80960XA
EMBEDDED 32-BIT MICROPROCESSOR WITH 33RD TAG BIT TO SUPPORT OBJECT-ORIENTED PROGRAMMING AND DATA SECURITY

Military

- Implements JIAWG 32-Bit ISA Standard
- High-Performance Embedded Architecture
  - 25 MIPS Burst Execution at 25 MHz
  - 9.4 MIPS* Sustained Execution at 25 MHz
- On-Chip Floating-Point Unit
  - Supports IEEE 754 Floating-Point Standard
  - Full Transcendental Support
  - Four 80-Bit Registers
  - 5.2 Million Whetstones/Second at 25 MHz
- Multiple Register Sets
  - Sixteen Global 32-Bit Registers
  - Sixteen Local 32-Bit Registers
  - Four Local Register Sets Stored On-Chip (Sixteen 32-Bit Registers per Set)
  - Register Scoreboarding
- Object Oriented Programming and Data Security Supported within Hardware
  - 33rd Tag Bit to Distinguish Data from Object Pointer
  - Unforgeable Pointers to Memory
- On-Chip Memory Management Unit
  - 4 Gigabyte Linear Address Space per Task
  - 4 Kbyte Pages with Supervisor/User Protection
  - 2^{58} Virtual Address Space for Object Addressing and Protection
- Built-In Interrupt Controller
  - 32 Priority Levels
  - 248 Vectors
  - Supports M8259A
  - 3.4 μs Latency
- Easy to Use, High Bandwidth 32-Bit Bus
  - 66.7 MBytes/s Burst at 25 MHz
  - Up to 16-Bytes Transferred per Burst
- Multitasking Support
  - Automatic Task Dispatching
  - Prioritized Task Queues
- Advanced Package Technology
  - 132 Lead Ceramic Pin Grid Array
  - 164 Lead Ceramic Quad Flatpack
- Military Temperature Range
  - $-55°C$ to $+125°C$ (T_C)

10

The 80960XA is the second military member of Intel's 32-bit i960® microprocessor family designed especially for embedded applications. It is based on the high performance, common core architecture of the i960 family. As a superset of the military 80960MC, the 80960XA contains the same 512-byte instruction cache, on-chip interrupt controller, integrated floating-point unit and memory management unit of the 80960MC. In addition to the features of the 80960MC, the 80960XA supports object-oriented programming via a 33rd tag bit within the hardware. This tag bit provides the distinction between a 32-bit data word and a 32-bit pointer to memory. In addition, the tag bit prohibits the possibility of forged pointers to protected areas within memory, thus providing a high level of data security within the architecture. The 80960XA implements the Extended Architecture chosen by the Joint Integrated Avionics Working Group (JIAWG) as a 32-bit Instruction Set Architecture (ISA) standard. Thus, the 80960XA is well-suited for military and other high reliability applications which require high levels of data security or require compliance to JIAWG standards.

*Relative to Digital Equipment Corporation's VAX-11/780** at 1 MIPS

**VAX-11 is a trademark of Digital Equipment Corporation.

October 1992
Order Number: 271159-003

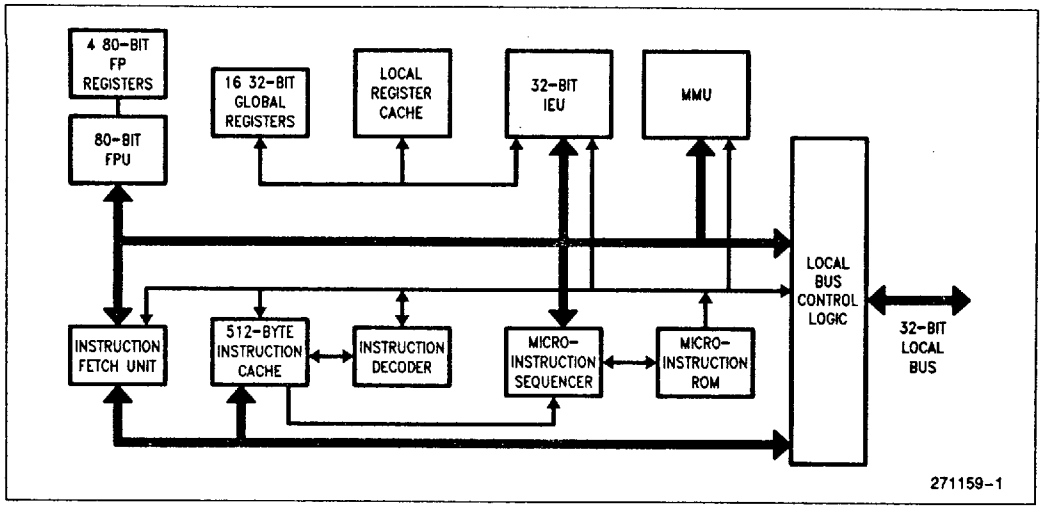


Figure 1. The 80960XA's Highly Parallel Microarchitecture

THE 960 SERIES

The 80960XA is an enhanced military member of the family of 32-bit microprocessors from Intel known as the i960 Processor Series. This series was especially designed to serve the needs of embedded applications. The embedded market includes applications as diverse as industrial automation, avionics, image processing, graphics, robotics, telecommunications, and automobiles. These types of applications require high integration, low power consumption, quick interrupt response times, and high performance. Since time to market is critical, embedded microprocessors need to be easy to use in both hardware and software designs.

All members of the 80960 series share a common core architecture which utilizes RISC technology so that, except for special functions, the family members are object code compatible. Each new processor in the series will add its own special set of functions to the core to satisfy the needs of a specific application or range of applications in the embedded market.

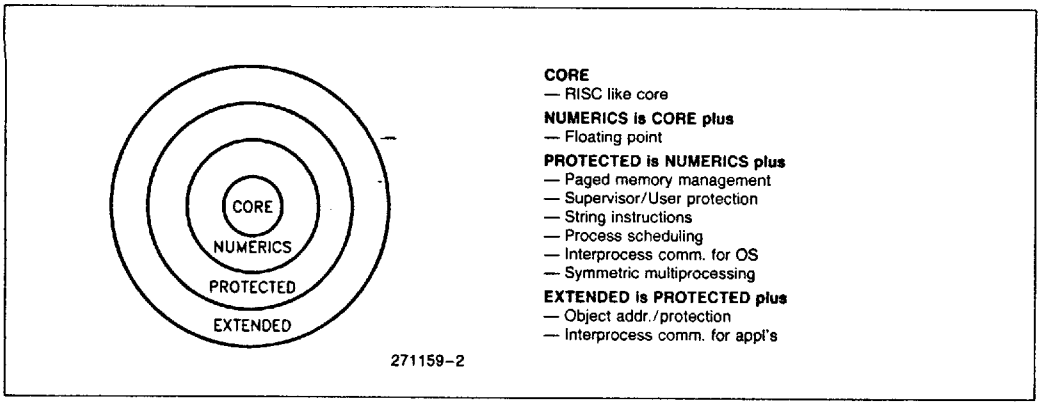
The present generation of the i960 family features four distinct architectures. Figure 2 depicts the four architectures in a ring structure to indicate upward binary compatibility. Thus, the inner ring architectures are fully binary compatible to the outer ring architectures. Future implementations of these i960 architectures will maintain this binary compatibility.

There are currently five i960 microprocessor products representing the four architectures of the family. The Core architecture is featured in Intel's commercial product line by the 80960KA and 80960CA products. The Core architecture contains the basic i960 processor instruction set. The Numerics architecture is featured in Intel's commercial product line by the 80960KB. This architecture contains the Core architecture plus an on-chip IEEE 754 Standard Floating Point Unit (FPU).

The Protected architecture is represented in Intel's military product line by the 80960MC. This architecture contains the Numerics architecture plus an on-chip Memory Management Unit (MMU) and multitasking capabilities to support the Ada programming language.

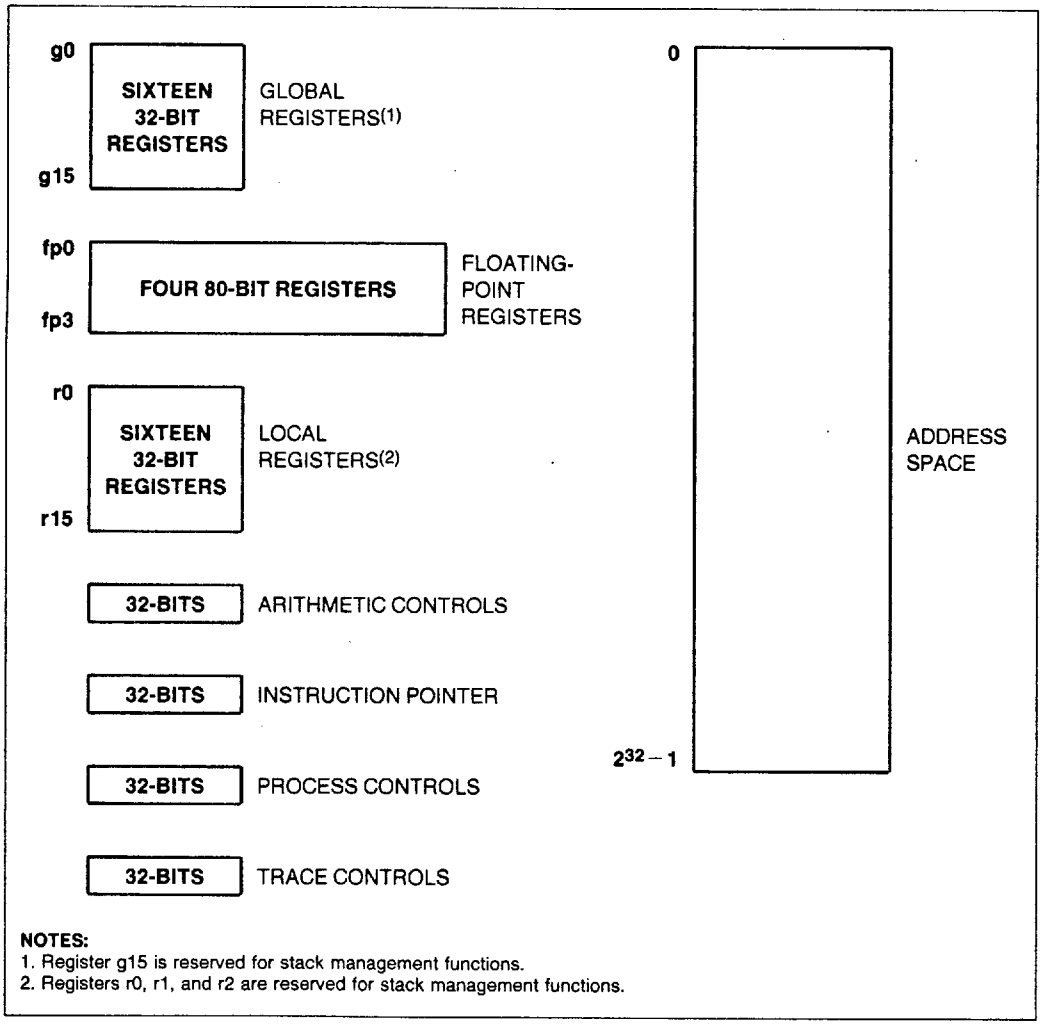
The Extended architecture, implemented by the 80960XA, was defined to support object addressing and protection in response to the increasing popularity of object-oriented computing. A superset of the Protected architecture, the Extended architecture features a 33rd tag bit on the data bus. This 33rd tag bit allows a 32-bit word to be defined as either a data word or a pointer to an object in memory, thus supporting object addressing. In addition, the tag bit provides a high level of data security by making it impossible to forge pointers to protected objects within memory.

The Extended architecture was chosen by the Joint Integrated Avionics Working Group (JIAWG) as a 32-bit Instruction Set Architecture (ISA) standard for use in Military avionics applications. Thus, the 80960XA is an ideal fit in applications for which JIAWG ISA compliance is required.



10

Figure 2. i960® Product Family



NOTES:
1. Register g15 is reserved for stack management functions.
2. Registers r0, r1, and r2 are reserved for stack management functions.

Figure 3. Register Set

KEY PERFORMANCE FEATURES

The 80960XA's architecture is based on the most recent advances in RISC technology and is grounded in Intel's long experience in designing embedded controllers. Many features contribute to the 80960XA's exceptional performance:

1. Large Register Set. Having a large number of registers reduces the number of times that a processor needs to access memory. Modern compilers can take advantage of this feature to optimize execution speed. For maximum flexibility, the 80960XA provides thirty-two 32-bit registers (sixteen local and sixteen global) and four 80-bit floating-point global registers. (See Figure 3.)

2. Fast Instruction Execution. Simple functions make up the bulk of instructions in most programs, so that execution speed can be greatly improved by ensuring that these core instructions execute in as short a time as possible. The most-frequently executed instructions such as register-register moves, add/subtract, logical operations, and shifts execute in one to two cycles (Table 1 contains a list of instructions.)

3. Load/Store Architecture. Like other processors based on RISC technology, the 80960XA has a Load/Store architecture. Only the LOAD and STORE instructions reference memory; all other instructions operate on registers. This type of architecture simplifies instruction decoding and is used in combination with other techniques to increase parallelism.

4. Simple Instruction Formats. All instructions in the 80960XA are 32-bits long and must be aligned on word boundaries. This alignment makes it possible to eliminate the instruction-alignment stage in the pipeline. To simplify the instruction decoder further, there are only five instruction formats and each instruction uses only one format. (See Figure 4.)

5. Overlapped Instruction Execution. A load operation allows execution of subsequent instructions to continue before the data has been returned from memory, so that these instructions can overlap the load. The 80960XA manages this process transparently to software through the use of a register scoreboard. Conditional instructions also make use of a scoreboard so that subsequent unrelated instructions can be executed while the conditional instruction is pending.

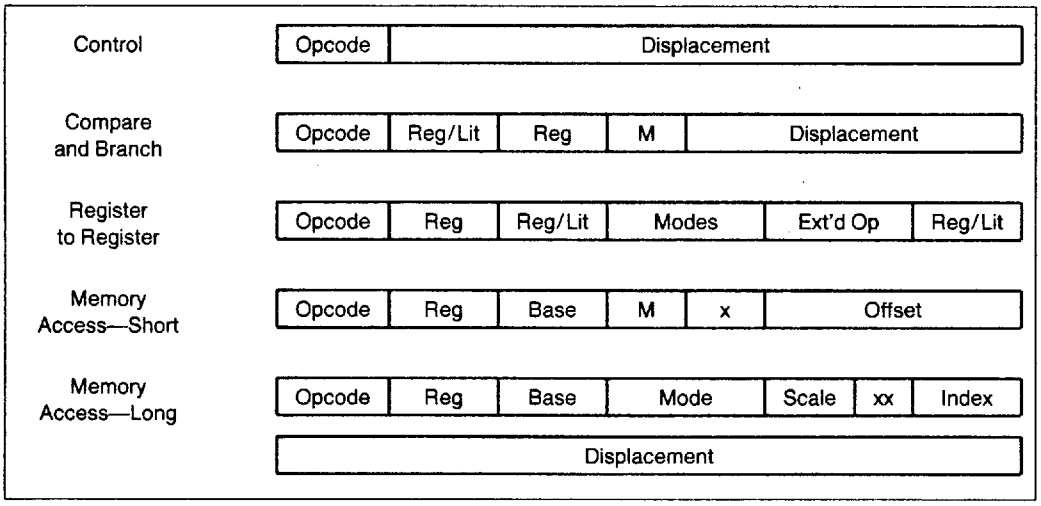


Figure 4. Instruction Formats

Table 1. 80960XA Instruction Set

Data Movement	Arithmetic	Floating Point	Logical
Load Store Move Load Address Load Physical Address Load Virtual Load Mixed Load Virtual Mixed Store Virtual Store Mixed Store Virtual Mixed Move Mixed Load TDO Load Control Stack Pointer Load from Process Globals	Add Subtract Multiply Divide Remainder Modulo Shift	Add Subtract Multiply Divide Remainder Scale Round Square Root Sine Cosine Tangent Arctangent Log Log Binary Log Natural Exponent Classify Copy Real Extended Compare	And Not And And Not Or Exclusive Or Not Or Or Not Nor Exclusive Nor Not Nand Rotate
Comparison	**Branch**	**Bit and Bit Field**	**String**
Compare Conditional Compare Compare and Increment Compare and Decrement Compare Mixed	Unconditional Branch Conditional Branch Compare and Branch	Set Bit Clear Bit Not Bit Check Bit Alter Bit Scan for Bit Scan over Bit Extract Modify	Move String Move Quick String Fill String Compare String Scan Byte for Equal
Conversion	**Decimal**	**Call/Return**	**Process Management**
Convert Real to Integer Convert Integer to Real Convert Address	Move Add with Carry Subtract with Carry	Call Call Extended Call System Return Branch and Link Call Domain	Schedule Process Saves Process Resume Process Load Process Time Modify Process Controls Wait Conditional Wait Signal Receive Conditional Receive Send Send Service Atomic Add Atomic Modify Atomic Replace Mixed
Fault	**Debug**	**Miscellaneous**	**AD Manipulation**
Conditional Fault Synchronize Faults	Modify Trace Controls Mark Force Mark	Flush Local Registers Inspect Access Modify Arithmetic Controls Test Condition Code	Amplify Rights Restrict Rights Create AD Check Tag

6. Integer Execution Optimization. When the result of an operation is used as an operand in a subsequent calculation, the value is sent immediately to its destination register. Yet at the same time, the value is put back on a bypass path to the ALU, thereby saving the time that otherwise would be required to retrieve the value for the next operation.

7. Bandwidth Optimizations. The 80960XA gets optimal use of its memory bus bandwidth because the bus is tuned for use with the cache: the line size of the instruction cache matches the maximum burst size for instruction fetches. The 80960XA automatically fetches four words in a burst and stores them directly in the cache. Due to the size of the cache and the fact that it is continually filled in anticipation of needed instructions in the program flow, the 80960XA is exceptionally insensitive to memory wait states. The benefit is that the 80960XA will deliver outstanding performance even with a low cost memory system.

8. Cache Bypass. If there is a cache miss, the processor fetches the needed instruction, then sends it on to the instruction decoder at the same time it updates the cache. Thus, no extra time is taken to load and read the cache.

Memory Space and Addressing Modes

The 80960XA allows each process to address a linear memory space of up to 4 Gbytes. This address space is divided into four 1-Gbyte regions. Each of these regions can be mapped to physical addresses by zero, one or two levels of page tables. The region with the highest addresses (Region 3) is common to all processes.

By utilizing object addressing, the linear address space of the 80960XA for a single process is extended to a virtual address space of 2^{58} bytes. In addition, the number of distinct address spaces within a running process is effectively unlimited so that each software module can be placed in its own address space.

The processor provides two address-interpretation modes: physical-addressing mode and virtual-addressing mode. When operating in physical-addressing mode, the processor interprets each address operand in an instruction as a physical address and sends the address out to the bus unchanged.

In virtual-addressing mode, the processor interprets each address operand as a virtual address. The on-chip memory management unit (MMU) translates the virtual address into a physical address, which the processor sends out to the bus. The physical addresses of the most recently accessed pages in memory are stored on-chip within the translation lookaside buffer (TLB).

The 80960XA provides a rich set of addressing modes to support efficient execution of the Ada progamming language. Table 2 lists these memory addressing modes.

Data Types

The 80960XA recognizes the following data types:

Numeric:
- 8-, 16-, 32- and 64-bit ordinals
- 8-, 16, 32- and 64-bit integers
- 32-, 64- and 80-bit real numbers

Non-Numeric:
- Bit
- Bit Field
- Triple-Word (96 bits)
- Quad-Word (128 bits)
- Mixed (33 bits)

Large Register Set

The programming environment of the 80960XA includes a large number of registers. In fact, 36 registers are available at any time. The availability of this many registers greatly reduces the number of memory accesses required to execute most programs, which leads to greater instruction processing speed.

There are two types of general-purpose registers: local and global. The 20 global registers consist of sixteen 32-bit registers (G0 through G15) and four 80-bit registers (FP0 through FP3). These registers perform the same function as the general-purpose registers provided in other popular microprocessors. The term global refers to the fact that these registers retain their contents across procedure calls.

The local registers, on the other hand, are procedure specific. For each procedure call, the 80960XA allocates 16 local registers (R0 through R15). Each local register is 32 bits wide. Any register can also be used for floating-point operations; the 80-bit floating-point registers are provided for extended precision.

Multiple Register Sets

To further increase the efficiency of the register set, multiple sets of local registers are stored on-chip. This cache holds up to four local register frames, which means that up to three procedure calls can be made without having to access the procedure stack resident in memory.

Table 2. Memory Addressing Modes

- 12-Bit Offset
- 32-Bit Offset
- Register-Indirect
- Register + 12-Bit Offset
- Register + 32-Bit Offset
- Register + (Index-Register × Scale-Factor)
- Register × Scale Factor + 32-Bit Displacement
- Register + (Index-Register × Scale-Factor) + 32-Bit Displacement
- Object Pointer + 12-Bit Offset
- Object Pointer + 32-Bit Offset
- Object Pointer + (Index-Register × Scale-Factor)
- Object Pointer × Scale Factor + 32-Bit Displacement

Scale-Factor is 1, 2, 4, 8 or 16

Although programs may have procedure calls nested many calls deep, a program typically oscillates back and forth between only two or three levels. As a result, with four stack frames in the cache, the probability of there being a free frame on the cache when a call is made is very high.

If there are four or more active procedures and a new procedure is called, the processor moves the oldest set of local registers in the register cache to a procedure stack in memory to make room for a new set of registers. Global register G15 is used by the processor as the frame pointer (FP) for the procedure stack.

Note that the global and floating-point registers are not exchanged on a procedure call, but retain their contents, making them available to all procedures for fast parameter passing. An illustration of the register cache is shown in Figure 5.

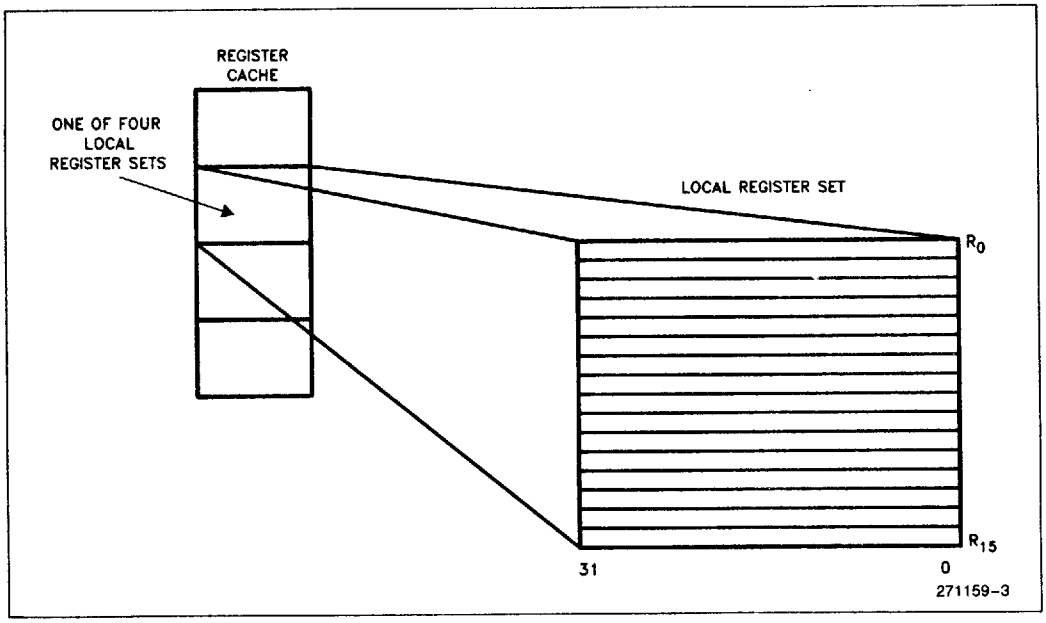


Figure 5. Multiple Register Sets Are Stored On-Chip

Instruction Cache

To further reduce memory accesses, the 80960XA includes a 512-byte on-chip instruction cache. The instruction cache is based on the concept of locality of reference; that is, most programs are not usually executed in a steady stream but consist of many branches and loops that lead to jumping back and forth within the same small section of code. Thus, by maintaining a block of instructions in a cache, the number of memory references required to read instructions into the processor can be greatly reduced.

To load the instruction cache, instructions are fetched in 16-byte blocks, so that up to four instructions can be fetched at one time. An efficient prefetch algorithm increases the probability that an instruction will already be in the cache when it is needed.

Code for small loops will often fit entirely within the cache, leading to a great increase in processing speed since further memory references might not be necessary until the program exits the loop. Similarly, when calling short procedures, the code for the calling procedure is likely to remain in the cache, so it will be there upon the procedure's return.

Register Scoreboarding

The instruction decoder has been optimized in several ways. One of these optimizations is the ability to do instruction overlapping by means of register scoreboarding.

Register scoreboarding occurs when a LOAD instruction is executed to move a variable from memory into a register. When the instruction is initiated, a scoreboard bit on the target register is set. When the register is actually loaded, the bit is reset. In between, any reference to the register contents is accompanied by a test of the scoreboard bit to insure that the load has completed before processing continues. Since the processor does not have to wait for the LOAD to be completed, it can go on to execute additional instructions placed in between the LOAD instruction and the instruction that uses the register contents, as shown in the following example:

```
LOAD R4, address 1
LOAD R5, address 2
Unrelated instruction
Unrelated instruction
ADD R4, R5, R6
```

In essence, the two unrelated instructions between the LOAD and ADD instructions are executed for free (i.e., take no apparent time to execute) because they are executed while the register is being loaded. Up to three LOAD instructions can be pending at one time with three corresponding scoreboard bits set. By exploiting this feature, system programmers and compilers have a useful tool for optimizing execution speed.

Memory Management, Object Addressing and Protection

The 80960XA is especially useful for multitasking applications that require software protection and a very large address space. To ensure a high level of integration, the memory management unit and the translation look-aside buffer are contained on-chip.

The 80960XA supports a conventional form of demand-paged linear memory in which each process is designated its own address space of up to 4 Gbytes. This address space is divided into 4 Kbyte pages. Studies have shown that a 4 Kbyte page is the optimum size for a broad range of applications.

The memory locations of the pages are kept within a page table. Each page table entry includes a 2-bit page rights field that specifies whether the page is a no-access, read-only, or read-write page. This field is interpreted differently depending on whether the current process is executing in user or supervisor mode, as shown below:

Rights	User	Supervisor
00	No Access	Read-Only
01	No Access	Read-Write
10	Read-Only	Read-Write
11	Read-Write	Read-Write

In addition, the 80960XA supports an object-oriented form of addressing and protection. With object-oriented programming, the 2^{32} byte linear address space can be expanded to a 2^{58} byte virtual address space. This virtual space is divided into typed, protected segments of memory known as **objects**. An object ranges in size from 64 bytes to 2^{32} bytes. Up to 2^{26} objects can be addressed, thus resulting in the 2^{58} byte virtual address space.

The objects are addressed with a 33-bit memory pointer called an **access descriptor** (see Figure 6). In a 33-bit memory word, the 33rd bit, called the **tag bit**, identifies the contents of the word as either data or an access descriptor. The creation and modification techniques of access descriptors are carefully controlled. Any unauthorized attempts to modify a data word into an access descriptor will turn the tag bit off, thus making it impossible to forge a pointer to memory.

Because the access descriptors are unforgeable, an added level of data security is introduced with this architecture. The protection provided by the tag bit isolates the individual address spaces and ensures that an error introduced in one software task cannot corrupt the reliability and security of other tasks.

Floating-Point Arithmetic

In the 80960XA, floating-point arithmetic has been made an integral part of the architecture. Having the floating-point unit integrated on-chip provides two advantages. First, it improves the performance of the chip for floating-point applications, since no additional bus overhead is associated with floating-point calculations, thereby leaving more time for other bus operations such as I/O. Second, the cost of using floating-point operations is reduced because a separate coprocessor chip is not required.

The 80960XA floating-point (real number) data types include single-precision (32-bit), double-precision (64-bit), and extended precision (80-bit) floating-point numbers. Any register may be used to execute floating-point operations.

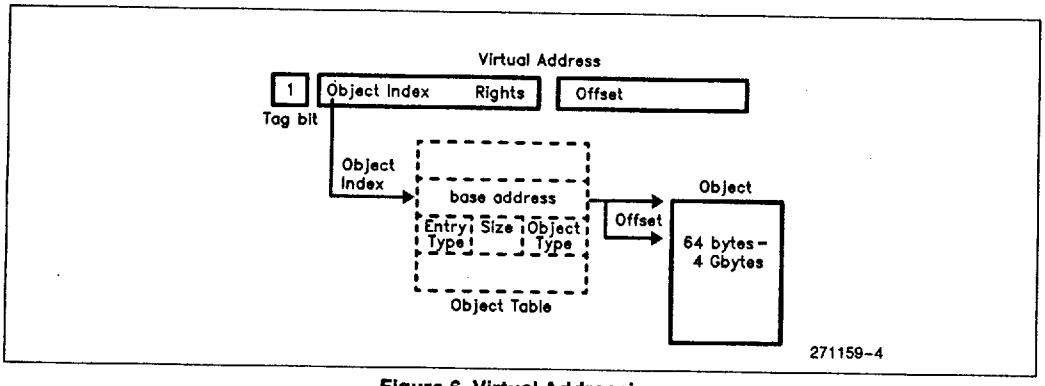


Figure 6. Virtual Addressing

The processor provides hardware support for both mandatory and recommended portions of IEEE Standard 754 for floating-point arithmetic, including all arithmetic, exponential, logarithmic, and other transcendental functions. Table 3 shows execution times for some representative instructions.

Table 3. Sample Floating-Point Execution Times (μs) at 25 MHz

	32-Bit	64-Bit
Add	0.4	0.5
Subtract	0.4	0.5
Multiply	0.7	1.3
Divide	1.3	2.9
Square Root	3.7	3.9
Arctangent	10.1	13.1
Exponent	11.3	12.5
Sine	15.2	16.6
Cosine	15.2	16.6

Multitasking Support

Multitasking programs commonly involve the monitoring and control of an external operation, such as the activities of a process controller or the movements of a machine tool. These programs generally consist of a number of processes that run independently of one another, but share a common database or pass data among themselves.

The 80960XA offers several hardware functions designed to support multitasking systems. One unique feature, called self-dispatching, allows a processor to switch itself automatically among scheduled tasks. When self-dispatching is used, all the operating system is required to do is place the task in the scheduling queue.

When the processor becomes available, it dispatches the task from the beginning of the queue and then executes it until it becomes blocked, interrupted, or until its time-slice expires. It then returns the task to the end of the queue (i.e., automatically reschedules it) and dispatches the next ready task. During these operations, no communication between the processor and the operating system is necessary until the running task is complete or an interrupt is issued.

Synchronization and Communication

The 80960XA also offers instructions to set up and test semaphores to ensure that concurrent tasks remain synchronized and no data inconsistency results. Special data structures, known as communication ports, provide the means for exchanging parameters and data structures. Transmission of information by means of communication ports is asynchronous and automatically buffered by the processor.

Communication between tasks by means of ports can be carried out independently of the operating system. Once the ports have been set up by the programmer, the processor handles the message passing automatically.

High Bandwidth Local Bus

An 80960XA CPU resides on a high-bandwidth address/data bus known as the local bus (L-Bus). The L-Bus provides a direct communication path between the processor and the memory and I/O subsystem interfaces. The processor uses the local bus to fetch instructions, manipulate memory, and respond to interrupts. Its features include:

- 32-bit multiplexed address/data path
- Four-word burst capability, which allows transfers from 1 to 16 bytes at a time
- High bandwidth reads and writes at 66.7 MBytes per second
- Special signal to indicate whether a memory transaction can be cached

Figure 7 identifies the groups of signals which constitute the L-Bus. Table 4 lists the function of the L-Bus and other processor-support signals, such as the interrupt lines.

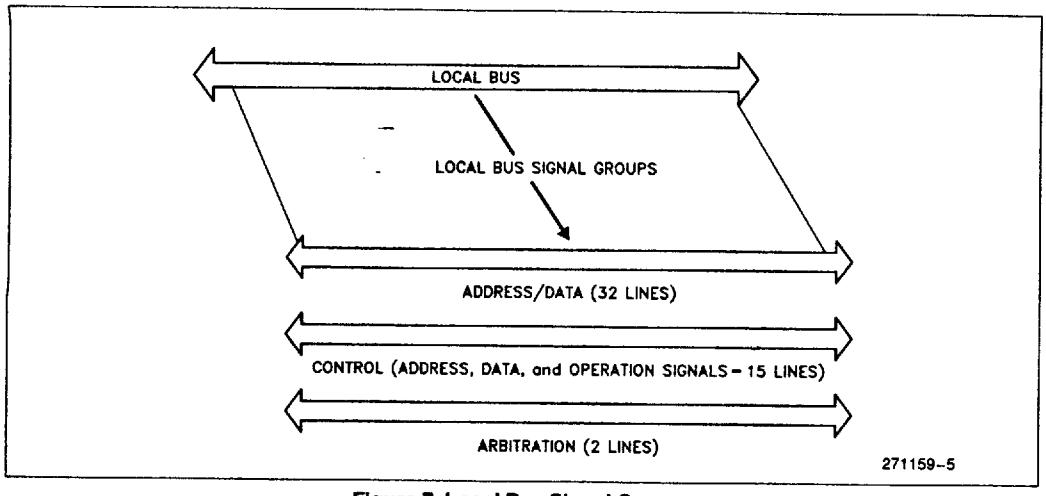


Figure 7. Local Bus Signal Groups

Multiple Processor Support

One means of increasing the processing power of a system is to run two or more processors in parallel. Since microprocessors are not generally designed to run in tandem with other processors, designing such a system is usually difficult and costly.

The 80960XA solves this problem by offering a number of functions to coordinate the actions of multiple processors. First, messages can be passed between processors to initiate actions such as flushing a cache, stopping or starting another processor, or preempting a task. The messages are passed on the bus and allow multiple processors to run together smoothly, with rare need to lock the bus or memory.

Second, a set of synchronization instructions help maintain the coherency of memory. These instructions permit several processors to modify memory at the same time without inserting inaccuracies or ambiguities into shared data structures.

The self-dispatching mechanism, in addition to being used in single-processor systems, provides the means to increase the performance of a system merely by adding processors. Each processor can either work on the same pool of tasks (sharing the same queue with other processors) or can be restricted to its own queue.

When processors perform system operations, they synchronize themselves by using atomic operations and sending special messages between each other. And changing the number of processors in a system never requires a software change. Software will execute correctly regardless of the number of processors in the system; systems with more processors simply execute faster.

Interrupt Handling

The 80960XA can be interrupted in one of two ways: by the activation of one of four interrupt pins or by sending a message on the processor's data bus.

The 80960XA is unusual in that it automatically handles interrupts on a priority basis and tracks pending interrupts through its on-chip interrupt controller. Two of the interrupt pins can be configured to provide M8259A handshaking for expansion beyond four interrupt lines.

An interrupt message is made up of a vector number and an interrupt priority. If the interrupt priority is greater than that of the currently running task, the processor accepts the interrupt and uses the vector as an index into the interrupt table. If the priority of the interrupt message is below that of the current task, the processor saves the information in a section of the interrupt table reserved for pending interrupts.

Debug Features

The 80960XA has built-in debug capabilities. There are two types of breakpoints and six different trace modes. The debug features are controlled by two

internal 32-bit registers, the Process-Controls Word and the Trace-Controls Word. By setting bits in these control words, a software debug monitor can closely control how the processor responds during program execution.

The 80960XA has both hardware and software breakpoints. It provides two hardware breakpoint registers on-chip which can be set by a special command to any value. When the instruction pointer matches the value in one of the breakpoint registers, the breakpoint will fire, and a breakpoint handling routine is called automatically.

The 80960XA also provides software breakpoints through the use of two instructions, MARK and FMARK. These instructions can be placed at any point in a program and will cause the processor to halt execution at that point and call the breakpoint handling routine. The breakpoint mechanism is easy to use and provides a powerful debugging tool.

Tracing is available for instructions (single-step execution), calls and returns, and branching. Each different type of trace may be enabled separately by a special debug instruction. In each case, the 80960XA executes the instruction first and then calls a trace handling routine (usually part of a software debug monitor). Further program execution is halted until the trace routine is completed. When the trace event handling routine is completed, instruction execution resumes at the next instruction. The 80960XA's tracing mechanisms, which are implemented completely in hardware, greatly simplify the task of testing and debugging software.

FAULT DETECTION

The 80960XA has an automatic mechanism to handle faults. There are fourteen fault types including trace, arithmetic, and floating-point faults. When the processor detects a fault, it automatically calls the appropriate fault handling routine and saves the current instruction pointer and necessary state information to make efficient recovery possible. The processor posts diagnostic information on the type of fault to a Fault Record. Like interrupt handling routines, fault handling routines are usually written to meet the needs of a specific application and are often included as part of the operating system or kernel.

For each of the fourteen fault types, there are numerous subtypes that provide specific information about a fault. For example, a floating-point fault may have its subtype set to an Overflow or Zero-Divide fault. The fault handler can use this specific information to respond correctly to the fault.

Interagent Communications (IAC)

In order to coordinate their actions, processors in a multiple processor system need a means for communicating with each other. The 80960XA does this through a mechanism known as Interagent Communication messages or IACs.

IAC messages cause a variety of actions including starting and stopping processors, flushing instruction caches and TLBs, and sending interrupts to other processors in the system. The upper 16 Mbytes of the processor's physical memory space is reserved for sending and receiving IAC messages.

BUILT-IN TESTABILITY

Upon reset, the 80960XA automatically conducts an exhaustive internal test of its major blocks of logic.

Then, before executing its first instruction, it does a zero check sum on the first eight words in memory to ensure that the system has been loaded correctly. If a problem is discovered at any point during the self-test, the 80960XA will assert its FAILURE pin and will not begin program execution. The self-test takes approximately 47,000 cycles to complete.

System manufacturers can use the 80960XA's self-test feature during incoming parts inspection. No special diagnostic programs need to be written, and the test is both thorough and fast. The self-test capability helps ensure that defective parts will be discovered before systems are shipped, and once in the field, the self-test makes it easier to distinguish between problems caused by processor failure and problems resulting from other causes.

COMPATIBILITY WITH OTHER i960® PROCESSOR PRODUCTS

Application programs written for the 80960K-Series and the 80960MC microprocessors can be run on the 80960XA without modification. The 80960K-Series and 80960MC instruction sets form a subset of the 80960XA's instructions, so binary compatibility is assured.

CHMOS

The 80960XA is fabricated using Intel's CHMOS IV (Complementary High Speed Metal Oxide Semiconductor) process. This advanced technology eliminates the frequency and reliability limitations of older CMOS processes and opens a new era in microprocessor performance. It combines the high performance capabilities of Intel's industry-leading HMOS III technology with the high density and low power characteristics of CMOS. The 80960XA is available at 16, 20 and 25 MHz versions.

Table 4a. 80960XA Pin Description: L-Bus Signals

Symbol	Type	Name and Function
CLK2	I	**SYSTEM CLOCK** provides the fundamental timing for 80960XA systems. It is divided by two inside the 80960XA to generate the internal processor clock. CLK2 is shown in Figure 11.
LAD_{31} $-LAD_0$	I/O T.S.	**LOCAL ADDRESS/DATA BUS** carries 32-bit physical addresses and data to and from memory. During an address (T_a) cycle, bits 2–31 contain a physical word address (bits 0–1 indicate SIZE; see below). During a data (T_d) cycle, bits 0–31 contain read or write data. The LAD lines are active HIGH and float to a high impedance state when not active. **SIZE,** which is comprised of bits 0–1 of the LAD lines during a T_a cycle, specifies the size of a transfer in words for a burst transaction. $\underline{LAD_1}$ $\underline{LAD_0}$ 0 0 1 Word 0 1 2 Words 1 0 3 Words 1 1 4 Words
$\overline{ALE}$	O T.S.	**ADDRESS-LATCH ENABLE** indicates the transfer of a physical address. ALE is asserted during a T_a cycle and deasserted before the beginning of the T_d state. It is active LOW and floats to a high impedance state when the processor is idle or is at the end of any bus access.
$\overline{ADS}$	O O.D.	**ADDRESS STATUS** indicates an address state. ADS is asserted every T_a state and deasserted during the the following T_d state. For a burst transaction, $\overline{ADS}$ is asserted again every T_d state where $\overline{READY}$ was asserted in the previous cycle.
W/$\overline{R}$	O O.D.	**WRITE/READ** specifies, during a T_a cycle, whether the operation is a write or read. It is latched on-chip and remains valid during T_d and T_w states.
DT/$\overline{R}$	O O.D.	**DATA TRANSMIT/RECEIVE** indicates the direction of data transfer to and from the L-Bus. It is low during T_a, T_w and T_d cycles for a read or interrupt acknowledgement; it is high during T_a, T_w and T_d cycles for a write. DT/$\overline{R}$ never changes state when $\overline{DEN}$ is asserted (see Timing Diagrams).
$\overline{DEN}$	O O.D.	**DATA ENABLE** is asserted during T_d and T_w cycles and indicates transfer of data on the LAD bus lines.
$\overline{READY}$	I	**READY** indicates that data on LAD lines can be sampled or removed. If $\overline{READY}$ is not asserted during a T_d cycle, the T_d cycle is extended to the next cycle by inserting wait states (T_w), and $\overline{ADS}$ is not asserted in the next cycle.
$\overline{LOCK}$	I/O O.D.	**BUS LOCK** prevents other bus masters from gaining control of the L-Bus following the current cycle (if they would assert $\overline{LOCK}$ to do so). $\overline{LOCK}$ is used by the processor or any bus agent when it performs indivisible Read/Modify/Write (RMW) operations. For a read that is designated as a RMW-read, $\overline{LOCK}$ is examined. if asserted, the processor waits until it is not asserted; if not asserted, the processor asserts $\overline{LOCK}$ during the T_a cycle and leaves it asserted. A write that is designated as an RMW-write deasserts $\overline{LOCK}$ in the T_a cycle.

I/O = Input/Output, O = Output, I = Input, O.D. = Open-Drain, T.S. = three state
$T_a = T_{Address}$, $T_d = T_{Data}$, $T_W = T_{Wait}$, $T_r = T_{Recovery}$, $T_i = T_{Idle}$, $T_h = T_{Hold}$

Table 4a. 80960XA Pin Description: L-Bus Signals (Continued)

Symbol	Type	Name and Function
$\overline{BE_3}$-$\overline{BE_0}$	O O.D.	**BYTE ENABLE LINES** specify which data bytes (up to four) on the bus take part in the current bus cycle. $\overline{BE_3}$ corresponds to LAD_{31}-LAD_{24} and $\overline{BE_0}$ corresponds to LAD_7-LAD_0. The byte enables are provided in advance of data. The byte enables asserted during T_a specify the bytes of the first data word. The byte enables asserted during T_d specify the bytes of the next data word (if any), that is, the word to be transmitted following the next assertion of $\overline{READY}$. The byte enables during the T_d cycles preceding the last assertion of $\overline{READY}$ are undefined. The byte enables are latched on-chip and remain constant from one T_d cycle to the next when $\overline{READY}$ is not asserted. For reads, the byte enables specify the byte(s) that the processor will actually use. 80960XA's will assert only adjacent byte enables (e.g., asserting just $\overline{BE_0}$ and $\overline{BE_2}$ is not permitted), and are required to assert at least one byte enable. Accesses must also be naturally aligned (e.g., asserting $\overline{BE_1}$ and $\overline{BE_2}$ is not allowed even though they are adjacent). To produce address bits A_0 and A_1 externally, they can be decoded from the byte enables.
HOLD (HLDAR)	I	**HOLD** indicates a request from a secondary bus master to acquire the bus. If the processor is initialized as the primary bus master this input will be interpreted as HOLD. When the processor receives HOLD and grants another master control of the bus, it floats its three-state bus lines, asserts HOLD ACKNOWLEDGE, and enters the T_h state. When HOLD is deasserted, the processor will deassert HOLD ACKNOWLEDGE and go to either the T_i or T_a state. **HOLD ACKNOWLEDGE RECEIVED** indicates that the processor has acquired the bus. If the processor is initialized as the secondary bus master this input is interpreted as HLDAR. HOLD timing is shown in Figure 13.
HLDA (HOLDR)	O T.S.	**HOLD ACKNOWLEDGE** relinquishes control of the bus to another bus master. If the processor is initialized as the primary bus master this output will be interpreted as HLDA. When HOLD is deasserted, the processor will deassert HLDA and go to either the T_i or T_a state. **HOLD REQUEST** indicates a request to acquire the bus. If the processor is initialized as the secondary bus master this output will be interpreted as HOLDR. HOLD timing is shown in Figure 13.
CACHE/TAG	I/O T.S.	**CACHE** is an output signal that indicates if an access is cacheable during a T_a cycle. The CACHE signal floats to a high impedance state when the processor is idle. TAG is an input/output signal that during T_d and T_w cycles identifies the contents of a 32-bit word as either data (TAG = 0) or an access descriptor (TAG = 1).

I/O = Input/Output, O = Output, I = Input, O.D. = Open-Drain, T.S. = three state
$T_a = T_{Address}$, $T_d = T_{Data}$, $T_W = T_{Wait}$, $T_r = T_{Recovery}$, $T_i = T_{Idle}$, $T_h = T_{Hold}$

Table 4b. 80960XA Pin Description: Module Support Signals

Symbol	Type	Name and Function
$\overline{BADAC}$	I	**BAD ACCESS,** if asserted in the cycle following the one in which the last $\overline{READY}$ of a transaction is asserted, indicates that an unrecoverable error has occurred on the current bus transaction, or that a synchronous load/store instruction has not been acknowledged. **STARTUP:** During system reset, the $\overline{BADAC}$ signal is interpreted differently. If the signal is high, it indicates that this processor will perform system initialization. If it is low, another processor in the system will perform system initialization instead.
RESET	I	**RESET** clears the internal logic of the processor and causes it to re-initialize. During RESET assertion, the input pins are ignored (except for $\overline{BADAC}$ and $\overline{IAC}/\overline{INT_0}$), the tri-state output pins are placed in a high impedance state, and other output pins are placed in their non-asserted state. RESET must be asserted for at least 41 CLK2 cycles for a predictable RESET. The HIGH to LOW transition of RESET should occur after the rising edge of both CLK2 and the external bus CLK, and before the next rising edge of CLK2. RESET timing is shown in Figure 12.
$\overline{FAILURE}$	O O.D.	**INITIALIZATION FAILURE** indicates that the processor has failed to initialize correctly. After RESET is deasserted and before the first bus transaction begins, $\overline{FAILURE}$ is asserted while the processor performs a self-test. If the self-test completes successfully, then $\overline{FAILURE}$ is deasserted. Next, the processor performs a zero checksum on the first eight words of memory. If it fails, $\overline{FAILURE}$ is asserted for a second time and remains asserted; if it passes, system initialization continues and $\overline{FAILURE}$ remains deasserted.
N.C.	N/A	**NOT CONNECTED** indicates pins should not be connected. Never connect any pin marked N.C.
$\overline{IAC}$ $(\overline{INT_0})$	I	**INTERAGENT COMMUNICATION REQUEST/INTERRUPT 0** indicates either that there is a pending IAC message for the processor or an interrupt. The bus interrupt control register determines in which way the signal should be interpreted. To signal an interrupt or IAC request in a synchronous system, this pin (as well as the other interrupt pins) must be enabled by being deasserted for at least one bus cycle and then asserted for at least one additional bus cycle; in an asynchronous system, the pin must remain deasserted for at least two bus cycles and then be asserted for at least two more bus cycles. **LOCAL PROCESSOR NUMBER:** This signal is interpreted differently during system reset. If the signal is at a high voltage level, it indicates that this processor is a primary bus master (Local Processor Number = 0); if it is at a low voltage level, it indicates that this processor is a secondary bus master (Local Processor Number = 1).
INT_1	I	**INTERRUPT 1,** like $\overline{INT_0}$, provides direct interrupt signaling.
INT_2 (INTR)	I	**INTERRUPT 2/INTERRUPT REQUEST:** The bus control registers determines how this pin is interpreted. If INT_2, it has the same interpretation as the $\overline{INT_0}$ and INT1 pins. If INTR, it is used to receive an interrupt request from an external interrupt controller.
$\overline{INT_3}$ $(\overline{INTA})$	I/O O.D.	**INTERRUPT 3/INTERRUPT ACKNOWLEDGE:** The bus interrupt control register determines how this pin is interpreted. If $\overline{INT_3}$, it has the same interpretation as the $\overline{INT_0}$, INT_1, and INT_2 pins. If $\overline{INTA}$, it is used as an output to control interrupt-acknowledge bus transactions. The $\overline{INTA}$ output is latched on-chip and remains valid during T_d cycles; as an output, it is open-drain.

I/O = Input/Output, O = Output, I = Input, O.D. = Open-Drain, T.S. = three state
$T_a = T_{Address}$, $T_d = T_{Data}$, $T_W = T_{Wait}$, $T_r = T_{Recovery}$, $T_i = T_{Idle}$, $T_h = T_{Hold}$

ELECTRICAL SPECIFICATIONS

Power and Grounding

The 80960XA is implemented in CHMOS IV technology and has modest power requirements. Its high clock frequency and numerous output buffers (address/data, control, error and arbitration signals) can cause power surges as multiple output buffers drive new signal levels simultaneously. For clean on-chip power distribution at high frequency, 12 V_{CC} and 13 V_{SS} pins separately feed functional units of the 80960XA.

Power and ground connections must be made to all power and ground pins of the 80960XA. On the circuit board, all V_{CC} pins must be strapped closely together, preferably on a power plane. Likewise, all V_{SS} pins should be strapped together, preferably on a ground plane.

Power Decoupling Recommendations

Liberal decoupling capacitance should be placed near the 80960XA. The processor can cause transient power surges when driving the L-Bus, particularly when it is connected to a large capacitive load.

Low inductance capacitors and interconnects are recommended for best high frequency electrical performance. Inductance can be reduced by shortening the board traces between the processor and decoupling capacitors as much as possible.

Connection Recommendations

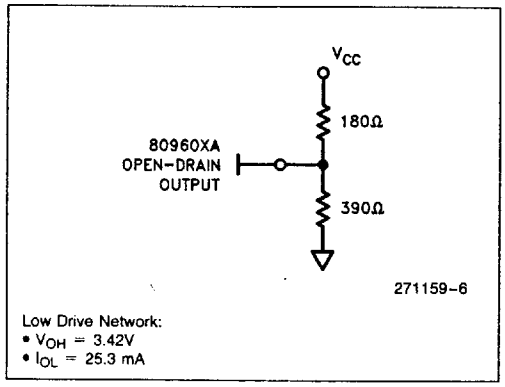
For reliable operation, always connect unused inputs to an appropriate signal level. In particular, if one or more interrupt lines are not used, they should be pulled up or down to their respective deasserted states. No inputs should ever be left floating.

All open-drain outputs require a pullup device. While in some cases a simple pullup resistor will be adequate, we recommend a network of pullup and pulldown resistors biased to a valid V_{IH} ($\geq$ 3.4V) and terminated in the characteristic impedance of the circuit board. Figure 8 shows our recommendations for the resistor values for both a low and high current drive network, which assumes that the circuit board has a characteristic impedance of 100Ω. The advantage of terminating the output signals in this fashion is that it limits signal swing and reduces AC power consumption.

Characteristic Curves

Figure 9 shows the typical supply current requirements over the operating temperature range of the processor at supply voltage (V_{CC}) of 5V. Figure 10 shows the typical power supply current (I_{CC}) required by the 80960XA at various operating frequencies when measured at three input voltage (V_{CC}) levels.

Figure 11 shows the typical capacitive derating curve for the 80960XA measured from 1.5V on the system clock (CLK) to 0.8V on the falling edge and 2.0V on the rising edge of the L-Bus address/data (LAD) signals.



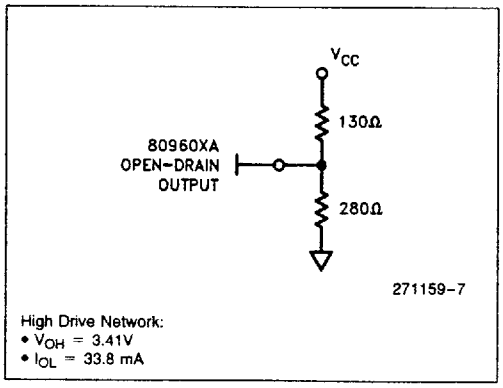


Figure 8. Connection Recommendations for Low and High Current Drive Networks

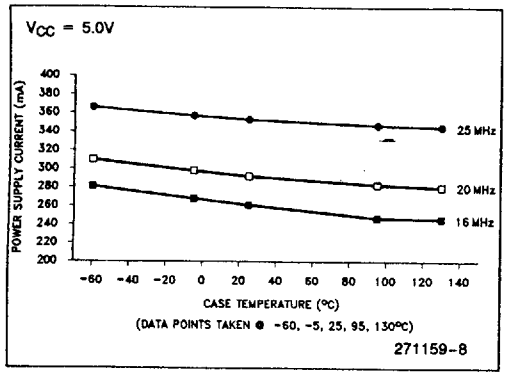


Figure 9. Typical Supply Current (I_{CC})

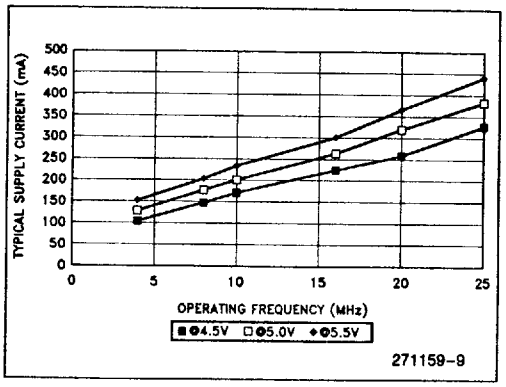


Figure 10. Typical Current vs Frequency

10

Test Load Circuit

Figure 12 illustrates the load circuit used to test the 80960XA's tristate pins, and Figure 13 shows the load circuit used to test the open drain outputs. The open drain test uses an active load circuit in the form of a matched diode bridge. Since the open-drain outputs sink current, only the I_{OL} legs of the bridge are necessary and the I_{OH} legs are not used. When the 80960XA driver under test is turned off, the output pin is pulled up to V_{REF} (i.e., V_{OH}). Diode D_1 is turned off and the I_{OL} current source flows through diode D_2.

When the 80960XA open-drain driver under test is on, diode D_1 is also on, and the voltage on the pin being tested drops to V_{OL}. Diode D_2 turns off and I_{OL} flows through diode D_1.

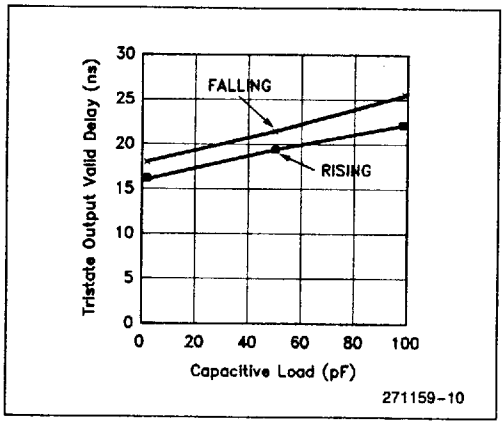


Figure 11. Capacitive Derating Curve

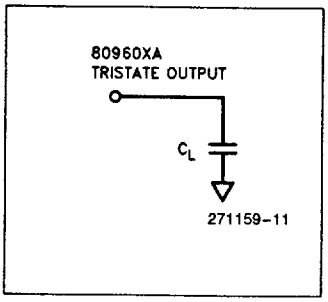


Figure 12. Test Load Circuit for Tri-State Output Pins

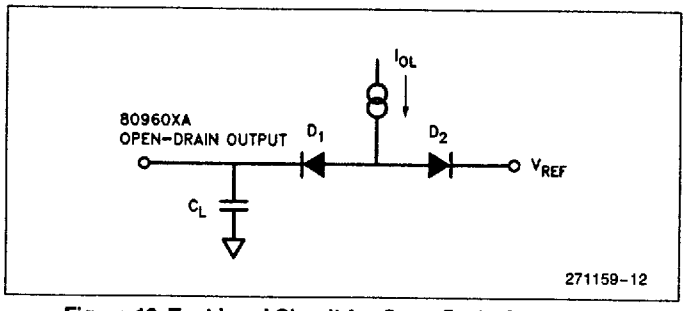


Figure 13. Test Load Circuit for Open-Drain Output Pins

ABSOLUTE MAXIMUM RATINGS*

Case Temperature under Bias[6] ..... −55°C to +125°C
Storage Temperature ..... −65°C to +150°C
Voltage on Any Pin ..... −0.5V to V_{CC} + 0.5V
Power Dissipation ..... 2.6W (25 MHz)

NOTICE: This data sheet contains information on products in the sampling and initial production phases of development. The specifications are subject to change without notice. Verify with your local Intel Sales office that you have the latest data sheet before finalizing a design.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

DC CHARACTERISTICS

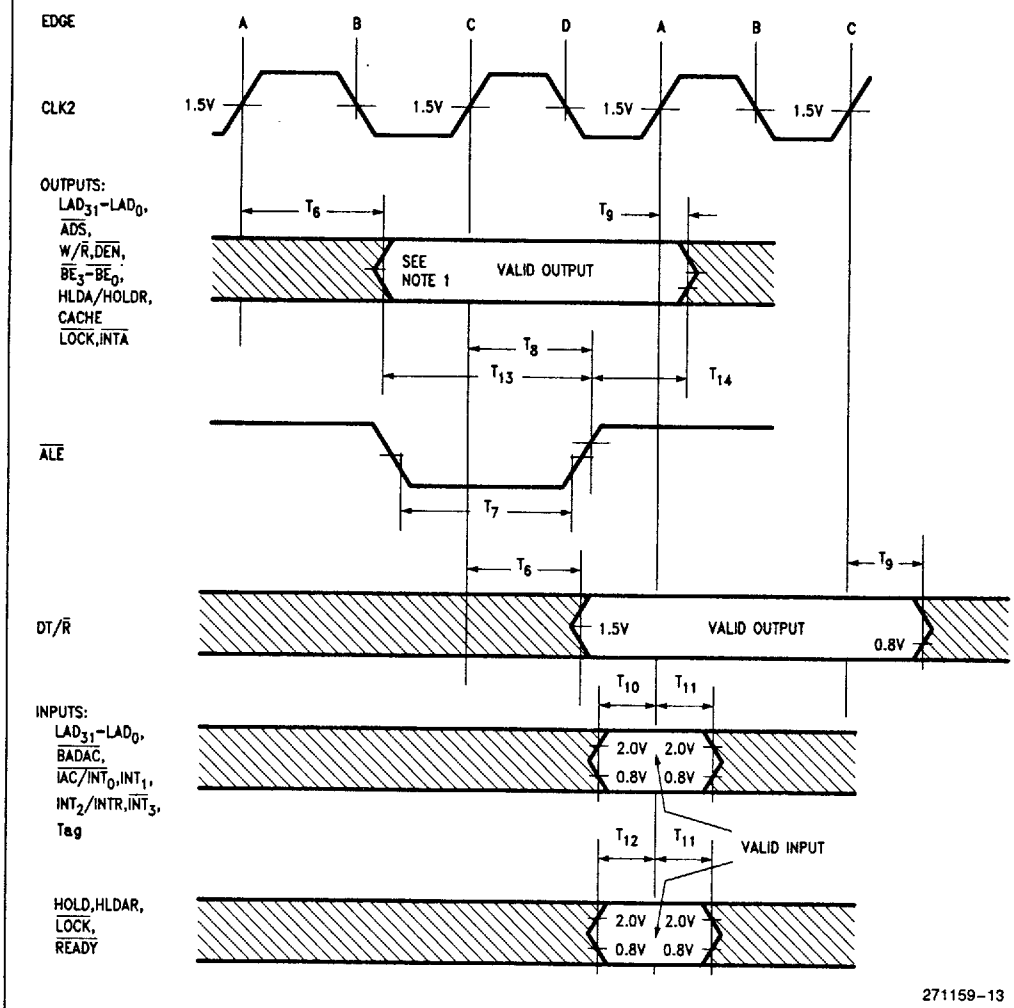
80960XA: T_{CASE}[6] = −55°C to +125°C, V_{CC} = 5V ± 5%

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	−0.3	+0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC} + 0.3	V	
V_{CL}	CLK2 Input Low Voltage	−0.3	+0.8	V	
V_{CH}	CLK2 Input High Voltage	0.55 V_{CC}	V_{CC} + 0.3	V	
V_{OL}	Output Low Voltage		0.45	V	(1, 5)
V_{OH}	Output High Voltage	2.4		V	(2, 4)
I_{CC}	Power Supply Current: 16 MHz 20 MHz 25 MHz		 375 420 480	 mA mA mA	
I_{LI}	Input Leakage Current		±15	μA	$0 \le V_O \le V_{CC}$
I_{LO}	Output Leakage Current		±15	μA	$0.45 \le V_O \le V_{CC}$
C_{IN}	Input Capacitance		10	pF	f_C = 1 MHz[3]
C_O	I/O or Output Capacitance		12	pF	f_C = 1 MHz[3]
C_{CLK}	Clock Capacitance		10	pF	f_C = 1 MHz[3]
θ_{JA}	Thermal Resistance (Junction-to-Ambient) Pin Grid Array Ceramic Quad Flatpack		 21 29	 °C/W °C/W	
θ_{JC}	Thermal Resistance (Junction-to-Case) Pin Grid Array Ceramic Quad Flatpack		 4 8	 °C/W °C/W	

NOTES:
1. For three-state outputs, this parameter is measured at:
   Address/Data ..... 4.0 mA
   Controls ..... 5.0 mA
2. This parameter is measured at:
   Address/Data ..... −1.0 mA
   Controls ..... −0.9 mA
   ALE ..... −5.0 mA
3. Input, output, and clock capacitance are not tested.
4. Not measured on open-drain outputs.
5. For open-drain outputs ..... 25 mA
6. Case temperatures are "instant on".

AC SPECIFICATIONS

This section describes the AC specifications for the 80960XA pins. All input and output timings are specified relative to the 1.5V level of the rising edge of CLK2, and refer to the time at which the signal reaches (for output delay and input setup) or leaves (for hold time) the TTL levels of LOW (0.8V) or HIGH (2.0V). All AC testing should be done with input voltages of 0.4V and 2.4V, except for the clock (CLK2), which should be tested with input voltages of 0.45V and 0.55 V_{CC}.



NOTE 1:
For three-state pins, T_6 and T_9 are measured at 1.5V.
For Open-Drain pins, T_6 is measured at 1.5V, T_9 at 0.8V.

Figure 14. Drive Levels and Timing Relationships for 80960XA Signals

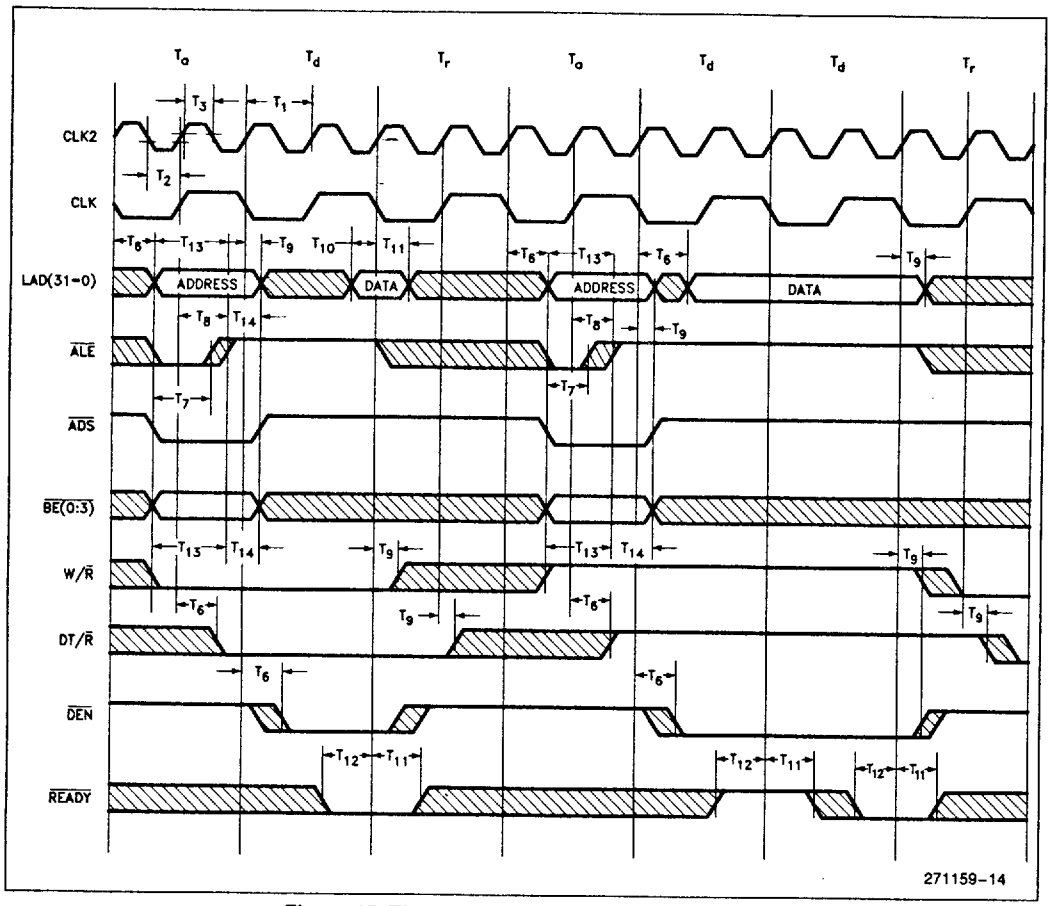


Figure 15. Timing Relationship of L-Bus Signals

AC Specification Tables

80960XA AC Characteristics (16 MHz)
T_{CASE}[3] = −55°C to +125°C, V_{CC} = 5V ±5%

Symbol	Parameter	Min	Max	Units	Test Conditions
T_1	Processor Clock Period (CLK2)	31.25	125	ns	V_{IN} = 1.5V
T_2	Processor Clock Low Time (CLK2)	8		ns	V_{IL} = 10% Point = 1.2V
T_3	Processor Clock High Time (CLK2)	8		ns	V_{IH} = 90% Point = 0.1V + 0.5 V_{CC}
T_4	Processor Clock Fall Time (CLK2)		10	ns	V_{IN} = 90% Point to 10% Point
T_5	Processor Clock Rise Time (CLK2)		10	ns	V_{IN} = 10% Point to 90% Point
T_6	Output Valid Delay	2	25	ns	C_L = 100 pF (LAD) C_L = 75 pF (Controls)
T_{6H}	HOLDA Output Valid Delay	4	31	ns	C_L = 75 pF
T_7	ALE Width	15		ns	C_L = 75 pF
T_8	ALE Invalid Delay	0	20	ns	C_L = 75 pF[2]
T_9	Output Float Delay	2	20	ns	C_L = 100 pF (LAD) C_L = 75 pF (Controls)[2]
T_{9H}	HOLDA Output Float Delay	4	20	ns	C_L = 75 pF
T_{10}	Input Setup 1	3		ns	(Note 1)
T_{11}	Input Hold	5		ns	(Note 1)
T_{11H}	HOLD Input Hold	4		ns	
T_{12}	Input Setup 2	8		ns	
T_{13}	Setup to ALE Inactive	10		ns	C_L = 100 pF (LAD) C_L = 75 pF (Controls)
T_{14}	Hold after ALE Inactive	8		ns	C_L = 100 pF (LAD) C_L = 75 pF (Controls)
T_{15}	Reset Hold	3		ns	
T_{16}	Reset Setup	5		ns	
T_{17}	Reset Width	1281		ns	41 CLK2 Periods Minimum

NOTES:
1. IAC/INT_0, INT_1, INT_2/INTR, INT_3 can be asynchronous.
2. A float condition occurs when the maximum output current becomes less than I_{LO}. Float delay is not tested, but should be no longer than the valid delay.
3. Case temperatures are "instant on".

AC Specification Tables (Continued)

80960XA AC Characteristics (20 MHz)
T_{CASE}[3] = −55°C to +125°C, V_{CC} = 5V ±5%

Symbol	Parameter	Min	Max	Units	Test Conditions
T_1	Processor Clock Period (CLK2)	25	125	ns	V_{IN} = 1.5V
T_2	Processor Clock Low Time (CLK2)	6		ns	V_{IL} = 10% Point = 1.2V
T_3	Processor Clock High Time (CLK2)	6		ns	V_{IH} = 90% Point = 0.1V + 0.5 V_{CC}
T_4	Processor Clock Fall Time (CLK2)		10	ns	V_{IN} = 90% Point to 10% Point
T_5	Processor Clock Rise Time (CLK2)		10	ns	V_{IN} = 10% Point to 90% Point
T_6	Output Valid Delay	2	20	ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)
T_{6H}	HOLDA Output Valid Delay	4	26	ns	C_L = 50 pF
T_7	ALE Width	12		ns	C_L = 50 pF
T_8	ALE Invalid Delay	0	20	ns	C_L = 50 pF[2]
T_9	Output Float Delay	2	20	ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)[2]
T_{9H}	HOLDA Output Float Delay	4	20	ns	C_L = 50 pF
T_{10}	Input Setup 1	3		ns	(Note 1)
T_{11}	Input Hold	5		ns	(Note 1)
T_{11H}	HOLD Input Hold	4		ns	
T_{12}	Input Setup 2	7		ns	
T_{13}	Setup to ALE Inactive	10		ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)
T_{14}	Hold after ALE Inactive	8		ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)
T_{15}	Reset Hold	3		ns	
T_{16}	Reset Setup	5		ns	
T_{17}	Reset Width	1025		ns	41 CLK2 Periods Minimum

NOTES:
1. IAC/INT_0, INT_1, INT_2/INTR, INT_3 can be asynchronous.
2. A float condition occurs when the maximum output current becomes less than I_{LO}. Float delay is not tested, but should be no longer than the valid delay.
3. Case temperatures are "instant on".

AC Specification Tables (Continued)

80960XA AC Characteristics (25 MHz)

T_{CASE}[3] = −55°C to +125°C, V_{CC} = 5V ±5%

Symbol	Parameter	Min	Max	Units	Test Conditions
T_1	Processor Clock Period (CLK2)	20	125	ns	V_{IN} = 1.5V
T_2	Processor Clock Low Time (CLK2)	5		ns	V_{IL} = 10% Point = 1.2V
T_3	Processor Clock High Time (CLK2)	5		ns	V_{IH} = 90% Point = 0.1V + 0.5 V_{CC}
T_4	Processor Clock Fall Time (CLK2)		10	ns	V_{IN} = 90% Point to 10% Point
T_5	Processor Clock Rise Time (CLK2)		10	ns	V_{IN} = 10% Point to 90% Point
T_6	Output Valid Delay	2	19	ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)
T_{6H}	HOLDA Output Valid Delay	4	24	ns	C_L = 50 pF
T_7	ALE Width	12		ns	C_L = 50 pF
T_8	ALE Invalid Delay	0	20	ns	C_L = 50 pF[2]
T_9	Output Float Delay	2	19	ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)[2]
T_{9H}	HOLDA Output Float Delay	4	20	ns	C_L = 50 pF
T_{10}	Input Setup 1	3		ns	(Note 1)
T_{11}	Input Hold	5		ns	(Note 1)
T_{11H}	HOLD Input Hold	4		ns	
T_{12}	Input Setup 2	7		ns	
T_{13}	Setup to ALE Inactive	8		ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)
T_{14}	Hold after ALE Inactive	8		ns	C_L = 60 pF (LAD) C_L = 50 pF (Controls)
T_{15}	Reset Hold	3		ns	
T_{16}	Reset Setup	5		ns	
T_{17}	Reset Width	820		ns	41 CLK2 Periods Minimum

NOTES:
1. IAC/INT_0, INT_1, INT_2/INTR, INT_3 can be asynchronous.
2. A float condition occurs when the maximum output current becomes less than I_{LO}. Float delay is not tested, but should be no longer than the valid delay.
3. Case temperatures are "instant on".

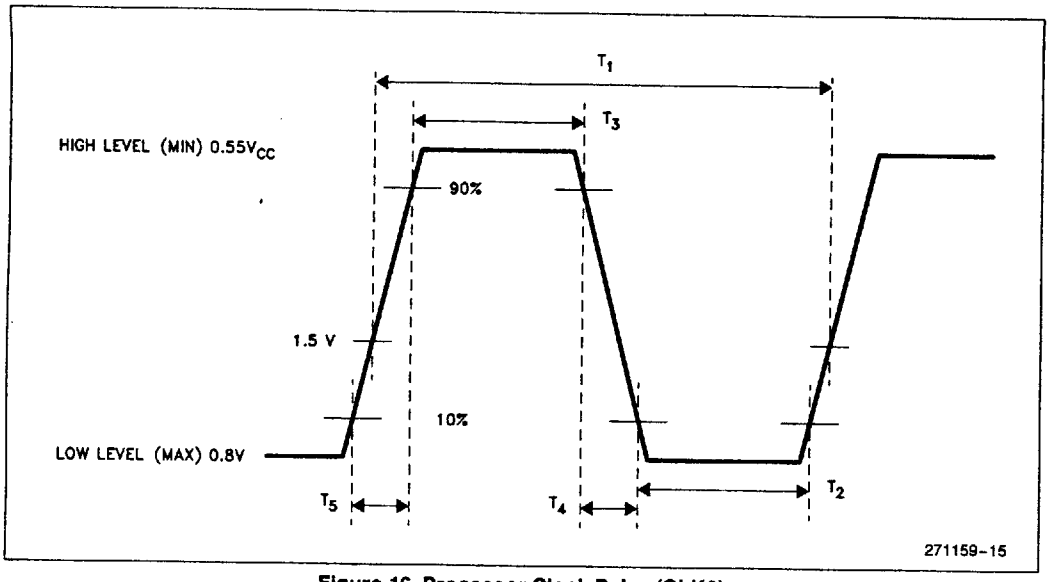


Figure 16. Processor Clock Pulse (CLK2)

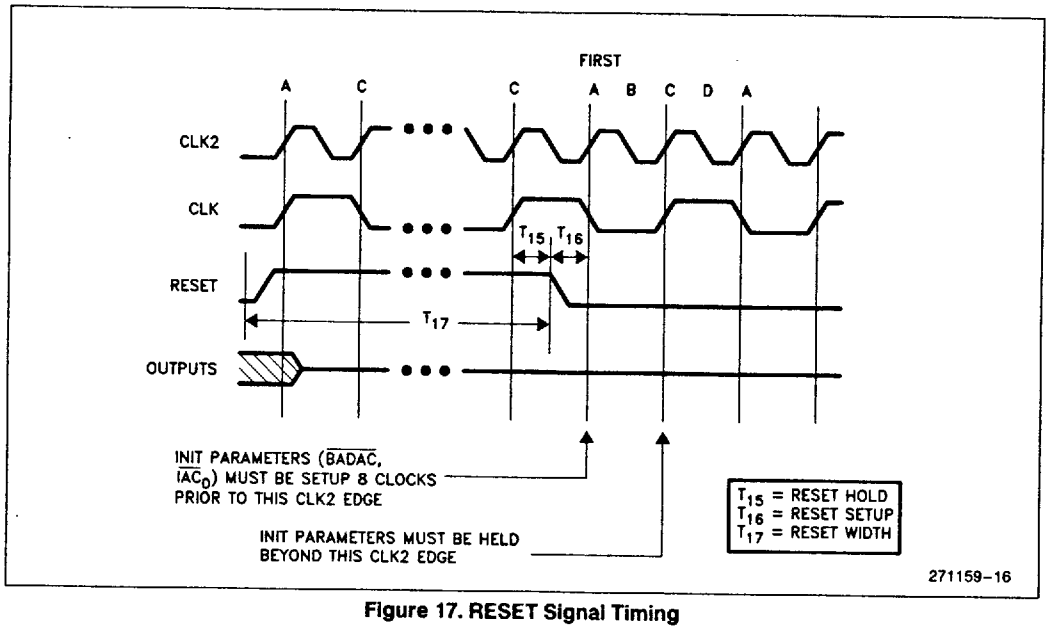


Figure 17. RESET Signal Timing

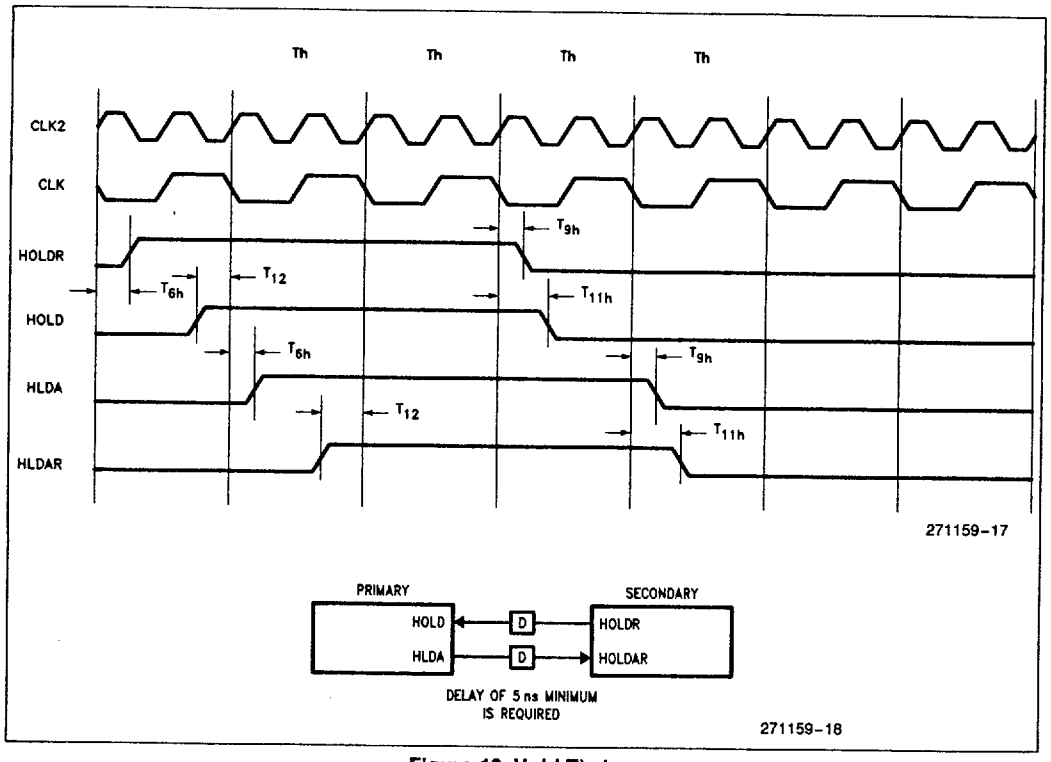


Figure 18. Hold Timing

Design Considerations

Input hold times can be disregarded by the designer whenever the input is removed because a subsequent output from the processor is deasserted (e.g., $\overline{DEN}$ becomes deasserted).

In other words, whenever the processor generates an output that indicates a transition into a subsequent state, the processor must have sampled any inputs for the previous state.

Similarly, whenever the processor generates an output that indicates a transition into a subsequent state, any outputs that are specified to be three stated in this new state are guaranteed to be three stated.

MECHANICAL DATA

Pin Assignment

The 80960XA is packaged in a 132-lead ceramic pin grid array and a 164-lead ceramic quad flatpack. The 80960XA pin grid array pinout as viewed from the substrate side of the component is shown in Figure 19 and from the pin side in Figure 20. The 80960XA ceramic quad flatpack pinout as viewed from the top of the package is shown in Figure 21.

V_{CC} and GND connections must be made to multiple V_{CC} and GND pins. Each V_{CC} and GND pin must be connected to the appropriate voltage or ground and externally strapped close to the package. Preferably, the circuit board should include power and ground planes for power distribution. Tables 5, 6, 7 and 8 list the function of each pin.

NOTE:
Pins identified as N.C., "No Connect," should never be connected under any circumstances.

Package Dimensions and Mounting

Pins in the pin grid array package are arranged 0.100 inch (2.54mm) center-to-center, in a 14 by 14 matrix, three rows around. (See Figure 22.)

A wide variety of available sockets allow low-insertion or zero-insertion force mountings, and a choice of terminals such as soldertail, surface mount, or wire wrap. Several applicable sockets are shown in Figure 23.

Package Thermal Specification

The 80960XA is specified for operation when its case temperature is within the range of −55°C to +125°C. The PGA case temperature should be measured at the center of the top surface opposite the pins as shown in Figure 24. The ceramic quad flatpack case temperature should be measured at the center of the lid on the top surface of the package.

WAVEFORMS

Figures 25 through 31 show the waveforms for various transactions on the 80960XA's local bus.

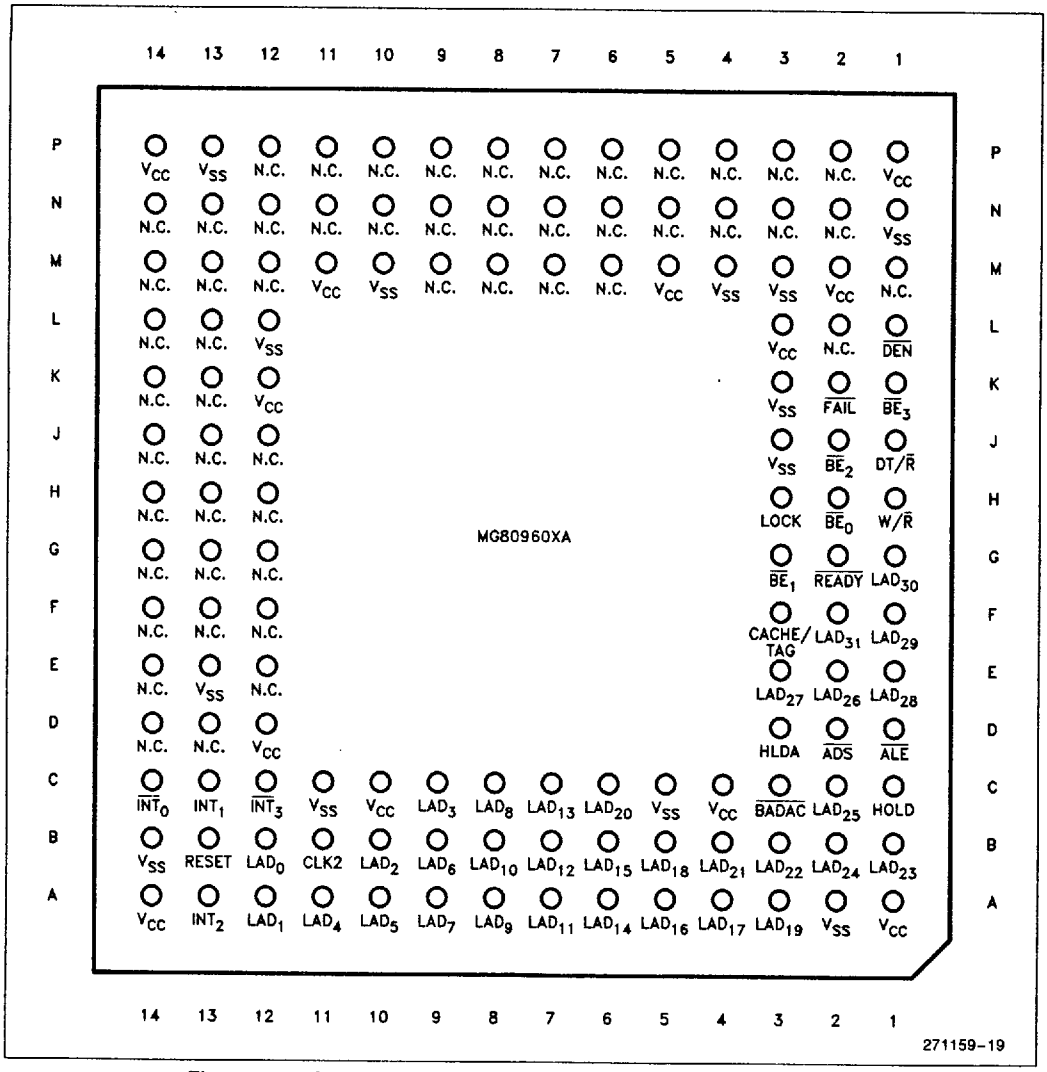


Figure 19. MG80960XA Pinout—View from Top (Pins Facing Down)

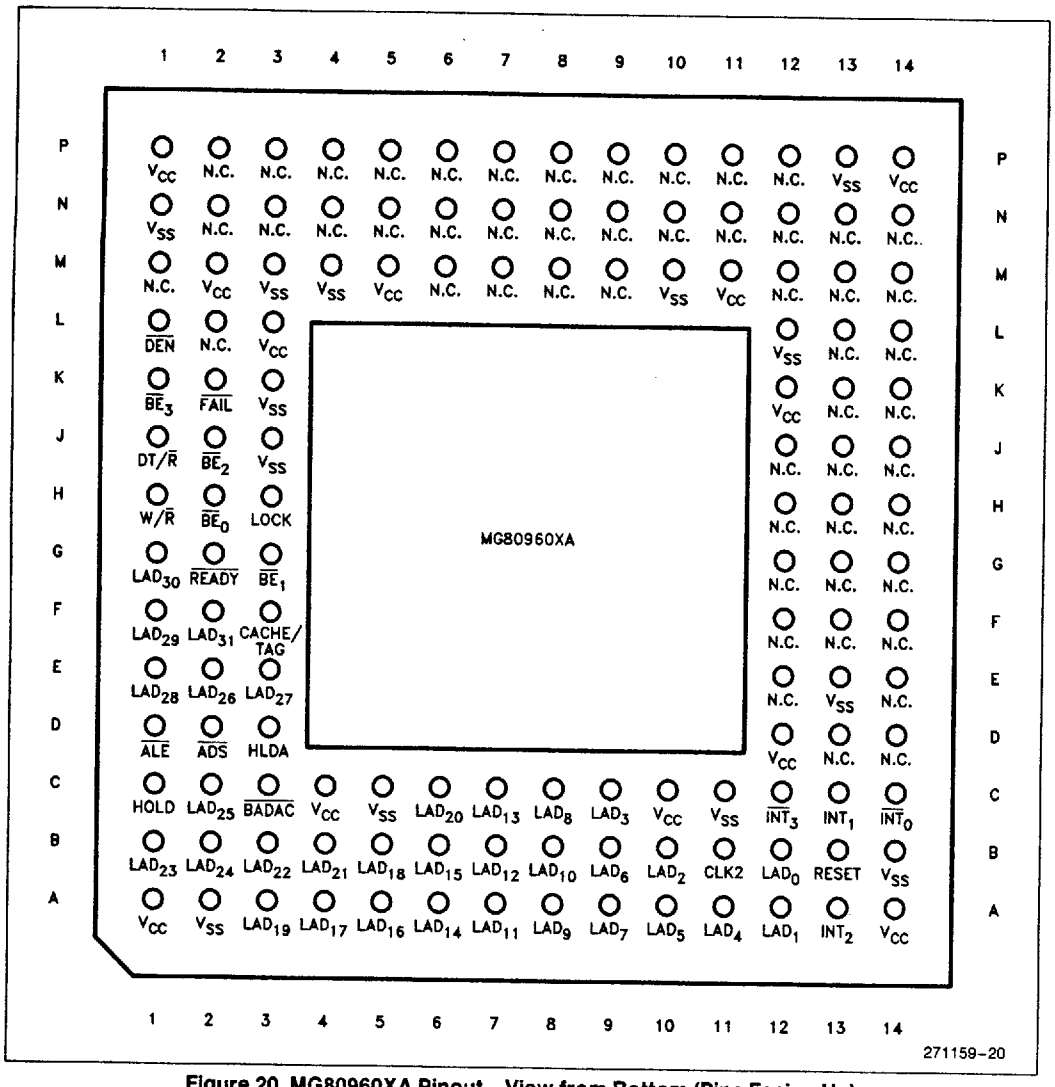
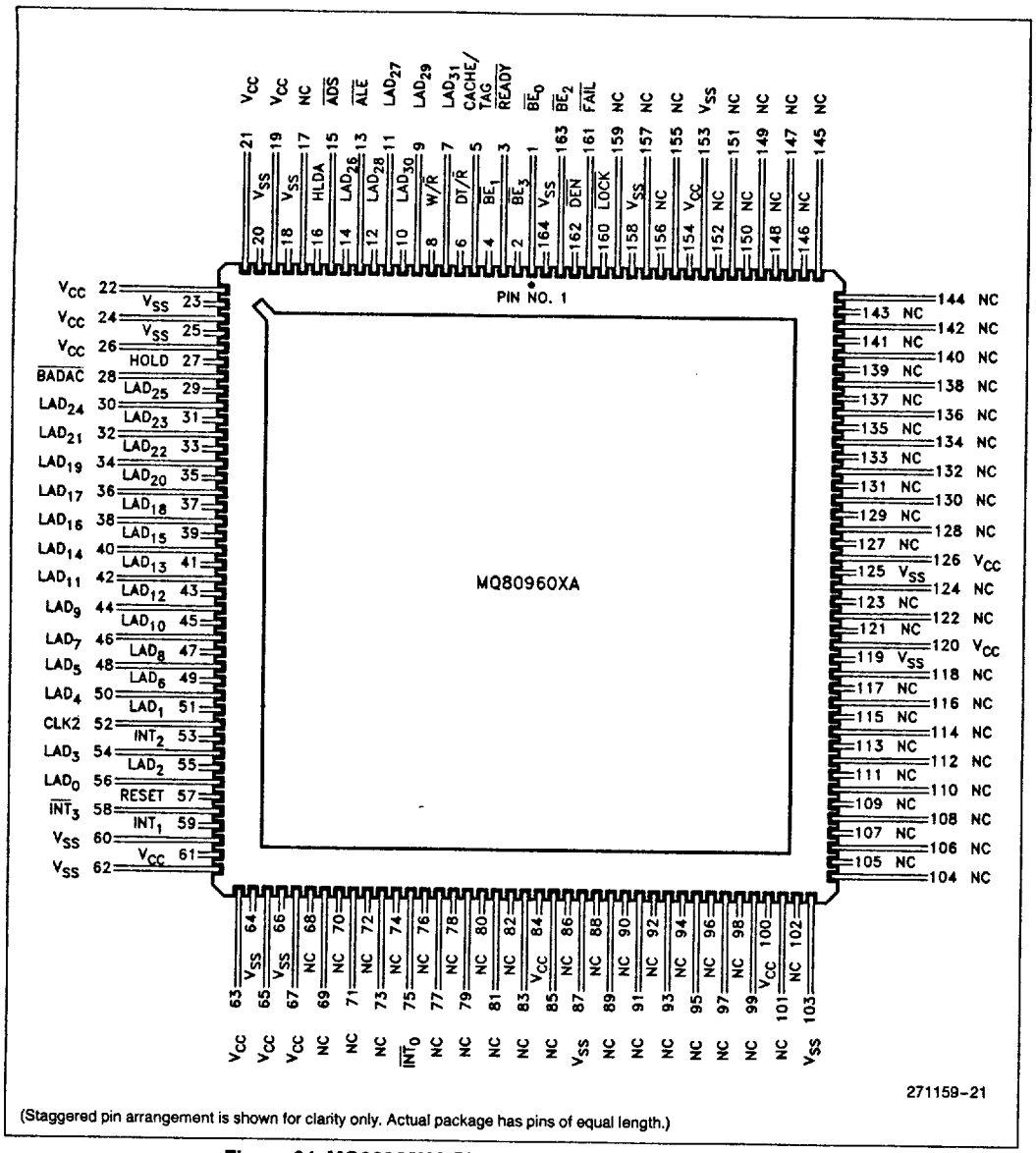


Figure 20. MG80960XA Pinout—View from Bottom (Pins Facing Up)

10



(Staggered pin arrangement is shown for clarity only. Actual package has pins of equal length.)

Figure 21. MQ80960XA Pinout—View from Top of Package

Table 5. MG80960XA (PGA) Pinout—In Pin Order

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
A1	V_{CC}	C6	LAD_{20}	H1	$W/\overline{R}$	M10	V_{SS}
A2	V_{SS}	C7	LAD_{13}	H2	$\overline{BE_0}$	M11	V_{CC}
A3	LAD_{19}	C8	LAD_8	H3	$\overline{LOCK}$	M12	N.C.
A4	LAD_{17}	C9	LAD_3	H12	N.C	M13	N.C.
A5	LAD_{16}	C10	V_{CC}	H13	N.C.	M14	N.C.
A6	LAD_{14}	C11	V_{SS}	H14	N.C.	N1	V_{SS}
A7	LAD_{11}	C12	$\overline{INT_3}/\overline{INTA}$	J1	$DT/\overline{R}$	N2	N.C.
A8	LAD_9	C13	INT_1	J2	$\overline{BE_2}$	N3	N.C.
A9	LAD_7	C14	$\overline{IAC}/\overline{INT_0}$	J3	V_{SS}	N4	N.C.
A10	LAD_5	D1	$\overline{ALE}$	J12	N.C	N5	N.C.
A11	LAD_4	D2	$\overline{ADS}$	J13	N.C.	N6	N.C.
A12	LAD_1	D3	HLDA/HLDR	J14	N.C.	N7	N.C.
A13	INT_2/INTR	D12	V_{CC}	K1	$\overline{BE_3}$	N8	N.C.
A14	V_{CC}	D13	N.C.	K2	$\overline{FAILURE}$	N9	N.C.
B1	LAD_{23}	D14	N.C.	K3	V_{SS}	N10	N.C.
B2	LAD_{24}	E1	LAD_{28}	K12	V_{CC}	N11	N.C.
B3	LAD_{22}	E2	LAD_{26}	K13	N.C.	N12	N.C.
B4	LAD_{21}	E3	LAD_{27}	K14	N.C.	N13	N.C.
B5	LAD_{18}	E12	N.C.	L1	$\overline{DEN}$	N14	N.C.
B6	LAD_{15}	E13	V_{SS}	L2	N.C.	P1	V_{CC}
B7	LAD_{12}	E14	N.C.	L3	V_{CC}	P2	N.C.
B8	LAD_{10}	F1	LAD_{29}	L12	V_{SS}	P3	N.C.
B9	LAD_6	F2	LAD_{31}	L13	N.C.	P4	N.C.
B10	LAD_2	F3	CACHE/TAG	L14	N.C.	P5	N.C.
B11	CLK2	F12	N.C.	M1	N.C.	P6	N.C.
B12	LAD_0	F13	N.C.	M2	V_{CC}	P7	N.C.
B13	RESET	F14	N.C.	M3	V_{SS}	P8	N.C.
B14	V_{SS}	G1	LAD_{30}	M4	V_{SS}	P9	N.C.
C1	HOLD/HLDAR	G2	$\overline{READY}$	M5	V_{CC}	P10	N.C.
C2	LAD_{25}	G3	$\overline{BE_1}$	M6	N.C.	P11	N.C.
C3	$\overline{BADAC}$	G12	N.C.	M7	N.C.	P12	N.C.
C4	V_{CC}	G13	N.C.	M8	N.C.	P13	V_{SS}
C5	V_{SS}	G14	N.C.	M9	N.C.	P14	V_{CC}

NOTE:
Pins identified as N.C. ("No Connect") should never be connected under any circumstances.

Table 6. MG80960XA (PGA) Pinout—In Signal Order

Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
$\overline{ADS}$	D2	LAD_{15}	B6	N.C.	J14	N.C.	P9
$\overline{ALE}$	D1	LAD_{16}	A5	N.C.	K13	N.C.	P10
$\overline{BADAC}$	C3	LAD_{17}	A4	N.C.	K14	N.C.	P11
$\overline{BE_0}$	H2	LAD_{18}	B5	N.C.	L13	N.C.	P12
$\overline{BE_1}$	G3	LAD_{19}	A3	N.C.	L14	N.C.	L2
$\overline{BE_2}$	J2	LAD_{20}	C6	N.C.	M1	$\overline{READY}$	G2
$\overline{BE_3}$	K1	LAD_{21}	B4	N.C.	M6	RESET	B13
CACHE/TAG	F3	LAD_{22}	B3	N.C.	M7	V_{CC}	A1
CLK2	B11	LAD_{23}	B1	N.C.	M8	V_{CC}	A14
$\overline{DEN}$	L1	LAD_{24}	B2	N.C.	M9	V_{CC}	C4
DT/$\overline{R}$	J1	LAD_{25}	C2	N.C.	M12	V_{CC}	C10
$\overline{FAILURE}$	K2	LAD_{26}	E2	N.C.	M13	V_{CC}	D12
HLDA/HOLDR	D3	LAD_{27}	E3	N.C.	M14	V_{CC}	K12
HOLD/HLDAR	C1	LAD_{28}	E1	N.C.	N2	V_{CC}	L3
$\overline{IAC}/\overline{INT_0}$	C14	LAD_{29}	F1	N.C.	N3	V_{CC}	M2
INT_1	C13	LAD_{30}	G1	N.C.	N4	V_{CC}	M5
INT_2/INTR	A13	LAD_{31}	F2	N.C.	N5	V_{CC}	M11
$\overline{INT_3}/\overline{INTA}$	C12	$\overline{LOCK}$	H3	N.C.	N6	V_{CC}	P1
LAD_0	B12	N.C.	D13	N.C.	N7	V_{CC}	P14
LAD_1	A12	N.C.	D14	N.C.	N8	V_{SS}	A2
LAD_2	B10	N.C.	E12	N.C.	N9	V_{SS}	B14
LAD_3	C9	N.C.	E14	N.C.	N10	V_{SS}	C5
LAD_4	A11	N.C.	F12	N.C.	N11	V_{SS}	C11
LAD_5	A10	N.C.	F13	N.C.	N12	V_{SS}	E13
LAD_6	B9	N.C.	F14	N.C.	N13	V_{SS}	J3
LAD_7	A9	N.C.	G12	N.C.	N14	V_{SS}	K3
LAD_8	C8	N.C.	G13	N.C.	P2	V_{SS}	L12
LAD_9	A8	N.C.	G14	N.C.	P3	V_{SS}	M3
LAD_{10}	B8	N.C.	H12	N.C.	P4	V_{SS}	M4
LAD_{11}	A7	N.C.	H13	N.C.	P5	V_{SS}	M10
LAD_{12}	B7	N.C.	H14	N.C.	P6	V_{SS}	N1
LAD_{13}	C7	N.C.	J12	N.C.	P7	V_{SS}	P13
LAD_{14}	A6	N.C.	J13	N.C.	P8	W/$\overline{R}$	H1

NOTE:
Pins identified as N.C. ("No Connect") should never be connected under any circumstances.

Table 7. MQ80960XA (CQP) Pinout—In Pin Order

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	$\overline{BE_0}$	42	LAD_{11}	83	N.C.	124	N.C.
2	$\overline{BE_3}$	43	LAD_{12}	84	V_{CC}	125	V_{SS}
3	$\overline{READY}$	44	LAD_9	85	N.C.	126	V_{CC}
4	$\overline{BE_1}$	45	LAD_{10}	86	N.C.	127	N.C.
5	CACHE/TAG	46	LAD_7	87	V_{SS}	128	N.C.
6	DT/$\overline{R}$	47	LAD_8	88	N.C.	129	N.C.
7	LAD_{31}	48	LAD_5	89	N.C.	130	N.C.
8	W/$\overline{R}$	49	LAD_6	90	N.C.	131	N.C.
9	LAD_{29}	50	LAD_4	91	N.C.	132	N.C.
10	LAD_{30}	51	LAD_1	92	N.C.	133	N.C.
11	LAD_{27}	52	CLK_2	93	N.C.	134	N.C.
12	LAD_{28}	53	INT_2	94	N.C.	135	N.C.
13	$\overline{ALE}$	54	LAD_3	95	N.C.	136	N.C.
14	LAD_{26}	55	LAD_2	96	N.C.	137	N.C.
15	$\overline{ADS}$	56	LAD_0	97	N.C.	138	N.C.
16	HLDA	57	RESET	98	N.C.	139	N.C.
17	N.C.	58	$\overline{INT_3}$	99	N.C.	140	N.C.
18	V_{SS}	59	INT_1	100	V_{CC}	141	N.C.
19	V_{CC}	60	V_{SS}	101	N.C.	142	N.C.
20	V_{SS}	61	V_{CC}	102	N.C.	143	N.C.
21	V_{CC}	62	V_{SS}	103	V_{SS}	144	N.C.
22	V_{CC}	63	V_{CC}	104	N.C.	145	N.C.
23	V_{SS}	64	V_{SS}	105	N.C.	146	N.C.
24	V_{CC}	65	V_{CC}	106	N.C.	147	N.C.
25	V_{SS}	66	V_{SS}	107	N.C.	148	N.C.
26	V_{CC}	67	V_{CC}	108	N.C.	149	N.C.
27	HOLD	68	N.C.	109	N.C.	150	N.C.
28	$\overline{BADAC}$	69	N.C.	110	N.C.	151	N.C.
29	LAD_{25}	70	N.C.	111	N.C.	152	N.C.
30	LAD_{24}	71	N.C.	112	N.C.	153	V_{SS}
31	LAD_{23}	72	N.C.	113	N.C.	154	V_{CC}
32	LAD_{21}	73	N.C.	114	N.C.	155	N.C.
33	LAD_{22}	74	N.C.	115	N.C.	156	N.C.
34	LAD_{19}	75	$\overline{INT_0}$	116	N.C.	157	N.C.
35	LAD_{20}	76	N.C.	117	N.C.	158	V_{SS}
36	LAD_{17}	77	N.C.	118	N.C.	159	N.C.
37	LAD_{18}	78	N.C.	119	V_{SS}	160	$\overline{LOCK}$
38	LAD_{16}	79	N.C.	120	V_{CC}	161	$\overline{FAIL}$
39	LAD_{15}	80	N.C.	121	N.C.	162	$\overline{DEN}$
40	LAD_{14}	81	N.C.	122	N.C.	163	$\overline{BE_2}$
41	LAD_{13}	82	N.C.	123	N.C.	164	V_{SS}

NOTE:
Pins identified as N.C. ("No Connect") should never be connected under any circumstances.

Table 8. MQ80960XA (CQP) Pinout—In Signal Order

Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
$\overline{ADS}$	15	LAD_{23}	31	N.C.	102	N.C.	148
$\overline{ALE}$	13	LAD_{24}	30	N.C.	104	N.C.	149
$\overline{BADAC}$	28	LAD_{25}	29	N.C.	105	N.C.	150
$\overline{BE_0}$	1	LAD_{26}	14	N.C.	106	N.C.	151
$\overline{BE_1}$	4	LAD_{27}	11	N.C.	107	N.C.	152
$\overline{BE_2}$	163	LAD_{28}	12	N.C.	108	N.C.	155
$\overline{BE_3}$	2	LAD_{29}	9	N.C.	109	N.C.	156
CACHE/TAG	5	LAD_{30}	10	N.C.	110	N.C.	157
CLK2	52	LAD_{31}	7	N.C.	111	N.C.	159
$\overline{DEN}$	162	$\overline{LOCK}$	160	N.C.	112	$\overline{READY}$	3
DT/$\overline{R}$	6	N.C.	17	N.C.	113	RESET	57
$\overline{FAILURE}$	161	N.C.	68	N.C.	114	V_{CC}	19
HLDA/HOLDR	16	N.C.	69	N.C.	115	V_{CC}	21
HOLD/HLDAR	27	N.C.	70	N.C.	116	V_{CC}	22
$\overline{IAC}/\overline{INT_0}$	75	N.C.	71	N.C.	117	V_{CC}	24
INT_1	59	N.C.	72	N.C.	118	V_{CC}	26
INT_2/INTR	53	N.C.	73	N.C.	121	V_{CC}	61
$\overline{INT_3}/\overline{INTA}$	58	N.C.	74	N.C.	122	V_{CC}	63
LAD_0	56	N.C.	76	N.C.	123	V_{CC}	65
LAD_1	51	N.C.	77	N.C.	124	V_{CC}	67
LAD_2	55	N.C.	78	N.C.	127	V_{CC}	84
LAD_3	54	N.C.	79	N.C.	128	V_{CC}	100
LAD_4	50	N.C.	80	N.C.	129	V_{CC}	120
LAD_5	48	N.C.	81	N.C.	130	V_{CC}	126
LAD_6	49	N.C.	82	N.C.	131	V_{CC}	154
LAD_7	46	N.C.	83	N.C.	132	V_{SS}	18
LAD_8	47	N.C.	85	N.C.	133	V_{SS}	20
LAD_9	44	N.C.	86	N.C.	134	V_{SS}	23
LAD_{10}	45	N.C.	88	N.C.	135	V_{SS}	25
LAD_{11}	42	N.C.	89	N.C.	136	V_{SS}	60
LAD_{12}	43	N.C.	90	N.C.	137	V_{SS}	62
LAD_{13}	41	N.C.	91	N.C.	138	V_{SS}	64
LAD_{14}	40	N.C.	92	N.C.	139	V_{SS}	66
LAD_{15}	39	N.C.	93	N.C.	140	V_{SS}	87
LAD_{16}	38	N.C.	94	N.C.	141	V_{SS}	103
LAD_{17}	36	N.C.	95	N.C.	142	V_{SS}	119
LAD_{18}	37	N.C.	96	N.C.	143	V_{SS}	125
LAD_{19}	34	N.C.	97	N.C.	144	V_{SS}	153
LAD_{20}	35	N.C.	98	N.C.	145	V_{SS}	158
LAD_{21}	32	N.C.	99	N.C.	146	V_{SS}	164
LAD_{22}	33	N.C.	101	N.C.	147	W/$\overline{R}$	8

NOTE:
Pins identified as N.C. ("No Connect") should never be connected under any circumstances.

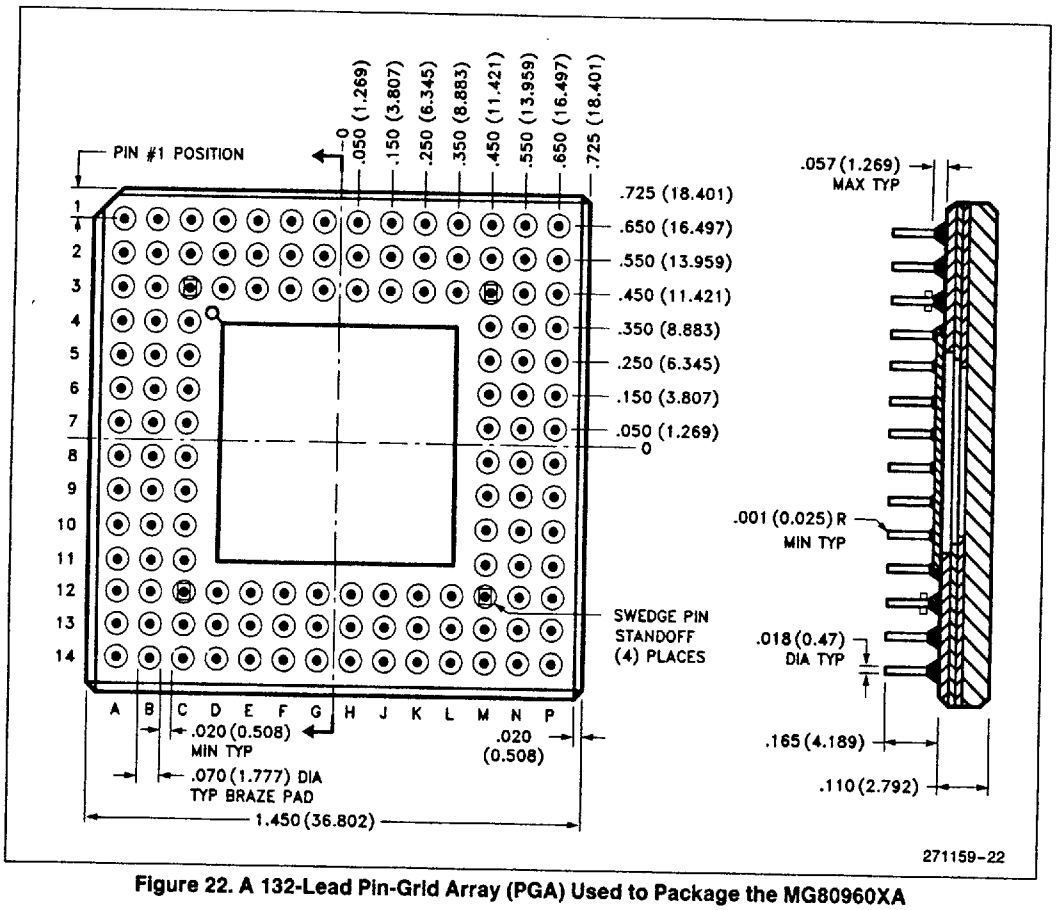


Figure 22. A 132-Lead Pin-Grid Array (PGA) Used to Package the MG80960XA

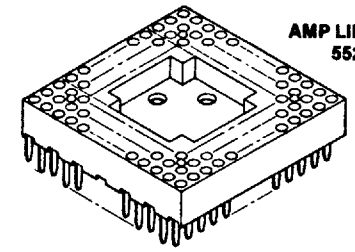
10

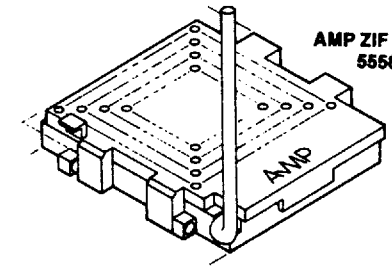
- Low insertion force (LIF) soldertail 55274-1
- Amp tests indicate 50% reduction in insertion force compared to machined sockets

Other socket options

- Zero insertion force (ZIF) soldertail 55583-1
- Zero insertion force (ZIF) Burn-in version 55573-2

Amp Incorporated
(Harrisburg, PA 17105 U.S.A
Phone 717-564-0100)





271159-23

Cam handle locks in low profile position when MG80960XA is installed (handle UP for open and DOWN for closed positions).

Courtesy Amp Incorporated

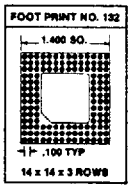
Peel-A-Way* Mylar and Kapton Socket Terminal Carriers

- Low insertion force surface mount CS132-37TG
- Low insertion force soldertail CS132-01TG
- Low insertion force wire-wrap CS132-02TG (two-level) CS132-03TG (three-level)
- Low insertion force press-fit CS132-05TG

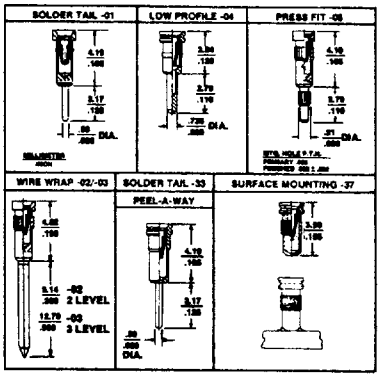
Advanced Interconnections
(5 Division Street
Warwick, RI 02818 U.S.A.
Phone 401-885-0485)

Peel-A-Way Carrier No. 132:
Kapton Carrier is KS132
Mylar Carrier is MS132

Molded Plastic Body KS132 is shown below:



271159-24



271159-25

Courtesy Advanced Interconnections
(Peel-A-Way Terminal Carriers
U.S. Patent No. 4442938)

*Peel-A-Way is a trademark of Advanced Interconnections.

Figure 23. Several Socket Options for Mounting the MG80960XA

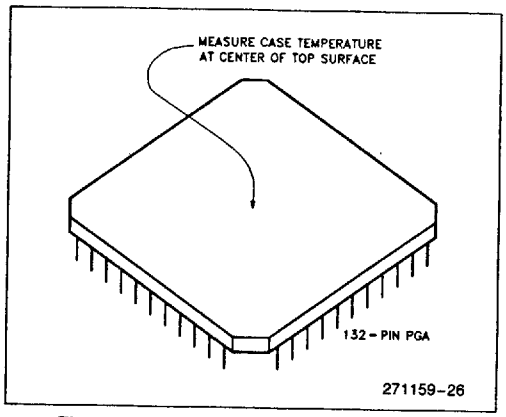


Figure 24. Measuring MG80960XA PGA Case Temperature (T_C)

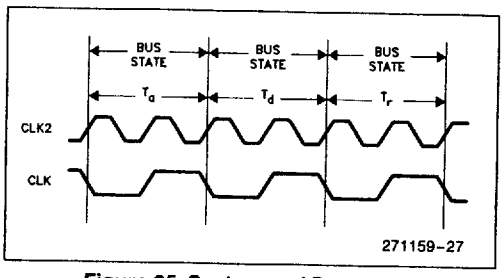


Figure 25. System and Processor Clock Relationship

10

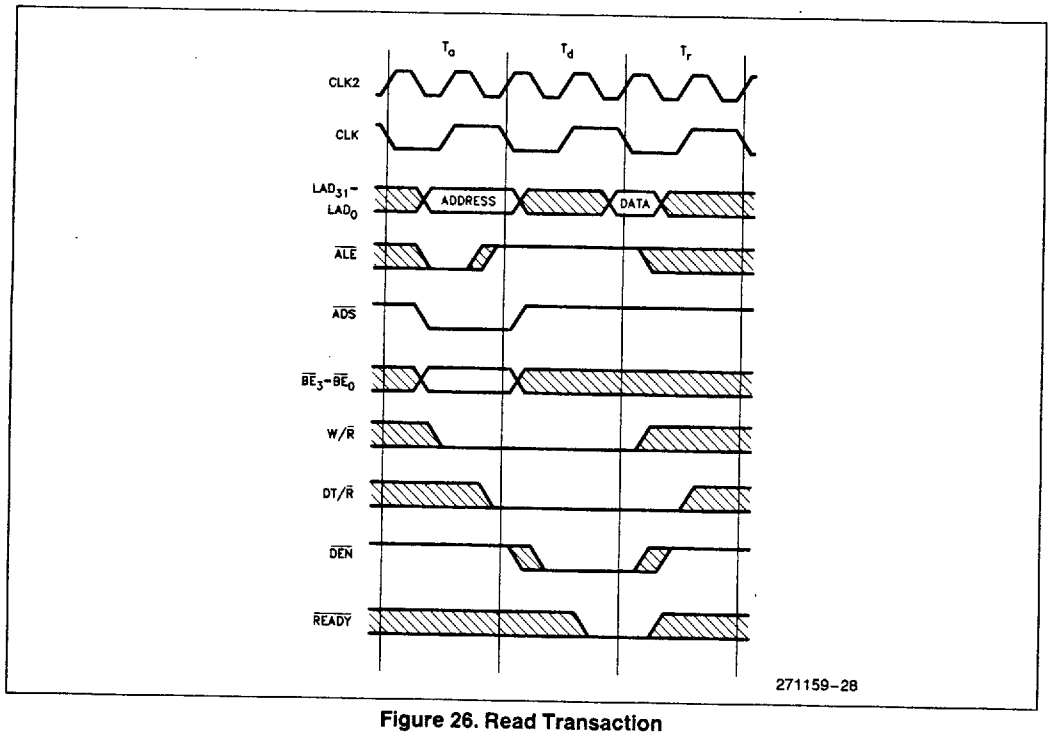


Figure 26. Read Transaction

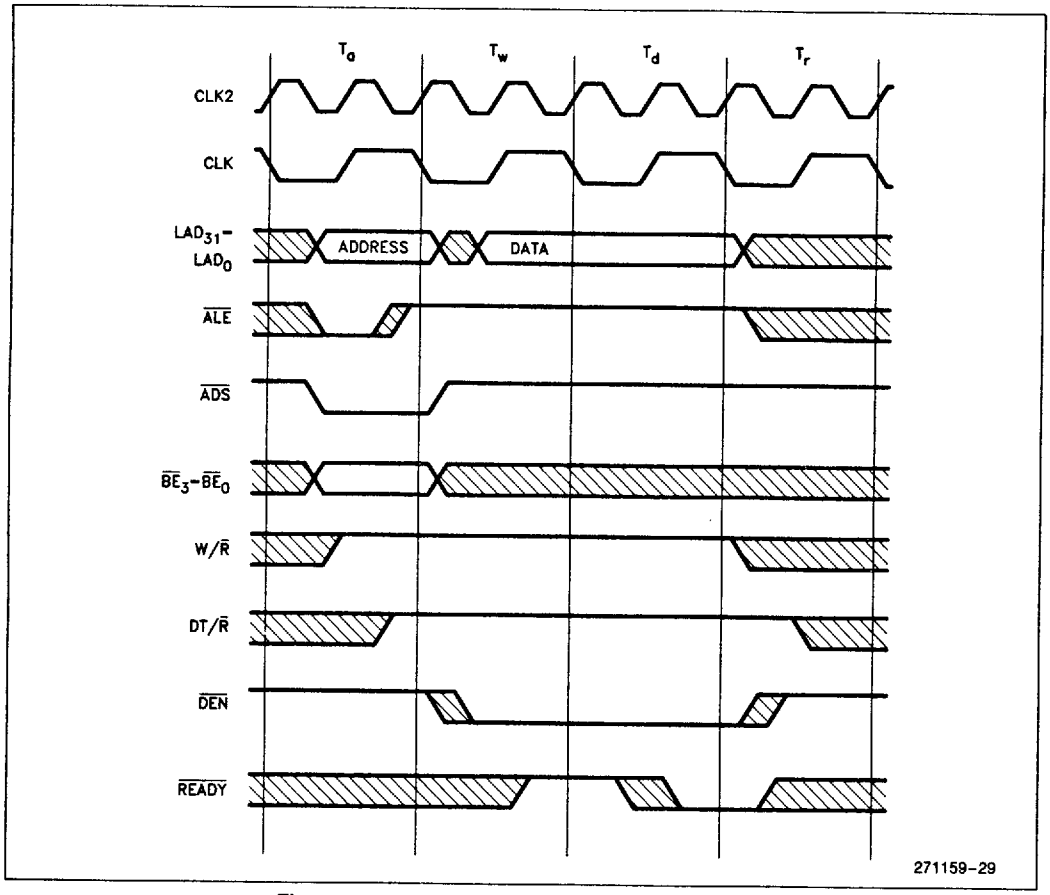


Figure 27. Write Transaction with One Wait State

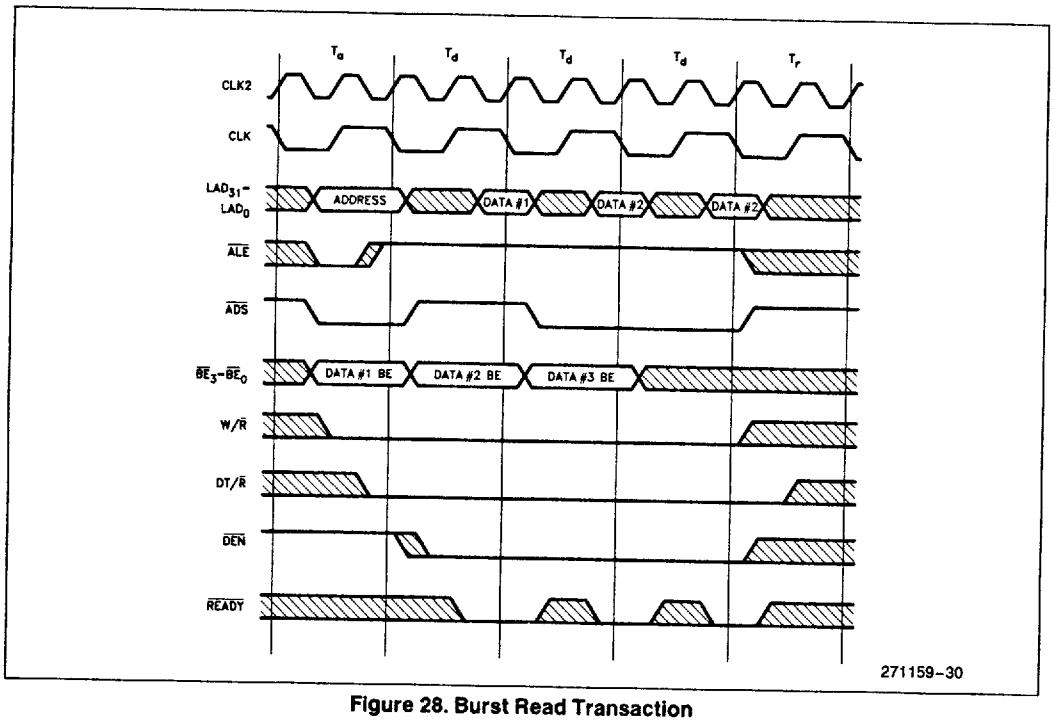


Figure 28. Burst Read Transaction

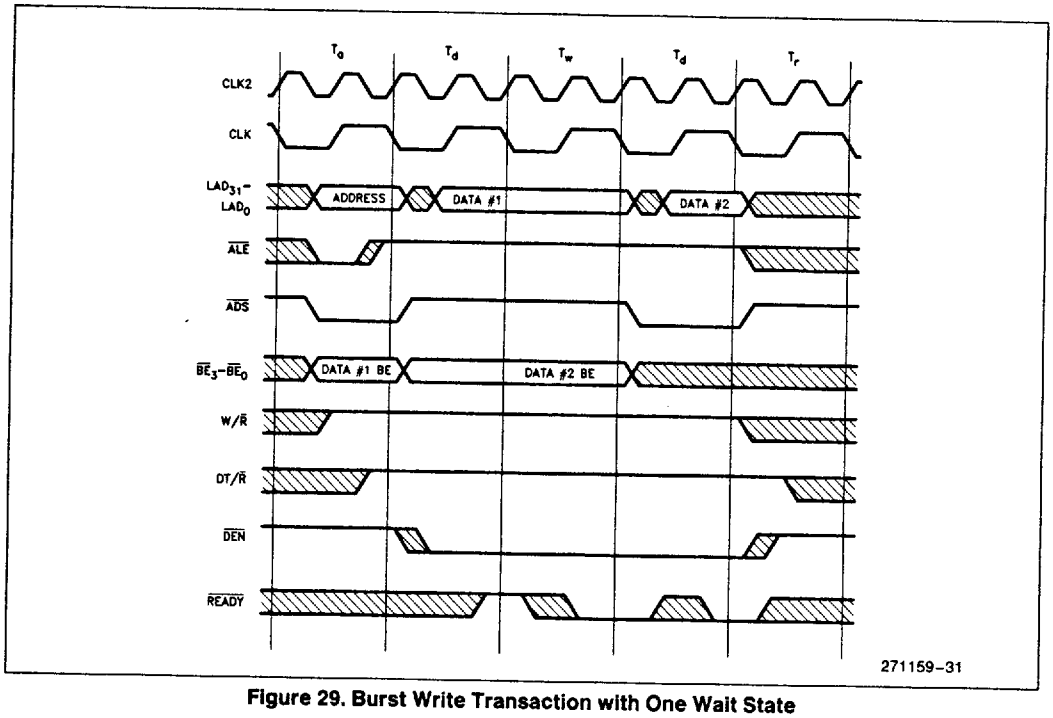


Figure 29. Burst Write Transaction with One Wait State

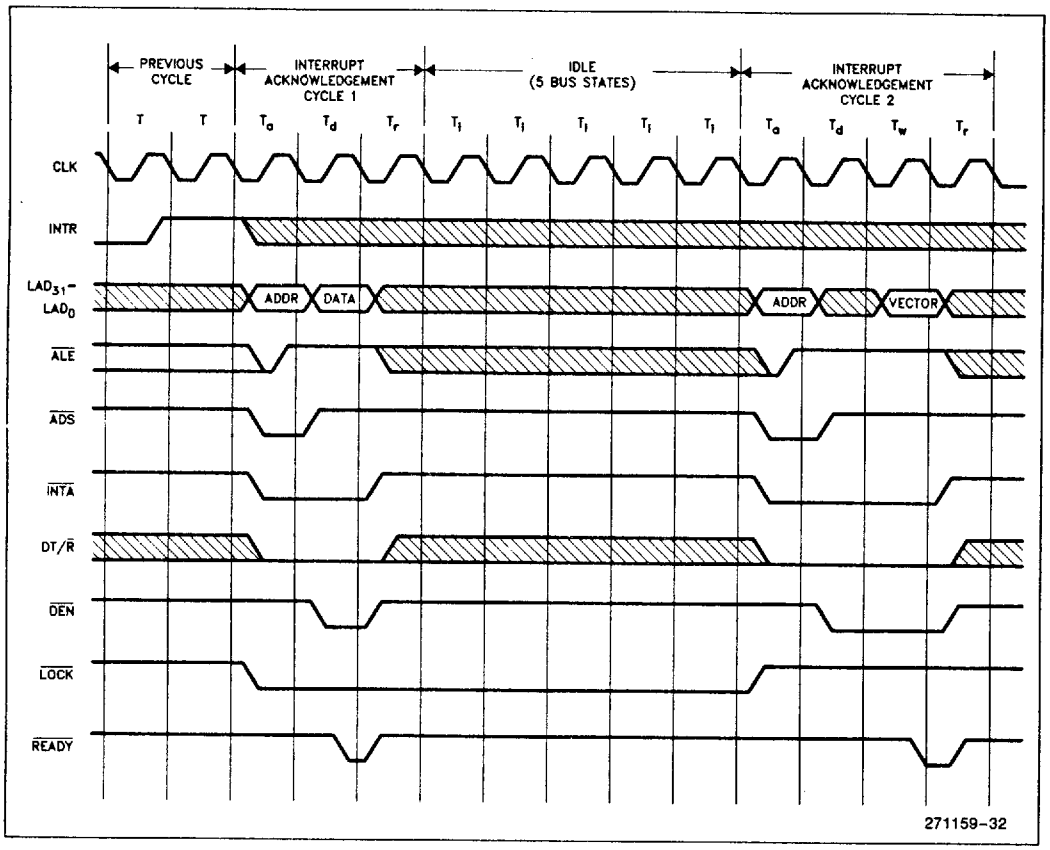


Figure 30. Interrupt Acknowledge Transaction

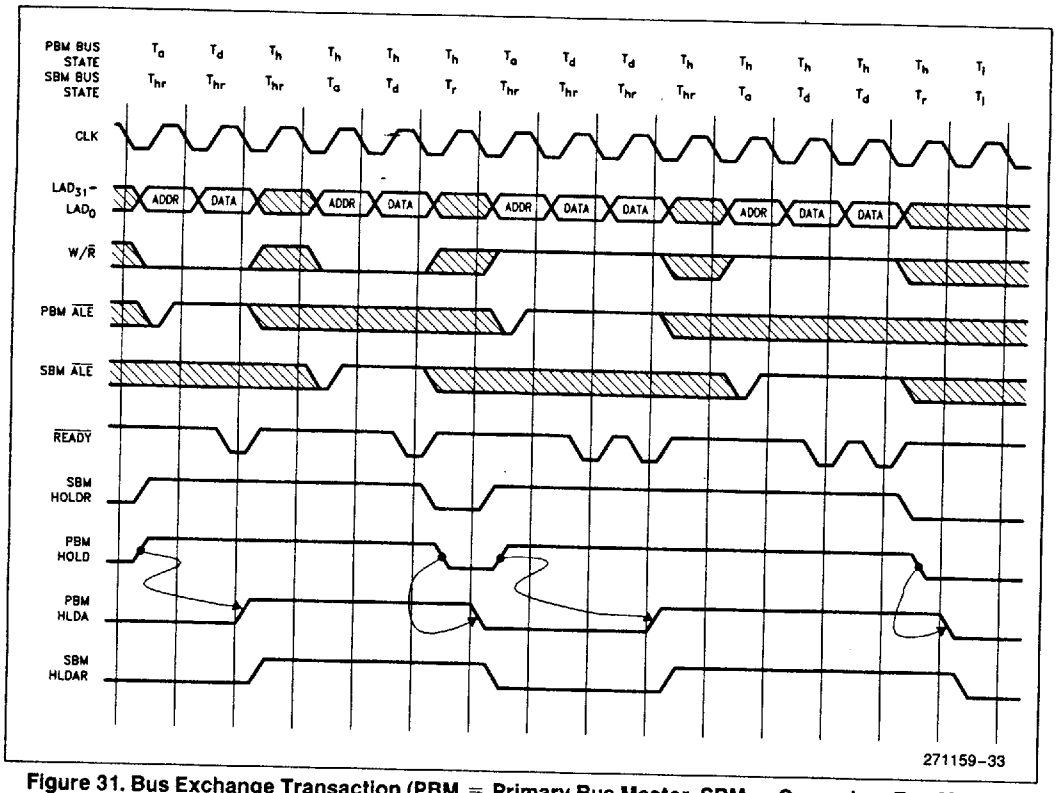


Figure 31. Bus Exchange Transaction (PBM = Primary Bus Master, SBM = Secondary Bus Master)

intel

0

DOMESTIC SERVICE OFFICES

ALABAMA

*Intel Corp.
5015 Bradford Dr., Suite 2
Huntsville 35805
Tel: (205) 830-4010

ALASKA

Intel Corp.
c/o TransAlaska Data Systems
300 Old Steese Hwy.
Fairbanks 99701-3120
Tel: (907) 452-4401

Intel Corp.
c/o TransAlaska Data Systems
1551 Lore Road
Anchorage 99507
Tel: (907) 522-1776

ARIZONA

*Intel Corp.
11225 N. 28th Dr.
Suite D-214
Phoenix 85029
Tel: (602) 869-4980

*Intel Corp.
500 E. Fry Blvd., Suite M-15
Sierra Vista 85635
Tel: (602) 459-5010

CALIFORNIA

†Intel Corp.
21515 Vanowen St., Ste. 116
Canoga Park 91303
Tel: (818) 704-8500

*Intel Corp.
2250 E. Imperial Hwy., Ste. 218
El Segundo 90245
Tel: (213) 640-6040

*Intel Corp.
1900 Prairie City Rd.
Folsom 95630-9597
Tel: (916) 351-6143
1-800-468-3548

Intel Corp.
9665 Cheasapeake Dr., Suite 325
San Diego 92123-1326
Tel: (619) 292-8086

**Intel Corp.
400 N. Tustin Avenue
Suite 450
Santa Ana 92705
Tel: (714) 835-9642

**†Intel Corp.
San Tomas 4
2700 San Tomas Exp., 2nd Floor
Santa Clara 95051
Tel: (408) 986-8086

COLORADO

*Intel Corp.
650 S. Cherry St., Suite 915
Denver 80222
Tel: (303) 321-8086

CONNECTICUT

*Intel Corp.
301 Lee Farm Corporate Park
83 Wooster Heights Rd.
Danbury 06810
Tel: (203) 748-3130

FLORIDA

**Intel Corp.
6363 N.W. 6th Way, Ste. 100
Ft. Lauderdale 33309
Tel: (305) 771-0600

*Intel Corp.
5850 T.G. Lee Blvd., Ste. 340
Orlando 32822
Tel: (407) 240-8000

GEORGIA

*Intel Corp.
3280 Pointe Pkwy., Ste. 200
Norcross 30092
Tel: (404) 449-0541

HAWAII

*Intel Corp.
U.S.I.S.C. Signal Batt.
Building T-1521
Shafter Flats
Shafter 96856

ILLINOIS

**†Intel Corp.
300 N. Martingale Rd., Ste. 400
Schaumburg 60173
Tel: (312) 605-8031

INDIANA

*Intel Corp.
8777 Purdue Rd., Ste. 125
Indianapolis 46268
Tel: (317) 875-0623

KANSAS

*Intel Corp.
10985 Cody, Suite 140
Overland Park 66210
Tel: (913) 345-2727

MARYLAND

**†Intel Corp.
10010 Junction Dr., Suite 200
Annapolis Junction 20701
Tel: (301) 206-2860
FAX: 301-206-3677

MASSACHUSETTS

**†Intel Corp.
3 Carlisle Rd., 2nd Floor
Westford 01886
Tel: (508) 692-1060

MICHIGAN

*†Intel Corp.
7071 Orchard Lake Rd., Ste. 100
West Bloomfield 48322
Tel: (313) 851-8905

MINNESOTA

*†Intel Corp.
3500 W. 80th St., Suite 360
Bloomington 55431
Tel: (612) 835-6722

MISSOURI

*Intel Corp.
4203 Earth City Exp., Ste. 131
Earth City 63045
Tel: (314) 291-1990

NEW JERSEY

**Intel Corp.
300 Sylvan Avenue
Englewood Cliffs 07632
Tel: (201) 567-0821

*Intel Corp.
Parkway 109 Office Center
328 Newman Springs Road
Red Bank 07701
Tel: (201) 747-2233

*Intel Corp.
280 Corporate Center
75 Livingston Ave., 1st Floor
Roseland 07068
Tel: (201) 740-0111

NEW YORK

*†Intel Corp.
2950 Expressway Dr. South
Islandia 11722
Tel: (516) 231-3300

*Intel Corp.
Westage Business Center
Bldg. 300, Route 9
Fishkill 12524
Tel: (914) 897-3860

NORTH CAROLINA

*Intel Corp.
5800 Executive Dr., Ste. 105
Charlotte 28212
Tel: (704) 568-8966

**Intel Corp.
2700 Wycliff Road
Suite 102
Raleigh 27607
Tel: (919) 781-8022

OHIO

**†Intel Corp.
3401 Park Center Dr., Ste. 220
Dayton 45414
Tel: (513) 890-5350

*†Intel Corp.
25700 Science Park Dr., Ste. 100
Beachwood 44122
Tel: (216) 464-2736

OREGON

Intel Corp.
15254 N.W. Greenbrier Parkway
Building B
Beaverton 97006
Tel: (503) 645-8051

*Intel Corp.
5200 N.E. Elam Young Parkway
Hillsboro 97123
Tel: (503) 681-8080

PENNSYLVANIA

*†Intel Corp.
455 Pennsylvania Ave., Ste. 230
Fort Washington 19034
Tel: (215) 641-1000

†Intel Corp.
400 Penn Center Blvd., Ste. 610
Pittsburgh 15235
Tel: (412) 823-4970

Intel Corp.
1513 Cedar Cliff Dr.
Camp Hill 17011
Tel: (717) 761-0860

PUERTO RICO

Intel Corp.
South Industrial Park
P.O. Box 910
Las Piedras 00671
Tel: (809) 733-8616

TEXAS

Intel Corp.
8815 Dyer St., Suite 225
El Paso 79904
Tel: (915) 751-0186

*Intel Corp.
313 E. Anderson Lane, Suite 314
Austin 78752
Tel: (512) 454-3628

**†Intel Corp.
12000 Ford Rd., Suite 401
Dallas 75234
Tel: (214) 241-8087

*Intel Corp.
7322 S.W. Freeway, Ste. 1490
Houston 77074
Tel: (713) 988-8086

UTAH

Intel Corp.
428 East 6400 South, Ste. 104
Murray 84107
Tel: (801) 263-8051

VIRGINIA

*Intel Corp.
1504 Santa Rosa Rd., Ste. 108
Richmond 23288
Tel: (804) 282-5668

WASHINGTON

*Intel Corp.
155 108th Avenue N.E., Ste. 386
Bellevue 98004
Tel: (206) 453-8086

CANADA

ONTARIO

Intel Semiconductor of
Canada, Ltd.
2650 Queensview Dr., Ste. 250
Ottawa K2B 8H6
Tel: (613) 829-9714
FAX: 613-820-5936

Intel Semiconductor of
Canada, Ltd.
190 Attwell Dr., Ste. 102
Rexdale M9W 6H8
Tel: (416) 675-2105
FAX: 416-675-2438

CUSTOMER TRAINING CENTERS

CALIFORNIA

2700 San Tomas Expressway
Santa Clara 95051
Tel: (408) 970-1700
1-800-421-0386

ILLINOIS

300 N. Martingale Road
Suite 300
Schaumburg 60173
Tel: (708) 706-5700
1-800-421-0386

MASSACHUSETTS

3 Carlisle Road, First Floor
Westford 01886
Tel: (301) 220-3380
1-800-328-0386

MARYLAND

10010 Junction Dr.
Suite 200
Annapolis Junction 20701
Tel: (301) 206-2860
1-800-328-0386

SYSTEMS ENGINEERING MANAGERS OFFICES

MINNESOTA

3500 W. 80th Street
Suite 360
Bloomington 55431
Tel: (612) 835-6722

NEW YORK

2950 Expressway Dr., South
Islandia 11722
Tel: (506) 231-3300

†System Engineering locations
*Carry-in locations
**Carry-in/mail-in locations

CG/SALE/101789

D-10