetical order.
6. During the first $\overline{IACK}$ pulse, $\overline{PAUSE}$ will be low long enough to allow for priority resolution and will not go high until after $\overline{RIP}$ goes low (TCLPH).
7. TKLQV applies only to second, third and fourth $\overline{IACK}$ pulses while $\overline{RIP}$ is low. During the first $\overline{IACK}$ pulse, Data Out will be valid following the falling edge of $\overline{RIP}$ (TCLQV).
8. $\overline{RIP}$ is pulled low to indicate that an interrupt request has been selected. $\overline{RIP}$ cannot be pulled low until EI is high following an internal delay. TKLCL will govern the falling edge of $\overline{RIP}$ when EI is always high or is high early in the acknowledge cycle. TEHCL will govern when EI goes high later in the cycle. The rising edge of EI will be determined by the length of the preceding priority resolution chain. $\overline{RIP}$ remains low until after the rising edge of the $\overline{IACK}$ pulse that transfers the last response byte for the selected IREQ.
9. Test conditions for the EI line assume timing reference levels of 0.8V and 2.0V with transition times of 10ns or less.
10. Test conditions for the EO line assume output loading of two LS TTL gates plus 30pF and timing reference levels of 0.8V and 2.0V. Since EO normally only drives EI of another Am9519, higher speed operation can be specified with this more realistic test condition.
11. The arrival of $\overline{IACK}$ will cause EO to go low, disabling additional circuits that may be connected to EO. If no valid interrupt is pending, EO will return high when EI is high. If a pending request is selected, EO will stay low until after the last $\overline{IACK}$ pulse for that interrupt is complete and $\overline{RIP}$ goes high.
12. VOH specifications do not apply to $\overline{RIP}$ or to $\overline{GINT}$ when active-low. These outputs are open-drain and VOH levels will be determined by external circuitry.

SWITCHING WAVEFORMS



Interrupt Operations



Data Bus Transfers

APPLICATIONS

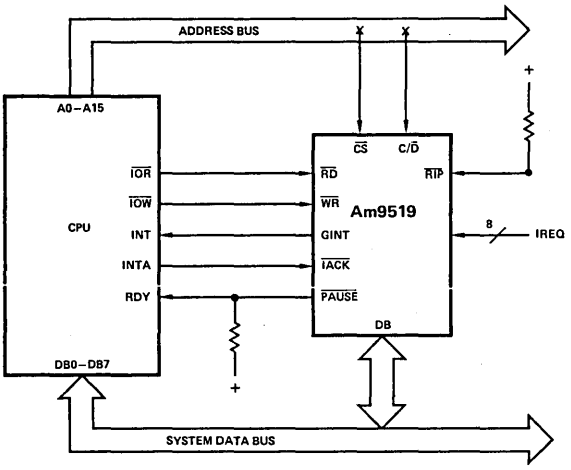


Figure 6. Base Interrupt System Configuration.

MOS-146

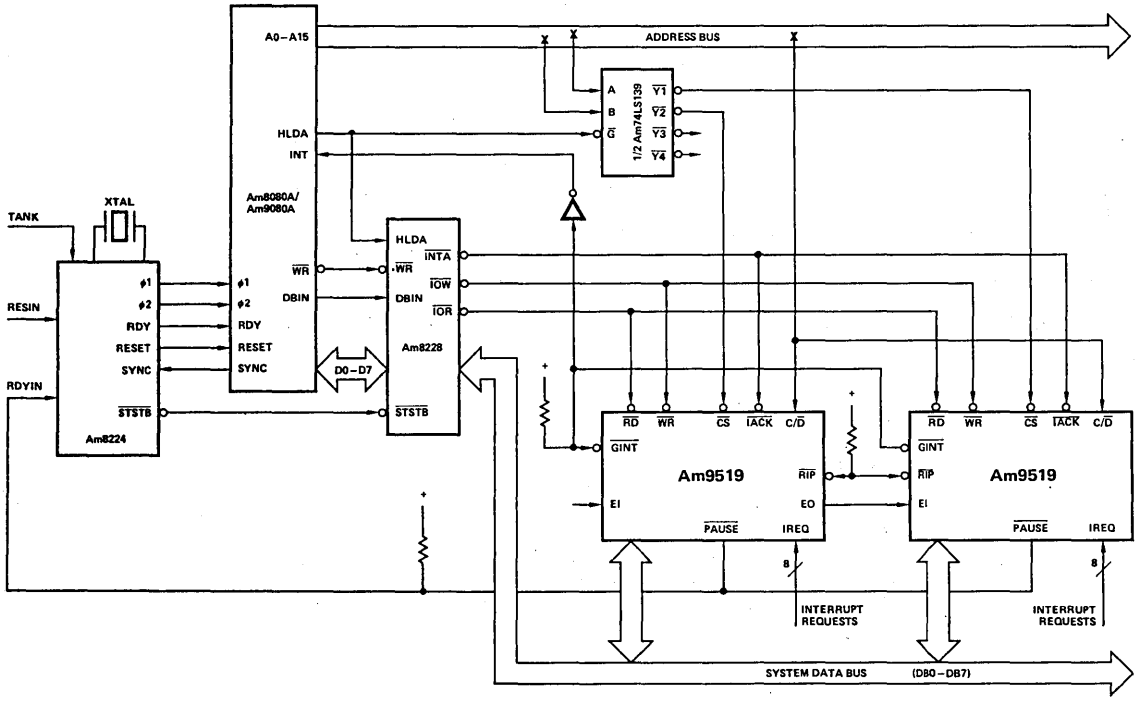


Figure 7. Expanded Interrupt System Configuration.

MOS-147

AmZ8000
Microprocessor
Family

5

CHAPTER 5

Analog Interface Components

Am6012

12-Bit High-Speed Multiplying D/A Converter

Distinctive Characteristics

- All grades 12-bit monotonic over temperature
- Differential nonlinearity to ±.012% (13 bits) max. over temperature
- Trimless design is inherently monotonic
- Fast settling output current: 250nsec
- Full scale current 4mA
- High output impedance and compliance: −5 to +10V
- Differential current outputs
- Low cost
- High-speed multiplying capability
- Direct interface to TTL, CMOS, ECL, HTL, NMOS
- Performance unchanged over supply range
- Low power consumption: 230mW
- R_{OUT}, C_{OUT} independent of logic code

GENERAL DESCRIPTION

The Am6012 series of 12-bit monolithic multiplying Digital to Analog Converters represent a new level of high speed and accuracy coupled with low cost. The Am6012 is the first 12-bit D/A Converter ever built using standard processing without the requirements of thin film resistors and/or active trimming of individual devices. The Am6012 uses sophisticated new circuit design concepts that give inherent monotonicity without requiring ultra precision internal components.

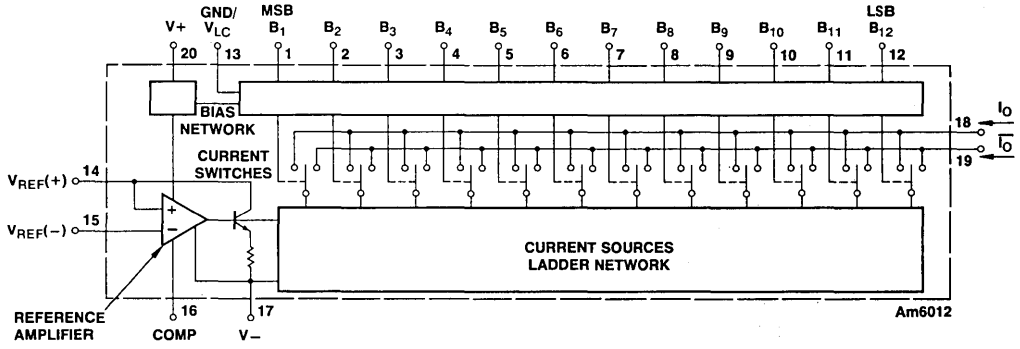
The Am6012 design guarantees a more uniform step size than is possible with standard binarily weighted DAC's. This ±1/2 LSB differential nonlinearity is desirable in many applications where local linearity is critical. The uniform step size allows finer resolution of levels and in most applications is more useful than conformance to an ideal straight line from zero to full scale.

The Am6012 has high voltage compliance, high impedance dual complementary outputs which increase its versatility and enable differential operation to effectively double the peak to peak output swing. These outputs can be used directly without op amps in many applications. The dual complementary outputs can also be connected in A/D converter applications to present a constant load current and significantly reduce switching transients and increase system throughput. Output full scale current is specified at 4mA, allowing use of smaller load resistors to minimize the output RC delay which usually dominates settling time at the 12-bit level.

The Am6012 series guarantees full 12-bit monotonicity for all grades and differential nonlinearity as tight as ±.012% (13 bits) over the entire temperature range. Device performance is essentially independent of power supply voltage. The devices work over a wide operating range of +5, −12 volts to ±18 volts.

5

FUNCTIONAL DIAGRAM

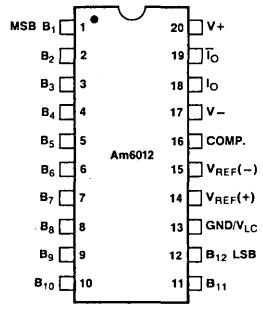


LIC-846

ORDERING INFORMATION

Order Number	Temperature Range	Differential Nonlinearity
AM6012ADM	−55°C to +125°C	±.012%
AM6012DM	−55°C to +125°C	±.025%
AM6012ADC	0°C to +70°C	±.012%
AM6012DC	0°C to +70°C	±.025%

CONNECTION DIAGRAM – Top View



Note:
Pin 1 is marked for orientation.

LIC-847

MAXIMUM RATINGS above which useful life may be impaired

Operating Temperature		Power Supply Voltage	±18V
Am6012ADM, Am6012DM	−55°C to +125°C	Logic Inputs	−5V to +18V
Am6012ADC, Am6012DC	0°C to +70°C	Analog Current Outputs	−8V to +12V
Storage Temperature	−65°C to +125°C	Reference Inputs V_{14}, V_{15}	V− to V+
Lead Temperature (Soldering, 60 sec)	300°C	Reference Input Differential Voltage (V_{14} to V_{15})	±18V
		Reference Input Current (I_{14})	1.25mA

ELECTRICAL CHARACTERISTICS

These specifications apply for V_+ = +15V, V_- = −15V, I_{REF} = 1.0mA, over the operating temperature range unless otherwise specified.

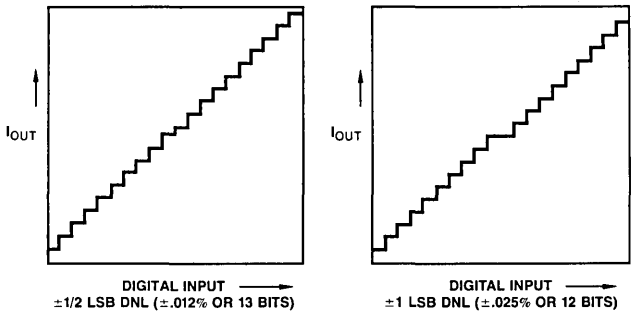
				Am6012A			Am6012			
Parameter	**Description**		**Test Conditions**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Units**
	Resolution			12	12	12	12	12	12	Bits
	Monotonicity			12	12	12	12	12	12	Bits
D.N.L.	Differential Nonlinearity		Deviation from ideal step size	–	–	±.012	–	–	±.025	%FS
				13	–	–	12	–	–	Bits
N.L.	Nonlinearity		Deviation from ideal straight line	–	–	±.05	–	–	±.05	%FS
I_{FS}	Full Scale Current		V_{REF} = 10.000V $R_{14} = R_{15}$ = 10.000kΩ T_A = 25°C	3.967	3.999	4.031	3.935	3.999	4.063	mA
TCI_{FS}	Full Scale Tempco			–	±5	±20	–	±10	±40	ppm/°C
				–	±.0005	±.002		±.001	±.004	%FS/°C
V_{OC}	Output Voltage Compliance		D.N.L. Specification guaranteed over compliance range R_{OUT} > 10 megohms typ.	−5	–	+10	−5	–	+10	Volts
I_{FSS}	Full Scale Symmetry		$I_{FS} - \overline{I_{FS}}$	–	±0.2	±1.0	–	±0.4	±2.0	μA
I_{ZS}	Zero Scale Current			–	–	0.10	–	–	0.10	μA
t_S	Settling Time		To ±1/2 LSB, all bits ON or OFF, T_A = 25°C	–	250	500	–	250	500	nsec
t_{PLH} t_{PHL}	Propagation Delay – all bits		50% to 50%	–	25	50	–	25	50	nsec
C_{OUT}	Output Capacitance			–	20	–	–	20	–	pF
V_{IL}	Logic Input Levels	Logic "0"		–	–	0.8	–	–	0.8	Volts
V_{IH}		Logic "1"		2.0	–	–	2.0	–	–	
I_{IN}	Logic Input Current		V_{IN} = −5 to +18V	–	–	40	–	–	40	μA
V_{IS}	Logic Input Swing		V− = −15V	−5	–	+18	−5	–	+18	Volts
I_{REF}	Reference Current Range			0.2	1.0	1.1	0.2	1.0	1.1	mA
I_{15}	Reference Bias Current			0	−0.5	−2.0	0	−0.5	−2.0	μA
dI/dt	Reference Input Slew Rate		$R_{14(eq)}$ = 800Ω CC = 0pF	4.0	8.0	–	4.0	8.0	–	mA/μs
$PSSI_{FS+}$	Power Supply Sensitivity		V+ = +13.5V to +16.5V, V− = −15V	–	±.00005	±.001	–	±0.0005	±.001	%FS/%
$PSSI_{FS-}$			V− = −13.5V to −16.5V, V+ = +15V	–	±.00025	±.001	–	±.00025	±.001	
V+	Power Supply Range		V_{OUT} = 0V	4.5	–	18	4.5	–	18	Volts
V−				−18	–	−10.8	−18	–	−10.8	
I+	Power Supply Current		V+ = +5V, V− = −15V	–	5.7	8.5	–	5.7	8.5	mA
I−				–	−13.7	−18.0	–	−13.7	−18.0	
I+			V+ = +15V, V− = −15V		5.7	8.5	–	5.7	8.5	
I−				–	−13.7	−18.0	–	−13.7	−18.0	
P_D	Power Dissipation		V+ = +5V, V− = −15V	–	234	312	–	234	312	mW
			V+ = +15V, V− = −15V	–	291	397	–	291	397	

ACCURACY SPECIFICATIONS

The design of the Am6012 emphasizes differential linearity which is a measure of the uniformity of each step in the transfer characteristic. The circuit design, described in greater detail on page 6, requires resistor matching and tracking tolerances of 8 times lower than that of previous designs to achieve and maintain monotonicity over temperature. This advantage has been used in the Am6012A to provide 13-bit differential nonlinearity over temperature, a level of performance not generally available in previous designs, even when using thin film resistors.

The figures illustrate that ±1/2 LSB (13-bit) differential nonlinearity guarantees a converter with 4096 distinct output levels. ±1 LSB D.N.L. guarantees monotonicity, so that when the input code is increased the output will not decrease. Note that nonlinearity, or deviation from an ideal straight line through zero and full scale, cannot be visually determined from the figures. In most applications, 12-bit resolution and differential linearity are more important than linearity. This is especially true in video and graphics, where the human eye has difficulty discerning nonlinearity of less than 5%.

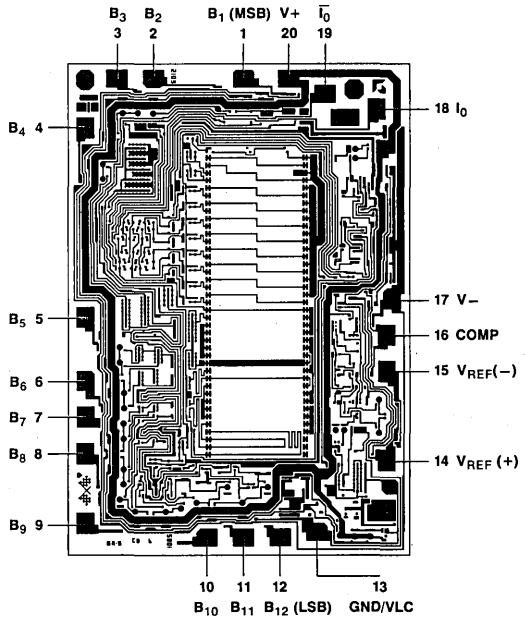
DIFFERENTIAL NONLINEARITY
WORST CASE AT TEMPERATURE EXTREME



LIC-848

5

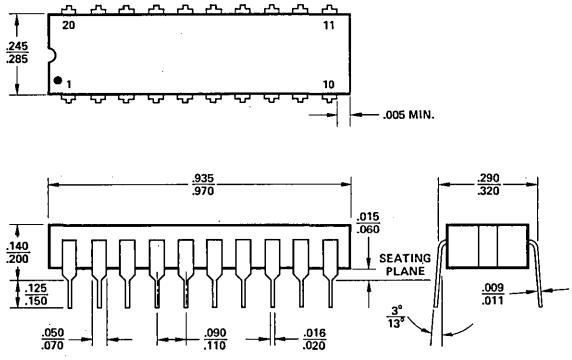
Metallization and Pad Layout



DIE SIZE 0.093" X 0.134"

PHYSICAL DIMENSIONS
Dual-In-Line

20-Pin Hermetic

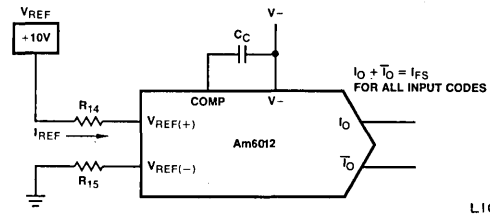


APPLICATION HINTS:

1. Reference current and reference resistor.

There is a 1 to 4 scale factor between the reference current (I_{REF}) and the full scale output current (I_{FS}). If V_{REF} = +10V and I_{FS} = 4mA, the value of the R_{14} is:

$$R_{14} = \frac{4 \times 10\ \text{Volt}}{4\text{mA}} = 10\text{k}\Omega \qquad R_{14} = R_{15}$$



LIC-849

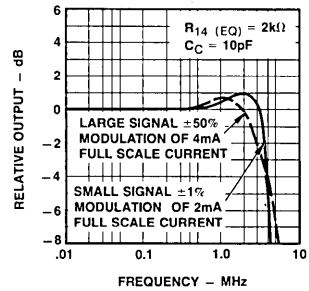
2. Reference amplifier compensation.

For AC reference applications, a minimum value compensation capacitor (C_C) is normally used. The value of this capacitor depends on the equivalent resistance at pin 14. The values to maximize bandwidth without oscillation are as follows:

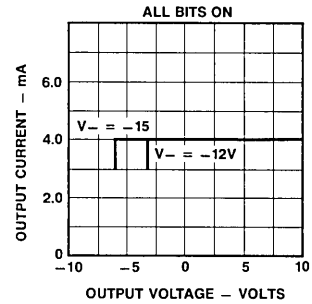
MINIMUM SIZE COMPENSATION CAPACITOR (I_{FS} = 4mA, I_{REF} = 1.0mA)

R_{14} (EQ) (kΩ)	C_C (pF)
10	50
5	25
2	10
1	5
.5	0

Reference Amplifier Frequency Response



Output Voltage Compliance

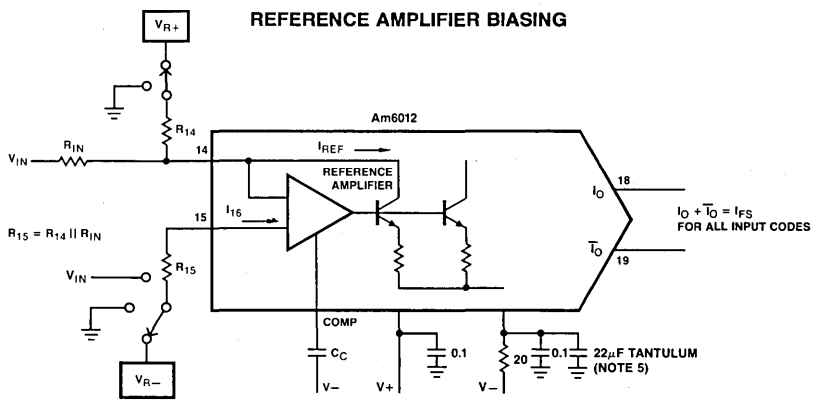


Note: A 0.01μF capacitor is recommended for fixed reference operation.

LIC-850

LIC-851

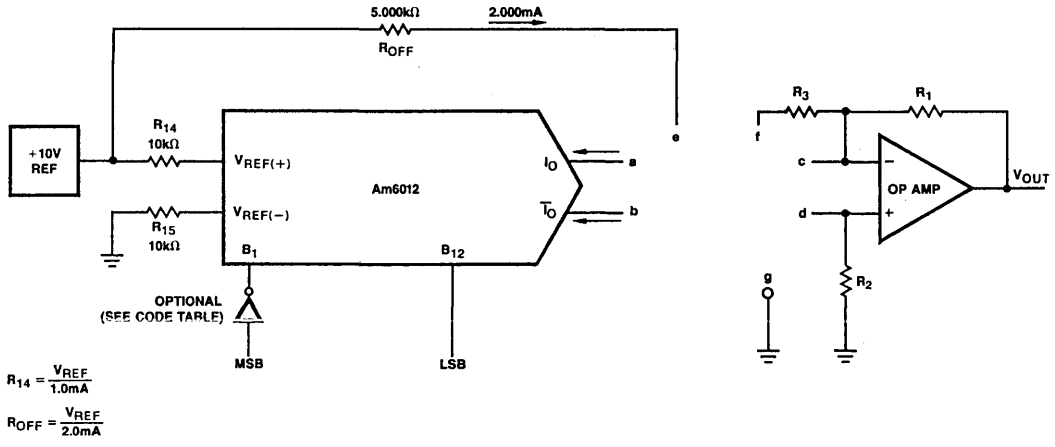
REFERENCE AMPLIFIER BIASING



LIC-852

Reference Configuration	R_{14}	R_{15}	R_{IN}	C_C	I_{REF}
Positive Reference	V_{R+}	0V	N/C	.01μF	V_{R+}/R_{14}
Negative Reference	0V	V_{R-}	N/C	.01μF	$-V_{R-}/R_{14}$
Lo Impedance Bipolar Reference	V_{R+}	0V	V_{IN}	(Note 1)	$(V_{R+}/R_{14}) + (V_{IN}/R_{IN})$ (Note 2)
Hi Impedance Bipolar Reference	V_{R+}	V_{IN}	N/C	(Note 1)	$(V_{R+} - V_{IN})/R_{14}$ (Note 3)
Pulsed Reference (Note 4)	V_{R+}	0V	V_{IN}	No Cap	$(V_{R+}/R_{14}) + (V_{IN}/R_{IN})$

Notes:
1. The compensation capacitor is a function of the impedance seen at the $+V_{REF}$ input and must be at least C = 5pF x $R_{14(eq)}$ in kΩ. For $R_{14} < 800\Omega$ no capacitor is necessary.
2. For negative values of V_{IN}, V_{R+}/R_{14} must be greater than $-V_{IN}$ Max/R_{IN} so that the amplifier is not turned off.
3. For positive values of V_{IN}, V_{R+} must be greater than V_{IN} Max so the amplifier is not turned off.
4. For pulsed operation, V_{R+} provides a DC offset and may be set to zero in some cases. The impedance at pin 14 should be 800Ω or less.
5. For optimum settling time, decouple V– with 20Ω and bypass with 22μF tantulum capacitor.



CODE FORMAT		CONNECTIONS	OUTPUT SCALE	MSB B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	LSB B12	I_O (mA)	$\overline{I_O}$ (mA)	V_{OUT}
UNIPOLAR	Straight binary; one polarity with true input code, true zero output.	a-c b-g R1 = R2 = 2.5K	Positive full scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9976
			Positive full scale − LSB	1	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9951
			Zero scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	.0000
	Complementary binary; one polarity with complementary input code, true zero output.	a-g b-c R1 = R2 = 2.5K	Positive full scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	9.9976
			Positive full scale − LSB	0	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	9.9951
			Zero scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	.0000
SYM-METRICAL OFFSET	Straight offset binary; offset half scale, symmetrical about zero, no true zero output.	a-c b-d f-g R1 = R3 = 2.5K R2 = 1.25K	Positive full scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9976
			Positive full scale − LSB	1	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9927
			(+) Zero scale	1	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.0024
			(−) Zero scale	0	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−.0024
			Negative full scale − LSB	0	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9927
			Negative full scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−9.9976
	1's complement; offset half scale, symmetrical about zero, no true zero output MSB complemented (need inverter at B1).	a-c b-d f-g R1 = R3 = 2.5K R2 = 1.25K	Positive full scale	0	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9976
			Positive full scale − LSB	0	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9927
			(+) Zero scale	0	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.0024
			(−) Zero scale	1	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−.0024
			Negative full scale − LSB	1	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9927
			Negative full scale	1	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−9.9976
OFFSET WITH TRUE ZERO	Offset binary; offset half scale, true zero output.	e-a-c b-g R1 = R2 = 5K	Positive full scale	1	1	1	1	1	1	1	1	1	1	1	1	3.999	.000	9.9951
			Positive full scale − LSB	1	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9902
			+ LSB	1	0	0	0	0	0	0	0	0	0	0	1	2.001	1.998	.0049
			Zero Scale	1	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.000
			− LSB	0	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−.0049
			Negative full scale + LSB	0	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9951
			Negative full scale	0	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−10.000
	2's complement; offset half scale true zero output MSB complemented (need inverter at B1).	e-a-c b-g R1 = R2 = 5K	Positive full scale	0	1	1	1	1	1	1	1	1	1	1	1	3.999	.006	9.9951
			Positive full scale − LSB	0	1	1	1	1	1	1	1	1	1	1	0	3.998	.001	9.9902
			+1 LSB	0	0	0	0	0	0	0	0	0	0	0	1	2.001	1.998	.0049
			Zero scale	0	0	0	0	0	0	0	0	0	0	0	0	2.000	1.999	.000
			−1 LSB	1	1	1	1	1	1	1	1	1	1	1	1	1.999	2.000	−0.049
			Negative full scale + LSB	1	0	0	0	0	0	0	0	0	0	0	1	.001	3.998	−9.9951
			Negative full scale	1	0	0	0	0	0	0	0	0	0	0	0	.000	3.999	−10.000

5

ADDITIONAL CODE MODIFICATIONS

1. Any of the offset binary codes may be complemented by reversing the output terminal pair.

SEGMENTED DAC DESIGN INFORMATION

The design of a 12-bit D/A converter has traditionally required precision thin film resistors, a trimming method, and a binarily weighted ladder network. The Am6012 is a 12-bit DAC which uses diffused resistors and requires no trimming, cutting, blowing, or zapping to guarantee monotonicity for all grades over the temperature range. A proprietary design technique, departing from the traditional R-2R approach used in virtually all high speed high resolution converters, provides inherent monotonicity and differential linearity as high as 13 bits. This guarantees a more uniform step size over the temperature range than available trimmed 12-bit converters. The converter's performance is immune to variations in temperature, time, process, and mechanical stress. The circuit also features differential high compliance current outputs, wide supply range, and a multiplying reference input.

In most converter applications, uniform step size is more important than conformance to an ideal straight line. Most 12-bit converters are used for high resolution rather than high linearity, since few transducers are more linear than ±0.1%. All classic binarily weighted converters require ±1/2 LSB (±.012%) linearity in order to guarantee monotonicity, which requires very tight resistor matching and tracking. This new circuit uses conventional bipolar processing to achieve high differential linearity and monotonicity without requiring correspondingly high linearity, or conformance to an ideal straight line.

One design approach which provides monotonicity without requiring high linearity is the MOS switch-resistor string. This circuit is actually a full complement to a current switched R-2R DAC since it is slower, has a voltage output, and if implemented at the 12-bit level would use 4096 low tolerance resistors rather than a minimum number of high tolerance resistors as in the R-2R network. Its lack of speed and density for 12 bits are its drawbacks.

The technique used in the Am6012 combines the advantages of both the R-2R and 2^nR approaches. It is inherently monotonic, fast, and uses untrimmed resistors which are actually fewer in number than the classic R-2R ladder.

In order to properly describe the new design technique, the standard R-2R ladder approach used in previous 12-bit DAC's will first be discussed. Figure 1 shows the twelve-bit currents which are used in all possible binary combinations to generate 4096 analog output levels. The resistor ladder tolerance is most critical for the major carry, where the 11 least significant bits turn off and the most significant bit turns on. If the MSB is more than 1μA low, or −.05%, the converter will be nonmonotonic. Table 1 shows the maximum tracking error which can be allowed over a 100°C range to maintain monotonicity, which is ±1 LSB D.N.L. Achieving ±1/2 LSB differential nonlinearity is especially difficult since it requires a tracking temperature coefficient of ±1.25 ppm/°C.

Figure 2 shows the transfer characteristic for the new technique, called the segmented DAC. The 4096 output levels are composed of 8 groups of 512 steps each. Each step group is generated by a 9-bit DAC, and each of the segment slopes is determined by one of 8 equal current sources, as shown in Figure 3. The resistors which determine monotonocity are in the 9-bit DAC. The major carry of the 9-bit DAC is repeated in each of the 8 segments, and requires eight times lower initial resistor accuracy and tracking to maintain a given differential nonlinearity over temperature.

The operation of the segmented DAC may be visualized by assuming an input code of all zeroes. The first segment current I_O is divided into 512 levels by the 9-bit multiplying DAC and fed to the output, I_{OUT}. As the input code increases, a new segment current is selected for each 512 counts. The previous segment is fed to output I_{OUT} where the new step group is added to it, thus ensuring monotonicity independent of segment resistor values. All higher order segments feed $\overline{I_{OUT}}$.

At each segment endpoint, monotonicity is assured because no critical resistor tolerances are involved. For example, at the midpoint of the transfer characteristic, as shown in Figure 2, $I_{4,0}$ is actually generated by the same segment resistor as $I_{3,511}$ and has been incremented by the remainder current of the 9-bit DAC.

In the segmented DAC, the precision of the 8 main resistors determines linearity only. The influence of each of these resistors on linearity is four times lower than that of the MSB resistor in an R-2R DAC. Hence, assuming the same resistor tolerances for both, the linearity of the segmented approach would actually be higher than that of an R-2R design.

The step generator or 9-bit DAC is composed of a master and a slave ladder. The slave ladder generates the four least significant bits from the remainder of the master ladder by active current splitting utilizing scaled emitters. This saves ladder resistors and greatly reduces the range of emitter scaling required in the 9-bit DAC. All current switches in the step generator are high speed fully differential switches which are capable of switching low currents at high speed. This allows the use of a binary scaled network all the way to the least significant bit which saves power and simplifies the circuitry.

Diffused resistors have advantages over thin film resistors beyond simple economy and bipolar process compatibility. The resistors are fabricated in single crystal rather than amorphous material which gives them better long term stability and tracking and much higher moisture resistance. They are diffused at 1000°C and so are resistant to changes in value due to thermal and chemical causes. Also, no burn-in is required for stability. The contact resistance between aluminum and silicon is more predictable than between aluminum and an amorphous thin film, and no sandwich metals are required to enhance or protect the contact or limit alloying. The initial match between two diffused resistors is similar to that of thin film since both are defined by photomasks and chemical etching. Since the resistors are not trimmed or altered after fabrication, their tracking and long term characteristics are not degraded.

TABLE 1
RESISTOR SPECIFICATIONS

Ladder Type	No. of Resistors	Initial Matching Required for ±1 LSB DNL (%)	Tracking Required for ±1 LSB DNL (ppm/°C)		Tracking Req'd. for ±1/2 LSB DNL (ppm/°C)
			0 Initial DNL	1/2 LSB Initial DNL	1/4 LSB Initial DNL
Straight R-2R	37	±.05	5	2.5	1.25
Segmented 3 Bits + 9 Bits	24	±.4	40	20	10

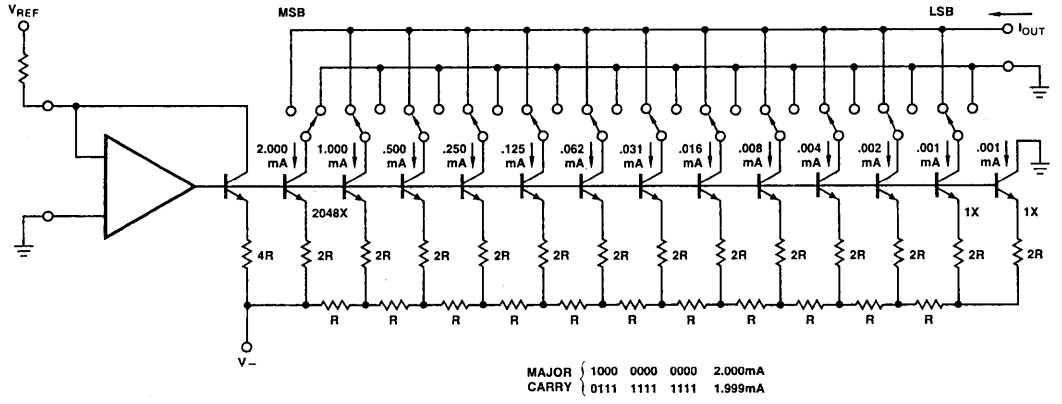


Figure 1. Traditional R-2R D/A Converter.

LIC-854

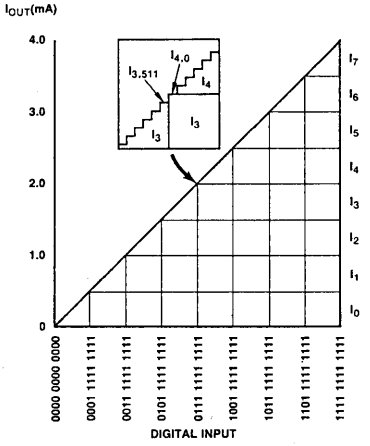


Figure 2. Transfer Characteristic of Segmented Design.

LIC-855

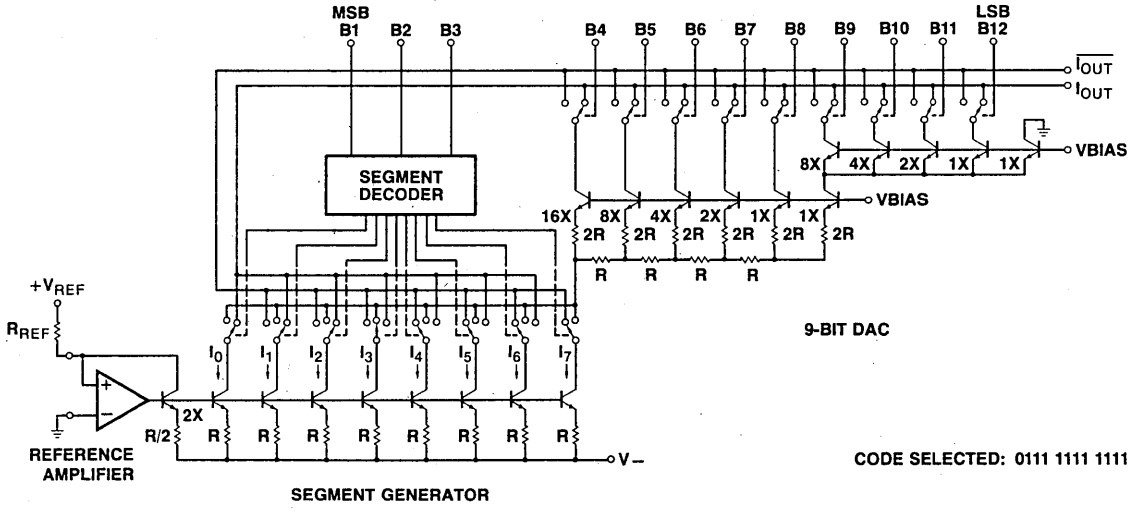
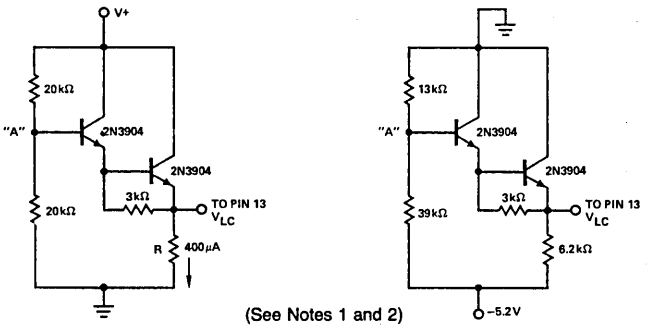


Figure 3. Segmented DAC Functional Diagram Used in Am6012.

LIC-856

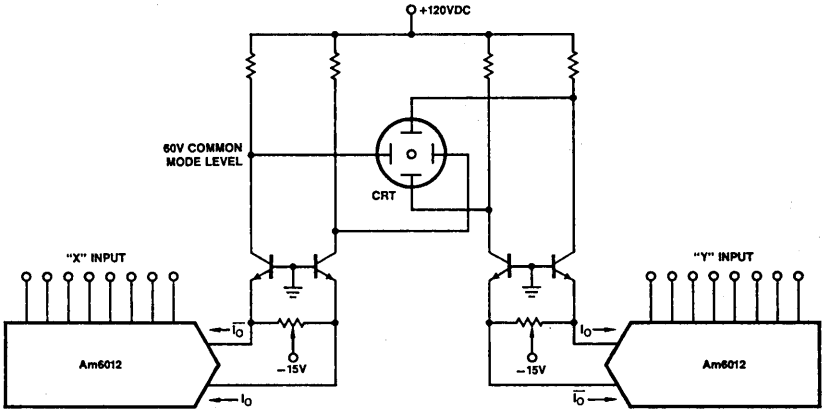


Notes:
1. Set the voltage "A" to the desired logic input switching theshold.
2. Allowable range of logic threshold is typically −5V to +13.5V when operating the DAC on ±15V supplies.

Interfacing Circuits for ECL, CMOS, HTL Logic Inputs

LIC-857

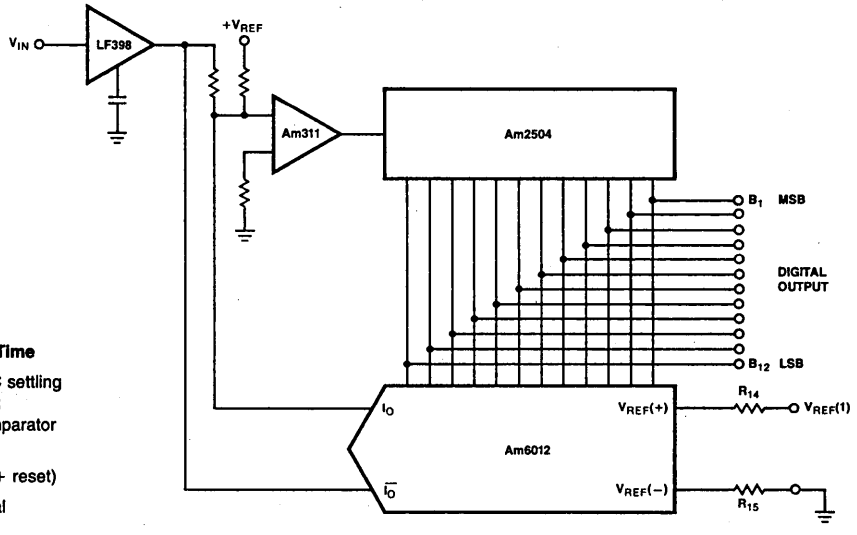
CRT DISPLAY DRIVER



Notes: 1. Full differential drive lowers power supply voltage.
2. Eliminates inverting amplifiers and transformers.
3. Independent beam centering controls.

LIC-858

HIGH-SPEED 12-BIT A/D CONVERTER



Conversion Time

300nsec DAC settling
50nsec SAR
300nsec Comparator
650nsec
x13 (12 bits + reset)
8.5μsec Total

LIC-859

Am6080

Microprocessor System Compatible 8-Bit High Speed Multiplying D/A Converter

DISTINCTIVE CHARACTERISTICS

- 8-Bit D/A with 8-Bit input data latch
- Compatible with most popular microprocessors including the AmZ8000 and the Am2900 Families
- Write, Chip Select and Data Enable logic on chip
- DAC appears as memory location to microprocessor
- MSB inversion under logic control
- Differential current output
- Choice of 6 coding formats
- Fast settling current output −160ns
- Nonlinearity to ±0.1% max over temperature range
- Full scale current pre-matched to ±1 LSB
- High output impedance and voltage compliance
- Low full scale current drift − ±5ppm/°C
- Wide range multiplying capability −2.0MHz bandwidth
- Direct interface to TTL, CMOS, NMOS
- High speed data latch − 80ns min write time

GENERAL DESCRIPTION

The Am6080 is a monolithic 8-bit multiplying Digital-to-Analog converter with an 8-bit data latch, chip select and other control signal lines which allow direct interface with microprocessor buses.

The converter allows a choice of 6 different coding formats. The most significant bit (D_7) can be inverted or non-inverted under the control of the code select input. The code control also provides a zero differential current output for 2's complement coding. A high voltage compliance, complementary current output pair is provided. The data latch is very high speed which makes the Am6080 capable of interfacing with high speed microprocessors.

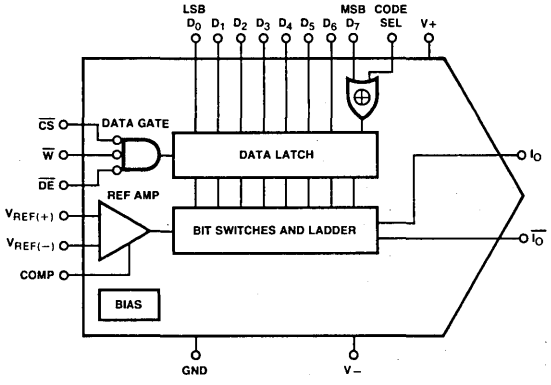
Monotonic multiplying performance is maintained over a more than 40 to 1 reference current range. Matching within ±1 LSB between reference and full scale current eliminates the need for full scale trimming in most applications.

The Am6080 guarantees full 8-bit monotonicity. Nonlinearities as tight as 0.1% over the entire operating temperature range are available. Device performance is essentially unchanged over the full power supply voltage and temperature range.

Applications for the Am6080 include microprocessor compatible data acquisition systems and data distribution systems, 8-bit A/D converters, servo-motor and pen drivers, waveform generators, programmable attenuators, analog meter drivers, programmable power supplies, CRT display drivers and high speed modems.

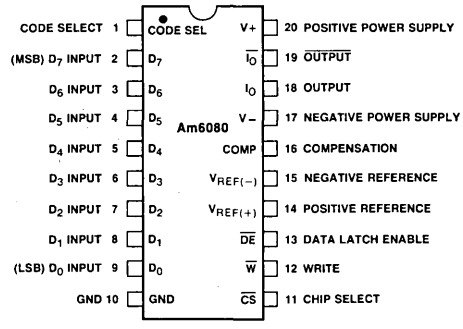
5

EQUIVALENT CIRCUIT



LIC-061

CONNECTION DIAGRAM
Top View



Pin 1 marked for orientation.

LIC-062

MAXIMUM RATINGS

Operating Temperature		Power Supply Voltage	±18V
Am6080ADM, Am6080DM	−55°C to +125°C	Logic Inputs	−5V to +18V
Am6080ADC, Am6080DC Am6080APC, Am6080PC	0°C to +70°C	Analog Current Outputs	−12V to +18V
		Reference Inputs ($V_{14}V_{15}$)	V− to V+
Storage Temperature	−65°C to +150°C	Reference Input Differential Voltage (V_{14} to V_{15})	±18V
Lead Temperature (Soldering, 60 sec)	300°C	Reference Input Current (I_{14})	1.25mA

ELECTRICAL CHARACTERISTICS

These specifications apply for $V_+ = +5V$, $V_- = -15V$, $I_{REF} = 0.5mA$, over the operating temperature range unless otherwise specified. Output characteristics refer to all outputs.

				Am6080A			Am6080			
Parameter	**Description**		**Conditions**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Unit**
	Resolution			8	8	8	8	8	8	bits
	Monotonicity			8	8	8	8	8	8	bits
D.N.L.	Differential Nonlinearity			–	–	±0.19	–	–	±0.39	%FS
N.L.	Nonlinearity			–	–	±0.1	–	–	±0.19	%FS
I_{FS}	Full Scale Current		$V_{REF} = 10.000V$ $R_{14} = R_{15} = 20.000k\Omega$ $T_A = 25°C$	1.984	1.992	2.000	1.976	1.992	2.008	mA
TCI_{FS}	Full Scale Tempco			–	±5	±20	–	±10	±40	ppm/°C
				–	.0005	±.002	–	.001	±.004	%FS/°C
V_{OC}	Output Voltage Compliance			−10	–	+18	−10	–	+18	Volts
I_{FSS}	Full Scale Symmetry		$I_{FS1} - \overline{I_{FS1}}$	–	±0.1	±1.0	–	±0.2	±2.0	μA
I_{ZS}	Zero Scale Current			–	0.01	0.4	–	0.01	0.8	μA
I_{RR}	Reference Current Range		V− = −5V	0	0.5	0.55	0	0.5	0.55	mA
			V− = −15V	0	0.5	1.1	0	0.5	1.1	
V_{IL}	Logic Input Levels	Logic "0"		–	–	0.8	–	–	0.8	Volts
V_{IH}		Logic "1"		2.0	–	–	2.0	–	–	
I_{IN}	Logic Input Current		V_{IN} = −5V to +18V	–	–	40	–	–	40	μA
V_{IS}	Logic Input Swing		V− = −15V	−5	–	+18	−5	–	+18	Volts
I_{15}	Reference Bias Current			–	−0.5	−2.0	–	−0.5	−2.0	μA
dI/dt	Reference Input Slew Rate		$R_{14(EQ)} = 800\Omega$ CC = 0pF	4.0	8.0	–	4.0	8.0	–	mA/μs
$PSSI_{FS+}$	Power Supply Sensitivity		V+ = +4.5V to +5.5V, V− = −15V V− = −13.5V to − 16.5V, V+ = +5V	–	±0.0003	±0.01	–	±0.0005	±0.01	%FS
$PSSI_{FS-}$				–	±0.0005	±0.01	–	±0.0005	±0.01	
V+	Power Supply Range		I_{REF} = 0.5mA, V_{OUT} = 0V	4.5	–	18	4.5	–	18	Volts
V−				−18	–	−4.5	−18	–	−4.5	
I+	Power Supply Current		V+ = +5V, V− = −5V	–	9.8	14.7	–	9.8	14.7	mA
I−				–	−7.4	−9.9	–	−7.4	−9.9	
I+			V+ = +5V, V− = −15V	–	9.8	14.7	–	9.8	14.7	
I−				–	−7.4	−9.9	–	−7.4	−9.9	
I+			V+ = +15V, V− = −15V		9.8	14.7	–	9.8	14.7	
I−				–	−7.4	−9.9	–	−7.4	−9.9	
P_D	Power Dissipation		V+ = +5V, V− = −5V	–	86	123	–	86	123	mW
			V+ = +5V, V− = −15V	–	160	222	–	160	222	
			V+ = +15V, V− = −15V	–	258	369	–	258	369	

Am6081

Microprocessor System Compatible 8-Bit High Speed Multiplying D/A Converter

DISTINCTIVE CHARACTERISTICS

- 8-Bit D/A with 8-Bit input data latch
- Compatible with most popular microprocessors including the AmZ8000 and the Am2900 Families
- Write, Chip Select and Data Enable logic on chip
- DAC appears as memory location to microprocessor
- MSB inversion under logic control
- Differential current outputs
- Output current mode multiplexer with logic selection
- 2-Bit status latch for output select and code select
- Choice of 8 coding formats
- Fast settling current output – 200ns
- Nonlinearity to ±0.1% max over temperature range
- Full scale current pre-matched to ±1 LSB
- High output impedance and voltage compliance
- Low full scale current drift – ±5ppm/°C
- Wide range multiplying capability –2.0MHz bandwidth
- Direct interface to TTL, CMOS, NMOS
- Output range selection with on chip multiplexer
- High speed data latch – 80ns min write time

GENERAL DESCRIPTION

The Am6081 is a monolithic 8-bit multiplying Digital-to-Analog converter with an 8-bit data latch, a 2-bit status latch, chip select and other control signal lines which allow direct interface with microprocessor buses.

The converter allows a choice of 8 different coding formats. The most significant bit (D_7) can be inverted or non-inverted under the control of the code select input. The code control also provides a zero differential current output for 2's complement coding. A pair of high voltage compliance, dual complementary current output channels is provided and is selected by the output status command. The output multiplexer also allows analog bus connection of several converters, range or output load selection, and time-shared operation between D/A and A/D functions. The data and status latches are high speed which makes the Am6081 capable of interfacing with high speed microprocessors. The DE and SE control signals allow the data and status latches to be updated individually or simultaneously.

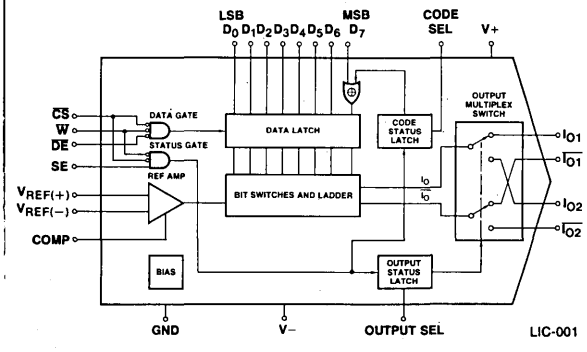
Monotonic multiplying performance is maintained over a more than 40 to 1 reference current range. Matching within ±1 LSB between reference and full scale current eliminates the need for full scale trimming in most applications.

The Am6081 guarantees full 8-bit monotonicity. Nonlinearities as tight as 0.1% over the entire operating temperature range are available. Device performance is essentially unchanged over the full power supply voltage and temperature range.

Applications for the Am6081 include microprocessor compatible data acquisition systems and data distribution systems, 8-bit A/D converters, servo-motor and pen drivers, waveform generators, programmable attenuators, analog meter drivers, programmable power supplies, CRT display drivers and high speed modems.

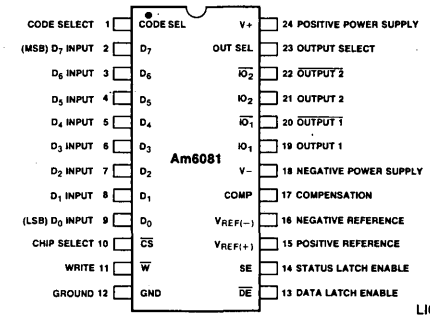
5

EQUIVALENT CIRCUIT



LIC-001

CONNECTION DIAGRAM
Top View



LIC-002

Am6081 FUNCTIONAL PIN DESCRIPTION

Symbol	Function
$\overline{CS}$	Chip Select – This active low input signal enables the Am6081. Writing into the data or status latches occurs only when the device is selected.
$\overline{DE}$	Data Latch Enable – This active low input is used to enable the data latch. The $\overline{CS}$, $\overline{DE}$, and $\overline{W}$ must be active in order to write into the data latch.
SE	Status Latch Enable – This active high input is used to enable the status latches. The $\overline{CS}$, SE, and $\overline{W}$ must be active in order to write into the status latches.
$\overline{W}$	Write – This active low control signal enables the data and status latches when the $\overline{CS}$, $\overline{DE}$, and SE inputs are active.
D_0-D_7	D_0-D_7 are the input bits 1-8 to the input data latch. Data is transferred to the data latch when $\overline{CS}$, $\overline{DE}$, and $\overline{W}$ are active and is latched when any of the enable signals go inactive.
CODE SEL	Code Select – Input to the CODE SEL latch. The latch is transparent when $\overline{CS}$, SE and $\overline{W}$ are active and is latched when any of the above signals go inactive. When CODE SEL latch = 0, the MSB (D_7) is inverted and 1 LSB balance current is added to the $\overline{I_O}$ output.
OUT SEL	Output Select – Input to the OUT SEL latch. The latch is transparent when $\overline{CS}$, SE and $\overline{W}$ are active and is latched when any of the above signals go inactive. When the OUT SEL latch is low, the channel 1 output pair (I_{O1}, $\overline{I_{O1}}$) is selected. When the OUT SEL latch is high, the channel 2 output pair (I_{O2}, $\overline{I_{O2}}$) is selected.
$V_{REF(+)}$ $V_{REF(-)}$	Positive and negative reference voltage to the reference bias amplifier. These differential inputs allow the use of positive, negative and bipolar references.
COMP	Compensation – Frequency compensating terminal for the reference amplifier.
I_{O1}, $\overline{I_{O1}}$ I_{O2}, $\overline{I_{O2}}$	These high impedance current output pairs are selected by the output select latch. I_{O1} and I_{O2} are true outputs and $\overline{I_{O1}}$ and $\overline{I_{O2}}$ are complementary outputs.

FUNCTION TABLES

DATA LATCH CONTROL

$\overline{CS}$	$\overline{W}$	$\overline{DE}$	Data Latch
0	0	0	Transparent
X	X	1	Latched
X	1	X	Latched
1	X	X	Latched

STATUS LATCH CONTROL

$\overline{CS}$	$\overline{W}$	SE	CODE SEL and OUT SEL Latch
0	0	1	Transparent
X	X	0	Latched
X	1	X	Latched
1	X	X	Latched

CODE SELECT AND OUTPUT SELECT

CODE SEL	OUT SEL	Function
0	–	MSB Inverted (Note 1)
1	–	MSB Non-inverted
–	0	Output Channel 1
–	1	Output Channel 2

X = Don't Care

Note 1. 1 LSB balance current is added to the $\overline{I_O}$ output.

MAXIMUM RATINGS

Operating Temperature		Power Supply Voltage	±18V
Am6081ADM, Am6081DM	−55°C to +125°C	Logic Inputs	−5V to +18V
Am6081ADC, Am6081DC Am6081APC, Am6081PC	0°C to +70°C	Analog Current Outputs	−12V to +18V
		Reference Inputs (V_{15}, V_{16})	V− to V+
Storage Temperature	−65°C to +150°C	Reference Input Differential Voltage (V_{15} to V_{16})	±18V
Lead Temperature (Soldering, 60 sec)	300°C	Reference Input Current (I_{15})	1.25mA

GUARANTEED FUNCTIONAL SPECIFICATIONS

Resolution	8 bits
Monotonicity	8 bits

Am6108

Microprocessor System Compatible 8-Bit A/D Converter

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

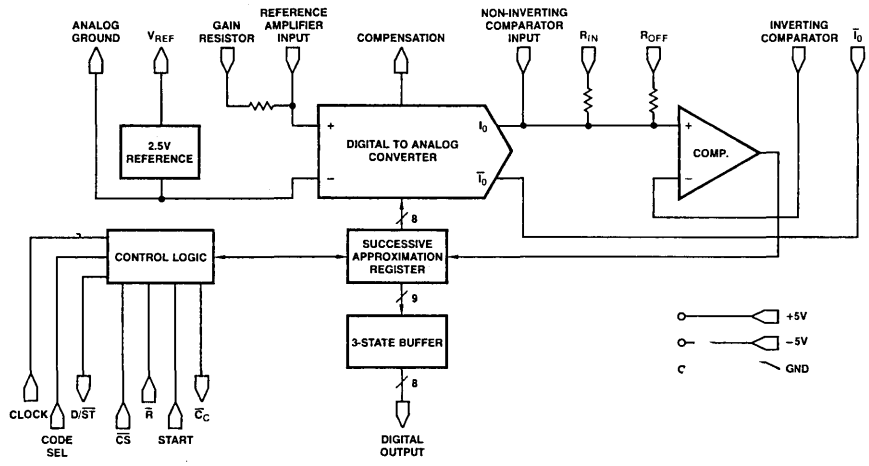
- 1μs conversion time
- Trimmed internal voltage reference
- 0.1% nonlinearity
- Ratiometric operation
- Low operating voltages
- Internal matched gain reference and offset resistors
- Microprocessor compatible
- 3-state outputs
- Pin-programmable unipolar or bipolar twos complement conversion
- Conversion complete available as interrupt or as multiplexed output on data bus

GENERAL DESCRIPTION

The Am6108 is a microprocessor compatible 8-bit high-speed analog-to-digital converter. The Am6108 is the first fully monolithic high-speed A/D to include a precision reference, DAC, comparator, SAR, scale resistors, output 3-state buffers and control logic. The Am6108 is capable of completing an 8-bit conversion in under one microsecond and can handle input voltage ranges of 0 to 10V, 0 to 5V, and ±5V without external components. With appropriate external resistors, the user can program the device to operate on other input signal ranges (2 or 3 precision resistors are required). Full 8-bit performance is guaranteed over temperature. The device has 3-state outputs for bus compatibility and two status outputs – one a standard TTL signal and the other available as a status output on the data bus.

The Am6108 is useful in microprocessor-based systems, or can be used in a stand-alone mode. The conversion time is short enough to allow most microprocessors to accept data immediately after requesting a conversion. Applications include Analog I/O subsystems and servomechanism control.

EQUIVALENT CIRCUIT



960

ORDERING INFORMATION

Order Number	Temperature Range
AM6108DM	−55 to +125°C
AM6108DC	0 to 70°C
AM6108PC	0 to 70°C
AM6108XM	Dice
AM6108XC	Dice

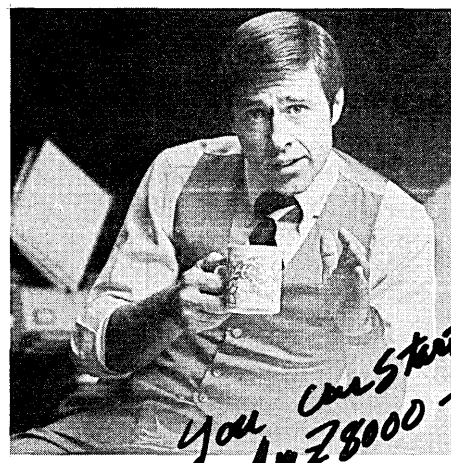
POSITIVE POWER

LSB D

D7 OUTPUT/CONT

CO

AmZ8000 Microprocessor Family



CHAPTER 6

Memories

AMD MOS ROM Selector Guide

AMD P/N	Size	Organization	t_{acc}	Power Supplies	Pins	Operating Range	Package Type
Am3514	4K	512 x 8	700nsec	+5	24	C	Plastic, hermetic
Am9214	4K	512 x 8	500nsec	+5	24	C, M	Plastic, hermetic
Am9208B	8K	1024 x 8	400nsec	+5, +12	24	C, M	Plastic, hermetic
Am9208C	8K	1024 x 8	300nsec	+5, +12	24	C, M	Plastic, hermetic
Am9208D	8K	1024 x 8	250nsec	+5, +12	24	C	Plastic, hermetic
Am9216B	16K	2048 x 8	400nsec	+5, +12	24	C, M	Plastic, hermetic
Am9216C	16K	2048 x 8	300nsec	+5, +12	24	C	Plastic, hermetic
Am9217A	16K	2048 x 8	550nsec	+5	24	C, M	Plastic, hermetic
Am9217B	16K	2048 x 8	450nsec	+5	24	C, M	Plastic, hermetic
Am9218B	16K	2048 x 8	450nsec	+5	24	C, M	Plastic, hermetic
Am9218C	16K	2048 x 8	350nsec	+5	24	C	Plastic, hermetic
Am9232B	32K	4096 x 8	450nsec	+5	24	C, M	Plastic, hermetic
Am9232C	32K	4096 x 8	300nsec	+5	24	C	Plastic, hermetic
Am9233B	32K	4096 x 8	450nsec	+5	24	C, M	Plastic, hermetic
Am9233C	32K	4096 x 8	300nsec	+5	24	C	Plastic, hermetic

Ordering Information

Package Type	Operating Temp. Range	Access Time						
		550ns	500ns	450ns	400ns	350ns	300ns	250ns
Plastic	$0°C \leq T_A \leq +70°C$	Am9217APC P8316A	Am9214PC	Am9217BPC Am9218BPC P8316E Am9232BPC Am9233BPC	Am9208BPC Am9216BPC	Am9218CPC	Am9208CPC Am9216CPC Am9232CPC Am9233CPC	Am9208DPC
Cerdip	$0°C \leq T_A \leq +70°C$	Am9217ACC C8316A	Am9214CC	Am9217BCC Am9218BCC C8316E Am9232BCC Am9233BCC	Am9208BCC Am9216BCC	Am9218CCC	Am9208CCC Am9216CCC Am9232CCC Am9233CCC	Am9208DCC
Side-Brazed	$0°C \leq T_A \leq +70°C$	Am9217ADC	Am9214DC	Am9217BDC Am9218BDC Am9232BDC Am9233BDC	Am9208BDC Am9216BDC	Am9218CDC	Am9208CDC Am9216CDC Am9232CDC Am9233CDC	Am9208DDC
Side-Brazed	$-55°C \leq T_A \leq +125°C$	Am9217ADM	Am9214DM	Am9217BDM Am9218BDM Am9232BDM Am9233BDM	Am9208BDM Am9216BDM		Am9208CDM	

Understanding the AMD ROM P/N

Example: Am 9218 B P C 31500

Basic Device

Speed
A = 500 or 550nsec
B = 400 or 450nsec
C = 300 or 350nsec
D = 250nsec

Package Type
P = Plastic C = Cerdip
D = Ceramic Side-Brazed

Temperature Range
C = 0°C to +70°C
M = − 55°C to +125°C
I = − 25°C to +85°C
L = Special

Customer Identifier
A unique 5 digit code is assigned to each customer pattern by product marketing

AMD ROM Flow

Key Points

1. Be sure to specify all Chip Selects.
2. Be sure to specify device marking.
3. Be sure that the format for the input data is included with the data.
4. Delivery times for both prototypes and production units are dependent on the turnaround time required by the customer for verification.

1 Week after receipt of customer data, a printout in Hexadecimal and/or AMD Hexadecimal IBM cards will be sent for customer verification.

4 Weeks after verification of the printouts, 10 prototypes per code are shipped to the customer for verification.

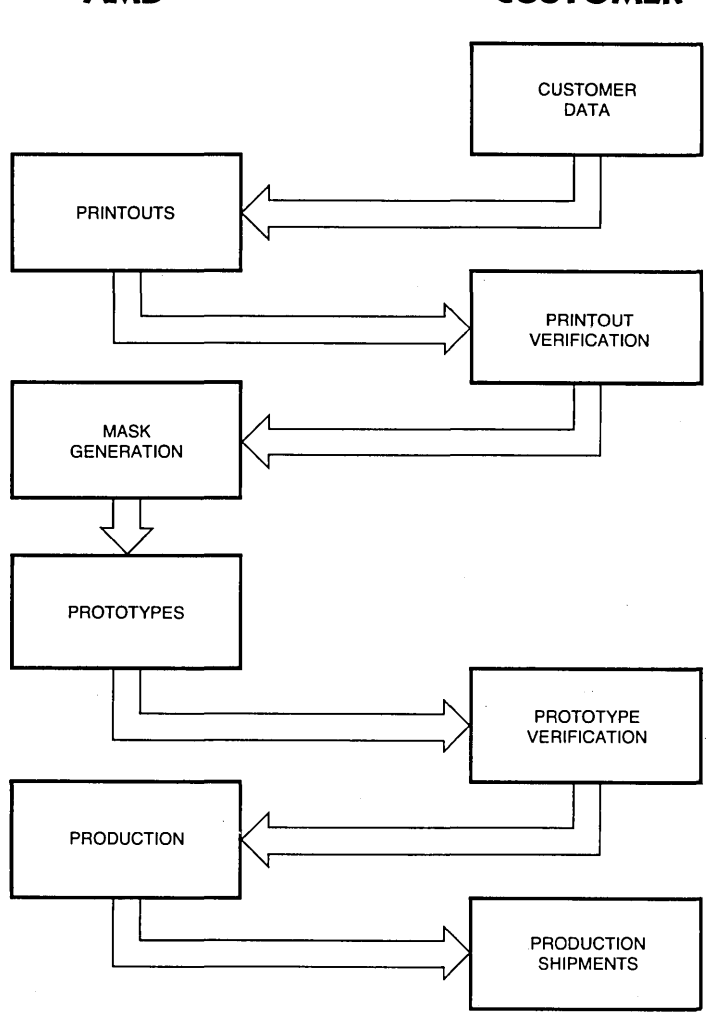
6 Weeks after verification of prototypes, production quantities (up to 1000 pcs. per code) can be shipped.

Preferred Data Format:

AMD HEXADECIMAL

Acceptable Formats:

E-PROMs (2708/2716)
Intel Hexadecimal
Intel BPNF
E-A Octal
G.I. Binary
Motorola Hexadecimal
T.I. Octal



AMD Preferred Format

PROGRAMMING INSTRUCTIONS
Custom Pattern Ordering Information

The Am9208/Am9216/Am9217/C8316A/ Am9218/C8316E/Am9232/Am9233 are programmed from punched cards, card coding forms or from paper tape in card image form in the format as shown below.

Logic "1" = a more positive voltage (normally +5.0V)
Logic "0" = a more negative voltage (normally 0V)

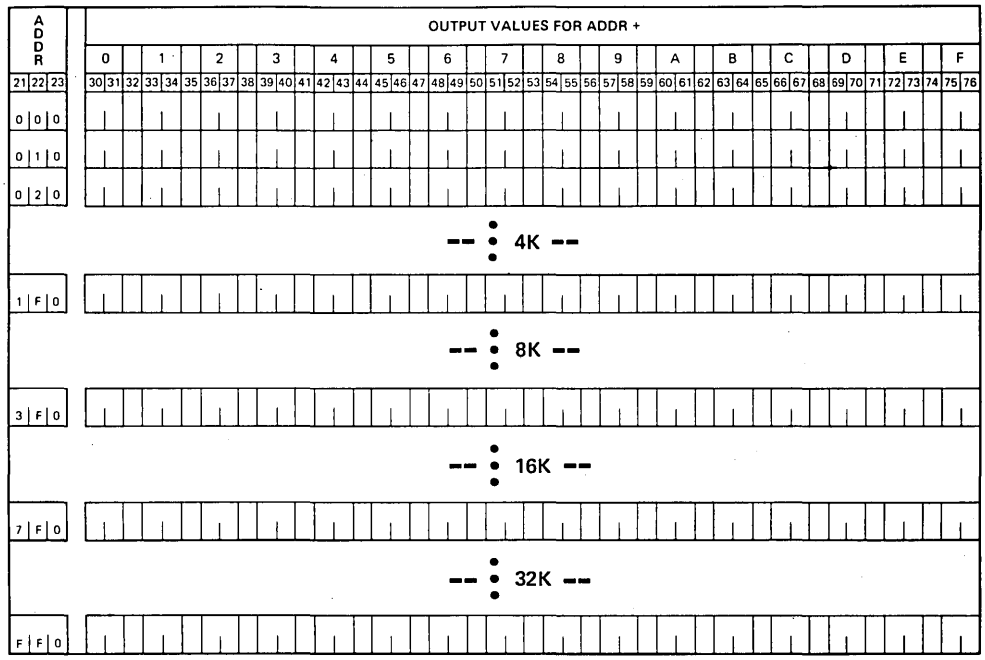
FIRST CARD

Column Number	Description
10 through 29	Customer Name
32 through 37	Total number of "1"s contained in the data. This is optional and should be left blank if not used.
50 through 62	9208B, 9208C, 9208D, 9216B, 9216C, 8316A, 9217A, 9217B, 8316E, 9218B 9218C, 9232B, 9232C, 9233B or 9233C
65 through 72	Data.

SECOND CARD

Column Number	Description
29	CS_3 input required to select chip (0 or 1) only for 9217, 8316A, 9218, 8316E
31	CS_2 input required to select chip (0 or 1)
33	CS_1 input required to select chip (0 or 1)
35	CS_0 input required to select chip (0 or 1) only for 9214

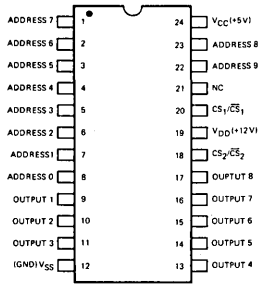
The Hexadecimal Option is a compact way of presenting the data. This format requires only 32/64/128/256 data cards. Each data card contains the 8-bit output information for 16 storage locations in the memory. The address indicated in columns 21, 22 and 23 is the address of the data presented in columns 30 and 31. Addresses for successive data are assumed to be in incremental ascending order from the initial address. Since the address in columns 21, 22 and 23 indexes the first data on the card, column 23 is always a zero. Columns 21 and 22 take all hex values from 00 through 1F/3F/7F/FF cards in all. Data is entered in hex values and may be any combination of 8 bits, that is, hex values from 00 through FF.



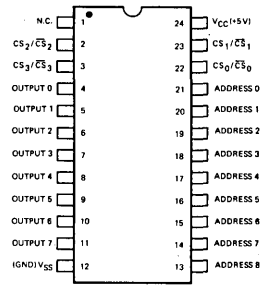
6

Connection Diagrams (Top Views)

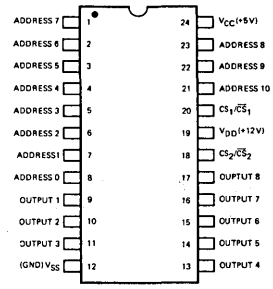
Am9208



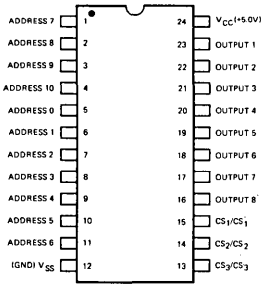
Am9214



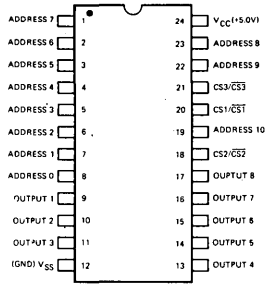
Am9216



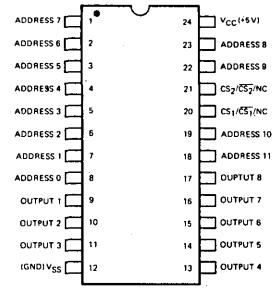
Am9217 8316A



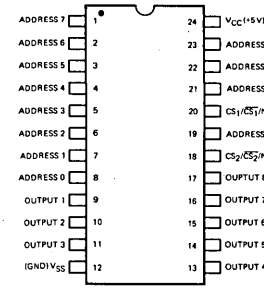
Am9218 8316E



Am9232



Am9233



Bipolar Memory Cross Reference Guide

PROMs COMMON GENERIC SERIES CHARACTERISTICS

- High speed
- Temperature and voltage compensated for excellent military performance
- High reliability fuse technology – platinum-silicide
- Ultra-fast programming
- High programming yields
- Over 4 billion life test hours – no fuse failures
- Low current PNP inputs
- Access time tested with N^2 patterns

PROMs CROSS REFERENCE GUIDE

AMD Part No.	Size	Organization	Output	Package Pins	TAA COML/MIL	Fairchild	Harris	Intel	Interail	Monolithic Memories	National	Signetics	TI
Am27LS18 (Note 2)	256	32 x 8	OC	16	50/65					53/63LS080		N/S82S23	
Am27LS19 (Note 2)	256	32 x 8	3S	16	50/65					53/63LS081		N/S82S123	
Am27S18	256	32 x 8	OC	16	40/50		HM7602		IM5600	53/6330-1	DM75/8577 DM54/74S188	N/S82S23	SN54/74188A SN54/74S188
Am27S19	256	32 x 8	3S	16	40/50		HM7603		IM5610	53/6331-1	DM75/8578 DM54/74S288	N/S82S123	SN54/74S288
Am27S20	1024	256 x 4	OC	16	45/60	93417	HM7610	3601	IM5603A IM56S03	53/6300-1	DM54/74S387	N/S82S126	SN54/74S387
Am27S21	1024	256 x 4	3S	16	45/60	93427	HM7611	3621	IM5623 IM56S23	53/6301-1	DM54/74S287	N/S82S129	SN54/74S287
Am27S12	2048	512 x 4	OC	16	50/60	93436	HM7620	3602	IM5604 IM56S04	53/6305-1	DM54/74S570	N/S82S130	
Am27S13	2048	512 x 4	3S	16	50/60	93446	HM7621	3622	IM5624 IM56S24	53/6306-1	DM54/74S571	N/S82S131	
Am27S15	4096	512 x 8	3S	24	60/90		HM7647R					N/S82S115	
Am27S25* (Note 3)	4096	512 x 8	3S	24	N.A. (Note 4)								
Am27S26	4096	512 x 8	OC	22	N.A. (Note 4)								
Am27S27	4096	512 x 8	3S	22	N.A. (Note 4)								
Am27S28	4096	512 x 8	OC	20	55/70		HM7648			53/6348	DM54/74S473	N/S82S146	SN54/74S473
Am27S29	4096	512 x 8	3S	20	55/70		HM7649			53/6349	DM54/74S472	N/S82S147	SN54/74S472
Am27S30	4096	512 x 8	OC	24	55/70	93438	HM7640	3604	IM5605	53/6340	DM77/87S475	N/S82S140	SN54/74S475
Am27S31	4096	512 x 8	3S	24	55/70	93448	HM7641	3624	IM5625	53/6341	DM77/87S474	N/S82S141	SN54/74S474
Am27S32	4096	1024 x 4	OC	18	55/70	93452	HM7642	3605	IM5606	53/6352	DM54/74S572	N/S82S136	SN54/74S477
Am27S33	4096	1024 x 4	3S	18	55/70	93453	HM7643	3625	IM5626	53/6353	DM54/74S573	N/S82S137	SN54/74S476
Am27S180	8192	1024 x 8	OC	24	60/80	93450	HM7680	3608		53/6380	DM77/87S180	N/S82S180	SN54/74S479
Am27S181	8192	1024 x 8	3S	24	60/80	93451	HM7681	3628		53/6381	DM77/87S181	N/S82S181	SN54/74S478
Am27S184**	8192	2048 x 4	OC	18			HM7684			53/63100	DM77/87S184	N/S82S184	
Am27S185**	8192	2048 x 4	3S	18			HM7685			53/63101	DM77/87S185	N/S82S185	

*Available 4th Qtr. 1979
**Available 1st Qtr. 1980

Notes: 1. COM'L = 0 to 75°C, V_{CC} = 5V ±5%
MIL = −55 to +125°C, V_{CC} = 5V ±10%
2. Replaces Am27LS08/09.
3. Slimline 24-pin package – 300 mil lateral centers.
4. Normal access time not applicable – this product contains built in pipeline registers – nominal address to clock set up time 40ns, clock to output 15ns.

6

PROMs COMMON GENERIC SERIES PROGRAMMING INFORMATION

All AMD Bipolar PROMs must be programmed on AMD approved equipment only.

Approved manufacturers are:
Data I/O Corp.
Pro-Log Corp.

Card Set #
909-1286-1
PM9058

ADAPTORS: AMD P/N	DATA I/O	PRO-LOG	AMD P/N	DATA I/O	PRO-LOG
AM27LS18/19	715-1407-1	PA 16-2 and 32 X 8 (L)	AM27S26/27	715-1412-2	PA 22-4 and 512 X 8 (L)
AM27S18/19	715-1407-1	PA 16-6 and 32 X 8 (L)	AM27S28/29	715-1413	PA 20-4 and 512 X 8 (L)
AM27S20/21	715-1408-1	PA 16-5 and 256 X 4 (L)	AM27S30/31	715-1545	PA 24-13 and 512 X 8 (L)
AM27S12/13	715-1408-2	PA 16-5 and 512 X 4 (L)	AM27S32/33	715-1414	PA 18-6 and 1024 X 4 (L)
AM27S15	715-1411-1	PA 24-14 and 512 X 8 (L)	AM27S180/181	715-1545-2	PA 24-13 and 1024 X 8 (L)
AM27S25	715-1617	PA 24-16 and 512 X 8 (L)	AM27S184/185	715-1616	PA 18-8 and 2048 X 4 (L)

RAMs COMMON CHARACTERISTICS

- High speed
- Low power
- Temperature and voltage compensated for excellent military performance
- Internal ECL circuitry for optimum speed/power performance
- Functional and switching characteristics tested for all data and address patterns

RAMs CROSS REFERENCE GUIDE

AMD Part No.	Description	Organization	Output	TAA COM'L/MIL	Power Max (mW) COM'L/MIL	Fairchild	Intel	Motorola	M.M.I.	National	Signetics	TI
Am27S02A/03A	Ultra-High Speed	16 x 4	OC/3S	25/30	525/580							
Am27S02 Am74/54S289 Am3101A	High Speed	16 x 4	OC	35/50	550/580	74S289	3101A		65/5560	DM74/54S289	N/S82S25 N/S74/54S289 N/S3101A	SN74/54S289
Am27S03 Am74/54S189	High Speed	16 x 4	3S	35/50	550/580	74S189			65/5561	DM85/7599	N/S74/54S189	SN74/54S189
Am27S06/07	Ultra-High Speed Non-Inverting Output	16 x 4	OC/3S	25/30	525/580							
Am3101-1 Am74/5489-1		16 x 4	OC	60/75	525/580	7489	3101			DM74/5489		SN74/5489
Am27LS02	Low Power High Speed	16 x 4	OC	55/65	185/210				L65/5560	DM74/54LS289		
Am27LS03	Low Power High Speed	16 x 4	3S	55/65	185/210				L65/5561	DM74/54LS189 DM86L99		
Am27LS06/07	Low Power High Speed Non-Inverting Output	16 x 4	OC/3S	55/65	185/210							
Am31L01A	Low Power Transparent	16 x 4	OC	55/65	185/210							
Am27LS00A/01A	Ultra-High Speed	256 x 1	3S/OC	35/45	525/550							
Am27LS00	Low Power High Speed	256 x 1	3S	45/55	370/385	93L420 93421 93L421	3106		65/5531	DM74/54S200	N/S82S116 N/S82S16 N/S74/54S200 N/S74/54S201	SN74/54S201 SN74/54S200 SN74/54LS200
Am27LS01	Low Power High Speed	256 x 1	OC	45/55	370/385	93411 93L411	3107		65/5530	DM74/54S206	N/S82S117 N/S82S17 N/S74/54S301	SN74/54S301 SN74/54S300 SN74/54LS300
Am93415*	High Speed	1K x 1	OC	45/60	814/935	93415		MCM93415			N/S82S10	SN74/54S314
Am93425*	High Speed	1K x 1	3S	45/60	814/935	93425		MCM93425			N/S82S11	SN74/54S214
Am93412**	High Speed	256 x 4	OC	45/60	814/935	93412						
Am93422**	High Speed	256 x 4	3S	45/60	814/935	93422						

*Available 4th Qtr. 1979 **Available 1st Qtr. 1980

Am9016

16,384 x 1 Dynamic R/W Random Access Memory

DISTINCTIVE CHARACTERISTICS

- High density 16k x 1 organization
- Direct replacement for MK4116
- Low maximum power dissipation — 462mW active, 20mW standby
- High speed operation — 150ns access, 320ns cycle
- ±10% tolerance on standard +12, +5, −5 voltages
- TTL compatible interface signals
- Three-state output
- RAS only, RMW and Page mode clocking options
- 128 cycle refreshing
- Unlatched data output
- Standard 16-pin, .3 inch wide dual in-line package
- Double poly N-channel silicon gate MOS technology
- 100% MIL-STD-883 reliability assurance testing

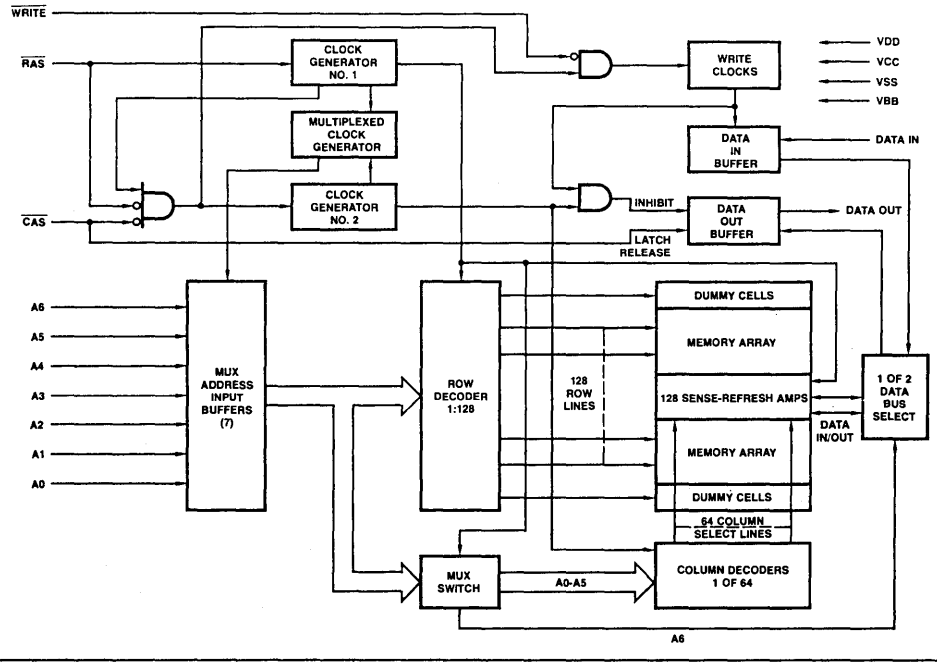
GENERAL DESCRIPTION

The Am9016 is a high speed, 16k-bit, dynamic, read/write random access memory. It is organized as 16,384 words by 1 bit per word and is packaged in a standard 16-pin DIP. The basic memory element is a single transistor cell that stores charge on a small capacitor. This mechanism requires periodic refreshing of the memory cells to maintain stored information.

All input signals, including the two clocks, are TTL compatible. The Row Address Strobe ($\overline{RAS}$) loads the row address and the Column Address Strobe ($\overline{CAS}$) loads the column address. The row and column address signals share 7 input lines. Active cycles are initiated when $\overline{RAS}$ goes low, and standby mode is entered when $\overline{RAS}$ goes high. In addition to normal read and write cycles, other types of operations are available to improve versatility, performance, and power dissipation.

The three-state output buffer turns on when the column access time has elapsed and turns off after $\overline{CAS}$ goes high. Input and output data are the same polarity.

BLOCK DIAGRAM

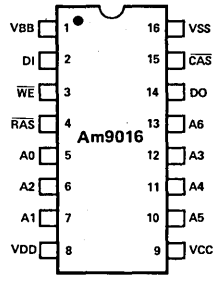


MOS-190

ORDERING INFORMATION

Ambient Temperature	Package Type	Access Time			
		300ns	250ns	200ns	150ns
0°C ≤ T_A ≤ +70°C	Hermetic DIP	AM9016CDC	AM9016DDC	AM9016EDC	AM9016FDC
	Molded DIP	AM9016CPC	AM9016DPC	AM9016EPC	AM9016FPC

CONNECTION DIAGRAM



Top View

Pin 1 is marked for orientation.

A0 – A6	ADDRESS INPUTS
$\overline{CAS}$	COLUMN ADDRESS STROBE
DI	DATA IN
DO	DATA OUT
$\overline{RAS}$	ROW ADDRESS STROBE
VDD	POWER (+12V)
VCC	POWER (+5V)
VSS	GROUND
VBB	POWER (–5V)
$\overline{WE}$	WRITE ENABLE

MOS-191

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	–65°C to +150°C
Ambient Temperature Under Bias	0°C to +70°C
Voltage on Any Pin Relative to VBB	–0.5V to +20V
VDD and VCC Supply Voltages with Respect to VSS	–1.0V to +15.0V
VBB – VSS (VDD – VSS > 0V)	0V
Power Dissipation	1.0W
Short Circuit Output Current	50mA

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulation of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling, and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Ambient Temperature	VDD	VCC	VSS	VBB
$0°C \leq T_A \leq +70°C$	+12V ±10%	+5V ±10%	0	–5.0V ±10%

ELECTRICAL CHARACTERISTICS over operating range (Notes 1, 11)

				Am9016X			
Parameters	Description		Test Conditions	Min.	Typ.	Max.	Units
VOH	Output HIGH Voltage		IOH = –5.0mA	2.4		VCC	Volts
VOL	Output LOW Voltage		IOL = 4.2mA	VSS		0.40	Volts
VIH	Input HIGH Voltage for Address, Data In			2.4		7.0	Volts
VIHC	Input HIGH Voltage for $\overline{CAS}$, $\overline{RAS}$, $\overline{WE}$			2.7		7.0	Volts
VIL	Input LOW Voltage			–1.0		0.80	Volts
IIX	Input Load Current		VSS ≤ VI ≤ 7V	–10		10	µA
IOZ	Output Leakage Current		VSS ≤ VO ≤ VCC, Output OFF	–10		10	µA
ICC	VCC Supply Current		Output OFF (Note 4)	–10		10	µA
IBB	VBB Supply Current, Average		Standby, $\overline{RAS} \geq$ VIHC			100	µA
			Operating, Minimum Cycle Time			200	
IDD	VDD Supply Current, Average	Operating IDD1	$\overline{RAS}$ Cycling, $\overline{CAS}$ Cycling, Minimum Cycle Times			35	mA
		Page Mode IDD4	$\overline{RAS} \leq$ VIL, $\overline{CAS}$ Cycling, Minimum Cycle Times			27	
		$\overline{RAS}$ Only Refresh IDD3	$\overline{RAS}$ Cycling, $\overline{CAS} \geq$ VIHC, Minimum Cycle Times			27	
		Standby IDD2	$\overline{RAS} \geq$ VIHC			1.5	
CI	Input Capacitance	$\overline{RAS}$, $\overline{CAS}$, $\overline{WE}$	Inputs at 0V, f = 1MHz, Nominal Supply Voltages			10	pF
		Address, Data In				5.0	
CO	Output Capacitance		Output OFF			7.0	

SWITCHING CHARACTERISTICS over operating range (Notes 2, 3, 5, 10)

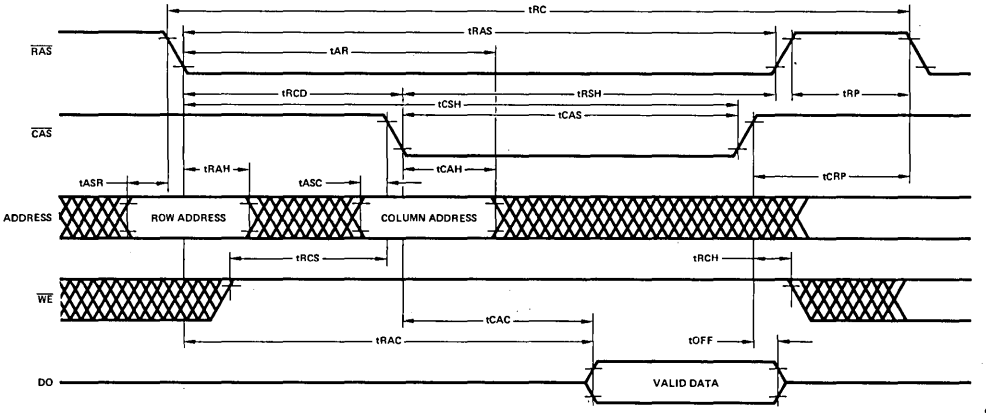
Parameters	Description	Am9016C Min	Am9016C Max	Am9016D Min	Am9016D Max	Am9016E Min	Am9016E Max	Am9016F Min	Am9016F Max	Units
tAR	$\overline{RAS}$ LOW to Column Address Hold Time	200		160		120		95		ns
tASC	Column Address Set-up Time	−10		−10		−10		−10		ns
tASR	Row Address Set-up Time	0		0		0		0		ns
tCAC	Access Time from $\overline{CAS}$ (Note 6)		185		165		135		100	ns
tCAH	$\overline{CAS}$ LOW to Column Address Hold Time	85		75		55		45		ns
tCAS	$\overline{CAS}$ Pulse Width	185	10,000	165	10,000	135	10,000	100	10,000	ns
tCP	Page Mode $\overline{CAS}$ Precharge Time	100		100		80		60		ns
tCRP	$\overline{CAS}$ to $\overline{RAS}$ Precharge Time	−20		−20		−20		−20		ns
tCSH	$\overline{CAS}$ Hold Time	300		250		200		150		ns
tCWD	$\overline{CAS}$ LOW to $\overline{WE}$ LOW Delay (Note 9)	145		125		95		70		ns
tCWL	$\overline{WE}$ LOW to $\overline{CAS}$ HIGH Set-up Time	100		85		70		50		ns
tDH	$\overline{CAS}$ LOW or $\overline{WE}$ LOW to Data In Valid Hold Time (Note 7)	85		75		55		45		ns
tDHR	$\overline{RAS}$ LOW to Data In Valid Hold Time	200		160		120		95		ns
tDS	Data In Stable to $\overline{CAS}$ LOW or $\overline{WE}$ LOW Set-up Time (Note 7)	0		0		0		0		ns
tOFF	$\overline{CAS}$ HIGH to Output OFF Delay	0	60	0	60	0	50	0	40	ns
tPC	Page Mode Cycle Time	295		275		225		170		ns
tRAC	Access Time from $\overline{RAS}$ (Note 6)		300		250		200		150	ns
tRAH	$\overline{RAS}$ LOW to Row Address Hold Time	45		35		25		20		ns
tRAS	$\overline{RAS}$ Pulse Width	300	10,000	250	10,000	200	10,000	150	10,000	ns
tRC	Random Read or Write Cycle Time	460		410		375		320		ns
tRCD	$\overline{RAS}$ LOW to $\overline{CAS}$ LOW Delay (Note 6)	35	115	35	85	25	65	20	50	ns
tRCH	Read Hold Time	0		0		0		0		ns
tRCS	Read Set-up Time	0		0		0		0		ns
tREF	Refresh Interval		2		2		2		2	ms
tRMW	Read Modify Write Cycle Time	600		500		405		320		ns
tRP	RAS Precharge Time	150		150		120		100		ns
tRSH	$\overline{CAS}$ LOW to $\overline{RAS}$ HIGH Delay	185		165		135		100		ns
tRWC	Read/Write Cycle Time	525		425		375		320		ns
tRWD	$\overline{RAS}$ LOW to $\overline{WE}$ LOW Delay (Note 9)	260		210		160		120		ns
tRWL	$\overline{WE}$ LOW to $\overline{RAS}$ HIGH Set-up Time	100		85		70		50		ns
tT	Transition Time	3	50	3	50	3	50	3	35	ns
tWCH	Write Hold Time	85		75		55		45		ns
tWCR	$\overline{RAS}$ LOW to Write Hold Time	200		160		120		95		ns
tWCS	$\overline{WE}$ LOW to $\overline{CAS}$ LOW Set-up Time (Note 9)	−20		−20		−20		−20		ns
tWP	Write Pulse Width	85		75		55		45		ns

NOTES

1. Typical values are for T_A = 25°C, nominal supply voltages and nominal processing parameters.
2. Signal transition times are assumed to be 5ns. Transition times are measured between specified high and low logic levels.
3. Timing reference levels for both input and output signals are the specified worst-case logic levels.
4. VCC is used in the output buffer only. ICC will therefore depend only on leakage current and output loading. When the output is ON and at a logic high level, VCC is connected to the Data Out pin through an equivalent resistance of approximately 135Ω. In standby mode VCC may be reduced to zero without affecting stored data or refresh operations.
5. Output loading is two standard TTL loads plus 100pF capacitance.
6. Both $\overline{RAS}$ and $\overline{CAS}$ must be low to read data. Access timing will depend on the relative positions of their falling edges. When tRCD is less than the maximum value shown, access time depends on $\overline{RAS}$ and tRAC governs. When tRCD is more than the maximum value shown access time depends on $\overline{CAS}$ and tCAC governs. The maximum value listed for tRCD is shown for reference purposes only and does not restrict operation of the part.
7. Timing reference points for data input set-up and hold times will depend on what type of write cycle is being performed and will be the later falling edge of $\overline{CAS}$ or $\overline{WE}$.
8. At least eight initialization cycles that exercise $\overline{RAS}$ should be performed after power-up and before valid operations are begun.
9. The tWCS, tRWD and tCWD parameters are shown for reference purposes only and do not restrict the operating flexibility of the part. When the falling edge of $\overline{WE}$ follows the falling edge of $\overline{CAS}$ by at most tWCS, the data output buffer will remain off for the whole cycle and an "early write" cycle is defined. When the falling edge of $\overline{WE}$ follows the falling edges of $\overline{RAS}$ and $\overline{CAS}$ by at least tRWD and tCWD respectively, the Data Out from the addressed cell will be valid at the access time and a "read/write" cycle is defined. The falling edge of $\overline{WE}$ may also occur at intermediate positions, but the condition and validity of the Data Out signal will not be known.
10. Switching characteristics are listed in alphabetical order.
11. All voltages referenced to VSS.

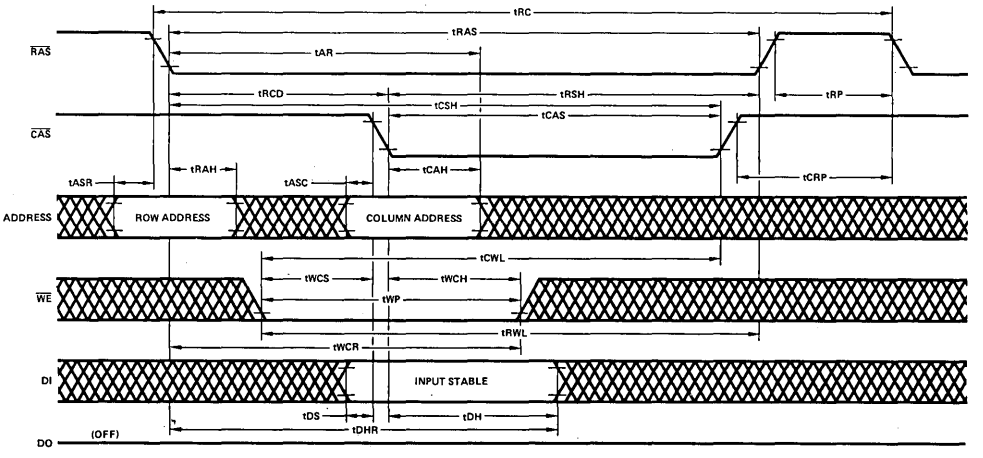
SWITCHING WAVEFORMS

READ CYCLE



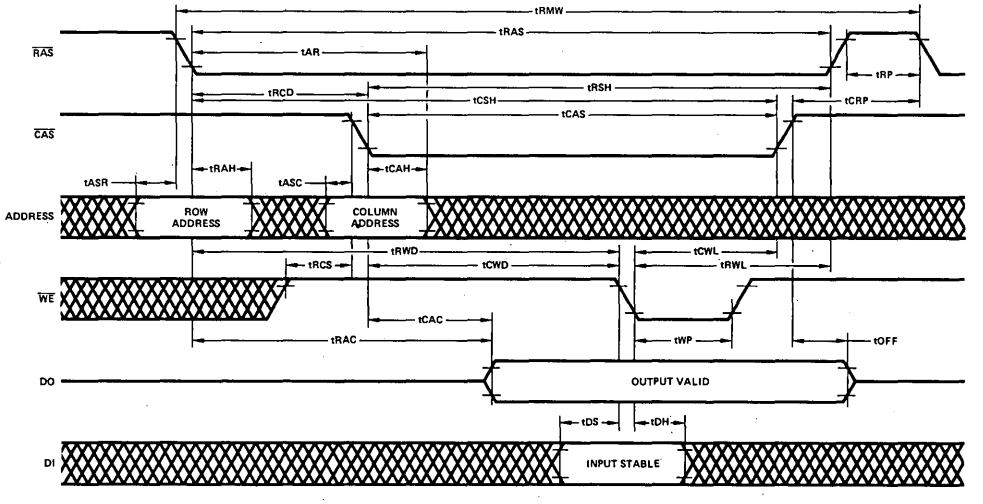
MOS-192

WRITE CYCLE (EARLY WRITE)



MOS-193

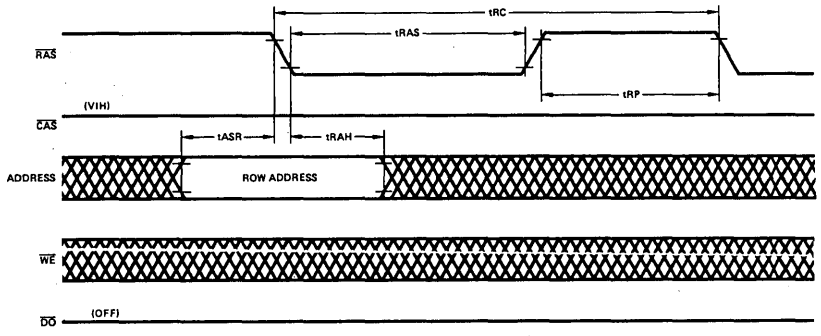
READ-WRITE/READ-MODIFY-WRITE CYCLE



MOS-194

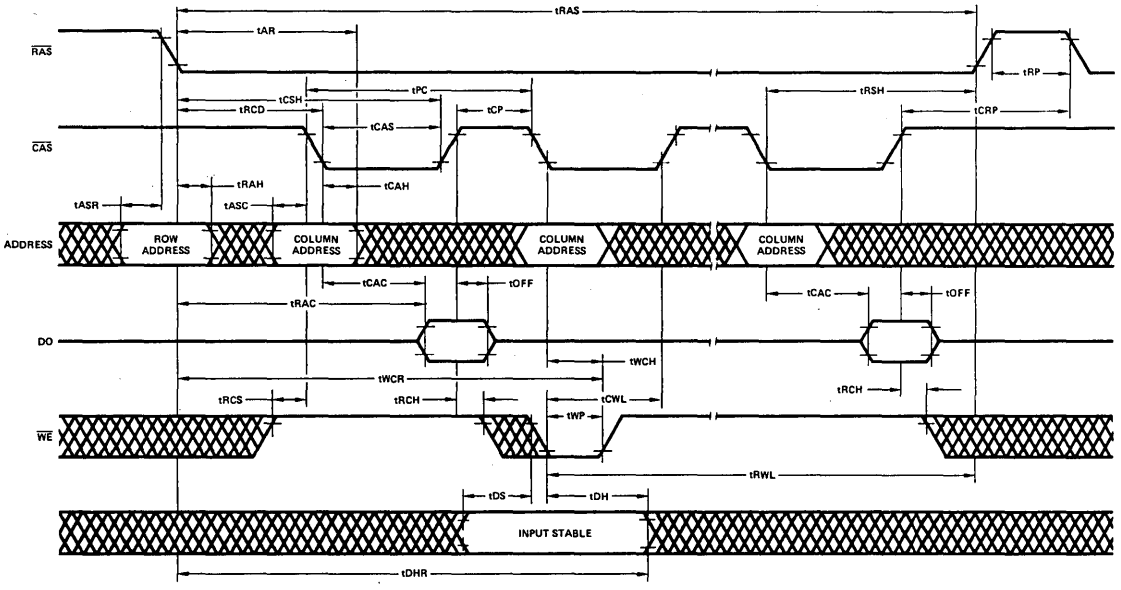
SWITCHING WAVEFORMS (Cont.)

$\overline{RAS}$ ONLY REFRESH CYCLE



MOS-195

PAGE MODE CYCLE



MOS-196

6

APPLICATION INFORMATION

The Am9016 electrical connections are such that if power is applied with the device installed upside down it will be permanently damaged. Precautions should be taken to avoid this mishap.

OPERATING CYCLES

Random read operations from any location hold the $\overline{WE}$ line high and follow this sequence of events:

1) The row address is applied to the address inputs and $\overline{RAS}$ is switched low.
2) After the row address hold time has elapsed, the column address is applied to the address inputs and $\overline{CAS}$ is switched low.
3) Following the access time, the output will turn on and valid read data will be present. The data will remain valid as long as $\overline{CAS}$ is low.
4) $\overline{CAS}$ and $\overline{RAS}$ are then switched high to end the operation. A new cycle cannot begin until the precharge period has elapsed.

Random write operations follow the same sequence of events, except that the $\overline{WE}$ line is low for some portion of the cycle. If the data to be written is available early in the cycle, it will usually be convenient to simply have $\overline{WE}$ low for the whole write operation.

Sequential Read and Write operations at the same location can be designed to save time because re-addressing is not necessary. A read/write cycle holds $\overline{WE}$ high until a valid read is established and then strobes new data in with the falling edge of $\overline{WE}$.

After the power is first applied to the device, the internal circuit requires execution of at least eight initialization cycles which exercise $\overline{RAS}$ before valid memory accesses are begun.

ADDRESSING

14 address bits are required to select one location out of the 16,384 cells in the memory. Two groups of 7 bits each are multiplexed onto the 7 address lines and latched into the internal address registers. Two negative-going external clocks are used to control the multiplexing. The Row Address Strobe ($\overline{RAS}$) enters the row address bits and the Column Address Strobe ($\overline{CAS}$) enters the column address bits.

When $\overline{RAS}$ is inactive, the memory enters its low power standby mode. Once the row address has been latched, it need not be changed for successive operations within the same row, allowing high-speed page-mode operations.

Page-mode operations first establish the row address and then maintain $\overline{RAS}$ low while $\overline{CAS}$ is repetitively cycled and designated operations are performed. Any column address within the selected row may be accessed in any sequence. The maximum time that $\overline{RAS}$ can remain low is the factor limiting the number of page-mode operations that can be performed.

Multiplexed addressing does not introduce extra delays in the access path. By inserting the row address first and the column address second, the memory takes advantage of the fact that the delay path through the memory is shorter for column addresses. The column address does not propagate through the cell matrix as the row address does and it can therefore arrive somewhat later than the row address without impacting the access time.

REFRESH

The Am9016 is a dynamic memory and each cell must be refreshed at least once every refresh interval in order to maintain the cell contents. Any operation that accesses a row serves to refresh all 128 cells in the row. Thus the refresh requirement is met by accessing all 128 rows at least once every refresh interval. This may be accomplished, in some applications, in the course of performing normal operations. Alternatively, special refresh operations may be initiated. These special operations could be simply additional conventional accesses or they could be "$\overline{RAS}$-only" cycles. Since only the rows need to be addressed, $\overline{CAS}$ may be held high while $\overline{RAS}$ is cycled and the appropriate row addresses are input. Power required for refreshing is minimized and simplified control circuitry will often be possible.

DATA INPUT/OUTPUT

Data is written into a selected cell by the combination of $\overline{WE}$ and $\overline{CAS}$ while $\overline{RAS}$ is low. The later negative transition of $\overline{WE}$ or $\overline{CAS}$ strobes the data into the internal register. In a write cycle, if the $\overline{WE}$ input is brought low prior to $\overline{CAS}$, the data is strobed by $\overline{CAS}$, and the set-up and hold times are referenced to $\overline{CAS}$. If the cycle is a read/write cycle then the data set-up and hold times are referenced to the negative edge of $\overline{WE}$.

In the read cycle the data is read by maintaining $\overline{WE}$ in the high state throughout the portion of the memory cycle in which $\overline{CAS}$ is low. The selected valid data will appear at the output within the specified access time.

DATA OUTPUT CONTROL

Any time $\overline{CAS}$ is high the data output will be off. The output contains either one or zero during read cycle after the access time has elapsed. Data remains valid from the access time until $\overline{CAS}$ is returned to the high state. The output data is the same polarity as the input data.

The user can control the output state during write operations by controlling the placement of the $\overline{WE}$ signal. In the "early write" cycle (see note 9) the output is at a high impedance state throughout the entire cycle.

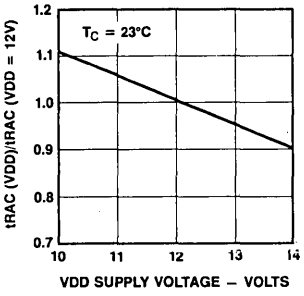
POWER CONSIDERATIONS

$\overline{RAS}$ and/or $\overline{CAS}$ can be decoded and used as a chip select signal for the Am9016 but overall system power is minimized if $\overline{RAS}$ is used for this purpose. The devices which do not receive $\overline{RAS}$ will be in low power standby mode regardless of the state of $\overline{CAS}$.

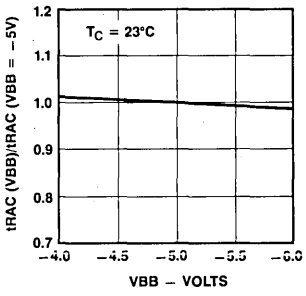
At all times the Absolute Maximum Rating Conditions must be observed. During power supply sequencing VBB should never be more positive than VSS when power is applied to VDD.

TYPICAL CHARACTERISTICS

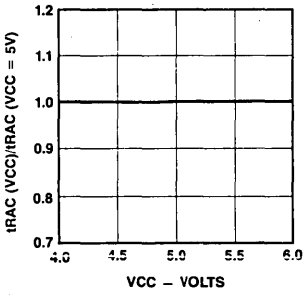
Typical Access Time (Normalized) tRAC Versus VDD



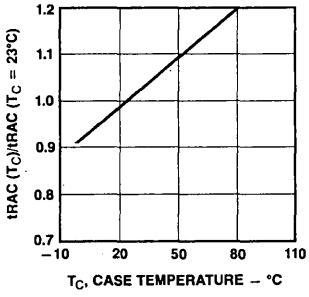
Typical Access Time (Normalized) tRAC Versus VBB



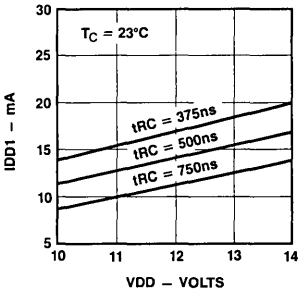
Typical Access Time (Normalized) tRAC Versus VCC



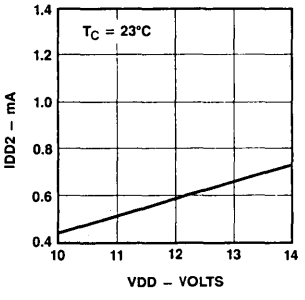
Typical Access Time (Normalized) tRAC Versus Case Temperature



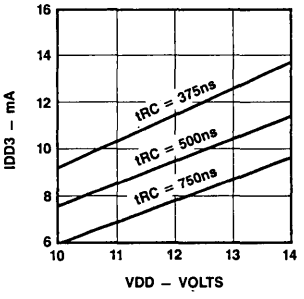
Typical Operating Current IDD1 Versus VDD



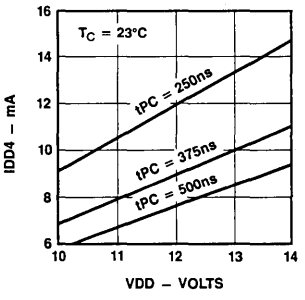
Typical Standby Current IDD2 Versus VDD



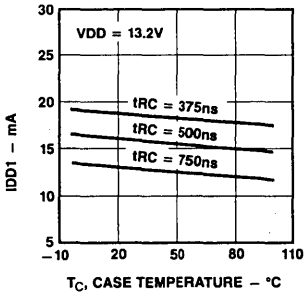
Typical Refresh Current IDD3 Versus VDD



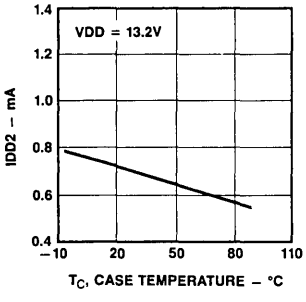
Typical Page Mode Current IDD4 Versus VDD



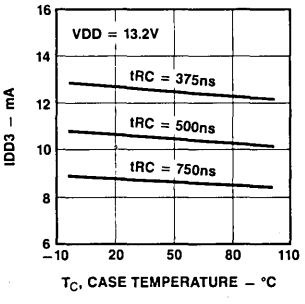
Typical Operating Current IDD1 Versus Case Temperature



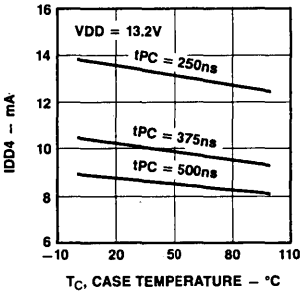
Typical Standby Current IDD2 Versus Case Temperature



Typical Refresh Current IDD3 Versus Case Temperature



Typical Page Mode Current IDD4 Versus Case Temperature

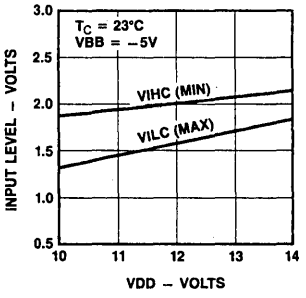


MOS-197

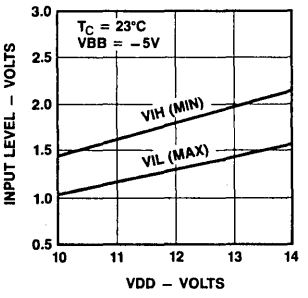
6

TYPICAL CHARACTERISTICS (Cont.)

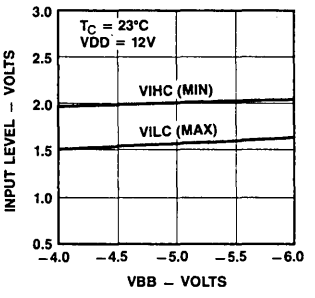
Input Voltage Levels Versus VDD



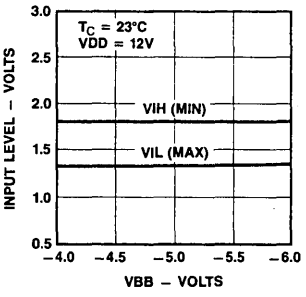
Input Voltage Levels Versus VDD



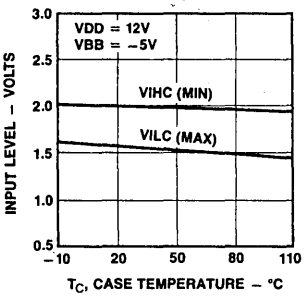
Input Voltage Levels Versus VBB



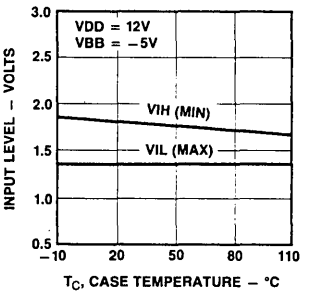
Input Voltages Levels Versus VBB



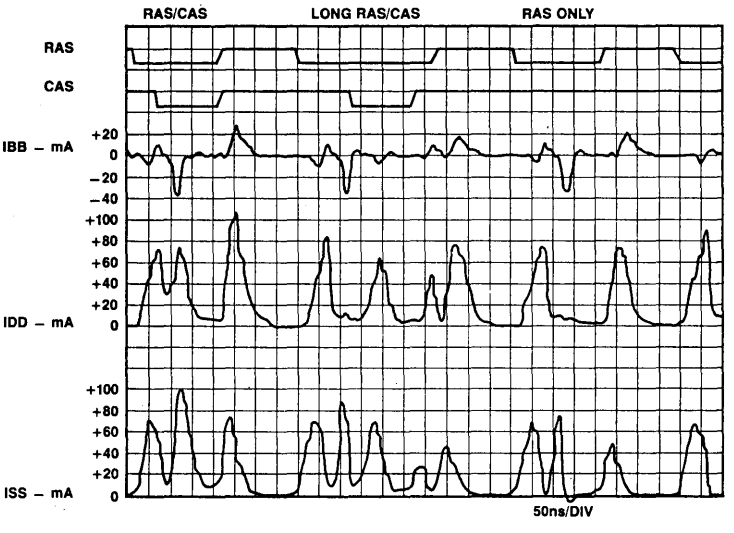
Input Voltage Levels Versus Case Temperature



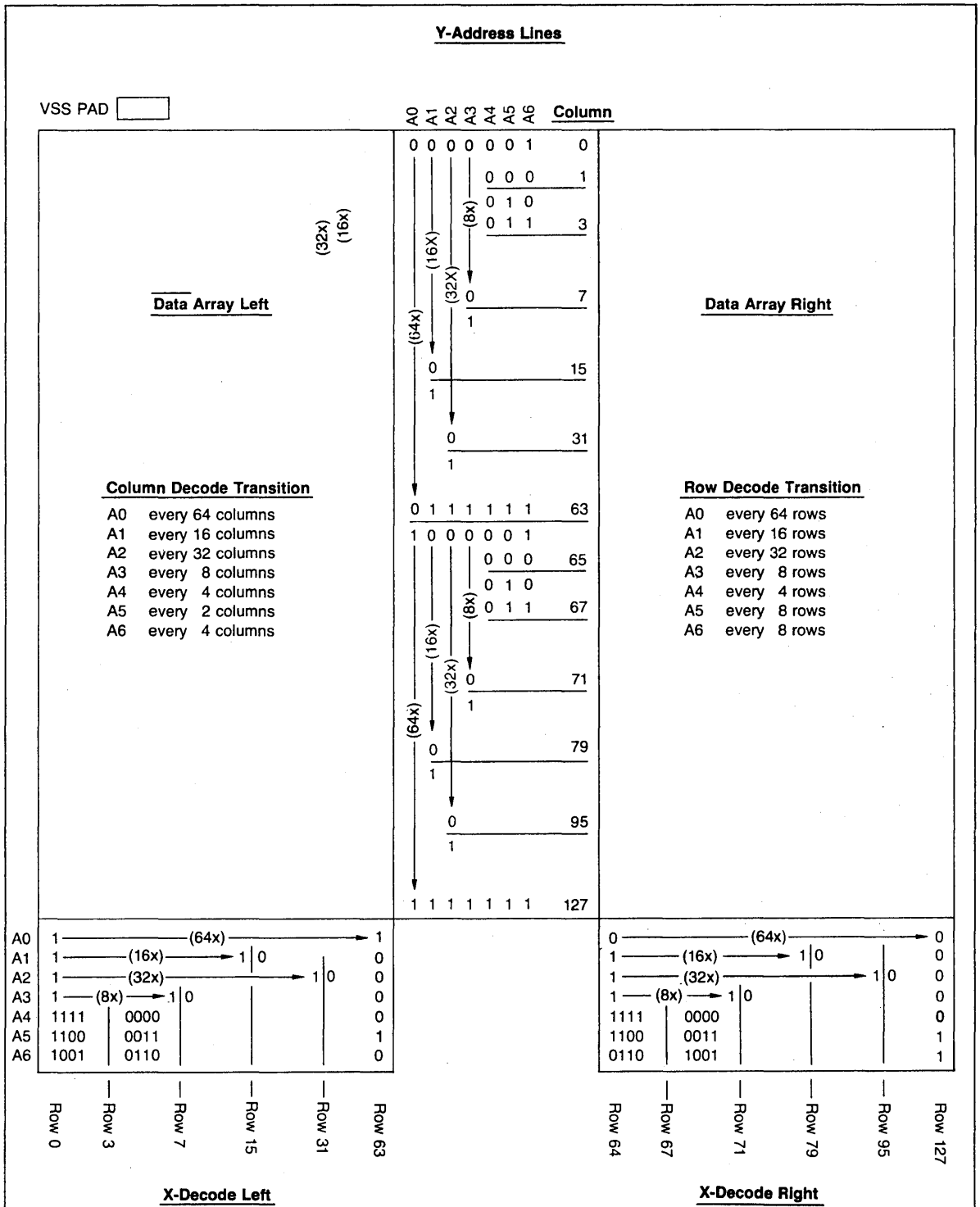
Input Voltage Levels Versus Case Temperature



TYPICAL CURRENT WAVEFORMS

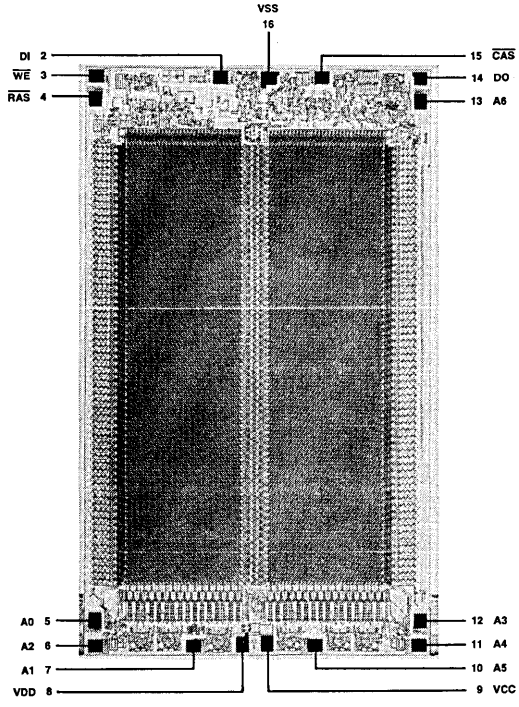


MOS-198



TOPOLOGICAL BIT MAP

Metallization and Pad Layout



DIE SIZE 0.106" X 0.205"

Am9044 • Am9244

4096 x 1 Static R/W Random Access Memory

DISTINCTIVE CHARACTERISTICS

- **LOW OPERATING POWER (MAX)**
  Am9044/Am9244 385mW (70mA)
  Am90L44/Am92L44 275mW (50mA)
- **LOW STANDBY POWER (MAX)**
  Am92L44 110mW (20mA)
- Access times down to 200ns (max)
- Military temperature range available to 250ns (max)
- Am9044 is a direct plug-in replacement for 4044
- Am9244 pin and function compatible with Am9044 and 4044 plus $\overline{CS}$ power down feature
- Fully static – no clocking
- Identical access and cycle time
- High output drive – 4.0mA sink current @ 0.4V
- TTL identical interface logic levels
- 100% MIL-STD-883 reliability assurance testing

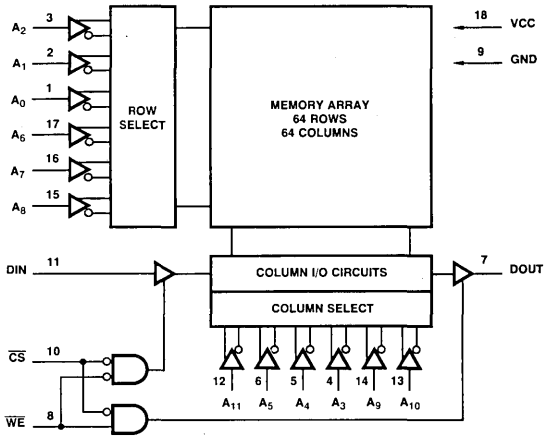
GENERAL DESCRIPTION

The Am9044 and Am9244 are high performance, static, N-Channel, read/write, random access memories organized as 4096 x 1. Operation is from a single 5V supply, and all input/output levels are identical to standard TTL specifications. Low power versions of both devices are available with power savings of about 30%. The Am9044 and Am9244 are the same except that the Am9244 offers an automatic $\overline{CS}$ power down feature.

The Am9244 remains in a low power standby mode as long as $\overline{CS}$ remains high, thus reducing its power requirements. The Am9244 power decreases from 385mW to 165mW in the standby mode, and the Am92L44 from 275mW to 110mW. The $\overline{CS}$ input does not affect the power dissipation of the Am9044.

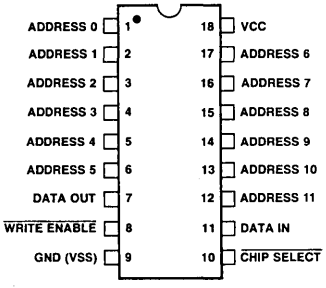
Data readout is not destructive and the same polarity as data input. $\overline{CS}$ provides for easy selection of an individual package when the outputs are OR-tied. The outputs of 4.0mA for Am9244 and Am9044 provide increased short circuit current for improved compacitive drive.

BLOCK DIAGRAM



MOS-256

CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

MOS-257

ORDERING INFORMATION

Ambient Temperature	Package Type	ICC Current Level	Access Times							
			Am9044				Am9244			
			450ns	300ns	250ns	200ns	450ns	300ns	250ns	200ns
0°C ≤ T_A ≤ +70°C	Plastic	70mA	AM9044BPC	AM9044CPC	AM9044DPC	AM9044EPC	AM9244BPC	AM9244CPC	AM9244DPC	AM9244EPC
		50mA	AM90L44BPC	AM90L44CPC	AM90L44DPC		AM92L44BPC	AM92L44CPC	AM92L44DPC	
	Hermetic	70mA	AM9044BDC	AM9044CDC	AM9044DDC	AM9044EDC	AM9244BDC	AM9244CDC	AM9244DDC	AM9244EDC
		50mA	AM90L44BDC	AM90L44CDC	AM90L44DDC		AM92L44BDC	AM92L44CDC	AM92L44DDC	
−55°C ≤ T_A ≤ +125°C	Hermetic	90mA	AM9044BDM	AM9044CDM	AM9044DDM		AM9244BDM	AM9244CDM	AM9244DDM	
		60mA	AM90L44BDM	AM90L44CDM			AM92L44BDM	AM92L44CDM		

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	−0.5V to +7.0V
All Signal Voltages with Respect to VSS	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.0W
DC Output Current	10mA

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VSS	VCC	Part Number	Ambient Temperature	VSS	VCC
Am9044DC/PC Am90L44DC/PC Am9244DC/PC Am92L44DC/PC	$0°C \leq T_A \leq +70°C$	0V	+5.0V ±10%	Am9044DM Am90L44DM Am9244DM Am92L44DM	$-55°C \leq T_A \leq +125°C$	0V	+5.0V ±10%

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions			Am9244XX Am92L44XX Min.	Typ.	Max.	Am9044XX Am90L44XX Min.	Typ.	Max.	Units
IOH	Output High Current	VOH = 2.4V	VCC = 4.5V	70°C	−1.0			−1.0			mA
		VOH = 2.4V	VCC = 4.5V	125°C	−.4			−.4			
IOL	Output Low Current	VOL = 0.4V	T_A = +70°C		4.0			4.0			mA
			T_A = +125°C		3.2			3.2			
VIH	Input High Voltage				2.0		VCC	2.0		VCC	Volts
VIL	Input Low Voltage				−0.5		0.8	−0.5		0.8	Volts
IIX	Input Load Current	VSS ≤ VI ≤ VCC					10			10	µA
IOZ	Output Leakage Current	0.4V ≤ VO ≤ VCC Output Disabled	T_A = +125°C		−50		50	−50		50	µA
			T_A = +70°C		−10		10	−10		10	
CI	Input Capacitance (Note 1)	Test Frequency = 1.0MHz T_A = 25°C, All pins at 0V				3.0	5.0		3.0	5.0	pF
CI/O	I/O Capacitance (Note 1)					5.0	6.0		5.0	6.0	

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am92L44 Typ.	Max.	Am9244 Typ.	Max.	Am90L44 Typ.	Max.	Am9044 Typ.	Max.	Units
ICC	VCC Operating Supply Current	Max. VCC $\overline{CS}$ ≤ VIL for Am9244/92L44	T_A = 0°C		50		70		50		70	mA
			T_A = −55°C		60		80		60		80	
IPD	Automatic $\overline{CS}$ Power Down Current	Max. V_{CC} ($\overline{CS} \geq V_{IH}$)	T_A = 0°C		20		30		–		–	mA
			T_A = −55°C		22		33		–		–	

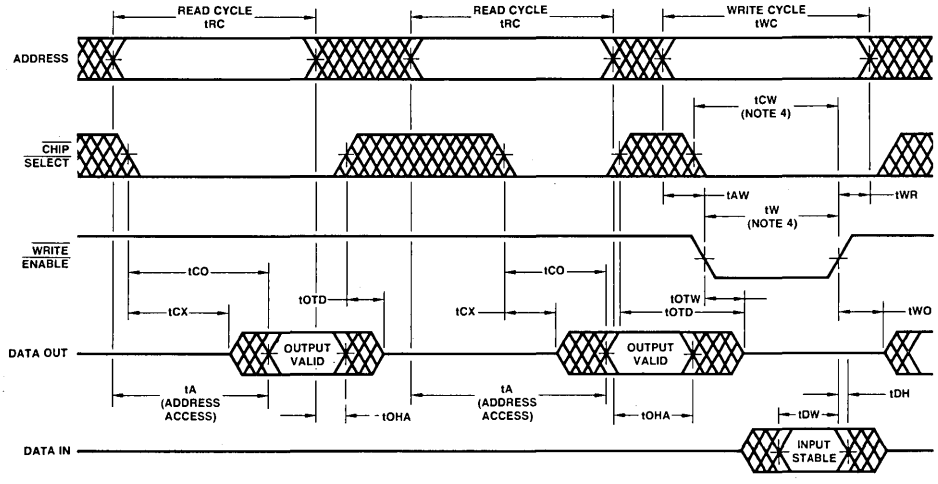
Notes:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. For test purposes, not more than one output at a time should be shorted. Short circuit test duration should not exceed 30 seconds.
3. Test conditions assume signal transition times of 10ns or less, timing reference levels of 1.5V and output loading of one standard TTL gate plus 100pF.
4. The internal write time of the memory is defined by the overlap of $\overline{CS}$ low and $\overline{WE}$ low. Both signals must be low to initiate a write and either signal can terminate a write by going high. The data input setup and hold timing should be referenced to the rising edge of the signal that terminates the write.
5. Chip Select access time (t_{CO}) is longer for the Am9244 than for the Am9044. The specified address access time will be valid only when Chip Select is low soon enough for t_{CO} to elapse.

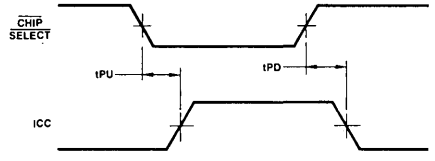
SWITCHING CHARACTERISTICS over operating range (Note 3)

Parameter	Description		Am9044B Am9244B Min.	Am9044B Am9244B Max.	Am9044C Am9244C Min.	Am9044C Am9244C Max.	Am9044D Am9244D Min.	Am9044D Am9244D Max.	Am9044E Am9244E Min.	Am9044E Am9244E Max.	Units
Read Cycle											
tRC	Address Valid to Address Do Not Care Time (Read Cycle Time)		450		300		250		200		ns
tA	Address Valid to Data Out Valid Delay (Address Access Time)			450		300		250		200	ns
tCO	Chip Select Low to Data Out Valid (Note 5)	Am9044		100		100		70		70	ns
		Am9244		450		300		250		200	ns
tCX	Chip Select Low to Data Out On		20		20		20		20		ns
tOTD	Chip Select High to Data Out Off			100		80		60		60	ns
tOHA	Address Unknown to Data Out Unknown Time		20		20		20		20		ns
Write Cycle											
tWC	Address Valid to Address Do Not Care Time (Write Cycle Time)		450		300		250		200		ns
tW	Write Enable Low to Write Enable High Time (Note 4)	Am9044	200		150		100		100		ns
		Am9244	250		200		150		150		ns
tWR	Write Enable High to Address Do Not Care Time		0		0		0		0		ns
tOTW	Write Enable Low to Data Out Off Delay			100		80		60		60	ns
tDW	Data In Valid to Write Enable High Time		200		150		100		100		ns
tDH	Write Enable Low to Data In Do Not Care Time		0		0		0		0		ns
tAW	Address Valid to Write Enable Low Time		0		0		0		0		ns
tPD	Chip Select High to Power Low Delay (Am9244 only)			200		150		100		100	ns
tPU	Chip Select Low to Power High Delay (Am9244 only)		0		0		0		0		ns
tCW	Chip Select Low to Write Enable High Time (Note 4)	Am9044	200		150		100		100		ns
		Am9244	250		200		150		150		ns
tWO	Write Enable High To Output Turn On			100		100		70		70	ns

SWITCHING WAVEFORMS



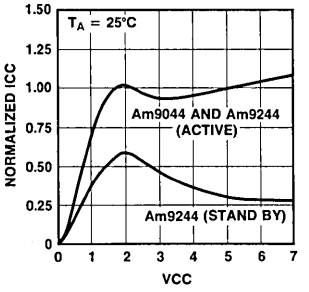
POWER DOWN WAVEFORM (Am9244 ONLY)



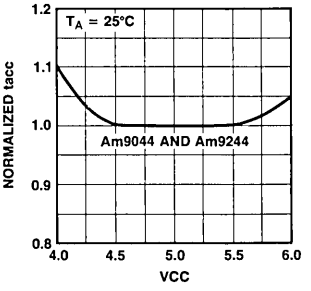
MOS-258

TYPICAL CHARACTERISTICS

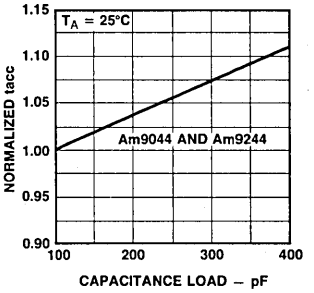
Typical ICC Versus VCC Characteristics



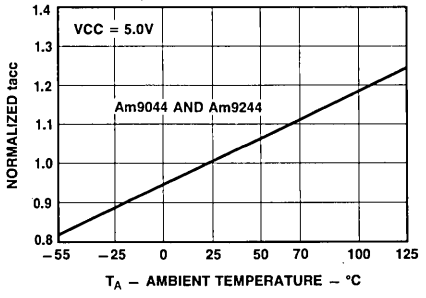
Typical tacc Versus VCC Characteristics



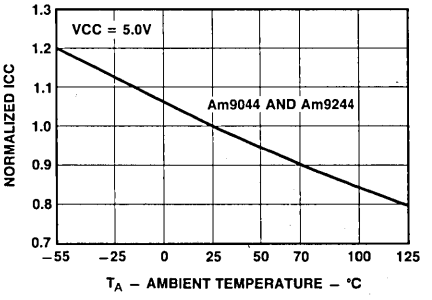
Typical C Load Versus Normalized tacc Characteristics



Normalized tacc Versus Ambient Temperature

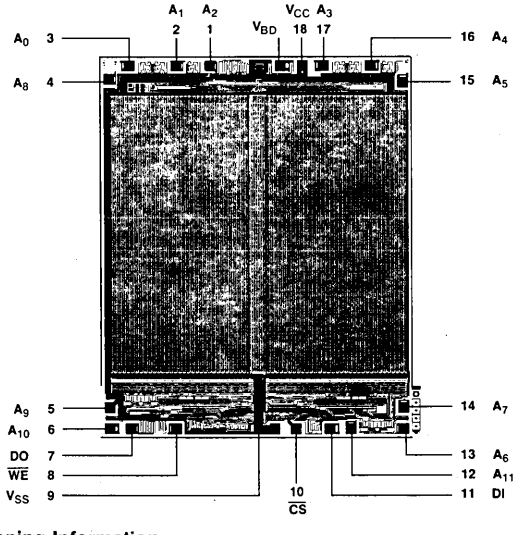


Normalized ICC Versus Ambient Temperature



MOS-259

BIT MAP



Address Designators	
External	Internal
A0	A2
A1	A1
A2	A0
A3	A8
A4	A9
A5	A10
A6	A3
A7	A4
A8	A5
A9	A7
A10	A6
A11	A11

Figure 1. Bit Mapping Information.

Am9114 • Am9124

1024 x 4 Static R/W Random Access Memory

DISTINCTIVE CHARACTERISTICS

- **LOW OPERATING POWER (MAX)**
  Am9124/Am9114 368mW (70mA)
  Am91L24/Am91L14 262mW (50mA)
- **LOW STANDBY POWER (MAX)**
  Am9124 158mW (30mA)
  Am91L24 105mW (20mA)
- Access times down to 200ns (max)
- Military temperature range available to 300ns (max)
- Am9114 is a direct plug-in replacement for 2114
- Am9124 pin and function compatible with Am9114 and 2114, plus $\overline{CS}$ power down feature
- Fully static – no clocking
- Identical access and cycle time
- High output drive –
  4.0mA sink current @ 0.4V – 9124
  3.2mA sink current @ 0.4V – 9114
- TTL identical input/output levels
- 100% MIL-STD-883 reliability assurance testing

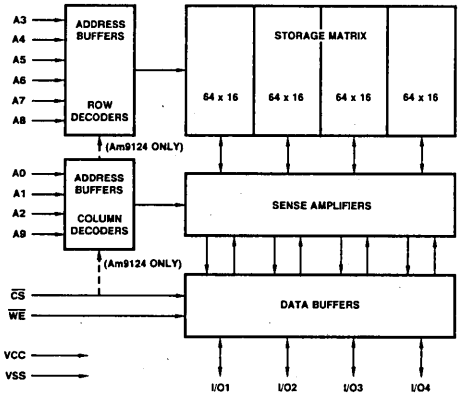
GENERAL DESCRIPTION

The Am9114 and Am9124 are high performance, static, N-Channel, read/write, random access memories organized as 1024 x 4. Operation is from a single 5V supply, and all input/output levels are identical to standard TTL specifications. Low power versions of both devices are available with power savings of over 30%. The Am9114 and Am9124 are the same except that the Am9124 offers an automatic $\overline{CS}$ power down feature.

The Am9124 remains in a low power standby mode as long as $\overline{CS}$ remains high, thus reducing its power requirements. The Am9124 power decreases from 368mW to 158mW in the standby mode, and the Am91L24 from 262mW to 105mW. The $\overline{CS}$ input does not affect the power dissipation of the Am9114. (See Figure 1, page 4).

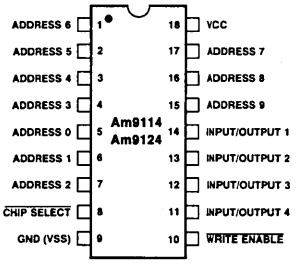
Data readout is not destructive and the same polarity as data input. $\overline{CS}$ provides for easy selection of an individual package when the outputs are OR-tied. The outputs of 4.0mA for Am9124 and 3.2mA for Am9114 provides increased short circuit current for improved capacitive drive.

BLOCK DIAGRAM



MOS-066

CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

MOS-067

ORDERING INFORMATION

Ambient Temperature	Package Type	ICC Current Level	Access Times					
			Am9114			Am9124 (Power Down Option)		
			450ns	300ns	200ns	450ns	300ns	200ns
$0°C \leq T_A \leq 70°C$	Plastic	70mA	Am9114BPC	Am9114CPC	Am9114EPC	Am9124BPC	Am9124CPC	Am9124EPC
		50mA	Am91L14BPC	Am91L14CPC		Am91L24BPC	Am91L24CPC	
	Hermetic	70mA	Am9114BDC	Am9114CDC	Am9114EDC	Am9124BDC	Am9124CDC	Am9124EDC
		50mA	Am91L14BDC	Am91L14CDC		Am91L24BDC	Am91L24CDC	
$-55°C \leq T_A \leq +125°C$	Hermetic	80mA	Am9114BDM	Am9114CDM		Am9124BDM	Am9124CDM	
		60mA	Am91L14BDM	Am91L14CDM		Am91L24BDM	Am91L24CDM	

MAXIMUM RATINGS beyond which useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
V_{CC} with Respect to V_{SS}	−0.5V to +7.0V
All Signal Voltages with Respect to V_{SS}	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.0W
DC Output Current	10mA

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	V_{SS}	V_{CC}	Part Number	Ambient Temperature	V_{SS}	V_{CC}
Am9114DC/PC Am91L14DC/PC Am9124DC/PC Am91L24DC/PC	$0°C \leq T_A \leq +70°C$	0V	+5.0V ± 5%	Am9114DM Am91L14DM Am9124DM Am91L24DM	$-55°C \leq T_A \leq +125°C$	0V	+5.0V ± 10%

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am9124XX Am91L24XX Min.	Typ.	Max.	Am9114XX Am91L14XX Min.	Typ.	Max.	Units
I_{OH}	Output High Current	V_{OH} = 2.4V	V_{CC} = 4.75V	−1.4			−1.0			mA
		V_{OH} = 2.2V	V_{CC} = 4.5V	−1.0			−1.0			
I_{OL}	Output Low Current	V_{OL} = 0.4V	T_A = +70°C	4.0			3.2			mA
			T_A = +125°C	3.2			2.4			
V_{IH}	Input High Voltage			2.0		V_{CC}	2.0		V_{CC}	Volts
V_{IL}	Input Low Voltage			−0.5		0.8	−0.5		0.8	Volts
I_{IX}	Input Load Current	$V_{SS} \leq V_I \leq V_{CC}$				10			10	µA
I_{OZ}	Output Leakage Current	$0.4V \leq V_O \leq V_{CC}$ Output Disabled	T_A = +125°C	−50		50	−50		50	µA
			T_A = +70°C	−10		10	−10		10	
I_{OS}	Output Short Circuit Current	(Note 2)	0°C to +70°C			95			75	mA
			−55°C to +125°C			115			75	
CI	Input Capacitance (Note 1)	Test Frequency = 1.0MHz			3.0	5.0		3.0	5.0	pF
CI/O	I/O Capacitance (Note 1)	T_A = 25°C, All pins at 0V			5.0	6.0		5.0	6.0	

ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am91L24 Typ.	Am91L24 Max.	Am9124 Typ.	Am9124 Max.	Am91L14 Typ.	Am91L14 Max.	Am9114 Typ.	Am9114 Max.	Units
I_{CC}	V_{CC} Operating Supply Current	Max. V_{CC}, $\overline{CS} \leq V_{IL}$ for Am9124/91L24	T_A = 25°C	40		60		40		60		mA
			T_A = 0°C		50		70		50		70	
			T_A = −55°C		60		80		60		80	
I_{PD}	Automatic $\overline{CS}$ Power Down Current	Max. V_{CC} ($\overline{CS} \geq V_{IH}$)	T_A = 25°C	15		24			–		–	mA
			T_A = 0°C		20		30		–		–	
			T_A = −55°C		22		33		–		–	

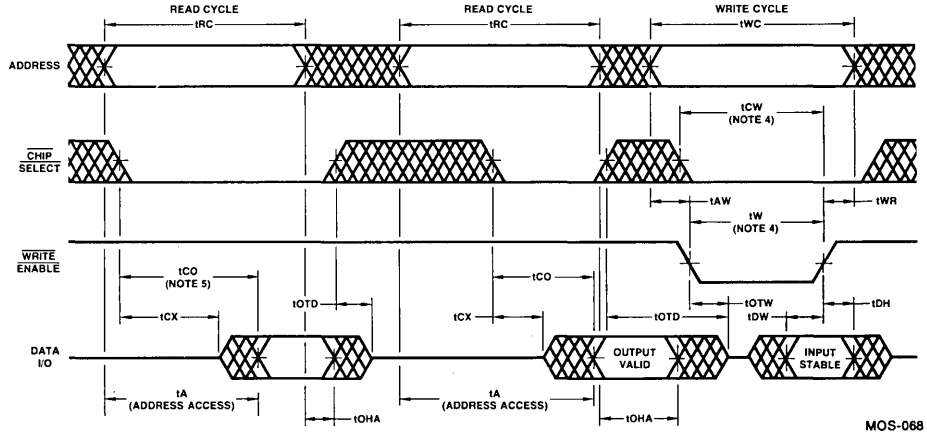
Notes:

1. Typical values are for T_A = 25°C, nominal supply voltage and nominal processing parameters.
2. For test purposes, not more than one output at a time should be shorted. Short circuit test duration should not exceed 30 seconds.
3. Test conditions assume signal transition times of 10ns or less, timing reference levels of 1.5V and output loading of one standard TTL gate plus 100pF.
4. The internal write time of the memory is defined by the overlap of $\overline{CS}$ low and $\overline{WE}$ low. Both signals must be low to initiate a write and either signal can terminate a write by going high. The data input setup and hold timing should be referenced to the rising edge of the signal that terminates the write.
5. Chip Select access time (t_{CO}) is longer for the Am9124 than for the Am9114. The specified address access time will be valid only when Chip Select is low soon enough for t_{CO} to elapse.

SWITCHING CHARACTERISTICS over operating range (Note 3)

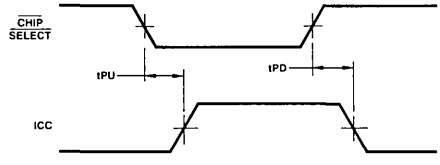
Parameter	Description		Am9114B Am9124B Min.	Am9114B Am9124B Max.	Am9114C Am9124C Min.	Am9114C Am9124C Max.	Am9114E Am9124E Min.	Am9114E Am9124E Max.	Unit
Read Cycle									
tRC	Address Valid to Address Do Not Care Time (Read Cycle Time)		450		300		200		ns
tA	Address Valid to Data Out Valid Delay (Address Access Time)			450		300		200	
tCO	Chip Select Low to Data Out Valid (Note 5)	Am9114		120		100		70	
		Am9124		420		280		185	
tCX	Chip Select Low to Data Out On		20		20		20		
tOTD	Chip Select High to Data Out Off			100		80		60	
tOHA	Address Unknown to Data Out Unknown Time		50		50		50		
Write Cycle									
tWC	Address Valid to Address Do Not Care Time (Write Cycle Time)		450		300		200		ns
tW	Write Enable Low to Write Enable High Time (Note 4)	Am9114	200		150		120		
		Am9124	250		200		150		
tWR	Write Enable High to Address Do Not Care Time		0		0		0		
tOTW	Write Enable Low to Data Out Off Delay			100		80		60	
tDW	Data In Valid to Write Enable High Time		200		150		120		
tDH	Write Enable Low to Data In Do Not Care Time		0		0		0		
tAW	Address Valid to Write Enable Low Time		0		0		0		
tPD	Chip Select High to Power Low Delay (Am9124 only)			200		150		100	
tPU	Chip Select Low to Power High Delay (Am9124 only)		0		0		0		
tCW	Chip Select Low to Write Enable High Time (Note 4)	Am9114	200		150		120		
		Am9124	250		200		150		

SWITCHING WAVEFORMS



MOS-068

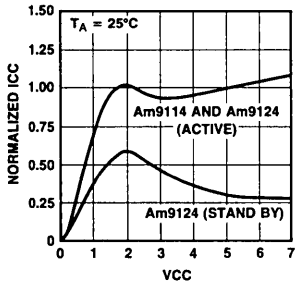
POWER DOWN WAVEFORM (Am9124 ONLY)



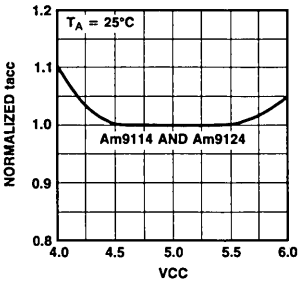
MOS-069

TYPICAL CHARACTERISTICS

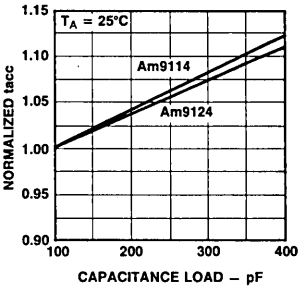
Typical ICC Versus VCC Characteristics



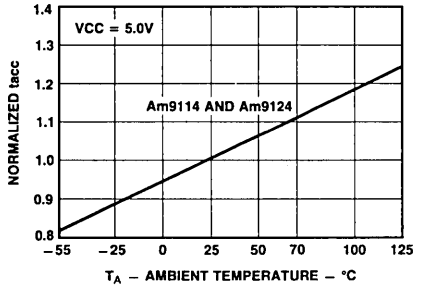
Typical tacc Versus VCC Characteristics



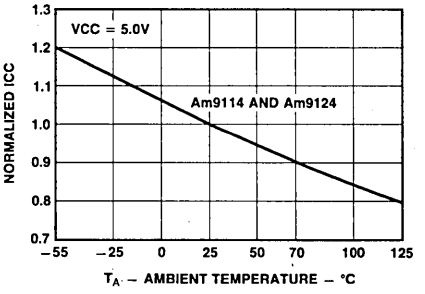
Typical C Load Versus Normalized tacc Characteristics



Normalized tacc Versus Ambient Temperature



Normalized ICC Versus Ambient Temperature



Configuration	Part Number	Worst Case Current (mA at 0°C) 100% Duty Cycle	50% Duty Cycle
2K x 8	9114	280	280
	91L14	200	200
	9124	200	160
	91L24	140	110
4K x 12	9114	840	840
	91L14	600	600
	9124	480	420
	91L24	330	285
8K x 16	9114	2240	2240
	91L14	1600	1600
	9124	1120	1040
	91L24	760	700

Figure 1. Supply Current Advantages of Am9124.

BIT MAP

Address Designators	
External	Internal
A0	A9
A1	A8
A2	A7
A3	A0
A4	A1
A5	A2
A6	A3
A7	A4
A8	A5
A9	A6

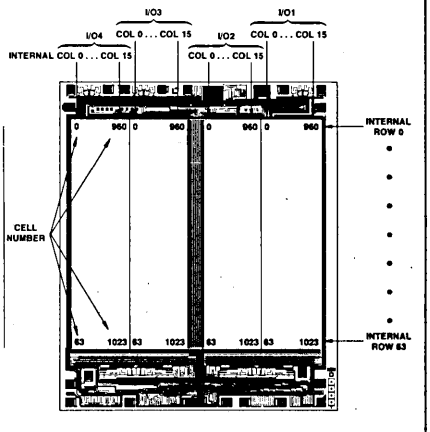


Figure 2. Bit Mapping Information.

Am2716

2048 x 8-Bit UV Erasable PROM

DISTINCTIVE CHARACTERISTICS

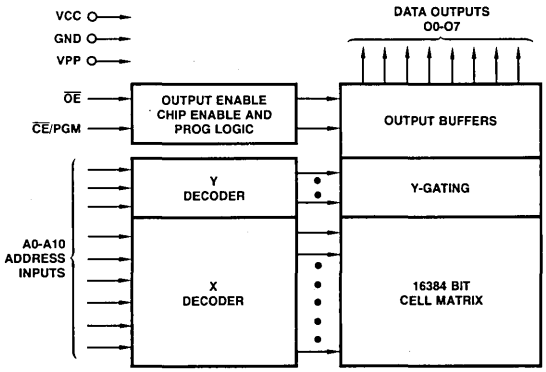
- Direct replacement for Intel 2716
- Interchangeable with Am9218 – 16K ROM
- Single +5V power supply
- Fast access time – 450ns
- Low power dissipation
  - –525mW active
  - –132mW standby
- Fully static operation – no clocks
- Three-state outputs
- TTL compatible inputs/outputs
- 100% MIL-STD-883 reliability assurance testing

GENERAL DESCRIPTION

The Am2716 is a 16384-bit ultraviolet erasable and programmable read-only memory. It is organized as 2048 words by 8 bits per word, operates from a single +5V supply, has a static standby mode, and features fast single address location programming.

Because the Am2716 operates from a single +5V supply, it is ideal for use in microprocessor systems. All programming signals are TTL levels, requiring a single pulse. For programming outside of the system, existing EPROM programmers may be used. Locations may be programmed singly, in blocks, or at random. Total programming time for all bits is 100 seconds.

BLOCK DIAGRAM

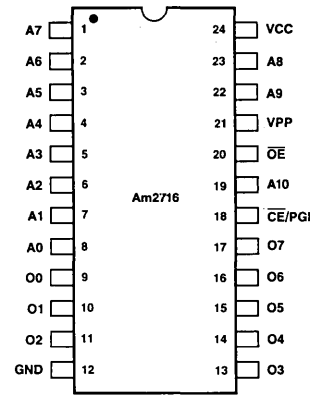


MOS-199

MODE SELECTION

Mode \ Pins	CE/PGM (18)	OE (20)	VPP (21)	VCC (24)	Outputs (9-11, 13-17)
Read	VIL	VIL	+5	+5	DOUT
Standby	VIH	Don't Care	+5	+5	High Z
Program	Pulsed VIL to VIH	VIH	+25	+5	DIN
Program Verify	VIL	VIL	+25	+5	DOUT
Program Inhibit	VIL	VIH	+25	+5	High Z

PIN CONFIGURATION



MOS-200

A0-A9: Addresses
O0-O7: Outputs
CE/PGM: Chip Enable/Program
OE: Output Enable

ORDERING INFORMATION

Package Type	Ambient Temperature Specification	Order Number
Hermetic DIP Transparent Window	$0°C \leq T_A \leq +70°C$	AM2716DC

MAXIMUM RATINGS above which the useful life may be impaired

Storage Temperature	−65°C to +125°C
Ambient Temperature Under Bias	−10°C to +80°C
Voltage on all inputs/outputs (except VPP) with respect to GND	+6V to −0.3V
Voltage on VPP During Program with Respect to GND	+26.5V to −0.3V

READ OPERATION

DC CHARACTERISTICS

$0°C \leq T_A \leq +70°C$, VCC (Notes 1, 2) = +5V ±5%, VPP (Note 2) = VCC

Parameters	Description	Test Conditions	Min	Max	Units
ILI	Input Load Current	VIN = 5.25V		10	μA
ILO	Output Leakage Current	VOUT = 5.25V		10	μA
IPP1 (Note 2)	VPP Current	VPP = 5.25V		5	mA
ICC1 (Note 2)	VCC Current (Standby)	$\overline{CE}$ = VIH, $\overline{OE}$ = VIL		25	mA
ICC2 (Note 2)	VCC Current (Active)	$\overline{OE}$ = $\overline{CE}$ = VIL		100	mA
VIL	Input Low Voltage		−0.1	0.8	Volts
VIH	Input High Voltage		2.0	VCC+1	Volts
VOL	Output Low Voltage	IOL = 2.1mA		0.45	Volts
VOH	Output High Voltage	IOH = −400μA	2.4		Volts

AC CHARACTERISTICS

$0°C \leq T_A \leq +70°C$, VCC (Notes 1, 2) = +5V ±5%, VPP (Note 2) = VCC

Parameters	Description	Test Conditions		Min	Max	Units
tACC	Address to Output Delay	Output Load: 1 TTL gate and CL = 100pF Input Rise and Fall Times: ≤20ns Input Pulse Levels: 0.8V to 2.2V Timing Measurement Reference Level: Inputs: 1V and 2V Outputs: 0.8V and 2V	$\overline{CE}$ = $\overline{OE}$ = VIL		450	ns
tCE	$\overline{CE}$ to Output Delay		$\overline{OE}$ = VIL		450	ns
tOE	Output Enable to Output Delay		$\overline{CE}$ = VIL		120	ns
tDF	Output Enable High to Output Float		$\overline{CE}$ = VIL	0	100	ns
tOH	Output Hold from Addresses, $\overline{CE}$ or $\overline{OE}$ Whichever Occurred First		$\overline{CE}$ = $\overline{OE}$ = VIL	0		ns

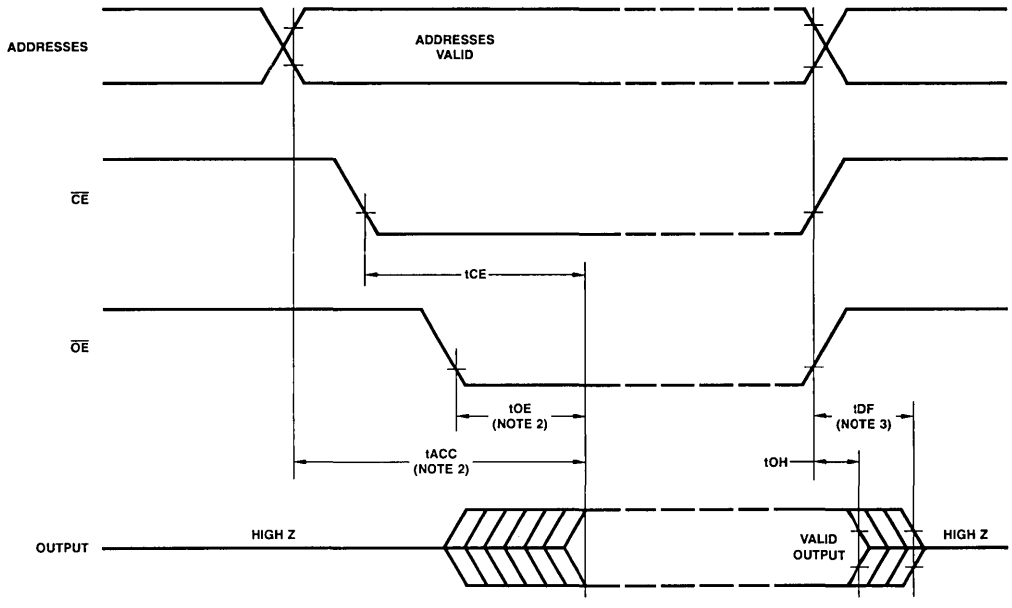
CAPACITANCE (Note 3)

T_A = +25°C, f = 1MHz

Parameters	Description	Test Conditions	Typ	Max	Units
CIN	Input Capacitance	VIN = 0V	4	6	pF
COUT	Output Capacitance	VOUT = 0V	8	12	pF

Notes: 1. VCC must be applied simultaneously or before VPP and removed simultaneously or after VPP.
2. VPP may be connected directly to VCC except during programming. The supply current would then be the sum of ICC and IPP1.
3. This parameter is only sampled and is not 100% tested.

AC WAVEFORMS (Note 1)



Notes: 1. VCC must be applied simultaneously or before VPP and removed simultaneously or after VPP.
2. $\overline{OE}$ may be delayed up to tACC − tOE after the falling edge of $\overline{CE}$ without impact on tACC.
3. tDF is specified from $\overline{OE}$ or $\overline{CE}$, whichever occurs first.

6

PROGRAM OPERATION

DC PROGRAMMING CHARACTERISTICS

T_A = +25°C ±5°C, VCC (Note 1) = 5V ±5%, VPP (Notes 1, 2) = 25V ±1V

Parameters	Description	Test Conditions	Min	Max	Units
ILI	Input Current	VIN = 5.25V/0.45V		10	μA
IPP1	VPP Supply Current	$\overline{CE}$/PGM = VIL		5	mA
IPP2	VPP Supply Current During Programming Pulse	$\overline{CE}$/PGM = VIH		30	mA
ICC	VCC Supply Current			100	mA
VIL	Input Low Level		−0.1	0.8	Volts
VIH	Input High Level		2.0	VCC+1	Volts

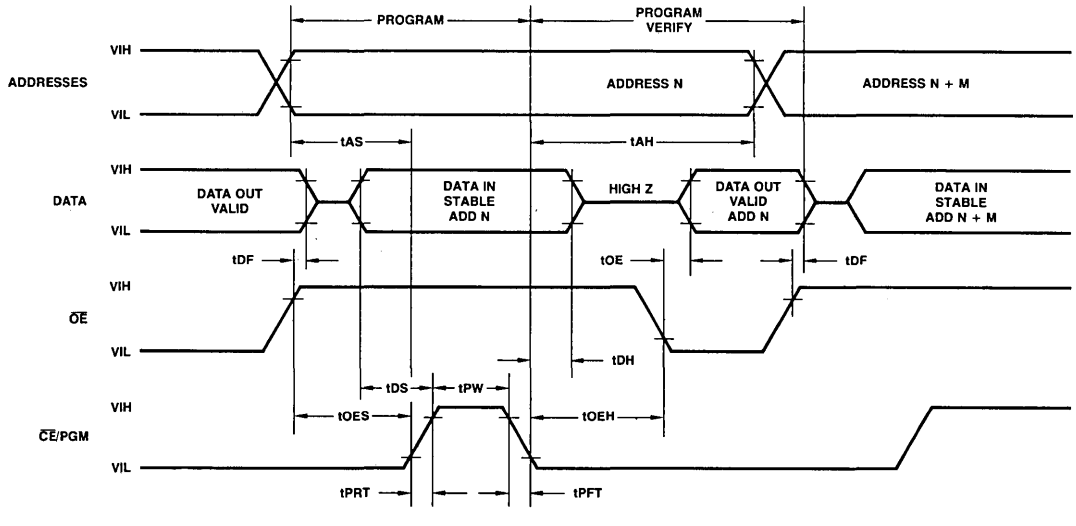
Notes: 1. VCC must be applied simultaneously or before VPP and removed simultaneously or after VPP.
2. VPP must not be greater than 26 volts including overshoot. Permanent device damage may occur if the device is taken out of or put into the socket when VPP = 25 volts is applied. Also, during $\overline{OE}$ = $\overline{CE}$/PGM = VIH, VPP must not be switched from 5 volts to 25 volts or vice versa.

AC PROGRAMMING CHARACTERISTICS

T_A = +25°C ±5°C, VCC (Note 1) = 5V ±5%, VPP (Notes 1, 2) = 25V ±1V

Parameters	Description	Test Conditions	Min	Max	Units
tAS	Address Set-up Time	Input tR and tF (10% to 90%) = 20ns Input Signal Levels = 0.8V to 2.2V Input Timing Reference Level = 1V and 2V Output Timing Reference Level = 0.8V and 2V	2		μs
tOES	Output Enable Set-up Time		2		μs
tDS	Data Set-up Time		2		μs
tAH	Address Hold Time		2		μs
tOEH	Output Enable Hold Time		2		μs
tDH	Data Hold Time		2		μs
tDF	Output Disable to Output Float Delay ($\overline{CE}$/PGM = VIL)		0	120	ns
tOE	Output Enable to Output Delay ($\overline{CE}$/PGM = VIL)		–	120	ns
tPW	Program Pulse Width		45	55	ms
tPRT	Program Pulse Rise Time		5	–	ns
tPFT	Program Pulse Fall Time		5	–	ns

PROGRAMMING WAVEFORMS



MOS-202

ERASING THE Am2716

In order to clear all locations of their programmed contents, it is necessary to expose the Am2716 to an ultraviolet light source. A dosage of 15 Wseconds/cm^2 is required to completely erase an Am2716. This dosage can be obtained by exposure to an ultraviolet lamp [(wavelength of 2537 Angstroms (Å)] with intensity of 12000μW/cm^2 for 15 to 20 minutes. The Am2716 should be about one inch from the source and all filters should be removed from the UV light source prior to erasure.

It is important to note that the Am2716, and similar devices, will erase with light sources having wavelengths shorter than 4000 Angstroms. Although erasure times will be much longer than with UV sources at 2537Å, nevertheless the exposure to florescent light and sunlight will eventually erase the Am2716, and exposure to them should be prevented to realize maximum system reliability. If used in such an environment, the package windows should be covered by an opaque label or substance.

PROGRAMMING THE Am2716

Upon delivery, or after each erasure the Am2716 has all 16384 bits in the "1", or high state. "0s" are loaded into the Am2716 through the procedure of programming.

The programming mode is entered when +25V is applied to the VPP pin and when $\overline{OE}$ is at VIH. The address to be programmed is applied to the proper address pins. 8-bit patterns are placed on the respective data output pins. The voltage levels should be standard TTL levels. When both the address and data are stable, a 50msec, TTL high level pulse is applied to the $\overline{CE}$/PGM input to accomplish the programming.

The procedure can be done manually, address by address, randomly, or automatically via the proper circuitry. All that is required is that one 50msec program pulse be applied at each address to be programmed. It is necessary that this program pulse width not exceed 55msec. Therefore, applying a DC level to the $\overline{CE}$/PGM input is prohibited when programming.

READ MODE

The Am2716 has two control functions, both of which must be logically satisfied in order to obtain data at the outputs. Chip Enable ($\overline{CE}$) is the power control and should be used for device selection. Output Enable ($\overline{OE}$) is the output control and should be used to gate data to the output pins, independent of device selection. Assuming that addresses are stable, address access time (tACC) is equal to the delay from $\overline{CE}$ to output (tCE). Data is available at the outputs 120ns (tOE) after the falling edge of $\overline{OE}$, assuming that $\overline{CE}$ has been low and addresses have been stable for at least tACC − tOE.

STANDBY MODE

The Am2716 has a standby mode which reduces the active power dissipation by 75%, from 525mW to 132mW. The Am2716 is placed in the standby mode by applying a TTL high signal to the $\overline{CE}$ input. When in standby mode, the outputs are in a high impedance state, independent of the $\overline{OE}$ input.

OUTPUT OR-TIEING

To accommodate multiple memory connections, a 2-line control function is provided to allow for:

1. Low memory power dissipation
2. Assurance that output bus contention will not occur.

It is recommended that $\overline{CE}$ be decoded and used as the primary device selecting function, while $\overline{OE}$ be made a common connection to all devices in the array and connected to the READ line from the system control bus. This assures that all deselected memory devices are in their low-power standby mode and that the output pins are only active when data is desired from a particular memory device.

PROGRAM INHIBIT

Programming of multiple Am2716s in parallel with different data is also easily accomplished. Except for $\overline{CE}$/PGM, all like inputs (including $\overline{OE}$) of the parallel Am2716s may be common. A TTL level program pulse applied to an Am2716's $\overline{CE}$/PGM input with VPP at 25V will program that Am2716. A low level $\overline{CE}$/PGM input inhibits the other Am2716 from being programmed.

PROGRAM VERIFY

A verify should be performed on the programmed bits to determine that they were correctly programmed. The verify may be performed with VPP at 25V. Except during programming and program verify, VPP must be at 5V.

Am9218/8316E

2048 x 8 Read Only Memory

DISTINCTIVE CHARACTERISTICS

- 2048 x 8 organization
- Plug-in replacement for 8316E
- Access times as fast as 350 ns
- Fully capacitive inputs — simplified driving
- 3 fully programmable Chip Selects — increased flexibility
- Logic voltage levels compatible with TTL
- Three-state output buffers — simplified expansion
- Drives two full TTL loads
- Single supply voltage — +5.0V
- Low power dissipation
- N-channel silicon gate MOS technology
- 100% MIL-STD-883 reliability assurance testing

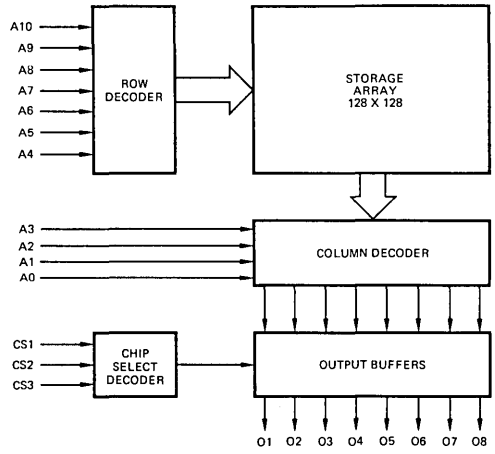
FUNCTIONAL DESCRIPTION

The Am9218 devices are high performance, 16384-bit, static, mask programmed, read only memories. Each memory is implemented as 2048 words by 8 bits per word. This organization simplifies the design of small memory systems and permits incremental memory sizes as small as 2048 words. The fast access times provided allow the ROM to service high performance microcomputer applications without stalling the processor.

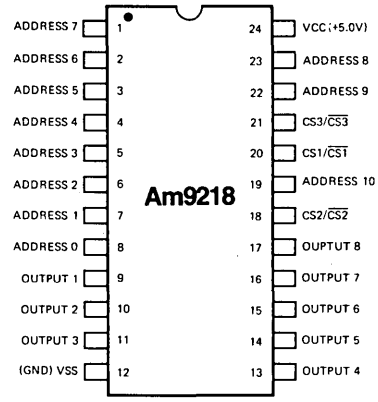
Three programmable Chip Select input signals are provided to control the output buffers. Each Chip Select polarity may be specified by the customer thus allowing the addressing of 8 memory chips without external gating. The outputs of unselected chips are turned off and assume a high impedance state. This permits wire-ORing with additional Am9218 devices and other three-state components.

These memories are fully static and require no clock signals of any kind. A selected chip will output data from a location specified by whatever address is present on the address input lines. Input and output voltage levels are compatible with TTL specifications.

BLOCK DIAGRAM



CONNECTION DIAGRAM



Top View
Pin 1 is marked for orientation.

ORDERING INFORMATION

Package Type	Ambient Temperature Specifications	Access Time	
		450ns	350ns
Molded	$0°C \leq T_A \leq 70°C$	AM9218BPC P8316E	AM9218CPC
Cerdip	$0°C \leq T_A \leq 70°C$	AM9218BCC	AM9218CCC
Side-Brazed Ceramic	$0°C \leq T_A \leq 70°C$	AM9218BDC C8316E	AM9218CDC
	$-55°C \leq T_A \leq +125°C$	AM9218BDM	

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	+7.0V
DC Voltage Applied to Outputs	−0.5V to +7.0V
DC Input Voltage	−0.5V to +7.0V
Power Dissipation	1.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

ELECTRICAL CHARACTERISTICS

Am9218BDC, Am9218CDC, C8316A: $T_A = 0°C$ to $+70°C$; VCC = 5.0V ±5%

Parameters	Description	Test Conditions		Am9218XDC Min.	Am9218XDC Max.	C8316E Min.	C8316E Max.	Units
VOH	Output HIGH Voltage	9218	IOH = −200μA	2.4				Volts
		8316E	IOH = −100μA			2.4		
VOL	Output LOW Voltage	9218	IOL = 3.2mA		0.4			Volts
		8316E	IOL = 2.1mA				0.4	
VIH	Input HIGH Voltage			2.0	VCC + 1.0	2.0	VCC + 1.0	Volts
VIL	Input LOW Voltage			−0.5	0.8	−0.5	0.8	Volts
ILO	Output Leakage Current	Chip Disabled			10		10	μA
ILI	Input Leakage Current				10		10	μA
ICC	VCC Supply Current				70		95	mA

ELECTRICAL CHARACTERISTICS

Am9218BDM: $T_A = -55°C$ to $+125°C$; VCC = 5.0V ±10%

Parameters	Description	Test Conditions	Am9218B Min.	Am9218B Max.	Units
VOH	Output HIGH Voltage	IOH = −200μA	2.2		Volts
VOL	Output LOW Voltage	IOL = 3.2mA		0.45	Volts
VIH	Input HIGH Voltage		2.0	VCC + 1.0	Volts
VIL	Input LOW Voltage		−0.5	0.8	Volts
ILO	Output Leakage Current	Chip Disabled		10	μA
ILI	Input Leakage Current			10	μA
ICC	VCC Supply Current			80	mA

SWITCHING CHARACTERISTICS OVER OPERATING RANGE

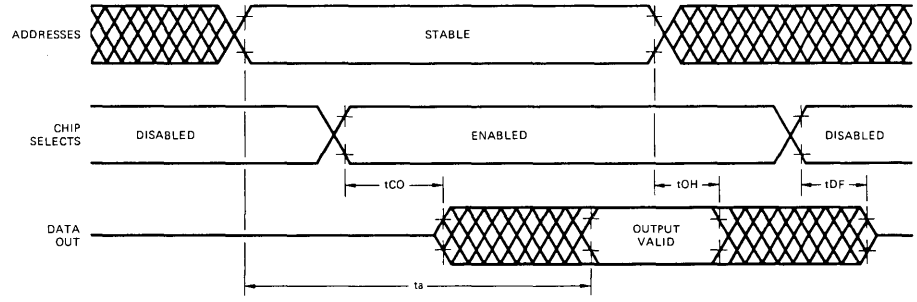
Am9218XDC/C8316E: $T_A = 0°C$ to $+70°C$; VCC = 5.0V ± 5%

Am9218BDM: $T_A = -55°C$ to $+125°C$; VCC = 5.0V ± 10%

Parameters	Description	Test Conditions	Am9218B Min.	Am9218B Max.	Am9218C Min.	Am9218C Max.	8316E Min.	8316E Max.	Units
ta	Address to Output Access Time	tr = tf = 20ns; Output Load: one standard TTL gate plus 100pF (Note 1)		450		350		450	ns
tCO	Chip Select to Output ON Delay			150		130		250	ns
tOH	Previous Read Data Valid with Respect to Address Change		20		20		–		ns
tDF	Chip Select to Output OFF Delay			150		130		250	ns
CI	Input Capacitance	$T_A = 25°C$, f = 1.0MHz; All pins at 0V		7.0		7.0		7.0	pF
CO	Output Capacitance			7.0		7.0		7.0	pF

Notes: 1. Timing reference levels: High = 2.0V, Low = 0.8V.

SWITCHING WAVEFORMS



PROGRAMMING INSTRUCTIONS

CUSTOM PATTERN ORDERING INFORMATION

The Am9218 is programmed from punched cards, card coding forms or paper tape in card image format as shown below.
Logic "1" = a more positive voltage (normally +5.0 V)
Logic "0" = a more negative voltage (normally 0 V)

FIRST CARD

Column Number	Description
10 thru 29	Customer Name
32 thru 37	Total number of "1's" contained in the data. This is optional and should be left blank if not used.
50 thru 62	8316E or 9218
65 thru 72	Optional information

SECOND CARD

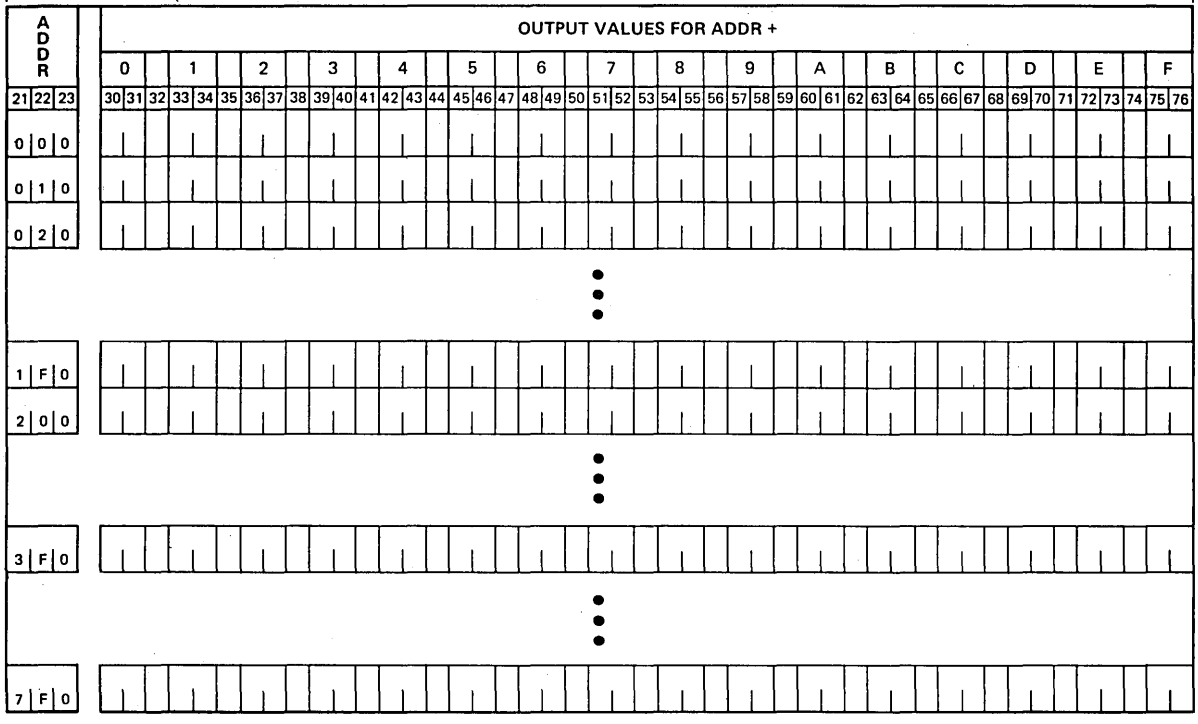
Column Number	Description
29	CS3 input required to select chip (0 or 1)
31	CS2 input required to select chip (0 or 1)
33	CS1 input required to select chip (0 or 1)

Two options are provided for entering the data pattern with the remaining cards.

OPTION 1 is the Binary Option where the address and data are presented in binary form on the basis of one word per card. With this option 2048 data cards are required.

Column Number	
10, 12, 14, 16, 18 20, 22, 24, 26, 28, 30	Address input pattern with the most significant bit (A10) in column 10 and the least significant bit (A0) in column 30.
40, 42, 44, 46, 48, 50, 52, 54	Output pattern with the most significant bit (O8) in column 40 and the least significant bit (O1) in column 54.
73 thru 80	Coding these columns is not essential and may be used for card identification purposes.

OPTION 2 is the Hexadecimal Option and is a much more compact way of presenting the data. This format requires only 128 data cards. Each data card contains the 8-bit output information for 16 storage locations in the memory. The address indicated in columns 21, 22 and 23 is the address of the data presented in columns 30 and 31. Addresses for successive data are assumed to be in incremental ascending order from the initial address. Since the address in columns 21, 22 and 23 always points only to the first data on the card, column 23 is always zero. Columns 21 and 22 take all hex values from 00 through 7F: 128 cards in all. Data in entered in hex values and may be any combination of 8 bits, that is, hex values from 00 through FF.



Am9232 • Am9233

4096 X 8 Read Only Memory

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

- 4096 X 8 organization
- No clocks or refresh required
- Access time selected to 300ns
- Fully capacitive inputs – simplified driving
- 2 mask programmable chip selects – increased flexibility
- Logic voltage levels compatible with TTL
- Three state output buffers – simplified expansion
- Drives two TTL loads
- Single +5 volt power supply
- Two different pinouts for universal application
- Low power dissipation
- 100% MIL-STD-883 reliability assurance testing
- Non-connect option on chip selects.

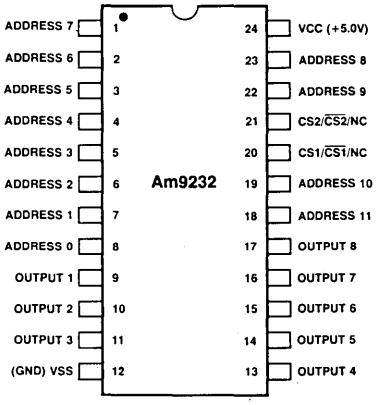
FUNCTIONAL DESCRIPTION

The Am9232/33 devices are high performance, 32,768-bit, static, mask programmed, read only memories. Each memory is implemented as 4096 words by 8 bits per word. This organization simplifies the design of small memory systems and permits incremental memory sizes of 4096 words. The fast access times provided allow the ROM to service high performance microcomputer applications without stalling the processor.

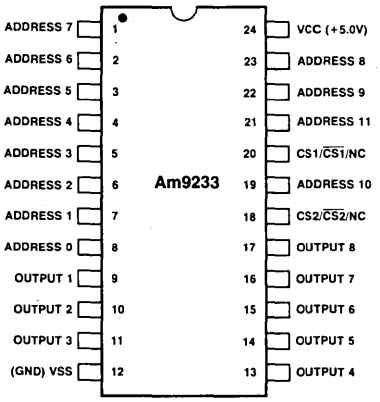
Two programmable Chip Select input signals are provided to control the output buffers. Each Chip Select polarity may be specified by the customer thus allowing the addressing of 4 memory chips without external gating. The outputs of unselected chips are turned off and assume a high impedance state. This permits wire-ORing with additional Am9232/33 devices and other three-state components.

These memories are fully static and require no clock signals of any kind. A selected chip will output data from a location specified by the address present on the address input lines. Input and output voltage levels are compatible with TTL specifications.

CONNECTION DIAGRAMS
Top Views



MOS-103



MOS-104

Note: Pin 1 is marked for orientation.

ORDERING INFORMATION

Package Type	Ambient Temperature Specifications	Access Time	
		450ns	300ns
Molded	$0°C \leq T_A \leq +70°C$	AM9232/33BPC	AM9232/33CPC
Cerdip	$0°C \leq T_A \leq +70°C$	AM9232/33BCC	AM9232/33CCC
Side-Brazed Ceramic	$-55°C \leq T_A \leq +125°C$	AM9232/33BDM	
	$0°C \leq T_A \leq +70°C$	AM9232/33BDC	AM9232/33CDC

MAXIMUM RATINGS beyond which the useful life may be impaired

Storage Temperature	−65°C to +150°C
Ambient Temperature Under Bias	−55°C to +125°C
VCC with Respect to VSS	+7.0V
DC Voltage Applied to Outputs	−0.5V to +7.0V
DC Input Voltage	−0.5V to +7.0V
Power Dissipation (Package Limitation)	1.0W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested nevertheless, that conventional precautions be observed during storage, handling and use in order to avoid exposure to excessive voltages.

OPERATING RANGE

Part Number	Ambient Temperature	VCC	VSS
Am9232DC/PC/CC	$0°C \leq T_A \leq +70°C$	+5.0V ±5%	0V
Am9232/33DM	$-55°C \leq T_A \leq +125°C$	+5.0V ±10%	0V

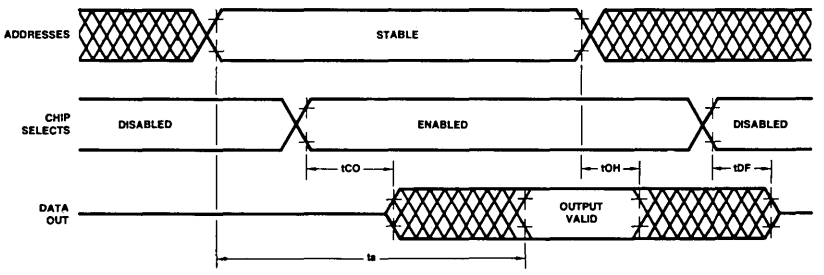
ELECTRICAL CHARACTERISTICS over operating range

Parameter	Description	Test Conditions		Am9232/Am9233 Min.	Max.	Unit
VOH	Output HIGH Voltage	IOH = −200μA	VCC = 4.75	2.4		Volts
			VCC = 4.50	2.2		
VOL	Output LOW Voltage	IOL = 3.2mA			0.4	Volts
VIH	Input HIGH Voltage			2.0	VCC+1.0	Volts
VIL	Input LOW Voltage			−0.5	0.8	Volts
I_{LI}	Input Load Current	$VSS \leq VI \leq VCC$			10	μA
IOZ	Output Leakage Current	$VSS \leq VO \leq VCC$ Chip Disabled	+70°C		10	μA
			+125°C (DM)		50	
ICC	VCC Supply Current		0°C		80	mA
			−55°C (DM)		100	
CI	Input Capacitance	T_A = 25°C, f = 1.0MHz All pins at 0V			7.0	pF
CO	Output Capacitance				7.0	pF

SWITCHING CHARACTERISTICS over operating range

Parameter	Description	Test Conditions	Am9232/33B Min.	Am9232/33B Max.	Am9232/33C Min.	Am9232/33C Max.	Unit
ta	Address to Output Access Time	tr = tf = 20ns Output Load: one standard TTL gate plus 100pF (Note 1)		450		300	ns
tCO	Chip Select to Output ON Delay			150		120	ns
tOH	Previous Read Data Valid with Respect to Address Change		20		20		ns
tDF	Chip Select to Output OFF Delay			150		120	ns

SWITCHING WAVEFORMS



MOS-105

PROGRAMMING INTRUCTIONS

CUSTOM PATTERN ORDERING INFORMATION

The Am9232 is programmed from punched cards, card coding forms or paper tape in card image format as shown below.

Logic "1" = a more positive voltage (normally +5.0V)
Logic "0" = a more negative voltage (normally 0V)

FIRST CARD

Column Number	Description
10 thru 29	Customer Name
32 thru 37	Total number of "1's" contained in the data. This is optional and should be left blank if not used.
50 thru 62	9232 or 9233
65 thru 72	Optional information

SECOND CARD

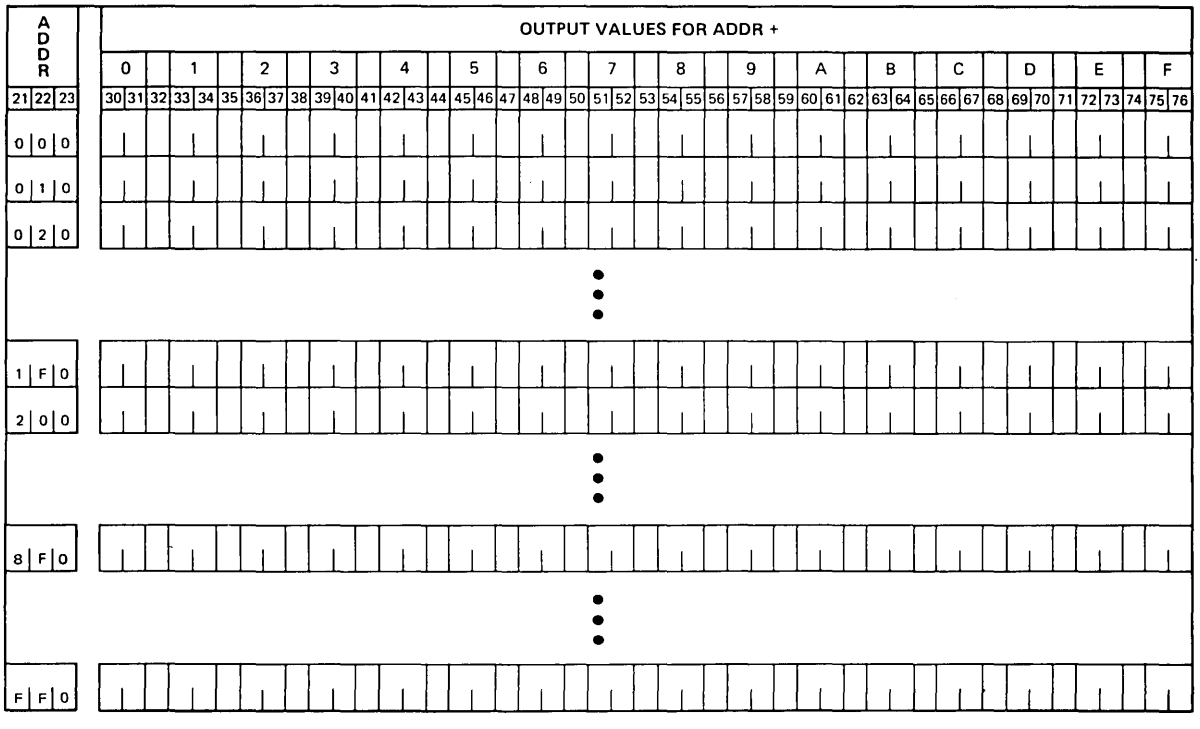
Column Number	Description
31	CS2 input required to select chip (0 or 1); If CS2 = NC, column 31 = 2.
33	CS1 input required to select chip (0 or 1); If CS1 = NC, column 33 = 2.

Two options are provided for entering the data pattern with the remaining cards.

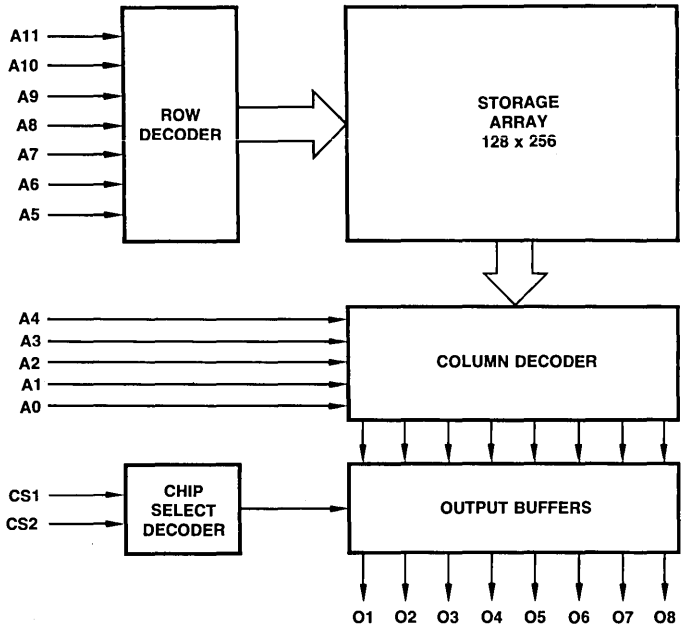
OPTION 1 is the Binary Option where the address and data are presented in binary form on the basis of one word per card. With this option 4096 data cards are required.

Column Number	
8, 10, 12, 14, 16, 18 20, 22, 24, 26, 28, 30	Address input pattern with the most significant bit (A11) in column 8 and the least significant bit (A0) in column 30.
40, 42, 44, 46, 48 50, 52, 54	Output pattern with the most significant bit (O8) in column 40 and the least significant bit (O1) in column 54.
73 thru 80	Coding these columns is not essential and may be used for card identification purposes.

OPTION 2 is the Hexadecimal Option and is a much more compact way of presenting the data. This format requires only 256 data cards. Each data card contains the 8-bit output information for 16 storage locations in the memory. The address indicated in columns 21, 22 and 23 is the address of the data presented in columns 30 and 31. Addresses for successive data are assumed to be in incremental ascending order from the initial address. Since the address in columns 21, 22 and 23 always points only to the first data on the card, column 23 is always zero. Columns 21 and 22 take all hex values from 00 through FF:256 cards in all. Data is entered in hex values and may be any combination of 8 bits, that is, hex values from 00 through FF.



Am9232/Am9233
BLOCK DIAGRAM



MOS-106

Am27S28 • Am27S29

4096-Bit Generic Series Bipolar PROM

DISTINCTIVE CHARACTERISTICS

- High Speed – 55ns max commercial range access time
- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S28 and Am27S29 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test rows are pre-programmed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliabilty. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

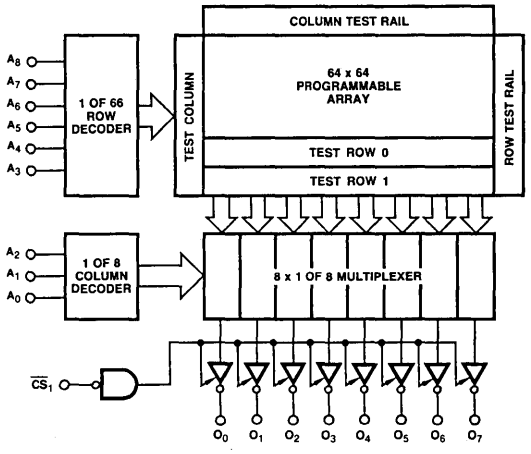
ORDERING INFORMATION

Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0°C to +75°C	AM27S28DC
Hermetic DIP	−55°C to +125°C	AM27S28DM
	Three-State Outputs	
Hermetic DIP	0°C to +75°C	AM27S29DC
Hermetic DIP	−55°C to +125°C	AM27S29DM

FUNCTIONAL DESCRIPTION

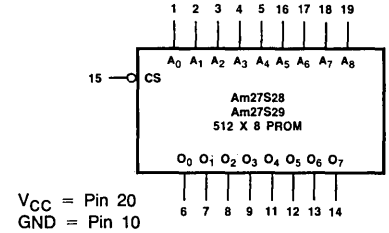
The Am27S28 and Am27S29 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 512 x 8 configuration, they are available in both open collector Am27S28 and three-state Am27S29 output versions. After programming, stored information is read on outputs O_0-O_7 by applying unique binary addresses to A_0-A_8 and holding the chip select input, $\overline{CS}$, at a logic LOW. If the chip select input goes to a logic HIGH, O_0-O_7 go to the off or high impedance state.

BLOCK DIAGRAM



BPM-083

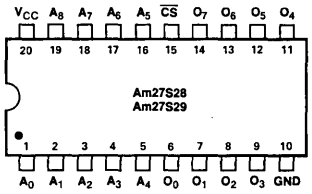
LOGIC SYMBOL



V_{CC} = Pin 20
GND = Pin 10

BPM-084

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BPM-085

6

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential (Pin 20 to Pin 10) Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5V to +5.5V
DC Input Current	−30mA to +5mA

OPERATING RANGE

COM'L	Am27S28XC, Am27S29XC	T_A = 0°C to +75°C	V_{CC} = 5.0V ±5%
MIL	Am27S28XM, Am27S29XM	T_A = −55°C to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions	Min.	Typ. (Note 1)	Max.	Units
V_{OH} (Am27S29 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}			0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs	2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs			0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V		−0.010	−0.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V			25	µA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V			1.0	mA
I_{SC} (Am27S29 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)	−20	−40	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.		105	160	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA			−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX., $V_{\overline{CS}}$ = 2.4V; Am27S29 only; V_O = 4.5V			40	µA
		V_{CC} = MAX., $V_{\overline{CS}}$ = 2.4V; Am27S29 only; V_O = 2.4V			40	µA
		V_{CC} = MAX., $V_{\overline{CS}}$ = 2.4V; Am27S29 only; V_O = 0.4V			−40	µA
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1 MHz (Note 3)		4		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1 MHz (Note 3)		8		pF

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but periodically sampled.

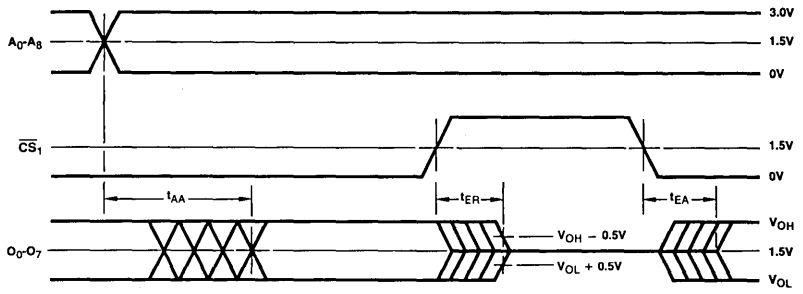
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	Max COM'L	Max MIL	Units
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	35	55	70	ns
t_{EA}	Enable Access Time		15	25	30	ns
t_{ER}	Enable Recovery Time		15	25	30	ns

Notes:
1. t_{AA} is tested with switch S_1 closed and C_L = 30pF.
2. For open collector outputs, t_{EA} and t_{ER} are tested with S_1 closed to the 1.5V output level. C_L = 30pF.
3. For three state outputs, t_{EA} is tested with C_L = 30pF to the 1.5V level; S_1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. t_{ER} is tested with C_L = 5pF. HIGH to high impedance tests are made with S_1 open to an output voltage of V_{OH} – 0.5V; LOW to high impedance tests are made with S_1 closed to the V_{OL} + 0.5V level.

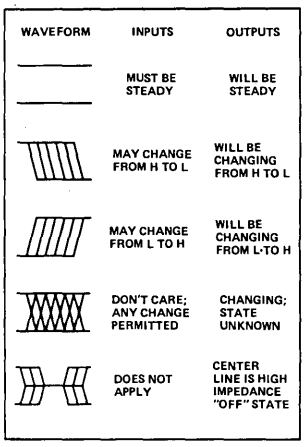
SWITCHING WAVEFORMS



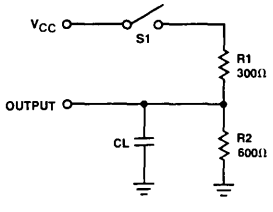
Note: Level on output while CS is HIGH is determined externally.

BPM-086

KEY TO TIMING DIAGRAM



AC TEST LOAD

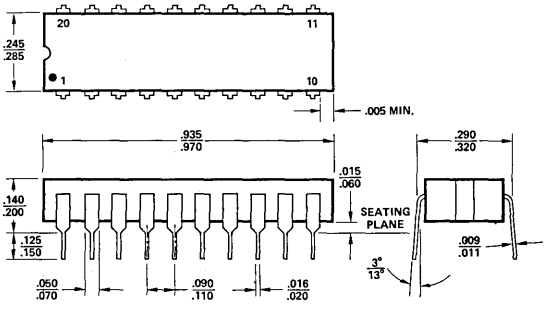


BPM-087

PHYSICAL DIMENSIONS

Dual-In-Line

20-Pin Hermetic



6

PROGRAMMING

The Am27S28 and Am27S29 are manufactured with a conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to the memory output after the $\overline{CS}$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip is enabled and the output level is sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be continuously applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

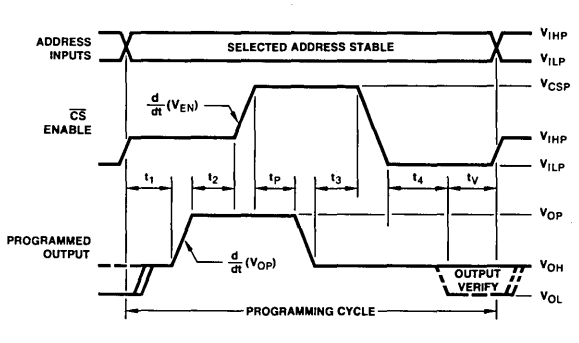
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	Volts
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	Volts
V_{ILP}	Input LOW Level During Programming	0.0	0.45	Volts
V_{CSP}	$\overline{CS}$ Voltage During Programming	14.5	15.5	Volts
V_{OP}	Output Voltage During Programming	19.5	20.5	Volts
V_{ONP}	Voltage on Outputs Not to be Programmed	0.0	V_{CCP}+0.3	Volts
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{EN})/dt$	Rate of $\overline{CS}$ Voltage Change	100	1000	v/μsec
t_p	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

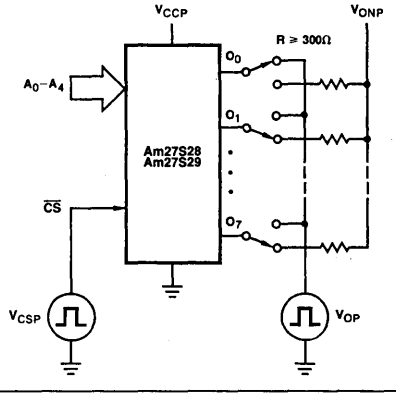
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1 through t_4 must be greater than 100ns; maximum delays of 1μsec are recommended to minimize heating during programming.
3. During t_V, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-088

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-089

PROGRAMMING EQUIPMENT

Generic programming boards and device adapters are available from the sources listed below. In each case, the programming boards are used in these manufacturer's automatic programmers to program all AMD generic series bipolar PROMs; individual adapters are required for each basic part type in the series.

SOURCE AND LOCATION	Data I/O Corp. P.O. Box 308 Issaquah, Wash. 98027	Pro-Log Corp. 2411 Garden Road Monterey, Ca. 93940
PROGRAMMER MODEL(S)	Model 5, 7 and 9	M900 and M920
AMD GENERIC BIPOLAR PROM PERSONALITY BOARD	909-1286-1	PM9058
Am27S28 • Am27S29 ADAPTERS AND CONFIGURATOR	715-1413	PA20-4 and 512 x 8 (L)

OBTAINING PROGRAMMED UNITS

Programmed devices may be purchased from your distributor or Advanced Micro Devices. The program data should be submitted in the form of a punched paper tape and must be accompanied by a written truth table. The punched tape can be delivered with your order or may be transmitted over a TWX machine or time-sharing terminal. ASCII BPNF is our preferred paper tape format.

Truth tables are also acceptable, however, much less desirable especially for larger density PROMs. Submission of a truth table requires the generation of a punched paper tape at the distributor or factory resulting in longer lead times, greater possibility of error, and higher cost.

ASCII BPNF

An example of an ASCII tape in the BPNF format is shown below. They can be punched on any Teletype® or on a TWX or Telex machine. The format chosen provides relatively good error detection. Paper tapes must consist of:

1. A leader of at least 25 rubouts.
2. The data patterns for all 512 words, starting with word 0, in the following format:
   a. Any characters, including carriage return and line feed, except "B".
   b. The letter "B", indicating the beginning of the data word.
   c. A sequence of eight Ps or Ns, starting with output O_7.
   d. The letter "F", indicating the finish of the data word.
   e. Any text, including carriage return and line feed, except the letter "B".
3. A trailer of at least 25 rubouts.

A P is a HIGH logic level = 2.4 volts.
An N is a LOW logic level = 0.4 volts.

A convenient pattern to use for the data words is to prefix the word (or every few words) with the word number, then type the data word, then a comment, then carriage return and line feed as shown below. There must be no characters between the B and the F except for the eight Ps and Ns. If an error is made in a word, the entire word must be cancelled with rubouts back to the letter B, then the word re-typed beginning with the B.

When TWXing your tape, be sure the tape is in even parity. Parity is not necessary if the tape is mailed.

TYPICAL PAPER TAPE FORMAT

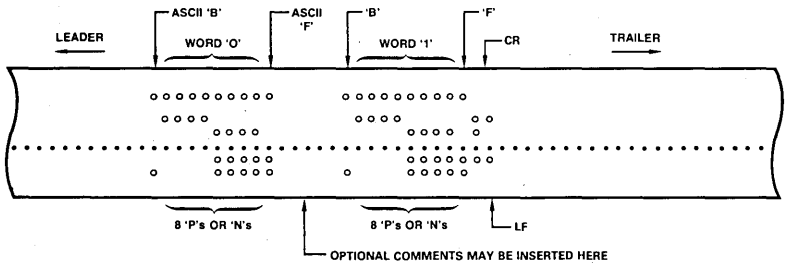
```
ØØØ   BPNPPNNNPF   WORD ZERO (R) (L)
      BPPPPPPNNF   COMMENT FIELD (R) (L)
ØØ2   BNNNPPPPNF   ANY (R) (L)
      BNNNNNNNNF   TEXT (R) (L)
ØØ4   BPNNNNNNPF   CAN (R) (L)
      BNPPNPPNNF   GO (R) (L)
ØØ6   BPNNPPPNNF   HERE (R) (L)
 :    :::::::::::   :
511   BNNNNPPPNF   END (R) (L)
```

(R) = CARRIAGE RETURN
(L) = LINE FEED

RESULTING DEVICE TRUTH TABLE ($\overline{CS}$ LOW)

A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
L	L	L	L	L	L	L	L	L	H	L	H	H	L	L	L	H
L	L	L	L	L	L	L	L	H	H	H	H	H	H	H	L	L
L	L	L	L	L	L	L	H	L	L	L	L	H	H	H	H	L
L	L	L	L	L	L	L	H	H	L	L	L	L	L	L	L	L
L	L	L	L	L	L	H	L	L	H	L	L	L	L	L	L	H
L	L	L	L	L	L	H	L	H	L	H	H	L	H	H	L	L
L	L	L	L	L	L	H	H	L	H	L	L	H	H	H	L	L
				⋮								⋮				
H	H	H	H	H	H	H	H	H	L	L	L	L	H	H	H	L

ASCII PAPER TAPE



BPM-017

6

Am27S30 • Am27S31

4096-Bit Generic Series Bipolar PROM

DISTINCTIVE CHARACTERISTICS

- High Speed – 55 ns max commercial range access time
- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S30 and Am27S31 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test rows are preprogrammed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliabilty. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

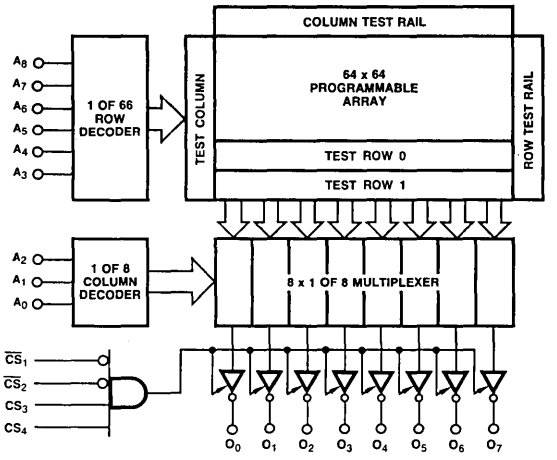
ORDERING INFORMATION

Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0°C to +75°C	AM27S30DC
Hermetic DIP	−55°C to +125°C	Am27S30DM
	Three-State Outputs	
Hermetic DIP	0°C to +75°C	AM27S31DC
Hermetic DIP	−55°C to +125°C	AM27S31DM

FUNCTIONAL DESCRIPTION

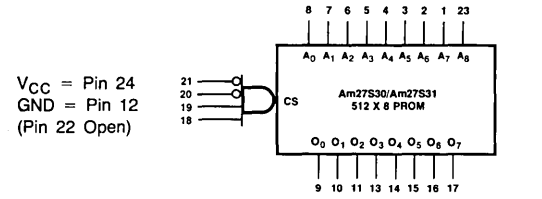
The Am27S30 and Am27S31 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 512 x 8 configuration, they are available in both open collector Am27S30 and three-state Am27S31 output versions. After programming, stored information is read on outputs O_0-O_7 by applying unique binary addresses to A_0-A_8 and holding $\overline{CS}_1$ and $\overline{CS}_2$ LOW and CS_3 and CS_4 HIGH. All other valid input conditions on $\overline{CS}_1$, $\overline{CS}_2$, CS_3 and CS_4 place O_0-O_7 into the OFF or high impedance state.

BLOCK DIAGRAM



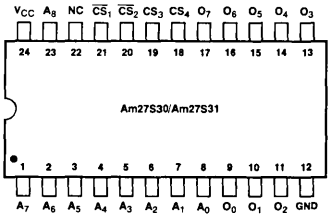
BPM-114

LOGIC SYMBOL



BPM-115

CONNECTION DIAGRAM
Top View



Note: Pin 1 is marked for orientation.

BPM-116

MAXIMUM RATINGS (Above which the useful life may be impaired)	
Storage Temperature	−65°C to +150°C
Temperature (Ambient) Under Bias	−55°C to +125°C
Supply Voltage to Ground Potential (Pin 24 to Pin 12) Continuous	−0.5V to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5V to +5.5V
DC Input Current	−30mA to +5mA

OPERATING RANGE

COM L	Am27S30XC, Am27S31XC	T_A = 0°C to +75°C	V_{CC} = 5.0V ±5%
MIL	Am27S30XM, Am27S31XM	T_A = −55°C to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions	Min.	Typ. (Note 1)	Max.	Units
V_{OH} (Am27S31 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}			0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs	2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs			0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V		−0.010	−0.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V			25	µA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V			1.0	mA
I_{SC} (Am27S31 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)	−20	−40	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.		115	175	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA			−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX, $V_{\overline{CS1}}$ = 2.4V; Am27S31 only; V_O = 4.5V			40	µA
		V_{CC} = MAX, $V_{\overline{CS1}}$ = 2.4V; Am27S31 only; V_O = 2.4V			40	µA
		V_{CC} = MAX, $V_{\overline{CS1}}$ = 2.4V; Am27S31 only; V_O = 0.4V			−40	µA
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1 MHz (Note 3)		4		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1 MHz (Note 3)		8		pF

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but periodically sampled.

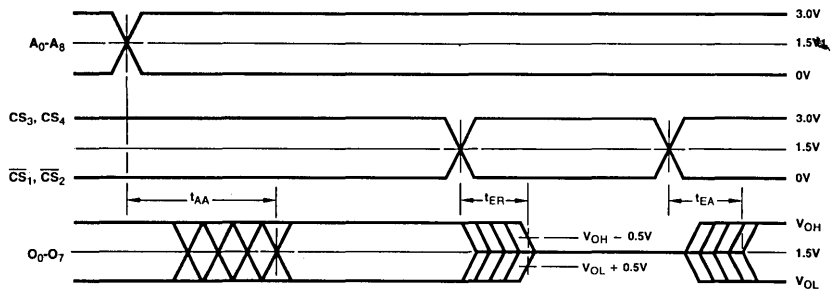
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	Max COM'L	Max MIL	Units
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	35	55	70	ns
t_{EA}	Enable Access Time		15	25	30	ns
t_{ER}	Enable Recovery Time		15	25	30	ns

Notes: 1. t_{AA} is tested with switch S_1 closed and C_L = 30pF.
2. For open collector outputs, t_{EA} and t_{ER} are tested with S_1 closed to the 1.5V output level. C_L = 30pF.
3. For three state outputs, t_{EA} is tested with C_L = 30pF to the 1.5V level; S_1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. t_{ER} is tested with C_L = 5pF. HIGH to high impedance tests are made with S_1 open to an output voltage of V_{OH} − 0.5V; LOW to high impedance tests are made with S_1 closed to the V_{OL} + 0.5V level.

SWITCHING WAVEFORMS



Note: Level on output while chip is disabled is determined externally.

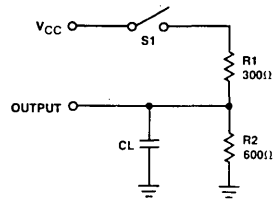
BPM-117

KEY TO TIMING DIAGRAM

WAVEFORM	INPUTS	OUTPUTS
	MUST BE STEADY	WILL BE STEADY
	MAY CHANGE FROM H TO L	WILL BE CHANGING FROM H TO L
	MAY CHANGE FROM L TO H	WILL BE CHANGING FROM L TO H

WAVEFORM	INPUTS	OUTPUTS
	DON'T CARE; ANY CHANGE PERMITTED	CHANGING; STATE UNKNOWN
	DOES NOT APPLY	CENTER LINE IS HIGH IMPEDANCE "OFF" STATE

AC TEST LOAD



BPM-118

PROGRAMMING

The Am27S30 and Am27S31 are manufactured with a conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to the memory output after the $\overline{CS}_1$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}_1$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip is enabled and the output level is sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}_1$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be continuously applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

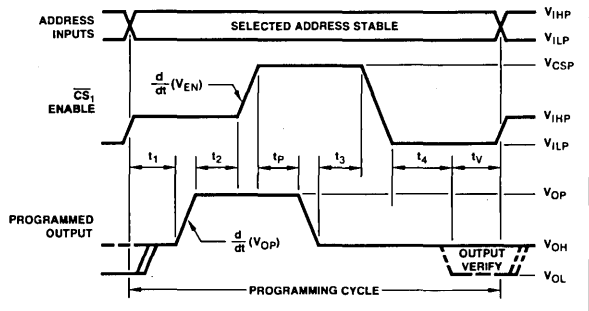
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	Volts
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	Volts
V_{ILP}	Input LOW Level During Programming	0.0	0.45	Volts
V_{CSP}	$\overline{CS}_1$ Voltage During Programming	14.5	15.5	Volts
V_{OP}	Output Voltage During Programming	19.5	20.5	Volts
V_{ONP}	Voltage on Outputs Not to be Programmed	0.0	V_{CCP}+0.3	Volts
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{EN})/dt$	Rate of $\overline{CS}_1$ Voltage Change	100	1000	v/μsec
t_p	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

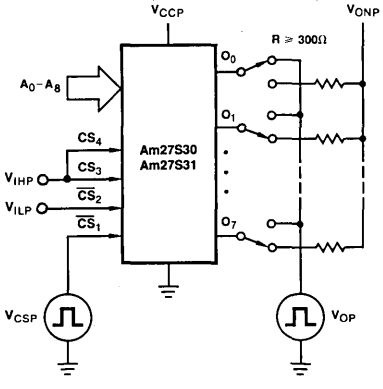
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1 through t_4 must be greater than 100ns; maximum delays of 1μsec are recommended to minimize heating during programming.
3. During t_V, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-119

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-120

PROGRAMMING EQUIPMENT

Generic programming boards and device adapters are available from the sources listed below. In each case, the programming boards are used in these manufacturer's automatic programmers to program all AMD generic series bipolar PROMs; individual adapters are required for each basic part type in the series.

SOURCE AND LOCATION	Data I/O Corp. P.O. Box 308 Issaquah, Wash. 98027	Pro-Log Corp. 2411 Garden Road Monterey, Ca. 93940
PROGRAMMER MODEL(S)	Model 5, 7 and 9	M900 and M920
AMD GENERIC BIPOLAR PROM PERSONALITY BOARD	909-1286-1	PM9058
Am27S30 • Am27S31 ADAPTERS AND CONFIGURATOR	715-1545	PA24-13 and 512 x 8 (L)

OBTAINING PROGRAMMED UNITS

Programmed devices may be purchased from your distributor or Advanced Micro Devices. The program data should be submitted in the form of a punched paper tape and must be accompanied by a written truth table. The punched tape can be delivered with your order or may be transmitted over a TWX machine or time-sharing terminal. ASCII BPNF is our preferred paper tape format.

Truth tables are also acceptable, however, much less desirable especially for larger density PROMs. Submission of a truth table requires the generation of a punched paper tape at the distributor or factory resulting in longer lead times, greater possibility of error, and higher cost.

ASCII BPNF

An example of an ASCII tape in the BPNF format is shown below. They can be punched on any Teletype® or on a TWX or Telex machine. The format chosen provides relatively good error detection. Paper tapes must consist of:

1. A leader of at least 25 rubouts.
2. The data patterns for all 512 words, starting with word 0, in the following format:
   a. Any characters, including carriage return and line feed, except "B".
   b. The letter "B", indicating the beginning of the data word.
   c. A sequence of eight Ps or Ns, starting with output O_7.
   d. The letter "F", indicating the finish of the data word.
   e. Any text, including carriage return and line feed, except the letter "B".
3. A trailer of at least 25 rubouts.

A P is a HIGH logic level = 2.4 volts.
An N is a LOW logic level = 0.4 volts.

A convenient pattern to use for the data words is to prefix the word (or every few words) with the word number, then type the data word, then a comment, then carriage return and line feed as shown below. There must be no characters between the B and the F except for the eight Ps and Ns. If an error is made in a word, the entire word must be cancelled with rubouts back to the letter B, then the word re-typed beginning with the B.

When TWXing your tape, be sure the tape is in even parity. Parity is not necessary if the tape is mailed.

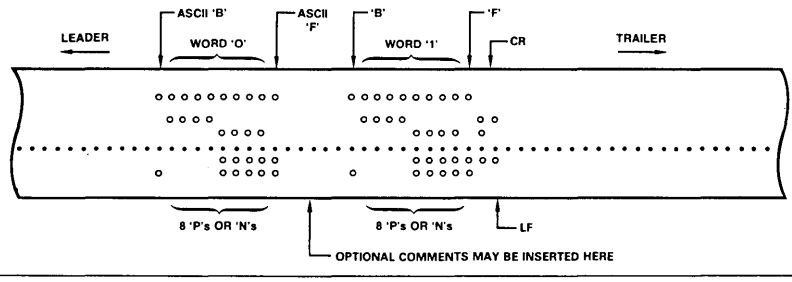
TYPICAL PAPER TAPE FORMAT

```
ØØØ   BPNPPNNNFF   WORD ZERO (R) (L)
      BPPPPPPNNF   COMMENT FIELD (R) (L)
Ø02   BNNNPPPPNF   ANY (R) (L)
      BNNNNNNNNF   TEXT (R) (L)
Ø04   BPNNNNNNFF   CAN (R) (L)
      BNPPNPPNNF   GO (R) (L)
Ø06   BPNNPPPNNF   HERE (R) (L)
 :    :::::::::::    :
511   BNNNNPPPNF   END (R) (L)
(R) = CARRIAGE RETURN
(L) = LINE FEED
```

RESULTING DEVICE TRUTH TABLE
($\overline{CS}_1$ AND $\overline{CS}_2$ LOW, CS_3 AND CS_4 HIGH)

A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
L	L	L	L	L	L	L	L	L	H	L	H	H	L	L	L	H
L	L	L	L	L	L	L	L	H	H	H	H	H	H	H	L	L
L	L	L	L	L	L	L	H	L	L	L	L	H	H	H	H	L
L	L	L	L	L	L	L	H	H	L	L	L	L	L	L	L	L
L	L	L	L	L	L	H	L	L	H	L	L	L	L	L	L	H
L	L	L	L	L	L	H	L	H	L	H	H	L	H	H	L	L
L	L	L	L	L	L	H	H	L	H	L	L	H	H	H	L	L
				⋮								⋮				
H	H	H	H	H	H	H	H	H	L	L	L	L	H	H	H	L

ASCII PAPER TAPE



BPM-017

Am27S180 • Am27S181

8192-Bit Generic Series Bipolar PROM

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

- High Speed – 60ns max commercial range access time
- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S180 and Am27S181 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test words are pre-programmed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliabilty. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

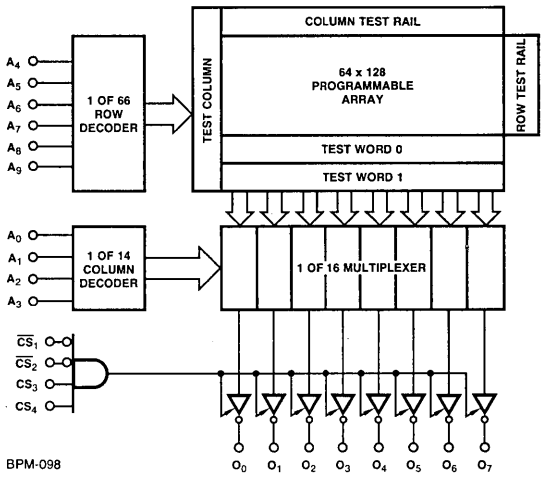
ORDERING INFORMATION

Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0°C to +75°C	AM27S180DC
Hermetic DIP	−55°C to +125°C	AM27S180DM
	Three-State Outputs	
Hermetic DIP	0°C to +75°C	AM27S181DC
Hermetic DIP	−55°C to +125°C	AM27S181DM

FUNCTIONAL DESCRIPTION

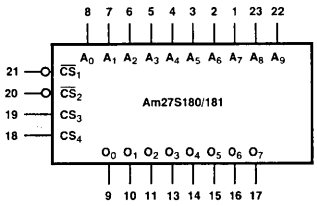
The Am27S180 and Am27S181 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 1024 x 8 configuration, they are available in both open collector Am27S180 and three-state Am27S181 output versions. After programming, stored information is read on outputs O_0-O_7 by applying unique binary addresses to A_0-A_9 and enabling the chip ($\overline{CS}_1$, $\overline{CS}_2$, low and CS_3, CS_4 high). Changes of chip select input levels disables the outputs causing them to go to the off or high impedance state.

BLOCK DIAGRAM



BPM-098

LOGIC DIAGRAM

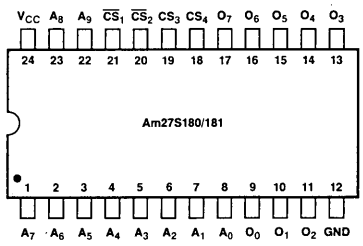


BPM-099

V_{CC} = Pin 24
GND = Pin 12

CONNECTION DIAGRAM

Top View



BPM-100

Note: Pin 1 is marked for orientation.

6

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential (Pin 24 to Pin 12) Continuous	−0.5 to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5 to +5.5V
DC Input Current	−30 to +5mA

OPERATING RANGE

COM'L	Am27S180XC, Am27S181XC	T_A = 0 to 75°C	V_{CC} = 5.0V ±5%
MIL	Am27S180XM, Am27S181XM	T_C = −55 to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions	Min.	Typ. (Note 1)	Max.	Units
V_{OH} (Am27S181 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}	2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}		0.38	0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs	2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs			0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V		−0.010	−0.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V			25	μA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V			1.0	mA
I_{SC} (Am27S181 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)	−20	−40	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.		120	185	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA			−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX., $V_{\overline{CS}1,2}$ = 2.4V, $V_{CS3,4}$ = 0.4V; Am27S181 Only; V_O = 4.5V			40	μA
		V_{CC} = MAX., $V_{\overline{CS}1,2}$ = 2.4V, $V_{CS3,4}$ = 0.4V; Am27S181 Only; V_O = 2.4V			40	μA
		V_{CC} = MAX., $V_{\overline{CS}1,2}$ = 2.4V, $V_{CS3,4}$ = 0.4V; Am27S181 Only; V_O = 0.4V			−40	μA
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1MHz (Note 3)		5		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1MHz (Note 3)		12		pF

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but are periodically sampled.

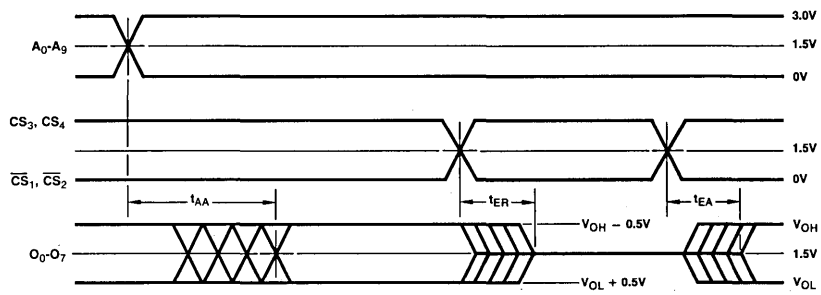
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ 5V 25°C	Max COM'L	Max MIL	Units
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	40	60	80	ns
t_{EA}	Enable Access Time		20	40	50	ns
t_{ER}	Enable Recovery Time		20	40	50	ns

Notes: 1. t_{AA} is tested with switch S_1 closed and C_L = 30pF.
2. For open collector outputs, t_{EA} and t_{ER} are tested with S_1 closed to the 1.5V output level. C_L = 30pF.
3. For three state outputs, t_{EA} is tested with C_L = 30pF to the 1.5V level; S_1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. t_{ER} is tested with C_L = 5pF. HIGH to high impedance tests are made with S_1 open to an output voltage of V_{OH} − 0.5V; LOW to high impedance tests are made with S_1 closed to the V_{OL} + 0.5V level.

SWITCHING CHARACTERISTICS



Note: Level on output while chip is disabled is determined externally.

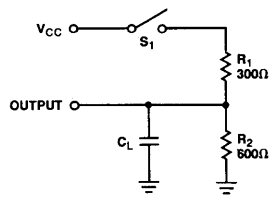
BPM-101

KEY TO TIMING DIAGRAM

WAVEFORM	INPUTS	OUTPUTS
	MUST BE STEADY	WILL BE STEADY
	MAY CHANGE FROM H TO L	WILL BE CHANGING FROM H TO L
	MAY CHANGE FROM L TO H	WILL BE CHANGING FROM L TO H

WAVEFORM	INPUTS	OUTPUTS
	DON'T CARE; ANYCHANGE PERMITTED	CHANGING; STATE UNKNOWN
	DOES NOT APPLY	CENTER LINE IS HIGH IMPEDANCE "OFF" STATE

AC TEST LOAD



BPM-102

6

PROGRAMMING

The Am27S180 and Am27S181 are manufactured with a conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. To program the device, the fusible links are selectively opened.

The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to one memory output after the $\overline{CS}_1$ input is at a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}_1$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip enabled, and the output level sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}_1$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

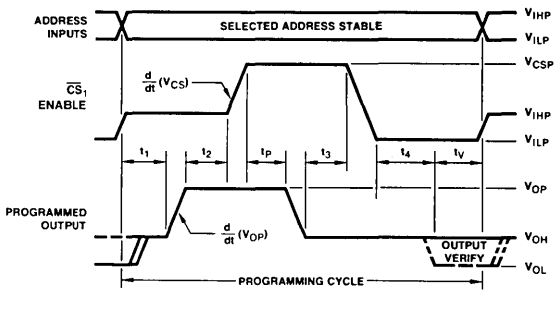
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min.	Max.	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	Volts
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	Volts
V_{ILP}	Input LOW Level During Programming	0.0	0.45	Volts
V_{CSP}	$\overline{CS}_1$ Voltage During Programming	14.5	15.5	Volts
V_{OP}	Output Voltage During Programming	19.5	20.5	Volts
V_{ONP}	Voltage on Outputs Not to be Programmed	0	V_{CCP}+0.3	Volts
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{CS})/dt$	Rate of $\overline{CS}_1$, Voltage Change	100	1000	V/μsec
t_P	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

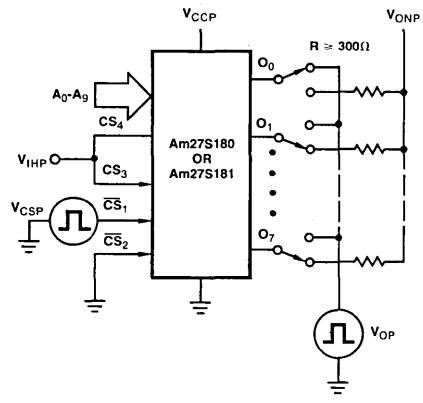
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1, t_2, t_3 and t_4 must be greater than 100 ns; maximum delays of 1 μsec are recommended to minimize heating during programming.
3. During t_V, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-103

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-104

PROGRAMMING EQUIPMENT

Generic programming boards and device adapters are available from the sources listed below. In each case, the programming boards are used in these manufacturer's automatic programmers to program all AMD generic series bipolar PROMs; individual adapters are required for each basic part type in the series.

SOURCE AND LOCATION	Data I/O Corp. P.O. Box 308 Issaquah, Wash. 98027	Pro-Log Corp. 2411 Garden Road Monterey, Ca. 93940
PROGRAMMER MODEL(S)	Model 5, 7 and 9	M900 and M920
AMD GENERIC BIPOLAR PROM PERSONALITY BOARD	909-1286-1	PM9058
Am27S180 • Am27S181 ADAPTERS AND CONFIGURATOR	715-1545-2	PA24-13 and 1024 x 8 (L)

OBTAINING PROGRAMMED UNITS

Programmed devices may be purchased from your distributor or Advanced Micro Devices. The program data should be submitted in the form of a punched paper tape and must be accompanied by a written truth table. The punched tape can be delivered with your order or may be transmitted over a TWX machine or time-sharing terminal. ASCII BPNF is our preferred paper tape format.

Truth tables are also acceptable, however, much less desirable especially for larger density PROMs. Submission of a truth table requires the generation of a punched paper tape at the distributor or factory resulting in longer lead times, greater possibility of error, and higher cost.

ASCII BPNF

An example of an ASCII tape in the BPNF format is shown below. They can be punched on any Teletype® or on a TWX or Telex machine. The format chosen provides relatively good error detection. Paper tapes must consist of:

1. A leader of at least 25 rubouts.
2. The data patterns for all 1024 words, starting with word 0, in the following format:
   a. Any characters, including carriage return and line feed, except "B".
   b. The letter "B", indicating the beginning of the data word.
   c. A sequence of eight Ps or Ns, starting with output O_7.
   d. The letter "F", indicating the finish of the data word.
   e. Any text, including carriage return and line feed, except the letter "B".
3. A trailer of at least 25 rubouts.

A P is a HIGH logic level = 2.4 volts.
An N is a LOW logic level = 0.4 volts.

A convenient pattern to use for the data words is to prefix the word (or every few words) with the word number, then type the data word, then a comment, then carriage return and line feed as shown below. There must be no characters between the B and the F except for the eight Ps and Ns. If an error is made in a word, the entire word must be cancelled with rubouts back to the letter B, then the word re-typed beginning with the B.

When TWXing your tape, be sure the tape is in even parity. Parity is not necessary if the tape is mailed.

TYPICAL PAPER TAPE FORMAT

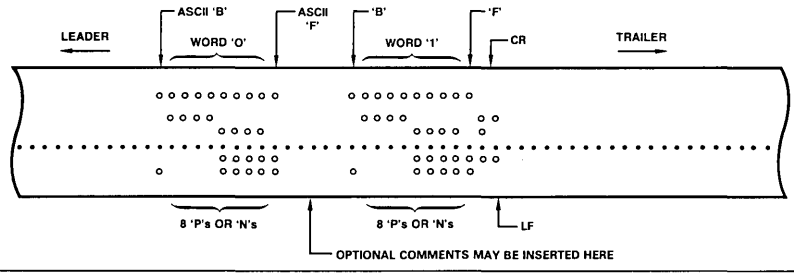
```
ØØØ    BPNPPNNNPF   WORD ZERO Ⓡ Ⓛ
       BPPPPPPNNF   COMMENT FIELD Ⓡ Ⓛ
ØØ2    BNNNPPPPNF   ANY Ⓡ Ⓛ
       BNNNNNNNNF   TEXT Ⓡ Ⓛ
ØØ4    BPNNNNNNPF   CAN Ⓡ Ⓛ
       BNPPNPPNNF   GO Ⓡ Ⓛ
ØØ6    BPNNPPPNNF   HERE Ⓡ Ⓛ
  :    ::::::::::::   :
1023   BNNNNPPPNF   END Ⓡ Ⓛ
```

Ⓡ = CARRIAGE RETURN
Ⓛ = LINE FEED

RESULTING DEVICE TRUTH TABLE ($\overline{CS}_1$ AND $\overline{CS}_2$ LOW, CS_3 AND CS_4 HIGH)

A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
L	L	L	L	L	L	L	L	L	L	H	L	H	H	L	L	L	H
L	L	L	L	L	L	L	L	L	H	H	H	H	H	H	H	L	L
L	L	L	L	L	L	L	L	H	L	L	L	L	H	H	H	H	L
L	L	L	L	L	L	L	L	H	H	L	L	L	L	L	L	L	L
L	L	L	L	L	L	L	H	L	L	H	L	L	L	L	L	L	H
L	L	L	L	L	L	L	H	L	H	L	H	H	L	H	H	L	L
L	L	L	L	L	L	L	H	H	L	H	L	L	H	H	H	L	L
							:								:		
H	H	H	H	H	H	H	H	H	H	L	L	L	L	H	H	H	L

ASCII PAPER TAPE



BPM-105

Am27S184 • Am27S185

8192-Bit Generic Series Bipolar PROM

PRELIMINARY DATA

DISTINCTIVE CHARACTERISTICS

- Excellent performance over full MIL and commercial ranges
- Highly reliable, ultra-fast programming Platinum-Silicide fuses
- High programming yield
- Low current PNP inputs
- High current open collector and three-state outputs
- Fast chip select
- Access time tested with N^2 patterns
- Pin for pin replacements for industry standard products
- Common Generic PROM series electrical characteristics and simple programming procedures.

GENERIC SERIES CHARACTERISTICS

The Am27S184 and Am27S185 are members of an Advanced PROM series incorporating common electrical characteristics and programming procedures. All parts in this series are produced with a fusible link at each memory location storing a logic LOW and can be selectively programmed to a logic HIGH by applying appropriate voltages to the circuit.

All parts are fabricated with AMD's fast programming highly reliable Platinum-Silicide Fuse technology. Utilizing easily implemented programming (and common programming personality card sets) these products can be rapidly programmed to any customized pattern. Extra test words are pre-programmed during manufacturing to insure extremely high field programming yields, and produce excellent parametric correlation.

Platinum-Silicide was selected as the fuse link material to achieve a well controlled melt rate resulting in large non-conductive gaps that ensure very stable long term reliability. Extensive operating testing has proven that this low-field, large-gap technology offers the best reliability for fusible link PROMs.

Common design features include active loading of all critical AC paths regulated by a built-in temperature and voltage compensated bias network to provide excellent parametric performance over MIL supply and temperature ranges. Selective feedback techniques have been employed to minimize delays through all critical paths producing the fastest speeds possible from Schottky processed PROMs.

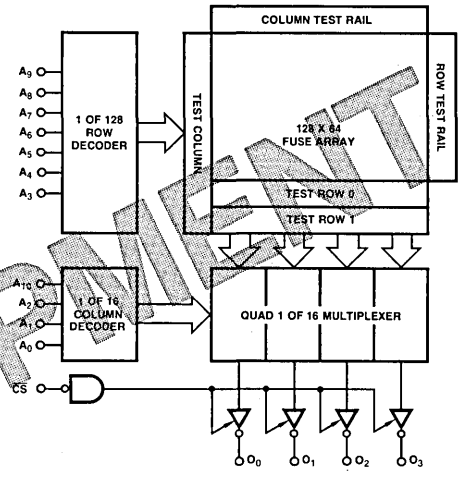
ORDERING INFORMATION

Package Type	Temperature Range	Order Number
	Open Collectors	
Hermetic DIP	0 to +75°C	AM27S184DC
Hermetic DIP	−55 to +125°C	AM27S184DM
	Three-State Outputs	
Hermetic DIP	0 to +75°C	AM27S185DC
Hermetic DIP	−55 to +125°C	AM27S185DM

FUNCTIONAL DESCRIPTION

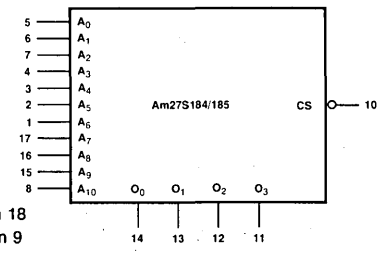
The Am27S184 and Am27S185 are high speed electrically programmable Schottky read only memories. Organized in the industry standard 2048 x 4 configuration, they are available in both open collector Am27S184 and three-state Am27S185 output versions. After programming, stored information is read on outputs O_0-O_3 by applying unique binary addresses to A_0-A_{10} and holding the chip select input $\overline{CS}$ LOW. If the chip select input goes to a logic HIGH, O_0-O_3 go to the off or high-impedance state.

BLOCK DIAGRAM



BPM-106

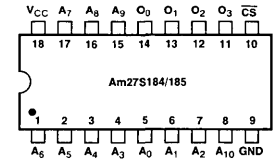
LOGIC SYMBOL



V_{CC} = Pin 18
GND = Pin 9

BPM-107

CONNECTION DIAGRAM
Top View



Note 1: Pin 1 is marked for orientation.

BPM-108

MAXIMUM RATINGS (Above which the useful life may be impaired)

Storage Temperature	−65 to +150°C
Temperature (Ambient) Under Bias	−55 to +125°C
Supply Voltage to Ground Potential (Pin 18 to Pin 9) Continuous	−0.5 to +7.0V
DC Voltage Applied to Outputs (Except During Programming)	−0.5V to +V_{CC} max.
DC Voltage Applied to Outputs During Programming	21V
Output Current into Outputs During Programming (Max. Duration of 1 sec.)	200mA
DC Input Voltage	−0.5 to +5.5V
DC Input Current	−30 to +5mA

OPERATING RANGE

COM'L	Am27S184XC, Am27S185XC	T_A = 0 to 75°C	V_{CC} = 5.0V ±5%
MIL	Am27S184XM, Am27S185XM	T_C = −55 to +125°C	V_{CC} = 5.0V ±10%

ELECTRICAL CHARACTERISTICS OVER OPERATING RANGE (Unless Otherwise Noted)

PRELIMINARY DATA

Parameters	Description	Test Conditions			Min	Typ (Note 1)	Max	Units
V_{OH} (Am27S185 only)	Output HIGH Voltage	V_{CC} = MIN., I_{OH} = −2.0mA V_{IN} = V_{IH} or V_{IL}			2.4			Volts
V_{OL}	Output LOW Voltage	V_{CC} = MIN., I_{OL} = 16mA V_{IN} = V_{IH} or V_{IL}					0.50	Volts
V_{IH}	Input HIGH Level	Guaranteed input logical HIGH voltage for all inputs			2.0			Volts
V_{IL}	Input LOW Level	Guaranteed input logical LOW voltage for all inputs					0.8	Volts
I_{IL}	Input LOW Current	V_{CC} = MAX., V_{IN} = 0.45V				−0.020	−.250	mA
I_{IH}	Input HIGH Current	V_{CC} = MAX., V_{IN} = 2.7V					25	μA
I_I	Input HIGH Current	V_{CC} = MAX., V_{IN} = 5.5V					40	mA
I_{SC} (Am27S185 only)	Output Short Circuit Current	V_{CC} = MAX., V_{OUT} = 0.0V (Note 2)			−20	−45	−90	mA
I_{CC}	Power Supply Current	All inputs = GND V_{CC} = MAX.				80	130	mA
V_I	Input Clamp Voltage	V_{CC} = MIN., I_{IN} = −18mA					−1.2	Volts
I_{CEX}	Output Leakage Current	V_{CC} = MAX $V_{\overline{CS}}$ = 2.4V		V_O = 4.5V			40	μA
			Am27S185 only	V_O = V_{CC}			40	
				V_O = 0.4V			−40	
C_{IN}	Input Capacitance	V_{IN} = 2.0V @ f = 1MHz (Note 3)				5		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0V @ f = 1MHz (Note 3)				8		

Notes: 1. Typical limits are at V_{CC} = 5.0V and T_A = 25°C.
2. Not more than one output should be shorted at a time. Duration of the short circuit should not be more than one second.
3. These parameters are not 100% tested, but are periodically sampled.

6

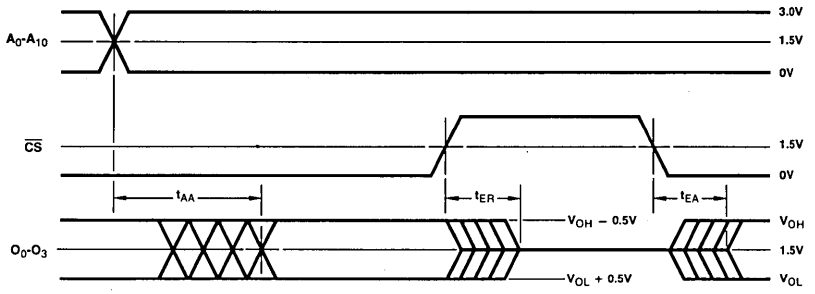
SWITCHING CHARACTERISTICS OVER OPERATING RANGE

PRELIMINARY DATA

Parameter	Description	Test Conditions	Typ	Max		Units
			5V 25°C	COM'L	MIL	
t_{AA}	Address Access Time	AC Test Load (See Notes 1-3)	40	–	–	ns
t_{EA}	Enable Access Time		10	–	–	ns
t_{ER}	Enable Recovery Time		10	–	–	ns

Notes: 1. t_{AA} is tested with switch S_1 closed and C_L = 30pF.
2. For open collector outputs, t_{EA} and t_{ER} are tested with S_1 closed to the 1.5V output level. C_L = 30pF.
3. For three state outputs, t_{EA} is tested with C_L = 30pF to the 1.5V level; S_1 is open for high impedance to HIGH tests and closed for high impedance to LOW tests. t_{ER} is tested with C_L = 5pF. HIGH to high impedance tests are made with S_1 open to an output voltage of V_{OH} – 0.5V; LOW to high impedance tests are made with S_1 closed to the V_{OL} + 0.5V level.

SWITCHING CHARACTERISTICS



Note: Level on output while $\overline{CS}$ is HIGH is determined externally.

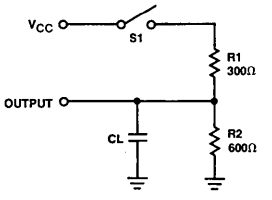
BPM-109

KEY TO TIMING DIAGRAM

WAVEFORM	INPUTS	OUTPUTS
	MUST BE STEADY	WILL BE STEADY
	MAY CHANGE FROM H TO L	WILL BE CHANGING FROM H TO L
	MAY CHANGE FROM L TO H	WILL BE CHANGING FROM L TO H

WAVEFORM	INPUTS	OUTPUTS
	DON'T CARE; ANYCHANGE PERMITTED	CHANGING; STATE UNKNOWN
	DOES NOT APPLY	CENTER LINE IS HIGH IMPEDANCE "OFF" STATE

AC TEST LOAD



BPM-110

PROGRAMMING

The Am27S184 and Am27S185 are manufactured with conductive Platinum-Silicide link at each bit location. The output of the memory with the link in place is LOW. To program the device, the fusible links are selectively opened.

The fusible links are opened one at a time by passing current through them from a 20 volt supply which is applied to one memory output after the $\overline{CS}$ input is a logic HIGH. Current is gated through the addressed fuse by raising the $\overline{CS}$ input from a logic HIGH to 15 volts. After 50 μsec, the 20 volt supply is removed, the chip enabled, and the output level sensed to determine if the link has opened. Most links will open within 50 μsec. Occasionally a link will be stronger and require additional programming cycles. The recommended duration of additional programming periods is 5 msec. If a link has not opened after a total elapsed programming time of 400 msec, further programming of the device should not be attempted. Successive links are programmed in the same manner until all desired bit locations have been programmed to the HIGH level.

Typical current into an output during programming will be approximately 140mA until the fuse link is opened, after which the current drops to approximately 40mA. Current into the $\overline{CS}$ pin when it is raised to 15 volts is typically 1.5mA.

The memories may become hot during programming due to the large currents being passed. Programming cycles should not be applied to one device more than 5 seconds to avoid heat damage. If this programming time is exceeded, all power to the chip including V_{CC} should be removed for a period of 5 seconds after which programming may be resumed.

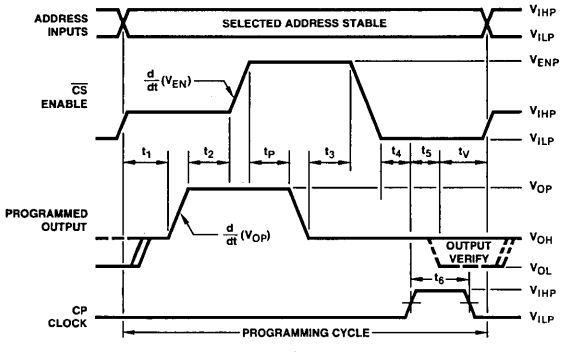
When all programming has been completed, the data content of the memory should be verified by sequentially reading all words. Occasionally this verification will show that an extra undesired link has been fused. Should this occur, immediately check the programming equipment to make sure that all device pins are firmly contacting the programming socket, that the input signal levels exhibit sufficient noise margins, and that the programming voltages are within the specified limits. All of these conditions must be maintained during programming. AMD PROMs are thoroughly tested to minimize unwanted fusing; fusing extra bits is generally related to programming equipment problems.

PROGRAMMING PARAMETERS

Parameter	Description	Min	Max	Units
V_{CCP}	V_{CC} During Programming	5.0	5.5	Volts
V_{IHP}	Input HIGH Level During Programming	2.4	5.5	Volts
V_{ILP}	Input LOW Level During Programming	0.0	0.45	Volts
V_{CSP}	$\overline{CS}$ Voltage During Programming	14.5	15.5	Volts
V_{OP}	Output Voltage During Programming	19.5	20.5	Volts
V_{ONP}	Voltage on Outputs Not to be Programmed	0	V_{CCP}+0.3	Volts
I_{ONP}	Current into Outputs Not to be Programmed		20	mA
$d(V_{OP})/dt$	Rate of Output Voltage Change	20	250	V/μsec
$d(V_{CS})/dt$	Rate of $\overline{CS}$, Voltage Change	100	1000	V/μsec
t_P	Programming Period – First Attempt	50	100	μsec
	Programming Period – Subsequent Attempts	5.0	15	msec

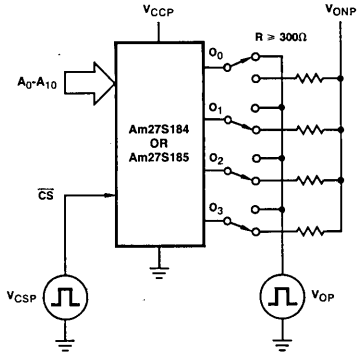
Notes:
1. All delays between edges are specified from completion of the first edge to beginning of the second edge; i.e., not to the midpoints.
2. Delays t_1, t_2, t_3 and t_4 must be greater than 100 ns; maximum delays of 1 μsec are recommended to minimize heating during programming.
3. During t_v, a user defined period, the output being programmed is switched to the load R and read to determine if additional pulses are required.
4. Outputs not being programmed are connected to V_{ONP} through resistor R which provides output current limiting.

PROGRAMMING WAVEFORMS



BPM-111

SIMPLIFIED PROGRAMMING DIAGRAM



BPM-112

PROGRAMMING EQUIPMENT

Generic programming boards and device adapters are available from the sources listed below. In each case, the programming boards are used in these manufacturer's automatic programmers to program all AMD generic series bipolar PROMs; individual adapters are required for each basic part type in the series.

SOURCE AND LOCATION	Data I/O Corp. P.O. Box 308 Issaquah, Wash. 98027	Pro-Log Corp. 2411 Garden Road Monterey, Ca. 93940
PROGRAMMER MODEL(S)	Model 5, 7 and 9	M900 and M920
AMD GENERIC BIPOLAR PROM PERSONALITY BOARD	909-1286-1	PM9058
Am27S184 • Am27S185 ADAPTERS AND CONFIGURATOR	715-1616	PA18-8 and 2048 x 4 (L)

OBTAINING PROGRAMMED UNITS

Programmed devices may be purchased from your distributor or Advanced Micro Devices. The program data should be submitted in the form of a punched paper tape and must be accompanied by a written truth table. The punched tape can be delivered with your order or may be transmitted over a TWX machine or time-sharing terminal. ASCII BPNF is our preferred paper tape format.

Truth tables are also acceptable, but are much less desirable especially for larger density PROMs. Submission of a truth table requires the generation of a punched paper tape at the distributor or factory resulting in longer lead times, greater possibility of error, and higher cost.

ASCII BPNF

An example of an ASCII tape in the BPNF format is shown below. They can be punched on any Teletype® or on a TWX or Telex machine. The format chosen provides relatively good error detection. Paper tapes must consist of:

1. A leader of at least 25 rubouts.
2. The data patterns for all 2048 words, starting with word 0, in the following format:
   a. Any characters, including carriage return and line feed, except "B".
   b. The letter "B", indicating the beginning of the data word.
   c. A sequence of four Ps or Ns, starting with output O_3.
   d. The letter "F", indicating the finish of the data word.
   e. Any text, including carriage return and line feed, except the letter "B".
3. A trailer of at least 25 rubouts.

A P is a HIGH logic level = 2.4 volts.
An N is a LOW logic level = 0.4 volts.

A convenient pattern to use for the data words is to prefix the word (or every few words with the word number, then type the data word, then a comment, then carriage return and line feed as shown below. There must be no characters between the B and the F except for the four Ps and Ns. If an error is made in a word, the entire word must be cancelled with rubouts back to the letter B, then the word re-typed beginning with the B.

When TWXing your tape, be sure the tape is in even parity. Parity is not necessary if the tape is mailed.

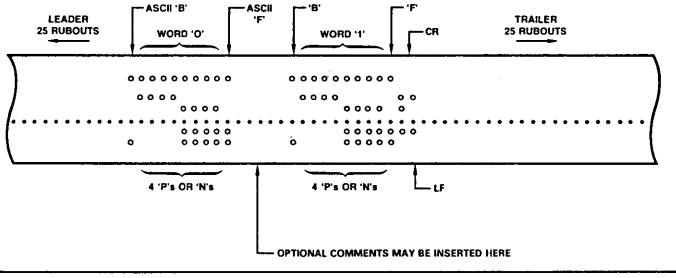
TYPICAL PAPER TAPE FORMAT

ϕϕϕ	BNNNPF	WORD ZERO (R) (L)
	BPPNNF	COMMENT FIELD (R) (L)
ϕϕ2	BPPPNF	ANY (R) (L)
	BNNNNF	TEXT (R) (L)
ϕϕ4	BNNNPF	CAN (R) (L)
	BPPNNF	GO (R) (L)
ϕϕ6	BPPNNF	HERE (R) (L)
:	:	:
2047	BPPPNF	END (R) (L)

RESULTING DEVICE TRUTH TABLE ($\overline{CS}$ LOW)

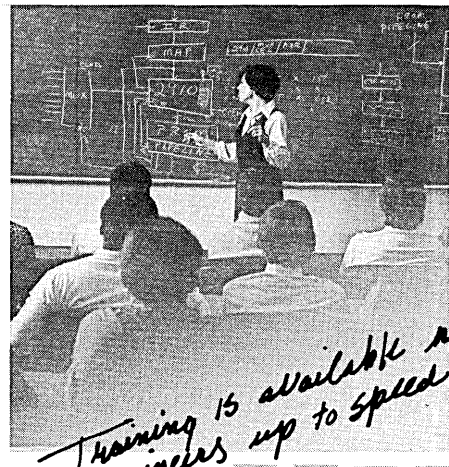
A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	O_3	O_2	O_1	O_0
L	L	L	L	L	L	L	L	L	L	L	L	L	L	H
L	L	L	L	L	L	L	L	L	L	H	H	H	L	L
L	L	L	L	L	L	L	L	L	H	L	H	H	H	L
L	L	L	L	L	L	L	L	L	H	H	L	L	L	L
L	L	L	L	L	L	L	L	H	L	L	L	L	L	H
L	L	L	L	L	L	L	L	H	L	H	H	H	L	L
L	L	L	L	L	L	L	L	H	H	L	H	H	L	L
H	H	H	H	H	H	H	H	H	H	H	H	H	H	L

ASCII PAPER TAPE



BPM-113

AmZ8000 Microprocessor Family



CHAPTER 7

Development System Products

AmSYS™8/8 Microcomputer Development System

Especially Designed to Support
the AmZ8000 Microprocessor
and the 8080A, 8085A, Z80 & 8048

7

For Total Capability, AMC gives you the AmSYS 8/8 Microcomputer Development System

SYSTEM DESCRIPTION

The AmSYS 8/8 is designed to support a variety of microprocessors that include the AmZ8000, Z80, Am9080, Am8085, and Am8048.

This system is especially designed to support the AmZ8000 CPU in both hardware and software development. An AmZ8000 Macro Assembler, linker and line oriented editor give the system powerful software development tools. The use of the RTE 8/8050 In-Circuit Emulator for AmZ8000 hardware/software debug and availability of high level languages allow the user great flexibility in development of AmZ8000 products.

The basic system is contained in a single enclosure that fits on a desk top. It has the option of being rack mountable in a 19" relay rack by the addition of slide mounts. It has 64K bytes of main memory, two single density floppy disk drives, providing 512K bytes of storage, an RS232 serial port and two parallel I/O channels as standard features in the system. This system contains extra card slots for use for prototyping hardware.

The AmSYS 8/8 has a multi-master bus structure that allows multiple 8 and 16 bit CPUs to be used at the same time. This bus structure allows the system to be easily upgraded to a full 16 bits by replacing the CPU board on the bus and adding the appropriate software.

The AmSYS 8/8 has very powerful software included as part of the standard system to be delivered. This software includes a Disk Operating system, an excellent text editor, debugger, library manager and linking loader. The powerful macroassemblers included support the AmZ8000 and the 8080, 8085, & Z80. Optionally, an 8048 macroassembler can be added.

The AmSYS 8/8 features the use of Pascal along with extended Basic, Fortran IV and Cobol for high level language support. A Pascal program can be compiled to generate either Am9080 or AmZ8000 code.

In-Circuit Emulation capability is available for both 8 & 16 bit processors. This emulation support is optional for the AmZ8000, Z80, 8080, 8085, & 8048 microprocessors.

The AmSYS 8/8 also has a number of peripherals that are optional to the system. These include double density Floppy Disk Drives, 60, 120 CPS and 300 LPM Line Printers, CRT's, and soon a cartridge disk drive.

SYSTEM FEATURES

Complete Turnkey Development System with Powerful Hardware including:

- ☐ Microprocessor CPU
- ☐ 64 kilobytes of read/write main memory
- ☐ Serial interface RS232 compatible
- ☐ Two parallel Channels
- ☐ Power up Monitor
- ☐ Dual Floppy Disk Drives with 512K bytes of storage capacity
- ☐ Extra card slots for prototyping
- ☐ Multi-master bus

Powerful Software Including:

- ☐ AMDOS Disk Operating System
- ☐ Sophisticated Editor
- ☐ Debugger
- ☐ Macroassemblers for AmZ8000, Z80, 8080, 8085
- ☐ Translator to Z8000
- ☐ Linkers

Powerful Options Including:

- ☐ High level languages
  - Pascal
  - Extended Basic
  - Fortran
  - Cobol
- ☐ Line Printers 60 & 120 CPS & 300 LPM
- ☐ CRT w/extra keypad
- ☐ PROM Programmer
- ☐ Double Density Floppy Disk Drives
- ☐ Expansion Chassis containing two additional Disk Drives
- ☐ Additional Serial I/O lines
- ☐ In-Circuit Emulation for
  - AmZ8000
  - 8080, 8085
  - 8048, 8021, 8049, & 8035

Additional Features that will be available in the near future include:

- ☐ Expansion to 1 megabyte of main memory
- ☐ Upgrade to AmZ8000 CPU
- ☐ Cartridge Disk
- ☐ Additional High level languages
- ☐ Expansion chassis for more Main Memory

System Hardware Features

The AmSYS 8/8 features the use of a multiple master bus structure. This allows the system to use both 8 and 16 bit microprocessors. The SYS 8/8 can be upgraded to include a 16 bit CPU board and appropriate software so that both microprocessors may operate simultaneously using the multi-master bus structure. This allows the system user maximum flexibility in the development of hardware & software.

MAIN CPU MODULE

The Central Processor (CPU) module uses an 8-bit Am9080 microprocessor. This module includes a ROM based auto bootstrap that first performs a diagnostic confidence check of the system and next loads the operating system from the system disk drives. The CPU module has multi-master capability allowing operation in a master/slave environment with other 8-bit CPU modules or the 16-bit CPU module. The CPU module also includes 4 high speed DMA channels and 8 vectored interrupt channels. An optional Am9511A/Am9512 Arithmetic Processing Unit can be added to provide high speed fixed and floating point computations and floating point transcendental functions to the system. A 1.2KHz crystal controlled Real Time Clock is also part of this module.

MEMORY MODULE

This memory module is organized as a 64 Kilobyte storage unit. This module is organized to provide either 8 or 16-bit compatibility. It also contains internal memory refresh capability and is multimaster bus compatible.

FLOPPY DISK MODULE

This module features a controller with its own internal microprocessor and controls up to 4 single or double sided floppy disks. This controller has a 20-bit address and DMA capability allowing it to address up to 1 megabyte of main memory.

This module is IBM 3740 soft sector compatible. AmSYS 8/8 provides a capacity of 512K bytes of disk storage in the basic system. This storage capacity is contained on two floppy disk drives each with 256K bytes of storage. On board buffer memory allows high speed data transfer to the CPU module. Additional storage is available by the use of the optional double density disk controller or by the addition of a double disk drive expansion chassis.

CHASSIS

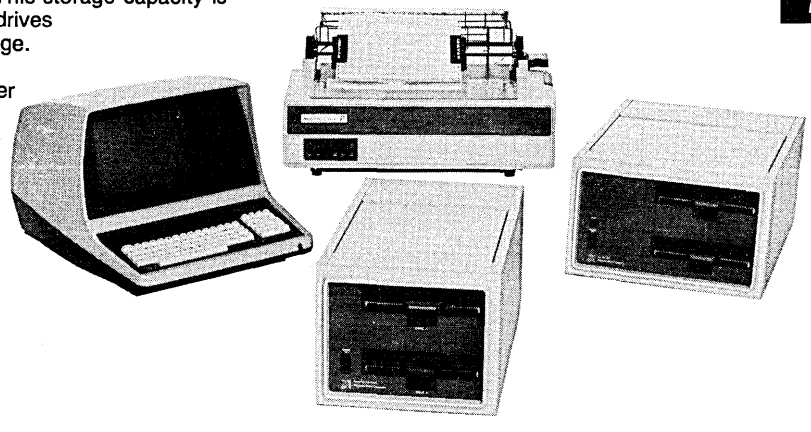
The AmSYS 8/8 is normally supplied as a desk top chassis. As an option, this chassis can be rack mounted with the addition of a slide mount kit. It contains seven card slots with three available for use with prototyping or additional I/O P.C. cards. This chassis is totally self contained with cooling and internal power supplies.

The front of the chassis contains an illuminated off/on switch and also a system reset button. The rear of this unit contains all I/O and power connections including fuses.

The system includes as standard, one serial port and two parallel channels. The serial port is RS232 compatible and allows the user to connect a variety of terminals to the system. The parallel channels can be used to connect high and low speed printers or other peripheral devices. Additional I/O ports can be added to the System by the use of plug in cards.

The parallel channels each consist of 24 parallel I/O lines. These lines can be configured under software control as sets of input, output, or bidirectional I/O lines. Sockets are provided within the system to add drivers or terminators to each line.

The Am96/4016 Evaluation Board can be plugged into one of the empty card cage slots to provide a 16 bit execution module within AmSYS 8/8. The board is powered by the system power supplies and uses the system floppy disks via the parallel up-load/down-load link to the CPU module. A second card cage slot can be used to upgrade the Evaluation Board RAM to 64k.



Development Software

The AmSYS 8/8 combines a sophisticated set of hardware along with an even more powerful set of software to make the complex process of product development easier to handle for the user. This software includes an Editor, File Manager, Macroassemblers, disk operating system, and a number of high level languages.

The software that comes standard with the AmSYS 8/8 is the AMDOS® Operating System, AmZ8000, 8080, 8085, and Z80 Macro Assemblers, a linking loader, a powerful Editor, debugger and Translator. Optional software includes an 8048 Macroassembler, Pascal, Fortran, Basic, Cobol, and upload/download packages for execution of AmZ8000 code. The following describes the major software modules provided with or as options to the system.

OPERATING SYSTEM

AMDOS 8 is the disk operating system for the AmSYS 8/8 and provides rapid access to programs through a comprehensive file management structure. A file subsystem supports a named file structure allowing the dynamic allocation of file space as well as sequential and random file access. System calls are provided permitting files to be opened, closed, renamed, read, written onto disk, or searched for by name. The AMDOS 8 file system allows a large number of distinct programs to be stored in both source and machine executable forms. AMDOS 8 also provides the ability to access disk storage, Terminal, Printer, and support for PROM programming.

LIBR is an object file library manager that permits the creation and listing of user libraries for the Am8080, Am8085, and Z80 microcomputers. Individual object-module files may be selectively processed to extract modules from existing libraries or relocatable object files to build libraries for later selective searching by the LINK linkage editor.

EDIT is a line-oriented context editor providing the user the ability to create and modify ASCII source text for the AmSYS 8/8 compilers and assemblers. A powerful set of user commands is available to simplify the task of generating line oriented or character oriented files.

In the line oriented mode, numbers will automatically be added to new lines as they are appended to the file. Also, as lines are inserted or deleted within the text, these numbers are updated. The users position within the file can then be controlled by referring to the desired line number. The ALTER mode allows the user to insert, delete, and replace individual characters, strings in a line, or a range of lines in the source file. Searching for occurrences of characters or strings is also included along with substitution of characters or strings into the source text.

DEBUG is designed to provide dynamic interactive testing and debugging of 8-bit programs generated in the AmSYS 8/8 system. This program consists of two parts: The debug nucleus and the assembler/disassembler module. These along with powerful user commands make debug a very valuable tool for debugging software programs.

MACRO ASSEMBLERS

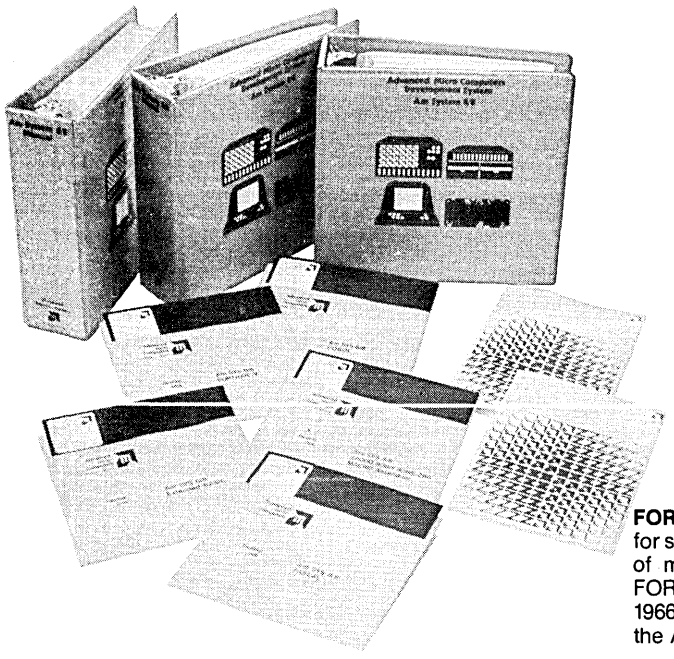
Two macroassemblers come as part of the standard AmSYS 8/8. These are MACRO 8 supporting the 8080, 8085, and Z80 microprocessors and MACRO 8000 for support of the AmZ8000 microprocessor. An optional macroassembler supports the 8048 family.

MACRO 8 is a comprehensive macroassembler providing the ability to assemble relocatable 8080/ 8085/ Z80 programs. These can be combined with object files produced by high-level language compilers (using LINK) to form composite executable programs. In addition to comprehensive conditional assembly directives, hierarchical expression evaluation, definition of alternate entry-points and external symbols, MACRO 8 provides a companion cross-reference facility (CREF8) which lists the assembler output with line numbers.An alphabetic variable-name directory is also provided and includes line number references for each occurrence of the variable name.

MACRO 8000 is a powerful macroassembler providing the ability to assemble relocatable AmZ8000 programs. This assembler includes sophisticated features like segmentation, block-structured program organization, algorithmic assignment statements, IF-THEN-ELSE conditional assembly, and execution provisions as well as recursive macro calls.

LINK 8 & LINK 8000 are linkage editors used to combine the relocatable module ready for loading as a program. Link 8 supports the editing of 8 bit segments. The linkers process a series of interactive subcommands that specify files to be linked, files to be searched for satisfaction of unresolved external references, and specific directives which allow a memory map to be printed or direct execution to be initiated.

TRANZ 8000 is an AmZ8000 translator program accepting standard 8080, 8085, and Z80 source code as input and provides standard AmZ8000 source code as output, aiding the user in conversion of existing 8-bit programs to the AmZ8000 environment.



HIGH LEVEL LANGUAGES

PASCAL is a new, up-to-date, language developed for users who are seeking new features to solve today's problems. PASCAL incorporates new features like the concept of variable data types: bits, bytes, words, records, sets, scalars ... and others that are appropriate to the solution of complex problems. The block structured nature of the language permits the user to create software in a structured environment resulting in a lower development costs, more concise documentation and lower maintenance costs.

The AMC PASCAL is upwards compatible with Jensen and Wirth PASCAL. Extensions provided include:

Interactive files	Comparison of arrays
Untyped files	Comparison of records
Random access of files	Text intrinsics
Strings intrinsics	

The Pascal user can compile a Pascal program and then execute the program on AmSYS 8/8 with the Pascal interpreter and run-time library. The output of this compiler is a special code called P-code and is compatible with the well-known UCSD P-code. The Pascal interpreter executes the P-code instructions when running the Pascal program. The run-time library routines are used as needed in Pascal program execution.

The Pascal user can compile a Pascal program and then generate either 8080, or AmZ8000 code. For 8080 code generation, an 8080 macro library, the MACRO 8 8-bit assembler, and the linker LINK 8 are automatically used. For AmZ8000 code generation, the AmZ8000 macro library, MACRO8000 16-bit assembler, and the linker LINK 8000 are used automatically.

FORTRAN has long been accepted as the standard for scientific programming and is the "native" language of many professional programmers. AMC supports FORTRAN with a compiler conforming to the ANSI 1966 specification for the 8080 microprocessor in the AmSYS 8/8 system.

Full conformance to the ANSI standard insures that accumulated libraries of FORTRAN programs will be immediately usable in the AmSYS 8/8 environment. AMC FORTRAN opens the door to the richest traditions of scientific programming in a small, inexpensive environment.

BASIC is one of the most comprehensive 8-bit BASIC language software programs available today. It contains many unique features not found in other implementations, like:

a. Direct access to the CPU I/O Ports
b. Full Print Using Capability
c. Trace facilities
d. Four variable types — Integer, String, Single (7 digits) and Double (16 digits) precision floating point.

These are only a few of the features available in AMC's BASIC. It also has one of the largest sets of statements making it a very powerful language available to the AmSYS 8/8 user.

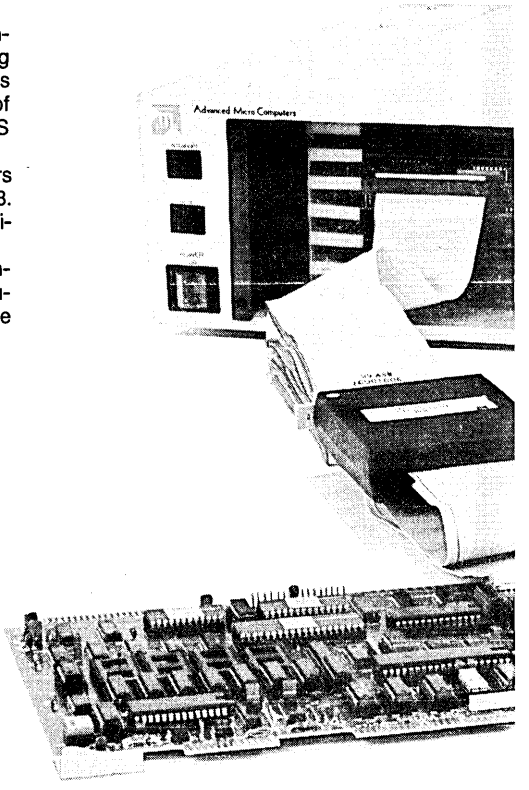
COBOL has been developed in strict accordance to ANSI '74 standards for support of the 8080 within the AmSYS System. At the root of the language is a full ANSI '74 Level 1 COBOL. Beyond that, many Level 2 features have been incorporated to make AMC's COBOL more powerful in every instance where the fundamental speed and size of the package is extremely important. In addition to this, special display-oriented features (ACCEPT, DISPLAY, etc.) have been added to the language in anticipation of a strong emphasis on interactive data entry applications in the microcomputer environment. A powerful interactive debug structure has also been added to greatly decrease program development time.

In-Circuit Emulation

The AmSYS 8/8 Development System provides optional In-Circuit Emulation capability to support a wide range of microprocessors including the AmZ8000, 8080, 8085, Z80, and 8048. The AmSYS series of emulators contain many unique features to assist the designer in the debugging of hardware and software. There are two types of emulators in the AmSYS 8/8 Development System Series. These are:

(1) AmSYS 8/8800 In-Circuit Emulator series for 8-bit microprocessors including individual emulators for the 8080, 8085, Z80, and 8048.
(2) AmSYS 8/8050 In-Circuit Emulator for the AmZ8001 and 8002 microprocessors.

These in-circuit emulators come as complete subsystems and are connected to the AmSYS 8/8 by a high speed serial I/O interface. Each emulator provides its own internal memory storage to give the user real time emulation.



AmSYS 8/8050 16 BIT IN-CIRCUIT EMULATION SUBSYSTEM

The AmSYS RTE 8/8050 Emulator has been designed especially to support the AmZ8000 microprocessor. This subsystem provides real time emulation for both the AmZ8001 segmented and AmZ8002 non-segmented versions.

The AmSYS RTE 8/8050 Emulator has a number of key features including TRACE, 8 trigger points, memory mapping of internal high speed static RAM, and medium speed dynamic RAM providing real time emulation in the user's target system. The AmSYS RTE 8/8050 Emulator has two modes of operation:

1) Interrogation — this mode allows the user to access AmZ8000 resources, registers, I/O ports, and target RAM.
2) Emulation — this mode allows the designer the choice of an internal or user designated clock for software execution or emulation in his target system

RTE 8/8050 provides two types of memory for the user. One type is high speed static memory providing the user with real time emulation at up to 4 MHz for the AmZ8000 with no wait states.It is available in 4K or 8K bytes. The second type is medium speed dynamic RAM available in 64K or 128K byte increments with one wait state at 4 MHz and is expandable up to 192K bytes.

RTE 8/8050 allows for memory mapping on 1K byte boundaries within the AmZ8000 8M byte address space .

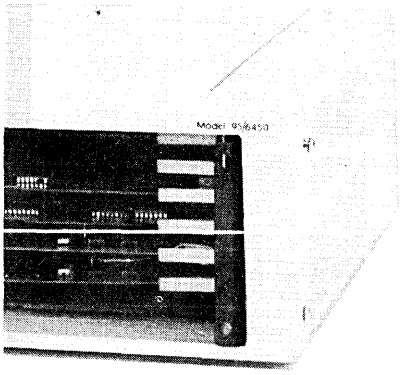
There are up to 8 trigger points available to the user that can be used as breakpoints, enabling patching trace qualifiers or allowing selective trace after the breakpoint is encountered. Two compound breakpoints are also permitted. These allow breakpoints on a value within or outside a specified range, i.e., address, data, address and data, I/O address, I/O data, I/O, address and data.

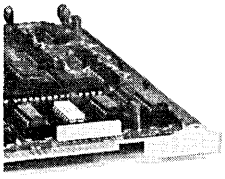
Trace is an integral part of the system. This feature provides for two basic modes of operation. These are:

Micro Trace — this saves Address/Data bus and status together with 8 or 16 user designated probes every machine cycle.

Macro Trace — this saves Address bus and status during T1 displaying address bus in HEX format and status in binary and symbolically decoded form. Also saved is Data Bus during T3 in disassembled form with operand values in HEX format.

The Micro or Macro trace can be enabled only for selected cycles by use of a trigger point match. A user option is available to tag each traced event with a 16 bit number for counting the number of machine states since the counter was enabled. A unique feature of the RTE 8/8050 is the PATCH allowing patching of ROM/RAM code while running in user memory. Up to 4 patches are allowed with up to 16 locations each providing a substantial improvement over conventional patching techniques that require subroutines.



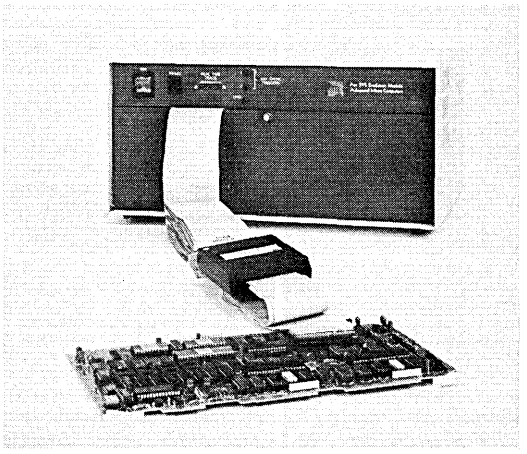


AmSYS 8/8800 8-BIT IN-CIRCUIT EMULATION SUBSYSTEM

The RTE 8/8800 in-circuit emulator is designed to allow replacement of the target microprocesser during the debugging and prototyping phase. It provides the user with real time emulation of each designated microprocessor together with sophisticated debug tools for hardware/software integration resulting in a reduction of overall development time.

AmSYS RTE 8/8800 provides versatile emulation capabilities for the 8080, 8085, Z80A, or 8048 depending upon the personality module being used. It also has the capability to examine and alter registers, memory, and I/O ports. The emulator utilizes the users system clock, thus eliminating potential timing problems caused by separate clocks.

There are up to 8K bytes of high speed Static RAM Emulator memory for mapping on 1K byte boundaries in the target system, thereby utilizing known memory into the target system. This unit has a real time trace for storing the last 128 bus operations as well as the 8 external probes. During selected emulator operations, 16 address lines, 8 data lines, and the clock signal are stored during emulator operation. Disassemblers for each supported microprocessor are provided together with host software for the AmSYS 8/8 Development System.



The AmSYS RTE 8/8800 in-circuit emulator consists of a basic sub-system containing a trace module, emulator module, serial communications module and 8K bytes of high speed static RAM. The connection to the AmSYS 8/8 Development System is via a serial I/O port. Each microprocessor personality module is supplied with its own emulator pod attached to a cable terminating in a 40-pin connector to provide the interface to the target system. Host software operating on AmSYS 8/8 provides interrogation mapping and a command structure for the RTE 8/8800 in-circuit emulators.

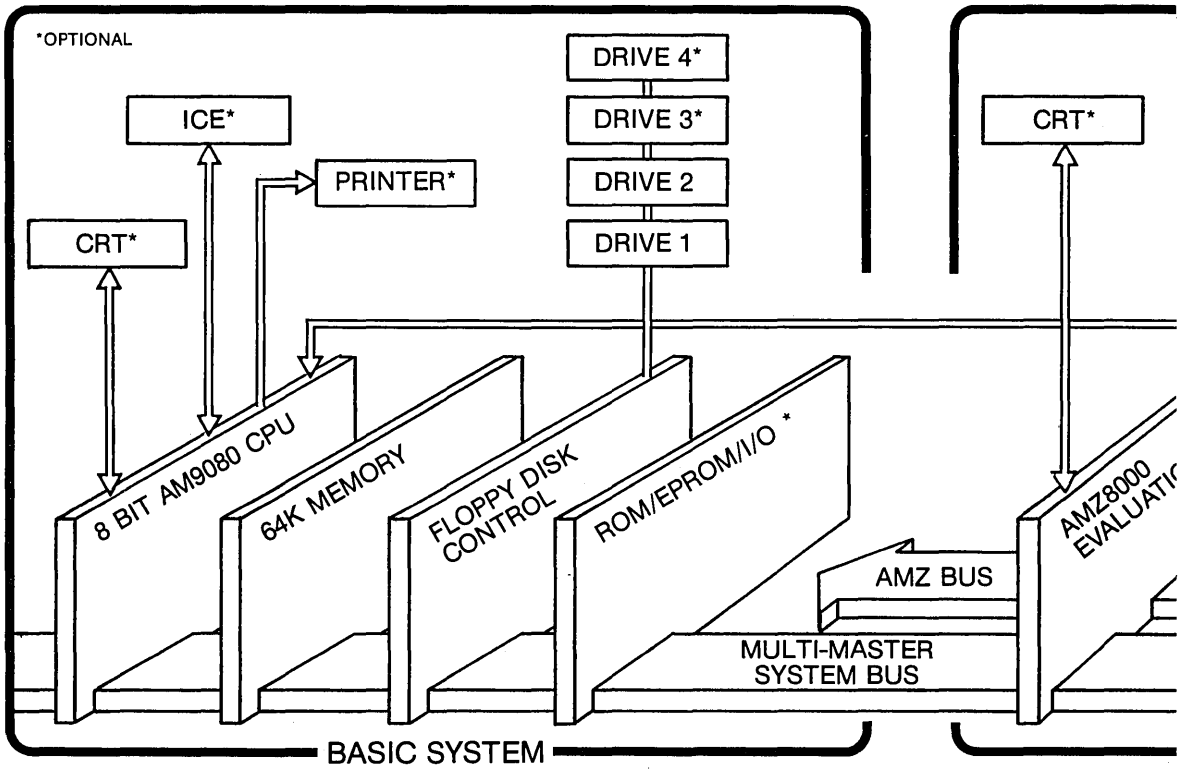
The trace RAM buffer is 256 words deep and 48 bits wide and is expandable to 1024 words. This unit has the ability to time stamp each entry in the trace buffer.

The RTE 8/8050 Emulation Subsystem is provided as a stand-alone unit with integral power supplies,emulator module, trace module and serial I/O port. Programs prepared on the AmSYS 8/8 development system are down-line loaded to the emulator subsystem under the control of the emulator control processor, which services and issues all emulator commands. RTE 8/8050 Emulator commands and results can be displayed via the AmSYS 8/8 Development System CRT or via a separate CRT console plugged into the RTE 8/8050 Subsystem. (This configuration leaves the AmSYS 8/8 Development System free to perform software development, i.e., assembly, editing, etc.) The RTE 8/8050 Emulator has been designed to give the designer maximum support in system debugging.

7

The AmSYS 8/8 is Easy to Adapt

The AmSYS 8/8 is designed to allow easy addition or reconfiguration into a more powerful system. These optional additions and configurations allow execution and debug of AmZ8000 programs, the addition of Double Density Floppy Disks, Hard memory expansion, and reconfiguration as an AmZ8000 based development system.



AmZ8000 EXECUTION AND DEBUG (8/8610 and 8620)

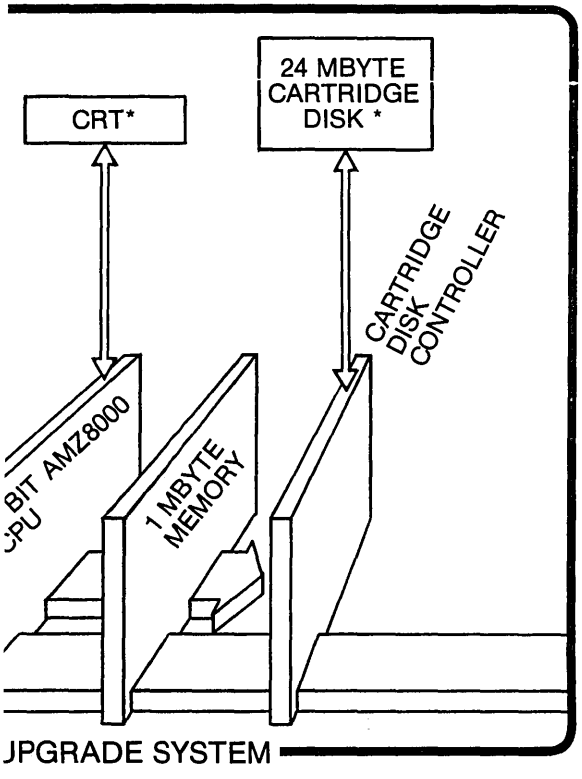
The 8/8610 up-load/down-load package provides an ideal breadboarding, software execution, and debug tool when used with the AmSYS 8/8 development system. It features the AMC 96/4016 AmZ8000 Evaluation Board. AmZ8000 programs generated on AmSYS 8/8 can be down-loaded into or up-loaded from RAM on the Evaluation board where the resident monitor allows execution and debugging to take place under the control of the AmSYS 8/8 console. This Evaluation board provides the user with 8K byte RAM. The 8/8610 can be upgraded to the 8/8620 package with the addition of the 64k RAM board and the AmZ bus motherboard. This expands the RAM storage of the Evaluation Board to 64k. The connection between the RAM board and the Evaluation Board is via the AmZ address and data bus. These options are conveniently packaged as options to AmSYS 8/8.

ADDITIONAL FLOPPY DISK DRIVES

The Floppy Disk controller contained within the AmSYS S 8/8 has the capability of addressing up to four (4) Floppy Disk drives. The addition of the optional 8/8510 Floppy Disk chassis gives the user two (2) additional Floppy Disk drives.

DUAL DENSITY FLOPPY DISK EXPANSION

AmSYS 8/8 can be configured to accept Dual Density Floppy Disk capability. This will give the user Floppy Disk storage capacity of up to 1024K bytes within the AmSYS 8/8. This expansion can be accomplished as a field upgrade.



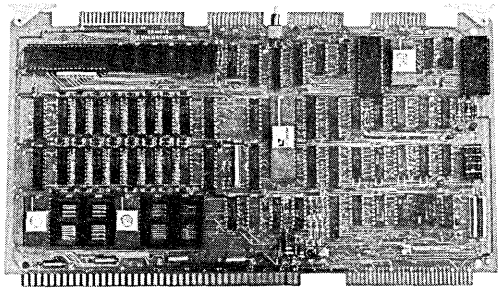
HARD DISK

AmSYS 8/8 is designed for the addition of a Hard Disk with bulk storage of up to 24 megabytes. This Hard Disk option will be available in the near future to be added to the present 8 bit operating system and also as part of an upgrade to the AmZ8000 based development system.

This Hard Disk option will give the user access to larger amounts of bulk storage and also increases his system throughput.

AmZ8000 EVALUATION BOARD

For evaluation of the AmZ8000, the 96/4016 provides a standalone monoboard computer. The 4016 has 8k bytes of dynamic RAM, 2 serial RS-232 ports, 24 parallel I/O lines, and 3-16 bit counters. A 4k byte monitor program provides debug capability and an interface to the optional 8k line by-line assembler. The monitor also provides the drivers necessary to up-load or down-load files through the parallel or serial interface. The 96/4016 is SBC form factor compatible and can plug into AmSYS 8/8 to get its power from the P1 connector. The memory and I/0 on the board can be expanded through the AmZ bus on the P2 connector.



AmZ8000 UPGRADE

The AmSYS 8/8 multimaster bus provides an 8/16 bit data bus with 20 address lines. The peripheral controllers and memory boards all have 8/16 bit compatibility and allow the reconfiguration of the AmSYS 8/8 into a powerful AmZ8000 based development system. The Am9080 board is replaced by the AmZ8000 CPU board, allowing up to 1M byte addressing capability. The addition of AMC's multitasking foreground background system, together with additional 64K or 128K byte RAM memory modules and AmZ8000 software development programs provide the user with a powerful 16-bit system with greatly increased utility and performance.

SPECIFICATIONS

CPU

Am9080 Upgradable to AmZ8000 (see sections entitled "Options")

Memory

64K bytes standard

Disk Storage

512K bytes (2 Single Density Floppy disks, 256K bytes each) (Dual Density is optional) (see section entitled "Optional Upgrades")

I/O Channels

1 serial port RS232 compatible
2 parallel I/O channels consisting of three 8-bit ports each

Interrupt

8 fully programmable vectored channels

AC Power Requirement

60 Hz, 115 VAC std.
100, 120, 220, 240 VAC optional
50/60 Hz Optional

Environmental Requirements

Operating Temperature:	10°C to +40°C
Humidity:	10% - 90% relative (non-condensing)

Physical Characteristics

Dimensions:	24" deep, 17" wide, 11" high
Weight:	60 lbs.
User Panel:	Contains: a) on/off switch, power indicator b) system reset
Chassis:	Desk top stds., rack mountable (optional)

Baud Rates

Selectable 50-9600 Baud

Cooling

Internal fans

Standard System Includes the Following:

Am9080 CPU
64K Bytes RAM
512K Bytes Disk storage (2 single sided floppy disks)
1 RS232 serial port
6 8-bit parallel ports
1 Chassis consisting of 7 Card slots
AmZ8000 Macroassembler
AmZ8000 Translator
8080, 8085, and Z80 Macroassemblers
Linker
Editor
Debugger
AMDOS 8 Disk Operating System
1 set System Documentation
2 Diskettes
Multi-master Bus

Optional Items

Universal Prototyping Board
High-Speed Extender Board
Additional Serial and Parallel I/O Ports
Expansion Chassis with 2 floppy disk drives
CRT Terminal
Line Printers 60 CPS, 120 CPS, and 300 LPM
FORTRAN (8 bit)
Extended BASIC (8 bit)
PASCAL (8 bit and 16 bit)
COBOL (8 bit)
9511/9512 Arithmetic Processing Unit
8048, 8049, 8021 Macroassembler
8048, 8049, 8021 In-Circuit Emulation/TRACE
8080 In-Circuit Emulation/TRACE
8085 In-Circuit Emulation/TRACE
Z80 In-Circuit Emulation/TRACE

Options and Upgrades

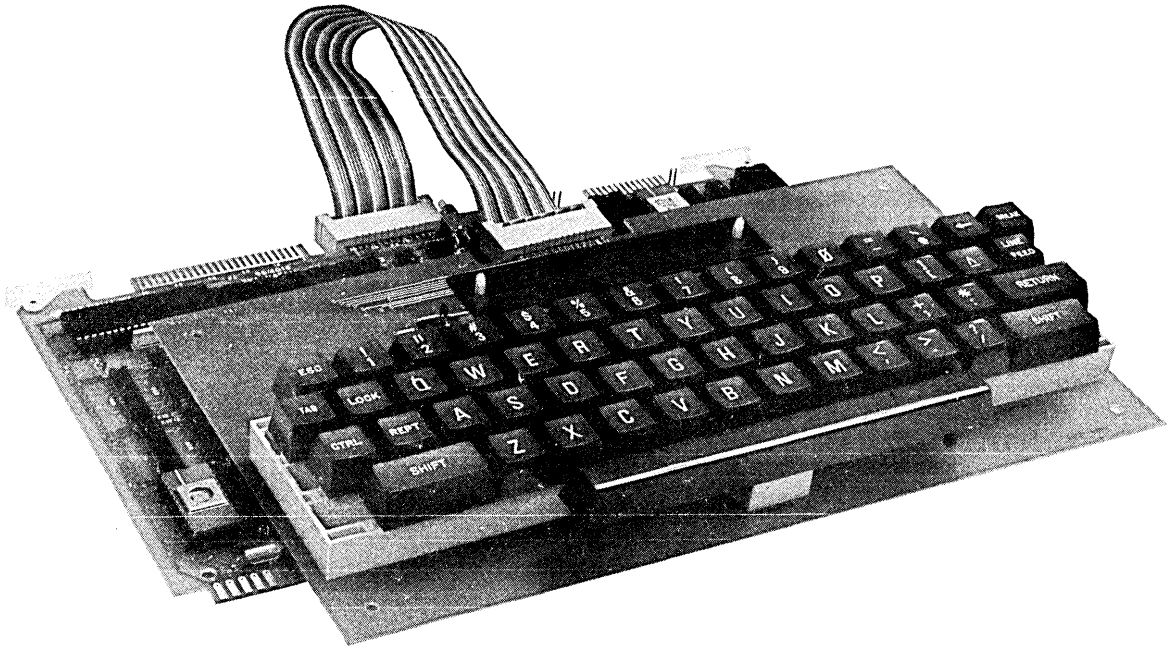
(not available with initial shipments—contact your AMD salesman for delivery information)

Z8000 In-Circuit Emulation/TRACE
Double Density Floppy Disk Drives
Expansion to 1 megabyte of Main Memory
AmZ8000 Microcomputer Board Upgrades
Cartridge Disk with Capabilities of 12 or 24 Megabytes
High Level 16-bit languages
Additional Expansion chassis

AmSYS 8/8 Ordering Information

PART NO.	DESCRIPTION
8/8010	Standard AmSYS 8/8 Development System 117V, 60 Hz
8/8012	Standard AmSYS 8/8 Development System 117V, 60 Hz Double Density Floppy Disks
8/8020	Standard AmSYS 8/8 Development System 220V, 50 Hz
8/8022	Standard AmSYS 8/8 Development System 220V, 50 Hz Double Density Floppy Disks
8/8030	Standard AmSYS 8/8 Development System 100V, 50 Hz
8/8032	Standard AmSYS 8/8 Development System 100V, 50 Hz Double Density Disks
8/3310	Serial I/O Board (4 Ports)
8/5032	ROM/EPROM Board I/O Board
8/6410	Univ. Prototyping Board
8/6420	High Speed Extender Board
8/8210	CRT with extra key pad
8/8310	Line Printer 120 CPS
8/8340	Line Printer 300 LPM
8/8410	Fortran 8
8/8420	Basic 8
8/8430	Cobol 8
8/8440	Pascal compiler (with code generators)
8/8610	Up/Down Load Execution Package w/8K bytes
8/8620	Up/Down Load Execution Package w/64K bytes
8/8800	In-Circuit Emulator Subsystem
8/8880	8080 In-Circuit Emulator POD
8/8885	8085 In-Circuit Emulator POD
8/8888	Z80 In-Circuit Emulator POD
8/8848	8048 In-Circuit Emulator POD
8/8050	Z8000 In-Circuit Emulator Subsystem
8/8250	Diskette Package of 10
8/8510	Optional Floppy Disk Chassis (adds add'l two Floppy Disk drives to system)

NOW! Evaluate the features of the AmZ8000 . . . develop software . . . execute programs with THE AMC 96/4016 EVALUATION BOARD.



The new generation of microprocessors is here. Now you can evaluate the AmZ8000 with the system-oriented **AMC 96/4016 Evaluation Board** that makes it easy to utilize the latest microcomputer technology. The **AMC 96/4016 Evaluation Board** puts a versatile and intelligent tool in the hands of engineers, designers and programmers allowing them to explore the exceptional capabilities of the AmZ8000. The **AMC 96/4016** integrates powerful hardware and extensive software resources on an assembled and tested printed-circuit board that allows the evaluation of the AmZ8000 by the addition of a power supply and I/O device. Power can be provided by plugging the board into an SBC 80 type card cage, AMC's development system, or with a lab supply. Two ports are provided to interface to a CRT terminal or to the 96/4016-KBD keyboard/display board.

Some of the features of the **AMC 96/4016 Evaluation Board** are:

- ☐ Fully assembled and tested computer board
- ☐ AmZ8002 Microprocessor – 4MHz operation
- ☐ 8K bytes (4K words) of RAM memory
- ☐ Sockets for up to 12K bytes of PROM/EPROM
- ☐ PROM-based monitor with debugging capability
- ☐ Two serial ports with programmable baud rates
- ☐ 24-line parallel port (three byte-wide ports)
- ☐ Three interval timers
- ☐ Optional PROM-based ASCII Assembler
- ☐ Interfaces for direct I/O to a CRT terminal or the 96/4016-KBD keyboard/display board
- ☐ SBC 80 physical size
- ☐ Can be used as an execution vehicle with the **AmSYS 8/8 Development System**
- ☐ Provides up-load/down-load capability with the **AmSYS 8/8 Development System**

The AmZ8000 architecture – in terms of CPU resources, instruction set, system interface and software-oriented features – represents a major advance in microprocessor sophistication and system-level performance. It is efficient enough to service simple tasks effectively, yet can easily handle complex, high-performance applications as well.

The AmZ8000 architecture and partitioning is well suited for today's technology and for a very wide range of today's applications. It is also a significant departure from the constraints of past architectures, and establishes a clean attractive and nearly open-ended base for evolution and development.

The **AMC 96/4016 Board** makes use of these features and assists software and system engineers in evaluating these features of the AmZ8000 for existing and future needs. Moreover, with the cleaner architecture of this new-generation microprocessor, the Evaluation Board can help hardware and software professionals develop better interface circuits and programs.

ABUNDANT CPU RESOURCES

The AmZ8000 offers sixteen 16-bit general-purpose registers in addition to special system registers. All 16 registers may be used as accumulators and all but one can serve as index registers. The first eight of these 16-bit registers may be used as sixteen 8-bit byte registers if needed. The AmZ8000 also supports seven main data types; bits, BCD digits, bytes, words (16 bits), long words (32 bits), byte strings and word strings. Additionally, many other data elements such as memory addresses, I/O addresses, segment table entries and program status words are also provided.

The **AMC 96/4016** gives the user access to the AmZ8000 and the ability to reduce programming overhead and shorten product and project development time. Hands-on experience with the Evaluation Board can help demonstrate the effectiveness of the AmZ8000 to provide fewer program modifications and less debug time.

Compared to other microprocessors or even 16-bit minicomputers, the number and power of individual instructions have greatly increased. Over 110 distinct instruction types are available with the AmZ8000, compared to approximately 60 for the PDP 11/45. With few exceptions, byte, word and long-word data elements can be processed by all the instructions. Each instruction – again with few exceptions – can use any of the five main addressing modes.

System designers, and especially programmers, will find the **AMC 96/4016 Board** useful in evaluating the AmZ8000's instruction set and its ability to generate higher code densities that can result in significant memory savings and shorter execution times.

HIGHER THROUGHPUT COMPILERS

Many applications normally involve high-level languages, operating systems and data-base management. The AmZ8000, with proven N-channel MOS technology and a 4MHz clock, allows the use of lower-cost dynamic RAMs. The AmZ8000 overlaps instruction execution with next instruction fetch to avoid the problems associated with deep unconditional prefetching.

The AmZ8000 can achieve this high degreee of performance because its regular architecture does not have critical bottlenecks and because the sophisticated instruction set substantially reduces the number of instructions. Some examples of this sophistication are:

a. 32-bit operations (including multiply and divide in single instructions)
b. String manipulation, including compare & translate
c. Block I/O instructions
d. Direct addressing of the entire memory
e. Two operating modes (systems/normal or supervisor/user)
f. Powerful interrupt handling

The **AMC 96/4016 Evaluation Board** makes effective use of these features so designers and managers can interpret these appealing features in specific terms and assess the capability for improved systems that can be designed quicker, easier and with more efficient results.

The AmZ8000 is designed to span a wide variety of applications. It's features allow it to be used effectively in complex high-throughput systems, yet it remains efficient for simpler systems as well.

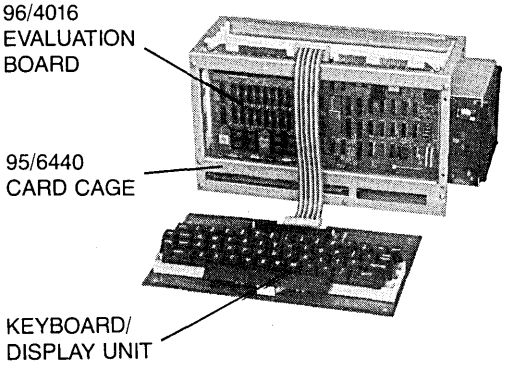
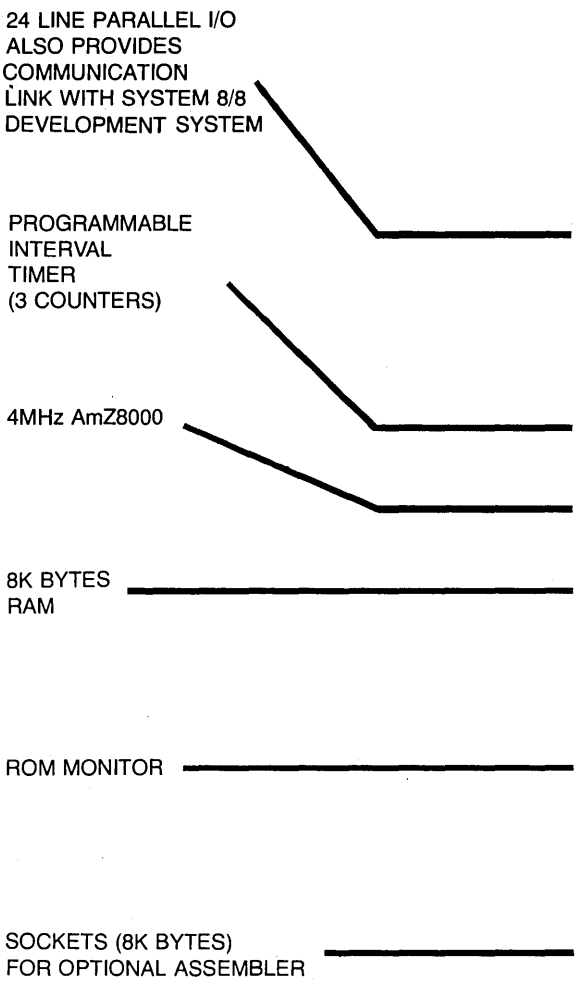
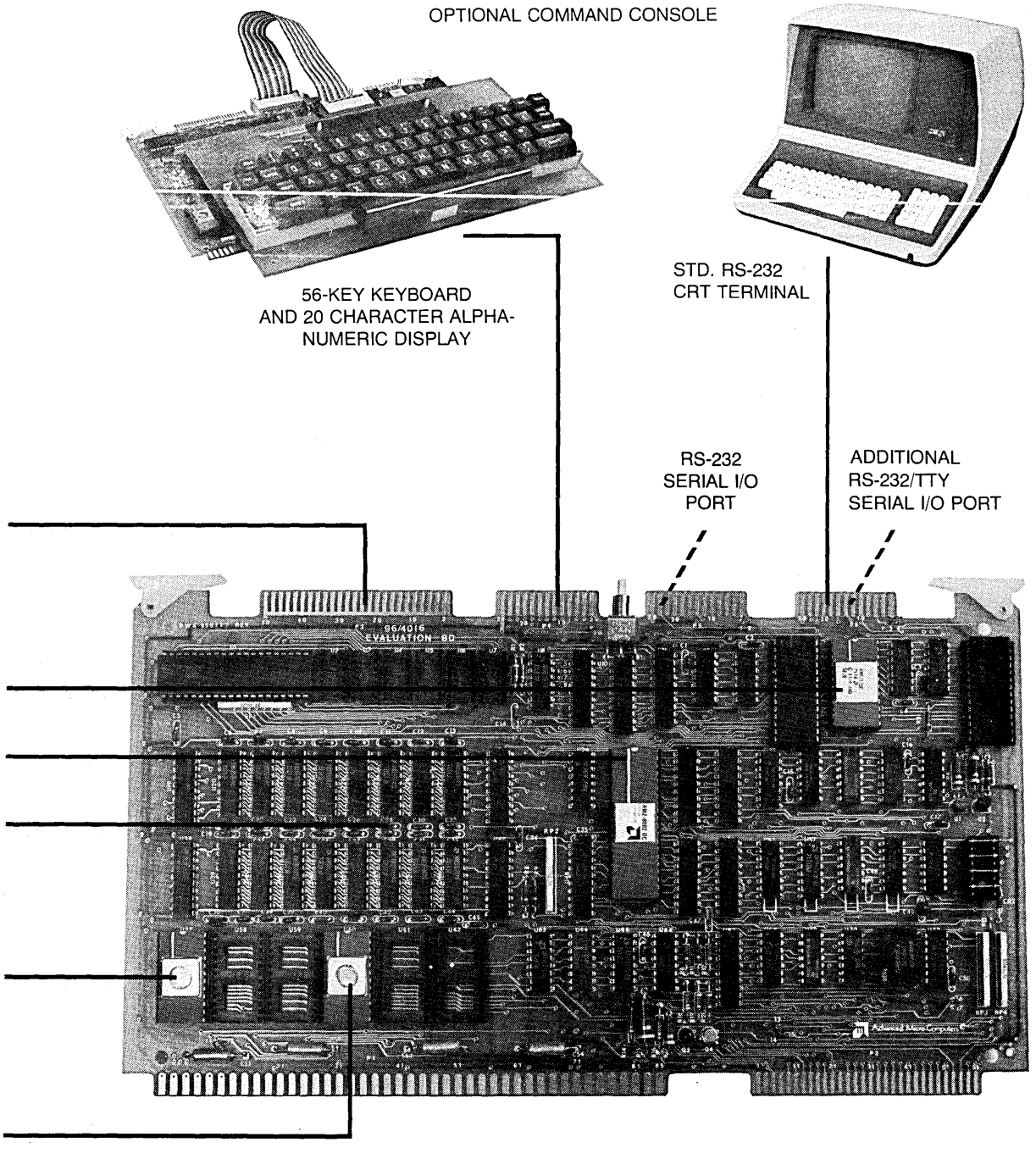


Photo showing use of the AMC 95/6440 Card Cage to house and power the 96/4016 Evaluation Board. Optional Keyboard/Display Unit shown above (or standard CRT Terminal) attaches via cable to edge-card connector. This arrangement provides both convenience and expansion capabilities for specific project and product development.

THE AmZ8000 EVALUATION BOARD NOW FOR BEST RESULTS

SPECIFICATIONS: AMC 96/4016 EVALUATION BOARD	
CPU	AmZ8000, non-segmented
Time Base	Crystal controlled; 3.9936MHz
Serial I/O	Two RS-232C serial ports with software programmable baud rates (50-9600). One port jumper selectable for 20mA operation.
Parallel I/O	24 parallel I/O lines. Also provides interconnection to AmSYS 8/8 Development System.
RAM Memory	8K bytes of on-board dynamic memory; CPU refreshed (transparent).
PROM Space	12K bytes of PROM/ROM Space provided; six sockets; ROM monitor occupies two sockets.
Counter/Timer	Three programmable 16-bit interval counters; two counters used for baud rate control; third counter available for user programs.
Power	+12VDC; −12VDC; +5VDC
Dimensions	12.0" (305 mm) x 6.75" (172 mm); SBC 80 form factor with six edge-card connectors
Memory Mapping	ROM monitor: 0-0FFF(H) PROM space: 0-2FFF(H) RAM space: 4000-5FFF(H)
Environmental Conditions	0 to 55°C ambient in free-air space with relative humidity to 90% without condensation.
Edge-of-Card Connectors	P1: 86-Pin, for power and ground P2: 60-Pin; CPU bus P3: 50-Pin parallel I/O for up/down link as execution vehicle P4: 26-Pin; interface for optional keyboard and display P5: 26-Pin; RS-232 and counter/timer interface P6: 26-Pin; RS-232 or 20mA current loop for CRT terminal or TTY as command console.
Monitor	4K PROM monitor included
Assembler	Optional ASCII, one-pass, line assembler in EPROM
Up/Down-Load Capability	Can be plugged directly into AmSYS 8/8 Development System to provide up-load and down-load capability.
Execution Vehicle	Can be used with other computer systems to execute AmZ8000 code.
Optional Keyboard/ Display	56-key keyboard; 20-character alpha-numeric LED display; same physical form as 96/4016 Board with attaching standoff connectors and interconnecting ribbon cable.




OPTIONAL COMMAND CONSOLE
56-KEY KEYBOARD
AND 20 CHARACTER ALPHA-
NUMERIC DISPLAY
STD. RS-232
CRT TERMINAL
RS-232
SERIAL I/O
PORT
ADDITIONAL
RS-232/TTY
SERIAL I/O PORT
96/4016
EVALUATION BD

7

With Powerful Software Support . . .

AMC 96/4016 ROM MONITOR

The Monitor Program provides the capability to examine and change data in RAM and in any and all of the AmZ8000 registers. It features a hardware breakpoint command along with a single-step routine to provide precise monitoring and debugging capability through step-by-step execution of AmZ8000 code. Additional commands allow users to load registers from memory and execute routines. The combination of 'Breakpoint/Load/Single Step' (with display) provides a versatile and efficient debugging capability. This monitor also supports the Evaluation Board when it is used with Advanced Micro Computer's **AmSYS 8/8 Development System** as an up/down-load device through its SAVE (to disk) and LOAD (disk to RAM) commands. The monitor resides in 4K bytes of EPROM and uses two of the six ROM/EPROM sockets available on the Evaluation Board.

AMC 96/4016 ASSEMBLER

The **AMC 96/4016 Evaluation Board** is supported by a one-pass line-by-line ASCII code Assembler (AMC 96/4016-ASM). This PROM-resident assembler provides the capability to enter symbolic programs into RAM and translates mnemonic op codes, symbolic labels and symbolic or absolute operands to machine code. The assembler reads user-supplied symbolic assembly statements from the command console and assembles the statements as received. Diagnostic messages are displayed when incorrect statements are entered as well as at the end-of-assembly for unsatisfied references with associated locations. Forward references are also permitted. The assembler features:

- ☐ A set of ten special characters for syntax and definition
- ☐ Seven pseudo op codes
- ☐ Six register formats to handle various word lengths
- ☐ Six direct-addressing formats to reference counter locations, labels and strings (ASCII, decimal, hex)
- ☐ Eight diagnostic messages for errors involving syntax, system operation, duplication, overflow and undefined references.

. . . and many ways to use it

AS A STANDALONE MICROCOMPUTER

The **96/4016 Evaluation Board**, along with available options, has the capability of standalone operation. On-board resources, including RAM, I/O interfaces, monitor and interval timer, allow it to serve as a self-contained single-board computer. A terminal or optional keyboard/display can be attached as a command console. Communications to peripherals and other equipment can be achieved through the I/O ports.

EXECUTION DEVICE WITH CROSS-ASSEMBLERS

Many users will want to evaluate the AmZ8000 execution speed and throughput performance with respect to existing programs and system designs. The **AMC 96/4016** is designed to serve as a vehicle to execute AmZ8000 machine codes that may have originated from development systems and cross assemblers. I/O ports on the Evaluation Board provide the necessary parallel and serial interfaces. AmZ8000 CPU signals are brought out to an edge connector which allows the needed connection for other circuitry.

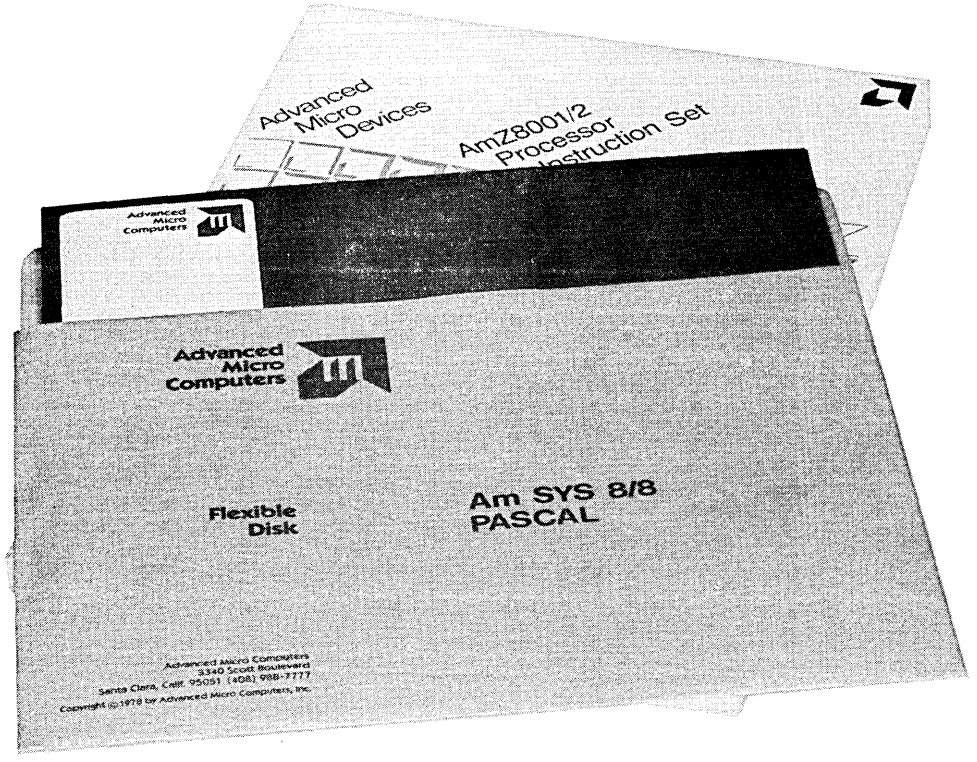
UP-LOAD/DOWN-LOAD CAPABILITY

The **AMC 96/4016 Evaluation Board** features both up-load and down-load capability when used as an execution vehicle with other computer systems, such as the AmSYS 8/8. Programs generated in a development system can be down-loaded to the RAM on the Evaluation Board. Similarly, programs in the board's RAM memory can be up-loaded to the system for further development and for disk storage.

The Evaluation Board is hardware and software compatible with Advanced Micro Computer's **AmSYS 8/8 Development System**. It can plug directly into the AmSYS 8/8 as an execution vehicle to run developed AmZ8000 code. The on-board ROM monitor provides SAVE and LOAD commands that control bidirectional data flow between the RAM on the Evaluation Board and disk files on the Development System.

The **AMC AmSYS 8/8 Development System** contains a comprehensive set of hardware and software resources to fully utilize AmZ8000 capabilities. The system contains dual floppy-disk drives, 64K bytes of RAM, serial and parallel ports, hardware computation and an SBC 80 Multi-Master bus. Existing programming support includes an operating system with linking loader, editor and debugger and, of particular interest, an AmZ8000 macroassembler, 8080 macroassembler and AmZ8000 translator. High-level languages, including PASCAL, are available. See the AmSYS 8/8 brochure for more details or contact one of the sales offices listed on the rear of this brochure.

AMC's PASCAL
Compiler



7

Features

- ☐ The Compiler Runs in the AmSYS 8/8 or the AmSYS 29 Microcomputer Development Systems Environments
- ☐ Object Program may be one of the Following:
  - P-code (Executed by an Interpreter)
  - Am9080 Machine Code
  - AmZ8000 Machine Code
- ☐ Block Structured High Level Language that Supports Structured Design and Programming
- ☐ Many Extensions such as Separate Compilation, Strings, etc.

The AMC PASCAL compiler gives the AmSYS 8/8 user a powerful language for software development. The PASCAL language offers structured design capabilities to the user. The AMC PASCAL user can work with data of varying types and can manipulate sets, arrays, records and files with great flexibility. A large number of intrinsic functions and procedures are available to the user. In general, PASCAL provides powerful language constructs that are combined with an efficient block structure to produce PASCAL programs.

PASCAL is fast becoming a widely used language on microcomputers. The PASCAL language itself is an Algol derivative invented by Niklaus Wirth. AMC PASCAL is a PASCAL implementation with additional features that extend the capabilities for a broad range of software development.

Producing an AMC PASCAL program is a two-step process. First the source program is converted (compiled) into an intermediate pseudo code (P-code). At this point, the user has the following choice: to execute the P-code interpretively on AmSYS 8/8, or to generate native machine code for the 8080 or AmZ8000 microprocessor (see Figure 1).

Interpretive Execution

The AMC PASCAL user can compile a PASCAL program and then execute the compiled program on AmSYS 8/8 with the PASCAL interpreter and run-time library. The AMC P-code is compatible with Version 1.4 of the P-code developed at the University of California at San Diego. In this mode of execution, the interpreter in fact simulates in AmSYS 8/8 the run time environment of a P-machine which is the host hardware to which P-code is native. A run time library supports the interpreter in executing built-in functions or procedures (intrinsics) or system defined operators (see Figure 2).

Code Generation

Alternatively, the P-code generated by the compiler may be converted into object code for the 8080 or AmZ8000 microprocessors. The code generation process involved is as follows (see Figure 3). First the P-code file is translated to produce an assembly language file which can then be assembled by MACRO8 for the 8080 or MACRO8000 for AmZ8000 to produce a relocatable object module. This module contains only the user written part of the PASCAL program; all the intrinsic operators and functions are as yet undefined external subroutines at this point. These subroutines have been prewritten and preassembled, and are contained in a library file. The runtime routines in this library are then selectively combined with the user object module through a linker, generating an executable object code file.

Data Types

PASCAL is very specific about the "type" of a variable. The type defines precisely the set of values a variable may assume (its domain), which in turn indirectly determines the set of operators that may act upon it. In addition to the predefined standard types, the user may create his own types of data. A variable may also be either static or dynamic, defining storage requirement at compile time or run time respectively. With such rich data typing and data structuring facilities, the programming job becomes much easier for the user.

- Integer – A 16-bit number, $-32768 \leq n \leq 32767$
- Real – A 32-bit floating point number, $10^{-38} \leq n \leq 10^{38}$
- Boolean – True or false
- Character – Alphabets, digits, blank and some special characters
- Scalar – Domain is user program defined; this allows great flexibility in creating new data types and provides a means for a more explanatory statement of the problem
- Subrange – A proper subset of a previously defined type; increases program clarity by being very specific about the intended domain of variable
- Set – A collection of components of the same type; components in a set are also ordered
- Array – A fixed number of components arranged in consecutive order either linearly or as a multidimensional matrix; any arbitrary element may be randomly accessed directly
- File – A structure consisting of components which may be accessed sequentially
- String – A linear array of characters
- Record – A structure consisting of a fixed number of unordered components which need not be of the same type
- Pointer – A dynamic structure bounded only to a type at compile time and remains otherwise free until binding to a specific variable occurs at runtime when memory space is allocated; very useful in creating and processing data structures like linked lists and trees

Functions and Procedures

A function is a "subroutine" that returns a single value. Syntactically, a function call may appear anywhere in place of a variable within an expression. Its effect as a constituent of an expression amounts to the result value that it yields. A procedure, however, is a "subroutine" that may indirectly return more than a single value. Syntactically, it is a stand alone statement. Some common properties of both are given below:

- May be recursive, very useful in system programming
- Parameters may be passed by "value"; a parameter, possibly an expression, is evaluated at call time to a value which is bounded to the formal parameter at that time and remains unchanged
- Parameters may also be passed by "variable"; at call time, the formal parameter is bounded only to a variable, whose value does not necessarily remain constant throughout the function or procedure

Dynamic Memory Allocation

Being a block structured language, scopes of variables are limited to procedures or functions within which they are declared, and have no significance externally. As long as a procedure or function has not been activated, variables local to it need not exist. The P-code has been designed to take advantage of this fact. Memory spaces for variables are allocated on top of stack only upon a function or procedure entry, and relinquished automatically upon exit. Expression evaluation is also performed on this stack.

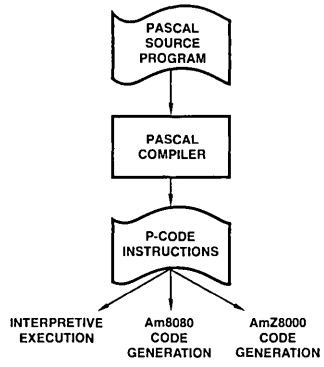


FIGURE 1. The Compilation Process.

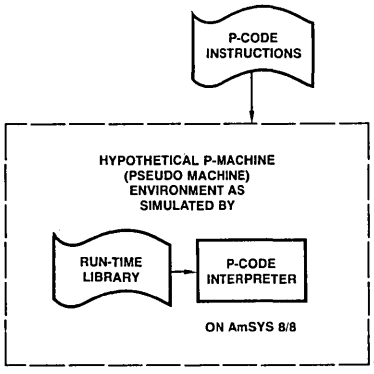


FIGURE 2. Interpretive Execution.

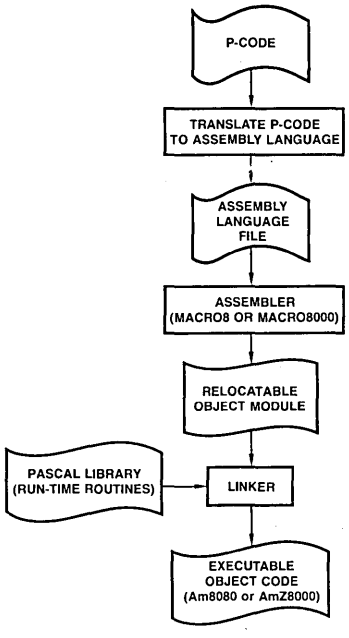


FIGURE 3. Code Generation.

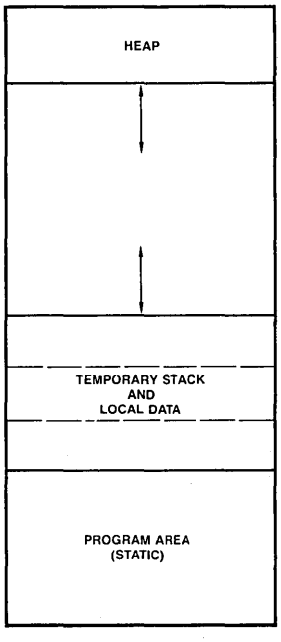


FIGURE 4. Dynamic Memory Allocation.

Therefore the total memory space required for data storage dynamically expands and contracts at run time coinciding with function or procedure entries and exists, taking up memory space only as needed. This allows execution of much larger programs than would otherwise be possible.

It is also possible to dynamically create, at run-time, data structures such as lists and trees whose storage space sizes vary throughout their lifetimes. In order to avoid conflict, memory allocation for this purpose is at the opposite end of memory (heap) for optimum memory utilization. Intrinsics are provided within the language for user control (MARK and RELEASE) of the heap.

AMC PASCAL Operators and Intrinsics

A summary of operators that may be used in an expression is shown in Figure 5. An intrinsic is a system defined function or procedure. A PASCAL intrinsic function is a function in the mathematical sense; its net effect is the result value yielded without any other side effects. A summary of intrinsic functions is shown in Figure 6. An intrinsic procedure produces either return values through variable parameters, or its purpose may be to produce some side effects, or it may be a combination of both. A summary of intrinsic procedures is shown in Figure 7.

Operator	Operation	Type of Operands	Result Type
Definition:			
:=	assignment	any type but files	as appropriate
Arithmetic:			
+(unary)	identity	INTEGER or REAL	same as operand
−(unary)	sign conversion	INTEGER or REAL	same as operand
+	addition	INTEGER or REAL	INTEGER or REAL
−	subtraction	INTEGER or REAL	INTEGER or REAL
*	multiplication	INTEGER or REAL	INTEGER or REAL
/	REAL division	INTEGER or REAL	REAL
DIV	INTEGER division	INTEGER	INTEGER
MOD	modulus	INTEGER	INTEGER
Relational:			
=	equality	scalar or string	BOOLEAN
<>	inequality	set or pointer	BOOLEAN
<	less than	scalar or string	BOOLEAN
>	greater than	scalar or string	BOOLEAN
<=	less or equal	scalar or string	BOOLEAN
<=	set inclusion	set	BOOLEAN
>=	greater or equal	scalar or string	BOOLEAN
>=	set inclusion	set	BOOLEAN
IN	set membership	scalar and set type	BOOLEAN
Logical:			
NOT	negation	BOOLEAN	BOOLEAN
OR	disjunction	BOOLEAN	BOOLEAN
AND	conjunction	BOOLEAN	BOOLEAN
Set:			
+	union	any set of type T	T
−	set difference	any set of type T	T
*	intersection	any set of type T	T

FIGURE 5. Summary of Operations.

String:	Character Array:	Input/Output:	Mathematical:
LENGTH	SCAN	EOF	ABS
POS	SIZE OF	EOLN	SQR
CONCAT		IORESULT	SQRT
COPY			SIN
			COS
			ARCTAN
			EXP
			LN
			PWROFTEN

FIGURE 6. Summary of Intrinsic Functions.

String:	Character Array:	Input/Output:	Dynamic Memory Allocation:
DELETE	MOVELEFT	RESET	NEW
INSERT	MOVERIGHT	REWRITE	MARK
		UNITREAD	RELEASE
		UNITWRITE	
		BLOCKREAD	
		BLOCKWRITE	
		CLOSE	
		GET	
		PUT	
		READ	
		READLN	
		WRITE	
		WRITELN	
		PAGE	
		SEEK	

FIGURE 7. Summary of Intrinsic Procedures.

Compiler Options

AMC PASCAL provides a number of compilation options for the user. The most important options are:

GOTO – PASCAL GOTO statements can be disabled within the program.

IOCHECK – Code can be added by the compiler to verify the completion of I/O Operations.

INCLUDE – Different source files can be included during the compilation process.

LISTING – Listing can be selectively disabled.

QUIET COMPILATION – Console device output during compilation can be suppressed.

RANGECHECK – Code can be added by the compiler to check array subscript range and assignment to variables of subrange types.

Assignment
Procedure/Function calls
IF . . . THEN . . . ELSE
CASE . . . OF . . . END
WHILE . . . DO . . .
REPEAT . . . UNTIL . . .
FOR . . . TO . . . DO . . .
FOR . . . DOWNTO . . . DO . . .
WITH . . . DO . . .
GOTO

FIGURE 8. Summary of Statement Types.

```
  1    4     1:D      1 PROGRAM SPECIALMENU (OUTPUT);
  2    4     1:D      3
  3    4     1:D      3 TYPE DAYTYPE = (MON,TUE,WED,THUR,FRI,SAT,SUN);  (* SCALAR *)
  4    4     1:D      3      WKDAYTYPE = MON..FRI;                      (* SUBRANGE *)
  5    4     1:D      3      WKDAYSET = SET OF WKDAYTYPE;               (* SET *)
  6    4     1:D      3      FOODTYPE = (LAMB,STEAK,CRAB);              (* SCALAR *)
  7    4     1:D      3
  8    4     1:D      3 VAR DAY: DAYTYPE; SPECIAL: FOODTYPE; WKDAY: WKDAYSET;
  9    4     1:D      6     DAYS: ARRAY[MON..SUN] OF STRING;            (* ARRAY *)
 10    4     1:D    293
 11    4     2:D      1 PROCEDURE DISPLAY(PD:DAYTYPE; PS:FOODTYPE);     (* PROCEDURE *)
 12    4     2:C      0     BEGIN WRITE(OUTPUT,DAYS[PD]);
 13    4     2:C     17           CASE  PS OF                           (* CASE STATEMENT *)
 14    4     2:C     20                 STEAK: WRITELN(OUTPUT,'STEAK');
 15    4     2:C     46                 LAMB:  WRITELN(OUTPUT,'LAMB');
 16    4     2:C     71                 CRAB:  WRITELN(OUTPUT,'CRAB')
 17    4     2:C     94           END
 18    4     2:C    110     END;
 19    4     2:C    122
 20    4     1:C      0 BEGIN                                            (* MAIN PROGRAM *)
 21    4     1:C      0       DAYS[MON] :='MONDAY    - ';               (* STRING ASSIGNMENT *)
 22    4     1:C     26       DAYS[TUE] :='TUESDAY   - ';  DAYS[WED] :='WEDNESDAY - ';
 23    4     1:C     74       DAYS[THUR]:='THURSDAY  - ';  DAYS[FRI] :='FRIDAY    - ';
 24    4     1:C    122       DAYS[SAT] :='SATURDAY  - ';  DAYS[SUN] :='SUNDAY    - ';
 25    4     1:C    170       WKDAY:=[MON,TUE,WED,THUR,FRI];             (* SET ASSIGNMENT *)
 26    4     1:C    176       WRITELN(OUTPUT,'TODAY''S SPECIAL');
 27    4     1:C    210       WRITELN(OUTPUT,'');
 28    4     1:C    229       FOR DAY := MON TO SUN DO
 29    4     1:C    243           BEGIN  IF DAY IN WKDAY                 (* SET MEMBERSHIP *)
 30    4     1:C    244                      THEN CASE DAY OF
 31    4     1:C    252                                MON,WED,FRI: SPECIAL:=STEAK;
 32    4     1:C    257                                TUE,THUR:    SPECIAL:=LAMB
 33    4     1:C    257                           END
 34    4     1:C    280                      ELSE SPECIAL:=CRAB;
 35    4     1:C    285                 DISPLAY(DAY,SPECIAL)           (* PROCEDURE CALL *)
 36    4     1:C    287           END
 37    4     1:C    289 END.

SYSTEM 8 PASCAL RUNNING.....

TODAY'S SPECIAL

MONDAY    - STEAK
TUESDAY   - LAMB
WEDNESDAY - STEAK
THURSDAY  - LAMB
FRIDAY    - STEAK
SATURDAY  - CRAB
SUNDAY    - CRAB
```

FIGURE 9. Example PASCAL Program – Data Types.

```
 1   4    1:D     1
 2   4    1:D     1 PROGRAM BUBBLESORT;
 3   4    1:D     3
 4   4    1:D     3 VAR      SWAPS:  INTEGER;
 5   4    1:D     4          THIS:   INTEGER;
 6   4    1:D     5          THISVAL:CHAR;
 7   4    1:D     6          POS:    INTEGER;
 8   4    1:D     7          BUFFER: STRING;
 9   4    1:D    48
10   4    1:C     0 BEGIN
11   4    1:C     0
12   4    1:C     0 (* READ INPUT LINE *)
13   4    1:C     0 READLN(INPUT,BUFFER);
14   4    1:C    21  * SAVE CHARACTER COUNT *)
15   4    1:C    21 POS := LENGTH(BUFFER);
16   4    1:C    26
17   4    1:C    26 (* IF LINE HAS > 1 CHARACTER *)
18   4    1:C    26 IF POS > 1
19   4    1:C    27 THEN
20   4    1:C    31          (* SORT CHARACTERS IN THE LINE *)
21   4    1:C    31          REPEAT
22   4    1:C    31          BEGIN
23   4    1:C    31          (* RESET SWAP COUNTER *)
24   4    1:C    31          SWAPS := 0;
25   4    1:C    34          FOR THIS := 1 TO (POS - 1) DO
26   4    1:C    48                  (* CHECK EACH PAIR OF CHARACTERS *)
27   4    1:C    48                  IF BUFFER[THIS] > BUFFER[THIS + 1]
28   4    1:C    59                  THEN
29   4    1:C    63                          BEGIN
30   4    1:C    63                          (* SWAP THE CHARACTERS *)
31   4    1:C    63                          THISVAL := BUFFER[THIS];
32   4    1:C    70                          BUFFER[THIS] := BUFFER[THIS + 1];
33   4    1:C    82                          BUFFER[THIS + 1] := THISVAL;
34   4    1:C    90                          SWAPS := SWAPS + 1
35   4    1:C    91                          END;
36   4    1:C   122          END;
37   4    1:C   102          UNTIL SWAPS = 0;
38   4    1:C   107
39   4    1:C   107 (* WRITE OUT SORTED LINE *)
40   4    1:C   107 WRITELN(OUTPUT,BUFFER);
41   4    1:C   126
42   4    1:C   126 END.

SYSTEM 8 PASCAL RUNNING... ..

THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG
       ABCDDEEEEEFGHEIJKLMNOOOPQRRTTUUVWXYZ
```

FIGURE 10. Example PASCAL Program – Bubble Sort.

AMC PASCAL Extensions

AMC PASCAL has a number of extensions to the standard PASCALs defined by Jensen and Wirth. The most important extensions are:

- Separate compilation – A PASCAL program may be written in many UNITS and compiled separately one at a time. This also allows linkage of assembly language subroutines to PASCAL programs.
- Interactive files – An additional predeclared type INTERACTIVE is provided, and the files INPUT and OUTPUT are automatically typed as INTERACTIVE.
- Untyped files – The BLOCKREAD and BLOCKWRITE intrinsics are provided for read and write operations on untyped files.
- Random access – For typed files, the SEEK intrinsic provides random access to records. For untyped files BLOCKREAD and BLOCKWRITE provide random access.
- Strings – An additional predeclared type STRING is provided. The STRING type is essentially equivalent to PACKED ARRAY OF CHAR.
- EXIT – Before the normal end of any procedure or function, the user can exit with the EXIT intrinsic.
- Comparisons – The user can compare entire arrays or records with the equal and not equal operators.
- Dynamic Memory Allocation – The MARK and RELEASE intrinsics are provided instead of the DISPOSE intrinsic. MARK and RELEASE are used to set a pointer in the user heap and then release heap space from the pointer to the top of the heap.

What is PASCAL and P-Code

PASCAL is a block structured high level language with strong typing of variables. Virtues of block structured languages are much discussed and well understood. In addition, PASCAL offers rich data types that are not found in most languages. Some more notable ones are: set, scalar, structured array and record, and dynamic pointer. This allows data structures to be constructed both statically and dynamically making the language a very useful tool not only at the application level, but at the systems programming level as well. The major strength of PASCAL, however, lies in the fact that it is a powerful language even with a relatively few number of basic constructs. This is attributed significantly to its powerful and flexible data types. As a result, PASCAL has gained the following advantages: easy to learn, simple to program, straight-forward to compile, efficient object code, fast execution, program clarity, self documenting and easy to maintain.

Being a block structured language, it lends itself to efficient execution on a host CPU with a stack architecture. Instructions of a stack machine typically require zero operand specification since they are implicitly assumed to be on top of the stack. As a result, the instructions are very compact.

A companion part to the language definition of PASCAL itself is just such a pseudo machine level instruction set (called P-code) for a hypothetical pseudo stack machine (P-machine). The instruction set has been tailor designed to represent PASCAL programs very compactly and concisely.

Due to this well defined transformation from PASCAL statements to P-code instructions, a compiler can be written very efficiently to generate P-code instructions from PASCAL source programs. Execution of a PASCAL program via its P-code equivalent can then be done in one of two ways. It can be either executed in its native P-machine environment as simulated by a software interpreter, or alternatively, P-code instructions may be translated into native instructions of host CPUs such as AmS080 or AmZ8000 to be executed.

The AMC PASCAL is upward compatible with that developed at the University of California at San Diego both at the source program level and at the P-code level. It includes a number of extensions to the standard PASCAL to enhance its usefulness in the microcomputer environment where hardware resources are more directly controlled by programmer at a lower level.

Specifications:

Operating Environment

Required Hardware

AmSYS Development System:
AmSYS 8/8, AmSYS 29

64K bytes of RAM memory

Floppy diskette drive, single or double density

System console: CRT or interactive hard copy device

Required Software

64K version of AMDOS operating system

Shipping Media

Flexible diskette

Reference Manuals

AmSYS PASCAL User's Manual 00680127

Ordering Information

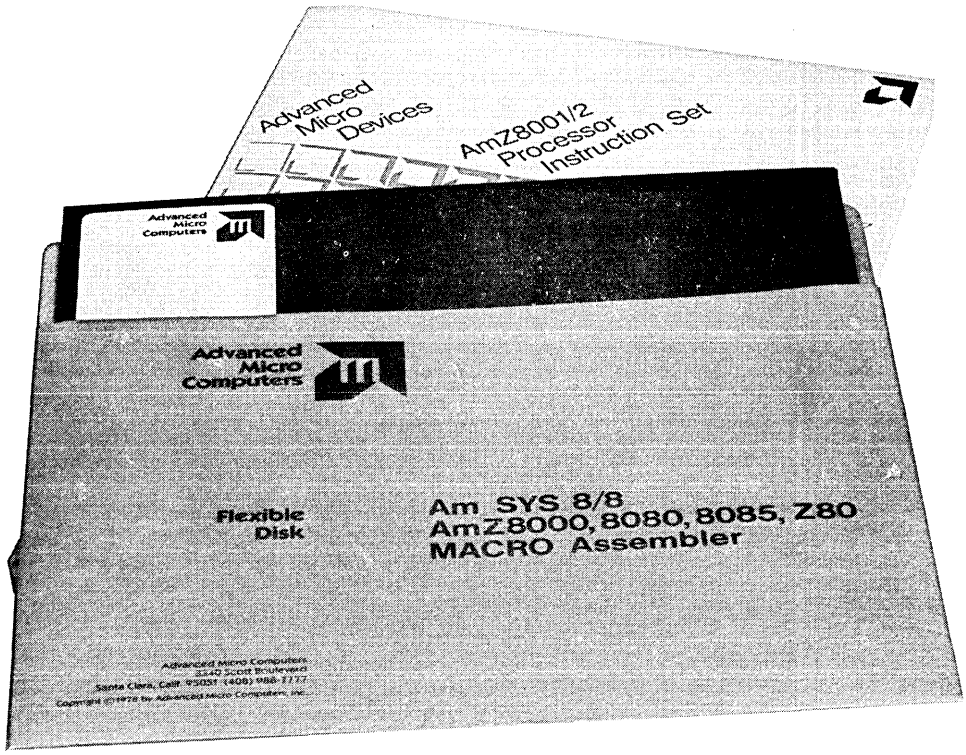
AmSYS Software can only be purchased as part of or as an addition to an AmSYS 8 or AmSYS 29 computer development system.

Part Number	Description
8/8440†	AMC PASCAL compiler, with interpreter, run-time library, Am9080 code generator and AmZ8000 code generator
298450*	AmZ8000 16-bit macroassembler; 8080, 8085 and Z80 8-bit assemblers

†298440 for AmSYS29.

*Provided as part of the AmSYS 8/8 Software package.

AMC's MACRO8000
AmZ8000 Macro Assembler



Features

- ☐ Functions in the AmSYS 8/8 or AmSYS 29 microcomputer development system environments
- ☐ Generates absolute or relocatable AmZ8000 object code
- ☐ Supports the full AmZ8000 instruction set
- ☐ Symbolic operands that are constants or variables
- ☐ Syntactic macros
- ☐ Program segmentation for address space control
- ☐ Arithmetic expressions, string expressions, and comparisons
- ☐ Superior assembly speed and a variety of assembly options

The AMC MACRO8000 assembler gives the AmSYS 8/8 or AmSYS 29 user a dynamic way to develop AmZ8000 code for any application. A typical MACRO8000 program is a combination of AmZ8000 assembly instructions and higher-level constructs. MACRO8000 provides great flexibility in structuring the program and in fine-tuning to produce the most efficient AmZ8000 code for the application.

Assembler Output Format

MACRO8000 produces an output file of either absolute or relocatable AmZ8000 code. The user can choose to write absolute or relocatable code for either the nonsegmented AmZ8002 or segmented AmZ8001 processor. An absolute source program is monolithic in structure, and the program is assembled as a single unit. The output from the assembler can be an object file which can be loaded into the target system for the execution. A modular source program is relocatable and the user can choose to write separate modules or routines that are later merged into a coherent program. A linking procedure handles the merging of the program components and produces a single relocatable file containing the program modules linked together. A loading procedure assigns absolute addresses to a relocatable file and loads the resulting object file into memory for subsequent execution in the target system.

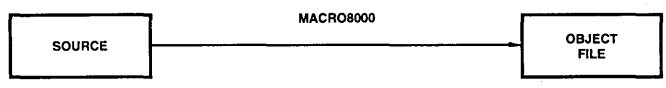
Instruction Format

The full AmZ8000 instruction set can be used in writing programs. The general format of an instruction is the opcode (instruction mnemonic such as ADD) followed by the operands. MACRO8000 generates the appropriate bit pattern for each instruction, which is 1 to 5 words in length. The user can specify whatever operands are appropriate for the instruction. AmZ8000 instructions have from zero to four operands, and the operands can be in various addressing modes.

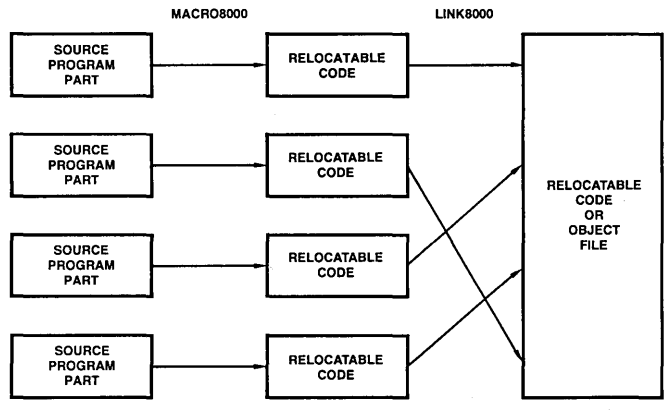
Addressing Modes

The AmZ8000 addressing modes are for different types of operands and are supported through the hardware. The addressing modes give the user great flexibility in choosing whether an operand is directly or indirectly specified to be in a register, in memory or in part of the instruction.

- The immediate mode specifies an operand which is a constant imbedded within the instruction itself.
- The register mode specifies an 8-bit (byte) register, 16-bit (word) register, 32-bit (long word) register, or 64-bit (quad word) register that contains the operand value.
- The indirect register mode specifies a word register (nonsegmented AmZ8002) or register pair (segmented AmZ8001) that contains the address of the operand.
- The direct address mode specifies a label. The label is associated with the address of a data value that is used as the operand, or the label is associated with an instruction to which a jump or call instruction branches to.
- The relative address mode specifies an operand whose location is found relative to the present instruction location.
- The indexed mode is a directly addressed operand followed by a displacement that is in a register. The label and displacement added together forms the address of the operand value.
- The base address mode consists of an indirect register specification followed by a displacement. The address of the operand is calculated by adding together the content of the register and the displacement value.
- The base indexed mode is a register indirect address specification followed by a displacement that is in another register. The address of the operand is calculated by adding together the contents of the two specified registers.



Monolithic Program for AmZ8002 or AmZ8001



Modular Program for AmZ8002 or AmZ8001

7

- The port immediate mode consists of a port address for I/O operations that is specified as a 16-bit immediate value.
- The port register mode specifies a register that contains a 16-bit port address for I/O operations.

Opcodes

The AmZ8000 instruction opcodes are the names for the individual instructions. The naming scheme involves a few characters for the basic name of the instruction, such as LD for load, and sometimes additional characters that specify other properties of the instruction. For instance, LDIRB is a Load with auto Increment and Repeat for Byte operands. Well over 200 different opcodes can be written.

- CLR – Clear (byte, word or long word)
- EX – Exchange (byte or word)
- LD –Load (byte, word, or long word; special form for multiple registers; special form for addresses; optional autoincrement or autodecrement; optional repeat)
- POP and PUSH – Pop and Push stack (word or long word)
- ADD – Add (byte, word, or long word; optionally with carry)
- CP – Compare (byte, word, or long word; optional autoincrement or autodecrement; optional repeat)
- DAB – Decimal adjust (byte)
- DEC – Decrement (byte or word)
- DIV – Divide (word or long word dividend)
- EXTS – Extend sign (byte, word or long word)
- INC – Increment (byte or word)
- MULT – Multiply (word or long word result)
- NEG – Negate (byte or word)
- SUB – Subtract (byte, word, or long word; optionally with carry)
- AND – Logical AND (byte or word)
- COM – Complement (byte or word)
- OR – Logical OR (byte or word)
- TEST – Logical test (byte, word, or long word)
- TCC – Test condition code (byte or word)
- XOR – Exclusive OR (byte or word)
- R – Rotate (left or right; special form for digit; optionally through carry; byte or word)
- S – Shift (left or right; optionally dynamic; arithmetic or logical; byte, word, or long word)
- BIT – Bit test (static or dynamic; byte or word)
- RES – Reset bit (static or dynamic, byte or word)
- SET – Set bit (static or dynamic, byte or word)
- TSET – Test and set (byte or word)
- CPS – Compare strings (autoincrement or autodecrement; optional repeat; byte or word)
- TR – Translate (autoincrement or autodecrement; optional repeat; byte)
- TRT – Translate and test (autoincrement or autodecrement; optional repeat; byte)
- IN – Input (optional autoincrement or autodecrement; optional repeat; optional special input; byte or word)
- OUT – Output (optional autoincrement or autodecrement; optional repeat; optional special output; byte or word)
- CALL – Call routine (normal or relative)
- RET – Return
- JP – Jump (normal or relative)
- DJNZ – Decrement and jump on non-zero (byte or word)
- SC – System call
- INRET – Interrupt return
- COMFLG, RESFLG, SETFLG – Flag complement, reset, or set
- DI, EI – Interrupt disable or enable
- HALT – Halt
- NOP – No operation
- MBIT, MRES, MSET, MREQ – Multi-micro test, reset, set, or request
- LDCTL – Load control (flag byte or special control word)
- LDPS – Load program status

High Level Constructs

MACRO8000 offers a variety of high-level constructs that can be used in assembler programs. High-level constructs make most programs easier to write, simpler to debug and maintain than functionally-equivalent programs containing only instructions. The constructs in MACRO8000 are either very similar or identical to PASCAL statements. Programmers familiar with PASCAL or another block-structured language will be comfortable with the MACRO8000 high-level constructs.

- IF-THEN-ELSE is for conditional assembly, or for run-time code generation. A test such as register equal to register generates code.
- FOR-DO is for repetitive assembly.

Symbolic Operands

MACRO8000 supports symbolic constants and symbolic variables. The values of symbolic constants and variables are essentially any operands that can be used in the program, including arithmetic expressions. The symbolic constants are names used for constant values. This provides for the assignment of meaningful names such as BLOCK_COUNT to a register used as the block count. The symbolic variables are defined and then throughout the program can be redefined to new values such as different registers. All of the evaluation of the variables occurs at assembly time, so that the generated code reflects the values substituted for the symbolic constants and variables.

Syntactic Macros

The macro capability allows the programmer to reduce the amount of source code that must be written. Identical or similar sections of code can be prepared as a macro and called out as appropriate. During assembly the call is replaced by the code of the macro body. The macros in MACRO8000 are internally processed as syntactic macros, which are assembled at up to over an order of magnitude faster than the conventional lexical macros.

Program Segmentation

The MACRO8000 relocatable module system is based on the concept of "segment" meaning "a piece or part of a program." Thus, a "program" may be defined as a collection or sequence of program segments. Named program segments may be assigned attributes, and arranged and combined in any arbitrary manner by LINK8000. This simple unifying concept applies to both the AmZ8001 and AmZ8002 CPUs. In the case of the AmZ8001 (segmented CPU), program segments may be identified with (i.e., equated to) logical segments. Program segmentation has many uses beyond the specification of logical segments for the AmZ8001 – for example, the partitioning of program space into RAM space and ROM space.

LINK8000 offers simple default modes of operation for the novice user. For more advanced users, LINK8000 offers incremental linking, providing an enhanced program segment mapping capability.

Expressions

The ability to write expressions often reduces the work involved in writing programs and helps to keep the programs smaller.

- Arithmetic expressions can be written to produce 32-bit signed results.
- Arithmetic signed or unsigned comparisons can be written to yield true or false values for logical tests.
- Strings can be used and concatenated in string expressions.
- String comparisons can be used to compare strings by collating sequence and yield true or false values for logical tests.

Assembly Options

In addition to fast assembly, with one pass through the source and a second pass for the listing, MACRO8000 offers a variety of options.

- Option to suppress first pass listing.
- Option to direct listing to the console, to the printer, to a default file, or to a user-specified file. The listing shows the source program, any diagnostic messages produced by the assembler, and the AmZ8000 code and data values generated by the program.
- Option for error messages only, and option to suppress warning messages.
- Option for macro trace that displays code expansion.
- Option for object file generation.

Linker

The LINK8000 linker is a separate module used for linking together relocatable modules of any size. The linker can bring in library routines to satisfy external references. The user can control the mapping of modules into the object file, as well as the mapping of user segments within modules. LINK8000 can also handle relative address references across segments. Output of the linker is an object file that can be run on either the nonsegmented AmZ8002 or the segmented AmZ8001.

```
EXAMPLES...                    MACRO 8/8000 Z8000 ASSEMBLER  1.2.1E   PAGE 1


0000                    MODULE 'EXAMPLES...';
0000
0000
0000                    (*      SYMBOLIC CONSTANTS AND NUMERIC VALUES ARE
0000                            ALWAYS TREATED THE SAME ...                    *)
0000
0000                    CONST   ABC     = #4000,
0000                            DEF     = 15,
0000                            REG     = R3,
0000                            RIR     = R4^,
0000                            RBS     = R5^(R7),
0000                            SLAB    = LABEL,
0000                            NLAB    = #4000^;
0000
0000
0000                    %       USE OF SYMBOLIC CONSTANTS:
0000
0000
0000   2103 4000                LD R3,#4000;    % IMMEDIATE
0004   2103 4000                LD REG,ABC;
0008
0008   6103 4000                LD R3,#4000^;   % #4000^ USED AS "LABEL"
000C   6103 4000                LD REG,ABC^;
0010   6103 4000                LD REG,NLAB;
0014
0014                    %       LABELS:
0014
0014   5E08 0038                JP LABEL;       % SAME AS M80
0018   4103 0038                ADD R3,LABEL;
001C   4103 0038                ADD REG,SLAB;
0020
0020                    %       LABELS WITH OFFSET:
0020
0020   5E08 003C                JP LABEL(4);
0024   4103 003C                ADD R3,LABEL(4);
```

```
EXAMPLES...                    MACRO 8/8000 Z8000 ASSEMBLER  1.2.1E   PAGE 2


0028   4103 003C                ADD REG,SLAB(4);
002C
002C                    %       ADDRESS CONSTANTS:
002C
002C   7603 0038                LD R3,^LABEL;   % "^LABEL" MEANS "ADDRESS OF" OR "POINTER TO" LABEL
0030                                            % THIS INSTRUCTION IS THE SAME AS ZILOG'S LDA
0030
0030                    %       INDIRECT:
0030
0030   0143                     ADD R3,R4^;     % R4 CONTAINS POINTER TO OPERAND
0032   0143                     ADD REG,RIR;
0034
0034                    %       INDEXED:
0034
0034   4143 0038                ADD R3,LABEL(R4); % OPERAND IS LABEL OFFSET BY R4
0038
0038
0038                    %       LABEL DEFINITION:
0038
0038                    LABEL:
0038
0038
0038                    %       STATEMENTS TERMINATE BY SEMICOLON UNLESS FOLLOWED BY
0038                    %       ELSE OR END -
0038
0038   0B43                     IF REG EQ RIR THEN
003A   EE03                             LD REG,DEF
003C   2103 000F                ELSE
0040   E805                             BEGIN
0042   0103 000F                        ADD REG,DEF;
0046                                    LD LABEL,ABC    % ; OPTIONAL
0046   4D05 0038 4000                   END;
004C
004C
004C                    END.

NEITHER WARNING NOR ERROR MESSAGES

****(EXECUTIVE)         NORMAL TERMINATION
```

Specifications:

Operating Environment

Required Hardware

AMC Microcomputer Development System:
AmSYS 8/8 or AmSYS 29

64K bytes of RAM memory

Dual Floppy diskette drives, single or double density

System console: CRT or interactive hardcopy device

Required Software

64K version of AMDOS Operating System

Shipping Media

Floppy diskette, single or double density

Reference Manual

MACRO8000 Macro Assembler Manual #00680119

7

Ordering Information*

AmSYS Software can only be purchased as part of or as an addition to an AmSYS Microcomputer Development System.

Part Number	Description
AmSYS MACRO Z8000	AmZ8000 16-bit macroassembler with relocatable output and Linker LINK8000 to produce complete relocatable modules or absolute programs.

*MACRO8000 is included as a standard part of the AmSYS 8/8 Microcomputer Development System.

AmZ8000 Microprocessor Family

APPENDICES

Advanced Micro Devices Commitment to Excellence

Product Assurance Programs for Military and Commercial Integrated Circuits

A

A COMMITMENT TO EXCELLENCE

Advanced Micro Devices was conceived on the premise that there was a place in the semiconductor community for a manufacturer dedicated to excellence.

In product assurance procedures, Advanced Micro Devices is unique. Only Advanced Micro Devices processes all integrated circuits, commercial as well as military, to the demanding requirements of MIL-STD-883. The Rome Air Development Center (RADC), which is the Air Force's principal authority on component reliability, has issued MIL-HDBK-217B which indicates that parts processed to Military Standard 883, Level C (Advanced Micro Devices' standard processing) yield a product nearly ten times better in failure rates than the industry commercial average.

Our Sunnyvale facility has been certified by the Defense Electronics Supply Center (DESC) to produce parts to JAN Class B and C under Military Specification MIL-M-38510. The National Aeronautics and Space Administration (NASA) has certified this production line for the manufacture of Class A products for programs requiring the highest levels of reliability. Advanced Micro Devices is the only integrated circuit company formed within the last ten years to achieve such line certification.

This brochure outlines Advanced Micro Devices' standard programs for Class B, C and A devices for military and commercial operating range applications. These will cover the majority of system requirements today. Alternative screening flows for specific user needs can be performed on request. Check with your local sales office for further information.

ADVANCED MICRO DEVICES' STANDARD PRODUCTS ARE MANUFACTURED TO MIL-STD-883 REQUIREMENTS

Advanced Micro Devices' product assurance programs are based on two key documents.

MIL-M-38510 – General Specification for Microcircuits

MIL-STD-883 – Test Methods and Procedures for Microelectronics

The screening charts in this brochure show that every integrated circuit shipped by Advanced Micro Devices receives the critical screening procedures defined in MIL-STD-883, Method 5004 for Class C product. This includes molded plastic devices.

In addition, documentation, design, processing and assembly workmanship guidelines are patterned after MIL-M-38510 specifications.

Commercial and industrial users receive the quality and reliability benefits of this aerospace-type screening and documentation at no additional cost.

STANDARD PRODUCT TESTING CATEGORIES

Advanced Micro Devices offers integrated circuits to four standard testing categories.

1. Commercial operating range product (typically 0°C to 70°C)
2. Commercial product with 100% temperature testing
3. Military operating range product (typically −55°C to +125°C)
4. JAN qualified product

Categories 1, 2 and 3 are available on most Advanced Micro Devices circuits. Category 4 is offered on a more limited line. Check with your local sales office for details.

STANDARD PRODUCT ASSURANCE CATEGORIES

Devices produced to the above testing categories are available to the three standard classes of product assurance defined by MIL-STD-883. As a minimum, every device shipped by Advanced Micro Devices meets the screening requirements of Class C.

Class C – For commercial and ground-based military systems where replacement can be accomplished without difficulty.

According to MIL-HDBK-217B, this assures relative failure rates 9.4 times better than that of regular industry commercial product.

Class B – For flight applications and commercial systems where maintenance is difficult or expensive and where reliablity is vital.

Devices are upgraded from Class C to Class B by burn-in screening and additional testing.

According to MIL-HDBK-217B, Class B failure rate is improved 30 times over regular industry commercial product. Advanced Micro Devices Class B processing conforms to MIL-STD-883 requirements. MIL-HDBK-217B indicates that this may provide failure rates as much as two times better than some other manufacturers' "equivalent" or "pseudo" Class B programs.

Class S – For space applications where replacement is extremely difficult or impossible and reliability is imperative.

Class S screening includes x-ray and other special inspections tailored to the specific requirements of the user.

The 100% screening and quality conformance testing performed within these Advanced Micro Devices programs is shown in TABLES I, II and III. A full description of the process flow is provided in Product Assurance Document 15-010, available on request.

CLASS C SCREENING FLOW FOR COMMERCIAL SYSTEMS AND GROUND BASED MILITARY SYSTEMS

TABLE I
CLASS C
INTEGRATED CIRCUITS

Screening Procedure per MIL-STD-883 Method 5004, Class C		COMMERCIAL OPERATING RANGE		MILITARY OPERATING RANGE	
		HERMETIC AND MOLDED PACKAGES		HERMETIC PACKAGE ONLY	
Screen	Test Method	C1 Commercial Product	C2 Commercial Product With 100% Temperature Testing	C3 Military Product	C4 Jan Qualified Product
VISUAL AND MECHANICAL					
Internal visual	2010, Condition B	100%	100%	100%	100%
High temperature storage	1008, Condition C, 24 hours	100%	100%	100%	100%
Temperature cycle	1010, Condition C	100%	100%	100%	100%
Constant acceleration	2001	100% (1)	100% (1)	100%	100%
Hermeticity, Fine and Gross	1014	100% (1)	100% (1)	100%	100%
FINAL ELECTRICAL TESTS		AMD Data Sheet	AMD Data Sheet	AMD Data Sheet	38510 Slash Sheet
Static (dc)	a) At 25°C, and power supply extremes	100%	100%	100%	100%
	b) At temperature and power supply extremes	(2)	100% (3)	–	–
Functional	a) At 25°C, and power supply extremes	100%	100%	100%	100%
	b) At temperature and power supply extremes	(2)	100% (3)	–	–
Switching (ac) or Dynamic	At 25°C, nominal power supply	(2)	(2)	–	–
QUALITY CONFORMANCE	5005, Group A (See Table II)	Sample	Sample	Sample	Sample
Sample Tests	Group B	–	–	–	Sample
	Group C	–	–	–	Sample
	Group D	–	–	–	Sample
EXTERNAL VISUAL	2009 (Note 5)	100%	100%	100%	100%

TABLE II

GROUP A QUALITY CONFORMANCE LEVELS

Advanced Micro Devices employs the military-recommended LTPD sampling system to assure quality. MIL-STD-883, Method 5005, TABLE I, Group A, subgroups 1 through 9 as appropriate to the device family are performed on every lot. Quality levels defined for Class B product are applied by Advanced Micro Devices to both Class B and Class C orders.

	LTPD	INITIAL SAMPLE SIZE
Subgroup 1 – Static tests at 25°C	5	45
Subgroup 2 – Static tests at maximum rated operating temperature	7	32
Subgroup 3 – Static tests at minimum rated operating temperature	7	32
Subgroup 4 – Dynamic tests at 25°C – LINEAR devices	5	45
Subgroup 5 – Dynamic tests at maximum rated operating temperature – LINEAR devices	7	32
Subgroup 6 – Dynamic tests at minimum rated operating temperature – LINEAR devices	7	32
Subgroup 7 – Functional tests at 25°C	5	45
Subgroup 8 – Functional tests at maximum and minimum rated operating temperatures	10	22
Subgroup 9 – Switching tests at 25°C – DIGITAL devices	7	32
Subgroup 10 – Switching tests at maximum rated operating temperatures – DIGITAL devices	*	
Subgroup 11 – Switching tests at minimum rated operating temperatures – DIGITAL devices	*	

*These subgroups, where applicable, are usually performed during initial characterization only for all except JAN Qualified product.

CLASS B SCREENING FLOW FOR HIGH RELIABILITY COMMERCIAL AND MILITARY SYSTEMS

TABLE III
CLASS B
INTEGRATED CIRCUITS
(Class C plus burn in screening and additional testing.)

Screening Procedure per MIL-STD-883 Method 5004, Class B		COMMERCIAL OPERATING RANGE		MILITARY OPERATING RANGE	
		HERMETIC AND MOLDED PACKAGES		HERMETIC PACKAGE ONLY	
		B1	B2	B3	B4
Screen	Test Method	Commercial Product	Commercial Product With 100% Temperature Testing	Military Product	Jan Qualified Product
VISUAL AND MECHANICAL					
Internal visual	2010, Condition B	100%	100%	100%	100%
High temperature storage	1008, Condition C, 24 hours	100%	100%	100%	100%
Temperature cycle	1010, Condition C	100%	100%	100%	100%
Constant acceleration	2001	100% (1)	100% (1)	100%	100%
Hermeticity, Fine and Gross	1014	100% (1)	100% (1)	100%	100%
BURN IN					
Interim (pre burn in) electricals	Per applicable device specification	100%	100%	100%	100%
Burn in	1015, 160 hours at 125°C or equivalent.*	100%	100%	100%	100%
FINAL ELECTRICAL TESTS		AMD Data Sheet	AMD Data Sheet	AMD Data Sheet	38510 Slash Sheet
Static (dc)	a) At 25°C, and power supply extremes	100%	100%	100%	100%
	b) At temperature and power supply extremes	(2)	100% (3)	100%	100%
Functional	a) At 25°C, and power supply extremes	100%	100%	100%	100%
	b) At temperature and power supply extremes	(2)	100% (3)	100%	100%
Switching (ac) or Dynamic	At 25°C, nominal power supply	(2)	(2)	100%	100%
QUALITY CONFORMANCE	5005, Group A (See Table II)	Sample	Sample	Sample	Sample
Sample Tests	Group B	–	–	(4)	Sample
	Group C	–	–	(4)	Sample
	Group D	–	–	(4)	Sample
EXTERNAL VISUAL	2009 (Note 5)	100%	100%	100%	100%

Notes: 1. Not applicable to molded packages.
2. All MOS RAMs and many other MOS devices receive a.c. testing and 100% d.c. screening at high temperature and power supply extremes as standard. Other products sampled at Group A (Table III).
3. Tested at high temperature, 100°C, only on commercial range product. Note that this is a full d.c. check of all parameters in addition to the simple "hot-rail" functional sequence performed on most other commercial programs.
4. Available to special order.
5. Without optical aid for commercial devices. *Unless otherwise specified on the device data sheet.

CLASS S
FOR AEROSPACE SYSTEMS. (FORMERLY CLASS A)

Advanced Micro Devices offers a Class S program.

This program together with other high reliability screening options, such as SEM and x-ray, is described as Option A in Advanced Micro Devices' Extended Processing Options Document 00-003. Contact your local Advanced Micro Devices' sales office for more information.

A

STANDARD PRODUCT
SCREENING SUMMARY AND ORDERING INFORMATION

1. COMMERCIAL PRODUCT

- Screened per MIL-STD-883, Method 5004.
- Electrically tested per AMD Data Sheet.
- Supplied in hermetic and molded packages.
- Quality conformance testing, Method 5005, Group A, performed to levels specified for Class B on both Class C and Class B options.

Class C (Flow C1)

- Order standard AMD part number.
- Marked same as order number.

  Example: Am2901ADC

Class B (Flow B1)

- Burn in performed in AMD circuit condition.
- Order standard AMD part number, add suffix B (or /883B for 1, 2 and 300 Series Linear devices).
- Marked same as order number.

  Example: Am2901ADC-B

2. COMMERCIAL PRODUCT WITH 100% TEMPERATURE TESTING

- Identical to standard commercial operating range product with the addition of 100% dc and functional testing at 100°C and power supply extremes.

Class C (Flow C2)

- Order standard AMD part number, add suffix T.
- Marked same as order number.

  Example: Am2901ADC-T

Class B (Flow B2)

- Burn in performed in AMD circuit condition.
- Order standard AMD part number, add suffix TB.
- Marked same as order number.

  Example: Am2901ADC-TB

3. MILITARY PRODUCT

- Screened per MIL-STD-883, Method 5004.
- Electrically tested per AMD Data Sheet.
- Supplied in hermetic package only.
- Quality conformance testing, Method 5005, Group A, performed to levels specified for Class B on both Class B and Class C options.

Class C (Flow C3)

- Order standard AMD part number.
- Marked same as order number.

  Example: Am2901ADM

Class B (Flow B3)

- Burn in performed in AMD circuit condition.
- AC at 25°C, dc and functional testing at 25°C as well as temperature and power supply extremes performed on 100% of every lot.
- Quality conformance testing, Method 5005, Groups B, C and D available to special order.
- Order standard AMD part number, add suffix B.
- Marked same as order number.

  Example: Am2901ADM-B

4. JAN QUALIFIED PRODUCT

- Screened per MIL-STD-883, Method 5004.
- Electrically tested to JAN detail specification (slash sheet).
- Manufactured in Defense Logistics Agency certified facility.
- Quality conformance testing, Method 5005, Groups A, B, C and D performed as standard and must be completed prior to shipment.
- It is a product for which AMD has gained QPL listing.*

Class C (Flow C4)

- Order per military document.
- Marked per military document.

  Example: JM38510/44001CQB

Class B (Flow B4)

- Burn in performed in circuit condition approved for JAN devices.
- Order per military document.
- Marked per military document.

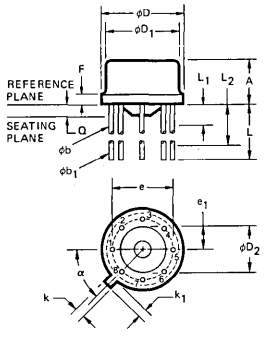
  Example: JM38510/44001BRC

*In certain cases where JAN Qualified product is specified but is not available, Advanced Micro Devices can provide devices to the electrical limits and burn-in criteria of the slash sheet. This class of product has been called JAN Equivalent and marked M38510/ by some manufacturers. This identification is no longer permitted by DESC. Check with your local sales office for availability of specific device types.

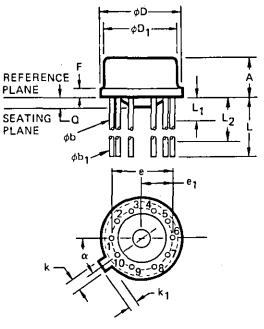
PACKAGE OUTLINES

METAL CAN PACKAGES

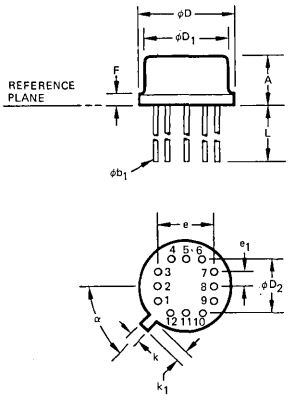
H-8-1



H-10-1



G-12-1



AMD Pkg.	H-8-1		H-10-1		G-12-1	
Common Name	TO-99 Metal Can		TO-100 Metal Can		TO-8 Metal Can	
38510 Appendix C	A-1		A-2		–	
Parameters	Min.	Max.	Min.	Max.	Min.	Max.
A	.165	.185	.165	.185	.155	.180
e	.185	.215	.215	.245	.390	.410
e_1	.090	.110	.105	.125	.090	.110
F	.013	.033	.013	.033	.020	.030
k	.027	.034	.027	.034	.024	.034
k_1	.027	.045	.027	.045	.024	.038
L	.500	.570	.500	.610	.500	.600
L_1		.050		.050		
L_2	.250		.250			
α	45° BSC		36° BSC		45°	
ϕb	.016	.019	.016	.019		
ϕb_1	.016	.021	.016	.021	.016	.021
ϕD	.350	.370	.350	.370	.590	.610
ϕD_1	.305	.335	.305	.335	.540	.560
ϕD_2	.120	.160	.120	.160	.390	.410
Q	.015	.045	.015	.045		

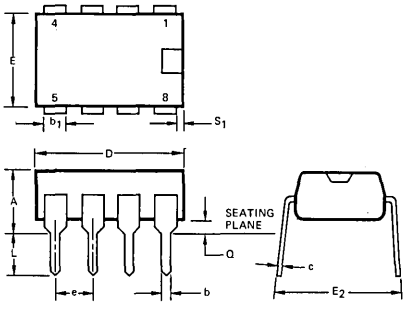
Notes: 1. Standard lead finish is bright acid tin plate or gold plate.
2. ϕb applies between L_1 and L_2. ϕb_1 applies between L_1 and 0.500" beyond reference plane.

B

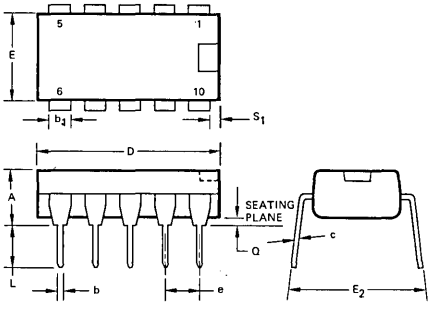
PACKAGE OUTLINES (Cont.)

MOLDED DUAL IN-LINE PACKAGES

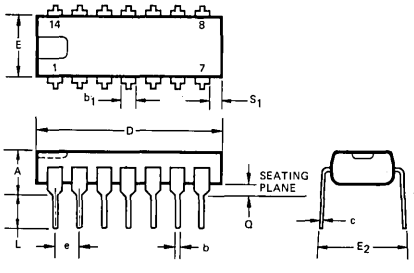
P-8-1



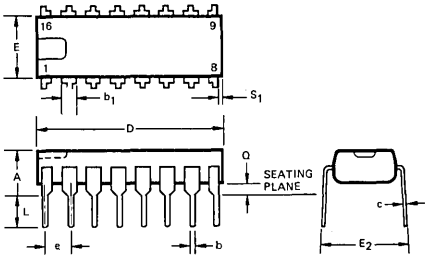
P-10-1



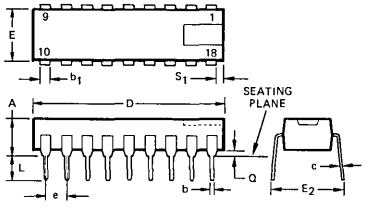
P-14-1



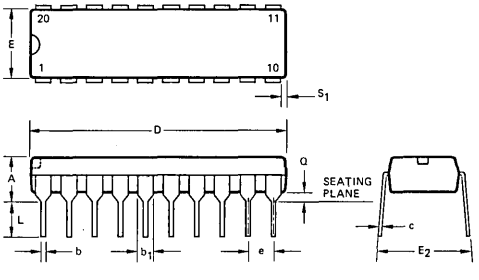
P-16-1



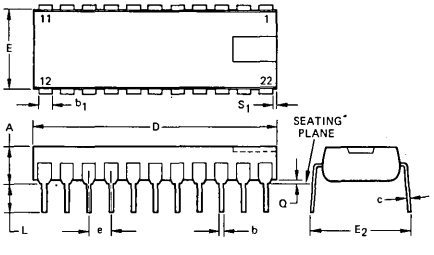
P-18-1



P-20-1



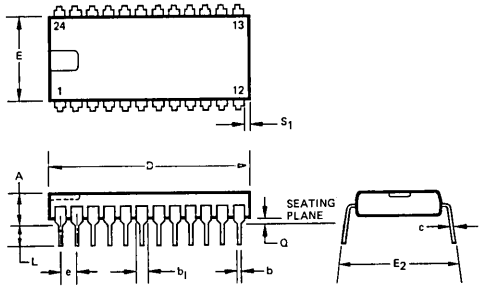
P-22-1



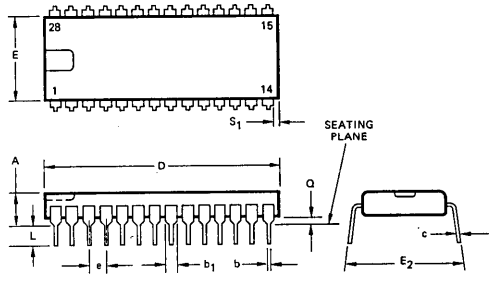
PACKAGE OUTLINES (Cont.)

MOLDED DUAL IN-LINE PACKAGES (Cont.)

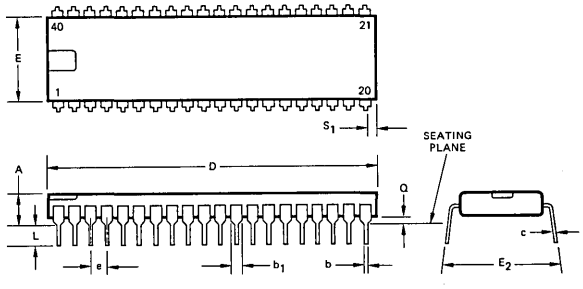
P-24-1



P-28-1



P-40-1



AMD Pkg.	P-8-1		P-10-1		P-14-1		P-16-1		P-18-1		P-20-1		P-22-1		P-24-1		P-28-1		P-40-1	
Parameters	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
A	.150	.200	.150	.200	.150	.200	.150	.200	.150	.200	.150	.200	.150	.200	.170	.215	.150	.200	.150	.200
b	.015	.022	.015	.020	.015	.020	.015	.020	.015	.020	.015	.020	.015	.020	.015	.020	.015	.020	.015	.020
b_1	.055	.065	.055	.065	.055	.065	.055	.065	.055	.065	.055	.065	.055	.065	.055	.065	.055	.065	.055	.065
c	.009	.011	.009	.011	.009	.011	.009	.011	.009	.011	.009	.011	.009	.011	.009	.011	.009	.011	.009	.011
D	.375	.395	.505	.550	.745	.775	.745	.775	.895	.925	1.010	1.050	1.080	1.120	1.240	1.270	1.450	1.480	2.050	2.080
E	.240	.260	.240	.260	.240	.260	.240	.260	.240	.260	.250	.290	.330	.370	.515	.540	.530	.550	.530	.550
E_2	.310	.385	.310	.385	.310	.385	.310	.385	.310	.385	.310	.385	.410	.480	.585	.700	.585	.700	.585	.700
e	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110
L	.125	.150	.125	.150	.125	.150	.125	.150	.125	.150	.125	.150	.125	.160	.125	.160	.125	.160	.125	.160
Q	.015	.060	.015	.060	.015	.060	.015	.060	.015	.060	.015	.060	.015	.060	.015	.060	.015	.060	.015	.060
S_1	.010	.030	.040	.070	.040	.065	.010	.040	.030	.040	.025	.055	.015	.045	.035	.065	.040	.070	.040	.070

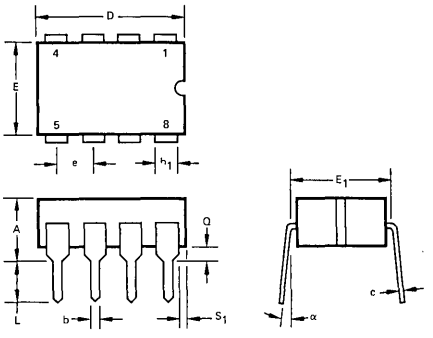
Notes: 1. Standard lead finish is tin plate or solder dip.
2. Dimension E_2 is an outside measurement.

B

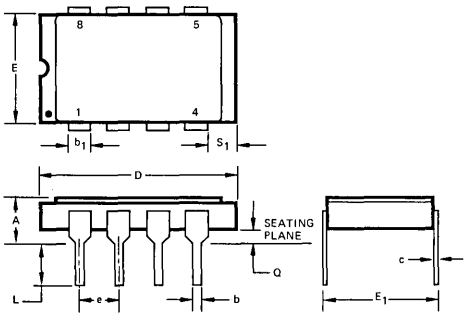
PACKAGE OUTLINES (Cont.)

HERMETIC DUAL IN-LINE PACKAGES

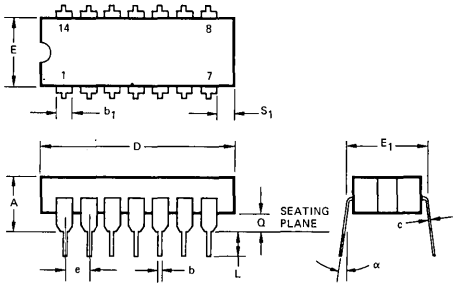
D-8-1



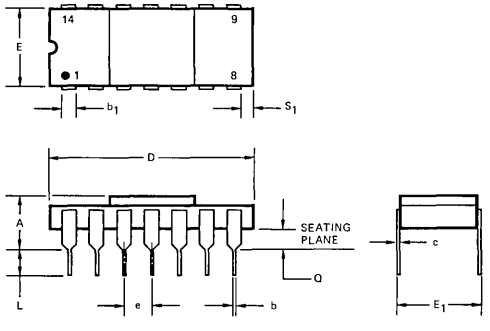
D-8-2



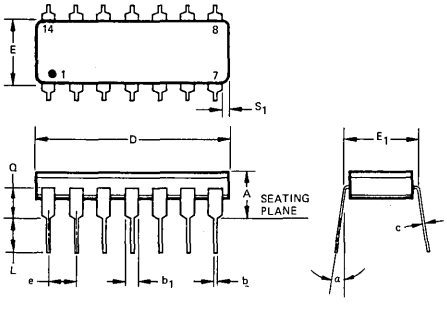
D-14-1



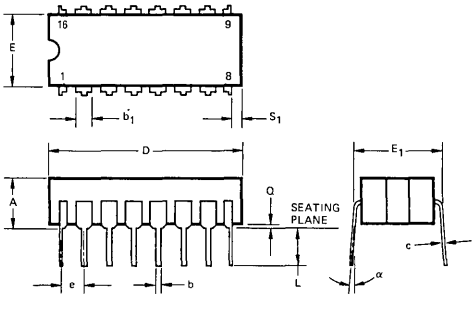
D-14-2



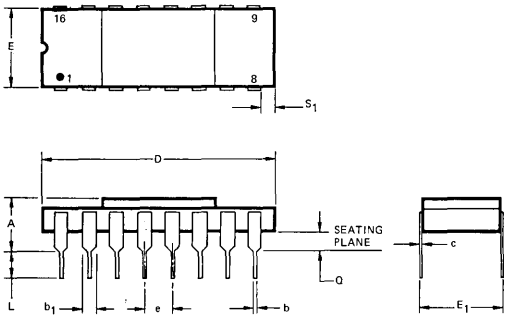
D-14-3



D-16-1



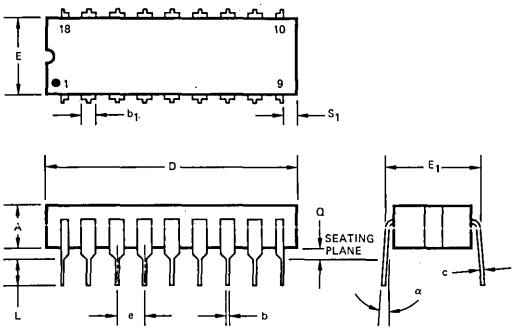
D-16-2



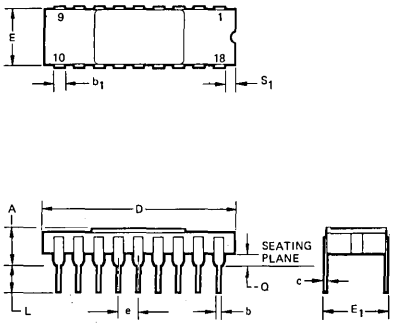
PACKAGE OUTLINES (Cont.)

HERMETIC DUAL IN-LINE PACKAGES (Cont.)

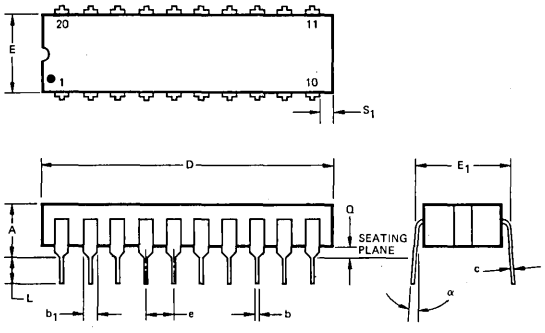
D-18-1



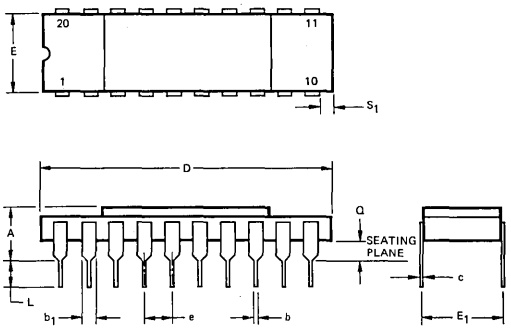
D-18-2



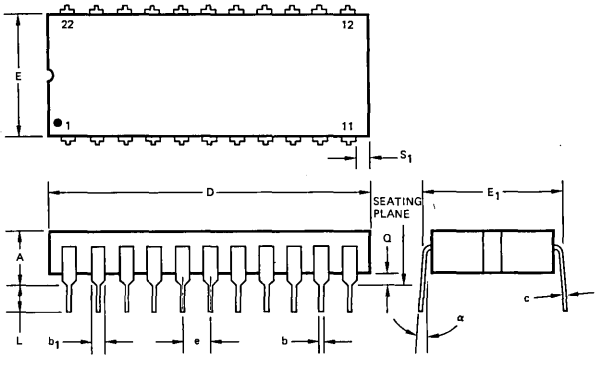
D-20-1



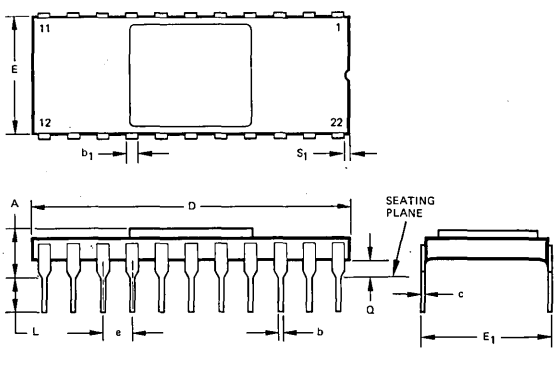
D-20-2



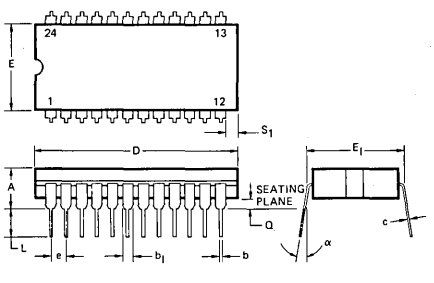
D-22-1



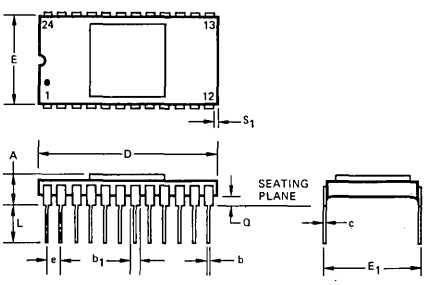
D-22-2



D-24-1 and D-24-4



D-24-2

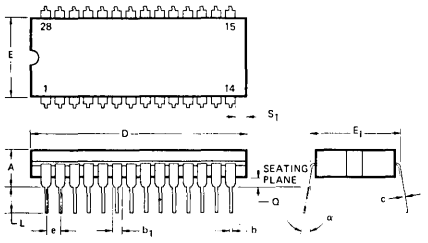


B

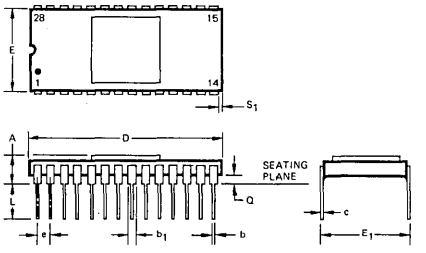
PACKAGE OUTLINES (Cont.)

HERMETIC DUAL IN-LINE PACKAGES (Cont.)

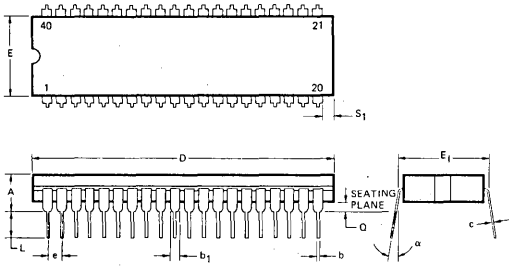
D-28-1



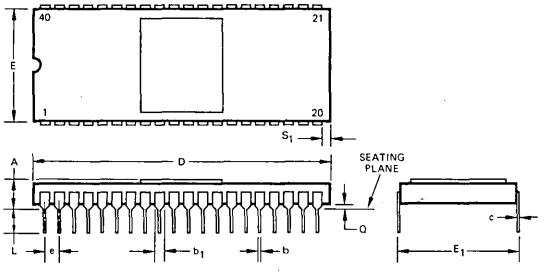
D-28-2



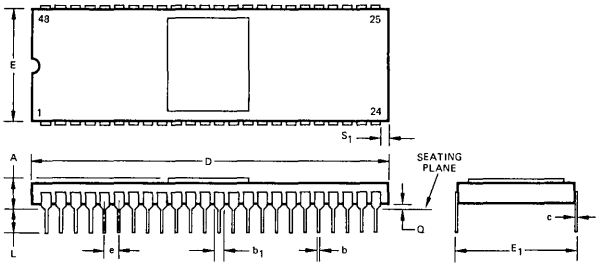
D-40-1



D-40-2



D-48-2



PACKAGE OUTLINES (Cont.)

HERMETIC DUAL IN-LINE PACKAGES (Cont.)

AMD Pkg.	D-8-1		D-8-2		D-14-1		D-14-2		D-14-3 (Note 2)		D-16-1		D-16-2	
Common Name	CERDIP		SIDE-BRAZED		CERDIP		SIDE-BRAZED		METAL DIP		CERDIP		SIDE-BRAZED	
38510 Appendix C	–		–		D-1(1)		D-1(3)		D-1(1)		D-2(1)		D-2(3)	
Parameters	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
A	.130	.200	.100	.200	.130	.200	.100	.200	.100	.200	.130	.200	.100	.200
b	.016	.020	.015	.022	.016	.020	.015	.022	.015	.023	.016	.020	.015	.022
b_1	.050	.070	.040	.065	.050	.070	.040	.065	.030	.070	.050	.070	.040	.065
c	.009	.011	.008	.013	.009	.011	.008	.013	.008	.011	.009	.011	.008	.013
D	.370	.400	.500	.540	.745	.785	.690	.730	.660	.785	.745	.785	.780	.820
E	.240	.285	.260	.310	.240	.285	.260	.310	.230	.265	.240	.310	.260	.310
E_1	.300	.320	.290	.320	.290	.320	.290	.320	.290	.310	.290	.320	.290	.320
e	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110
L	.125	.150	.125	.160	.125	.150	.125	.160	.100	.150	.125	.150	.125	.160
Q	.015	.060	.020	.060	.015	.060	.020	.060	.020	.080	.015	.060	.020	.060
S_1	.004		.005		.010		.005		.020		.005		.005	
α	3°	13°			3°	13°			3°	13°	3°	13°		
Standard Lead Finish	b		b or c		b		b or c		c		b		b or c	

AMD Pkg.	D-18-1		D-18-2		D-20-1		D-20-2		D-22-1		D-22-2		D-24-1	
Common Name	CERDIP		SIDE-BRAZED		CERDIP		SIDE-BRAZED		CERDIP		SIDE-BRAZED		CERDIP	
38510 Appendix C	–		–		–		–		–		–		D-3(1)	
Parameters	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
A	.130	.200	.100	.200	.140	.220	.100	.200	.140	.220	.100	.200	.150	.225
b	.016	.020	.015	.022	.016	.020	.015	.022	.016	.020	.015	.022	.016	.020
b_1	.050	.070	.040	.065	.050	.070	.040	.065	.045	.065	.030	.060	.045	.065
c	.009	.011	.008	.013	.009	.011	.008	.013	.009	.011	.008	.013	.009	.011
D	.870	.920	.850	.930	.935	.970	.950	1.010	1.045	1.110	1.050	1.110	1.230	1.285
E	.280	.310	.260	.310	.245	.285	.260	.310	.360	.405	.360	.410	.510	.545
E_1	.290	.320	.290	.320	.290	.320	.290	.320	.390	.420	.390	.420	.600	.620
e	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110
L	.125	.150	.125	.160	.125	.150	.125	.160	.125	.150	.125	.160	.120	.150
Q	.015	.060	.020	.060	.015	.060	.020	.060	.015	.060	.020	.060	.015	.060
S_1	.005		.005		.005		.005		.005		.005		.010	
α	3°	13°			3°	13°			3°	13°			3°	13°
Standard Lead Finish	b		b or c		b		b or c		b		b or c		b	

PACKAGE OUTLINES (Cont.)

HERMETIC DUAL IN-LINE PACKAGES (Cont.)

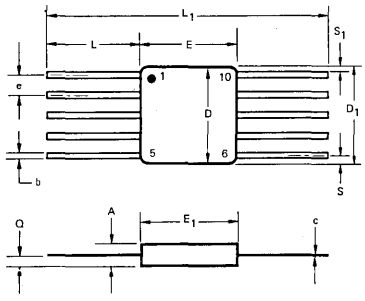
AMD Pkg.	D-24-2		D-24-4		D-28-1		D-28-2		D-40-1		D-40-2		D-48-2	
Common Name	SIDE-BRAZED		CERVIEW		CERDIP		SIDE-BRAZED		CERDIP		SIDE-BRAZED		SIDE-BRAZED	
38510 Appendix C	D-3(3)		–		–		–		D-5		–		–	
Parameters	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
A	.100	.200	.150	.225	.150	.225	.100	.200	.150	.225	.100	.200	.100	.200
b	.015	.022	.016	.020	.016	.020	.015	.022	.016	.020	.015	.022	.015	.022
b_1	.030	.060	.045	.065	.045	.065	.030	.060	.045	.065	.030	.060	.030	.060
c	.008	.013	.009	.011	.009	.011	.008	.013	.009	.011	.008	.013	.008	.013
D	1.170	1.200	1.235	1.280	1.440	1.500	1.380	1.420	2.020	2.100	1.960	2.040	2.370	2.430
E	.550	.610	.510	.550	.510	.550	.560	.600	.510	.550	.550	.610	.570	.610
E_1	.590	.620	.600	.630	.600	.630	.590	.620	.600	.630	.590	.620	.590	.620
e	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110	.090	.110
L	.120	.160	.120	.150	.120	.150	.120	.160	.120	.150	.120	.160	.125	.160
Q	.020	.060	.015	.060	.015	.060	.020	.060	.015	.060	.020	.060	.020	.060
S_1	.005		.010		.005		.005		.005		.005		.005	
α			3°	13°	3°	13°			3°	13°				
Standard Lead Finish	b or c				b		b		b		b or c		b or c	

Notes: 1. Load finish b is tin plate. Finish c is gold plate.
2. Used only for LM108/LM108A.
3. Dimensions E and D allow for off-center lid, meniscus and glass overrun.

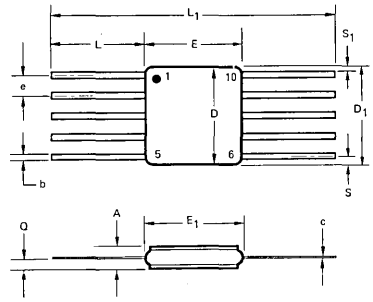
PACKAGE OUTLINES (Cont.)

FLAT PACKAGES

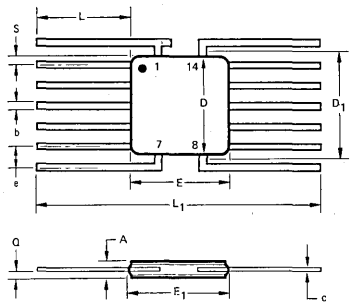
F-10-1



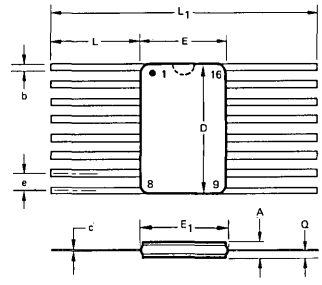
F-10-2



F-14-1 and F-14-2

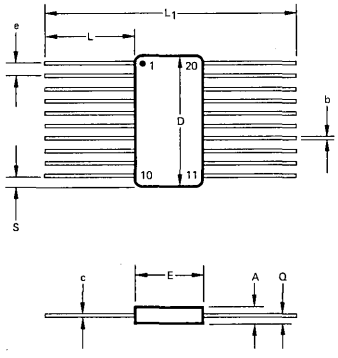


F-16-1 and F-16-2

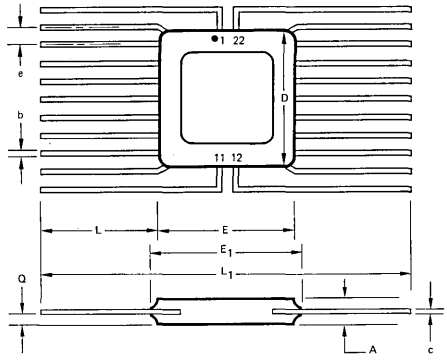


Note: Notch is pin 1 index on cerpack.

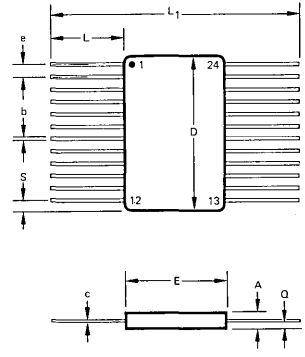
F-20-1



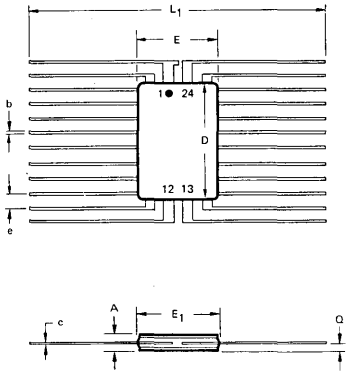
F-22-1



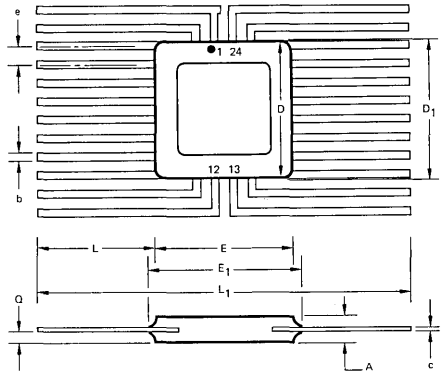
F-24-1



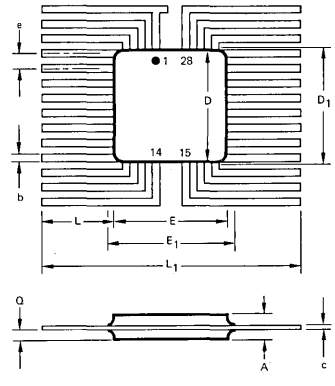
F-24-2



F-24-3



F-28-1

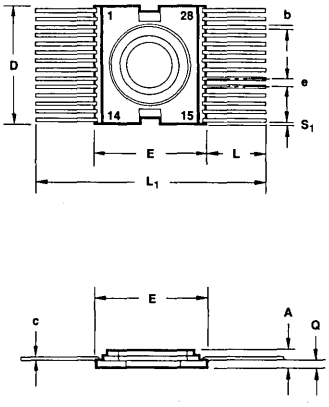


B

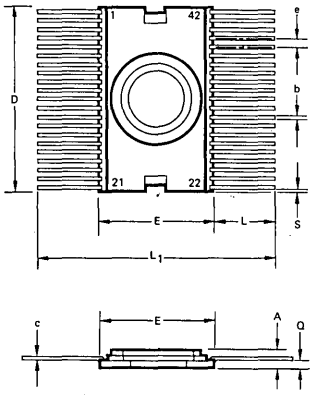
PACKAGE OUTLINES (Cont.)

FLAT PACKAGES (Cont.)

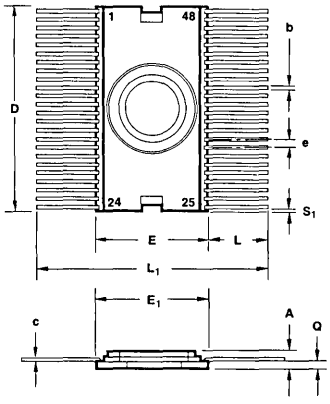
F-28-2



F-42-1



F-48-2



AMD Pkg.	F-10-1		F-10-2		F-14-1		F-14-2		F-16-1		F-16-2		F-20-1		F-22-1	
Common NAME	CERPACK		METAL FLAT PAK		CERPACK		METAL FLAT PAK		CERPACK		METAL FLAT PAK		CERPACK		METAL FLAT PAK	
38510 Appendix C	F-4		F-4		.F-1		F-1		F-5		–		–		–	
Parameters	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
A	.045	.080	.045	.080	.045	.080	.045	.085	.045	.085	.045	.085	.045	.085	.045	.090
b	.015	.019	.012	.019	.015	.019	.012	.019	.015	.019	.015	.019	.015	.019	.015	.019
c	.004	.006	.003	.006	.004	.006	.003	.006	.004	.006	.003	.006	.004	.006	.003	.006
D	.230	.255	.235	.275	.230	.255	.230	.270	.370	.425	.370	.400	.490	.520	.380	.420
D_1				.275				.280				.410				.440
E	.240	.260	.240	.260	.240	.260	.240	.260	.245	.285	.245	.285	.245	.285	.380	.420
E_1		.275		.280		.275		.280		.290		.305		.290		.440
e	.045	.055	.045	.055	.045	.055	.045	.055	.045	.055	.045	.055	.045	.055	.045	.055
L	.300	.370	.300	.370	.300	.370	.300	.370	.300	.370	.300	.370	.300	.370	.250	.320
L_1	.920	.980	.920	.980	.920	.980	.920	.980	.920	.980	.920	.980	.920	.980	.920	.980
Q	.010	.040	.010	.040	.010	.040	.010	.040	.020	.040	.010	.040	.020	.040	.010	.040
S_1	.005		.005		.005		.005		.005		.005		.005			
Standard Lead Finish	b		c		b		c		b		c		b		c	

AMD Pkg.	F-24-1		F-24-2		F-24-3		F-28-1		F-28-2		F-42-1		F-48-2	
Common Name	CERPACK		METAL FLAT PAK		METAL FLAT PAK		METAL FLAT PAK		CERAMIC FLAT PAK		CERAMIC FLAT PAK		CERAMIC FLAT PAK	
38510 Appendix C	F-6		F-8		–		–		–		–		–	
Parameters	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.
A	.050	.090	.045	.090	.045	.090	.045	.080	.065	.085	.070	.115	.070	.110
b	.015	.019	.015	.019	.015	.019	.015	.019	.016	.025	.017	.023	.018	.022
c	.004	.006	.003	.006	.003	.006	.003	.006	.007	.010	.006	.012	.006	.010
D	.580	.620	.360	.410	.380	.420	.360	.410	.700	.720	1.030	1.090	1.175	1.250
D_1				.420		.440		.410		.720		1.090		1.250
E	.360	.385	.245	.285	.380	.420	.360	.410	.625	.650	.620	.660	.615	.670
E_1		.410		.305		.440		.410		.650		.660		.670
e	.045	.055	.045	.055	.045	.055	.045	.055	.045	.055	.045	.055	.045	.055
L	.265	.320	.300	.370	.250	.320	.270	.320	.415	.435	.320	.370	.320	.370
L_1	.920	.980	.920	.980	.920	.980	.955	1.000	1.475	1.500	1.300	1.370	1.310	1.365
Q	.020	.040	.010	.040	.010	.040	.010	.040	.017	.025	.020	.060	.020	.055
S_1	.005		.005		0		0		.005		.005		.015	
Standard Lead Finish	b		c		c		c		c		c		c	

Notes: 1. Lead finish b is tin plate. Finish c is gold plate.
2. Dimensions E_1 and D_1 allow for off-center lid, meniscus, and glass overrun.

PRODUCT ASSURANCE
MIL-M-38510 • MIL-STD-883

The product assurance program at Advanced Micro Devices defines manufacturing flow, establishes standards and controls, and confirms the product quality at critical points. Standardization under this program assures that all products meet military and government agency specifications for reliable ground applications. Further screening for users desiring flight hardware and other higher reliability classes is simplified because starting product meets all initial requirements for high-reliability parts.

The quality standards and screening methods of this program are equally valuable for commercial parts where equipment must perform reliably with minimum field service.

Two military documents provide the foundation for this program. They are:

MIL-M-38510 – General Specification for Microcircuits
MIL-STD-883 – Test Methods and Procedures for Microelectronics

MIL-M-38510 describes design, processing and assembly workmanship guidelines for military and space-grade integrated circuits. All circuits manufactured by Advanced Micro Devices for full temperature range (−55°C to +125°C) operation meet these quality requirements of MIL-M-38510.

MIL-STD-883 defines detail testing and inspection methods for integrated circuits. Three of the methods are quality and processing standards directly related to product assurance:

Test Method 2010 defines the visual inspection of integrated circuits before sealing. By confirming fabrication and assembly quality, inspection to this standard assures the user of reliable circuits in long-term field applications. Standard inspection at Advanced Micro Devices includes all the requirements of the latest revision of Method 2010, condition B.

Test Method 5004 defines three reliability classes of parts. All must receive certain basic inspection, preconditioning and screening stresses. The classes are:

Class C – Used where replacement can be readily accomplished. Screening steps are given in the AMD processing flow chart.

Class B – Used where maintenance is difficult or expensive and where reliability is vital. Devices are upgraded from Class C to Class B by 160-hour burn-in at 125°C followed by more extensive electrical measurements. All other screening requirements are the same.

Class S – Used where replacement is extremely difficult and reliability is imperative. Class S screening selects extra reliability parts by expanded visual and X-ray inspection, further burn-in, and tighter sampling inspection.

All hermetically sealed integrated circuits (military and commercial) manufactured by Advanced Micro Devices are screened to MIL-STD-883, Class C. Molded integrated circuits receive Class C screening except that centrifuge and hermeticity steps are omitted.

Optional extended processing to MIL-STD-883, Class B is available for all AMD integrated circuits. Parts procured to this screening are marked with a "−B" following the standard part number, except that linear 100, 200 or 300 series are suffixed "/883B".

Test Method 5005 defines qualification and quality conformance procedures. Subgroups, tests and quality levels are given for Group A (electrical), Group B (mechanical quality related to the user's assembly environment), Group C (die related tests) and Group D (package related tests). Group A tests are always performed; Group B, C and D may be specified by the user.

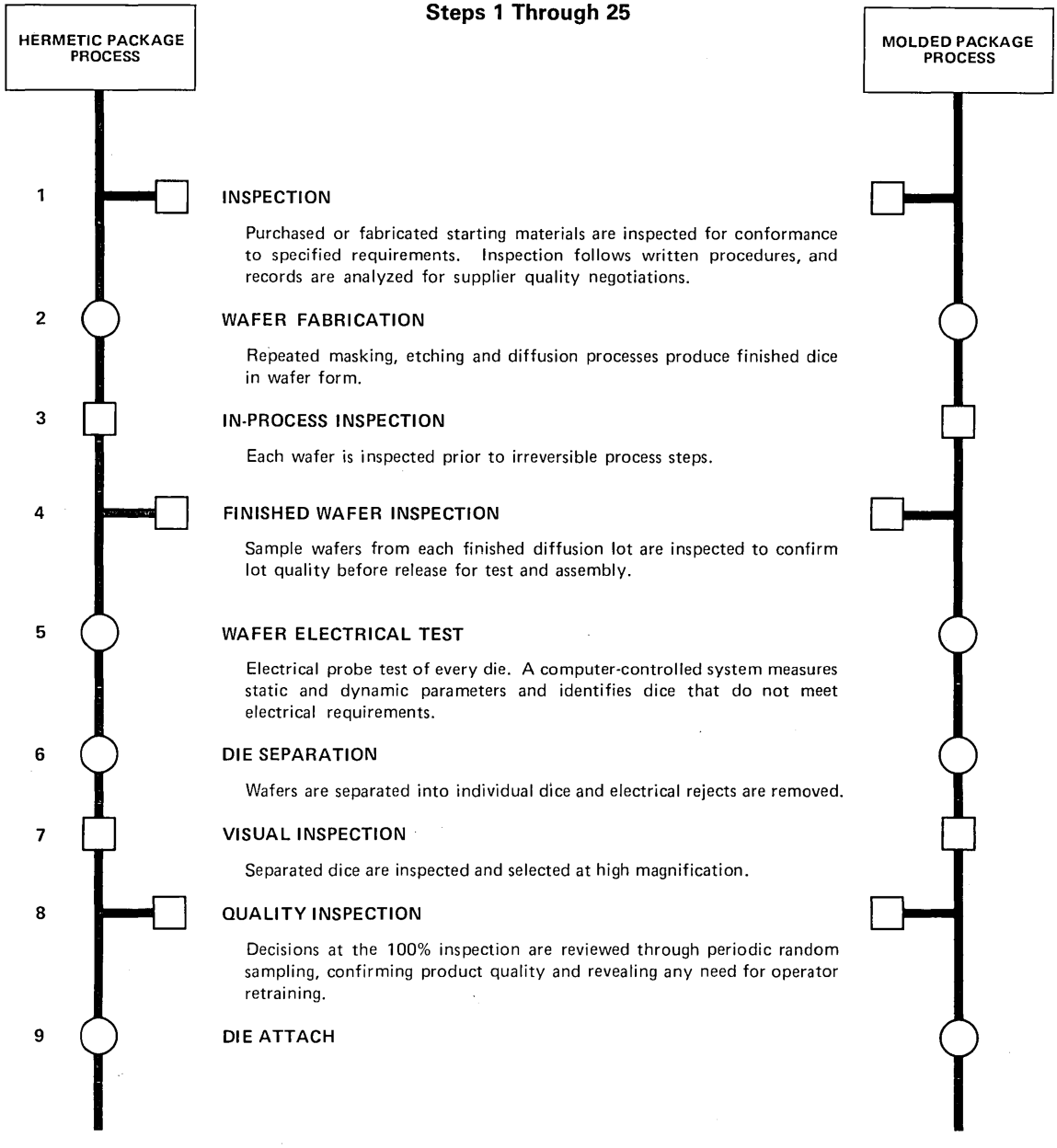
MANUFACTURING, SCREENING AND INSPECTION FOR INTEGRATED CIRCUITS

All integrated circuits are screened to MIL-STD-883, Method 5004, Class C; quality conformance inspection where required is performed to Class B quality levels on either Class B or Class C product.

All full-temperature-range (−55°C to +125°C) circuits are manufactured to the workmanship requirements of MIL-M-38510.

The flow chart identifies processing steps as they relate to MIL-STD-883 and MIL-M-38510.

STANDARD PROCESSING CLASS C Steps 1 Through 25



10 **QUALITY INSPECTION**

Strength of die attachment, position of die and visual quality of eutectic wetting are confirmed periodically by inspecting random samples and push-testing the attached dice.

11 **WIRE BOND**

Hermetic: Aluminum wires, ultrasonic bonding.
Molded: Gold wires, thermocompression bonding.

12 **QUALITY INSPECTION**

Weld strength, bond size and position, wire dress and general workmanship are confirmed periodically by comparing random samples with assembly instructions and quality standards. Bond strength is plotted on statistical control charts, providing early warning of process drifts.

13 **INTERNAL VISUAL INSPECTION**

Assembled but unsealed units are individually inspected at low and high power.

QUALITY STANDARDS:

All devices — MIL-STD-883, Method 2010, Condition B (latest revision). Full temperature devices — MIL-M-38510, Para. 3.7 for workmanship (re-bonding limits).

14 **QUALITY INSPECTION**

Decisions at the 100% inspection are reviewed through periodic random sampling, providing confirmation of product quality and revealing any need for operator retraining.

15a **FINAL SEAL**

(Hermetic devices)

15b **ENCAPSULATE**

(Molded Devices)

16 **HIGH TEMPERATURE STORAGE**

MIL-STD-883, Method 1008, Cond. C: 150°C, 24 hr

17 **TEMPERATURE CYCLE**

MIL-STD-883, Method 1010, Cond. C: −65°C, +150°C, 10 cycles

18 **CENTRIFUGE**

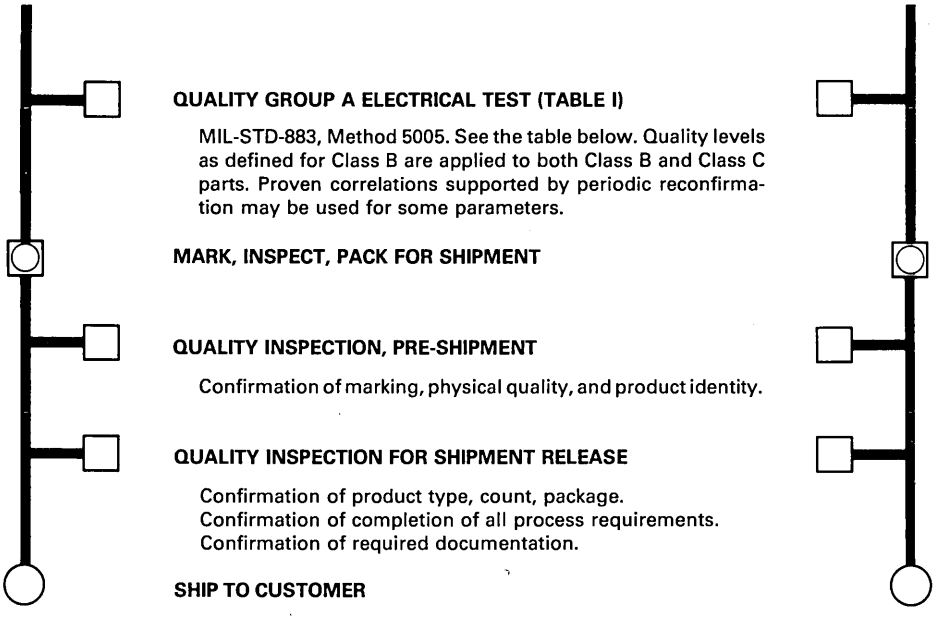
MIL-STD-883, Method 2001, Cond. E: 30,000 G

19 **SEAL (HERMETICITY) TEST**

MIL-STD-883, Method 1014, Cond. A or B: Fine Leak
MIL-STD-883, Method 1014, Cond. C2: Gross Leak

20 **ELECTRICAL TEST**

MIL-STD-883, Method 5004, Para. 3.1.12: Static, dynamic, functional tests at 25°C or in certain products at the most critical extreme temperature to assure accuracy of device selection.



21 **QUALITY GROUP A ELECTRICAL TEST (TABLE I)**

MIL-STD-883, Method 5005. See the table below. Quality levels as defined for Class B are applied to both Class B and Class C parts. Proven correlations supported by periodic reconfirmation may be used for some parameters.

22 **MARK, INSPECT, PACK FOR SHIPMENT**

23 **QUALITY INSPECTION, PRE-SHIPMENT**

Confirmation of marking, physical quality, and product identity.

24 **QUALITY INSPECTION FOR SHIPMENT RELEASE**

Confirmation of product type, count, package.
Confirmation of completion of all process requirements.
Confirmation of required documentation.

25 **SHIP TO CUSTOMER**

This AMD standard product meets screening requirements of MIL-STD-883, Class C.

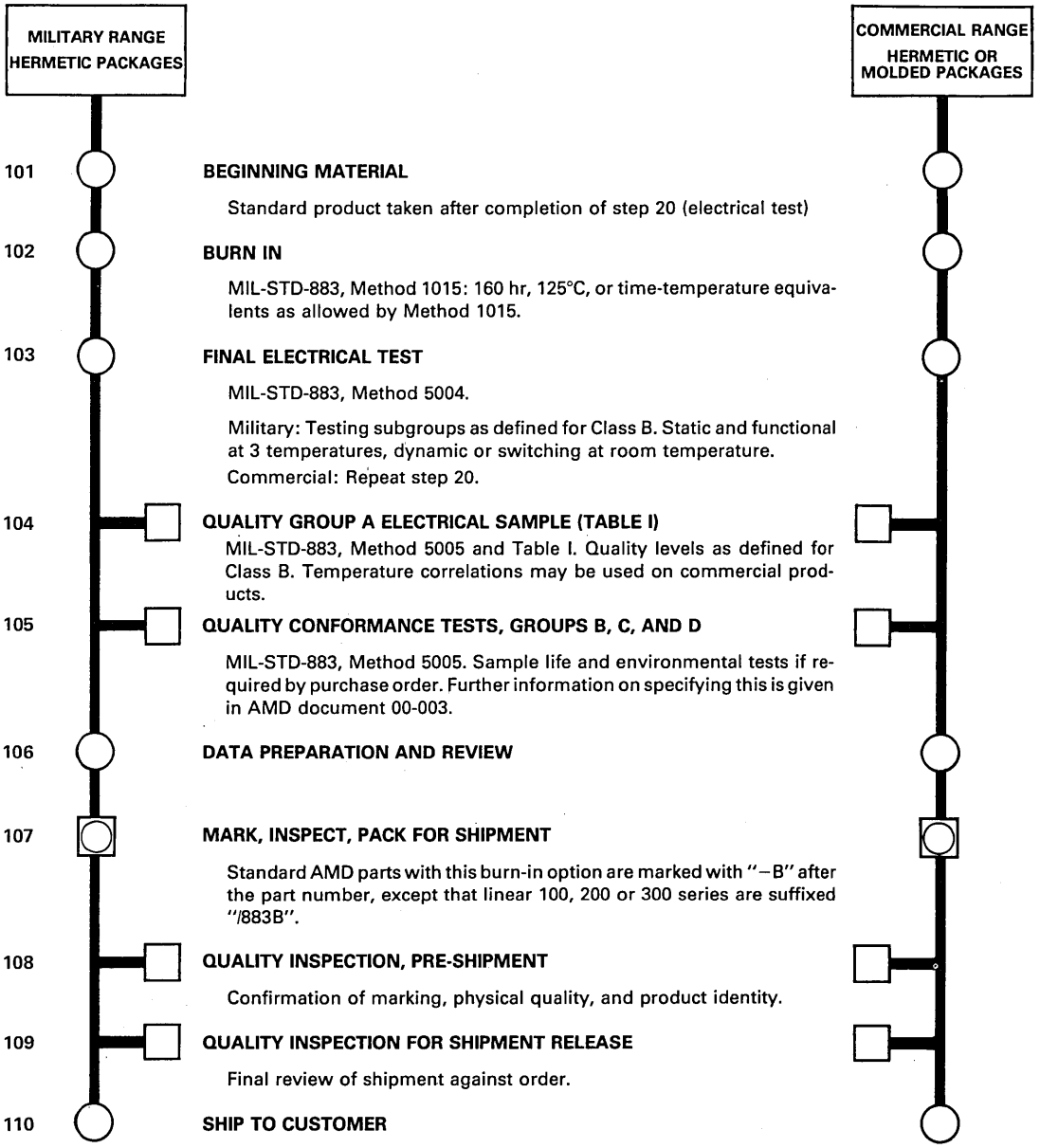
GROUP A ELECTRICAL TESTS
From MIL-STD-883, Method 5005, Table I

Subgroups	LTPD (Note 1)	Initial Sample Size
Subgroup 1 – Static tests at 25°C	5	45
Subgroup 2 – Static tests at maximum rated operating temperature	7	32
Subgroup 3 – Static tests at minimum rated operating temperature	7	32
Subgroup 4 – Dynamic tests at 25°C – Linear devices	5	45
Subgroup 5 – Dynamic tests at maximum rated operating temperature – Linear devices	7	32
Subgroup 6 – Dynamic tests at minimum rated operating temperature – Linear devices	7	32
Subgroup 7 – Functional tests at 25°C	5	45
Subgroup 8 – Functional tests at maximum and minimum rated operating temperatures	10	22
Subgroup 9 – Switching tests at 25°C – Digital devices	7	32
Subgroup 10 – Switching tests at maximum rated operating temperature – Digital devices (Note 2)	10	10
Subgroup 11 – Switching tests at minimum rated operating temperature – Digital devices (Note 2)	10	10

1. Sampling plans are based on LTPD tables of MIL-M-38510. The smaller initial sample size, based on zero rejects allowed, has been chosen unless otherwise indicated. If necessary, the sample size will be increased once to the quantity corresponding to an acceptance number of 2. The minimum reject number in all cases is 3.
2. These subgroups are usually performed during initial device characterization only.

OPTIONAL EXTENDED PROCESSING
CLASS B
Steps 101 Through 110

Advanced Micro Devices offers several extended processing options to meet customer high-reliability requirements. These are defined in AMD document 00-003. The flow chart below outlines Option B, a 160-hr burn in. Military temperature range devices processed to this flow (in the left column) meet the screening requirements of MIL-STD-883, Class B.



C

OTHER OPTIONS

Document 00-003, "Extended Processing Options", further defines Option B as well as other screening or sampling options available or special order. Available options are listed here for reference.

Option	Description	Effect
A	Modified Class A screen (Similar to Class S screening)	Provides space-grade product, following most Class S requirements of MIL-STD-883, Method 5004.
B	160-hr operating burn in	Upgrades a part from Class C to Class B.
X	Radiographic inspection (X-ray)	Related to Option A. Provides limited internal inspection of sealed parts.
S	Scanning Electron Microscope (SEM) metal inspection	Sample inspection of metal coverage of die.
V	Preseal visual inspection to MIL-STD-883, Method 2010, Cond. A	More stringent visual inspection of assemblies and die surfaces prior to seal.
P	Particle impact noise (PIN) screen with ultrasonic detection.	Detects loose particles of approximately 0.5 mil size or larger, which could affect reliability in zero-G or high vibration applications.
Q	Quality conformance inspection (Group B, C and D life and environmental tests)	Samples from the lot are stressed and tested per Method 5005. The customer's order must state which groups are required. Group B destroys 16 devices; Group C, 92 devices; Group D, 60 devices.

AmZ8000 Microprocessor Family

SALES OFFICES

SALES OFFICES AND REPRESENTATIVES

SOUTHWEST AREA
Advanced Micro Devices
9595 Wilshire Boulevard
Suite 401
Beverly Hills, California 90212
Tel: (213) 278-9700
(213) 278-9701
TWX: 910-490-2143

Advanced Micro Devices
1414 West Broadway Road
Suite 239
Tempe, Arizona 85282
Tel: (602) 244-9511
TWX: 910-950-0127

Advanced Micro Devices
1201 Dove Street
Suite 250
Newport Beach, California 92660
Tel: (714) 752-6262
TWX: 910-595-1575

Advanced Micro Devices
13777 No. Central Expy.
Suite 1008
Dallas, Texas 75243
Tel: (214) 234-5886
TWX: 910-867-4795

(SOUTHWEST AREA (Cont.)
Advanced Micro Devices
5955 Desoto Ave.
Woodland Hills, California 91367
Tel: (213) 992 4155
TWX: 910-494-4720

NORTHWEST AREA
Advanced Micro Devices
3350 Scott Boulevard
Suite 1002, Bldg. 10
Santa Clara, California 95051
Tel: (408) 727-1300
TWX: 910-338-0192

Advanced Micro Devices
7000 Broadway
Suite 401
Denver, Colorado 80221
Tel: (303) 426-7100
TWX: 910-931-2562

Advanced Micro Devices
7110 S.W. Fir Loop
Tigard, Oregon 97223
Tel: (503) 620-1021
TWX: 910-458-8797

MID-AMERICA AREA
Advanced Micro Devices
1111 Plaza Drive, Suite 420
Schaumburg, Illinois 60195
Tel: (312) 682-8660
TWX: 910-291-3589

Advanced Micro Devices
3400 West 65th Street
Suite 375
Edina, Minnesota 55435
Tel: (612) 854-6500
(612) 854-6520
TWX: 910-576-0929

Advanced Micro Devices
50 McNaughton Road
Suite 201
Columbus, Ohio 43213
Tel: (614) 457-7766
TWX: 810-339-2431

Advanced Micro Devices
33150 Schoolcraft
Livonia, Michigan 48150
Tel: (313) 425-3440
TWX: 810-242-8777

MID-ATLANTIC AREA
Advanced Micro Devices
40 Crossway Park Way
Woodbury, New York 11797
Tel: (516) 364-8020
TWX: 510-221-1819

Advanced Micro Devices
6806 Newbrook Ave.
E. Syracuse, New York 13057
Tel: (315) 437-7546
TELEX: 93-7201

Advanced Micro Devices
2 Kilmer Road
Edison, New Jersey 08817
Tel: (201) 985-6800
TWX: 710-480-6260

Advanced Micro Devices
1 Gibralter Plaza, Suite 219
110 Gibralter Road
Horsham, Pennsylvania 19044
Tel: (215) 441-8210
TWX: 510-665-7572

Advanced Micro Devices
82 Washington Street
Poughkeepsie, New York 12601
Tel: (914) 471-8180

NORTHEAST AREA
Advanced Micro Devices
300 New Boston Park
Woburn, Massachusetts 10801
Tel: (617) 933-1234
TWX: 710-348-1332

SOUTHEAST AREA
Advanced Micro Devices
793 Elkridge Landing, #11N
Linthicum, Maryland 21090
Tel: (301) 796-9310

Advanced Micro Devices
1001 N.W. 62nd Street
Suite 409
Ft. Lauderdale, Florida 33309
Tel: (305) 771-6510
TWX: 510-955-9490

Advanced Micro Devices
6755 Peachtree Industrial Boulevard
Suite 104
Atlanta, Georgia 30360
Tel: (404) 449-7920
TWX: 810-766-0430

Advanced Micro Devices International Sales Offices

BELGIUM
Advanced Micro Devices
Overseas Corporation
Avenue de Tervuren, 412, Bte 9
B-1150 Bruxelles
Tel: (02) 771 9993
TELEX: 61028

FRANCE
Advanced Micro Devices, S.A.
European Marketing Centre
27, bvd. General-Vanier
F-06400 Cannes
Tel: (093) 43.60.75
TELEX: 470966

Advanced Micro Devices, S.A.
Silic 314, Immeuble Helsinki
74, Rue d'Arcueil
F-94588 Rungis Cedex
Tel: (01) 686.91.86
TELEX: 202053

GERMANY
Advanced Micro Devices
Mikro-Elektronik GmbH
Rosenheimer Str. 139
D-8000 Muenchen 80
Tel: (089) 40 19 76
TELEX: 0-523883

Advanced Micro Devices
Mikro-Elektronik GmbH
Hartlaeuser Hauptstrasse 4
D-7024 Filderstadt 3
Tel: (07158) 62 63 0
TELEX: 0-721211

ITALY
Advanced Micro Devices S.R.L.
Centro Direzionale
Palazzo Vesalli, 3° Piano
I-20090 Mil2 – Segrate
Tel: (02) 215 4913-4-5
Telefax: (02) 656878
TLX: via Milano P.O. 311250PPMI

JAPAN
Advanced Micro Devices, K.K.
Dai 3 Hoya Building
9-17, Kaminarimon 1-Chome
Suginami-ku, Tokyo 168
Tel: (03) 329-2751
TELEX: 2324064

UNITED KINGDOM
Advanced Micro Devices (U.K.) Ltd.
A.M.D. House,
Goldsworth Road,
Woking,
Surrey GU21 1JT
Tel: Woking (24862) 22121
TELEX: 859103

International Sales Representatives and Distributors

ARGENTINA
Thiko, S.A. Electronica
Peru 653
1068 – Buenos Aires
Argentina
Tel: (1) 30-4132
(1) 33-4870
TELEX: 17825

Koti International Corporation
1660 Rollins Road
Burlingame, California 94010
Tel: (415) 692-3200
TELEX: 331491

AUSTRALIA
A.J. Distributors Pty. Ltd.
P. O. Box 71
Prospect, S.A. 5082
Tel: (8) 269-1244
TELEX: 82635

R and D Electronics
257 Burwood Highway
P.O. Box 206
Burwood 3125
Victoria
Tel: (03) 288-8232/62
TELEX: AA33288

R and D Electronics
P.O. Box 57
Crows Nest N.S.W. 2065
Sydney
Tel: (61) 439-5488
TELEX: (790) 25468

AUSTRIA
Kontron Ges.m.b.H.
Industriestr. D 13
A-2345 Brunn am Gebirge
Tel: (02236) 8 66 31
TLX: 79337

BELGIUM
MCA Tronix
62, Route du Condroz
B-4200 Ougree
Tel: (041) 362780/362795
TELEX: 42052

DENMARK
Advanced Electronic of Denmark ApS
55, Maniendalsvej
DK-2000 Copenhagen F
Tel: (01) 19 44 33
TELEX: 224 31

FINLAND
Komdel OY
Box 32
SF-02210 Espoo 21
Tel: (90) 88 50 11
TELEX: 12 1926

FRANCE
A2M
6, Avenue du Général De Gaulle
Hall A
F-78150 Le Chesnay
Tel: (01) 954.91.13
TELEX: 698 376

LED
18, Rue Henri Pensier
F-69352 Lyon Cedex 2
Tel: (078) 58.85.85
TELEX: 300 270

RTF
73, Avenue Charles De Gaulle
F-92200 Neuilly-sur-Seine
Tel: (01) 747.11.01
TELEX: 611985

GERMANY
Cosmos Elektronik GmbH
Hegelstrasse 16
D-8000 Muenchen 83
Tel: (089) 60 20 88
TELEX: 0-522545

EBV-Elektronik Vertriebs GmbH
Oberweg 6
D-8025 Unterhaching b. Muenchen
Tel: (089) 61 10 51
TLX: 0-524535

EBV-Elektronik Vertriebs GmbH
Oststr. 129
D-4000 Duesseldorf
Tel: (0211) 84 84 6
TELEX: 0-8587267

EBV-Elektronik Vertriebs GmbH
Kiebitzrain 18
D-3006 Burgwedel 1
Tel: (05139) 50 38
TLX: 0-923694

EBV-Elektronik Vertriebs GmbH
Myliusstr. 54
D-6000 Frankfurt 1
Tel: (0611) 72 04 16
TELEX: 0-413590

EBV-Elektronik Vertriebs GmbH
Alexanderstr. 42
D-7000 Stuttgart 1
Tel: (0711) 24 74 81
TELEX: 0-722271

Elbatex GmbH
Caecilienstr. 24
D-7100 Heilbronn
Tel: (07131) 89 00 1
TLX: 0-728362

Nordelektronik Vertriebs GmbH
Bahnhofstr. 14
D-2301 Kiel-Raisdorf
Tel: (04307) 54 83

Nordelektronik Vertriebs GmbH
Harksheiderweg 238-240
D-2085 Quickborn
Tel: (04106) 40 31
TELEX: 0-214299

MEV-Mikro Elektronik Vertriebs GmbH
Muenchner Strasse 16A
D-8021 Deining b. Muenchen
Tel: (08170) 71 61
TELEX: 0-527828

HOLLAND
Arcobel BV
Van Almondestraat 6
P.O. Box 344
Oss
Tel: (04120) 24200/27574
TELEX: 50835

HONG KONG
Flint Electronics Co.
Flat A, 10/F
62D Robinson Road
Tel: (5) 255732
TELEX: 63075

INDIA
SRI RAM Associates
778 Blue Sage Drive
Sunnyvale, CA 94086
Tel: (408) 737-2692
TELEX: 348369

Hindustan Semiconductor
24 Sayad Abdulla
Brahi Road
Bombay
Tel: 818105/812772
TELEX: 011-2726

SRI RAM Associates
245 Jayanagara III Block
Bangalore 560011

ISRAEL
Talviton Electronics
P.O. Box 21104
9, Bittmor Street
Tel-Aviv
Tel: (3) 444572
TELEX: VITKO 33400

ITALY
Indelco, S.r.L. – Roma
Via C. Colombo, 134
I-00147 Roma
Tel: (06) 5140722
TELEX: 611517

Indelco, S.r.L. – Milano
Via S. Simpliciano, 2
I-20121 Milano
Tel: (02) 862963

JAPAN
Advanced Technology Corporation
of Japan
Tashi Bldg., 3rd Floor
No. 8, Minami Motomachi
Shinjuku-ku, Tokyo 160
Tel: (03) 265-9416

Dainichi Electronics
Kohraku Building
1-8, 1-Chome, Kohraku
Bunkyo-ku, Tokyo
Tel: (03) 813-6876

Dainichi Electronics
Kintetsu-Takama Building
38-3 Takama-cho
Narashi 630

ISI Ltd.
8-3, 4-Chome, Lidabashi
Chiyoda-ku, Tokyo 102
Tel: (03) 264-3301

Kanematsu Denshi K.K.
Takanawa Bldg., 2nd Floor
19-26, 3-Chome, Takanawa
Minatoku, Tokyo 108

Microtek, Inc.
Naito Building
7-2-8 Nishishinjuku
Shinjuku-ku, Tokyo 160
Tel: (03) 363-2317
TELEX: J28497

KOREA
Duksung Trading Co.
Room 508 – Sindo Bldg. 65-4
2KA Chungmurq
Seoul
Tel: 778-3198
TELEX: K23459

Leewood International, Inc.
C.P.O. Box 4046
Seoul
Tel: 28-5927
778-5831
TELEX: K25582

NEW ZEALAND
G.T.S. Engineering, Ltd.
P.O. Box 42-053, Orakei
Auckland
Tel: (9) 543 214
(9) 546 745
TELEX: NZ2591

NORWAY
A/S Kjell Bakke
Nygatan 48
P.O. Box 143
N-2011 Stroemmen
Tel: (02) 71 18 72
(02) 71 43 30
TELEX: 19407

SOUTH AFRICA
South Continental Devices (Pty.) Ltd.
P.O. Box 56420
Pinegowrie 2123
Tel: (011) 789-2400
TELEX: 4-24849

SOUTH AMERICA
Intectra
2349 Charleston Road
Mountain View, CA 94043
Tel: (415) 967-8818/25
TELEX: 345 545

SPAIN
Reguia S.A.
Avda. de Ramón y Cajal, 5
Madrid-16
Tel: (1) 459 33 00/04/08
TELEX: 42 207

SWEDEN
Svensk Teleindustri AB
Box 5024
S-162 05 Vällingby
Tel: (08) 89 04 35
TELEX: 13033

SWITZERLAND
Kurt Hirt AG
Thurgauerstr. 74
CH-8050 Zuerich
Tel: (00411) 51 21 21
TELEX: 53461

TAIWAN
Multitech America Co.
19046 Bonnet Way
Saratoga, CA 95070
Tel: (408) 867-6201

Multitech International Corp.
2nd Floor
977 Min Shen E. Road
Taipei, 105, R.O.C.
Tel: 768-1232
CABLE: MULTITIC

Everdata International Corp.
11/F 219, Chung Hsiao E. Rd.
Sec. 4, Taipei
Tel: 752-9911
TELEX: 21528

UNITED KINGDOM
Candy Electronic Components
Eden House
32 Wall Road
Maidstone, Kent ME14 1XL
Tel: (0622) 54051
TELEX: 965633

Hawke-Cramer Components Ltd.
Hawke House
Green Street
Sunbury-on-Thames
Middlesex TW16 6RA
Tel: (01) 979-7799
TELEX: 923592

Dage (Great Britain), Ltd.
Haywood House
64 High Street
Pinner, Middlesex, HA5 5QA
Tel: (01) 868 0028
TELEX: 24506

ITT Electronic Services
Edinburgh Way
Harlow, Essex CM20 2DF
Tel: Harlow (0279) 26777
TELEX: 81146

Memec, Ltd.
Thame Park Industrial Estate
Thame
Oxon OX9 3RS
Tel: Thame (084 421) 3146
TELEX: 837508

Quarndon Electronics
(Semiconductors) Ltd.
Slack Lane
Derby DE3 3ED
Tel: Derby (0332) 32651
TELEX: 37163

U.S. and Canadian Sales Representatives

ALABAMA
Electronic Manufacturers Agents
2309 Stanmount Circle, S.W.
Huntsville, Alabama 35801
Tel: (205) 533-6440
TWX: 810-726-2110

CALIFORNIA
(Northern)
I² Incorporated
3350 Scott Boulevard
Suite 1001, Bldg. 10
Santa Clara, California 95050
Tel: (408) 988-3400
TWX: 910-338-0192

(Southern)
Bestronics
10150 Sorrento Valley Road
Suite 300
San Diego, California 92121
Tel: (714) 452-5500

CANADA (Eastern)
Vitel Electronics
3300 Cote Vertu, Suite 203
St. Laurent, Quebec,
Canada H4R 2B7
Tel: (514) 331-7393
TWX: 610-421-3124
TLX: 05-821762

Vitel Electronics
1 Vulcan St., Suite 203
Rexdale, Ontario,
Canada M9W 1L3
Tel: (416) 245-8528
TWX: 610-491-3728

Shipping:
Vitel Electronics
84 Main Street
Champlain, New York 12919

Vitel Electronics
85 Albert Street, Suite 1610
Ottawa, Ontario
Canada K1P 6A4
Tel: (613) 236-0396
TLX: 0533198

CANADA (Western)
Venture Electronics
P.O. Box 3004
Bellevue, Washington 98009
Tel: (206) 454-4594
TLX: 32-8951

Shipping:
1645 Rambling Lane
Bellevue, Washington 98004

Vitel Electronics
3665 Kingsway, Suite 211
Burnaby, British Columbia
Canada V5R 5W2
Tel: (604) 438-6121
TWX: 610-953-4925

COLORADO
R² Marketing
10018 No. Regency Place
P.O. Box 554
Parker, Colorado 80134
Tel: (303) 771-7580

CONNECTICUT
Scientific Components
1185 South Main Street
Cheshire, Connecticut 06410
Tel: (203) 272-2965
TWX: 710-455-2078

FLORIDA
Conley & Associates, Inc.
P.O. Box 309
235 South Central Ave.
Oviedo, Florida 32765
Tel: (305) 365-3283
TWX: 810-856-3520

Conley & Associates, Inc.
1612 N.W. Second Ave.
P.O. Box 700
Boca Raton, Florida 33432
Tel: (305) 395-6108
TWX: 510-953-7548

Conley & Associates, Inc.
4021 W. Waters Avenue
Suite 2
Tampa, Florida 33614
Tel: (813) 885-7658
TWX: 810-876-9136

GEORGIA
Electronic Manufacturers Agents
6755 Peachtree Industrial Blvd., N.E.
Suite 103
Atlanta, Georgia 30360
Tel: (404) 449-9430
TWX: 810-766-0430

ILLINOIS
Oasis Sales, Inc.
1101 Towne Road
Elk Grove Village, Illinois 60007
Tel: (312) 640-1850
TWX: 910-222-1775

INDIANA
C-S Electronic Sales, Inc.
2122-A Miami Street
South Bend, Indiana 46613
Tel: (219) 291-8383
TWX: 810-299-2535

C-S Electronic Sales, Inc.
1157-B South Jackson
Frankfort, Indiana 46041
Tel: (317) 659-1874

IOWA
Lorenz Sales, Inc.
5270 No. Park Pl., N.E.
Cedar Rapids, Iowa 52402
Tel: (319) 377-4666

KANSAS
Kebco Manufacturers
7070 West 107th Street
Suite 160
Overland Park, Kansas 66212
Tel: (913) 649-1051
TWX: 910-749-4977

Kebco Manufacturers
9813 England
Overland Park, Kansas 66211

MARYLAND
Burgin-Kreh Associates, Inc.
6100 Baltimore National Pike
Baltimore, Maryland 21228
Tel: (301) 788-5200
TWX: 710-862-1450

MICHIGAN
S.A.I. Marketing Corp.
P.O. Box N
Brighton, Michigan 48116
Tel: (313) 227-1786
TWX: 810-242-1518

Shipping:
First Federal Bank Building
Suite 109
9880 E. Grand River Avenue
Brighton, Michigan 48116

MISSOURI
Kebco Manufacturers
75 Worthington Drive, Ste. 10
Maryland Heights, Missouri 63043
Tel: (314) 576-4111
TWX: 910-764-0826

NEBRASKA
Lorenz Sales
2809 Garfield Avenue
Lincoln, Nebraska 68502
Tel: (402) 475-4660

NEW MEXICO
The Thorson Company
1101 Cardenas, N.E.
Suite 109
Albuquerque, New Mexico 87110
Tel: (505) 265-5655
TWX: 910-989-1174

NEW YORK
Nycom, Inc.
10 Adler Drive
East Syracuse, New York 13057
Tel: (315) 437-8343
TWX: 710-541-1506

NORTH CAROLINA
Burgin-Kreh Associates, Inc.
P.O. Box 19510
Raleigh, North Carolina 27609
Tel: (919) 781-1100
TWX: 510-928-0540

Shipping:
3901 Barrett Drive
Raleigh, North Carolina 27609

OHIO
Dolfuss-Root & Co.
13477 Prospect Road
Strongsville, Ohio 44136
Tel: (216) 238-0300
TWX: 810-427-9148

Dolfuss-Root & Co.
683 Miamisburg-Centerville Road
Suite 202
Centerville, Ohio 45459
Tel: (513) 433-6776

PENNSYLVANIA
Bacon Electronic Sales
115 South High Street
Waterford, Pennsylvania 16441
Tel: (814) 796-2381
TWX: 510-699-6870

TENNESSEE
(Western)
Burgin-Kreh Associates, Inc.
12 Skyline Drive
Kingston Heights
Kingston, Tennessee 37763

EMA
11305 Silver Springs Drive
Knoxville, Tennessee 37922
Tel: (615) 966-1286

(Eastern)
Burgin-Kreh Associates, Inc.
P.O. Box 268
12 Skyline Dr.
Kingston Heights,
Kingston, Tennessee 37763
Tel: (615) 690-6100

TEXAS
Bonser-Phillhower Sales
13777 N. Central Expressway
Suite 212
Dallas, Texas 75243
Tel: (214) 234-8436

Bonser-Phillhower
3300 Chimneyrock, Suite 208
Houston, Texas 77056
Tel: (713) 783-0063

Bonser-Phillhower Sales
8330 Burnett Rd.
Suite 133
Austin, Texas 78758
Tel: (512) 458-3569

UTAH
R²
940 North 400 East, Suite B
North Salt Lake, Utah 84054
Tel: (801) 298-2631
TWX: 910-925-5607

VIRGINIA
Burgin-Kreh Associates, Inc.
P.O. Box 4254
5521 Fort Avenue
Lynchburg, Virginia 24502
Tel: (804) 239-2626

WASHINGTON
Venture Electronics
150 116th N.E., Suite 109
P.O. Box 3034
Bellevue, Washington 98005
Tel: (206) 454-4594
TELEX: 32-8951

WISCONSIN
Oasis Sales, Inc.
N. 81 W. 12920 Leon Road
Suite 11
Menomonee Falls, Wisconsin 53051
Tel: (414) 445-6682

U.S. AND CANADIAN STOCKING DISTRIBUTORS

ALABAMA
Hamilton/Avnet Electronics
4812 Commercial Drive
Huntsville, Alabama 35805
Tel: (205) 533-1170

Hall-Mark Electronics
4739 Commercial Drive
Huntsville, Alabama 35805
Tel: (205) 837-8700

ARIZONA
Wyle Distribution Group
8155 North 24th Avenue
Phoenix, Arizona 85021
Tel: (602) 249-2232

Hamilton/Avnet Electronics
2615 S. 21st Street
Phoenix, Arizona 85034
Tel: (602) 275-7851
TWX: 910-951-1535

CALIFORNIA
Avnet Electronics
350 McCormick Avenue
Irvine Industrial Complex
Costa Mesa, California 92626
Tel: (714) 754-6084
TWX: 910-595-1928

Bell Industries
1161 N. Fairoaks Avenue
Sunnyvale, California 94086
Tel: (408) 734-8570
TWX: 910-339-9378

Hamilton Electro Sales
10950 W. Washington Blvd.
Culver City, California 90230
Tel: (213) 558-2100
(714) 522-8220
TWX: 910-340-6364
910-340-7073
TELEX: 67-36-92

Hamilton/Avnet Electronics
1175 Bordeaux
Sunnyvale, California 94086
Tel: (408) 743-3300
TWX: 910-339-9332

Hamilton/Avnet Electronics
4545 View Ridge Road
San Diego, California 92123
Tel: (714) 571-7500
TELEX: 69-54-15

Hamilton/Avnet Electronics
3170 Pullman
Costa Mesa, California 92626
Tel: (714) 979-6864

Wyle Distribution Group
9525 Chesapeake Drive
San Diego, California 92123
Tel: (714) 565-9171
TWX: 910-335-1590

Schweber Electronics
17811 Gillette
Irvine, California 92714
Tel: (213) 537-4320
TWX: 910-595-1720

Wyle Distribution Group
124 Maryland Avenue
El Segundo, California 90545
Tel: (213) 322-8100
TWX: 910-348-7140
910-348-7111

Arrow Electronics
720 Palomar Ave.
Sunnyvale, California 94086
Tel: (408) 739-3011
TWX: 910-339-9371

Wyle Distribution Group/Santa Clara
3000 Bowers Avenue
Santa Clara, California 95052
Tel: (408) 727-2500
TWX: 910-338-0296
910-338-0541

Wyle Distribution Group
17981 Skypark Circle
Suite M
Irvine, California 92713
Tel: (714) 751-9850

Wyle Distribution Group
Orange County Division
17872 Cowan
Irvine, California 92714
Tel: (714) 641-1600

CANADA
Hamilton/Avnet Electronics
2670 Sabourin
St. Laurent, Quebec, Canada H4S1M2
Tel: (514) 331-6443
TWX: 610-421-3731

Hamilton/Avnet Electronics
3688 Nashua Road
Mississauga, Ontario, Canada L4V1M5
Tel: (416) 677-7432
TWX: 610-492-8867

Hamilton/Avnet Electronics
1735 Courtwood Crescent
Ottawa, Ontario, Canada K2C3J2
Tel: (613) 226-1700
TWX: 610-562-1906

RAE Industrial Electronics, Ltd.
3455 Gardner Court
Burnaby, British Columbia
Canada V5G 4J7
Tel: (604) 291-8866
TWX: 610-929-3065
TLX: 04-356533

Future Electronics
5647 Ferrier Street
Montreal, Quebec, Canada H4P2K5
Tel: (514) 731-7441
TWX: 610/421-3251
05-827789

Future Electronics
4800 Dufferin Street
Downsview, Ontario
Canada M3H 5S9
Tel: (416) 663-5563

Future Electronics
3070 Kingsway
Vancouver, British Columbia
Canada V5R 5J7
Tel: (604) 438-5545
TWX: 610-922-1668

Future Electronics
Baxter Centre
1050 Baxter Rd.
Ottawa, Ontario
Canada K2C 3P2
Tel: (613) 820-8313

COLORADO
Wyle Distribution Group
6777 E. 50th Avenue
Commerce City, Colorado 80022
Tel: (303) 287-9611
TWX: 910-936-0770

Hamilton/Avnet Electronics
8765 E. Orchard Road
Suite 708
Englewood, Colorado 80111
Tel: (303) 740-1000

Bell Industries
8155 W. 48th Avenue
Weatridge, Colorado 80033
Tel: (303) 424-1985
TWX: 910-938-0393

CONNECTICUT
Hamilton/Avnet Electronics
643 Danbury Road
Georgetown, Connecticut 06829
Tel: (203) 762-0361

Schweber Electronics
Finance Drive
Commerce Industrial Park
Danbury, Connecticut 06810
Tel: (203) 792-3500

Arrow Electronics
295 Treadwell Street
Hamden, Connecticut 06514
Tel: (203) 248-3801
TWX: 710-465-0780

Wilshire Electronics
2554 State Street
Hamden, Connecticut 06517
Tel: (203) 281-1166
TWX: 710-465-0747

FLORIDA
Arrow Electronics
115 Palm Road N.W.
Suite 10
Palm Bay, Florida 22905
Tel: (305) 725-1480

Arrow Electronics
1001 N.W. 62nd St., Suite 402
Ft. Lauderdale, Florida 33300
Tel: (305) 776-7790

Hall-Mark Electronics
7233 Lake Ellenor Dr.
Orlando, Florida 32809
Tel: (305) 855-4020
TWX: 810-850-0183

Hall-Mark Electronics
1302 West McNabb Road
Ft. Lauderdale, Florida 33309
Tel: (305) 971-9280
TWX: 510-956-9720

Hamilton/Avnet Electronics
6800 N.W. 20th Ave.
Ft. Lauderdale, Florida 33309
Tel: (305) 971-2900

Hamilton/Avnet Electronics
3197 Tech Drive North
St. Petersburg, Florida 33702
Tel: (813) 576-3930

Pioneer/Florida
6220 S. Orange Blossom Trail
Suite 412
Orlando, Florida 32809
Tel: (305) 859-3600
TWX: 810-850-0177

GEORGIA
Arrow Electronics
2979 Pacific Drive
Norcross, Georgia 30071
Tel: (404) 449-8252
TWX: 810/766-0439

Hamilton/Avnet Electronics
6700 1-85
Suite 2B
Norcross, Georgia 30071
Tel: (404) 448-0800

ILLINOIS
Arrow Electronics
492 Lunt Avenue
Schaumburg, Illinois 60193
Tel: (312) 893-9420

Hall-Mark Electronics
1177 Industrial Drive
Bensenville, Illinois 60106
Tel: (312) 860-3800
TWX: 910-222-1815

Hamilton/Avnet Electronics
3901 North 25th Avenue
Schiller Park, Illinois 60176
Tel: (312) 678-6310
TWX: 910-227-0060

Pioneer/Chicago
1551 Carmen Drive
Elk Grove Village, Illinois 60007
Tel: (312) 437-9680
TWX: 910-222-1834

INDIANA
Pioneer/Indiana
6408 Castle Place Drive
Indianapolis, Indiana 46250
Tel: (317) 849-7300
TWX: 810-260-1794

KANSAS
Hall-Mark Electronics
11870 West 91st Street
Congleton Industrial Park
Shawnee Mission, Kansas 66214
Tel: (913) 888-4747
TWX: 510-928-1831

Hamilton/Avnet Electronics
9219 Quivira Road
Overland Park, Kansas 66215
Tel: (913) 888-8900

MARYLAND
Arrow Electronics
4801 Benson Avenue
Baltimore, Maryland 21227
Tel: (301) 247-5200

Hall-Mark Electronics
6655 Amberton Drive
Baltimore, Maryland 21227
Tel: (301) 796-9300
TWX: 710-862-1942

Hamilton/Avnet Electronics
7235 Standard Drive
Hanover, Maryland 21076
Tel: (301) 796-5000
TWX: 710-862-1861
TELEX: 8-79-68

Pioneer/Washington
9100 Gaither Road
Gaithersburg, Maryland 20760
Tel: (301) 948-0710
TWX: 710-828-0545

MASSACHUSETTS
Arrow Electronics
96D Commerce Way
Woburn, Massachusetts 01801
Tel: (617) 933-8130
TWX: 510-224-6494

Hamilton/Avnet Electronics
50 Tower Office Park
Woburn, Massachusetts 01801
Tel: (617) 935-9700
TWX: 710-393-0382

Schweber Electronics
25 Wiggins Road
Bedford, Massachusetts 01730
Tel: (617) 275-5100

Wilshire Electronics
One Wilshire Road
Burlington, Massachusetts 01803
Tel: (617) 272-8200
TWX: 710-332-6359

MICHIGAN
Arrow Electronics
3810 Varsity Drive
Ann Arbor, Michigan 48104
Tel: (313) 971-8220
TWX: 810-223-6020

Hamilton/Avnet Electronics
32487 Schoolcraft
Livonia, Michigan 48150
Tel: (313) 522-4700
TWX: 810-242-8775

Pioneer/Michigan
13485 Stamford
Livonia, Michigan 48150
Tel: (313) 525-1800
TWX: 810-242-3271

MINNESOTA
Arrow Electronics
5230 W. 73rd Street
Edina, Minnesota 55435
Tel: (612) 830-1800

Hall-Mark Electronics
9201 Penn Avenue South
Suite 10
Bloomington, Minnesota 55431
Tel: (612) 884-9056
TWX: 910-576-3187

Hamilton/Avnet Electronics
7449 Cahill Rd.
Edina, Minnesota 55435
Tel: (612) 941-3801

MISSOURI
Hall-Mark Electronics
13789 Rider Trail
Earth City, Missouri 63045
Tel: (314) 291-5350
TWX: 910-760-0671

Hamilton/Avnet Electronics
13743 Shoreline Court
Earth City, Missouri 63045
Tel: (314) 344-1200

NEW JERSEY
Arrow Electronics
Pleasant Valley Road
Moorestown, New Jersey 08057
Tel: (609) 235-1900

Arrow Electronics
285 Midland Ave.
Saddle Brook, New Jersey
Tel: (201) 797-5800
TWX: 710-988-2206

Hamilton/Avnet Electronics
10 Industrial Road
Fairfield, New Jersey 07006
Tel: (201) 575-3390

Hamilton/Avnet Electronics
1 Keystone Avenue
Cherry Hill, New Jersey 08003
Tel: (609) 424-0100

Schweber Electronics
18 Madison Road
Fairfield, New Jersey 07006
Tel: (201) 227-7880
TWX: 710-480-4733

Wilshire Electronics
1111 Paulison Avenue
Clifton, New Jersey 07015
Tel: (201) 340-1900
TWX: 710-989-7052

NEW MEXICO
Bell Industries
11728 Linn N.E.
Albuquerque, New Mexico 87123
Tel: (505) 292-2700
TWX: 910-989-0625

Hamilton/Avnet Electronics
2524 Baylor Drive S.E.
Albuquerque, New Mexico 87119
Tel: (505) 765-1500

Electronic Devices Co., Inc.
3301 Juan Tabo N.E.
Albuquerque, New Mexico 87111
Tel: (505) 293-1935

NEW YORK
Arrow Electronics
900 Broad Hollow Road
Farmingdale, New York 11735
Tel: (516) 694-6800
TWX: 510-224-6155

Arrow Electronics
7705 Maltage Drive
Liverpool, New York 13088
Tel: (315) 652-1000
TWX: 710-545-0230

Arrow Electronics
3000 South Winton Road
Rochester, New York 14623
Tel: (716) 275-0300
TWX: 510-253-4766

Hamilton/Avnet Electronics
167 Clay Road
Rochester, New York 14623
Tel: (716) 442-7820

Hamilton/Avnet Electronics
5 Hub Drive
Melville, New York 11746
Tel: (516) 454-6000
TWX: 510-224-6166

Hamilton/Avnet Electronics
6500 Joy Road
E. Syracuse, New York 13057
Tel: (315) 437-2642
TWX: 710-541-0959

Summit Distributors, Inc.
916 Main Street
Buffalo, NY 14202
Tel: (716) 884-3450
TWX: 710-522-1692

Wilshire Electronics
110 Parkway South
Hauppauge
Long Island, NY 11787
Tel: (516) 543-5599

Wilshire Electronics
1260 Scottsville Road
Rochester, New York 14623
Tel: (716) 235-7620
TWX: 510-253-5226

Wilshire Electronics
10 Hooper Road
Endwell, New York 13760
Tel: (607) 754-1570
TWX: 510-252-0194

Schweber Electronics
Jericho Turnpike
Westbury, New York 11590
Tel: (516) 334-7474
TWX: 510-222-9470
510-222-3660

NORTH CAROLINA
Arrow Electronics
1337-G South Park Drive
Kernersville, North Carolina 27284
Tel: (919) 996-2039

Hall-Mark Electronics
1208 Front Street, Building K
Raleigh, North Carolina 27609
Tel: (919) 832-4465
TWX: 510-928-1831

Hamilton/Avnet Electronics
2803 Industrial Drive
Raleigh, North Carolina 27609
Tel: (919) 829-8030

OHIO
Arrow Electronics
6238 Cochran
Solon, Ohio 44139
Tel: (216) 248-3990

Arrow Electronics
3100 Plainfield Road
Kettering, Ohio 45432
Tel: (513) 253-9176
TWX: 810-459-1611

Hamilton/Avnet Electronics
954 Senate Drive
Dayton, Ohio 45459
Tel: (513) 433-0610
TWX: 810-450-2531

Hamilton/Avnet
761 Beta Drive, Suite E
Cleveland, Ohio 44143
Tel: (216) 461-1400

Arrow Electronics
P.O. Box 37856
Cincinnati, Ohio 45222
Tel: (513) 761-5432
TWX: 810-461-2670

Pioneer/Cleveland
4800 E. 131st Street
Cleveland, Ohio 44105
Tel: (216) 587-3600
TWX: 810-422-2211

OKLAHOMA
Hall-Mark Electronics
5460 S. 103rd E. Avenue
Tulsa, Oklahoma 74145
Tel: (918) 835-8458
TWX: 910-845-2290

OREGON
Hamilton/Avnet Electronics
6024 S.W. Jean Road
Bldg. C, Suite 10
Lake Oswego, Oregon 97034

PENNSYLVANIA
Hall-Mark Electronics
458 Pike Road
Pike Industrial Park
Huntingdon Valley,
Pennsylvania 19006
Tel: (215) 355-7300
TWX: 510-667-1750

Schweber Electronics
101 Rock Road
Horsham, Pennsylvania 19044
Tel: (215) 441-0600

Pioneer/Pittsburgh
560 Alpha Drive
Pittsburgh, Pennsylvania 15238
Tel: (412) 782-2300
TWX: 710-795-3122

TEXAS
Hall-Mark Electronics
P.O. Box 22035
11333 Page Mill Road
Dallas, Texas 75222
Tel: (214) 234-7300
TWX: 910-867-4721

Hall-Mark Electronics
8000 Westglen
Houston, Texas 77063
Tel: (713) 781-6100
TWX: 910-881-2711

Hall-Mark Electronics
10109 McKalla Drive
Suite F
Austin, Texas 78758
Tel: (512) 837-2814
TWX: 910-874-2010

Hamilton/Avnet Electronics
4445 Sigma Road
Dallas, Texas 75240
Tel: (214) 661-8661
TELEX: 73-05-11

Hamilton/Avnet Electronics
3939 Ann Arbor Street
P.O. Box 42802
Houston, Texas 77042
Tel: (713) 780-1771

Hamilton/Avnet Electronics
10508A Boyer Blvd.
Austin, Texas 78757
Tel: (512) 837-8911

Schweber Electronics
14177 Proton Road
Dallas, Texas 75240
Tel: (214) 661-5010
TWX: 910-860-5493

Schweber Electronics
7420 Harwin Drive
Houston, Texas 77036
Tel: (713) 784-3600

UTAH
Bell Industries
3639 W. 2150 South
Salt Lake City, Utah 84120
Tel: (801) 972-6969
TWX: 910-925-5686

Hamilton/Avnet Electronics
1585 West 2100 South
Salt Lake City, Utah 84119
Tel: (801) 972-2800
TWX: 910-925-4018

WASHINGTON
Hamilton/Avnet Electronics
14212 N.E. 21st Street
Bellevue, Washington 98005
Tel: (206) 746-8750
TWX: 910-443-2449

Wyle Distribution Group
1750 132nd Avenue N.E.
Bellevue, Washington 98005
Tel: (206) 453-8300
TWX: 910-443-2526

WISCONSIN
Arrow Electronics
434 W. Rawson Avenue
Oak Creek, Wisconsin 53154
Tel: (414) 764-6600
TWX: 910-262-1193

Hall-Mark Electronics
9657 S. 20th St.
Oak Creek, Wisconsin 53154
Tel: (414) 761-3000

Hamilton/Avnet Electronics
2975 Moorland Road
New Berlin, Wisconsin 53151
Tel: (414) 784-4510

ADVANCED MICRO DEVICES, INC.
901 Thompson Place
P.O. Box 453
Sunnyvale,
California 94086
(408) 732-2400
TWX: 910-339-9280
TELEX: 34-6306
TOLL FREE
(800) 538-8450