3-Volt System Logic for Personal Computers

Data Book

ADVANCED MICRO DEVICES

3-VOLT SYSTEM LOGIC FOR PERSONAL COMPUTERS DATA BOOK

The desktop revolution has made the personal computer an indispensable business tool. Now, with the recent introduction of 3-volt ICs, designers are breaking the barrier to pervasive portable PC use by extending the useful battery life without weighing down their products. As manufacturers strike the ideal balance between power requirements, useful battery life and weight, users will have virtually unlimited portable machine choices.

AMD's announcement of the world's first 3-volt Am386 microprocessor in October 1991, plus the five other products described in this data book, encouraged designers to optimize designs to produce the battery powered products *customers really want.* In addition, over 50 other companies have announced various 3-volt components.

AMD recognizes that 3-volt operation alone does not make the ultimate notebook. For this reason, the Am386SXLV and Am386DXLV microprocessors support software power management schemes such as Microsoft's Advanced Power Management (APM) specification and the System Management Mode (SMM) of operation. APM allows the system BIOS, operating system, and APM-compliant applications to share critical power management data while preserving compatibility between hardware and software. SMM allows system hardware or software to interrupt the CPU in order to efficiently control devices and peripherals, further reducing overall power consumption. The combination of APM and SMM plus the overall power savings from 3-volt operation is opening new markets.

The Am386SXLV and Am386DXLV microprocessors provide the enabling technology for a new generation of portable systems, demonstrating AMD's commitment to accelerate the rate of developments for new markets.

In keeping with our philosophy of offering memories that solve customer's needs, AMD is proud to announce a family of 2.7 to 3.6 V EPROMs. This low-voltage family of EPROMs, designated as Am27LV, consists of 1 and 2 Mbit density devices with speeds as fast as 120 ns. In addition, the voltage range for each device has been extended to accommodate systems that have regulated power supplies (3.0 to 3.6 V) and those that are battery operated (2.7 to 3.6 V).

The Am53C94LV is the newest addition to the growing AMD SCSI Family. This device is the first 3.3-V SCSI controller in the market. The Am53C94LV consumes fifty percent less power than 5-V SCSI, while sustaining the same performance as that of the 5-V device (5 Mbyte/s SCSI transfer rate, 20 Mbyte/s DMA transfer rate). It is conveniently packaged in a 100-pin PQFP package to reduce board real estate. The Am53C94LV is the ideal SCSI solution for battery powered peripherals, low power, notebooks, sub-notebooks, and docking stations.

Consistent with AMD's leadership role in the world of programmable logic, AMD is releasing popular PAL®, device architecture for use in 3.3-V systems. The PALLV16V8Z-30 and the PALLV22V10Z-25 will be the first such devices. They are optimized for low-power operation. These low-power devices will provide logic anywhere power comes at a premium. This includes such systems as portable computers, telecommunications, and portable instrumentation.

Successive devices will also be optimized for high performance. These devices will serve those applications that are using 3.3-V advanced technology to push the limits of performance, such as high-speed desktop and mini-computers.

Chapter 1

Am386 Family Products

CHAPTER 1
Am386 Family Products

FINAL

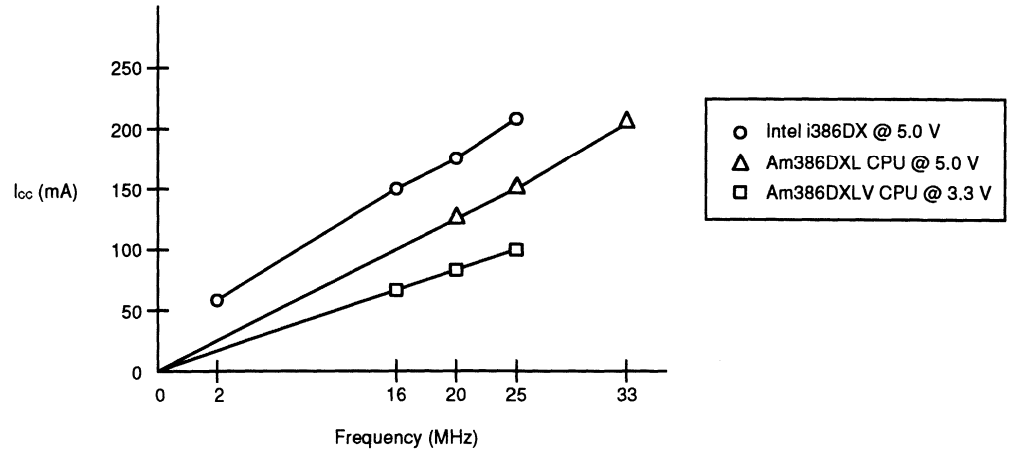
Am386DXLV

High-Performance, Low-Voltage, 32-Bit Microprocessor

Advanced Micro Devices

DISTINCTIVE CHARACTERISTICS

- **Operating voltage range 3.0 V to 5.5 V—ideal for portable PC applications**
  - —25-MHz operating frequency for 3.0 V to 5.5 V
  - —33-MHz operating frequency for 4.5 V to 5.5 V
  - —Twice the battery life of existing 5-V designs
  - —Wide range of chipsets and other logic available for 3-V systems with support for Standby Mode operation
  - —True static design for long battery life
  - —Power consumption 85% lower than Intel i386DX, 65% lower than Am386DXL processor
  - —Performance on demand (0 to 33 MHz)
- **SMM (System Management Mode) for system and power management**
  - —SMI (System Management Interrupt) for power management independent of processor operating mode and operating system
  - —SMI coupled with I/O instruction break feature provides transparent power off and auto resume of peripherals which may not be "power aware"
  - —SMI is non-maskable and has higher priority than Non-Maskable Interrupt (NMI)
  - —Automatic save and restore of the microprocessor state
  - —Wide range of chipsets supporting SMM available to allow product differentiation
- **Lower heat dissipation for fanless systems**
- **"Float" input to facilitate system debug and test**
- **Compatible with 386DX systems and software**
- **Supports 387DX-compatible math coprocessors**
- **132-pin PQFP package with optional protective ring for better lead coplanarity**
- **AMD® advanced 0.8 micron CMOS technology**

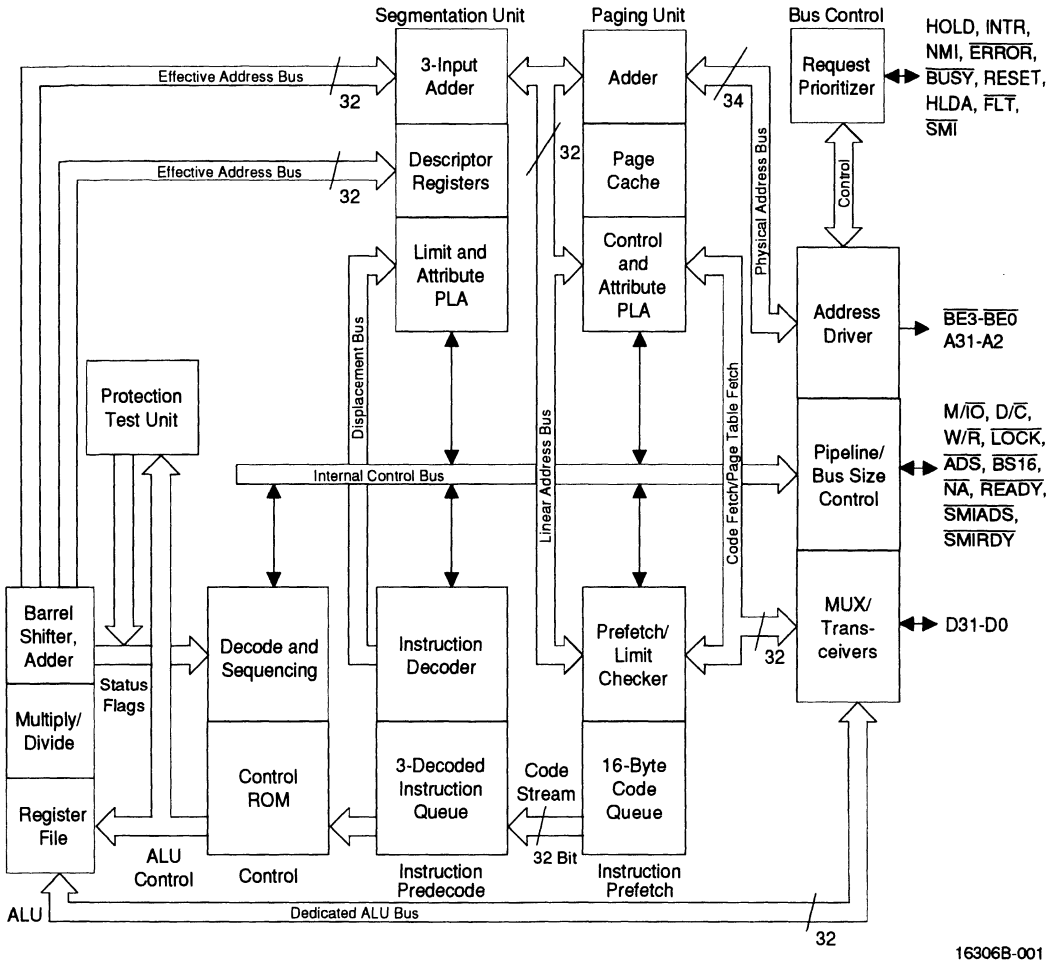


Note: Inputs at V_{CC} or V_{SS}.

Typical Power Consumption

Publication #: 16306 Rev. E Amendment /0
Issue Date: March 1993

BLOCK DIAGRAM



GENERAL DESCRIPTION

The Am386DXLV microprocessor is a low-voltage, true static implementation of the Intel i386DX. The operating voltage range is 3.0 V to 5.5 V. The low-voltage operation makes the Am386DXLV microprocessor ideal for both desktop and battery-powered portable personal computers. For desktop PCs, low heat dissipation allows the system designers to remove or reduce the size and cost of the system cooling fan. The Am386DXLV microprocessor operates at a maximum speed of 25 MHz from 3.0 to 5.5 V and at a maximum speed of 33 MHz from 4.5 to 5.5 V.

The Am386DXLV microprocessor's lower operating voltage and true static design enables longer battery life and/or lower weight for portable applications. At 25 MHz, this device has 80% lower operating Icc than the Intel i386DX. Lowering typical operating voltage from 5.0 V to 3.3 V doubles the battery life. Standby Mode allows the Am386DXLV microprocessor to be clocked down to 0 MHz (DC) and retain full register contents. In Standby Mode, typical current draw is 0.01 mA, a greater than 1000X reduction in power consumption versus the Intel i386DX.

The Am386DXLV processor is available in a small footprint 132-pin Plastic Quad Flat Pack (PQFP) package. This surface-mount package is 40% smaller than a PGA package, allowing smaller lower-cost board designs without the need for a socket.

Additionally, the Am386DXLV processors comes with SMM for system and power management. SMI is a non-maskable, higher priority interrupt than NMI and has its own code space (1 Mb in Real Mode and 4 Gb in Protected Mode). SMI can be coupled with the I/O instruction break feature to implement transparent power management of peripherals. SMM can be used by system designers to implement system and power management code independent of the operating system or the processor mode.

The Am386DXLV processor incorporates a float pin that places all outputs in a three-state mode to facilitate board test and debug.

FUNCTIONAL DESCRIPTION

Benefits of Lower Operating Voltage

The Am386DXLV microprocessor has an operating voltage range of 3.0 V to 5.5 V. Low voltage allows for lower operating power consumption, longer battery life, and/or smaller batteries for notebook applications.

Because power is proportional to the square of the voltage, reduction of the supply voltage from 5.0 V to 3.3 V reduces power consumption by 56%. This directly translates to a doubling of battery life for portable applications. Lower power consumption can also be used to reduce the size and weight of the battery. Thus, 3.3-V designs facilitate a reduction in the form factor.

A lower operating voltage results in a reduction of I/O voltage swings. This reduces noise generation and provides a less hostile environment for board design. Lower operating voltage also reduces electromagnetic radiation noise, and makes FCC approval easier to obtain.

SMM–System Management Mode

The Am386DXLV processor has a new System Management Mode (SMM) for system and power management. This mode consists of two features: System Management Interrupt (SMI) and I/O instruction break.

SMI–System Management Interrupt

SMI is implemented by using special bus interface pins. This interrupt method can be used to perform system management functions such as power management independent of processor operating mode (Real, Protected, or Virtual 8086 Modes).

SMI can also be invoked in software. This allows system software to communicate with SMI power management code. In addition, a UMOV instruction allows data transfers between SMI and normal system memory spaces.

Activating the $\overline{SMI}$ pin invokes a sequence that saves the operating state of the processor into a separate SMM memory space, independent of the main system memory. After the state is saved, the processor is forced into Real Mode and begins execution at address FFFFFFF0h where a far jump to the SMM code is executed. This Real Mode code can perform its system management function and then resume execution of the normal system software by executing an RES3 instruction that will reload the saved processor state and continue execution in the main system memory space. See Figure 1 for a general flowchart of an SMM operation.

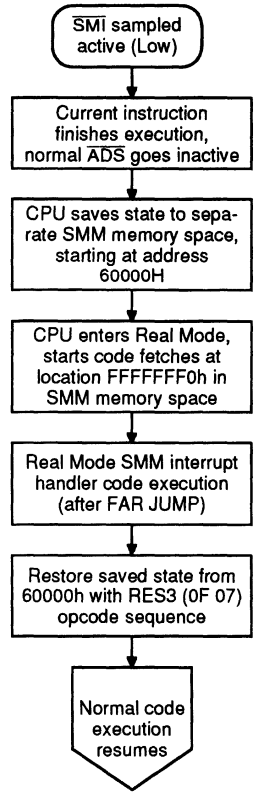


Figure 1. SMM Flow

CPU Interface—Pin Functions

The CPU interface for SMM consists of three pins dedicated to the SMI function. One pin, $\overline{SMI}$, is the new interrupt input. The other two pins, $\overline{SMIADS}$ and $\overline{SMIRDY}$, provide the control signals necessary for the separate SMI Mode memory space.

Description of SMM Operation

The execution of a System Management Interrupt has four distinct phases: the initiation of the interrupt via $\overline{SMI}$, a processor state save, execution of the SMM interrupt code, and a processor state restore (to resume normal operation).

Interrupt Initiation

A System Management Interrupt is initiated by the driving of a synchronous, active Low pulse on the $\overline{SMI}$ pin until the first $\overline{SMIADS}$ is asserted.. This pulse period will ensure recognition of the interrupt. The CPU drives the

$\overline{SMI}$ pin active Low after the completion of the current operation (active bus cycle, instruction execution, or both). The active drive of the pin by the CPU is released at the end of the interrupt routine, following the last register read of the saved state. The CPU will drive $\overline{SMI}$ High for two CLK2 cycles prior to releasing the drive of $\overline{SMI}$.

An SMI cannot be masked off by the CPU, and it will always be recognized by the CPU, regardless of operating mode. This includes the Real, Protected, and Virtual 8086 Modes of the processor.

While the CPU is in SMI Mode, a bus hold request via the HOLD pin is granted. The HLDA pin goes active after bus release and the $\overline{SMIADS}$ pin floats along with the other pins that normally float during a bus hold cycle. The $\overline{SMI}$ pin does not float during Bus Hold cycles.

Processor State Save

The first set of SMM bus transfer cycles after the CPU's recognition of an active SMI is the processor saving its state to an external RAM array in a separate address space from main system memory. This is accomplished by using the $\overline{SMIADS}$ and $\overline{SMIRDY}$ pins for initiation and termination of bus cycles, instead of the $\overline{ADS}$ and $\overline{READY}$ pins. The 32-bit addresses to which the CPU saves its state are 60000H–600CBh and 60100h–60127h. These are fixed address locations for each register saved.

Pipelining must be disabled by asserting the $\overline{NA}$ pin High in SMM. For $\overline{ADS}$ initiated bus cycles, $\overline{BS16}$ can be used. The value of $\overline{BS16}$ is ignored for $\overline{SMIADS}$ initiated bus cycles.

The save state takes approximately 630 CLK2 cycles with zero wait state memory. There are 61 data transfer cycles.

SMI Code Execution

After the processor state is saved to the separate SMM memory space, the execution of the SMI interrupt routine code begins. The processor enters Real Mode, sets most of the register values to "reset" values (those values normally seen after a CPU reset), and begins fetching code from address FFFFFFF0h in the separate SMM memory space. Normally, the first thing the interrupt routine code does is a FAR JUMP to the Real Mode entry point for the SMI interrupt routine, which is also in SMM memory space.

INTR and NMI are disabled upon entry to SMM. See the Am386DXLV/Am386SXLV Microprocessor Technical Manual for further details on interrupts in SMM. The SMM code can be located anywhere within the 1-Mb Real Mode address space, except where the processor state is saved. I/O cycles, as a result of the IN, OUT, INS, and OUTS instructions, will go to the normal address space, utilizing the normal $\overline{ADS}$ and $\overline{READY}$ bus interface signals. This facilitates power management code manipulating system hardware registers as needed through the standard I/O subsystem; a separate I/O space is not implemented.

Processor State Restore (Resuming Normal Execution)

Returning to normal code execution in the main system memory, including restoring the processor operating mode, is accomplished by executing a RES3 (0Fh 07h) instruction. This instruction invokes a restore CPU state operation that reloads the CPU registers from the saved data in the RAM controlled by $\overline{SMIADS}$ and $\overline{SMIRDY}$.

The ES:EDI register pair must point to the physical address of the saved state, this is normally at 60000h. In Real Mode, the address is calculated as ES · 16 + EDI. The saved state should not cross a 64K boundary. The special opcode sequence RES3 should be executed to start the restore state operation. After completion of the restore state operation, the $\overline{SMI}$ pin will be deactivated by the CPU and normal code execution will continue at the point where it left off before the SMI occurred.

In a zero wait state memory implementation, approximately 574 CLK2 cycles complete the restore state operation. There are 61 data transfer cycles.

Software Features

There are several features of the SMI function that provide support for special operations during the execution of the system's software. These features involve the execution of reserved opcodes to induce specific SMI related operations.

Software SMI Generation

Besides hardware initiation of the System Management Interrupt via the $\overline{SMI}$ pin, there is also a software induced SMI mechanism. Generating a soft $\overline{SMI}$ involves setting a control bit 12 in Debug Register 7 (DR7) and executing an SMI instruction (reserved opcode F1h).

The functional sequence of the software-based SMI is identical to the hardware-based SMI with the exception that the $\overline{SMI}$ pin is not initially driven active by an external source. Upon execution of a soft SMI opcode, the $\overline{SMI}$ pin is driven active (Low) by the processor before the save state operation begins.

Memory Transfers to Main System Memory

While executing an SMI routine, the interrupt code can initiate memory data reads and writes to the main system memory using the normal $\overline{ADS}$ and $\overline{READY}$ pins. This is accomplished by using reserved opcodes that are special forms of the MOV instruction (called UMOV). The UMOV opcodes can move byte, word, or doubleword register operands to or from main system memory. Multiple data transfers using the normal $\overline{ADS}$ and $\overline{READY}$ pins will occur if the operands are misaligned relative to the effective address used. The UMOV opcodes are 0Fh 10h, 0Fh 11h, 0Fh 12h, and 0Fh 13h.

The UMOV instruction can use any of the 386 addressing modes, as specified in the ModR/M byte of the opcode. Note that the 16- and 32-bit versions are the same opcodes with the exception of the 66h operand size prefix. The $\overline{BS16}$ line is recognized during the normal memory space data transfer(s) initiated by these instructions.

I/O Instruction Break

The Am386DXLV processor has an I/O instruction break feature that allows the system logic to implement I/O trapping for peripheral devices. To enable the I/O instruction break feature, $\overline{IIBEN}$ must first be asserted active Low. On detecting an I/O instruction, the processor prevents the execution unit from executing further instructions until $\overline{READY}$ is driven active Low by the system. Once $\overline{READY}$ is driven active, the execution unit either immediately responds to any active interrupt request or continues executing instructions following the I/O instruction that caused the break.

The I/O instruction break feature can be used to allow system logic to implement I/O trapping for peripheral devices. On sensing an I/O instruction, the system drives the $\overline{SMI}$ interrupt active before driving $\overline{READY}$ active. This ensures that the service routine is executed immediately following the I/O instruction that caused the break. (If the I/O instruction break feature is not enabled via $\overline{IIBEN}$, several instructions that follow the I/O instruction that caused the break will execute before the SMI service routine is executed.) The SMI service routine can access the peripheral for which $\overline{SMI}$ was asserted and modify the peripheral's state.

The SMI service routine normally returns to the instruction following the I/O instruction that caused the break. By modifying the saved state instruction pointer, the routine can choose to return to the I/O instruction that caused the break and re-execute that instruction. The default is to return to the following instruction (except for REP I/O strings). To re-execute the I/O instruction that caused the break, the SMI service routine must copy the I/O instruction pointer over the default pointer. This feature is particularly useful when an application program requests an access to a peripheral that has been powered down. The SMI service routine can restore power to the peripheral and initiate a re-execution sequence transparent to the application program. This re-execution feature should only be used if the $\overline{SMI}$ is in response to an I/O trap with $\overline{IIBEN}$ active.

Note that the I/O instruction break feature is not enabled for memory mapped I/O devices or for 80387 bus cycles, even if $\overline{IIBEN}$ is active.

I/O Instruction Break Timing

The I/O Instruction Break feature requires that $\overline{SMI}$ be sampled active (Low) by the processor at least three CLK2 edges before the CLK2 edge that ends the I/O cycle with an active $\overline{READY}$ signal. This timing applies for both pipelined and non-pipelined cycles. If this timing constraint is not met, additional instructions may be executed by the internal execution unit prior to entering SMI Mode.

Depending on the state of the prefetch queue at the time $\overline{SMI}$ is asserted, instruction fetch cycles may occur on the normal $\overline{ADS}$ interface before the SMI save state process begins with the assertion of $\overline{SMIADS}$. However, this fetched code will not be executed.

True Static Operation

The Am386DXLV microprocessor incorporates a true static design. Unlike dynamic circuit design, the Am386DXLV device eliminates the minimum operating frequency restriction. It may be clocked from its maximum speed all the way down to 0 MHz (DC). System designers can use this feature to design true 32-bit battery-powered portable PCs with long battery life.

Standby Mode

This true static design of the Am386DXLV microprocessor allows for a Standby Mode. At any of its operating speeds, the Am386DXLV microprocessor will retain its state (i.e., the contents of all of its registers). By shutting off the clock completely, the device enters Standby Mode. Since power consumption is a function of clock frequency, operating power consumption is reduced as the frequency is lowered. In Standby Mode, typical current draw is reduced to less than 0.01 mA at DC. This feature not only saves battery life, but it also simplifies the design of power-conscious notebook computers in the following ways.

1. Eliminates the need for software in BIOS to save and restore the contents of registers.
2. Allows simpler circuitry to control stopping of the clock since the system does not need to know the state of the processor.

Lower Operating Icc

True static design also allows lower operating Icc when operating at any speed.

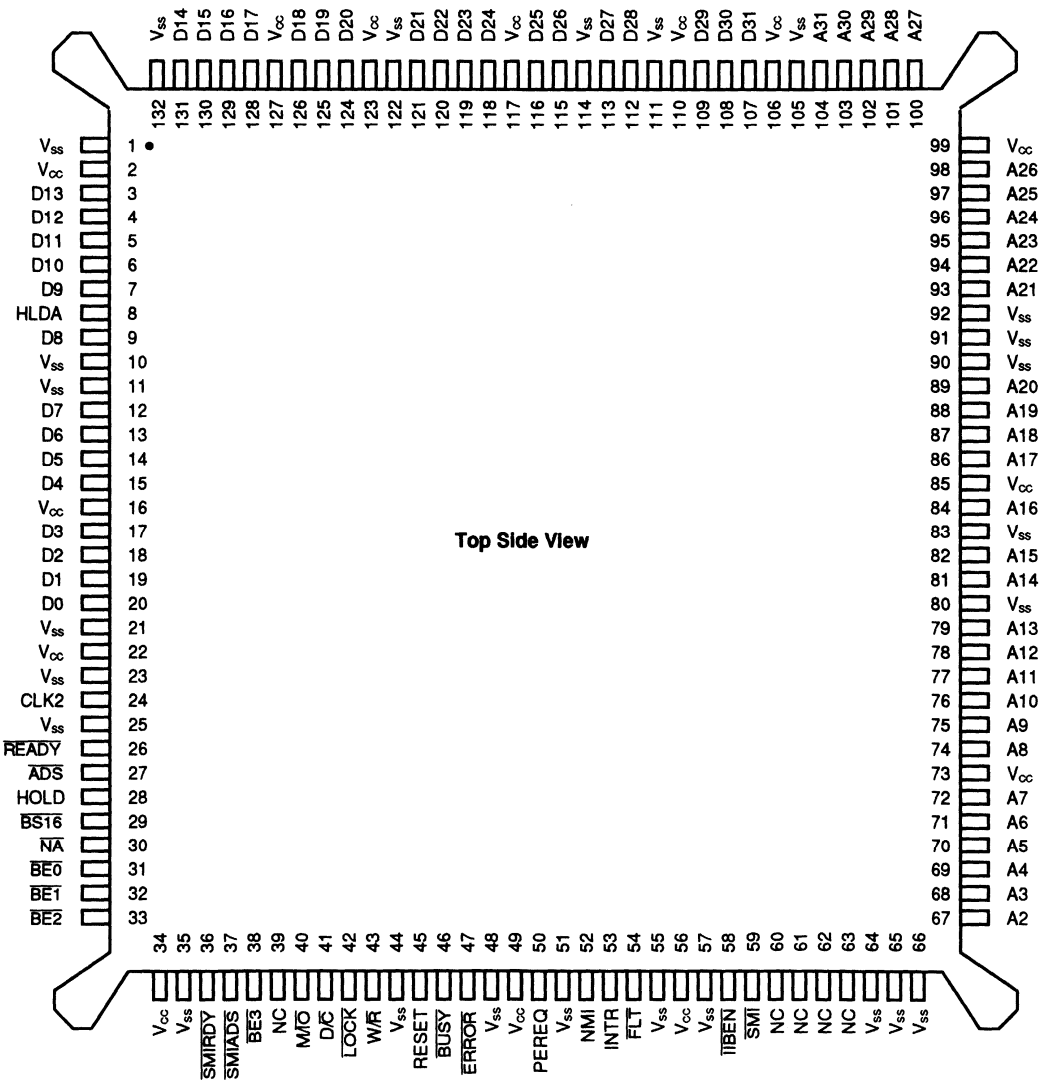
Performance on Demand

The Am386DXLV microprocessor retains its state at any speed from 0 MHz (DC) to its maximum operating speed. With this feature, system designers may vary the operating speed of the system to extend the battery life in portable systems.

For example, the system could operate at low speeds during inactivity or polling operations. However, upon interrupt, the system clock can be increased up to its maximum speed. After a user-defined time-out period, the system can be returned to a low (or 0 MHz) operating speed without losing its state. This design maximizes battery life while achieving optimal performance.

CONNECTION DIAGRAM
132-Lead Plastic Quad Flat Pack (PQFP) Package — Top Side View

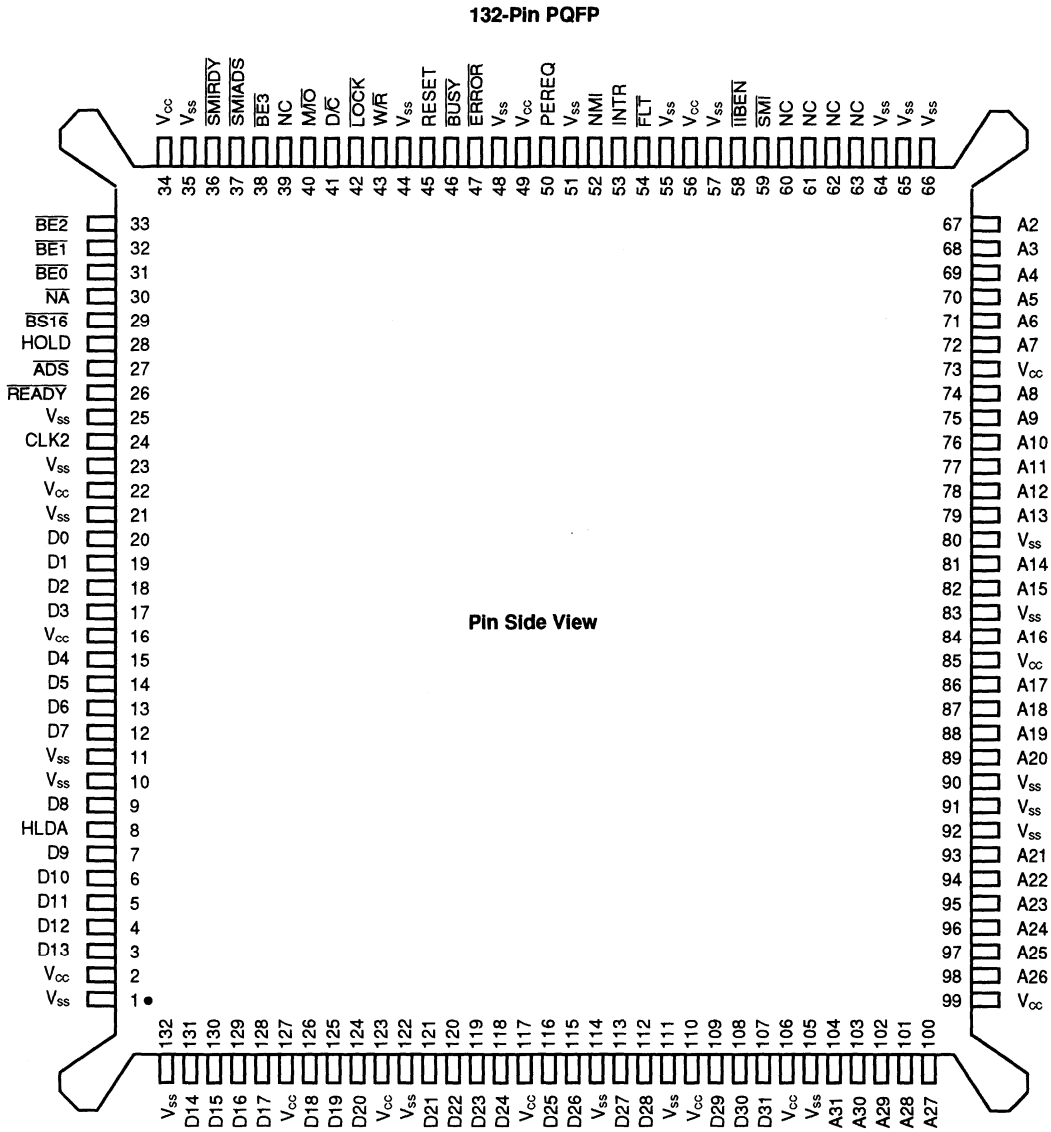
132-Pin PQFP



Notes: Pin 1 is marked for orientation.
NC = Not connected; connection of an NC pin may cause a malfunction or incompatibility with future shippings of the Am386DXLV microprocessor.

CONNECTION DIAGRAM

132-Lead Plastic Quad Flat Pack (PQFP) Package — Pin Side View



Notes: Pin 1 is marked for orientation.
NC = Not connected; connection of an NC pin may cause a malfunction or incompatibility with future shippings of the Am386DXLV microprocessor.

PQFP Pin Designations (Sorted by Functional Grouping)

Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.
A2	67	A24	96	D6	13	D28	112	V_{CC}	16	V_{SS}	51
A3	68	A25	97	D7	12	D29	109		22		55
A4	69	A26	98	D8	9	D30	108		34		57
A5	70	A27	100	D9	7	D31	107		49		64
A6	71	A28	101	D10	6	D/$\overline{C}$	41		56		65
A7	72	A29	102	D11	5	$\overline{ERROR}$	47		73		66
A8	74	A30	103	D12	4	$\overline{FLT}$	54		85		80
A9	75	A31	104	D13	3	HLDA	8		99		83
A10	76	$\overline{ADS}$	27	D14	131	HOLD	28		106		90
A11	77	$\overline{BE0}$	31	D15	130	$\overline{IIBEN}$	58		110		91
A12	78	$\overline{BE1}$	32	D16	129	INTR	53		117		92
A13	79	$\overline{BE2}$	33	D17	128	$\overline{LOCK}$	42		123		105
A14	81	$\overline{BE3}$	38	D18	126	M/$\overline{IO}$	40		127		111
A15	82	$\overline{BS16}$	29	D19	125	$\overline{NA}$	30	V_{SS}	1		114
A16	84	$\overline{BUSY}$	46	D20	124	NMI	52		10		122
A17	86	CLK2	24	D21	121	PEREQ	50		11		132
A18	87	D0	20	D22	120	$\overline{READY}$	26		21	W/$\overline{R}$	43
A19	88	D1	19	D23	119	RESET	45		23	NC	39
A20	89	D2	18	D24	118	$\overline{SMI}$	59		25		60
A21	93	D3	17	D25	116	$\overline{SMIADS}$	37		35		61
A22	94	D4	15	D26	115	$\overline{SMIRDY}$	36		44		62
A23	95	D5	14	D27	113	V_{CC}	2		48		63

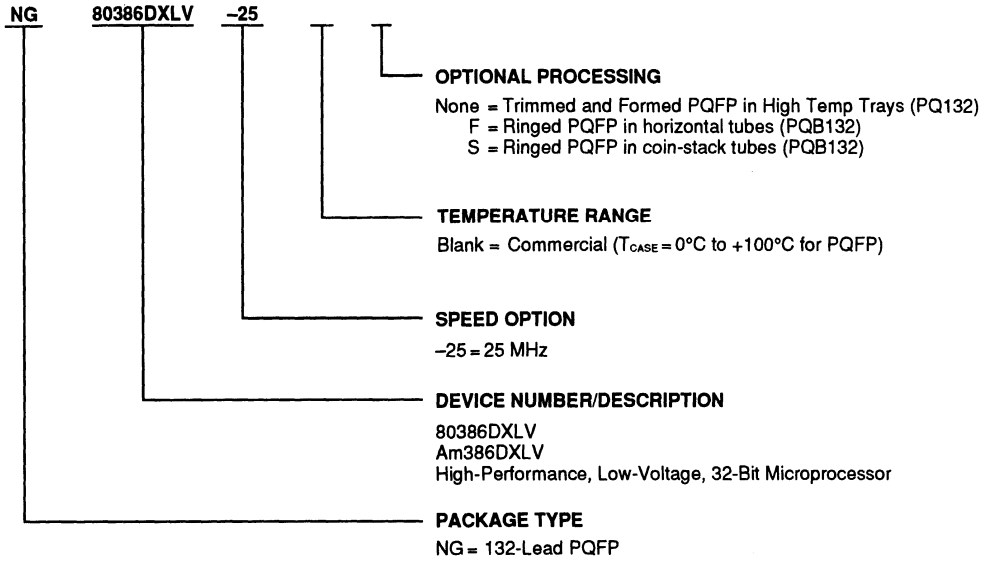
PQFP Pin Designations (Sorted by Pin Number)

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	V_{SS}	23	V_{SS}	45	RESET	67	A2	89	A20	111	V_{SS}
2	V_{CC}	24	CLK2	46	$\overline{BUSY}$	68	A3	90	V_{SS}	112	D28
3	D13	25	V_{SS}	47	$\overline{ERROR}$	69	A4	91	V_{SS}	113	D27
4	D12	26	$\overline{READY}$	48	V_{SS}	70	A5	92	V_{SS}	114	VSS
5	D11	27	$\overline{ADS}$	49	V_{CC}	71	A6	93	A21	115	D26
6	D10	28	HOLD	50	PEREQ	72	A7	94	A22	116	D25
7	D9	29	$\overline{BS16}$	51	V_{SS}	73	V_{CC}	95	A23	117	V_{CC}
8	HLDA	30	$\overline{NA}$	52	NMI	74	A8	96	A24	118	D24
9	D8	31	$\overline{BE0}$	53	INTR	75	A9	97	A25	119	D23
10	V_{SS}	32	$\overline{BE1}$	54	$\overline{FLT}$	76	A10	98	A26	120	D22
11	V_{SS}	33	$\overline{BE2}$	55	V_{SS}	77	A11	99	V_{CC}	121	D21
12	D7	34	V_{CC}	56	V_{CC}	78	A12	100	A27	122	V_{SS}
13	D6	35	V_{SS}	57	V_{SS}	79	A13	101	A28	123	V_{CC}
14	D5	36	$\overline{SMIRDY}$	58	$\overline{IIBEN}$	80	V_{SS}	102	A29	124	D20
15	D4	37	$\overline{SMIADS}$	59	$\overline{SMI}$	81	A14	103	A30	125	D19
16	V_{CC}	38	$\overline{BE3}$	60	NC	82	A15	104	A31	126	D18
17	D3	39	NC	61	NC	83	V_{SS}	105	V_{SS}	127	V_{CC}
18	D2	40	M/$\overline{IO}$	62	NC	84	A16	106	V_{CC}	128	D17
19	D1	41	D/$\overline{C}$	63	NC	85	V_{CC}	107	D31	129	D16
20	D0	42	$\overline{LOCK}$	64	V_{SS}	86	A17	108	D30	130	D15
21	V_{SS}	43	W/$\overline{R}$	65	V_{SS}	87	A18	109	D29	131	D14
22	V_{CC}	44	V_{SS}	66	V_{SS}	88	A19	110	V_{CC}	132	V_{SS}

ORDERING INFORMATION

Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of the elements below.

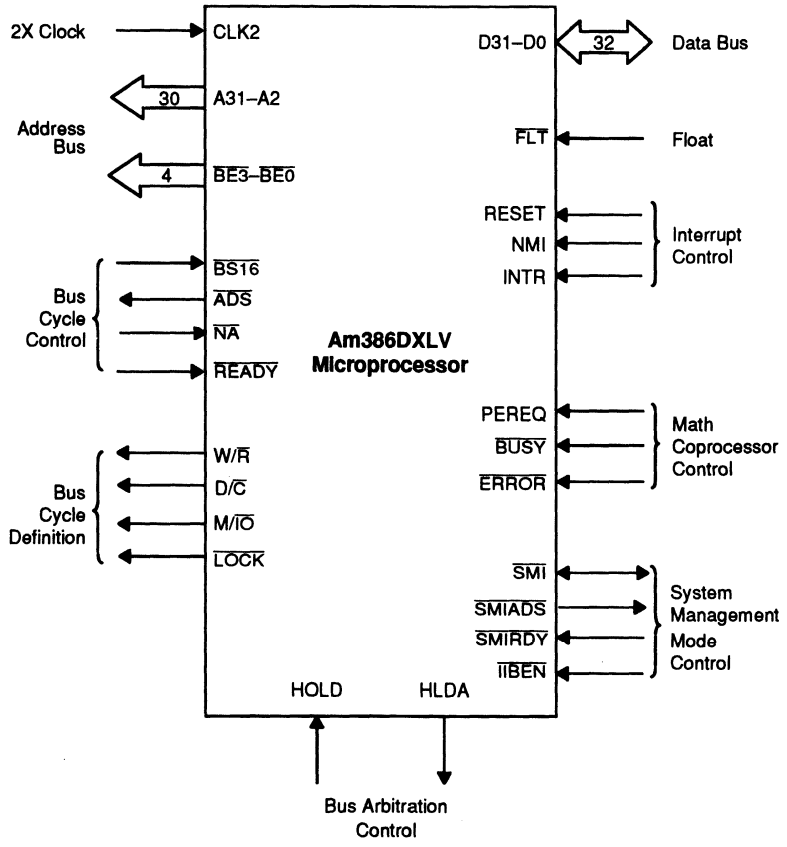


Valid Combinations		
NG	80386DXLV	–25 –25F –25S

Valid Combinations

Valid Combinations lists configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations and to check on newly released combinations.

LOGIC SYMBOL



15021B–003

PIN DESCRIPTION

A31–A2

Address Bus (Outputs)

Outputs physical memory or port I/O addresses.

$\overline{ADS}$

Address Status (Active Low; Output)

Indicates that a valid bus cycle definition and address (W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{BE3}$–$\overline{BE0}$, and A31–A2) are being driven at the Am386DXLV microprocessor pins. Bus cycles initiated by $\overline{ADS}$ must be terminated by $\overline{READY}$.

$\overline{BE3}$–$\overline{BE0}$

Byte Enable (Active Low; Outputs)

Indicates which data bytes of the data bus take part in a bus cycle.

$\overline{BS16}$

Bus Size 16 (Active Low; Input)

Allows direct connection of 32-bit and 16-bit data buses. $\overline{BS16}$ has an internal pull-up resistor.

$\overline{BUSY}$

Busy (Active Low; Input)

Signals a busy condition from a processor extension. $\overline{BUSY}$ has an internal pull-up resistor.

CLK2

Clock (Input)

Provides the fundamental timing for the Am386DXLV microprocessor.

D31–D0

Data Bus (Inputs/Outputs)

Inputs data during memory, I/O, and interrupt acknowledge read cycles and outputs data during memory and I/O write cycles.

D/$\overline{C}$

Data/Control (Output)

A bus cycle definition pin that distinguishes data cycles, either memory or I/O, from control cycles (which are interrupt acknowledge, halt, and instruction fetching).

$\overline{ERROR}$

Error (Active Low; Input)

Signals an error condition from a processor extension. $\overline{ERROR}$ has an internal pull-up resistor.

$\overline{FLT}$

Float (Active Low; Input)

An input signal which forces all bidirectional and output signals, including HLDA, to the three-state condition. $\overline{FLT}$ has an internal pull-up resistor.

HLDA

Bus Hold Acknowledge (Active High; Output)

Indicates that the Am386DXLV microprocessor has surrendered control of its local bus to another bus master.

HOLD

Bus Hold Request (Active High; Input)

Allows another bus master to request control of the local bus.

$\overline{IIBEN}$

I/O Instruction Break Enable (Active Low; Input)

Enables the I/O instruction break feature. $\overline{IIBEN}$ has an internal pull-up resistor. Once $\overline{IIBEN}$ is driven active Low, the internal pull-up resistor is disabled until the CPU is reset.

INTR

Interrupt Request (Active High; Input)

A maskable input that signals the Am386DXLV microprocessor to suspend execution of the current program and execute an interrupt acknowledge function.

$\overline{LOCK}$

Bus Lock (Active Low; Output)

A bus cycle definition pin that indicates that other system bus masters are denied access to the system bus while it is active.

M/$\overline{IO}$

Memory I/O (Output)

A bus cycle definition pin that distinguishes memory cycles from input/output cycles.

$\overline{NA}$

Next Address (Active Low; Input)

Used to request address pipelining.

NC

No Connect

Should always remain unconnected. Connection of an NC pin may cause the processor to malfunction or be incompatible with future steppings of the Am386DXLV microprocessor.

NMI

Non-Maskable Interrupt Request (Active High; Input)

A non-maskable input that signals the Am386DXLV microprocessor to suspend execution of the current program and execute an interrupt acknowledge function.

PEREQ

Processor Extension Request (Active High; Input)

Indicates that the processor extension has data to be transferred by the Am386DXLV microprocessor. PEREQ has an internal pull-down resistor.

$\overline{READY}$

Bus Ready (Active Low; Input)

Terminates the bus cycle initiated by $\overline{ADS}$.

RESET

Reset (Active High; Input)

Suspends any operation in progress and places the Am386DXLV microprocessor in a known reset state.

$\overline{SMI}$

System Management Interrupt (Active Low; Input/Output)

A Non-Maskable Interrupt (NMI) pin that signals the Am386DXLV microprocessor to suspend execution and enter System Management Mode. $\overline{SMI}$ has a dynamic pull-up resistor that is disabled when the processor is in SMM. $\overline{SMI}$ is not three-stated during hold acknowledge bus cycles.

$\overline{SMIADS}$

SMI Address Status (Active Low, Three-State; Output)

Indicates that a valid bus cycle definition and address (W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{BE3}$–$\overline{BE0}$, and A31–A2) are being driven at the Am386DXLV microprocessor pins while in System Management Mode. Bus cycles initiated by $\overline{SMIADS}$ must be terminated by $\overline{SMIRDY}$.

$\overline{SMIRDY}$

SMI Ready (Active Low; Input)

This input terminates the current bus cycle to the SMM address space in the same manner the $\overline{READY}$ pin does for the Normal Mode address space. $\overline{SMIRDY}$ has an internal pull-up resistor. $\overline{SMIRDY}$ must not be tied to $\overline{READY}$.

V_{CC}

System Power (Active High; Input)

Provides the DC supply input.

V_{SS}

System Ground (Input)

Provides 0-V connection from which all inputs and outputs are measured.

W/$\overline{R}$

Write/Read (Output)

A bus cycle definition pin that distinguishes write cycles from read cycles.

ABSOLUTE MAXIMUM RATINGS

Storage Temperature . . . . . −65°C to +150°C
Ambient Temperature Under Bias . . −65°C to +125°C

Stresses above those listed may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to ABSOLUTE MAXIMUM RATINGS for extended periods may affect device reliability.

OPERATING RANGES

Supply Voltage with Respect to V_{SS} . . −0.5 V to +7 V
Voltage on Other Pins . . . . . −0.5 V to V_{CC} +0.5 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over COMMERCIAL Operating Ranges

V_{CC} = 3.0 V to 3.6 V; T_{CASE} = 0°C to +100°C

			Final		
Symbol	**Parameter Description**	**Notes**	**Min**	**Max**	**Unit**
V_{IL}	Input Low Voltage	(Note 1)	−0.3	0.8	V
V_{IH}	Input High Voltage		2.0	V_{CC} + 0.3	V
V_{ILC}	CLK2 Input Low Voltage	(Note 1)	−0.3	0.8	V
V_{IHC}	CLK2 Input High Voltage (25 MHz)		2.4	V_{CC} + 0.3	V
V_{OL}	Output Low Voltage	(Note 5)			
	I_{OL} = 0.5 mA: A31–A2, D31–D0			0.2	V
	I_{OL} = 0.5 mA: $\overline{BE3}$–$\overline{BE0}$, W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, $\overline{SMIADS}$, HLDA, $\overline{SMI}$			0.2	V
	I_{OL} = 2 mA: A31–A2, D31–D0			0.45	V
	I_{OL} = 2.5 mA: $\overline{BE3}$–$\overline{BE0}$, W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, $\overline{SMIADS}$, HLDA, $\overline{SMI}$			0.45	V
V_{OH}	Output High Voltage	(Note 5)			
	I_{OH} = 0.1 mA: A31–A2, D31–D0	(Note 6)	V_{CC} − 0.2		V
	I_{OH} = 0.1 mA: $\overline{BE3}$–$\overline{BE0}$, W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, $\overline{SMIADS}$, HLDA, $\overline{SMI}$		V_{CC} − 0.2		V
	I_{OH} = 0.5 mA: A31–A2, D31–D0		V_{CC} − 0.45		V
	I_{OH} = 0.5 mA: $\overline{BE3}$–$\overline{BE0}$, W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, $\overline{SMIADS}$, HLDA, $\overline{SMI}$		V_{CC} − 0.45		V
I_{LI}	Input Leakage Current (All pins except $\overline{BS16}$, PEREQ, $\overline{IIBEN}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{ERROR}$, $\overline{SMI}$, and $\overline{SMIRDY}$)	0 V ≤ V_{IN} ≤ V_{CC} (Note 7)		±10	µA
I_{IH}	Input Leakage Current (PEREQ Pin)	V_{IH} = V_{CC} − 0.1 V V_{IH} = 2.4 V (Note 2)		300 200	µA
I_{IL}	Input Leakage Current ($\overline{BS16}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{ERROR}$, $\overline{SMI}$, $\overline{IIBEN}$, and $\overline{SMIRDY}$)	V_{IL} = 0.1 V V_{IL} = 0.45 V (Note 3)		−300 −200	µA
I_{LO}	Output Leakage Current	0.1 V ≤ V_{OUT} ≤ V_{CC}		±15	µA
I_{CC}	Supply Current (Note 8) CLK2 = 40 MHz: Oper. Freq. 20 MHz CLK2 = 50 MHz: Oper. Freq. 25 MHz	V_{CC} = 3.3 V I_{CC} Typ = 80 I_{CC} Typ = 95		V_{CC} = 3.6 V 95 115	 mA mA
I_{CCSB}	Standby Current (Note 8)	I_{CCSB} Typ = 10 µA		150	µA
C_{IN}	Input or I/O Capacitance	F_C = 1 MHz (Note 4)		10	pF
C_{OUT}	Output Capacitance	F_C = 1 MHz (Note 4)		12	pF
C_{CLK}	CLK2 Capacitance	F_C = 1 MHz (Note 4)		20	pF

Notes: 1. The Min value, −0.3, is not 100% tested.
2. PEREQ input has an internal pull-down resistor.
3. $\overline{BS16}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{ERROR}$, $\overline{SMI}$, $\overline{SMIRDY}$, and $\overline{IIBEN}$ inputs each have an internal pull-up resistor.
4. Not 100% tested.
5. Outputs are CMOS and will pull rail-to-rail if the load is not resistive.
6. V_{OH} is only valid for $\overline{SMI}$ when exiting SMM for two CLK2 cycles.
7. $\overline{SMI}$ and $\overline{IIBEN}$ leakage Low will be I_{LI} when pull-up is inactive and I_{IL} when pull-up is active.
8. Inputs are at either V_{CC} or V_{SS}.

ABSOLUTE MAXIMUM RATINGS

Storage Temperature ..... –65°C to +150°C
Ambient Temperature Under Bias . . –65°C to +125°C

Stresses above those listed may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to ABSOLUTE MAXIMUM RATINGS for extended periods may affect device reliability.

OPERATING RANGES

Supply Voltage with Respect to V_{SS} ..... –0.5 V to +7 V
Voltage on Other Pins ..... –0.5 V to V_{CC} +0.5 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over COMMERCIAL Operating Ranges

V_{CC} = 3.6 V to 5.5 V; T_{CASE} = 0°C to +100°C

			Final		
Symbol	**Parameter Description**	**Notes**	**Min**	**Max**	**Unit**
V_{IL}	Input Low Voltage	(Note 1)	–0.3	0.8	V
V_{IH}	Input High Voltage		2.0	V_{CC} + 0.3	V
V_{ILC}	CLK2 Input Low Voltage	(Note 1)	–0.3	0.8	V
V_{IHC}	CLK2 Input High Voltage (33 MHz)		2.7	V_{CC} + 0.3	V
V_{OL}	Output Low Voltage	(Note 5)			
	I_{OL} = 4 mA: A31–A2, D31–D0			0.45	V
	I_{OL} = 5 mA: $\overline{BE3}$–$\overline{BE0}$, W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, $\overline{SMIADS}$, HLDA, $\overline{SMI}$			0.45	V
V_{OH}	Output High Voltage	(Note 5)			
	I_{OH} = 1 mA: A31–A2, D31–D0	(Note 6)	2.4		V
	I_{OH} = 0.9 mA: $\overline{BE3}$–$\overline{BE0}$,W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, $\overline{SMIADS}$, HLDA, $\overline{SMI}$		2.4		V
I_{LI}	Input Leakage Current (All pins except $\overline{BS16}$, PEREQ, $\overline{IIBEN}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{SMI}$, $\overline{SMIRDY}$, and $\overline{ERROR}$)	0 V ≤ V_{IN} ≤ V_{CC} (Note 7)		±15	µA
I_{IH}	Input Leakage Current (PEREQ Pin)	V_{IH} = 2.4 V (Note 2)		200	µA
I_{IL}	Input Leakage Current ($\overline{BS16}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{SMI}$, $\overline{SMIRDY}$, $\overline{IIBEN}$, and $\overline{ERROR}$)	V_{IL} = 0.45 (Note 3)		–400	µA
I_{LO}	Output Leakage Current	0.45 V ≤ V_{OUT} ≤ V_{CC}		±15	µA
I_{CC}	Supply Current (Note 8)	V_{CC} = 5.0 V		V_{CC} = 5.5 V	
	CLK2 = 40 MHz: Oper. Freq. 20 MHz	I_{CC} Typ = 130		155	mA
	CLK2 = 50 MHz: Oper. Freq. 25 MHz	I_{CC} Typ = 160		190	mA
	CLK2 = 66 MHz: Oper. Freq. 33 MHz	I_{CC} Typ = 210		245	mA
I_{CCSB}	Standby Current (Note 8)	I_{CCSB} Typ = 20 µA		150	µA
C_{IN}	Input or I/O Capacitance	F_C = 1 MHz (Note 4)		10	pF
C_{OUT}	Output Capacitance	F_C = 1 MHz (Note 4)		12	pF
C_{CLK}	CLK2 Capacitance	F_C = 1 MHz (Note 4)		20	pF

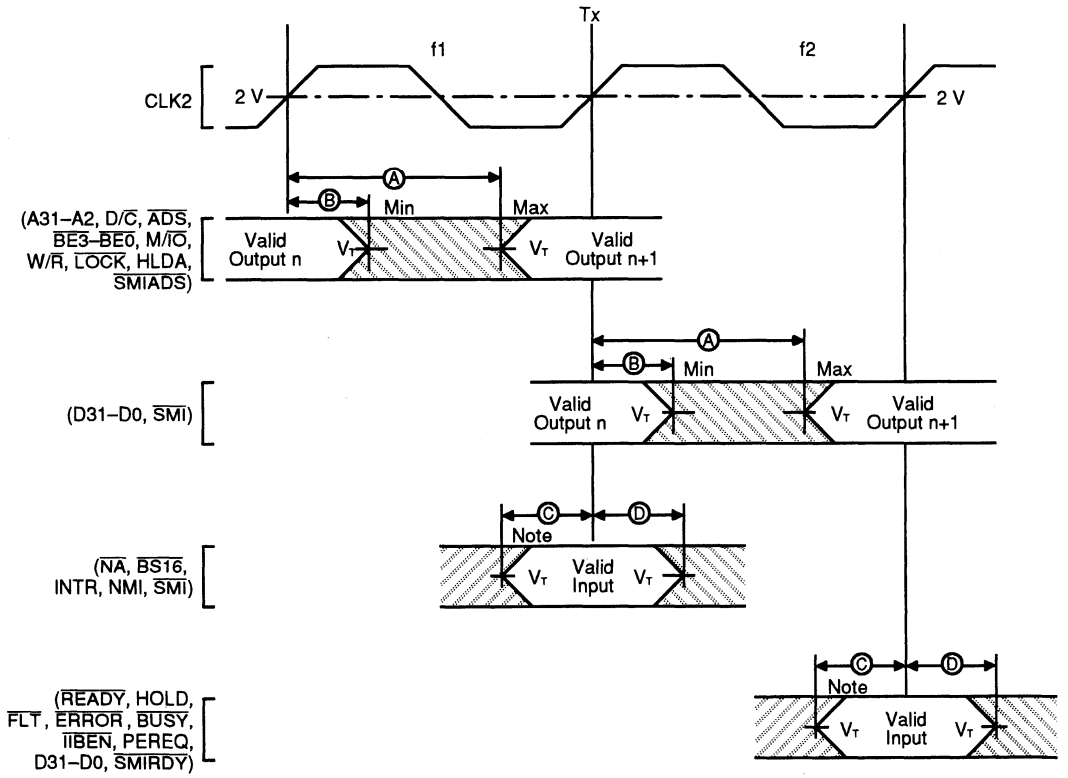
Notes: 1. The Min value, –0.3, is not 100% tested.
2. PEREQ input has an internal pull-down resistor.
3. $\overline{BS16}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{ERROR}$, $\overline{SMI}$, $\overline{SMIRDY}$, and $\overline{IIBEN}$ inputs each have an internal pull-up resistor.
4. Not 100% tested.
5. Outputs are CMOS and will pull rail-to-rail if the load is not resistive.
6. V_{OH} is only valid for $\overline{SMI}$ when exiting SMM for two CLK2 cycles.
7. $\overline{SMI}$ and $\overline{IIBEN}$ leakage Low will be I_{LI} when pull-up is inactive and I_{IL} when pull-up is active.
8. Inputs are at either V_{CC} or V_{SS}.

SWITCHING CHARACTERISTICS

The switching characteristics consist of output delays, input setup requirements, and input hold requirements. All characteristics are relative to the CLK2 rising edge crossing the 2.0-V level.

Switching characteristic measurement is defined in Figure 2. Inputs must be driven to the voltage levels indicated by this diagram. Am386DXLV CPU output delays are specified with minimum and maximum limits measured as shown. The minimum Am386DXLV microprocessor delay times are hold times provided to external circuitry. Am386DXLV microprocessor input setup and hold time are specified as minimums, defining the smallest acceptable sampling window. Within the sampling window, a synchronous input signal must be stable for correct Am386DXLV microprocessor operation.

Outputs W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{BE3}$–$\overline{BE0}$, $\overline{ADS}$, A31–A2, HLDA, and $\overline{SMIADS}$ only change at the beginning of phase one. D31–D0 (write cycles) and $\overline{SMI}$ only change at the beginning of phase two. The $\overline{READY}$, HOLD, $\overline{IIBEN}$, $\overline{BUSY}$, $\overline{ERROR}$, PEREQ, $\overline{FLT}$, D31–D0, and $\overline{SMIRDY}$ (read cycles) inputs are sampled at the beginning of phase one. The $\overline{NA}$, $\overline{BS16}$, INTR, NMI, and $\overline{SMI}$ inputs are sampled at the beginning of phase two.



Legend: A—Maximum Output Delay Spec
B—Minimum Output Delay Spec
C—Minimum Input Setup Spec
D—Minimum Input Hold Spec

Notes: 1. Input waveforms have $tr \leq 2.0$ ns from 0.8 V to 2.0 V.
2. $V_T = 1.0$ V at $V_{CC} \leq 3.6$; 1.5 V at $V_{CC} > 3.6$.

16306B–003

Figure 2. Drive Levels and Measurement Points

SWITCHING CHARACTERISTICS over operating range at 25 MHz

V_{CC} = 3.0 V to 5.5 V; T_{CASE} = 0°C to +100°C

No.	Parameter Description	Notes	Ref Figures	Final Min	Final Max	Unit
	Operating Frequency	Half of CLK2 Freq		0	25	MHz
1	CLK2 Period		4	20		ns
2	CLK2 High Time	at V_{IHC}	4	4		ns
3	CLK2 Low Time	at 0.8 V	4	5		ns
4	CLK2 Fall Time (Note 3)	2.4 V to 0.8 V	4		7	ns
5	CLK2 Rise Time (Note 3)	0.8 V to 2.4 V	4		7	ns
6	A31–A2 Valid Delay	C_L = 50 pF	3, 6	4	17	ns
7	A31–A2 Float Delay	(Note 1)	13	4	30	ns
8	$\overline{BE3}$–$\overline{BE0}$, $\overline{LOCK}$ Valid Delay	C_L = 50 pF	3, 6	4	17	ns
9	$\overline{BE3}$–$\overline{BE0}$, $\overline{LOCK}$ Float Delay	(Note 1)	13	4	30	ns
10	W/$\overline{R}$, M/$\overline{IO}$, D/$\overline{C}$, $\overline{ADS}$ Valid Delay	C_L = 50 pF	3, 6	4	17	ns
10s	$\overline{SMIADS}$ Valid Delay	C_L=50pF	3, 6	4	25	ns
11	W/$\overline{R}$, M/$\overline{IO}$, D/$\overline{C}$, $\overline{ADS}$ Float Delay	(Note 1)	13	4	30	ns
11s	$\overline{SMIADS}$ Float Delay	(Note 1)	13	4	30	ns
12	D31–D0 Write Data Valid Delay	C_L = 50 pF	6, 7	7	23	ns
12a	D31–D0 Write Data Hold Time	C_L = 50 pF	3, 8	2		ns
13	D31–D0 Float Delay	(Note 1)	13	4	22	ns
14	HLDA Valid Delay	C_L = 50 pF	3, 13	4	22	ns
14f	HLDA Float Delay	(Note 1)	14	4	30	ns
15	$\overline{NA}$ Setup Time		5	5		ns
16	$\overline{NA}$ Hold Time		5	3		ns
17	$\overline{BS16}$ Setup Time		5	5		ns
18	$\overline{BS16}$ Hold Time		5	3		ns
19	$\overline{READY}$ Setup Time		5	9		ns
19s	$\overline{SMIRDY}$ Setup Time		5	9		ns
20	$\overline{READY}$ Hold Time		5	4		ns
20s	$\overline{SMIRDY}$ Hold Time		5	4		ns
21	D31–D0 Read Setup Time		5	7		ns
22	D31–D0 Read Hold Time		5	5		ns
23	HOLD Setup Time		5	9		ns
24	HOLD Hold Time		5	3		ns
25	RESET Setup Time		15	8		ns
26	RESET Hold Time		15	3		ns
27	NMI, INTR Setup Time	(Note 2)	5	6		ns
27s	$\overline{SMI}$ Setup Time		5	6		ns
28	NMI, INTR Hold Time	(Note 2)	5	6		ns
28s	$\overline{SMI}$ Hold Time		5	4		ns
29	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{BEN}$ Setup Time	(Note 2)	5	6		ns
30	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{BEN}$ Hold Time	(Note 2)	5	5		ns
31	$\overline{SMI}$ Valid Delay		5, 13	4	22	ns
32	$\overline{SMI}$ Float Delay	(Note 1)	14	4	30	ns

Notes: 1. Float condition occurs when maximum output current becomes less than I_{LO} magnitude. Float delay is not 100% tested.
2. These inputs are allowed to be asynchronous to CLK2. The setup and hold specifications are given for testing purposes, to assure recognition within a specific CLK2 period.
3. Rise and fall times are not tested.

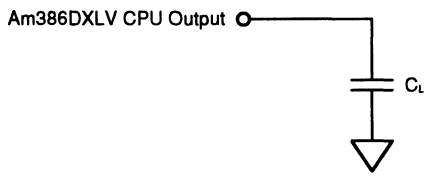
SWITCHING CHARACTERISTICS over operating range at 33 MHz

V_{CC} = 4.5 V to 5.5 V; T_{CASE} = 0°C to +100°C

				Final		
No.	**Parameter Description**	**Notes**	**Ref Figures**	**Min**	**Max**	**Unit**
	Operating Frequency	Half of CLK2 Freq		0	33.3	MHz
1	CLK2 Period		4	15.0		ns
2	CLK2 High Time	at V_{IHC}	4	4.5		ns
3	CLK2 Low Time	at 0.8 V	4	4.5		ns
4	CLK2 Fall Time (Note 3)	2.4 V to 0.8 V	4		4	ns
5	CLK2 Rise Time (Note 3)	0.8 V to 2.4 V	4		4	ns
6	A31–A2 Valid Delay	C_L = 50 pF	3, 6	4	15	ns
7	A31–A2 Float Delay	(Note 1)	13	4	20	ns
8	$\overline{BE3}$–$\overline{BE0}$, $\overline{LOCK}$ Valid Delay	C_L = 50 pF	3, 6	4	15	ns
9	$\overline{BE3}$–$\overline{BE0}$, $\overline{LOCK}$ Float Delay	(Note 1)	13	4	20	ns
10	W/$\overline{R}$, M/$\overline{IO}$, D/$\overline{C}$, $\overline{ADS}$ Valid Delay	C_L = 50 pF	3, 6	4	15	ns
10s	$\overline{SMIADS}$ Valid Delay	C_L=50 pF	2, 6	4	15	ns
11	W/$\overline{R}$, M/$\overline{IO}$, D/$\overline{C}$, $\overline{ADS}$ Float Delay	(Note 1)	13	4	20	ns
11s	$\overline{SMIADS}$ Float Delay	(Note 1)	13	4	20	ns
12	D31–D0 Write Data Valid Delay	C_L = 50 pF	6, 7	7	23	ns
12a	D31–D0 Write Data Hold Time	C_L = 50 pF	3, 8	2		ns
13	D31–D0 Float Delay	(Note 1)	13	4	17	ns
14	HLDA Valid Delay	C_L = 50 pF	3, 13	4	20	ns
14f	HLDA Float Delay	(Note 1)	14	4	20	ns
15	$\overline{NA}$ Setup Time		5	5		ns
16	$\overline{NA}$ Hold Time		5	2		ns
17	$\overline{BS16}$ Setup Time		5	5		ns
18	$\overline{BS16}$ Hold Time		5	2		ns
19	$\overline{READY}$ Setup Time		5	7		ns
19s	$\overline{SMIRDY}$ Setup Time		5	7		ns
20	$\overline{READY}$ Hold Time		5	4		ns
20s	$\overline{SMIRDY}$ Hold Time		5	4		ns
21	D31–D0 Read Setup Time		5	5		ns
22	D31–D0 Read Hold Time		5	3		ns
23	HOLD Setup Time		5	9		ns
24	HOLD Hold Time		5	2		ns
25	RESET Setup Time		15	5		ns
26	RESET Hold Time		15	2		ns
27	NMI, INTR Setup Time	(Note 2)	5	5		ns
27s	$\overline{SMI}$ Setup Time		5	5		ns
28	NMI, INTR Hold Time	(Note 2)	5	5		ns
28s	$\overline{SMI}$ Hold Time		5	4		ns
29	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{IIBEN}$ Setup Time	(Note 2)	5	5		ns
30	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{IIBEN}$ Hold Time	(Note 2)	5	4		ns
31	$\overline{SMI}$ Valid Delay		5, 13	4	17	ns
32	$\overline{SMI}$ Float Delay	(Note 1)	14	4	20	ns

Notes: 1. Float condition occurs when maximum output current becomes less than I_{LO} in magnitude. Float delay is not 100% tested.
2. These inputs are allowed to be asynchronous to CLK2. The setup and hold specifications are given for testing purposes, to assure recognition within a specific CLK2 period.
3. Rise and fall times are not tested.

SWITCHING CHARACTERISTICS (continued)



C_L includes all parasitic capacitances.

15021B–072

Figure 3. AC Test Load

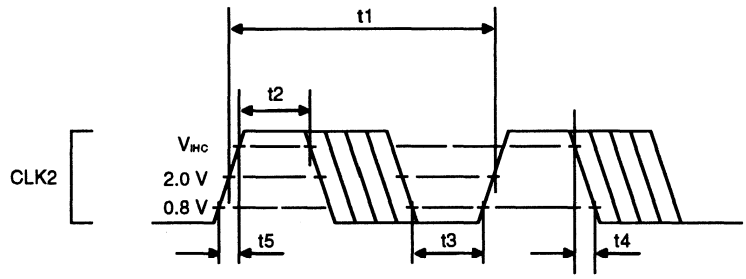


Figure 4. CLK2 Timing

16306B–004

SWITCHING WAVEFORMS

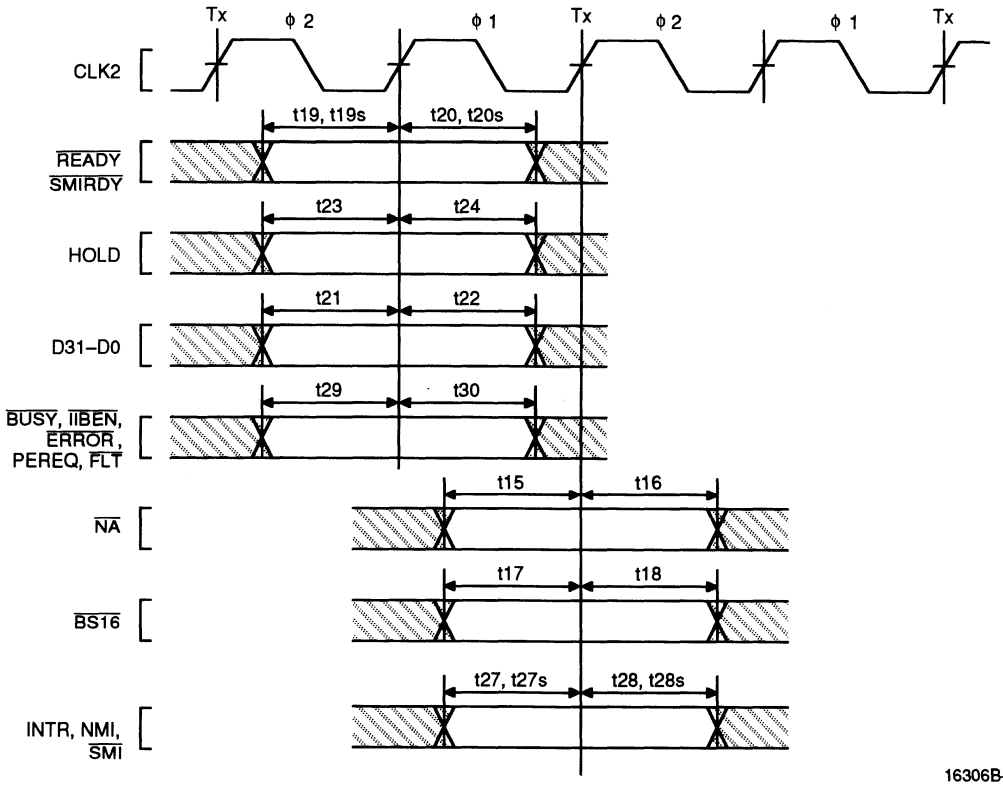


Figure 5. Input Setup and Hold Timing

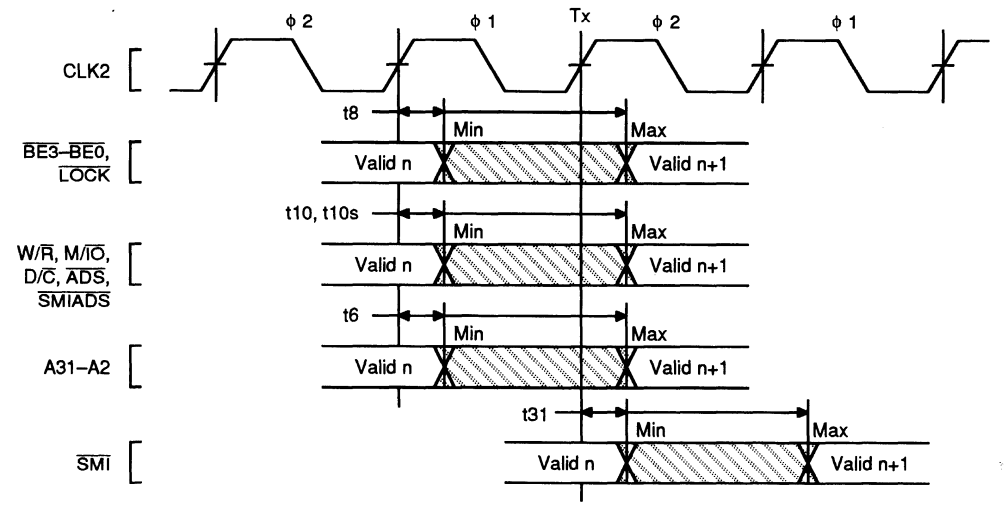
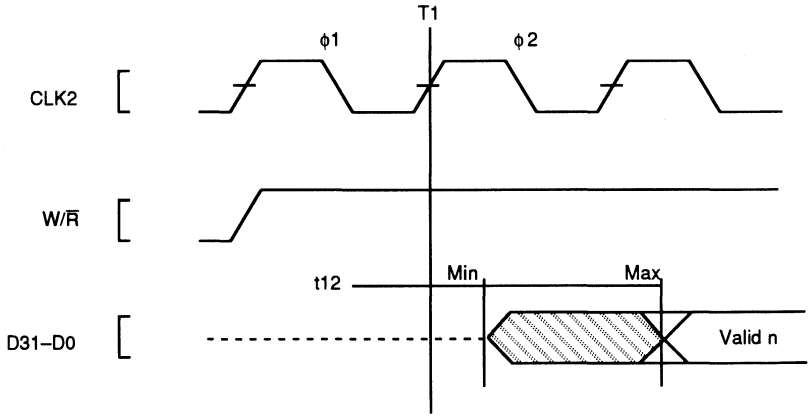
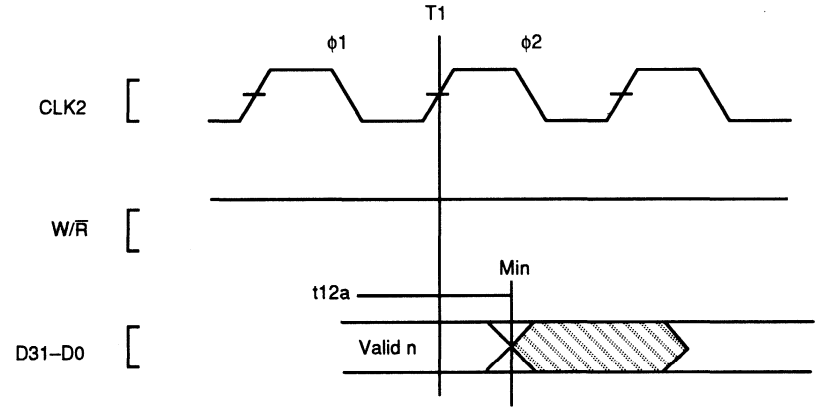


Figure 6. Output Valid Delay Timing



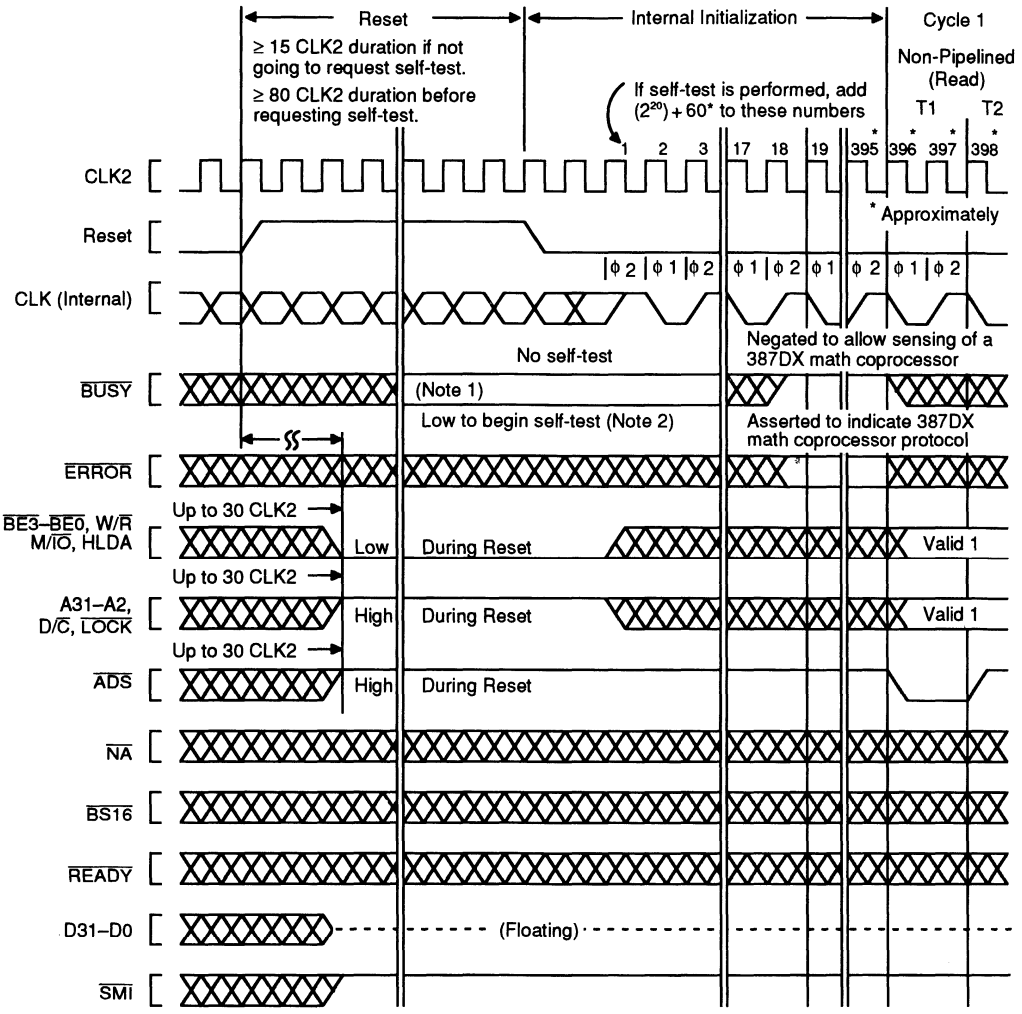
15021B–076

Figure 7. Write Data Valid Delay Timing



15021B–077

Figure 8. Write Data Hold Timing



Notes: 1. BUSY should be held stable for eight CLK2 periods before and after the CLK2 period in which RESET falling edge occurs.
2. If self-test is requested, the Am386DXLV microprocessor outputs remain in their reset state as shown here.

16306B–007

Figure 9. Bus Activity from Reset Until First Code Fetch

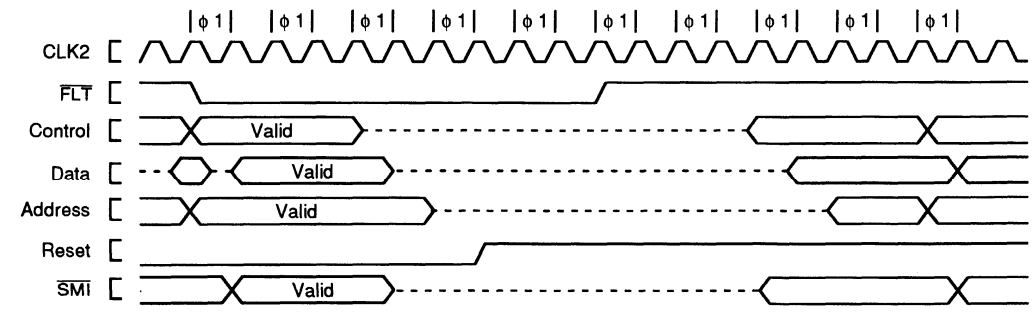
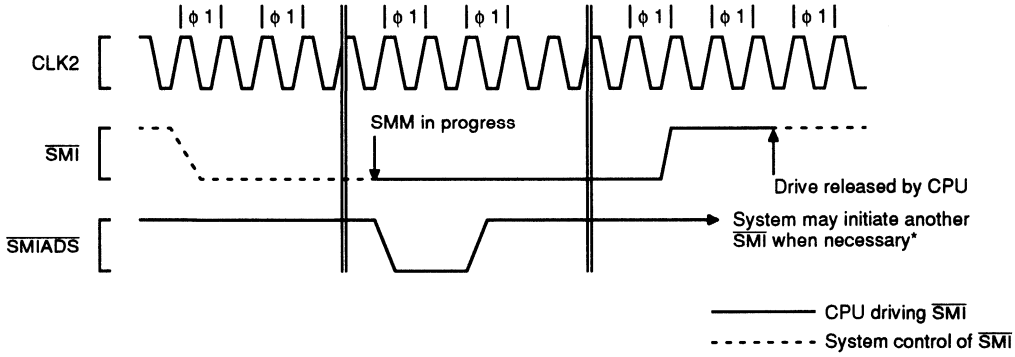


Figure 10. Entering and Exiting FLT

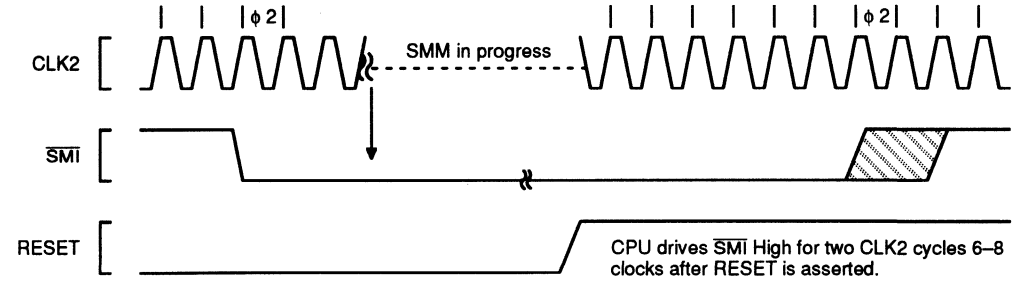
16306B–008



*Once initiated, the system must hold SMI Low until the first SMIADS. At this time, the system releases control of SMI and can not drive SMI until three CLK2 cycles after the CPU drives SMI High. CPU will drive SMI High for two CLK2 cycles. The additional clock allows the CPU to completely release SMI and prevents any driver overlap.

16306B–009

Figure 11. Initiating and Exiting SMM



16306B–010

Figure 12. RESET and SMI

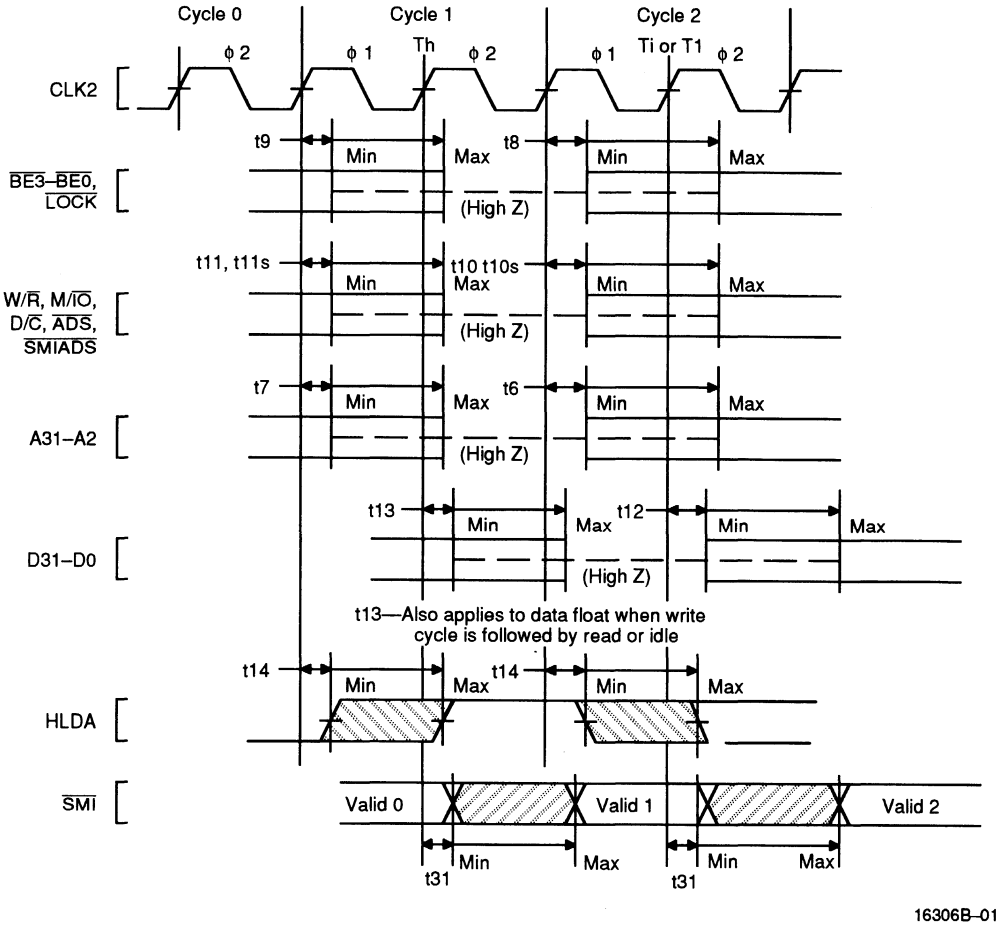


Figure 13. Output Float Delay and HLDA and $\overline{SMI}$ Valid Delay Timing

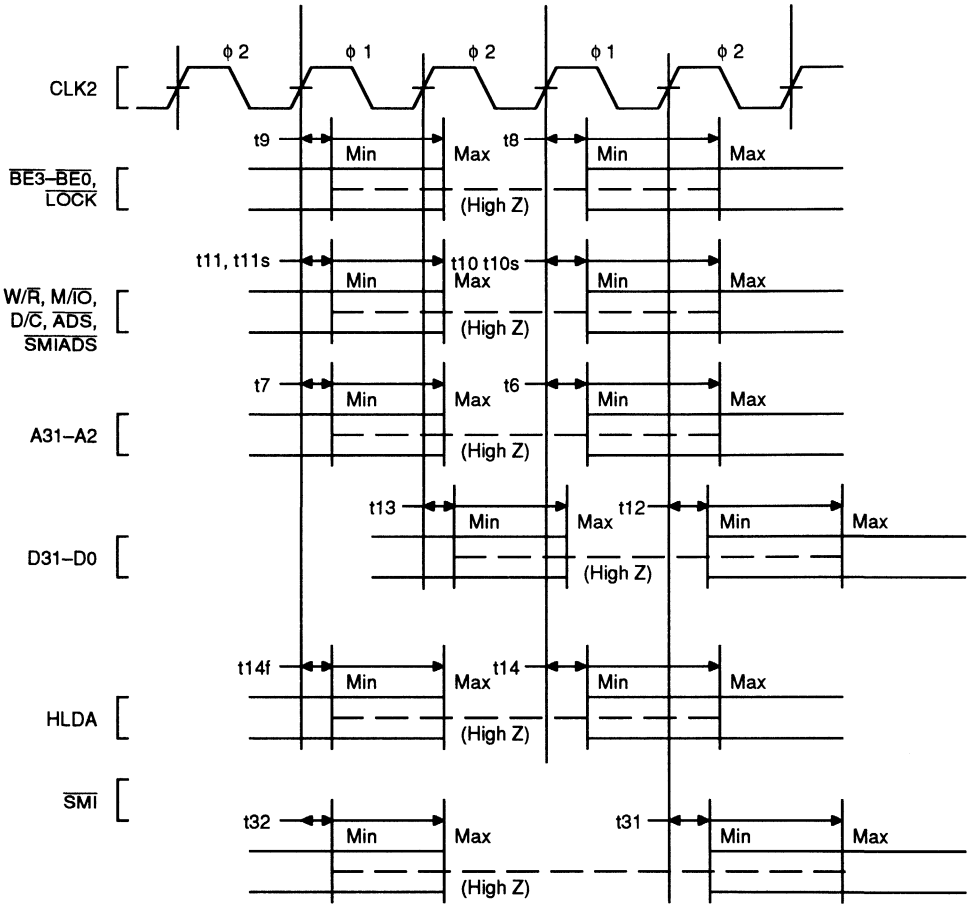
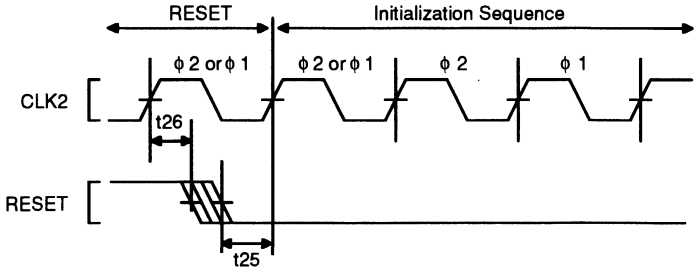


Figure 14. Output Float Delay Entering and Exiting FLT

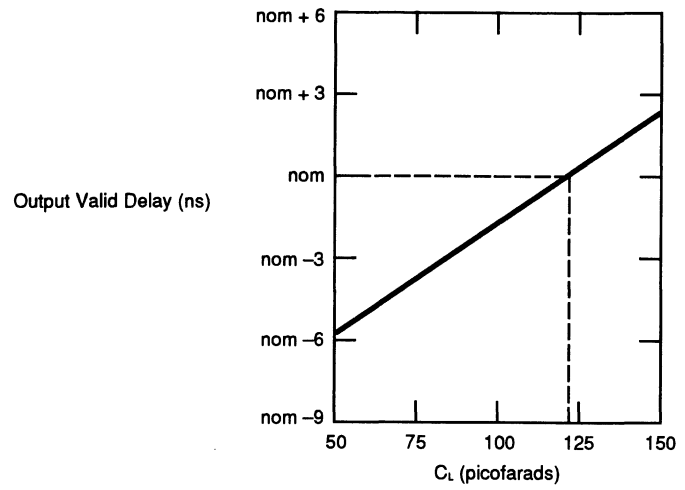
16306B–012



The second internal processor phase following RESET High-to-Low transition (provided t25 and t26 are met) is φ2.

15021B–084

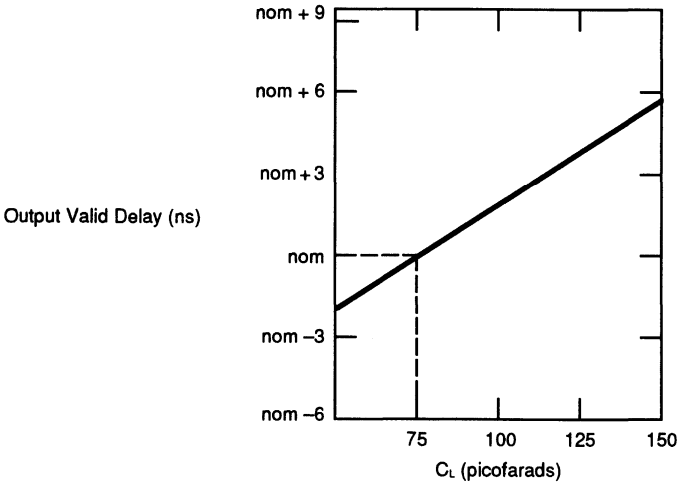
Figure 15. RESET Setup and Hold Timing and Internal Phase



Note: This graph will not be linear outside the C_L range shown.

15021B–079

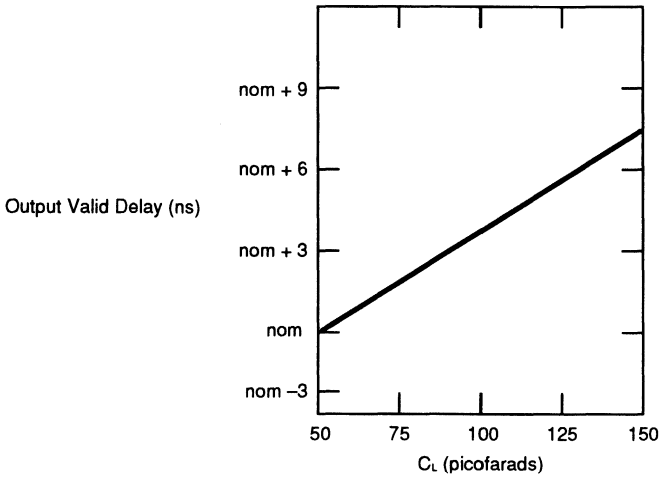
Figure 16. Typical Output Valid Delay Versus Load Capacitance at Maximum Operating Temperature (C_L=120 pF)



Note: This graph will not be linear outside the C_L range shown.

15021B–080

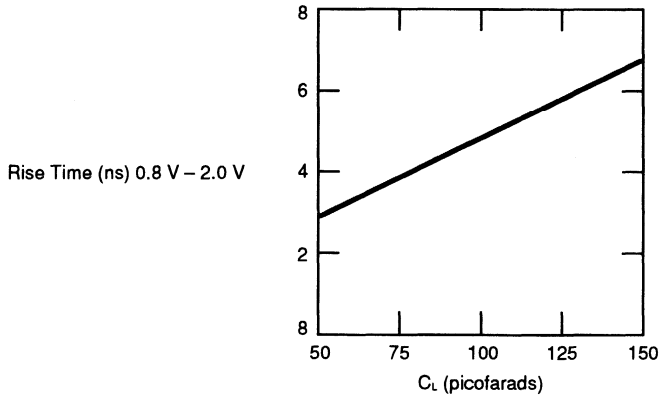
Figure 17. Typical Output Valid Delay Versus Load Capacitance at Maximum Operating Temperature (C_L = 75 pF)



Note: This graph will not be linear outside the C_L range shown.

15021B–081

Figure 18. Typical Output Valid Delay Versus Load Capacitance at Maximum Operating Temperature (C_L = 50 pF)



Note: This graph will not be linear outside the C_L range shown.

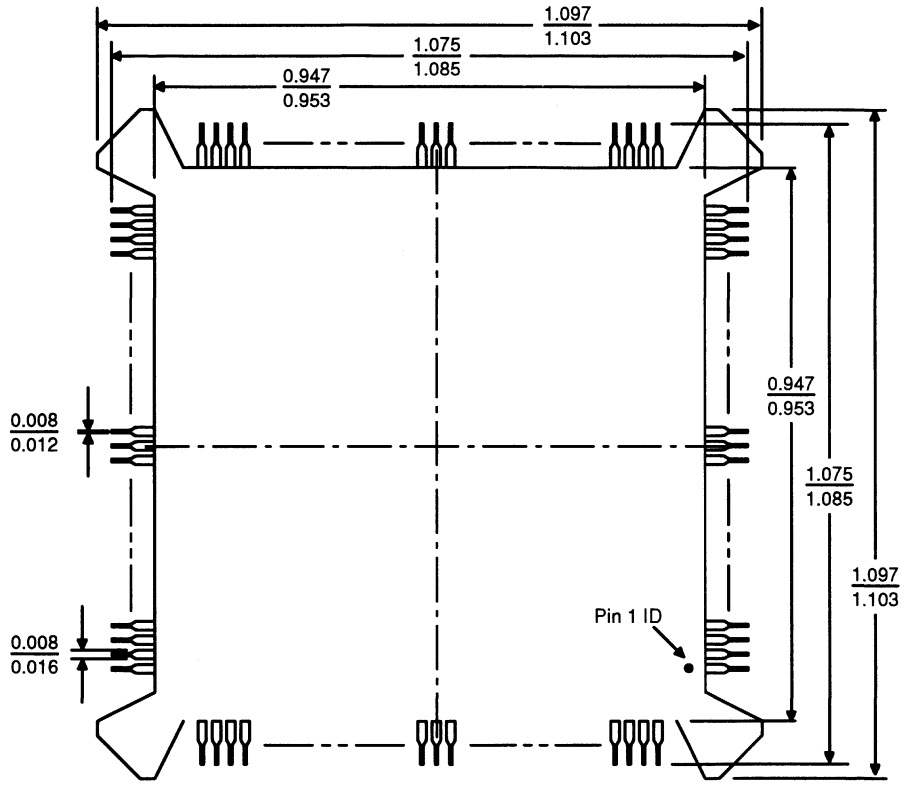
15021B–082

Figure 19. Typical Output Rise Time Versus Load Capacitance at Maximum Operating Temperature

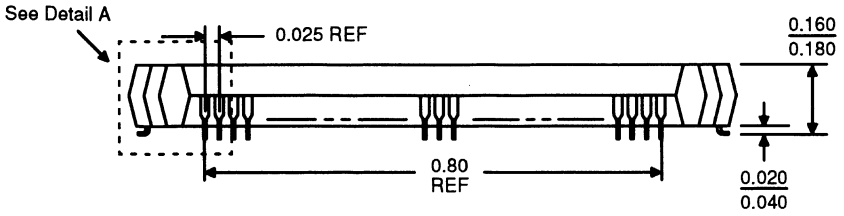
PHYSICAL DIMENSIONS

For reference only. All dimensions are in inches, except for the outer ring on PQB 132 which is in millimeters. BSC is an ANSI standard for Basic Space Centering.

PQB 132—Plastic Quad Flat Pack; Trimmed and Formed (all measurements in inches)



Top View

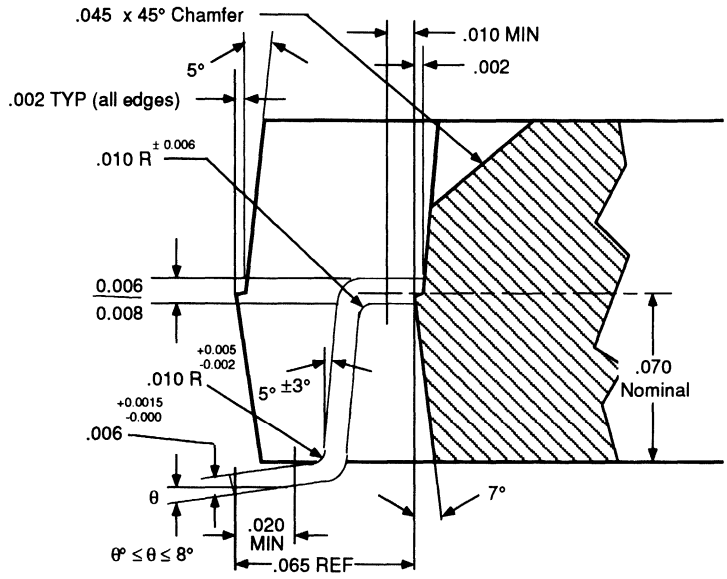


Side View

20011A
CL85
03/17/93 SG

PHYSICAL DIMENSIONS (continued)

PQB 132—Plastic Quad Flat Pack; Trimmed and Formed (continued)



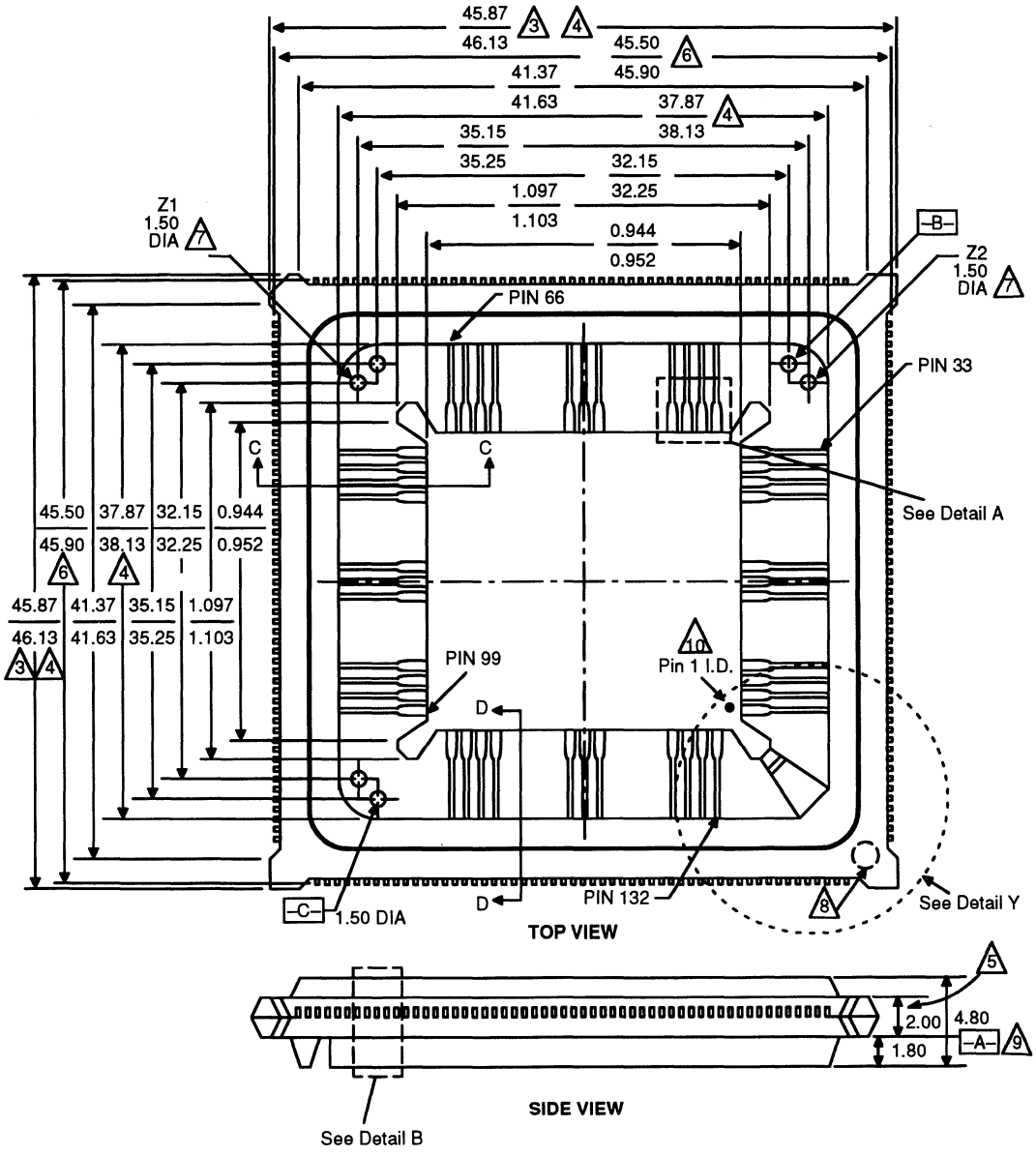
Detail A

Notes:

1. All dimensions are in inches unless otherwise specified.
2. Dimensions do not include mold protrusion.
3. Coplanarity of all leads will be within 0.004 inches measured from the seating plan. Coplanarity is measured per specification 06–500.
4. Deviation from Lead-tip true position shall be within ±0.003 inches.
5. Half span (center of package to lead-tip) shall be within ±0.0085 inches.

PHYSICAL DIMENSIONS (continued)

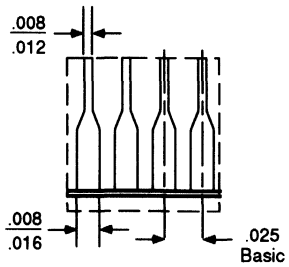
PQB 132—Plastic Quad Flat Pack with Molded Carrier Ring
(Inner device measured in inches; outer ring measured in millimeters)



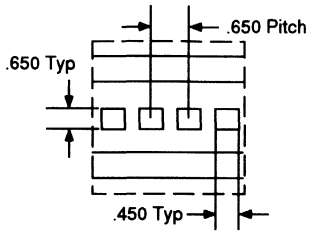
20008A
CJ83
03/17/93 SG

PHYSICAL DIMENSIONS (continued)

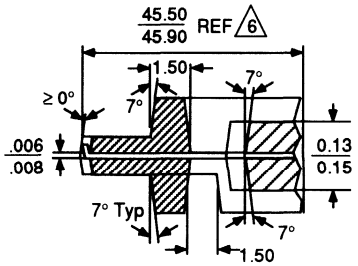
PQB 132—Plastic Quad Flat Pack with Molded Carrier Ring (continued)



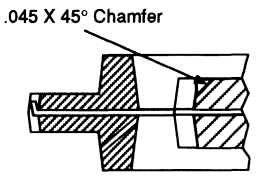
Detail A



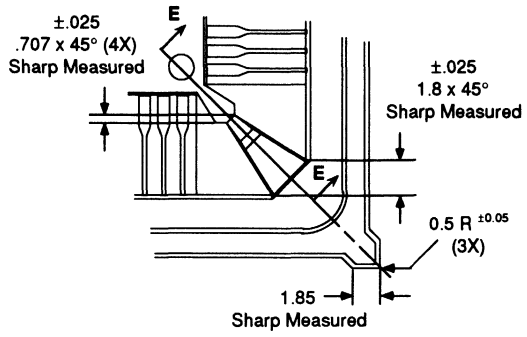
Detail B



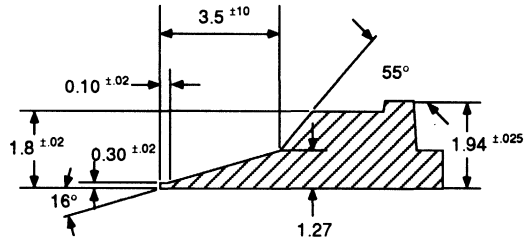
Section C-C



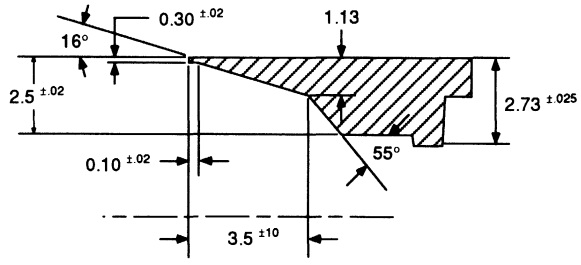
Section D-D



Detail Y



Top Gate



Bottom Gate

Section E-E

PHYSICAL DIMENSIONS (continued)

PQB 132 — Plastic Quad Flat Pack with Molded Carrier Ring

Notes:

1. All dimensions and tolerances conform to ANSI Y14.5M–1982.
2. Controlling dimensions: package is measured in inches and ring is measured in millimeters.
3. These dimensions do not include mold protrusion. Allowable mold protrusion is 0.2 mm per side.
4. These dimensions include mold mismatch and are measured at the parting line.
5. Dimensions are centered about centerline of lead material.
6. These dimensions are from the outside edge to the outside edge of the test points.
7. There are six locating holes in the ring. –B– and –C– datum holes are used for trim form and excise of the molded package only. Holes Z1 and Z2 are used for electrical testing only.
8. This area is reserved for vacuum pickup on each of the four corners of the ring and must be flat within 0.025 mm. No ejector pins in this area.
9. Datum –A– surface for seating in socket applications.
10. Pin one orientation with respect to carrier ring as indicated.

FINAL

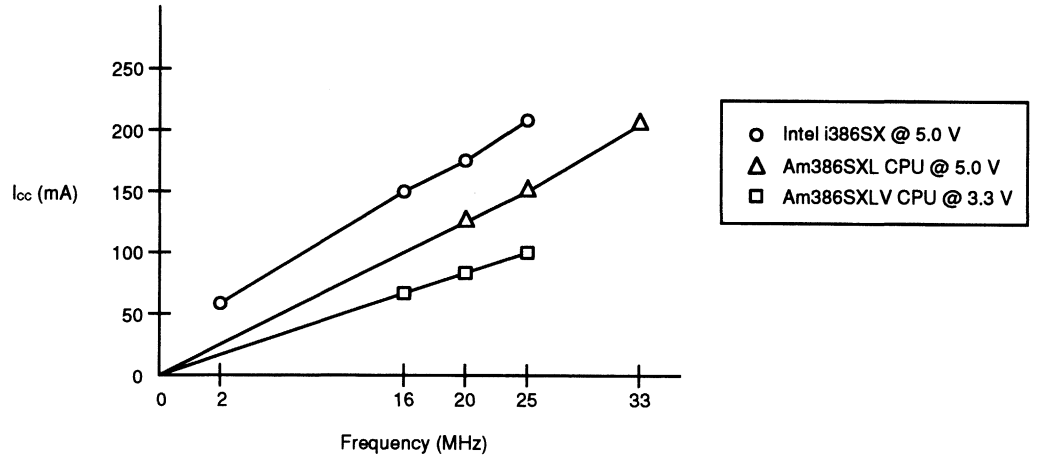
Am386SXLV

High-Performance, Low-Voltage, 32-Bit Microprocessor with 16-Bit Data Bus

Advanced Micro Devices

DISTINCTIVE CHARACTERISTICS

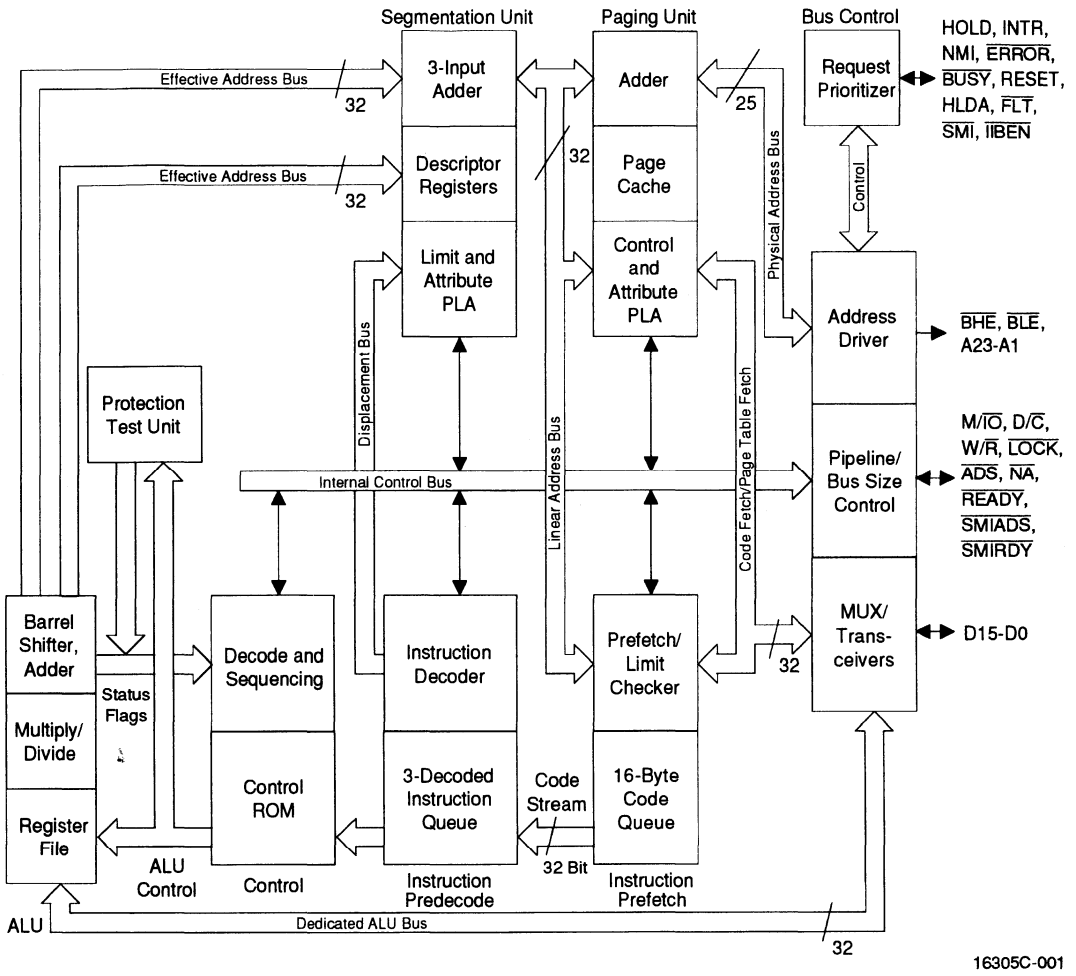
- **Operating range 3.0 V to 5.5 V—Ideal for notebook PC designs**
  - —25-MHz operating frequency for 3.0 V–5.5 V
  - —33-MHz operating frequency for 4.5 V–5.5 V
  - —Twice the battery life of existing 5-V designs
  - —Wide range of chipsets and other logic available for 3-V systems with support for Standby Mode operation
  - —True static design for long battery life
  - —Power consumption 75% lower than Intel i386SX, 65% lower than Am386SXL microprocessor
  - —Performance on demand (0 to 25 MHz)
- **System Management Mode (SMM) for system and power management**
  - —System Management Interrupt (SMI) for power management independent of processor operating mode and operating system
  - —SMI coupled with I/O instruction break feature provides transparent power off and auto resume of peripherals which may not be "power aware"
  - —SMI is non-maskable and has higher priority than Non-Maskable Interrupt (NMI)
  - —Automatic save and restore of the microprocessor state
  - —Wide range of chipsets supporting SMM available to allow product differentiation
- **Lower heat dissipation for fanless systems**
- **"Float" input to facilitate system debug and test**
- **Compatible with 386SX systems and software**
- **Supports 387SX-compatible math coprocessors**
- **100-pin PQFP package with optional protective ring for better lead coplanarity**
- **AMD advanced 0.8 micron CMOS technology**



Note: Inputs at V_{CC} or V_{SS}.

Typical Power Consumption

BLOCK DIAGRAM



GENERAL DESCRIPTION

The Am386SXLV microprocessor is a low-voltage, true static implementation of the Intel i386SX microprocessor. With the operating range of 3.0 V to 5.5 V, the Am386SXLV CPU is ideal for both desktop and battery-powered notebook personal computers. For desktop PCs, this device offers lower heat dissipation, allowing system designers to remove or reduce the size and cost of the cooling fan. The Am386SXLV microprocessor operates at a maximum speed of 25 MHz from 3.0 V–5.5 V and at maximum speed of 33 MHz from 4.5 V–5.5 V.

The Am386SXLV microprocessor's lower operating voltage and true static design enables longer battery life and/or lower weight for notebook applications. At 20 MHz, this device has 60% lower operating Icc than the Intel i386SX. Lowering typical operating voltage from 5.0 V to 3.3 V doubles the battery life. Standby Mode allows the Am386SXLV microprocessor to be clocked down to 0 MHz (DC) and retain full register contents. In Standby Mode, typical current draw is less than 0.01 mA, a greater than 1000X reduction in power consumption versus the Intel i386SX.

The Am386SXLV microprocessor is available in a small footprint 100-pin Plastic Quad Flat Pack (PQFP) package. This package may be shipped in an optional protective ring for better lead protection during shipping.

Additionally, the Am386SXLV microprocessor comes with System Management Mode (SMM) for system and power management. SMI (System Management Interrupt) is a non-maskable, higher priority interrupt than NMI and has its own code space (1 Mb in Real Mode and 16 Mb in Protected Mode). SMI can be coupled with the I/O instruction break feature to implement transparent power managment of peripherals. SMM can be used by system designers to implement system and power management code independent of the operating system or the processor mode.

The Am386SXLV microprocessor incorporates a float pin that places all outputs in a three-state mode to facilitate board test and debug.

FUNCTIONAL DESCRIPTION

Benefits of Lower Operating Voltage

The Am386SXLV microprocessor has an operating voltage range of 3.0 V to 5.5 V. Low voltage allows for lower operating power consumption, longer battery life, and/or smaller batteries for notebook applications.

Because power is proportional to the square of the voltage, reduction of the supply voltage from 5.0 V to 3.3 V reduces power consumption by 56%. This directly translates to a doubling of battery life for portable applications. Lower power consumption can also be used to reduce the size and weight of the battery. Thus, 3.3-V designs facilitate a reduction in the form factor.

A lower operating voltage results in a reduction of I/O voltage swings. This reduces noise generation and provide a less hostile environment for board design. Lower Operating Voltage also reduces electromagnetic radiation noise and makes to obtain FCC approval easier to obtain.

SMM—System Management Mode

The Am386SXLV microprocessor has a new System Management Mode (SMM) for system and power management. This mode consists of two features: System Management Interrupt (SMI) and I/O instruction break.

SMI—System Management Interrupt

SMI is implemented by using special bus interface pins. This interrupt method can be used to perform system management functions such as power management independent of processor operating mode (Real, Protected, or Virtual 8086 Modes).

SMI can also be invoked in software. This allows system software to communicate with SMI power management code. In addition, the UMOV instruction allows data transfers between SMI and normal system memory spaces.

Activating the $\overline{SMI}$ pin invokes a sequence that saves the operating state of the processor into a separate SMM memory space, independent of the main system memory. After the state is saved, the processor is forced into Real Mode and begins execution at address FFFFF0h in the SMM memory space where a far jump to the SMM code is executed. This Real Mode code can perform its system management function and then resume execution of the normal system software by executing an RES3 instruction which will reload the saved processor state and continue execution in the main system memory space. See Figure 1 for a general flowchart of an SMM operation.

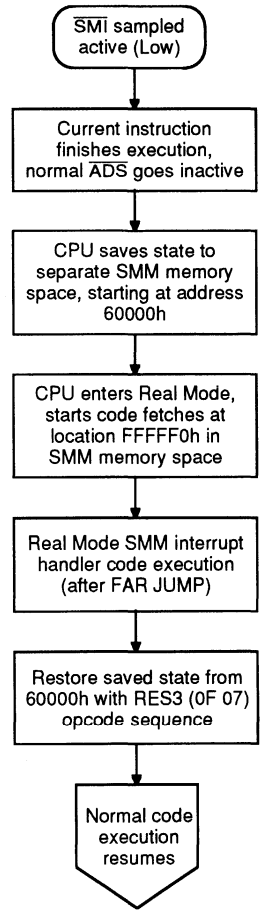


Figure 1. SMM Flow

CPU Interface—Pin Functions

The CPU interface for SMM consists of three pins dedicated to the SMI function. One pin, $\overline{SMI}$, is the new interrupt input. The other two pins, $\overline{SMIADS}$ and $\overline{SMIRDY}$, provide the control signals necessary for the separate SMM mode memory space.

Description of SMM Operation

The execution of a System Management Interrupt has four distinct phases: the initiation of the interrupt via $\overline{SMI}$, a processor state save, execution of the SMM

interrupt code, and a processor state restore (to resume normal operation).

Interrupt Initiation

A System Management Interrupt is initiated by the driving of a synchronous, active Low pulse on the $\overline{SMI}$ pin until the first $\overline{SMIADS}$ is asserted. This pulse period will ensure recognition of the interrupt. The CPU drives the $\overline{SMI}$ pin active after the completion of the current operation (active bus cycle, instruction execution, or both). The active drive of the pin by the CPU is released at the end of the interrupt routine following the last register read of the saved state. The CPU drives $\overline{SMI}$ High for two CLK2 cycles prior to releasing the drive of $\overline{SMI}$.

An SMI cannot be masked off by the CPU, and it will always be recognized by the CPU, regardless of operating modes. This includes the Real, Protected, and Virtual-8086 Modes of the processor.

While the CPU is in SMM, a bus hold request via the HOLD pin is granted. The HLDA pin goes active after bus release and the $\overline{SMIADS}$ pin floats along with the other pins that normally float during a bus hold cycle. SMI does not float during a Bus Hold cycle.

Processor State Save

The first set of SMM bus transfer cycles after the CPU's recognition of an active SMI is the processor saving its state to an external RAM array in a separate address space from main system memory. This is accomplished by using the $\overline{SMIADS}$ and $\overline{SMIRDY}$ pins for initiation and termination of bus cycles, instead of the $\overline{ADS}$ and $\overline{READY}$ pins. The 24-bit addresses to which the CPU saves its state are 60000h–600CBh and 60100h–60127h. These are fixed address locations for each register saved.

To ensure valid operation, pipelining must be disabled while the processor is in SMM. There are 114 data transfer cycles.

SMI Code Execution

After the processor state is saved to the separate SMM memory space, the execution of the SMI interrupt routine code begins. The processor enters Real Mode, sets most of the register values to "reset" values (those values normally seen after a CPU reset), and begins fetching code from address FFFFF0h in the separate SMM memory space. Normally, the first thing the interrupt routine code does is a FAR JUMP to the Real Mode entry point for the SMI interrupt routine, which is also in SMM memory space.

Both INTR and NMI are disabled upon entry into SMM. The SMM code can be located anywhere within the 1-Mb Real Mode address space, except for where the processor state is saved. I/O cycles, as a result of the IN, OUT, INS, and OUTS instructions, will go to the normal address space, utilizing the normal $\overline{ADS}$ and $\overline{READY}$ bus interface signals. This facilitates power management code manipulating system hardware registers as needed through the standard I/O subsystem; a separate I/O space is not implemented.

Processor State Restore (Resuming Normal Execution)

Returning to normal code execution in the main system memory, including restoring the processor operating mode, is accomplished by executing a special code sequence. This code invokes a restore CPU state operation that reloads the CPU registers from the saved data in the RAM controlled by $\overline{SMIADS}$ and $\overline{SMIRDY}$.

The ES:EDI register pair must point to the physical address of the processor save state (6000h). In Real Mode the address is calculated as ES · 16 + EDI offset. The saved state should not cross a 64K boundary. The RES3 instruction (0F 07) should be executed to start the restore state operation. After completion of the restore state operation, the $\overline{SMI}$ pin will be deactivated by the CPU and normal code execution will continue at the point where it left off before the SMI occurred. There are 114 data transfer cycles in the restore operation.

Software Features

Several features of the SMI function provide support for special operations during the execution of the system's software. These features involve the execution of reserved opcodes to induce specific SMI related operations.

Software SMI Generation

Besides hardware initiation of the SMI via the $\overline{SMI}$ pin, there is also a software induced SMI mechanism. Generating a soft SMI involves setting a control bit (Bit 12) in the Debug Control Register (DR7) and executing an SMI instruction (opcode F1h).

The functional sequence of the software-based SMI is identical to the hardware-based SMI with the exception that the $\overline{SMI}$ pin is not initially driven active by an external source. Upon execution of a soft SMI opcode, the $\overline{SMI}$ pin is driven active (Low) by the processor before the save state operation begins.

Memory Transfers to Main System Memory

While executing an SMI routine, the interrupt code can initiate memory data reads and writes to the main system memory using the normal $\overline{ADS}$ and $\overline{READY}$ pins. This initiation is accomplished by using reserved opcodes that are special forms of the MOV instruction (called UMOV). The UMOV opcodes can move byte, word, or double word register operands to or from main system memory. Multiple data transfers using the normal $\overline{ADS}$ and $\overline{READY}$ pins will occur if the operands

are misaligned relative to the effective address used. The UMOV opcodes are 0F 10h, 0F 11h, 0F 12h, and 0F 13h. The UMOV instruction can use any of the 386 addressing modes, as specified in the ModR/M byte of the opcode. Note that the 16- and 32-bit versions are the same opcodes with the exception of the 66h operand size prefix.

I/O Instruction Break

The Am386SXLV microprocessor has an I/O instruction break feature that allows the system logic to implement I/O trapping for peripheral devices. To enable the I/O Instruction break feature, $\overline{\text{IIBEN}}$ must first be asserted active Low. On detecting an I/O instruction, the processor prevents the execution unit from executing further instructions until $\overline{\text{READY}}$ is driven active Low by the system. Once $\overline{\text{READY}}$ is driven active, the execution unit either immediately responds to any active interrupt request or continues executing instructions following the I/O instruction that caused the break.

The I/O instruction break feature can be used to allow system logic to implement I/O trapping for peripheral devices. On sensing an I/O instruction, the system drives the $\overline{\text{SMI}}$ pin active before driving $\overline{\text{READY}}$ active. This ensures that the interrupt service routine is executed immediately following the I/O instruction that caused the break. (If the I/O instruction break feature is not enabled via $\overline{\text{IIBEN}}$, several instructions could execute before the SMI service routine is executed.)

The SMI service routine can access the peripheral for which $\overline{\text{SMI}}$ was asserted and modify its state.The SMI service routine normally returns to the instruction following the I/O instruction that caused the break. By modifying the saved state instruction pointer, the routine can choose to return to the I/O instruction that caused the break and re-execute that instruction. The default is to return to the following instruction (except for REP I/O string instruction). To re-execute the I/O instruction that caused the break, the SMI service routine must copy the I/O instruction pointer over the default pointer. This feature is particularly useful when an application program requests an access to a peripheral that has been powered down. The SMI service routine can restore power to the peripheral and initiate a re-execution sequence transparent to the application program. This re-execution feature should only be used if the SMI is in response to an I/O trap with $\overline{\text{IIBEN}}$ active. Note that the I/O instruction break feature is not enabled for memory mapped I/O devices or for coprocessor bus cycles even if $\overline{\text{IIBEN}}$ is active.

I/O Instruction Break Timing

The I/O Instruction Break feature requires that $\overline{\text{SMI}}$ be sampled active (Low) by the processor at least three CLK2 edges before the CLK2 edge that ends the I/O cycle with an active $\overline{\text{READY}}$ signal. This timing applies for both pipelined and non-pipelined cycles. If this timing constraint is not met, additional instructions may be executed by the internal execution unit prior to entering SMM. Depending on the state of the prefetch queue at the time the $\overline{\text{SMI}}$ is asserted, instruction fetch cycles may occur on the normal $\overline{\text{ADS}}$ interface before the SMM save state process begins with the assertion of $\overline{\text{SMIADS}}$. However, this fetched code will not be executed.

True Static Operation

The Am386SXLV microprocessor incorporates a true static design. Unlike dynamic circuit design, the Am386SXLV device eliminates the minimum operating frequency restriction. It may be clocked from its maximum speed all the way down to 0 MHz (DC). System designers can use this feature to design battery-powered notebook PCs with long battery life.

Standby Mode

This true static design of the Am386SXLV microprocessor allows for a Standby Mode. At any operating speed the Am386SXLV microprocessor will retain its state (i.e., the contents of all of its registers). By shutting off the clock completely, the device enters Standby Mode. Since power consumption is proportional to clock frequency, operating power consumption is reduced as the frequency is lowered. In Standby Mode, typical current draw is reduced to less than 0.01 mA. Not only does this feature save battery life, but it also simplifies the design of power-conscious notebook computers in the following ways.

1. Eliminates the need for software in BIOS to save and restore the contents of registers.
2. Allows simpler circuitry to control stopping of the clock (since) the system does not need to know the state of the processor.

Lower Operating Icc

True static design also allows lower operating Icc when operating at any speed.

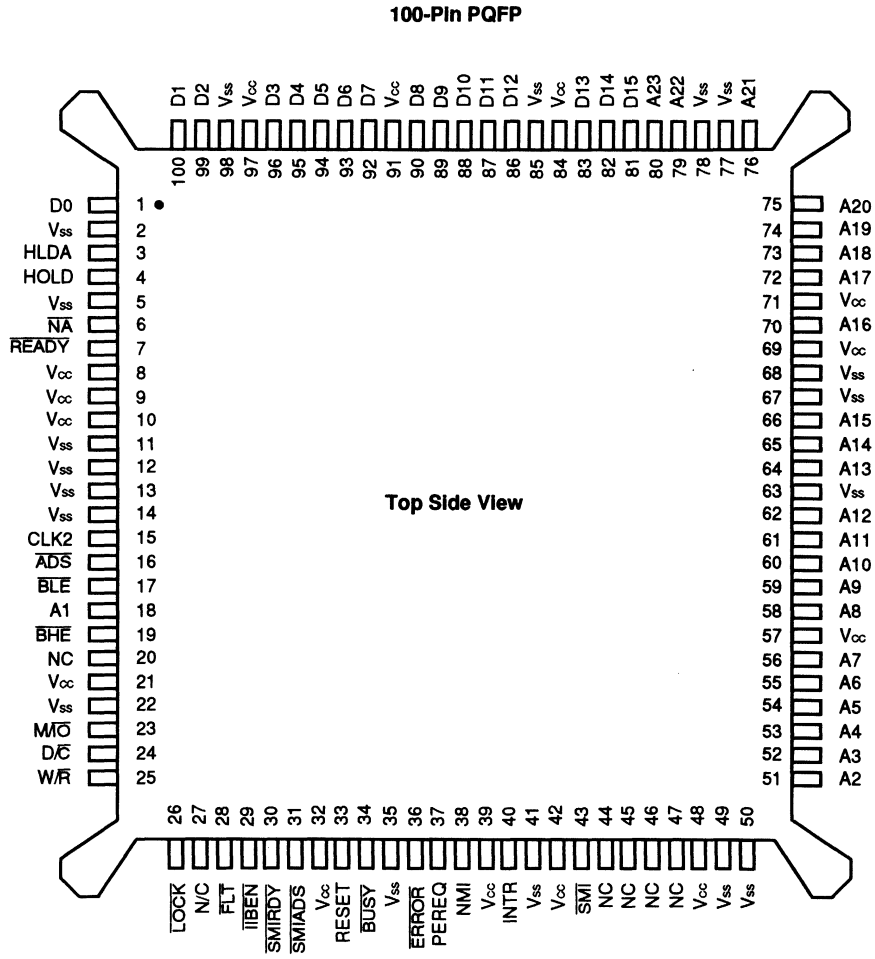
Performance on Demand

The Am386SXLV microprocessor retains its state at any speed from 0 MHz (DC) to its maximum operating speed. With this feature, system designers may vary the operating speed of the system to extend the battery life in notebook systems.

For example, the system could operate at low speeds during inactivity or polling operations. However, upon interrupt, the system clock can be increased up to its maximum speed. After a user-defined time-out period, the system can be returned to a low (or 0 MHz) operating speed without losing its state. This design maximizes battery life while achieving optimal performance.

CONNECTION DIAGRAM

100-Lead Plastic Quad Flat Pack (PQFP) Package—Top Side View

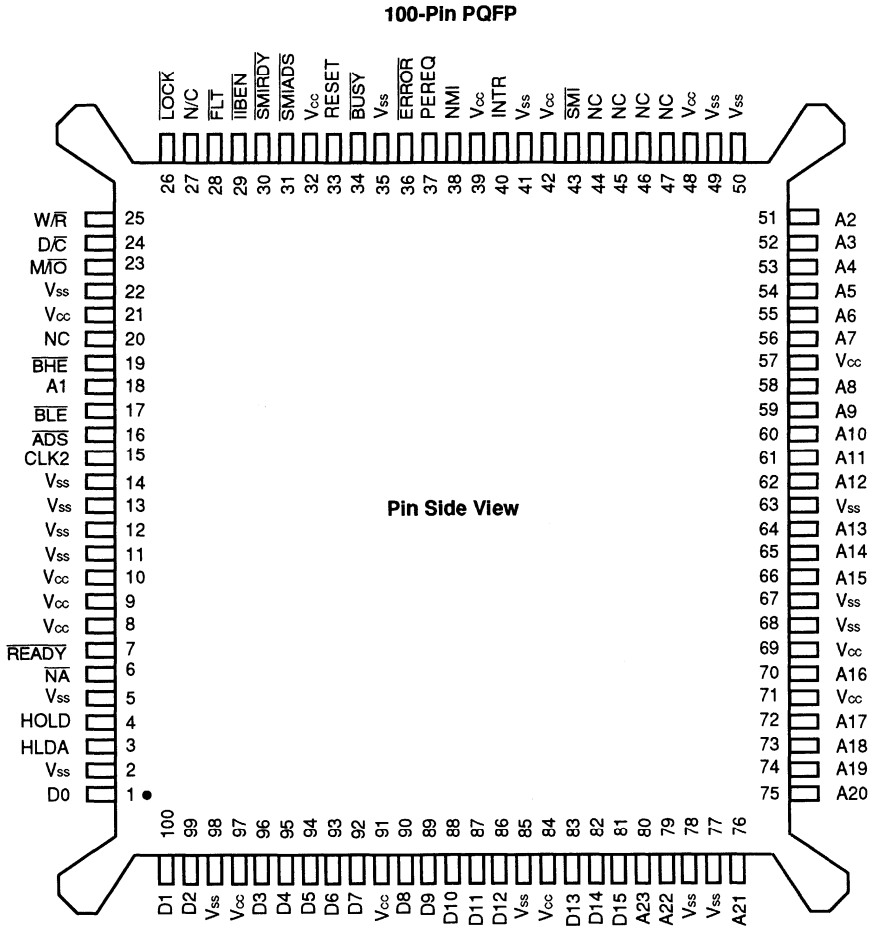


Notes: Pin 1 is marked for orientation.

N/C = Not connected; connection of an N/C pin may cause a malfunction or incompatibility with future shippings of the Am386SXLV microprocessor.

CONNECTION DIAGRAM

100-Lead Plastic Quad Flat Pack (PQFP) Package—Pin Side View



Notes: Pin 1 is marked for orientation.
N/C = Not connected; connection of an N/C pin may cause a malfunction or incompatibility with future shippings of the Am386SXLV microprocessor.

PIN DESIGNATION TABLES (Sorted by Functional Grouping)

Address		Data		Control		NC	V_{CC}	V_{SS}
Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin No.	Pin No.	Pin No.
A1	18	D0	1	$\overline{ADS}$	16	20	8	2
A2	51	D1	100	$\overline{BHE}$	19	27	9	5
A3	52	D2	99	$\overline{BLE}$	17	44	10	11
A4	53	D3	96	$\overline{BUSY}$	34	45	21	12
A5	54	D4	95	CLK2	15	46	32	13
A6	55	D5	94	D/$\overline{C}$	24	47	39	14
A7	56	D6	93	$\overline{ERROR}$	36		42	22
A8	58	D7	92	$\overline{FLT}$	28		48	35
A9	59	D8	90	HLDA	3		57	41
A10	60	D9	89	HOLD	4		69	49
A11	61	D10	88	$\overline{IIBEN}$	29		71	50
A12	62	D11	87	INTR	40		84	63
A13	64	D12	86	$\overline{LOCK}$	26		91	67
A14	65	D13	83	M/$\overline{IO}$	23		97	68
A15	66	D14	82	$\overline{NA}$	6			77
A16	70	D15	81	NMI	38			78
A17	72			PEREQ	37			85
A18	73			$\overline{READY}$	7			98
A20	75			RESET	33			
A21	76			$\overline{SMI}$	43			
A22	79			$\overline{SMIADS}$	31			
A23	80			$\overline{SMIRDY}$	30			
				W/$\overline{R}$	25			

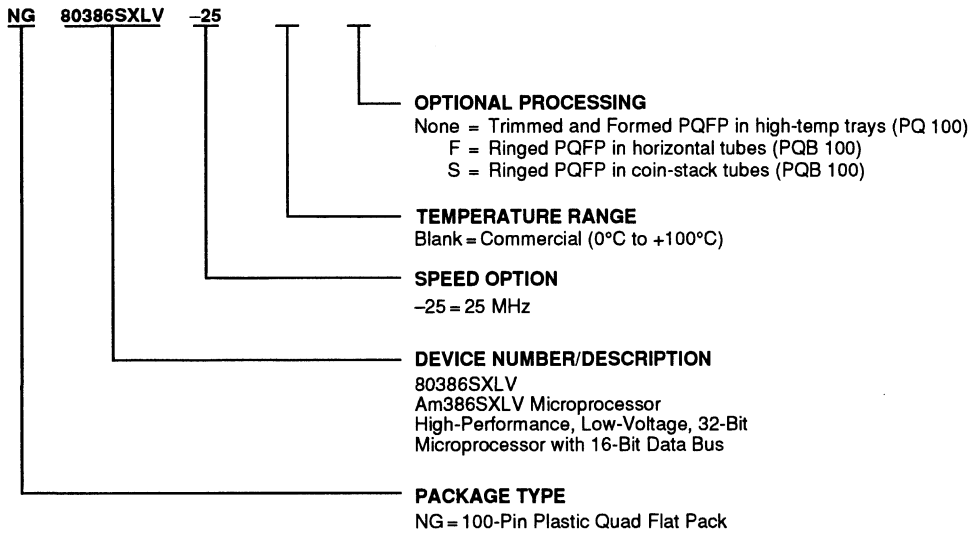
PIN DESIGNATION TABLES (Sorted by Pin Number)

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
1	D0	21	V_{CC}	41	V_{SS}	61	A11	81	D15
2	V_{SS}	22	V_{SS}	42	V_{CC}	62	A12	82	D14
3	HLDA	23	M/$\overline{IO}$	43	$\overline{SMI}$	63	V_{SS}	83	D13
4	HOLD	24	D/$\overline{C}$	44	NC	64	A13	84	V_{CC}
5	V_{SS}	25	W/$\overline{R}$	45	NC	65	A14	85	V_{SS}
6	$\overline{NA}$	26	$\overline{LOCK}$	46	NC	66	A15	86	D12
7	$\overline{READY}$	27	NC	47	NC	67	V_{SS}	87	D11
8	V_{CC}	28	$\overline{FLT}$	48	V_{CC}	68	V_{SS}	88	D10
9	V_{CC}	29	$\overline{IIBEN}$	49	V_{SS}	69	V_{CC}	89	D9
10	V_{CC}	30	$\overline{SMIRDY}$	50	V_{SS}	70	A16	90	D8
11	V_{SS}	31	$\overline{SMIADS}$	51	A2	71	V_{CC}	91	V_{CC}
12	V_{SS}	32	V_{CC}	52	A3	72	A17	92	D7
13	V_{SS}	33	RESET	53	A4	73	A18	93	D6
14	V_{SS}	34	$\overline{BUSY}$	54	A5	74	A19	94	D5
15	CLK2	35	V_{SS}	55	A6	75	A20	95	D4
16	$\overline{ADS}$	36	$\overline{ERROR}$	56	A7	76	A21	96	D3
17	$\overline{BLE}$	37	PEREQ	57	V_{CC}	77	V_{SS}	97	V_{CC}
18	A1	38	NMI	58	A8	78	V_{SS}	98	V_{SS}
19	$\overline{BHE}$	39	V_{CC}	59	A9	79	A22	99	D2
20	NC	40	INTR	60	A10	80	A23	100	D1

ORDERING INFORMATION

Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of the elements below.

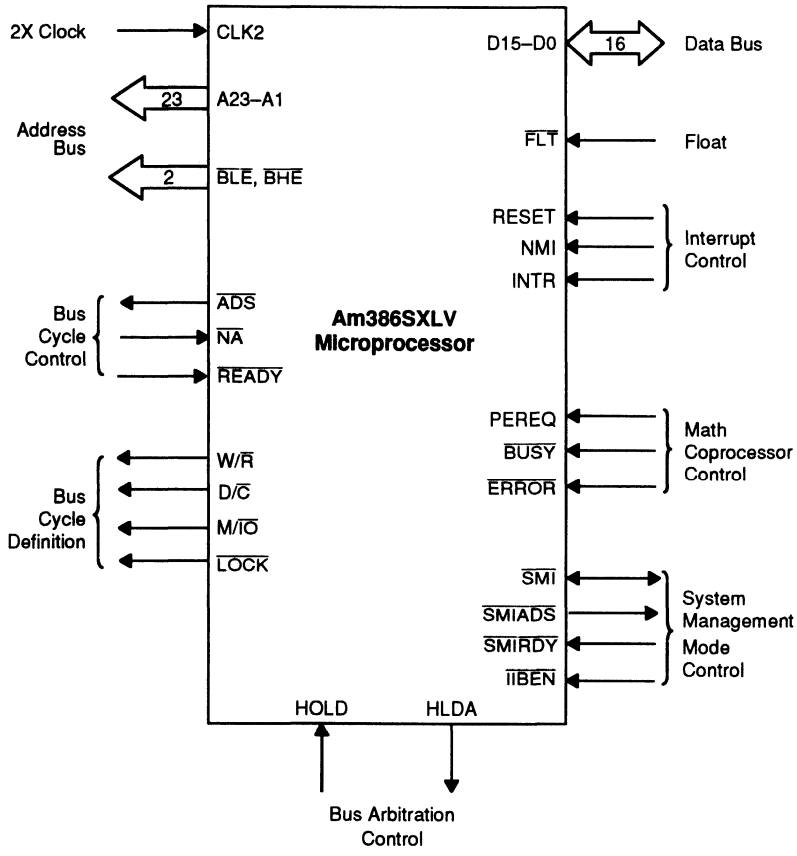


Valid Combinations		
NG	80386SXLV	–25 –25F –25S

Valid Combinations

Valid Combinations lists configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations and to check on newly released combinations.

LOGIC SYMBOL



16305C–003

PIN DESCRIPTIONS

A23–A1
Address Bus (Outputs)

Outputs physical memory or port I/O addresses.

$\overline{ADS}$
Address Status (Active Low; Output)

Indicates that a valid bus cycle definition and address (W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{BHE}$, $\overline{BLE}$, and A23–A1) are being driven at the Am386SXLV microprocessor pins. Bus cycles initiated by $\overline{ADS}$ must be terminated by $\overline{READY}$.

$\overline{BHE}$, $\overline{BLE}$
Byte Enables (Active Low; Outputs)

Indicate which data bytes of the data bus take part in a bus cycle.

$\overline{BUSY}$
Busy (Active Low; Input)

Signals a busy condition from a processor extension. $\overline{BUSY}$ has an internal pull-up resistor.

CLK2
CLK2 (Input)

Provides the fundamental timing for the Am386SXLV microprocessor.

D15–D0
Data Bus (Inputs/Outputs)

Inputs data during memory, I/O, and interrupt acknowledge read cycles; outputs data during memory and I/O write cycles.

D/$\overline{C}$
Data/Control (Output)

A bus cycle definition pin that distinguishes data cycles, either memory or I/O, from control cycles which are: interrupt acknowledge, halt, and code fetch.

$\overline{ERROR}$
Error (Active Low; Input)

Signals an error condition from a processor extension. $\overline{ERROR}$ has an internal pull-up resistor.

$\overline{FLT}$
Float (Active Low; Input)

An input which forces all bidirectional and output signals, including HLDA, to the three-state condition. $\overline{FLT}$ has an internal pull-up resistor. The pin, if not used, should be disconnected.

HLDA
Bus Hold Acknowledge (Active High; Output)

Output indicates that the Am386SXLV microprocessor has surrendered control of its logical bus to another bus master.

HOLD
Bus Hold Request (Active High; Input)

Input allows another bus master to request control of the local bus.

$\overline{IIBEN}$
I/O Instruction Break Enable (Active Low; Input)

Enables the I/O instruction break feature. $\overline{IIBEN}$ has a dynamic internal pull-up resistor. The $\overline{IIBEN}$ pull-up is active during RESET and whenever the signal is not driven active low by the system.

INTR
Interrupt Request (Active High; Input)

A maskable input that signals the Am386SXLV microprocessor to suspend execution of the current program and execute an interrupt acknowledge function.

$\overline{LOCK}$
Bus Lock (Active Low; Output)

A bus cycle definition pin that indicates that other system bus masters are not to gain control of the system bus while it is active.

M/$\overline{IO}$
Memory/IO (Output)

A bus cycle definition pin that distinguishes memory cycles from input/output cycles.

$\overline{NA}$
Next Address (Active Low; Input)

Used to request address pipelining.

N/C
No Connect

Should always be left unconnected. Connection of an N/C pin may cause the processor to malfunction or be incompatible with future steppings of the Am386SXLV microprocessor.

NMI
Non-Maskable Interrupt Request (Active High; Input)

A non-maskable input that signals the Am386SXLV microprocessor to suspend execution of the current

program and execute an interrupt acknowledge function.

PEREQ
Processor Extension Request (Active High; Input)

Indicates that the processor has data to be transferred by the Am386SXLV microprocessor. PEREQ has an internal pull-down resistor.

$\overline{\text{READY}}$
Bus Ready (Active Low; Input)

Terminates the bus cycle initiated by $\overline{\text{ADS}}$.

RESET
Reset (Active High; Input)

Suspends any operation in progress and places the Am386SXLV microprocessor in a known reset state.

$\overline{\text{SMI}}$
System Management Interrupt (Active Low; I/O)

A non-maskable interrupt pin that signals the Am386SXLV microprocessor to suspend execution and enter System Management Mode. $\overline{\text{SMI}}$ has an internal pull-up resistor. $\overline{\text{SMI}}$ has a dynamic internal pull-up resistor that is disabled when the processor is in SMM. $\overline{\text{SMI}}$ is not three-stated during Hold Acknowledge bus cycles.

$\overline{\text{SMIADS}}$
SMI Address Status (Active Low; Output)

When active, this pin indicates that a valid bus cycle definition and address (W/$\overline{\text{R}}$ D/$\overline{\text{C}}$, M/$\overline{\text{IO}}$, $\overline{\text{BHE}}$, $\overline{\text{BLE}}$, and A23–A1) are being driven at the Am386SXLV microprocessor pins while in the System Management Mode. Bus cycles initiated by $\overline{\text{SMIADS}}$ must be terminated by $\overline{\text{SMIRDY}}$.

$\overline{\text{SMIRDY}}$
SMI Ready (Active Low; Input)

This input terminates the current bus cycle to the SMM Mode address space in the same manner the $\overline{\text{READY}}$ pin does for the normal mode address space. $\overline{\text{SMIRDY}}$ has an internal pull-up resistor. $\overline{\text{READY}}$ and $\overline{\text{SMIRDY}}$ must not be tied together.

V_{CC}
System Power (Input)

Provides the DC supply input.

V_{SS}
System Ground (Input)

Provides the 0-V connection from which all inputs and outputs are measured.

W/$\overline{\text{R}}$
Write/Read (Output)

A bus cycle definition pin that distinguishes write cycles from read cycles.

ABSOLUTE MAXIMUM RATINGS

Storage Temperature . . . . . –65°C to +150°C
Ambient Temperature Under Bias . –65°C to +125°C

Stresses above those listed may cause permanent damage to the device. Functionality at or above these limits is not implied. Exposure to ABSOLUTE MAXIMUM RATING conditions for extended periods of time may affect device reliability.

OPERATING RANGES

Supply Voltage with respect to V_{SS} . . –0.5 V to +7.0 V
Voltage on Other Pins . . . . . –0.5 V to V_{cc} +0.5 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over COMMERCIAL operating ranges

V_{CC} = 3.0 V to 3.6 V; T_{CASE} = 0°C to +100°C

			Final		
Symbol	**Parameter Description**	**Notes**	**Min**	**Max**	**Unit**
V_{IL}	Input Low Voltage	(Note 1)	–0.3	+0.8	V
V_{IH}	Input High Voltage		2.0	V_{CC}+0.3	V
V_{ILC}	CLK2 Input Low Voltage	(Note 1)	–0.3	+0.8	V
V_{IHC}	CLK2 Input High Voltage (25 MHz)		2.4	V_{CC}+0.3	V
V_{OL}	Output Low Voltage				
	I_{OL} = 0.5 mA: A23–A1, D15–D0	(Note 5)		0.2	V
	I_{OL} = 0.5 mA: $\overline{BHE}$, $\overline{BLE}$,W/$\overline{R}$, D/$\overline{C}$, $\overline{SMIADS}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, HLDA, $\overline{SMI}$			0.2	V
	I_{OL} = 2 mA: A23–A1, D15–D0			0.45	V
	I_{OL} = 2.5 mA: $\overline{BHE}$, $\overline{BLE}$, W/$\overline{R}$, D/$\overline{C}$, $\overline{SMIADS}$, $\overline{LOCK}$, $\overline{ADS}$, M/$\overline{IO}$, HLDA, $\overline{SMI}$			0.45	V
V_{OH}	Output High Voltage				
	I_{OH} = 0.1 mA: A23–A1, D15–D0	(Note 5)	V_{CC}– 0.2		V
	I_{OH} = 0.1 mA: $\overline{BHE}$, $\overline{BLE}$, W/$\overline{R}$, D/$\overline{C}$, $\overline{SMIADS}$, $\overline{LOCK}$, $\overline{ADS}$, M/$\overline{IO}$, HLDA, $\overline{SMI}$	(Note 6)	V_{CC}– 0.2		V
	I_{OH} = 0.5 mA: A23–A1, D15–D0		V_{CC}– 0.45		V
	I_{OH} = 0.5 mA: $\overline{BHE}$, $\overline{BLE}$, W/$\overline{R}$, D/$\overline{C}$, $\overline{SMIADS}$, $\overline{LOCK}$, $\overline{ADS}$, M/$\overline{IO}$, HLDA, $\overline{SMI}$		V_{CC}– 0.45		V
I_{LI}	Input Leakage Current (All pins except PEREQ, $\overline{BUSY}$, $\overline{ERROR}$, $\overline{SMI}$, $\overline{SMIRDY}$, $\overline{FLT}$, $\overline{IIBEN}$)	0 V ≤ V_{IN} ≤ V_{CC} (Note 7)		±10	µA
I_{IH}	Input Leakage Current (PEREQ pin)	V_{IH} = V_{CC}–0.1 V V_{IH} = 2.4 V (Note 2)		300 200	µA µA
I_{IL}	Input Leakage Current ($\overline{BUSY}$, $\overline{ERROR}$, $\overline{SMI}$, $\overline{SMIRDY}$, $\overline{FLT}$, $\overline{IIBEN}$)	V_{IL} = 0.1 V V_{IL} = 0.45 V (Note 3)		–300 –200	µA µA
I_{LO}	Output Leakage Current	0.1 V ≤ V_{OUT} ≤ V_{CC}		±15	µA
I_{CC}	Supply Current (Note 8) CLK2 = 40 MHz: Oper. Freq. 20 MHz CLK2 = 50 MHz: Oper. Freq. 25 MHz	V_{CC} = 3.3 V I_{CC} Typ = 80 I_{CC} Typ = 95		V_{CC} = 3.6 V 95 115	 mA mA
I_{CCSB}	Standby Current (Note 8)	I_{CCSB} Typ = 10 µA		150	µA
C_{IN}	Input or I/O Capacitance	F_C = 1 MHz (Note 4)		10	pF
C_{OUT}	Output Capacitance	F_C = 1 MHz (Note 4)		12	pF
C_{CLK}	CLK2 Capacitance	F_C = 1 MHz (Note 4)		20	pF

Notes: 1. The Min value, –0.3, is not 100% tested.
2. PEREQ input has an internal pull-down resistor.
3. $\overline{BUSY}$, $\overline{ERROR}$, $\overline{FLT}$, $\overline{SMI}$, $\overline{IIBEN}$, and $\overline{SMIRDY}$ inputs each have an internal pull-up resistor.
4. Not 100% tested.
5. Outputs are CMOS and will pull rail-to-rail if the load is not resistive.
6. V_{OH} $\overline{SMI}$ only valid on $\overline{SMI}$ output when exiting SMM for two CLK2 periods.
7. $\overline{SMI}$ and $\overline{IIBEN}$ leakage Low will be I_{LI} when pull-up is inactive and I_{IL} when pull-up is active.
8. Inputs at rails (V_{CC} or V_{SS}).

ABSOLUTE MAXIMUM RATINGS

Storage Temperature . . . . . –65°C to +150°C
Ambient Temperature under Bias . –65°C to +125°C

Stresses above those listed may cause permanent damage to the device. Functionality at or above these limits is not implied. Exposure to ABSOLUTE MAXIMUM RATING conditions for extended periods of time may affect device reliability.

OPERATING RANGES

Supply Voltage with respect to V_{SS} . . . –0.5 V to +7 V
Voltage on Other Pins . . . . . –0.5 V to V_{CC} +0.5 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over COMMERCIAL operating ranges

V_{CC} = 3.6 V to 5.5 V; T_{CASE} = 0°C to +100°C

			Final		
Symbol	**Parameter Description**	**Notes**	**Min**	**Max**	**Unit**
V_{IL}	Input Low Voltage	(Note 1)	–0.3	+0.8	V
V_{IH}	Input High Voltage		2.0	V_{CC}+0.3	V
V_{ILC}	CLK2 Input Low Voltage	(Note 1)	–0.3	+0.8	V
V_{IHC}	CLK2 Input High Voltage (33 MHz)		2.7	V_{CC}+0.3	V
V_{OL}	Output Low Voltage I_{OL} = 4 mA: A23–A1, D15–D0 I_{OL} = 5 mA: $\overline{BHE}$, $\overline{BLE}$, W/$\overline{R}$, D/$\overline{C}$, $\overline{SMIADS}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{ADS}$, HLDA, $\overline{SMI}$	(Note 5)		0.45 0.45	V V
V_{OH}	Output High Voltage I_{OH} = 1.0 mA: A23–A1, D15–D0 I_{OH} = 0.2 mA: A23–A1, D15–D0 I_{OH} = 0.9 mA: $\overline{BHE}$, $\overline{BLE}$, W/$\overline{R}$, D/$\overline{C}$, $\overline{SMIADS}$, $\overline{LOCK}$, $\overline{ADS}$, M/$\overline{IO}$, HLDA, $\overline{SMI}$ I_{OH} = 0.18 mA: $\overline{BHE}$, $\overline{BLE}$, W/$\overline{R}$, D/$\overline{C}$, $\overline{SMIADS}$, $\overline{LOCK}$, $\overline{ADS}$, M/$\overline{IO}$, HLDA, $\overline{SMI}$	(Note 5) (Note 6) 	 2.4 V_{CC}– 0.5 2.4 V_{CC}– 0.5	 	 V V V V
I_{LI}	Input Leakage Current (All pins except PEREQ, $\overline{BUSY}$, $\overline{ERROR}$, $\overline{SMI}$, $\overline{SMIRDY}$, $\overline{FLT}$, and $\overline{IIBEN}$)	0 V ≤ V_{IN} ≤ V_{CC} (Note 7)		±15	µA
I_{IH}	Input Leakage Current (PEREQ pin)	V_{IH} = 2.4 V (Note 2)		200	µA
I_{IL}	Input Leakage Current ($\overline{BUSY}$, $\overline{ERROR}$, $\overline{SMI}$, $\overline{SMIRDY}$, $\overline{FLT}$, $\overline{IIBEN}$)	V_{IL} = 0.45 V (Note 3)		–400	µA
I_{LO}	Output Leakage Current	0.1 V ≤ V_{OUT} ≤ V_{CC}		±15	µA
I_{CC}	Supply Current (Note 8) CLK2 = 40 MHz: Oper. Freq. 20 MHz CLK2 = 50 MHz: Oper. Freq. 25 MHz CLK2 = 66 MHz: Oper. Freq. 33 MHz	V_{CC} Typ = 5.0 V I_{CC} Typ = 130 I_{CC} Typ = 160 I_{CC} Typ = 210		V_{CC} = 5.5 155 190 245	V mA mA mA
I_{CCSB}	Standby Current (Note 8)	I_{CCSB} Typ = 20 µA		150	µA
C_{IN}	Input or I/O Capacitance	F_C = 1 MHz (Note 4)		10	pF
C_{OUT}	Output Capacitance	F_C = 1 MHz (Note 4)		12	pF
C_{CLK}	CLK2 Capacitance	F_C = 1 MHz (Note 4)		20	pF

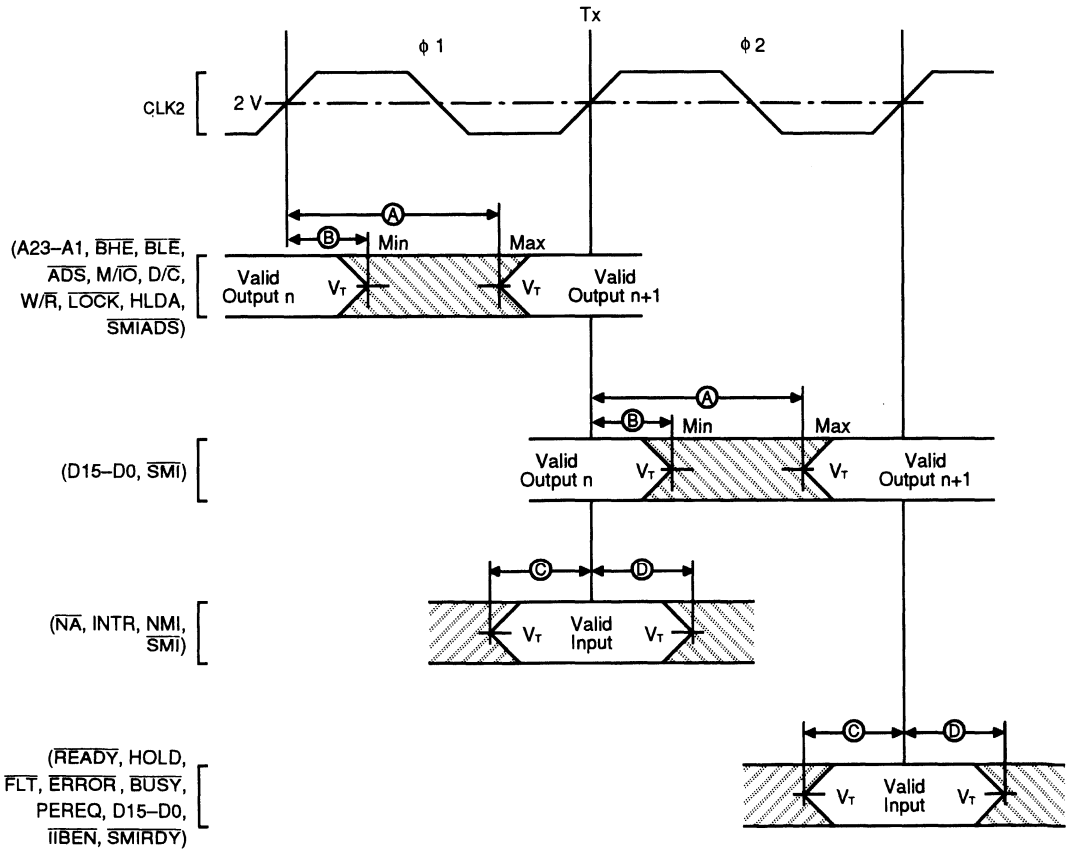
Notes: 1. The Min value, –0.3, is not 100% tested.
2. PEREQ input has an internal pull-down resistor.
3. $\overline{BUSY}$, $\overline{ERROR}$, $\overline{FLT}$, $\overline{SMI}$, $\overline{IIBEN}$, and $\overline{SMIRDY}$ inputs each have an internal pull-up resistor.
4. Not 100% tested.
5. Outputs are CMOS and will pull rail-to-rail if the load is not resistive.
6. V_{OH} $\overline{SMI}$ only valid on $\overline{SMI}$ output when exiting SMM for two CLK2 periods.
7. $\overline{SMI}$ and $\overline{IIBEN}$ leakage Low will be I_{LI} when pull-up is inactive and I_{IL} when pull-up is active.
8. Inputs at rails (V_{CC} or V_{SS}).

SWITCHING CHARACTERISTICS

The switching characteristics given consist of output delays, input setup requirements, and input hold requirements. All switching characteristics are relative to the CLK2 rising edge crossing the 2.0-V level.

Switching characteristic measurement is defined in Figure 2. Inputs must be driven to the voltage levels indicated by Figure 2 when switching characteristics are measured. Output delays are specified with minimum and maximum limits measured, as shown. The minimum delay times are hold times provided to external circuitry. Input setup and hold times are specified as minimums, defining the smallest acceptable sampling window. Within the sampling window, a synchronous input signal must be stable for correct operation.

Outputs $\overline{ADS}$, W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{LOCK}$, $\overline{BHE}$, $\overline{BLE}$, $\overline{SMIADS}$, A23–A1, and HLDA only change at the beginning of phase one. D15–D0 and $\overline{SMI}$ write cycles only change at the beginning of phase two. The $\overline{IIBEN}$, $\overline{READY}$, HOLD, $\overline{BUSY}$, $\overline{SMIRDY}$, $\overline{ERROR}$, PEREQ, $\overline{FLT}$, and D15–D0 (read cycles) inputs are sampled at the beginning of phase one. The $\overline{NA}$, INTR, NMI, and $\overline{SMI}$ inputs are sampled at the beginning of phase two.



Legend: A—Maximum Output Delay Characteristic
B—Minimum Output Delay Characteristic
C—Minimum Input Setup Characteristic
D—Minimum Input Hold Characteristic

Notes: 1. Input waveforms have tr ≤ 2.0 ns from 0.8 V–2.0 V.
2. V_T = 1.0 V for V_{CC} ≤ 3.6 V; 1.5 V for V_{CC} > 3.6 V

16305C–003

Figure 2. Drive Levels and Measurement Points for Switching Characteristics

SWITCHING CHARACTERISTICS over operating ranges at 25 MHz

V_{CC} = 3.0 V–5.5 V; T_{CASE} = 0°C to 100°C

Symbol	Parameter Description	Notes	Ref. Figures	Final Min	Final Max	Unit
	Operating Frequency	Half CLK2 freq.		0	25	MHz
1	CLK2 Period		3	20		ns
2	CLK2 High Time	at V_{IHC}	3	4		ns
3	CLK2 Low Time	at 0.8 V	3	5		ns
4	CLK2 Fall Time	2.4 V to 0.8 V (Note 3)	3		7	ns
5	CLK2 Rise Time	0.8 V to 2.4 V (Note 3)	3		7	ns
6	A23–A1 Valid Delay	C_L = 50 pF	6	4	17	ns
7	A23–A1 Float Delay	(Note 1)	13	4	30	ns
8	$\overline{BHE}$, $\overline{BLE}$, $\overline{LOCK}$ Valid Delay	C_L = 50 pF	6	4	17	ns
9	$\overline{BHE}$, $\overline{BLE}$, $\overline{LOCK}$ Float Delay	(Note 1)	13	4	30	ns
10	M/$\overline{IO}$, D/$\overline{C}$, W/$\overline{R}$, $\overline{ADS}$ Valid Delay	C_L = 50 pF	6	4	17	ns
10s	$\overline{SMIADS}$ Valid Delay	C_L = 50 pF	6	4	25	ns
11	W/$\overline{R}$, M/$\overline{IO}$, D/$\overline{C}$, $\overline{ADS}$ Float Delay	(Note 1)	13	4	30	ns
11s	$\overline{SMIADS}$ Float Delay	(Note 1)	13	4	30	ns
12	D15–D0 Write Data Valid Delay	C_L = 50 pF	6, 7	7	23	ns
12a	D15–D0 Write Data Hold Time	C_L = 50 pF	8	2		ns
13	D15–D0 Write Data Float Delay	(Note 1)	13	4	22	ns
14	HLDA Valid Delay	C_L = 50 pF	6	4	22	ns
14f	HLDA Float Delay	(Notes 1, 4)	14	4	30	ns
15	$\overline{NA}$ Setup Time		5	5		ns
16	$\overline{NA}$ Hold Time		5	3		ns
19	$\overline{READY}$ Setup Time		5	9		ns
19s	$\overline{SMIRDY}$ Setup Time		5	9		ns
20	$\overline{READY}$ Hold Time		5	4		ns
20s	$\overline{SMIRDY}$ Hold Time		5	4		ns
21	D15–D0 Read Data Setup Time		5	7		ns
22	D15–D0 Read Data Hold Time		5	5		ns
23	HOLD Setup Time		5	9		ns
24	HOLD Hold Time		5	3		ns
25	RESET Setup Time		15	8		ns
26	RESET Hold Time		15	3		ns
27	NMI, INTR Setup Time	(Note 2)	5	6		ns
27s	$\overline{SMI}$ Setup Time		5	6		ns
28	NMI, INTR Hold Time	(Note 2)	5	6		ns
28s	$\overline{SMI}$ Hold Time		5	4		ns
29	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{IIBEN}$ Setup Time	(Note 2)	5	6		ns
30	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{IIBEN}$ Hold Time	(Note 2)	5	5		ns
31	$\overline{SMI}$ Valid Delay		6, 13	4	22	ns
32	$\overline{SMI}$ Float Delay	(Notes 1, 4)	14	4	30	ns

Notes: 1. Float condition occurs when maximum output current becomes less than I_{LO} in magnitude. Float delay is not 100% tested.

2. These inputs are allowed to be asynchronous to CLK2. The setup and hold specifications are given for testing purposes, to assure recognition within a specific CLK2 period.

3. Rise and Fall are not tested. They are guaranteed by design characterization.

4. Only during $\overline{FLT}$ assertion.

SWITCHING CHARACTERISTICS over operating ranges at 33 MHz

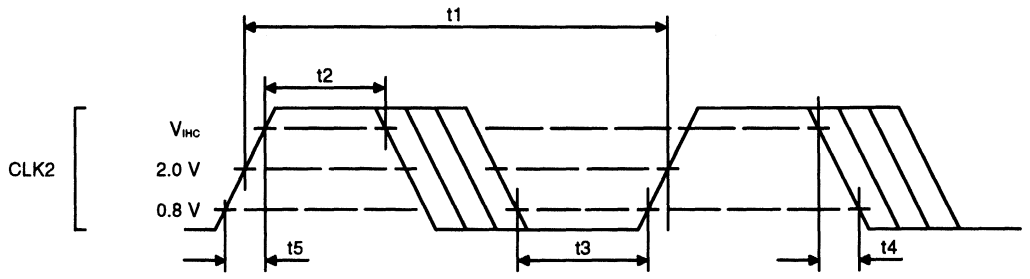
V_{CC} = 4.5 V–5.5 V; T_{CASE} = 0°C to 100°C

				Final		
Symbol	Parameter Description	Notes	Ref. Figures	Min	Max	Unit
	Operating Frequency	Half CLK2 freq.		0	33.3	MHz
1	CLK2 Period		3	15		ns
2	CLK2 High Time	at V_{IHC}	3	4		ns
3	CLK2 Low Time	at 0.8 V	3	4.5		ns
4	CLK2 Fall Time	2.4 V to 0.8 V (Note 3)	3		4	ns
5	CLK2 Rise Time	0.8 V to 2.4 V (Note 3)	3		4	ns
6	A23–A1 Valid Delay	C_L = 50 pF	6	4	15	ns
7	A23–A1 Float Delay	(Note 1)	13	4	20	ns
8	$\overline{BHE}$, $\overline{BLE}$, $\overline{LOCK}$ Valid Delay	C_L = 50 pF	6	4	15	ns
9	$\overline{BHE}$, $\overline{BLE}$, $\overline{LOCK}$ Float Delay	(Note 1)	13	4	20	ns
10	M/$\overline{IO}$, D/$\overline{C}$, W/$\overline{R}$, $\overline{ADS}$ Valid Delay	C_L = 50 pF	6	4	15	ns
10s	$\overline{SMIADS}$ Valid Delay	C_L = 50 pF	6	4	15	ns
11	W/$\overline{R}$, M/$\overline{IO}$, D/$\overline{C}$, $\overline{ADS}$ Float Delay	(Note 1)	13	4	20	ns
11s	$\overline{SMIADS}$ Float Delay	(Note 1)	13	4	20	ns
12	D15–D0 Write Data Valid Delay	C_L = 50 pF	6, 7	7	23	ns
12a	D15–D0 Write Data Hold Time	C_L = 50 pF	8	2		ns
13	D15–D0 Write Data Float Delay	(Note 1)	13	4	17	ns
14	HLDA Valid Delay	C_L = 50 pF	6	4	20	ns
14f	HLDA Float Delay	(Notes 1, 4)	14	4	20	ns
15	$\overline{NA}$ Setup Time		5	5		ns
16	$\overline{NA}$ Hold Time		5	2		ns
19	$\overline{READY}$ Setup Time		5	7		ns
19s	$\overline{SMIRDY}$ Setup Time		5	7		ns
20	$\overline{READY}$ Hold Time		5	4		ns
20s	$\overline{SMIRDY}$ Hold Time		5	4		ns
21	D15–D0 Read Data Setup Time		5	5		ns
22	D15–D0 Read Data Hold Time		5	3		ns
23	HOLD Setup Time		5	9		ns
24	HOLD Hold Time		5	2		ns
25	RESET Setup Time		15	5		ns
26	RESET Hold Time		15	2		ns
27	NMI, INTR Setup Time	(Note 2)	5	5		ns
27s	$\overline{SMI}$ Setup Time		5	5		ns
28	NMI, INTR Hold Time	(Note 2)	5	5		ns
28s	$\overline{SMI}$ Hold Time		5	4		ns
29	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{IIBEN}$ Setup Time	(Note 2)	5	5		ns
30	PEREQ, $\overline{ERROR}$, $\overline{BUSY}$, $\overline{FLT}$, $\overline{IIBEN}$ Hold Time	(Note 2)	5	4		ns
31	$\overline{SMI}$ Valid Delay		6, 13	4	17	ns
32	$\overline{SMI}$ Float Delay	(Notes 1, 4)	14	4	20	ns

Notes: 1. Float condition occurs when maximum output current becomes less than I_{LO} in magnitude. Float delay is not 100% tested.

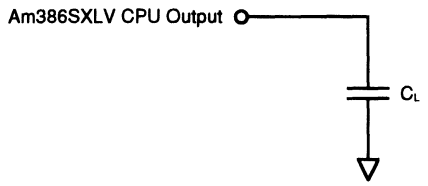
2. These inputs are allowed to be asynchronous to CLK2. The setup and hold specifications are given for testing purposes, to assure recognition within a specific CLK2 period.
3. Rise and Fall are not tested. They are guaranteed by design characterization.
4. Only during $\overline{FLT}$ assertion.

SWITCHING CHARACTERISTICS (continued)



16305C–004

Figure 3. CLK2 Timing



15022B–032

Figure 4. AC Test Circuit

SWITCHING WAVEFORMS

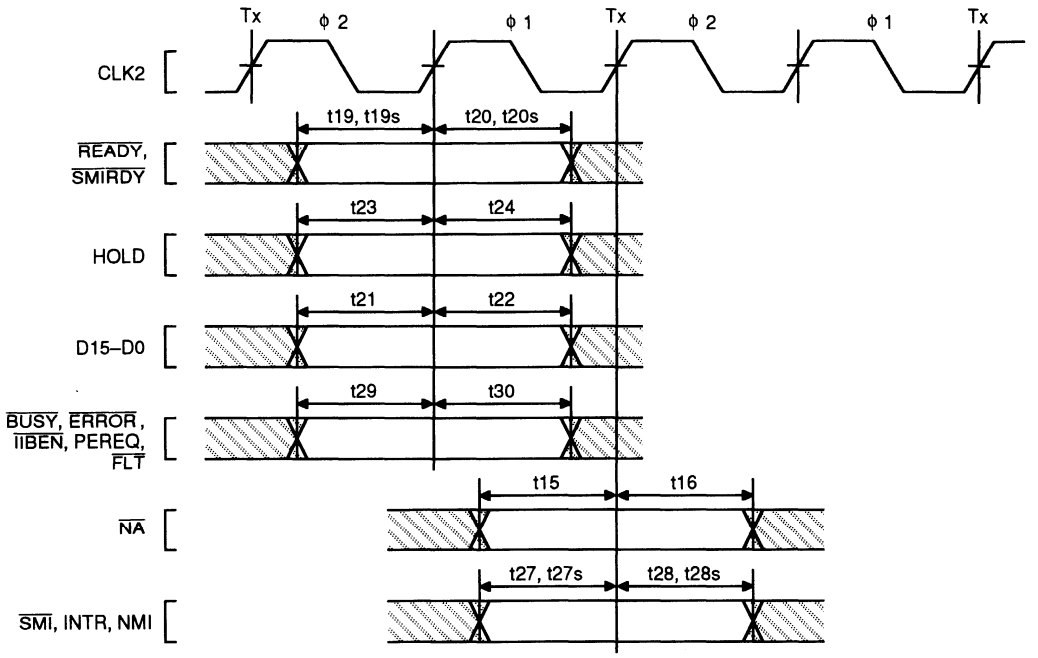


Figure 5. Input Setup and Hold Timing

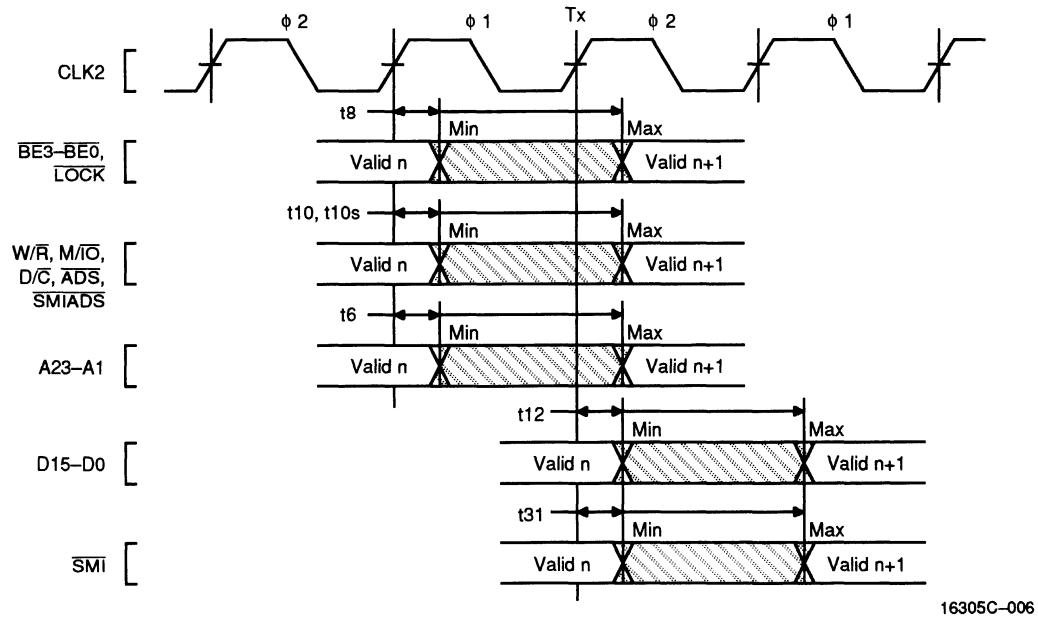
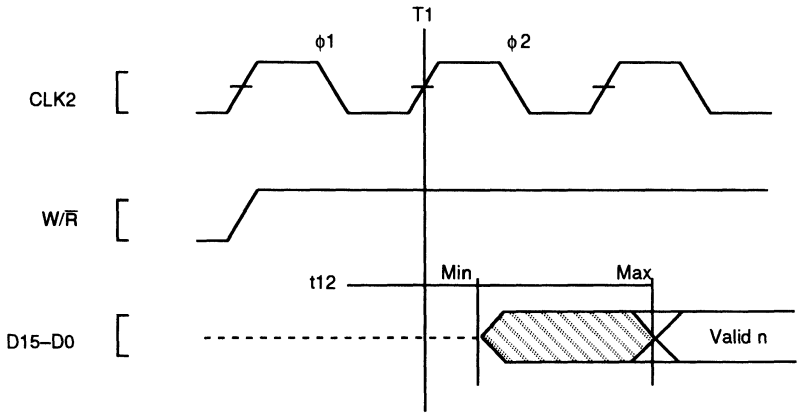
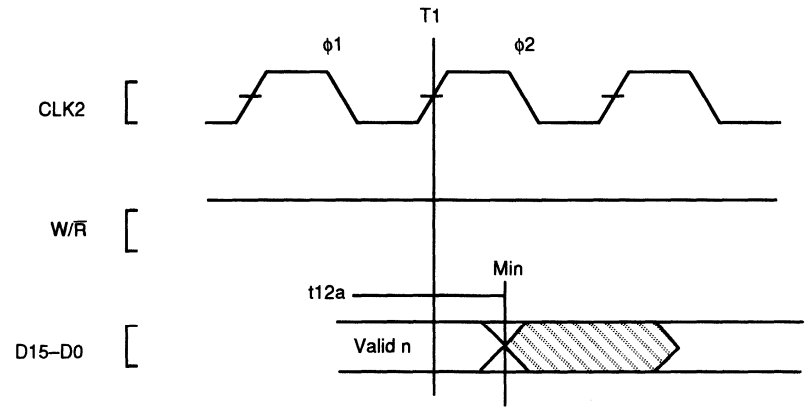


Figure 6. Output Valid Delay Timing



13605C–007

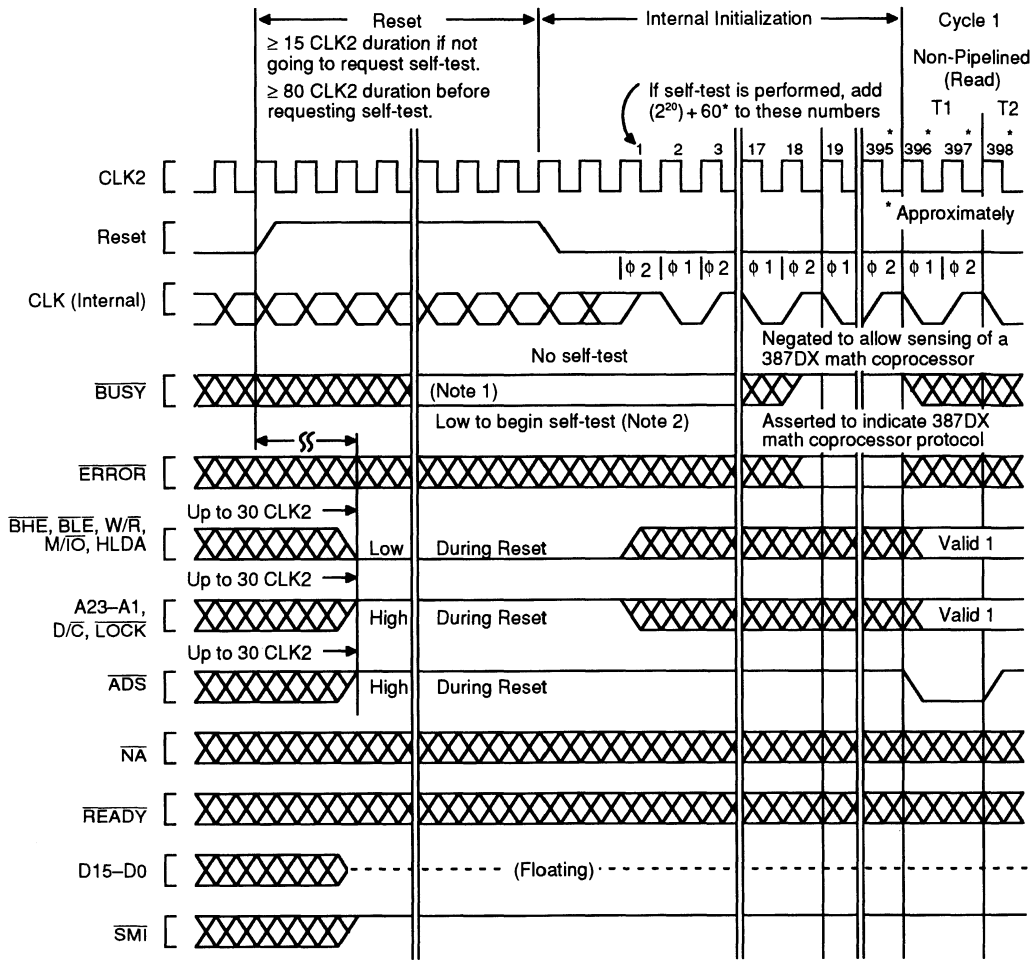
Figure 7. Write Data Valid Delay Timing



16305C–008

Figure 8. Write Data Hold Timing

SWITCHING WAVEFORMS (continued)



Notes: 1. BUSY should be held stable for eight CLK2 periods before and after the CLK2 period in which the RESET falling edge occurs.
2. If self-test is requested, the Am386SXLV microprocessor outputs remain in their reset state as shown here.

Figure 9. Bus Activity from Reset Until First Code Fetch 16305C–009

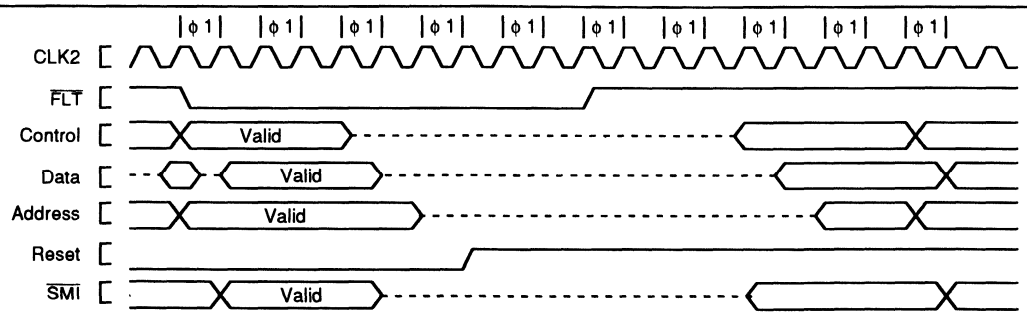
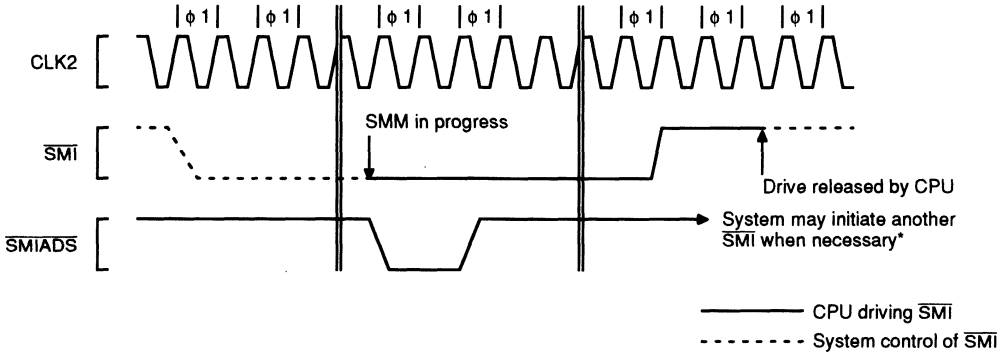


Figure 10. Entering and Exiting $\overline{FLT}$

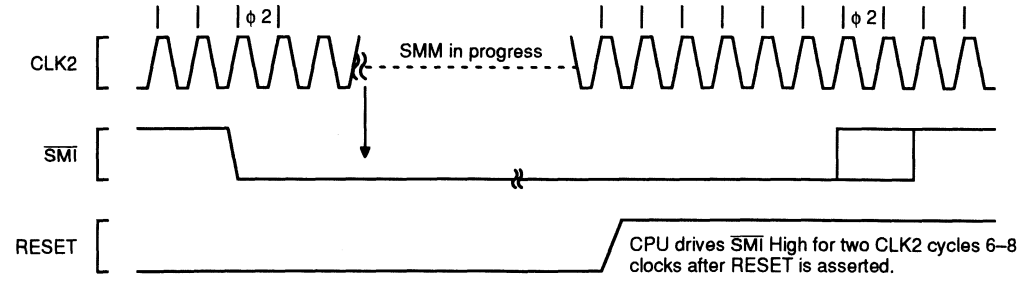
16306B–008



*Once initiated, the system must hold $\overline{SMI}$ Low until the first $\overline{SMIADS}$. At this time, the system cannot drive $\overline{SMI}$ until three CLK2 cycles after the CPU drives $\overline{SMI}$ High. (The CPU will drive $\overline{SMI}$ High for two CLK2 cycles. The additional clock allows the CPU to completely release $\overline{SMI}$ and prevents any driver overlap.)

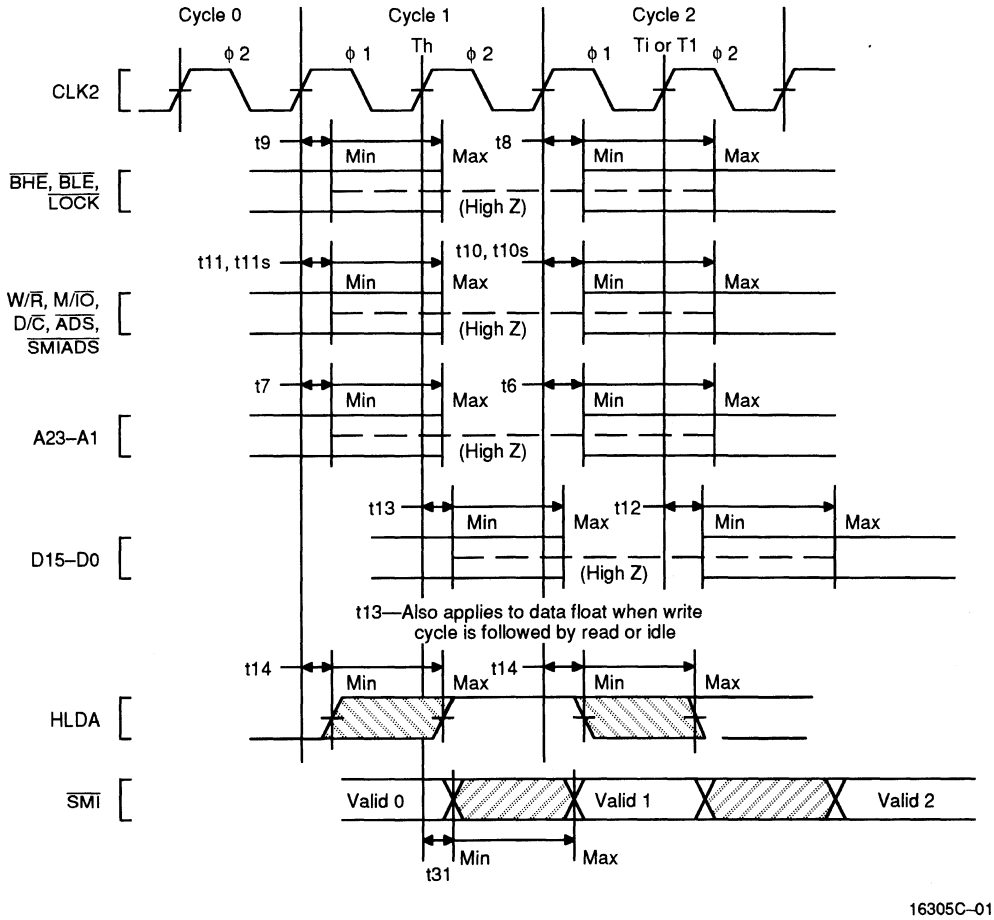
16306B–011

Figure 11. Initiating and Exiting SMM



16306B–010

Figure 12. RESET and $\overline{SMI}$



16305C–011

Figure 13. Output Float Delay and HLDA and $\overline{SMI}$ Valid Delay Timing

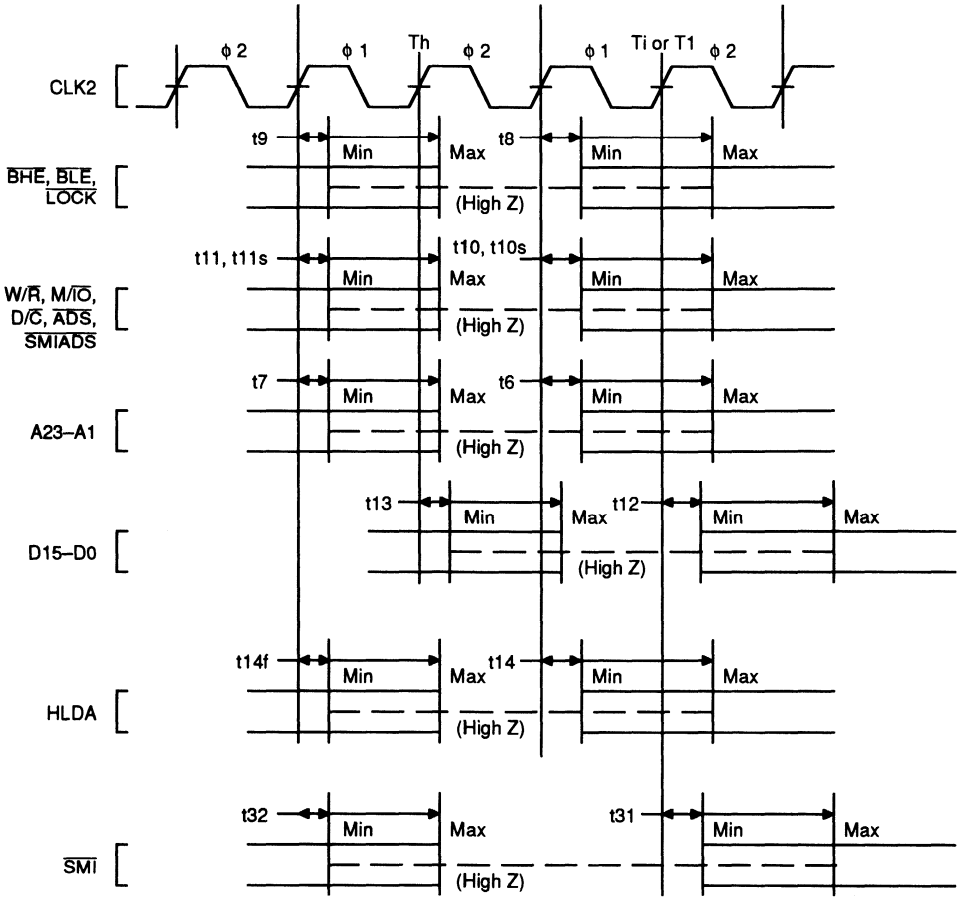
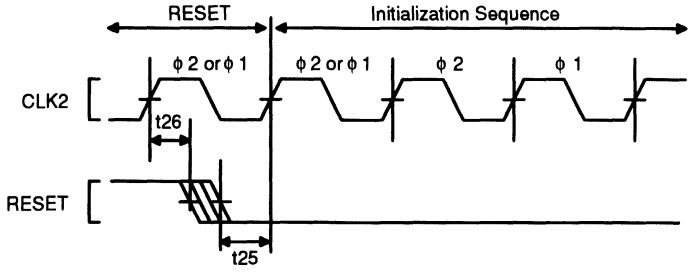


Figure 14. Output Float Delay Entering and Exiting $\overline{FLT}$

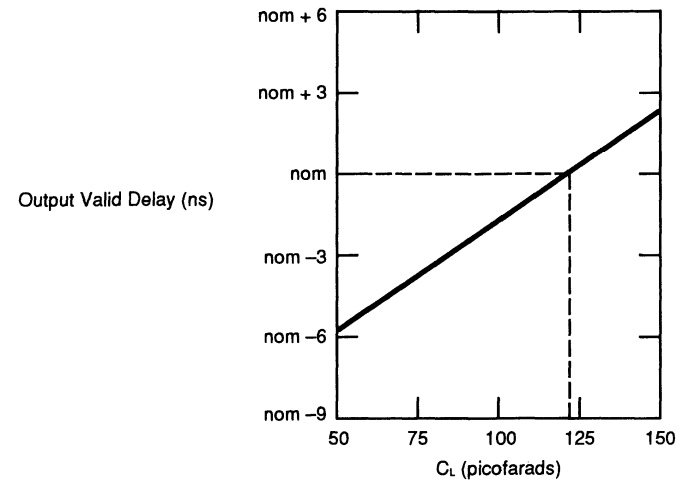
16305C–012



The second internal processor phase following RESET High-to-Low transition (provided t25 and t26 are met) is φ2.

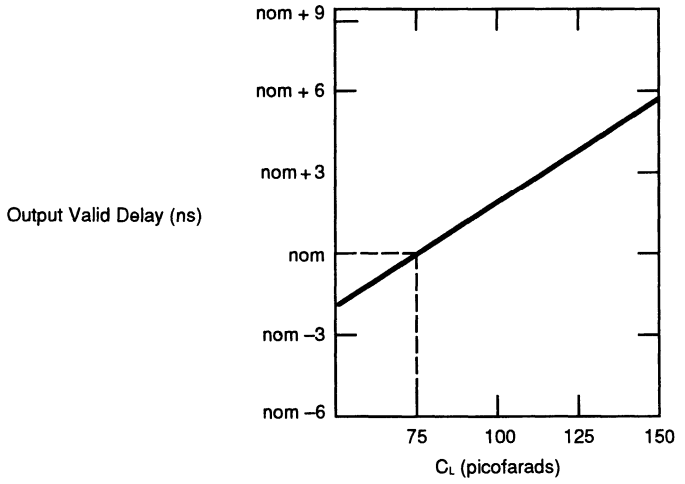
15021B–084

Figure 15. RESET Setup and Hold Timing and Internal Phase



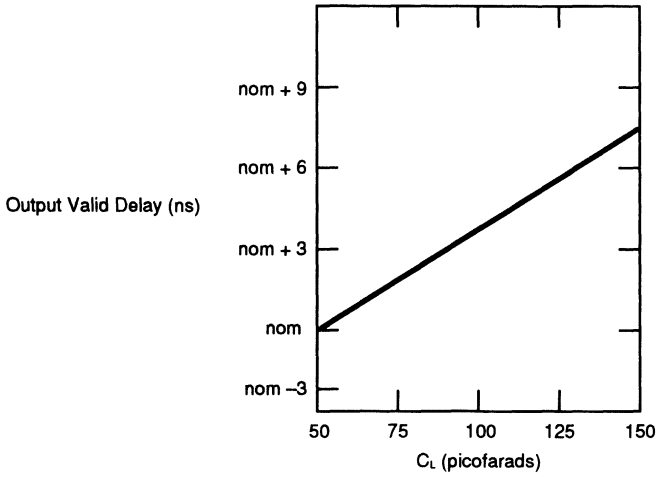
Note: This graph will not be linear outside the C_L range shown. 15021B–079

Figure 16. Typical Output Valid Delay Versus Load Capacitance at Maximum Operating Temperature (C_L=120 pF)



Note: This graph will not be linear outside the C_L range shown. 15021B–080

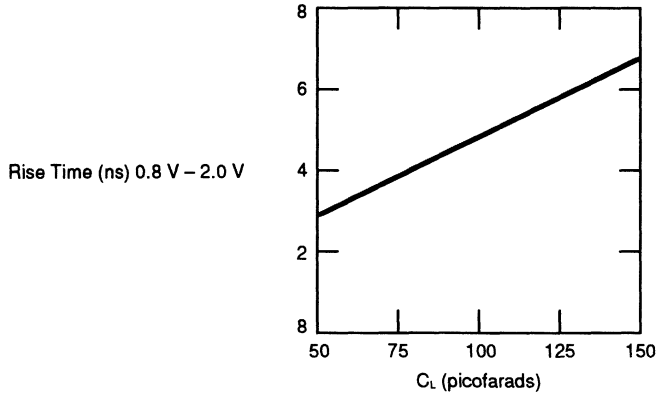
Figure 17. Typical Output Valid Delay Versus Load Capacitance at Maximum Operating Temperature (C_L = 75 pF)



Note: This graph will not be linear outside the C_L range shown.

15021B–081

Figure 18. Typical Output Valid Delay Versus Load Capacitance at Maximum Operating Temperature (C_L = 50 pF)



Note: This graph will not be linear outside the C_L range shown.

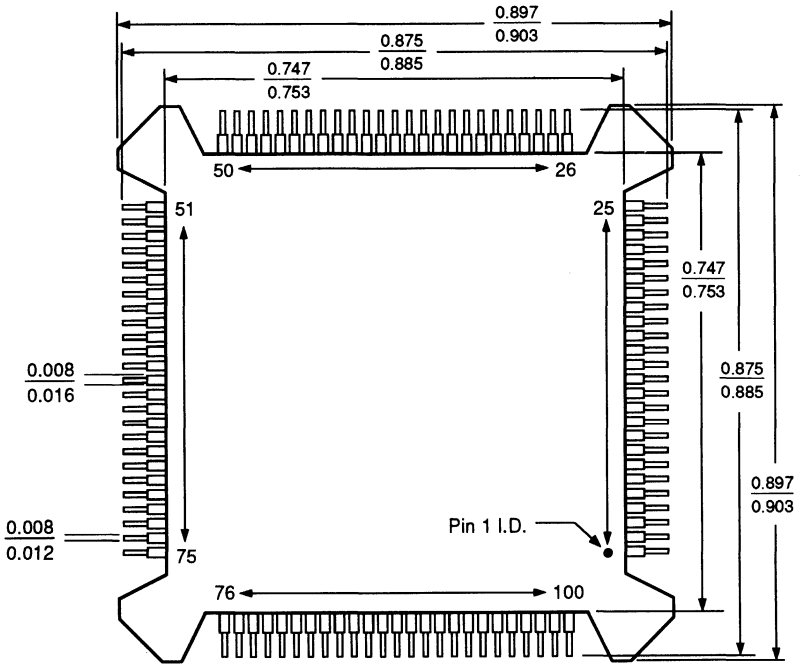
15021B–082

Figure 19. Typical Output Rise Time Versus Load Capacitance at Maximum Operating Temperature

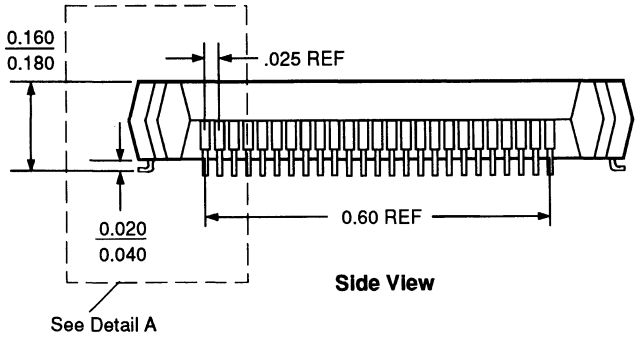
PHYSICAL DIMENSIONS

BSC is an ANSI standard for Basic Space Centering. All measurements are in inches unless otherwise specified (PQB 100 outer ring is measured in millimeters). For reference only.

PQB 100 (Plastic Quad Flat Pack, Trimmed-Formed)
(all measurements are in inches)



Top View

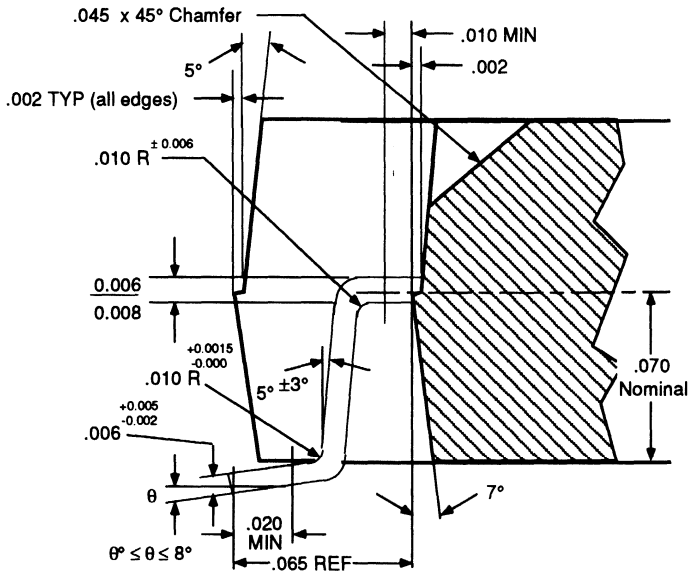


Side View

15679E
BX 44
3/31/92 SG

PHYSICAL DIMENSIONS (continued)

PQB 100 (Plastic Quad Flat Pack, Trimmed-Formed)



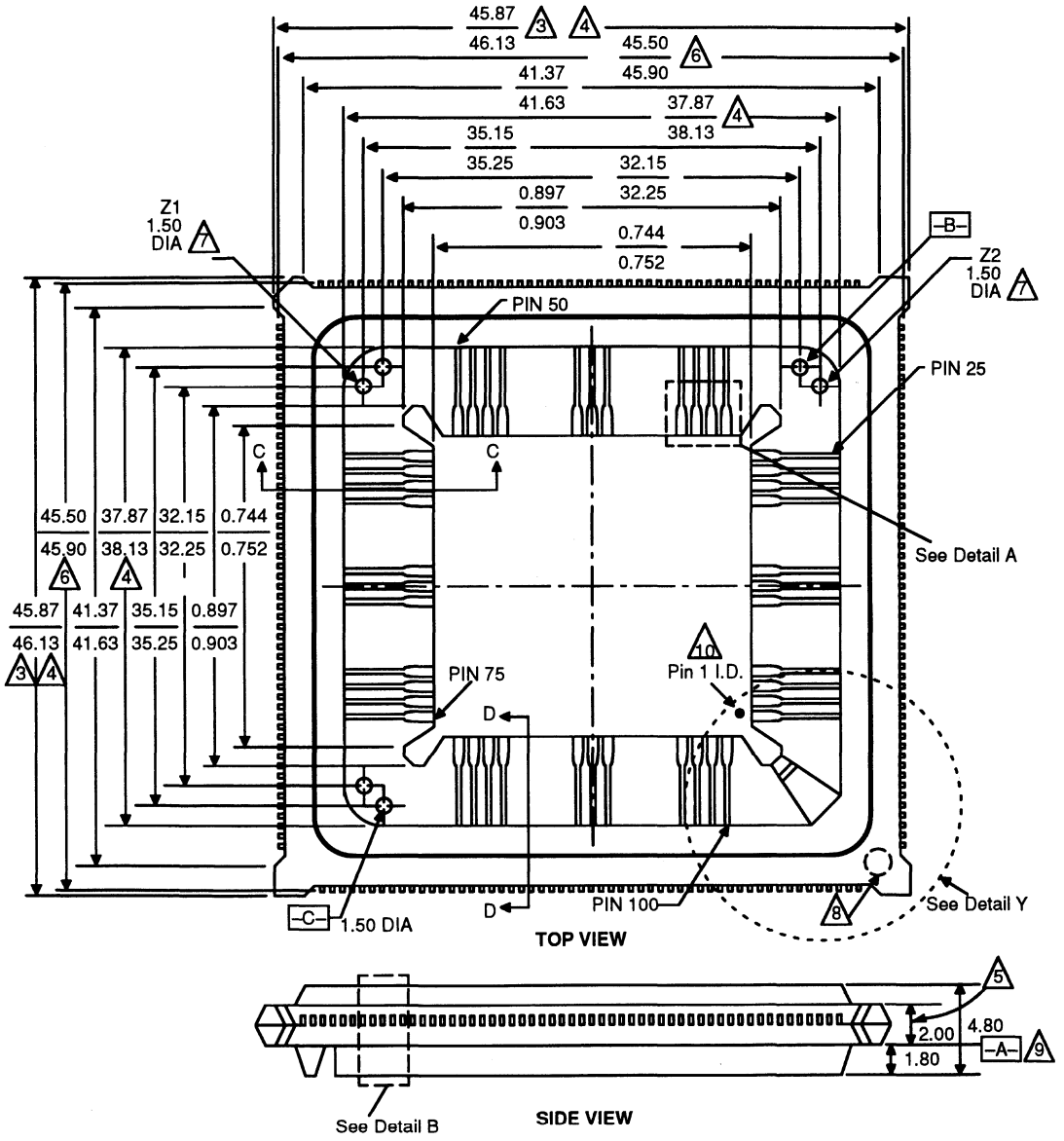
Detail A

Notes:

1. All dimensions are in inches unless otherwise specified.
2. Dimensions do not include mold protrusion.
3. Coplanarity of all leads will be within 0.004 inches measured from the seating plan. Coplanarity is measured per specification 06–500.
4. Deviation from Lead-tip true position shall be within ±0.003 inches.
5. Half span (center of package to lead-tip) shall be within ±0.085 inches.

PHYSICAL DIMENSIONS (continued)

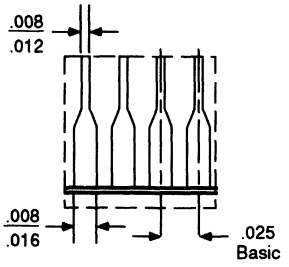
PQB 100—Plastic Quad Flat Pack with Molded Carrier Ring
(Inner device measured in inches; outer ring measured in millimeters)



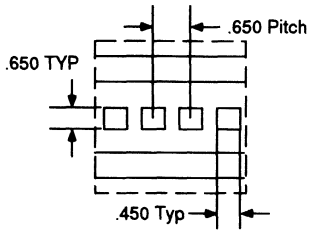
20008A
CJ83
03/17/93 SG

PHYSICAL DIMENSIONS (continued)

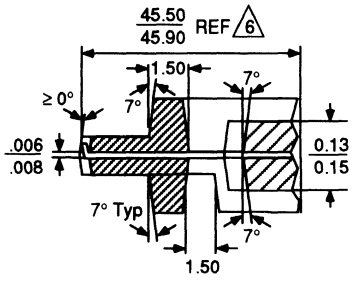
PQB 100—Plastic Quad Flat Pack with Molded Carrier Ring (continued)



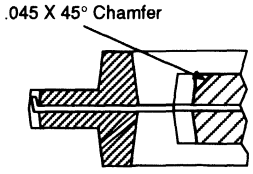
Detail A



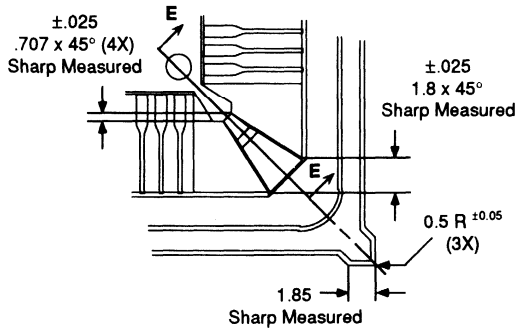
Detail B



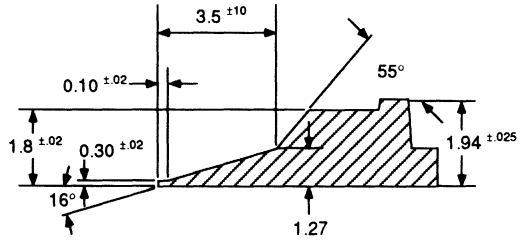
Section C-C



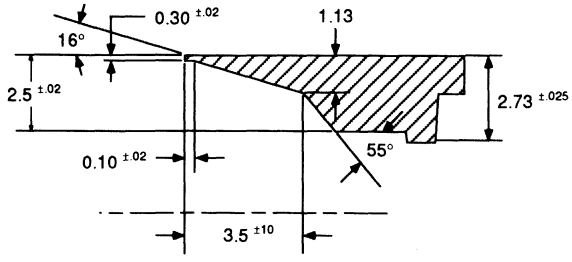
Section D-D



Detail Y



Top Gate



Bottom Gate

Section E-E

PHYSICAL DIMENSIONS (continued)

PQB 100—Plastic Quad Flat Pack with Molded Carrier Ring (continued)

Notes:

1. All dimensions and tolerances conform to ANSI Y14.5M–1982.
2. Controlling dimensions: package is measured in inches and ring is measured in millimeters.
3. These dimensions do not include mold protrusion. Allowable mold protrusion is 0.2 mm per side.
4. These dimensions include mold mismatch and are measured at the parting line.
5. Dimensions are centered about centerline of lead material.
6. These dimensions are from the outside edge to the outside edge of the test points.
7. There are six locating holes in the ring, –B– and –C– datum holes are used for trim form and excise of the molded package only. Holes Z1 and Z2 are used for electrical testing only.
8. This area is reserved for vacuum pickup on each of the four corners of the ring and must be flat within 0.025 mm. No ejector pins in this area.
9. Datum –A– surface for seating in socket applications.
10. Pin one orientation with respect to carrier ring as indicated.

Am386DXLV and Am386SXLV Microprocessors
Technical Reference Manual

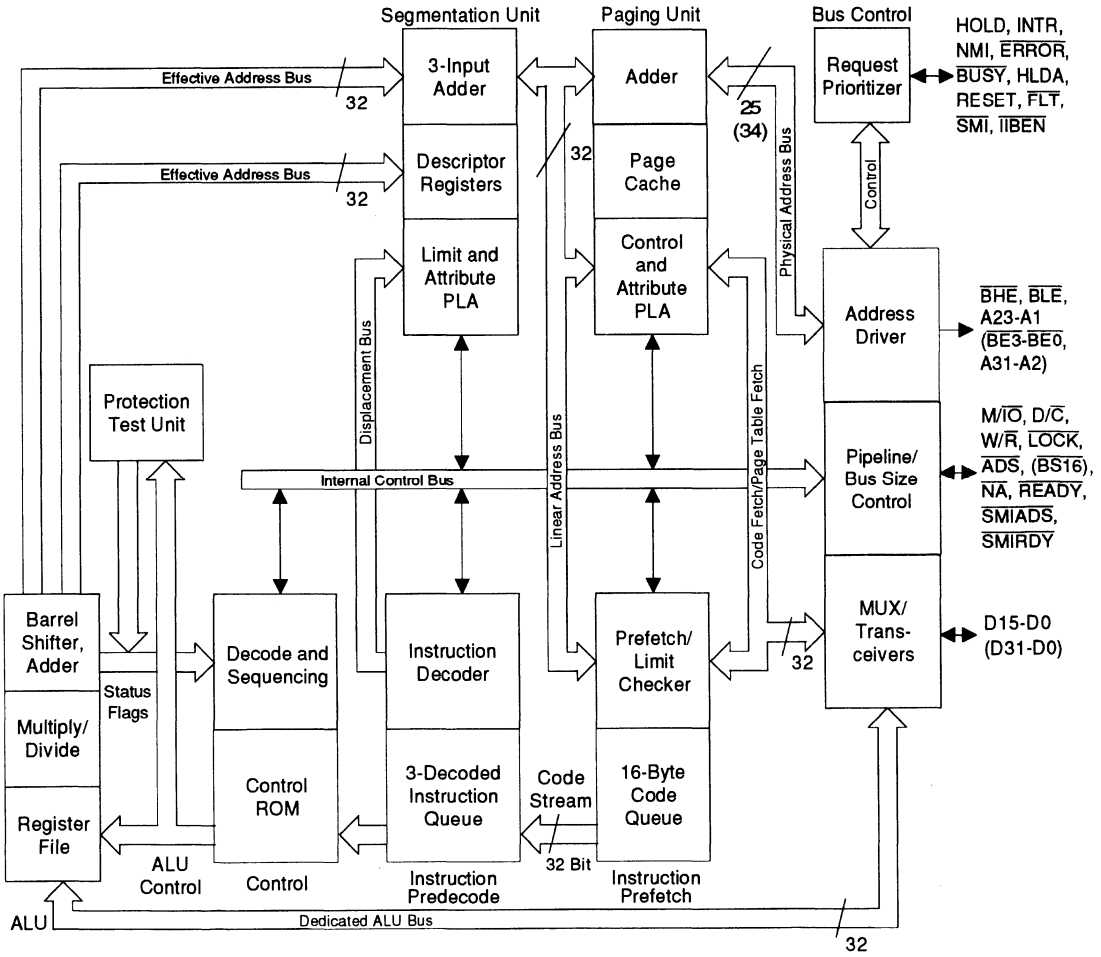
CHAPTER 1

INTRODUCTION

GENERAL DESCRIPTION

The Am386DXLV and Am386SXLV microprocessors are low voltage, true static derivatives of the Am386 microprocessors with System Management Mode support. These microprocessors are ideal for systems where battery life and system weight are major concerns. The System Management Mode allows such systems very flexible control over system peripherals.

Figure 1-1 CPU Block Diagram



Note: DX specific characteristics shown in ().

16944A-001

Low Voltage Features

The Am386DXLV and Am386SXLV microprocessors' lower operating voltage and true static design enable longer battery life and/or lower weight for portable systems. Lowering the operating voltage from 5.0 V to 3.3 V effectively halves CPU and core logic power consumption. Standby Mode of these microprocessors allows the clock frequency to be reduced to 0 MHz (DC) and retain full register contents. In Standby Mode, typical current draw is drastically reduced.

System Management Mode (SMM)

SMM is an advanced power management feature that allows interruption and resumption of normal CPU activity. These interrupts and resumes allow the system designer to implement very effective power management schemes. This mode of operation is implemented with an SMI interrupt (System Management Interrupt), new SMM related CPU instructions, and an $\overline{\text{IIBEN}}$ enable for an I/O trapping feature (I/O Instruction Break Enable). SMI is a non-maskable, higher priority interrupt than NMI and has its own separate code space. SMI can be coupled with the I/O Instruction Break feature to implement transparent power management of peripherals.

System Impact Summary

The Am386DXLV and Am386SXLV microprocessors embody the enabling technology for advanced portable systems. The features of low voltage, true static operation and SMM provide the system designer with new opportunities for innovation and system differentiation. The Am386DXLV and Am386SXLV CPU's are the ideal core for the next generation high-performance portable designs.

CHAPTER 2

SYSTEM MANAGEMENT MODE (SMM)

SMM OVERVIEW

Capabilities

SMM provides a mechanism that interrupts the processor operation and resumes the interrupted operation transparent to the operation system or application being run on the system. This service routine resides in its own System Management address space. Therefore, supporting logic can be designed to allow arbitrary interruption of the processor's activity to allow for the execution of other code tasks. One obvious application of SMM is portable system power management.

Due to the pipelined nature of the Am386DXLV and Am386SXLV processors, accurately interrupting processor code execution with single instruction granularity is normally very difficult. This difficulty is primarily a concern when trapping I/O instructions. For this reason SMM is implemented in the Am386DXLV and Am386SXLV microprocessors with I/O Instruction Break capability. This feature allows I/O instruction trapping implementation.

Pin Descriptions

The CPU interface for SMM consists of four pins dedicated to the SMM function. One pin, $\overline{SMI}$, is the System Management Interrupt input. Two pins, $\overline{SMIADS}$ and $\overline{SMIRDY}$, provide the control signals necessary for the separate SMI-mode memory space. The final pin, $\overline{IIBEN}$, allows for the enabling of non-pipelined I/O accesses to allow for I/O instruction trapping. The pin functions are defined as follows:

$\overline{SMI}$ — **System Management Interrupt** — Active Low I/O pin with active pull-up. This pin is the highest level interrupt input to the CPU.

$\overline{SMIADS}$ — **SMI Address Status** — Active Low three-state output pin. When active, this pin indicates that a valid bus cycle to the separate SMM memory space has begun. It also validates the values on the W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{BE3}$–$\overline{BE0}$, and A31–A2 pins (A23–A1, $\overline{BHE}$, and $\overline{BLE}$ in the case of the Am386SXLV CPU).

$\overline{SMIRDY}$ — **SMI Ready** — Active Low input pin. This input terminates the current bus cycle to the $\overline{SMIADS}$ initiated accesses in the same manner the $\overline{READY}$ pin does for the $\overline{ADS}$ initiated address accesses.

$\overline{IIBEN}$ — **I/O Instruction Break Enable** — Active Low input pin with active pull-up. When active, this pin enables the I/O instruction break feature. This feature disables execution pipelining for I/O instructions. If inactive, the feature is disabled and I/O cycles execute in a manner clock-for-clock compatible with the Am386DXL or Am386SXL processors.

Pin Locations

The four SMM interface pins are at locations identified as No Connects (NCs) in the Am386DX/DXL and Am386SX/SXL microprocessor products. The pin locations for the Plastic Quad Flat Pack (PQFP) packages are shown in the tables below. The pin locations of all other signals remain the same as the standard products. This includes the no-connect pins not taken up by the new SMM signals. All no-connect pins are reserved for future use.

The Am386DXLV and Am386SXLV microprocessors are available in PQFP packages. The SMM related pin locations are noted in Table 2-1 (for the Am386DXLV microprocessor), and Table 2-2 (for the Am386SXLV microprocessor).

Table 2-1 Am386DXLV CPU SMM Pin Locations

Name	PQFP Pin
$\overline{\text{SMI}}$	59
$\overline{\text{SMIADS}}$	37
$\overline{\text{SMIRDY}}$	36
$\overline{\text{IIBEN}}$	58

Table 2-2 Am386SXLV CPU SMM Pin Locations

Name	PQFP Pin
$\overline{\text{SMI}}$	43
$\overline{\text{SMIADS}}$	31
$\overline{\text{SMIRDY}}$	30
$\overline{\text{IIBEN}}$	29

Features

SYSTEM MANAGEMENT MODE

SMM is implemented through a high priority System Management Interrupt (SMI). SMI is non-maskable and higher in priority than NMI. An SMM-based system can be implemented with the Am386DXLV and Am386SXLV processors' special SMI interface pins. This interrupt method can be used to perform system management functions independent of processor operating mode (Real, Protected, or Virtual 8086 modes).

Activating the SMI invokes a sequence that saves the operating state of the processor into a separate SMM memory address space, independent of the main system memory. After the state is saved, the processor is forced into Real Mode and begins execution out of that separate address space at the processor reset address (address FFFFFFF0h for the Am386DXLV microprocessor, address FFFFF0h for the Am386SXLV microprocessor) where a jump to the SMM code is executed. This code performs its system management function and then resume execution of the normal system software by executing an SMM CPU state restore opcode sequence, which reloads the saved processor state and resume execution out of the main system memory space.

I/O TRAPPING

I/O trapping can be implemented with the I/O Instruction Break feature. I/O trapping allows the system to turn off peripherals when they are not needed. The I/O trapping-hardware can then intercept accesses to these peripherals by interrupting the instruction stream, turning on the peripheral, and then re-executing the trapped I/O instruction. When the signal $\overline{\text{IIBEN}}$ is active, the processor execution unit pauses upon execution of an I/O transfer until the end of the transfer cycle without affecting memory or register-related instructions. This pause allows non-pipelined interruption of the instruction via $\overline{\text{SMI}}$. After the SMM routine is complete, the I/O instruction can be re-executed and normal execution resumed. The I/O Instruction Break Enable signal is dynamic and can be enabled and disabled as the situation warrants. Thus, a system using this feature need only pause on I/O instructions when necessary.

SMM Instructions

There are three specific instructions for SMM operation: SMI, UMOV, and RES3.

SMI—CALL TO SYSTEM MANAGEMENT INTERRUPT PROCEDURE

Opcode	Instruction	Clocks	Description
F1	SMI	DX-325 SX-357	Call to System Management Interrupt Procedure

Operation

```
IF SMIE = 1
THEN ENTER SMM
ELSE
        enter DEBUG exception routine (INT 1)
END
```

Description When SMIE bit is set in Debug Control Register (DR7, bit 12) SMI forces the processor into SMM. If the SMIE bit is not set, the CPU executes an INT1 debug exception.

Flags Affected SMMS bit in the Debug Status Register is set (DR6, bit 12 = 1).

UMOV—MOVE DATA TO MAIN (USER) MEMORY

Opcode	Instruction	Clocks	Description
0F 10 /r	UMOV r/m8,r8	2/2	Move byte register to r/m byte
0F 11 /r	UMOV r/m16,r16	2/2	Move word register to r/m word
	UMOV r/m32,r32	2/2	Move dword register to r/m dword
0F 12 /r	UMOV r8,r/m8	2/4	Move r/m byte in main memory to byte register
0F 13 /r	UMOV r16,r/m16	2/4	Move r/m word in main memory to word register
	UMOV r32,r/m32	2/4	Move r/m dword in main memory to dword register

Note: /r indicates the ModR/M byte of the instruction contains both a register and an r/m operand. Clock counts may be greater if access is misaligned.

Operation DEST <– SRC;

Description UMOV copies the second operand to the first operand with all memory references being initiated with $\overline{ADS}$. UMOV functions like the MOV instruction with the exception that UMOV is an SMM instruction that references the normal memory system.

Flags Affected None

RES3—RESTORE ALL CPU REGISTERS

Opcode	Instruction	Clocks	Description
0F 07	RES3	DX-291 SX-366	Restore all CPU registers from memory

Operation CPU Registers <– ES:[EDI]

Description RES3 loads all CPU registers from a memory-based table referenced by ES:[EDI]. When in SMM, the CPU save state table is loaded with $\overline{SMIADS}$ initiated cycles. When not in SMM, the CPU save state table is loaded with $\overline{ADS}$ initiated cycles. A new execution context will be established in this process. Upon completion of the instruction, the CPU resumes execution of the instruction stream described by the CPU save state table. The function of the RES3 opcode may change for future processors.

Flags Affected All flags are loaded from the CPU save state table.

SMM FUNCTIONAL DESCRIPTION

Introduction

The execution of an SMI has four distinct phases: the initiation of the mode via an active $\overline{SMI}$ signal, a processor state save, execution of the SMM interrupt code, and a processor state restore (to resume normal operation). This process is shown in Figure 2-1.

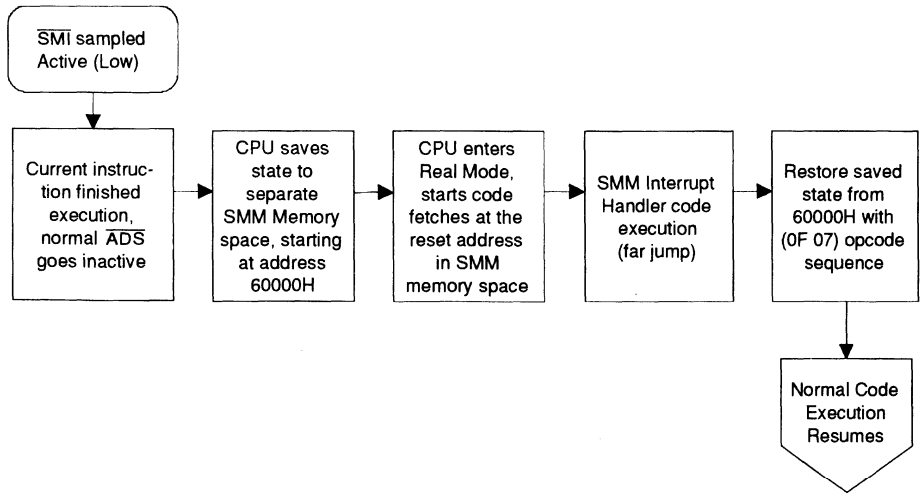
SMI Initiation

SMM is initiated by driving $\overline{SMI}$ active until the first active $\overline{SMIADS}$ pulse. The CPU drives the $\overline{SMI}$ pin active after the completion of the current operation. The active drive of the pin by the CPU is released at the end of the interrupt routine, following the last memory read of the stored save state.

An SMI cannot be masked by the CPU and is always recognized by the CPU, regardless of operating mode. This includes the Real, Protected, and Virtual 86 Modes of the processor.

SMI is the highest level external interrupt with precedence over both the NMI and INTR interrupts. SMIs cannot be nested. Another SMI request is not recognized until the completion of an SMI handler, when the CPU releases its drive of the SMI pin. Also upon entering SMM, NMI and INTR interrupts are not enabled. NMI requests are latched so processing of NMI's can be deferred until the SMM routine completes. INTR requests can be enabled by the EFLAGS register or the STI instruction. In the event of an exception, software interrupt (INT N or INT0), or INTR interrupt requests, the execution of the associated IRET instruction enables the recognition of pending NMI requests. In this case SMM software needs to comprehend the handling of NMI as well as standard INTR interrupts.

Figure 2-1 Complete SMM Execution Sequence



16944A-02

SMM can also be entered by setting the SMIE bit in debug register 7 and executing the soft SMI instruction. This instruction is the opcode 0F1h. Upon execution of this instruction, the system enters SMM as if a hardware SMI were detected.

Processor State Save

The first set of SMM bus transfer cycles after the CPU's recognition of an active $\overline{SMI}$ is the processor saving its state to an external memory in a separate address space from main system memory. There may be processor pre-fetch activity before entering SMM. The first SMM memory accesses are accomplished by using the $\overline{SMIADS}$ and $\overline{SMIRDY}$ pins for initiation and termination of bus cycles (instead of the $\overline{ADS}$ and $\overline{READY}$ pins). The addresses to which the CPU saves its state are 60000h–600CBh and 60100h–60127h. These are fixed address locations for each register saved. Note that each 32-bit register saved results in two 16-bit transfer cycles on the Am386SXLV CPU. A map of the save/restore state memory area is in Appendix A.

The state save transfer cycles are initiated with $\overline{SMIADS}$, and the bus cycle status pins indicate a memory data write: W/$\overline{R}$=1, D/$\overline{C}$=1, and M/$\overline{IO}$=1. These cycles must be terminated with $\overline{SMIRDY}$. The Am386DXLV processor accesses only SMM addresses as 32-bit accesses, ignoring $\overline{BS16}$ for SMM cycles.

A total of 114 data transfer cycles are required for the Am386SXLV CPU and 61 cycles for the Am386DXLV CPU to complete the save state operation: there are fifty-three 32-bit registers and eight 16-bit registers to be saved.

SMI Code Execution

After the processor state is saved to the separate SMM memory space, the execution of the interrupt routine code begins. The processor enters Real Mode and begins fetching code from the reset address in the separate SMM memory space. The processor state and register contents upon entering SMM are detailed in Appendix A. Typically, the first thing the interrupt routine code does is a jump to the Real-mode entry point for the SMM interrupt routine, which is also in SMM memory space.

The code fetch bus transfer cycles are initiated with $\overline{SMIADS}$, and the bus cycle status pins indicate an instruction fetch: W/$\overline{R}$=0, D/$\overline{C}$=0, and M/$\overline{IO}$=1. These cycles must be terminated with $\overline{SMIRDY}$. The Am386DXLV processor accesses only SMM addresses as 32-bit accesses, ignoring $\overline{BS16}$ for SMM cycles.

Any interrupt routine code can be executed within SMM. The SMM code can be located anywhere within the SMM address space, except for where the processor state is saved. If Protected Mode is enabled, SMM interrupt code has full access to the 4-Gb address space. I/O bus cycles, as a result of the IN, OUT, INS, and OUTS instructions, are directed to the normal address space, utilizing the normal $\overline{ADS}$ and $\overline{READY}$ bus interface signals. This facilitates code that manipulates system hardware registers through the standard I/O subsystem. A separate I/O space does not need to be implemented.

Processor State Restore

Returning to normal code execution in the main system memory, including restoring the processor operating mode, is accomplished by executing a CPU restore instruction RES3 (opcode 0Fh 07h). This code invokes a restore CPU state operation that reloads the CPU registers from the saved data in the memory space controlled by $\overline{SMIADS}$ and $\overline{SMIRDY}$ signals.

For the RES3 instruction to execute properly, the ES:EDI register pair must point to physical address 60000h where the interrupted CPU state is saved. The execution can

be accomplished in Real Mode by loading ES with 6000h and clearing the EDI register. Then RES3 should be executed to start the restore state operation. After completion of the restore state operation, the $\overline{SMI}$ pin is deactivated by the CPU and normal code execution continues at the address specified by the contents of the CS:EIP register in the processor save state.

The state restore transfer cycles are initiated with $\overline{SMIADS}$, and the bus cycle status pins indicate a memory data read: W/$\overline{R}$=0, D/$\overline{C}$=1, and M/$\overline{IO}$=1. These cycles must be terminated with $\overline{SMIRDY}$. There are 114 data transfer cycles required for the Am386SXLV processor and 61 cycles required for the Am386DXLV processor to complete the processor state restore. The Am386DXLV processor will only access SMM addresses as 32-bit accesses, ignoring $\overline{BS16}$ for SMM cycles.

If the CPU was powered down for power management purposes while in SMM, the RES3 instruction can be executed in Normal Mode after power up. This execution can be done if the system logic provides normal access to the saved CPU state in SMM memory space. The ES:EDI register pair should point to the Normal Mode address of the saved state and the RES3 opcode (0Fh 07h) can be executed. In this case the CPU executes the restore state with $\overline{ADS}$ initiated cycles.

I/O INSTRUCTION BREAK

Functional Description

The operation of the I/O Instruction Break feature allows external system logic to interrupt the processor on I/O instruction boundaries. This interruption is achieved by the system logic driving $\overline{IOBEN}$ active and providing logic to assert $\overline{SMI}$ after recognition of an I/O bus cycle. This causes the CPU to enter SMM immediately after executing the I/O instruction if $\overline{SMI}$ was asserted.

Internally, I/O Instruction Break pauses normal execution at the end of the I/O instruction but before execution of the following instruction. Assertion of $\overline{IOBEN}$ causes the CPU execution unit to pause on an I/O bus cycle until it receives the final $\overline{READY}$ associated with the I/O request. In the case of misaligned requests from the execution unit, more than one physical I/O bus cycle can occur. After the final $\overline{READY}$ is received, the I/O instruction is completed.

When an I/O transfer that needs to be trapped is issued with $\overline{IOBEN}$ active, the system logic must assert $\overline{SMI}$ for more than three CLK2 periods before asserting $\overline{READY}$ to terminate the cycle. The CPU immediately goes into SMM without executing the next instruction. Failure to meet the required setup time results in a standard SMM entry at some point after the execution of the I/O instruction (the I/O instruction will not be trapped).

The execution of coprocessor cycles is not affected, therefore coprocessor cycles cannot be trapped as other I/O cycles.

The "I/O breakable" instructions are listed below. With I/O Instruction Break enabled, the execution clock count for these instructions increases. However, coprocessor I/O instructions executions do not slow down. No memory or register instruction execution times are affected.

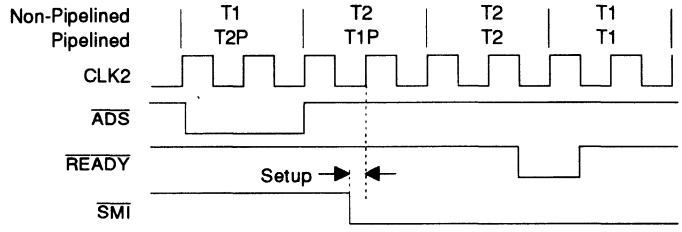
Instruction	Opcodes
IN	E4, E5, EC, ED
OUT	E6, E7, EE, EF
INS	6C, 6D
OUTS	6E, 6F
REP INS	F3 6C, F3 6D
REP OUTS	F3 6E, F3 6F

For the I/O string instructions (with or without the REP prefix), the assertion of $\overline{SMI}$ marks the last executed I/O transfer pair. With $\overline{IIBEN}$ active, an $\overline{SMI}$ asserted on the first I/O read or write does not usually result in another following I/O transfer cycle until after the SMI code is executed. An $\overline{SMI}$ asserted on the I/O read part of a REP INS instruction has a corresponding memory write cycle before entering SMM. The one exception to this function is the REP OUTS instruction. The operation of REP OUTS can, in some cases, execute two memory read-I/O write pairs before trapping depending upon the state of the CPU when $\overline{SMI}$ is asserted. In this case an overrun flag is set in the CPU save state so the need for corrective action can be detected by the SMM routine.

The I/O Instruction Break feature is implemented such that asserting $\overline{SMI}$ properly traps an I/O Instruction if the $\overline{SMI}$ pin is sampled active (Low) three CLK2 edges before the CLK2 edge that ends the I/O cycle with an active $\overline{READY}$ pin. Therefore, to implement I/O instruction break for a particular I/O device, the access for that device must be at least a 1 CPU wait state cycle. This timing applies to both non-pipelined and pipelined cycles. (see Figure 2-2). Note that $\overline{SMI}$ is set up to the rising edge of CLK2 on the beginning of the second phase of the processor state.

Depending on the state of the pre-fetch queue at the time $\overline{SMI}$ is asserted, instruction fetch cycles might occur on the normal $\overline{ADS}$ interface before the SMM save state begins with the assertion of $\overline{SMI}$$\overline{ADS}$. However, the pre-fetched code is not executed.

Figure 2-2 I/O Instruction Break $\overline{SMI}$ Timing



16944A-03

SMM HARDWARE INTERFACE

Pin Behavior

$\overline{SMI}$

System Management Interrupt is an active Low I/O pin with an active pull-up. This pin is the highest level interrupt input to the CPU. In order to initiate SMM, $\overline{SMI}$ should be driven Low by an open collector driver until the first $\overline{SMIADS}$ pulse occurs to ensure recognition. The signal should then be released. In input mode, $\overline{SMI}$ must be synchronous to CLK2. When the interrupt is recognized and service has begun, the CPU drives $\overline{SMI}$ active to indicate that the SMI service routine is executing. The CPU drives $\overline{SMI}$ inactive and then releases it to the internal CPU pull-up upon completing the CPU restore state process. The CPU drives $\overline{SMI}$ High for two CLK2 periods and then releases $\overline{SMI}$ to the internal weak pull-up after the saved state is reloaded. The $\overline{SMI}$ pull-up is active during RESET and whenever the signal is not driven active by the CPU. The pull-up is disabled when the CPU is driving $\overline{SMI}$ to minimize CPU power consumption. Note that $\overline{SMI}$ is not floated during HOLD states while in SMM.

$\overline{SMIADS}$

SMI Address Status is an active Low three-state output pin. When active, this pin indicates that a valid bus cycle to the separate SMI memory space has begun. $\overline{SMIADS}$ also validates the values on the W/$\overline{R}$, D/$\overline{C}$, M/$\overline{IO}$, $\overline{BE3}$–$\overline{BE0}$, and A31–A2 pins (A23–A1, $\overline{BHE}$, and $\overline{BLE}$ in the case of the Am386SXLV CPU). The function of $\overline{SMIADS}$ is analogous to $\overline{ADS}$. This is a three-state output which floats during a bus HOLD cycle, as indicated by an active HLDA pin.

$\overline{SMIRDY}$

SMI Ready is an active Low input pin with an internal pull-up. This input terminates the current bus cycle to $\overline{SMIADS}$ initiated accesses in the same manner that the $\overline{READY}$ pin terminates $\overline{ADS}$ initiated accesses. The signal $\overline{SMIRDY}$ is a CPU input that is synchronous to CLK2. $\overline{SMIRDY}$'s function is analogous to $\overline{READY}$; however, $\overline{SMIRDY}$ and $\overline{READY}$ may not be sourced from the same signal.

$\overline{IIBEN}$

I/O Instruction Break Enable is an active Low input pin with an active pull-up. $\overline{IIBEN}$ is an asynchronous CPU input that is internally synchronized by the CPU to CLK2 and must be valid for several CLK2 pulses to be recognized. When active, this pin enables the I/O instruction break feature, and disables execution pipelining for I/O instructions. If inactive, the feature is disabled and I/O cycles execute in a manner clock-for-clock compatible with the Am386DXL or Am386SXL processors. This pin can be changed dynamically to enable or disable I/O Instruction Break Enable as required by the system. To ensure that the I/O cycle is trapped, $\overline{IIBEN}$ must be active prior to the assertion of $\overline{SMI}$ on the I/O cycles. Therefore, if $\overline{IIBEN}$ is to be changed dynamically, a jump should be made to flush the instruction queue after transitioning $\overline{IIBEN}$ from inactive to active. The $\overline{IIBEN}$ pull-up is active during RESET and whenever the signal is not driven active low by the system. Once the CPU detects $\overline{IIBEN}$ being driven by the system, the active pull-up is disabled until the next active RESET pulse or until $\overline{IIBEN}$ is sampled High. Therefore, $\overline{IIBEN}$ must be driven at all times if the I/O Break Enable feature is used. In systems not using the I/O Instruction Break feature, the pin does not need to be connected.

Bus Cycles

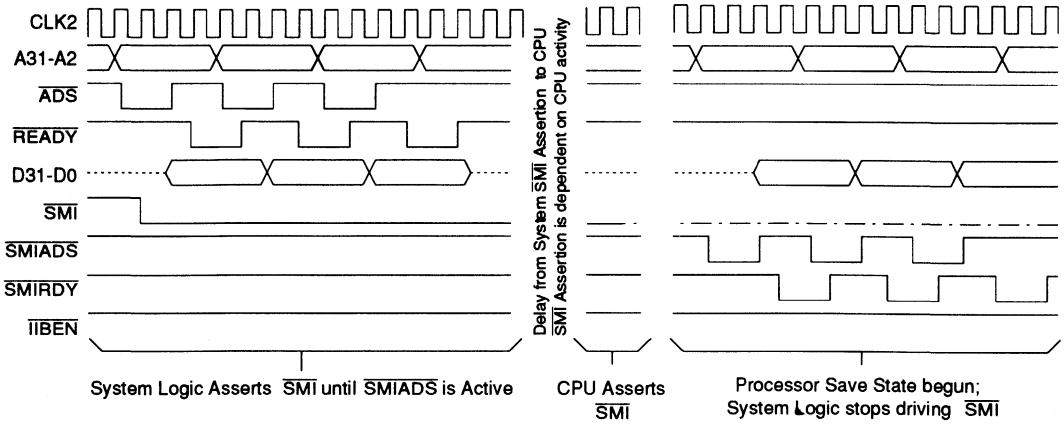
Bus cycles for the CPU during the SMM sequence are much like conventional CPU cycles. SMM memory transactions are requested and terminated with the SMM specific control lines ($\overline{\text{SMIADS}}$, $\overline{\text{SMIRDY}}$). Processor Save State and SMM code execution always occur with the SMM specific control lines. I/O transactions take place in the standard CPU I/O space and therefore use standard control lines ($\overline{\text{ADS}}$, $\overline{\text{READY}}$) to request and terminate the cycles. Special cycles such as interrupt acknowledge, HALT, and shut down cycles are also indicated by activity on the standard control lines.

The state of $\overline{\text{NA}}$ is ignored during SMM code fetches because only non-pipelined cycles can be generated in SMM. Because of this, the standard memory system should not be allowed to perform pipelined cycles while $\overline{\text{SMI}}$ is active. Due to pipelined $\overline{\text{ADS}}$ timing, memory transfers to/from non-pipelined SMM and a pipelined standard memory could have unpredictable results. Therefore, $\overline{\text{NA}}$ must not be active when in SMM. $\overline{\text{NA}}$ should be driven inactive at the receipt of the first $\overline{\text{SMIADS}}$ and held inactive until the processor drives $\overline{\text{SMI}}$ inactive.

All SMM address space cycles made by the Am386DXLV processor are 32-bit accesses. The 32-bit processor ignores $\overline{\text{BS16}}$ during cycles directed to the SMM address space. The Am386DXLV processor recognizes $\overline{\text{BS16}}$ during accesses to the normal address space. While a double word address space must be implemented, it should be noted that word and byte accesses, as required by instructions, function properly utilizing the byte enables.

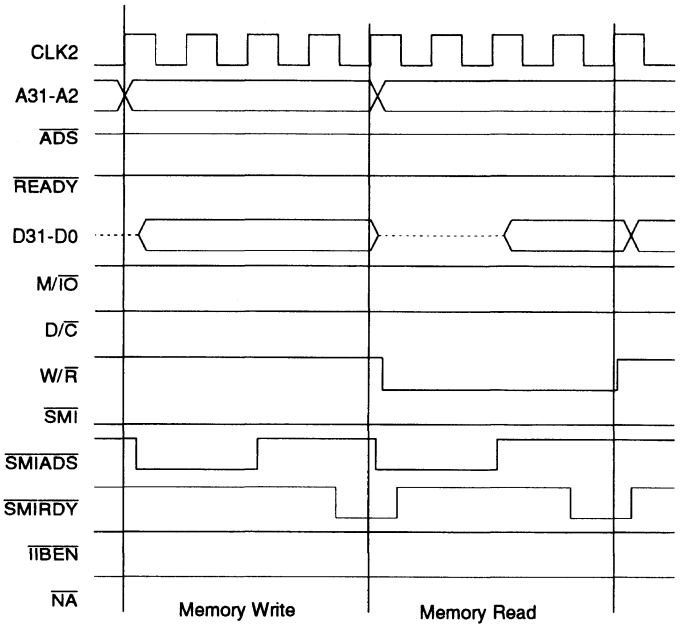
Figures 2-3 through 2-6 illustrate typical SMM bus cycles. The cycle sequence executed as the processor enters SMM is shown in Figure 2-3. Memory and I/O access while in SMM are detailed in Figures 2-4 and 2-5. Access to standard system memory from SMM with the UMOV instruction is illustrated in Figure 2-6. Figure 2-7 shows the processor exiting SMM and resuming the interrupted instruction sequence.

Figure 2-3 $\overline{\text{SMI}}$ Assertion, Start of CPU Save State



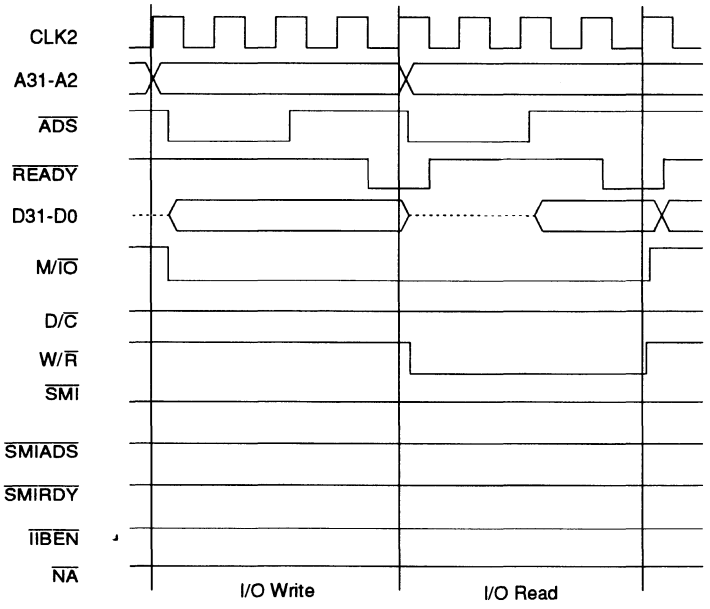
16944A-004

Figure 2-4 SMM Memory Write and Read



16944A-05

Figure 2-5 SMM I/O Write and Read



16944A-06

Figure 2-6 SMM UMOV Example

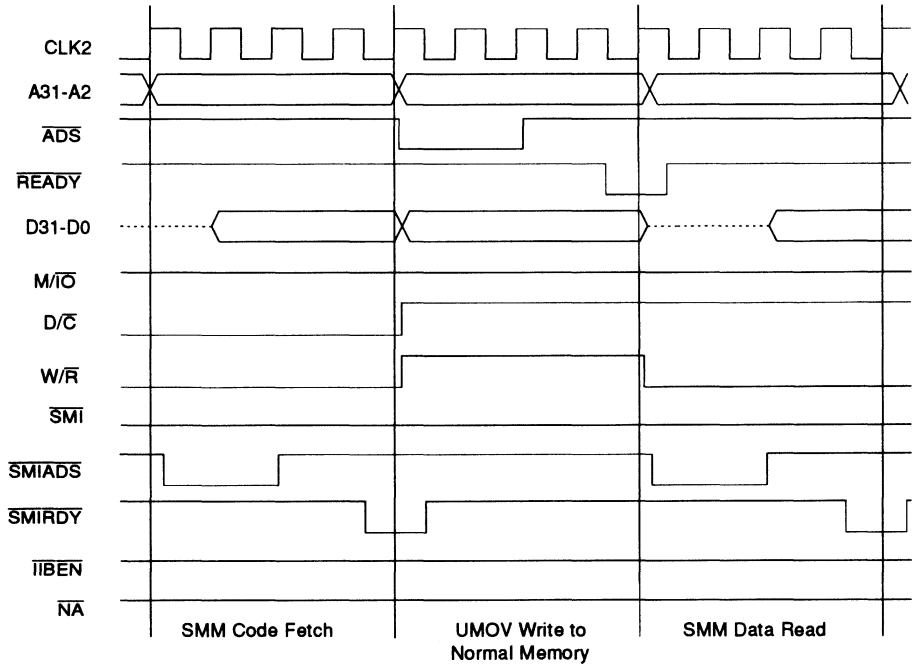
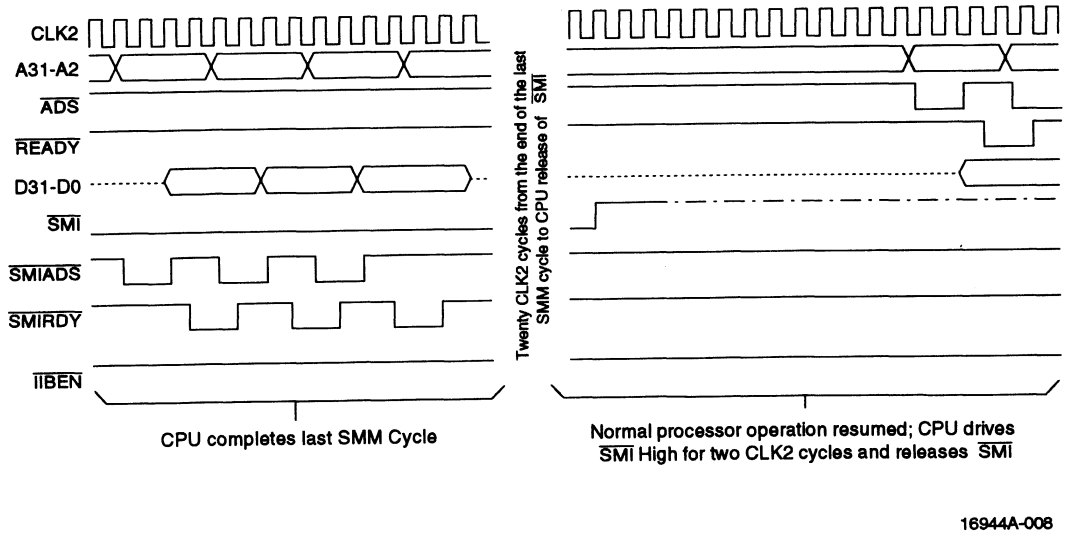


Figure 2-7 End of CPU Restore State



$\overline{SMI}$, Halt, and Shut Down

$\overline{SMI}$ can be used to interrupt the CPU in a Halt State. Upon return, the CPU will continue execution at the next instruction beyond the HALT. If SMM transparency is desired, the EIP must be set back to the HALT instruction before restore is executed. Attempts to interrupt a Shut Down state with $\overline{SMI}$ have inconsistent results and are not recommended.

If HALT or Shut Down cycles are generated while in SMM, they are generated with the standard $\overline{ADS}$ signal.

Hold, Reset, and Interrupts

If HOLD is asserted while in SMM, the processor completes the current cycle, floats its bus pins, and issues HLDA active, as in normal operation. The state of the SMM related output pins is shown in Table 2-3. SMM related input pins should maintain their state during the HOLD state. If HOLD is asserted between the time $\overline{SMI}$ is asserted and the first $\overline{SMIADS}$ is initiated, the system logic should maintain $\overline{SMI}$ active until it is recognized. RESET may be asserted at any time without regard to the processor's SMM state. The state of SMM related output pins and the required state of the SMM related input pins is given in Table 2-4.

Table 2-3 SMM Pin State during HOLD State

$\overline{SMI}$	Low
$\overline{SMIADS}$	Three-State

Table 2-4 SMM Pin State during RESET

$\overline{SMI}$	Not driven if not in SMM, may be driven High for two CLK2 periods if in SMM
$\overline{SMIADS}$	High
$\overline{SMIRDY}$	High (unless driven Low by system logic)
IIBEN	High (unless driven Low by system logic)

Upon entering SMM, interrupts are disabled and $\overline{NMI}$ is masked. Therefore, if INTR is asserted, the interrupt is not recognized until SMM is exited. If an NMI is asserted while the CPU is in SMM, the NMI is latched and serviced upon exit of SMM. Because $\overline{SMI}$ is the highest priority interrupt in the system, it is recommended that other interrupts not be serviced so that the interrupt priority hierarchy is preserved and the operation of SMM is transparent to system software.

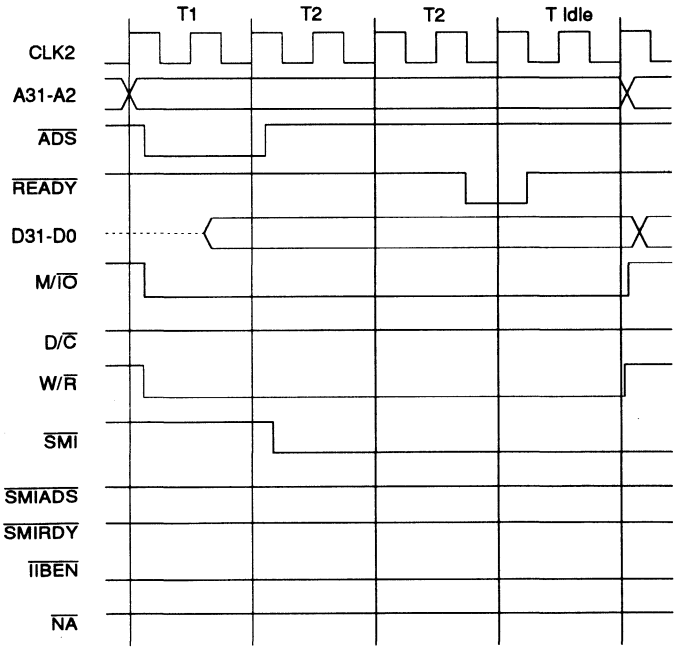
If so desired, INTR generated interrupts can be serviced by directly setting the IF bit in the EFLAGS register or by executing the STI instruction. If interrupts are to be serviced while in SMM, interrupt acknowledge cycles are initiated with $\overline{ADS}$ and interrupt table references are initiated with $\overline{SMIADS}$. Therefore, an SMM specific interrupt table must be created in SMM address space for the interrupts requiring service.

When servicing interrupts or exceptions, the execution of the IRET instruction enables the recognition of NMI requests. One possible strategy for the SMM code is to disable NMI in external system logic, a common feature in PC architecture systems. Therefore, if interrupts or exceptions are serviced while in SMM, the system designer must account for all possible INTR and NMI requests.

I/O Trapping

CPU cycles with I/O Instruction Break Enabled ($\overline{\text{IIBEN}}$ active) are not physically different in terms of cycle definition. The cycle simply pauses the execution unit of the CPU until the bus unit can complete the requested I/O cycle. The resulting bus traffic is different than normal traffic only because there are more idle cycles than normal and code pre-fetch cycles may appear earlier relative to data cycles. Figure 2-8 illustrates a typical I/O instruction break. In Figure 2-8, no more instructions will be executed before the CPU enters SMM. However, Bus Unit pre-fetch cycles are possible before the CPU enters SMM.

Figure 2-8 Typical I/O Instruction Break



16944A-09

SMM Software Features

There are several features of the SMM function that provide support for special operations during the execution of the system's software. These features involve the execution of reserved opcodes to invoke specific SMM related operations.

Processor Save State

Upon entering SMM, the processor saves its entire state in the SMM address block from 60000h–60127h. The complete use of this space is detailed in Appendix A. One detail of the Save State which the SMM routine regularly checks in I/O trapping applications is the REP OUTS overrun flag. This overrun flag is contained in bit 0 of the location 6006Ch (see Table 2-5). If the overrun flag is set, no overrun has occurred. If the overrun flag is cleared, a REP OUTS overrun has occurred (that is, two pairs of memory-I/O accesses have occurred). In the case of an overrun, the SMM routine must alter the affected registers to replay two cycles instead of one in order to ensure proper re-execution of the trapped I/O instruction.

Table 2-5 REP OUTS Overrun Flag

6006Ch Bit 0	0 = Overrun has occurred 1 = Normal Operation

Processor Modes

Processor modes can be changed while in SMM. Register use is the same as non-SMM operation. For Protected Mode the descriptor tables must be built in SMM memory space. Otherwise, operation is the same as in non-SMM applications. Example code for changing to Protected Mode while in SMM can be found in Appendix C.

Page Mode while in SMM is of limited usefulness. All page loads come from the normal memory and therefore are not recommended while in SMM.

Memory Transfers To Main System Memory

While executing an SMI routine, the interrupt code can initiate memory data reads and writes to the main system memory using the normal $\overline{ADS}$ and $\overline{READY}$ pins. This initiation is accomplished by using reserved opcodes that are special forms of the MOV instruction (called UMOV). The UMOV opcodes can move byte, word, or double word register operands to or from main system memory. Multiple data transfers using the normal $\overline{ADS}$ and $\overline{READY}$ pins as well as $\overline{SMIADS}$ and $\overline{SMIRDY}$ pins will occur if the operands are misaligned relative to the effective addresses used. The opcode formats are shown in Table 2-6.

The UMOV instruction can use any of the 386 addressing modes, as specified in the ModR/M byte of the opcode. Note that the 16- and 32-bit versions are the same opcodes with the exception of the 066h operand size prefix. Table 2-6 illustrates the UMOV opcode and encoded examples. Assembler macros examples can be found in Appendix B. The data in Table 2-6 assumes a default operand and address sizes of 16 bits. Using 32-bit Protected Mode changes the use of prefixes.

Table 2-6 UMOV Opcode Form

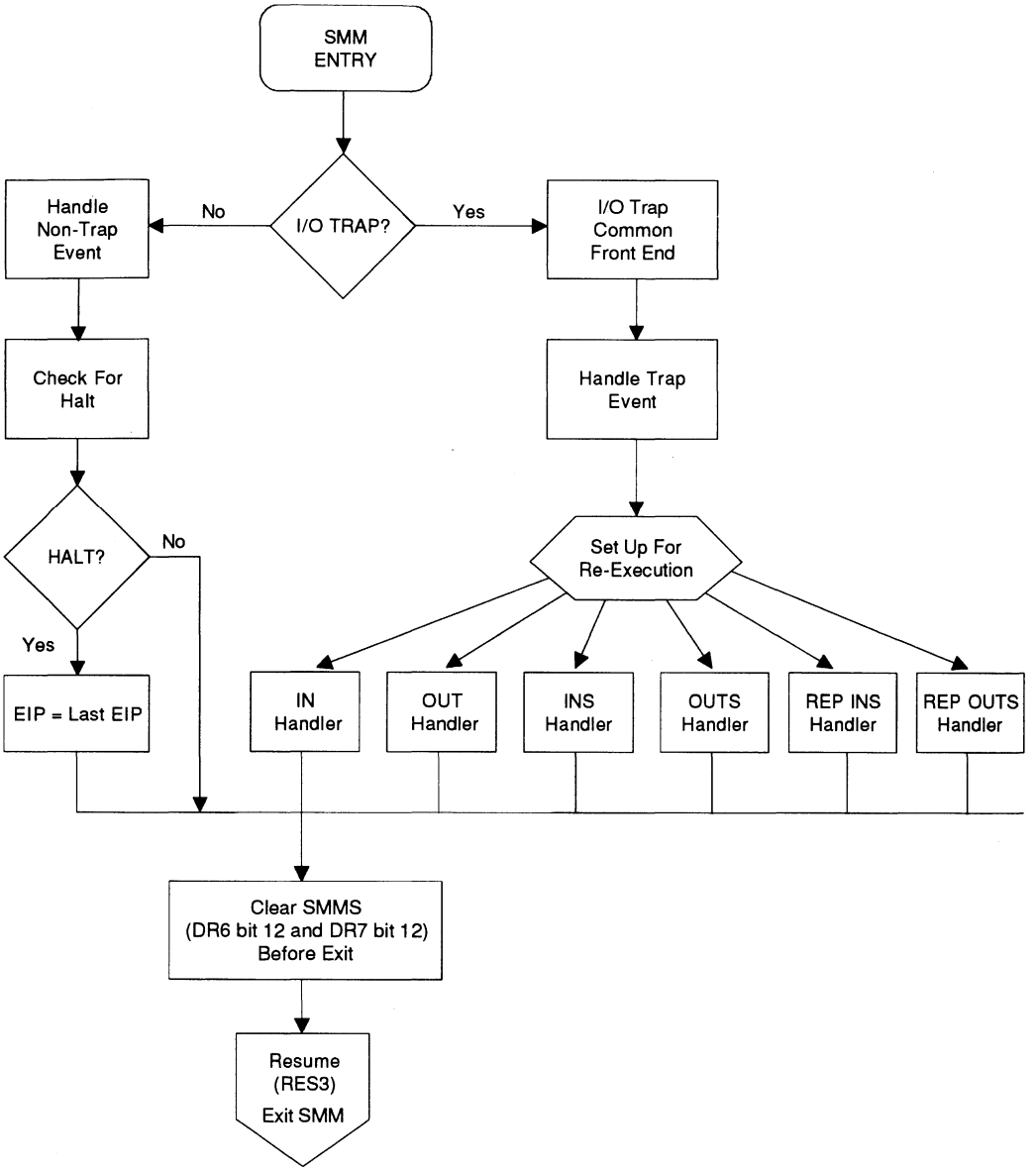
Opcode	Instruction	Description	Encoded Example
0F 10 ModR/M	UMOV r/m8,r8	Move byte register to r/m byte	0F 10 00 ; umov [bx+si],al
0F 11 ModR/M	UMOV r/m16,r16	Move word register to r/m word	0F 11 00 ; umov [bx+si],ax
0F 11 ModR/M	UMOV r/m32,r32	Move dword register to r/m dword	66 0F 11 00 ; umov [bx+si], eax
0F 12 ModR/M	UMOV r8,r/m8	Move r/m byte to byte register	0F 12 00 ; umov al,[bx+si]
0F 13 ModR/M	UMOV r16,r/m16	Move r/m word to word register	0F 13 00 ; umov ax,[bx+si]
0F 13 ModR/M	UMOV r32,r/m32	Move r/m dword to dword register	66 0F 13 00 ; umov eax,[bx+si]

(Note: These encodings assume a default size of 16 bits.)

SMM Driver And Code Strategy

The basic function of the SMM service routine can be diagrammed as shown in Figure 2-9. The first phase of the SMM routine is to determine why it was invoked. SMM interrupts are categorized by two separate types: non-execution system events (e.g., time-out of a system timer) and, I/O trapped instructions. In both cases, the SMM routine must query the system resources to determine which event requires service.

Figure 2-9 SMM Service Routine Flow Chart



16944A-10

The second phase of SMM code execution is the service phase. The path chosen on the flow chart depends on the determination of the SMI event source. The non-I/O trapped events are primarily system dependent with only one special instruction case to be handled. For I/O trapped events, it is assumed that the interrupted instruction requires re-execution to properly resume operation. This could be due to an I/O operation directed toward a peripheral which is powered down. In this case, the interrupted instruction must be examined and appropriate action taken to re-execute it properly.

Depending on the implementation of the system logic, multiple SMM interrupt sources may be pending. The SMM code may need to poll for any other possible sources before exiting the SMM routine.

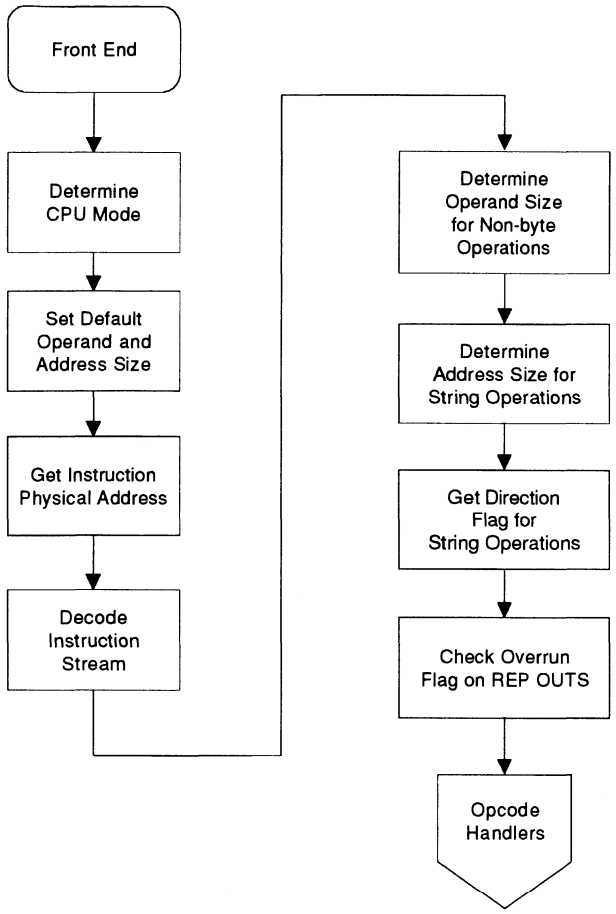
Service Phase

Non-I/O Trapped Based SMI—When the SMI routine is involved without I/O trapping, only the RES3 instruction is required to return to normal mode. In this case, before issuing the resume command, the SMM code must check the interrupted instruction stream to determine if the processor was interrupted in a HALT state. In the case of an interrupted HALT condition, (E)IP in the save state table must be set to the HLT instruction (EIP <= Last EIP) to prevent the processor from incorrectly resuming at the next instruction.

I/O Trapped SMI Events—In the event of trapped I/O instructions, the SMM routine may need to replay the trapped I/O instruction for proper task resumption. To accomplish this, the SMM code must examine both the CPU save state and the trapped I/O instruction to determine what system action must be taken and to prepare for the I/O instruction re-execution. This common front end code is diagramed in Figure 2-10.

The first task to be done is to determine the CPU mode at the time of the SMI. This can be determined by examining the CPU save state as detailed in the following pseudo-code example. It should be noted that the CPU mode is not specifically needed for instruction re-execution. However, the SMM code does need to know if paging was enabled to be able to determine the physical address of the trapped instruction.

Figure 2-10 I/O Trap Front End Flow Chart



16944A-11

DETERMINING THE CPU MODE:

```
IF NOT PROTECTED (CR0 Bit), Loc 60000H) THEN  (*Note: it is not re-
   MODE = REAL;                                quired to determine the
ELSE                                           exact mode of the proces-
   IF NOT VM (EFLAG Bit 17, Loc 60004H) THEN   sor to prepare for an in-
      MODE = PROTECT:                          struction re-execution.
   ELSE                                        This is shown to illustrate
      MODE = VIRT86;                           the process if other SMM
   ENDIF;                                      code requires it.*)
ENDIF;
IF PG=1 (CR0 Bit 31, Loc 60000H) THEN          (*We need to know if
   PAGING = ON;                                paging was enabled*)
ELSE
   PAGING = OFF;
ENDIF;
```

The next task is to set the default operand and address sizes to facilitate the proper decoding of the interrupted instruction's operand and address fields. This is accomplished by examining the code segment attributes from the CPU save state as shown in the following example.

SETTING DEFAULT SIZES:

```
IF (D = 0) (CS ATTR Bit 22, Loc 600B4H) THEN
   DEFOPERAND = 16;
   DEFADDR = 16;
      ELSE
   DEFOPERAND = 32;
   DEFADDR = 32;
ENDIF;
```

To access the interrupted instruction, the SMM code must now get the instruction's physical address. This is a simple calculation if the CPU was not in Page Mode. If paging was enabled the situation is more complicated since the instruction's physical address must be calculated from information in the CPU save state and the normal memory-based page tables. Again, data from the CPU save state is used as shown in the following code.

GET THE INSTRUCTION'S PHYSICAL ADDRESS:

```
CSBASE = Loc 600B8H (32-bit value); (*valid for Real,
OFFSET = LASTEIP (Loc 60124H);       Protect, and V86 Modes*)

LINEARADDR = CSBASE + OFFSET;

IF PAGING = OFF THEN
   PHYSICALADDR = LINEARADDR;
ELSE
   DIRENTRYADDR = PDBR(Actual CR3 Value) + (LINEARADDR[31:22]) * 4;
   PAGEENTRYADDR = (DIRENTRY AND FFFFF000H) + (LINEARADDR[21:12]) * 4;
   PHYSICALADDR = (PAGEENTRY AND FFFFF000H) + (LINEARADDR[11:0]);
ENDIF;

MAXINSTRLENGTH = 5 (bytes);          (*the longest I/O instruction is
                                     5 bytes w/o redundant overrides*)
```

Now that all the information needed to access the instruction has been extracted from the CPU save state, the instruction can be loaded into an array and decoded. This example assumes only required override bytes are used and there are no duplicate overrides. The order of override prefixes is assumed to be standard. Production code may want to take non-standard code into account. The maximum instruction length of a poorly formed instruction with redundant override prefixes is 15 bytes. Note that the pseudo-code does not show the detail of enabling Protected Mode for physical instruction addresses outside the Real Mode address space. An example of how to change to Protected Mode while in SMM can be found in Appendix C.

GET INSTRUCTION STREAM:

```
INSTR[0:4] = [PHYSICALADDR : PHYSICALADDR + 4]   (*use Protected Mode if
                                                  Address > 1M*)
```

DECODE INSTRUCTION STREAM:

```
REPEAT = NO;
OPOVERRIDE = NO;
ADDROVERRIDE = NO;
POINTER = 0;

IF INSTR[POINTER] = F3H THEN          (*check for repeat prefix*)
     REPEAT = YES;
     POINTER = POINTER + 1;
ENDIF;
IF INSTR[POINTER] = 67H THEN          (*check for address size override
     ADDROVERRIDE = YES               prefix*)
     POINTER = POINTER + 1;
ENDIF;
IF INSTR[POINTER] = 66H THEN          (*check for operand size override
     OPOVERRIDE = YES;                prefix*)
     POINTER = POINTER + 1;
ENDIF;

IF INSTR[POINTER] IN{ 2EH, 36H, 3EH, 26H, 64H, 65H} THEN  (*check for OUTS
     OUTSEG = INSTR[POINTER];                              segment overrides*)
     POINTER = POINTER + 1;
ELSE
     OUTSEG = 3EH;                    (*DS default segment*)
ENDIF;

OPCODE = INSTR[POINTER];              (*get the opcode*)
CASE (OPCODE) OF
     E4, EC :  OPTYPE = IN;  OPSIZE = BYTE;
     E5, ED :  OPTYPE = IN;  OPSIZE = WIDE;
     E6, EE :  OPTYPE = OUT; OPSIZE = BYTE;
     E7, EF :  OPTYPE = OUT; OPSIZE = WIDE;
     6C :      OPTYPE = INS; OPSIZE = BYTE
     6D :                    OPTYPE = INS; OPSIZE = WIDE;
     6E :                    OPTYPE = OUTS; OPSIZE = BYTE;
     6F :                    OPTYPE = OUTS; OPSIZE = WIDE;
ENDCASE;
```

With the instruction opcode decoded, the operand and address sizes embedded in the instruction can now also be decoded. The instruction handlers will use the sizes to choose which registers to modify for proper re-execution (for example DI or EDI).

DETERMINE OPERAND SIZE:

```
IF OPSIZE = WIDE THEN                 (* determine word or dword *)
   IF OPOVERRIDE = YES THEN
     IF DEFOPERAND = 16 bits THEN
               OPSIZE = DWORD;
     ELSE
               OPSIZE = WORD;
     ENDIF;
   ELSE                               (* no operand override *)
     IF DEFOPERAND = 16 bits
               OPSIZE = WORD;
     ELSE
               OPSIZE = DWORD;
     ENDIF;
   ENDIF;
ENDIF;
```

DETERMINE ADDRESS SIZE:

```
IF OPTYPE = INS OR OPTYPE = OUTS THEN
   IF ADDROVERRIDE = YES THEN
      IF DEFADDR = 16 bits THEN
                ADDRSIZE = 32 bits;
      ELSE
                ADDRSIZE = 16 bits;
      ENDIF;
   ELSE                          (* no override *)
      IF DEFADDR = 16 bits THEN
                ADDRSIZE = 16 bits;
      ELSE
                ADDRSIZE = 32 bits;
      ENDIF;
   ENDIF;
ENDIF;
```

To properly handle repeated I/O instructions, the state of the direction flag must be determined. The direction flag will determine how the instruction handlers set the index registers.

GET DIRECTION FLAG:

```
IF OPTYPE = INS OR OPTYPE = OUTS THEN
   IF DF = 0 (EFLG Bit 10, Loc 60004H) THEN
      DIRECTION = FORWARD;
   ELSE
      DIRECTION = BACKWARD;
   ENDIF;
ENDIF;
```

For use in restarting the REP OUTS instruction, the Overrun flag must be checked in the CPU save state. If this flag is clear, the REP OUTS instruction handler must account for the fact that two I/O cycles having taken place. If it is not, the handler calculates the index and count for only one I/O cycle.

CHECK OVERRUN:

```
IF OPTYPE = OUTS AND REPEAT = YES THEN
   IF OVERFLAG = 0 (Bit 0, Loc 6006CH) THEN
      OVERRUN = YES;
   ELSE
      OVERRUN = NO;
ENDIF;
ENDIF;
```

With this information in hand ,the SMM routine can now take any system dependent action required by the interrupt. When this action is complete, the SMM code is ready to resume the interrupted instruction stream. To accomplish this resumption, a separate course of action must be taken for each type of I/O instruction that was interrupted.

Depending on the type of I/O instruction trapped, a different instruction handler will be used to resume execution of the interrupted instruction stream. The following are pseudo-code handlers for each type of I/O instruction. Note that information created by the common front end code from the CPU save state and the instruction stream is used to determine how to properly prepare for I/O instruction resumption.

IN, OUT Handler—This instruction is the simplest to deal with because the instruction need only be re-executed. Therefore, the handler is reduced to assigning the save state EIP to the value of the last EIP stored in the CPU save state.

```
Opcodes IN:
     E4 immed8 byte
     E5 immed8 word/dword (depending on data size default or override)
     EC byte (from [DX])
     ED word/dword (from [DX] depending on data size)

Opcodes OUT:
     E6 immed8 byte
     E7 immed8 word/dword (depending on data size default or override)
     EE byte (from [DX])
     EF word/dword (from [DX] depending on data size)

Valid Overrides:
     Opcode Size (66) on E5 and ED only to indicate size.

BEGIN IN
     EIP = LAST EIP;          (*save state memory location changed*)
END IN;

BEGIN OUT
   EIP = LAST EIP;            (*save state memory location is changed*)
END OUT;
```

INS Handler—The INS handler must account for operand and address sizes as well as the state of the direction flag. The EIP is set to the last EIP and either DI or EDI is set to the appropriate value depending on the address size.

```
Dependencies:
     Address Size : 16 or 32 bit, depending on D bit and possible override
     Operand Size : 8, 16, or 32 bit, depending on D bit and possible
     override for word form
     Direction Flag : Affects String Increment or Decrement

Opcodes:
     6C byte
     6D word/dword (depending on data size)

Valid Overrides:
     OpSize (66) on 6D form to indicate word or dword
     AddrSize (67) to toggle default, select DI or EDI
     Segment Overrides : Ignore/Invalid
     REP—see REP INS case

BEGIN INS
   EIP = LAST EIP;            (*save state memory location is changed*)

CASE (OPSIZE) OF
     BYTE : OFFSET = 1;
     WORD : OFFSET = 2;
     DWORD : OFFSET = 4;
   ENDCASE;

   IF DIRECTION = BACKWARD THEN
     OFFSET = - OFFSET;
   ENDIF;

   IF ADDRSIZE = 16 BIT THEN
     REG = DI;
   ELSE
     REG = EDI;
   ENDIF;
   REG = REG - OFFSET;        (* save state memory location is changed *)
END INS;
```

OUTS Handler—The OUTS handler is similar to the INS handler and must also account for operand and address sizes as well as the state of the direction flag. The EIP is set to the Last EIP and either SI or ESI is set to the appropriate value depending on the address size. The "OUTSEG" variable holds the intended data segment for the transfer and does not need to be referenced in this example.

```
Dependencies:
     Address Size : 16 or 32 bit, depending on D bit and possible
   override
     Operand Size : 8, 16, or 32 bit, depending on D bit and
   possible override for word form
     Direction Flag : Affects String Increment or Decrement

Opcodes:
     6E byte
     6F word/dword (depending on data size)

Valid Overrides:
     OpSize (66) on 6F form to indicate word or dword
     AddrSize (67) to toggle default, select SI or ESI
     Segment Overrides : Any Segments allowed to override default DS for
source
     REP—see REP OUTS case

BEGIN OUTS
   EIP = LAST EIP;              (*save state memory location is changed*)
   CASE (OPSIZE) OF
     BYTE : OFFSET = 1;
     WORD : OFFSET = 2;
     DWORD : OFFSET = 4;
   ENDCASE;
   IF DIRECTION = BACKWARD THEN
     OFFSET = - OFFSET;
   ENDIF;

IF ADDRSIZE = 16 BIT THEN
     REG = SI;
   ELSE
      REG = ESI;
   ENDIF;
   REG = REG—OFFSET;          (*save state memory location is changed*)
END OUTS;
```

REP INS Handler—The REP INS handler is similar to the INS handler because it must account for operand and address sizes as well as the state of the direction flag. The repeat count must also be calculated and set appropriately. The EIP is set to the Last EIP and either the pair DI/CX or EDI/ECX are set to the appropriate values depending on the address size.

```
Dependencies:
     Address Size : 16 or 32 bit depending on D bit and possible
   override
     Operand Size : 8, 16, or 32 bit depending on D bit and
   possible override for word form
     Direction Flag : Affects String Increment or Decrement

Opcodes:
     F3 6C byte
     F3 6D word/dword (depending on data size)

Valid Overrides:
     OpSize (66) on 6D form to indicate word or dword
     AddrSize (67) to toggle default, select DI or EDI
     Segment Overrides : Ignore/Invalid
```

```
BEGIN REPINS
   EIP = LAST EIP;            (*save state memory location is changed*)

   CASE (OPSIZE) OF
      BYTE : OFFSET = 1;
      WORD : OFFSET = 2;
      DWORD : OFFSET = 4 ;
   ENDCASE;

   IF DIRECTION = BACKWARD THEN
      OFFSET = - OFFSET;
    ENDIF;

   IF ADDRSIZE = 16 BIT THEN
      REG = DI;
      CNTREG = CX;
   ELSE
      REG = EDI;
      CNTREG = ECX;
   ENDIF;

   REG = REG - OFFSET;        (*save state memory location is changed*)
   CNTREG = CNTREG + 1;       (*save state memory location is changed*)
END REPINS;
```

REP OUTS Handler—The REP OUTS handler is the most complex since it must account for the possibility of an overrun condition in addition to operand and address sizes and the state of the direction flag. The EIP is set to the Last EIP and either the pair DI/CX or EDI/ECX are set to the appropriate values depending on the address sizes. The "OUTSEG" variable holds the intended data segment for the transfer and does not need to be referenced in this example.

```
Dependencies:
     Address Size : 16 or 32 bit depending on D bit and possible
   override
     Operand Size : 8, 16, or 32 bit depending on D bit and
   possible override for word form
     Direction Flag : Affects String Increment or Decrement
     Overrun Flag : Affects resetting of CX/ECX

Opcodes:
     F3 6E byte
     F3 6F word/dword (depending on data size)

Valid Overrides:
     OpSize (66) on 6D form to indicate word or dword
     AddrSize (67) to toggle default, select SI or ESI
     Segment Overrides : Any Segments allowed to override default DS for
   source

BEGIN REPOUTS
   EIP = LAST EIP;            (*save state memory location is changed*)

   CASE (OPSIZE) OF
      BYTE : OFFSET = 1;
      WORD : OFFSET = 2;
      DWORD : OFFSET = 4 ;
   ENDCASE;

IF DIRECTION = BACKWARD THEN
      OFFSET = - OFFSET;
   ENDIF;
   IF ADDRSIZE = 16 BIT THEN
      REG = SI;
      CNTREG = CX;
   ELSE
      REG = ESI;
      CNTREG = ECX;
   ENDIF;
```

```
   IF OVERRUN = NO THEN
      COUNT = 1;
   ELSE
      OFFSET = 2 * OFFSET;
      COUNT = 2;
   ENDIF;

   REG = REG - OFFSET;          (*save state memory location is changed*)
   CNTREG = CNTREG + COUNT;     (*save state memory location is changed*)
END REPOUTS;
```

SMM STATUS BIT

Before exiting SMM, the SMM code must be sure to clear the System Management Mode Status bit (SMMS) in the debug status register (DR6, bit 12). Some diagnostic software checks this register and may indicate a CPU failure if the SMMS bit is not cleared before return. The definition of the lower 16 bits of the Debug Status Register is given in Table 2-7 for reference.

Table 2-7 Debug Register 6 (Low word only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	d0
BT	BS	BD	SMMS	0	0	0	0	0	0	0	0	B3	B2	B1	B0

SMMS = 1 SMM was entered
SMMS = 0 SMM status cleared

It should be noted that nothing prevents software from setting the SMMS bit. This can cause confusion in determining the source of the status change and should not be done. The protocol of allowing the CPU to set the bit and clearing the status with the SMM code should be maintained.

HANDLING MULTIPLE EVENTS

At the completion of the appropriate instruction handler, the SMM code may include a section to check for pending non-trap events and handle these events before exiting SMM. If there are no other pending SMI requests then the SMM code need only issue the resume command to exit SMM and continue proper execution of the interrupted instruction stream.

SOFTWARE SMI GENERATION

Besides hardware initiation of the system management interrupt via the $\overline{\text{SMI}}$ pin, there is also a software induced SMI mechanism. Generating a soft SMI involves setting control bit SMIE in Debug Control Register (DR7, bit 12) and executing a reserved opcode. The definition of the lower 16 bits of the Debug Control Register is given in Table 2-8 for reference.

A logic one written to the SMIE control bit enables the soft SMI opcode. The SMIE bit is located at bit 12 in DR7. The default (reset) state of SMIE is zero. The other reserved bits in DR7 remain as previously defined.

The soft SMI opcode is 0F1h. If the SMIE bit is a one, execution of opcode 0F1h generates a soft SMI. If the SMIE bit is a zero, then execution of this opcode generates a standard Interrupt 1 (Debug Exception). The code fragment in Figure 2-11 shows an example of how to generate a soft SMI.

After execution of the SMM routine, normal code execution resumes at the instruction following the 0F1h opcode. The SMIE bit should be set back to zero after the execution of the 0F1h opcode so that any errant execution of an 0F1h opcode by application software behaves the same as in a non-SMI based system.

The functional sequence of the software-based SMI is identical to the hardware-based SMI with the exception that the $\overline{SMI}$ pin is not initially driven active by an external source. Upon execution of a soft SMI opcode, the $\overline{SMI}$ pin is driven active (Low) by the processor before the save state operation begins. It should be noted for Protected Mode use that the soft SMI opcode is not a privileged instruction.

Figure 2-11 Soft SMI Generation

```
MOV   EAX,DR7          ;get current DR7 contents
OR    EAX,00001000h    ;set SMIE bit (#12)
MOV   DR7,EAX          ;load DR7 with SMIE set
DB    0F1h             ;execute soft SMI
```

Table 2-8 Debug Register 7 (Low word only)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	SMIE	0	0	GE	LE	G3	L3	G2	L2	G1	L1	G0	L0

SMIE = 1 enables soft SMI
SMIE = 0 disables soft SMI

HARDWARE ISSUES

The Am386DXLV and Am386SXLV microprocessors provide an unprecedented opportunity for product differentiation due to the general nature of the SMM solution. System logic hardware support for SMM comprehends both basic SMM interface support as well as I/O trapping via the I/O Instruction Break Enable feature. The following design guidelines and examples are provided as a basis for planning new SMM support logic.

SMM Signal Design Guidelines

$\overline{SMI}$ — This signal is a synchronous input, which does not have the multiple stages of synchronization of the NMI and INTR interrupt inputs. $\overline{SMI}$ is also bidirectional and is pulled up by a weak internal pull-up when not driven by the CPU. This pull-up disabled when the CPU uses $\overline{SMI}$ as an output to conserve power. After the processor drives the $\overline{SMI}$ signal High (inactive), external logic should allow $\overline{SMI}$ to be High two CLK2 periods before driving it Low again. The CPU drives $\overline{SMI}$ High during these CLK2 periods, thereby eliminating the larger current demand usually associated with open collector protocols.

$\overline{SMIADS}$ — This pin is a standard CPU output. No special treatment is required. Note that the signal floats in response to a HOLD state.

$\overline{SMIRDY}$ — This pin is a standard synchronous CPU input with an internal pull-up. Despite its similar function to $\overline{READY}$, the two signals cannot be sourced from the same signal. System logic should terminate $\overline{SMIADS}$ initiated accesses by asserting $\overline{SMIRDY}$. $\overline{READY}$ is ignored during $\overline{SMIADS}$ initiated cycles. Terminating $\overline{ADS}$ initiated cycles with $\overline{SMIRDY}$ results in unpredictable CPU behavior.

$\overline{IIBEN}$ — This signal is an asynchronous CPU input with active pull-up. $\overline{IIBEN}$ is synchronized by CLK2 internal to the CPU. The $\overline{IIBEN}$ pull-up is active during RESET pulses and whenever the signal is not driven active by the system. To conserve power, the pull-up is disabled by the CPU whenever the CPU detects that the system logic is driving $\overline{IIBEN}$ Low.

CHIPSET INDEPENDENT DESIGN EXAMPLE

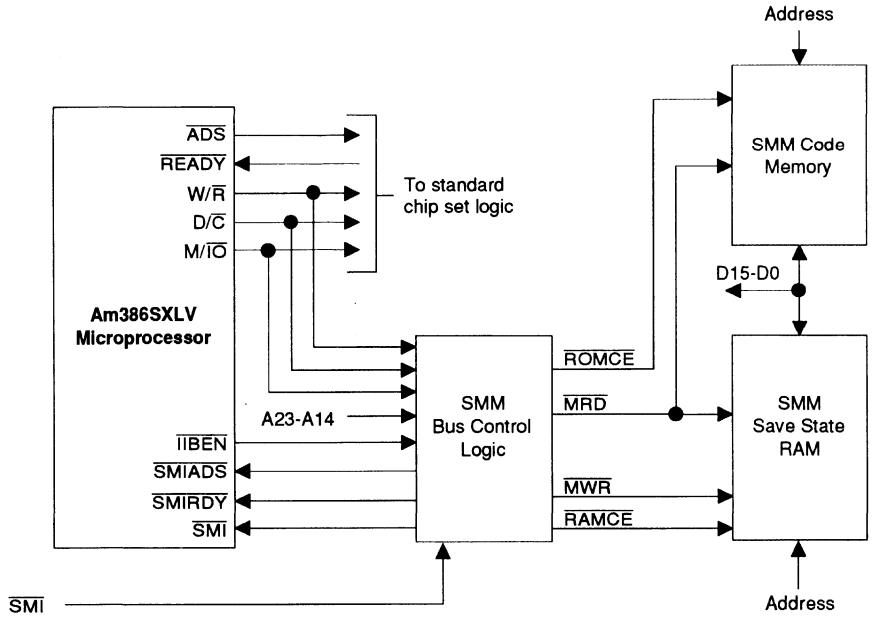
Figure 2-12 shows an Am386SXLV process-based system implementation that is independent of the normal system control logic (a chip-set independent design).

This design can be implemented on a CPU daughter-card or on a motherboard with existing chip sets. The design example consists of an Am386SXLV CPU, SMM Bus Control Logic, an SMM Code ROM, and an SMM State Save RAM.

The SMM Bus Control Logic contains a bus control state machine that starts cycles upon receiving an $\overline{SMIADS}$, generates command pulses with the appropriate number of wait states for the SMM RAM or ROM, and then generates an $\overline{SMIRDY}$ pulse. The logic also contains the address decoder for the ROM and RAM chip enables. The ROM chip enable should be active for a predefined code space, along with an alias in the last 16 bytes of the ROM for the initial FFFFF0h code fetches. The RAM chip enable should be active for at least the physical address range of 60000h–601FFh for the CPU save state table.

The SMM Code ROM is a simple 16-bit ROM implementation, with all bytes always driven on the D15–D0 pins on all ROM reads. The SMI State Save RAM is a simple 16-bit RAM implementation, with all bytes always written or read during bus transfer cycles to or from the RAM. Simple 16-bit capability is all that is required for the CPU state save and restore. If the RAM is also to be used as a "scratch pad" memory, then individual byte write cycles can be supported by factoring in the byte enable lines ($\overline{BHE}$ and $\overline{BLE}$) into separate write enables per byte. An Am386DXLV processor-based design would have a 32-bit ROM and RAM subsystem. All bus control logic would be the same.

Figure 2-12 SMM Function Discrete Implementation



16944A-13

The bus control logic should also include a register for the CPU to identify the SMI source. The SMM routine can use this information to determine its course of action dependent on whether the source was an I/O trap event, soft SMI, or general SMI.

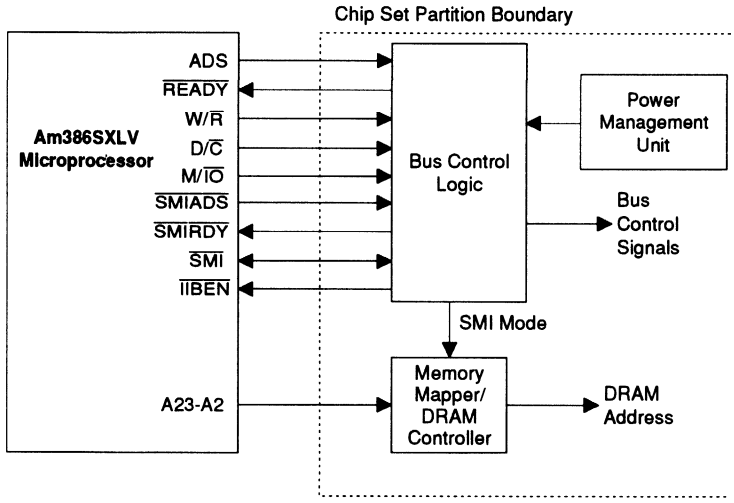
CHIP SET INTEGRATION

Figure 2-13 shows a suggested Am386SXLV processor-based system implementation that could be included in the design of the normal system control logic (SMM integration within the chip set). The two main areas in the system control logic that would be affected are the bus control logic and the memory mapper of the DRAM controller.

The Bus Control Logic would have additional connections for the SMIADS, SMIRDY, SMI, and IIBEN pins. It should generate the standard control signals for SMM bus cycles while indicating to the DRAM controller that SMM address mapping should be done for the cycle. The bus controller could also convert the 16-bit only cycles to 8-bit conversion cycles in order to support a smaller bus width for the SMI Code ROM and Save State RAM if they are separate from the system DRAM.

The DRAM controller can be designed to map SMM addresses into a protected area of system DRAM. A good example area would be unused Shadow RAM space. Both the SMM code and Save State addresses could be mapped into a single 16-KB to 128-KB segment of system DRAM. With the exception of the save state and SMM code memory area, the rest of the SMM address space can be mapped one for one onto the normal address space. This allows the use of standard memory reference instructions to access data areas and BIOS routines.

Figure 2-13 SMM System Logic Functions



16944A-14

The system power management unit should route its system management related interrupts to the Bus Control Logic. This solution allows the PMU to be asynchronous to the system clock. The bus control logic is then responsible for generating the synchronous $\overline{SMI}$ signal.

The Bus Control Logic should also include a register for the CPU to identify the SMI source. The SMM routine uses this information to determine its course of action dependent on whether the source was an I/O trap event, soft SMI, or general SMI.

SOFT SMI ALTERNATIVES

In cases where it is desirable to implement software initiated SMI routines, the system designer can use the soft SMI request opcode just as the software requested INT N command functions for normal interrupts. However, it may be desirable to implement a register in the system logic which asserts the $\overline{SMI}$ signal in response to an I/O register write.

The system core logic designer should include a register that indicates how the interrupt was generated (whether the source was an I/O trap or not) and what system event needs service if the event was not an I/O trap.

CHAPTER 3

LOW-VOLTAGE OPERATION

LOW-VOLTAGE OPERATION OVERVIEW

The low-voltage operation of the Am386DXLV and Am386SXLV microprocessors is an enabling technology for the design of portable systems with long battery life. This capability, combined with CPU clock management and SMM features, allows the design of very low power computing systems.

Low-Voltage Standard

Industry standards for low-voltage operation are emerging to facilitate the design of components which will make up a complete low-voltage system. As a guideline, the Am386DXLV and Am386SXLV processor specifications follow the first article or regulated version of the JEDEC 8.0 low-voltage proposal. This standard proposal calls for a V_{CC} range of 3.3 V ± 10%. To ease the design of a mixed voltage system, the standard also supports CMOS and TTL outputs.

Power Savings

CMOS Dynamic power consumption is proportional to the square of the operating voltage multiplied by capacitance and operating frequency. Static CPU operation can reduce power consumption by enabling the system designer to reduce operating frequency when possible. However, operating voltage is always the dominant factor in power consumption. By reducing the operating voltage from 5 V to 3.3 V for any device, the power consumed is reduced by 56% (see Figure 3-1).

Figure 3-1 3-V and 5-V System Dynamic Power Consumption

3.3 V	5 V
$P_3=V^2CFK$	$P_5=V^2CFK$
$P_3=K(3.3)^2CFK$	$P_5=(5)^2CFK$
$P_3=10.89CFK$	$P_5=25CFK$

$$\text{Reduction} = 1 - P_3/P_5 = 1 - 0.44 = 56\%$$

where V = Voltage
C = Capacitance
F = Frequency
K = Constant

The reduction of CPU and core logic operating voltage dramatically reduces overall system power consumption. Additional power savings can be realized as low-voltage mass storage and peripheral devices become available.

Two basic strategies exist in designing systems containing the Am386DXLV and Am386SXLV microprocessors. The first strategy is to design a homogenous system in which all logic components operate at 3.3 V. This provides the best overall power consumption. However, system designers may need to include devices for which 3.3-V versions are not available. In the second strategy, the system designer must then design a mixed 5-V/3.3-V system. This compromise allows the system designer to minimize the core logic power consumption while still including the functionality of the 5-V features. The choice of a mixed voltage system design also involves balancing design complexity with the need for the additional features.

PIN LEVEL INTERFACE

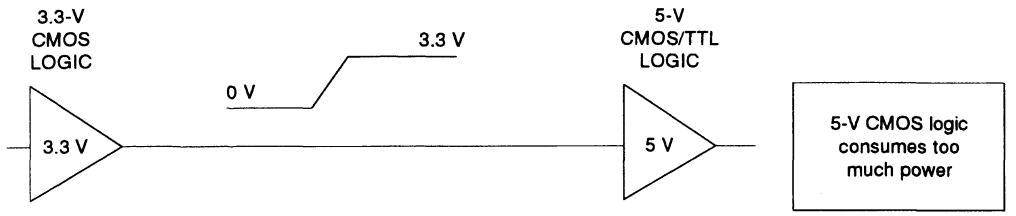
Ideally, Am386DXLV and Am386SXLV microprocessors are used in homogenous low-voltage systems. However, in some cases not all system logic devices are available in low-voltage versions. In this case, mixed voltage systems must be examined. There are two cases to consider when designing mixed voltage systems.

First consider a 3.3-V device driving a 5-V input (see Figure 3-2a). In this case, the 3.3-V signal is subject to lower noise immunity than a 5-V signal. If the buffer has a pure CMOS input, the 3.3-V signal does not drive the input buffer completely out of the transition region, thereby allowing excessive current to be consumed. The second case, a 5-V device driving a 3.3-V input (see Figure 3-2b) poses a more serious problem. In this case, the 5-V signal will over drive and possibly breakdown the 3.3-V input. This breakdown can lead to potentially damaging latch up of the 3.3-V device.

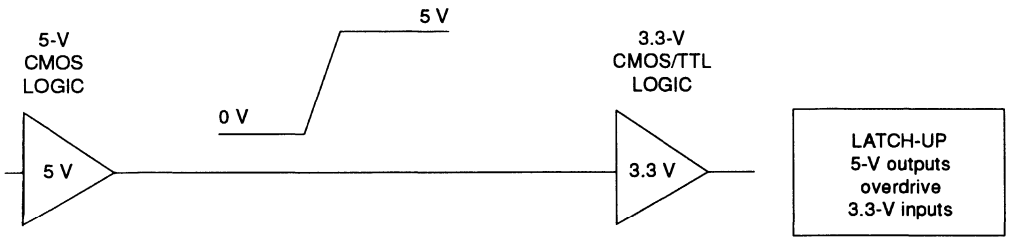
These interface problems can be avoided in two ways. The first consideration is the use of voltage translation buffers (see Figure 3-3). These dual voltage devices provide a seamless interface between different voltage devices.

A second solution addresses the 5-V driving a 3.3-V input case and uses discrete components to provide the 5-V to 3.3-V interface (see Figure 3-4). Here the use of a diode and pull-up simply and effectively translates a 5-V input signal to a 3.3-V input signal.

Figure 3-2 Mixed 5-V/3-V Considerations



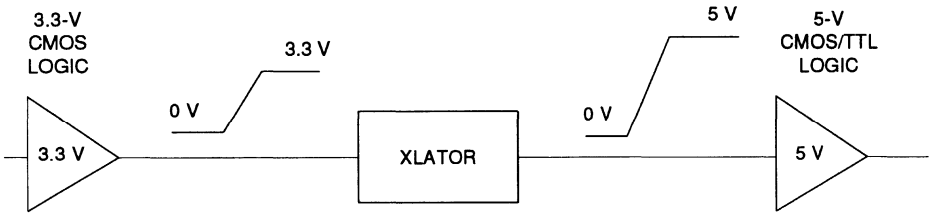
a. 3.3-V Driver to 5-V Input



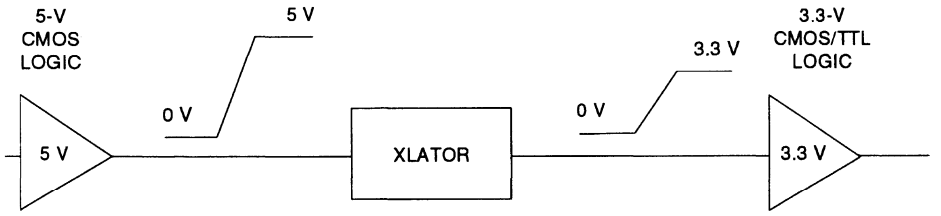
b. 5-V Driver to 3.3-V Input

16944A-16

Figure 3-3 Mixed System with Voltage Translators



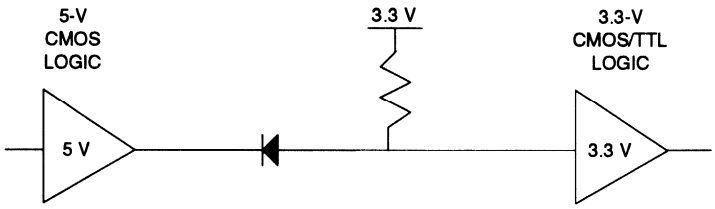
a. 3.3-V to 5-V Translation



b. 5-V to 3.3-V Translation

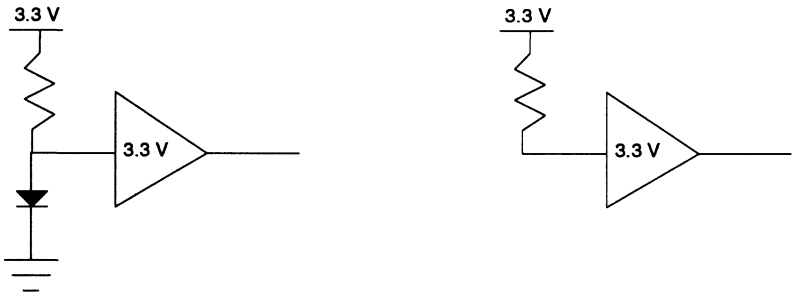
16944A-17

Figure 3-4 Mixed System with Discrete Translators



16944A-18

Figure 3-5 Discrete Translator Function



a. 5-V Output at Logic 0

b. 5-V Output at Logic 1

16944A-019

The operation of this discrete translator is diagramed in Figure 3-5. When the 5-V signal is Low, the diode is forward biased and the pull-up current is shunted through the 5-V driver (see Figure 3-5a). Thus, a low voltage is presented to the 3.3-V input. When the 5-V signal is High, the diode is reverse biased with respect to the 3.3-V pull up and is thus blocked from the 3.3-V input (see Figure 3-5b); therefore, the 3.3-V pull-up is presented to the input as a valid High. The choice of the pull-up value must balance the AC timing requirements of the signal with the desire for low current consumption.

Optionally, the use of discrete components can be eliminated by understanding the type of devices needed to interface to 3.3-V and 5-V logic and including the translation into the system core logic. Signals requiring a 5-V interface can be driven by circuitry with 5-V supplies. Any 3.3-V devices can be driven by circuitry with 3.3-V supplies. In this solution the core logic isolates different parts of the interface logic to drive the appropriate levels.

APPENDIX A

SMM CPU STATE AND REGISTER STORE/LOAD MAP

Table A–1 SMI Memory Register Store/Load Map In Address Order

SMM Address	Registers	Comments
60000	CR0	
60004	EFLAGS	
60008	EIP	
6000C	EDI	
60010	ESI	
60014	EBP	
60018	ESP	
6001C	EBX	
60020	EDX	
60024	ECX	
60028	EAX	
6002C	DR6	
60030	DR7	
60034	TR	16-bit Register
60038	LDT	16-bit Register
6003C	GS	16-bit Register
60040	FS	16-bit Register
60044	DS	16-bit Register
60048	SS	16-bit Register
6004C	CS	16-bit Register
60050	ES	16-bit Register
60054	SA:TSS	ATTR. TASK DESCR
60058	SB:TSS	BASE ADDR TSS
6005C	SL:TSS	ADDR LIMIT TSS
60060		Reserved
60064	SB:IDT	BASE ADDR IDT
60068	SL:IDT	ADDR LIMIT IDT
6006C	IOFLAG	REP OUTS Overrun Flag
60070	SB:GDT	BASE ADDR
60074	SL:GDT	ADDR LIMIT

Table A-1 SMI Memory Register Store/Load Map In Address Order (continued)

SMM Address	Registers	Comments
60078	SA:LDT	ATTR LDT DESCR.
6007C	SB:LDT	BASE ADDR LDT
60080	SL:LDT	ADDR LIMIT LDT
60084	SA:GS	ATTR GS DESCR
60088	SB:GS	BASE ADDR GS
6008C	SL:GS	ADDR LIMIT GS
60090	SA:FS	ATTR FS DESCR
60094	SB:FS	BASE ADDR FS
60098	SL:FS	ADDR LIMIT FS
6009C	SA:DS	ATTR DS DESCR
600A0	SB:DS	BASE ADDR DS
600A4	SL:DS	ADDR LIMIT DS
600A8	SA:SS	ATTR SS DESCR
600AC	SB:SS	BASE ADDR SS
600B0	SL:SS	ADDR LIMIT SS
600B4	SA:CS	ATTR CS DESCR
600B8	SB:CS	BASE ADDR
600BC	SL:CS	ADDR LIMIT
600C0	SA:ES	ATTR ES DESCR
600C4	SB:ES	BASE ADDR
600C8	SL:ES	ADDR LIMIT
60100		Temporary Register
60104		Temporary Register
60108		Temporary Register
6010C		Temporary Register
60110		Temporary Register
60114		Temporary Register
60118		Temporary Register
6011C		Temporary Register
60120		Temporary Register
60124	LEIP	Last EIP

Notes:
1. For the overrun flag at 6006CH, all bits 15–1 are reserved by AMD for future use.
2. All bits of 60060 are reserved by AMD.
3. Unused bits in 16-bit writes reserved by AMD.
4. Definitions: SA–Segment Attribute
   SB–Segment Base
   SL–Segment Limits

Table A-2 SMI Memory Register Store/Load Map In Bus Cycle Order

SMM Address	Registers	Comments
60000	CR0	
60100		Temporary Register
60104		Temporary Register
60108		Temporary Register
6010C		Temporary Register
60110		Temporary Register
60114		Temporary Register
60118		Temporary Register
6011C		Temporary Register
60120		Temporary Register
60124	LEIP	Last EIP
60004	EFLAGS	
60008	EIP	
6000C	EDI	
60010	ESI	
60014	EBP	
60018	ESP	
6001C	EBX	
60020	EDX	
60024	ECX	
60028	EAX	
6002C	DR6	
60030	DR7	
60034	TR	16-bit Register
60038	LDT	16-bit Register
6003C	GS	16-bit Register
60040	FS	16-bit Register
60044	DS	16-bit Register
60048	SS	16-bit Register
6004C	CS	16-bit Register
60050	ES	16-bit Register
60054	SA:TSS	ATTR TASK DESCR
60058	SB:TSS	BASE ADDR TSS
6005C	SL:TSS	ADDR LIMIT TSS
60060		Reserved
60064	SB:IDT	BASE ADDR IDT
60068	SL:IDT	ADDR LIMIT IDT
6006C	IOFLAG	REP OUTS Overrun Flag
60070	SB:GDT	BASE ADDR
60074	SL:GDT	ADDR LIMIT
60078	SA:LDT	ATTR LDT DESCR
6007C	SB:LDT	BASE ADDR LDT
60080	SL:LDT	ADDR LIMIT LDT

Table A-2 SMI Memory Register Store/Load Map In Bus Cycle Order (continued)

SMM Address	Registers	Comments
60084	SA:GS	ATTR GS DESCR
60088	SB:GS	BASE ADDR GS
6008C	SL:GS	ADDR LIMIT GS
60090	SA:FS	ATTR FS DESCR
60094	SB:FS	BASE ADDR FS
60098	SL:FS	ADDR LIMIT FS
6009C	SA:DS	ATTR DS DESCR
600A0	SB:DS	BASE ADDR DS
600A4	SL:DS	ADDR LIMIT DS
600A8	SA:SS	ATTR SS DESCR
600AC	SB:SS	BASE ADDR SS
600B0	SL:SS	ADDR LIMIT SS
600B4	SA:CS	ATTR CS DESCR
600B8	SB:CS	BASE ADDR
600BC	SL:CS	ADDR LIMIT
600C0	SA:ES	ATTR ES DESCR
600C4	SB:ES	BASE ADDR
600C8	SL:ES	ADDR LIMIT

Notes:
1. For the overrun flag at 6006CH, all bits 15–1 are reserved by AMD for future use.
2. All bits of 60060 are reserved by AMD.
3. Unused bits in 16-bit writes reserved by AMD.
4. Definitions: SA–Segment Attribute
   SB–Segment Base
   SL–Segment Limits

Table A-3 Processor State Upon Entering SMM

Value of Saved Register

Register	Value
CR0	0
EFLAGS	00000002
EIP	0000FFF0
EDI	0
ESI	0
EBP	0
ESP	0
EBX	0
EDX	0
ECX	0
EAX	0
DR6	Previous DR6 w/bit 12 = 1 and bits 3–0 reflect debug reg 0–3 status
DR7	0
TR	0
LDT	0
GS	0
FS	0
DS	0
SS	0
CS	F000
ES	0
CPL	0

Note: Page Unit TLB – FLUSHED

Table A–3 Processor State Upon Entering SMM (continued)

Values of Internal Descriptor Cache

Register	Value
SA:TSS	00008200
SB:TSS	0
SL:TSS	FFFFFFFF
SA:IDT	none exists
SB:IDT	0
SL:IDT	0000FFFF
SA:GDT	none exists
SB:GDT	0
SL:GDT	0000FFFF
SA:GS	00008200
SB:GS	0
SL:GS	0000FFFF
SA:FS	00008200
SB:FS	0
SL:FS	0000FFFF
SA:DS	00008200
SB:DS	0
SL:DS	0000FFFF
SA:SS	00008200
SB:SS	0
SL:SS	0000FFFF
SA:CS	00008200
SB:CS	FFFF0000
SL:CS	0000FFFF
SA:ES	00008200
SB:ES	0
SL:ES	0000FFFF

Note: 1. Definitions: SA–Segment Attribute
SB–Segment Base
SL–Segment Limits

Page Unit TLB – FLUSHED

Table A–3 Processor State Upon Entering SMM (continued)

Values of Non-Saved Registers

Register	Value
DR0	unchanged
DR1	unchanged
DR2	unchanged
DR3	unchanged
CR2	unchanged
CR3	unchanged
TR6	unchanged
TR7	unchanged

APPENDIX B

UMOV ASSEMBLER MACROS

The following listing can be used to construct the various forms of the UMOV instruction.

```
;************************MACROS**********
; UMOV Code Builder Macros
;*****************************************
;
; MACRO INVOCATION/FORMATS:
;
;     umovrdw  preg, pmem, padder  ;UMOV memory read,  word/doubleword
;     umovrd8  preg, pmem, padder  ;UMOV memory read,  byte, real mode
;     umovrd8p preg, pmem, padder  ;UMOV memory read,  4.25byte,32-bit prot
;     umovwrw  pmem, preg, padder  ;UMOV memory write, word/doubleword
;     umovwr8  pmem, preg, padder  ;UMOV memory write, byte, real mode
;     umovwr8p pmem, preg, padder  ;UMOV memory write, byte, 32-bit prot
;

; PARAMETERS:
;
;     preg -   register specification, (i.e. AX,AL,EAX,SI,etc.)
;
;     pmem -   memory address spec, one of the 386 addressing modes,
;              example would be [SI] or [EBX][ECX*4]+10
;
;     padder - displacement spec: this field specifies the added length
;              of the instruction due to the addressing mode, (i.e., the
;              number of bytes to add to the instruction besides the
;              basic ModRegR/M. Displacements and the SIB byte add to
;              the length).
;              This field should be one of the following:
;

;              Keyword  Adder    Example
;              ---------------------------------------
;              nod      0        umovrdw ax,[si],nod
;              d8       1        umovrd8 bl,[di+055h],d8
;              d16      2        umovwrw [bx+0AAAAh],dx,d16
;              d32      4        umovwr8 [ebx+0100000h],cx,d32
;              sib      1        umovrdw eax,[ebx][esi*2]
;              sibd8    2        umovwrw [ebx][ecx*4]+5,eax,sibd8
;              sibd32   5        umovwr8 [ebx][ecx*4]+100000H,dh,sibd32
;

; WARNINGS:
;     For the word and doubleword versions of the macros, all address
;     and operand sizing prefixes will be properly generated.
;     However, with the byte-sized versions, care must
;     be taken to ensure that incorrect address size overrides don't
;     get generated. To solve this problem, two versions of the byte
;     size macros are supplied. UMOVRD8 and UMOVWR8 are for use in
;     16-bit code segments (those with the USE16 segment attribute).
;     UMOVRD8P and UMOVWR8P are for use in 32-bit Protected-Mode code
;     segments (those with the USE32 segment attribute).
```

```
; COMMENTS:
;
;      -  Currently, only lowercase keywords are recognized for the
;         padder field.
;
;      -  Note that segment register overrides ARE ALLOWED in these
;         macros:
;
;                  umovrd8 al,es:[di],nod
;
;***********************************************
;*
;* UMOV READ WIDE (16/32) FROM MEMORY TO REGISTER
;*

;***********************************************
umovrdw macro preg,pmem,padder
       local istart,inext,dispcnt,iprefix,isize,iopcode

;;figure out displacement field size
       IFIDN <padder>,<nod>
dispcnt = 0
       ELSE
       IFIDN <padder>,<d8>
dispcnt = 1
       ELSE
       IFIDN <padder>,<d16>
dispcnt = 2
       ELSE
       IFIDN <padder>,<d32>
dispcnt = 4
       ELSE
       IFIDN <padder>,<sib>
dispcnt = 1
       ELSE
       IFIDN <padder>,<sibd8>
dispcnt = 2
       ELSE
       IFIDN <padder>,<sibd32>
dispcnt = 5
       ELSE
       .ERR
       ENDIF
       ENDIF
       ENDIF
       ENDIF
       ENDIF
       ENDIF
       ENDIF

;;start code generation
istart  = $
        lar         preg,pmem
inext   = $
isize   = $-istart
iprefix = isize - dispcnt - 3
iopcode = istart + iprefix + 1
        org iopcode             ;;locate for overlay opcode
        db          13H         ;;umove opcode
        org inext               ;;go back to next
        endm
```

```
;*************************************************************
;*
;* UMOV READ BYTE (8 BIT) FROM MEMORY TO REGISTER, 16-BIT CS
;*
;*************************************************************
umovrd8 macro preg,pmem,padder
        local istart,inext,dispcnt,iprefix,isize,iopcode,sreg

;;figure out displacement field size
        IFIDN <padder>,<nod>
dispcnt = 0
        ELSE
        IFIDN <padder>,<d8>
dispcnt = 1
        ELSE
        IFIDN <padder>,<d16>
dispcnt = 2
        ELSE
        IFIDN <padder>,<d32>
dispcnt = 4
        ELSE
        IFIDN <padder>,<sib>
dispcnt = 1
        ELSE
        IFIDN <padder>,<sibd8>
dispcnt = 2
        ELSE
        IFIDN <padder>,<sibd32>
dispcnt = 5
        ELSE
        .ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF

;;figure out register substitution
        IFIDN <preg>,<al>
sreg    EQU     <AX>
        ELSE
        IFIDN <preg>,<bl>
sreg    EQU     <BX>
        ELSE
        IFIDN <preg>,<cl>
sreg    EQU     <CX>
        ELSE
        IFIDN <preg>,<dl>
sreg    EQU     <DX>
        ELSE
        IFIDN <preg>,<ah>
sreg    EQU     <SP>
        ELSE
        IFIDN <preg>,<bh>
sreg    EQU     <DI>
        ELSE
        IFIDN <preg>,<ch>
sreg    EQU     <BP>
        ELSE
        IFIDN <preg>,<dh>
sreg    EQU     <SI>
        ELSE
        .ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
```

```
        ENDIF
        ENDIF
;;start code generation
istart   = $
         lar    sreg,pmem
inext    = $
isize    = $-istart
iprefix  = isize - dispcnt - 3
iopcode  = istart + iprefix + 1
        org iopcode     ;;locate for overlay opcode
        db      12H     ;;umove opcode
        org inext       ;;go back to next
        endm

;*************************************************************
;*
;* UMOV READ BYTE (8 BIT) FROM MEMORY TO REGISTER, 32-BIT CS
;*
;*************************************************************


umovrd8p macro preg,pmem,padder
         local istart,inext,dispcnt,iprefix,isize,iopcode,sreg
;;figure out displacement field size
        IFIDN <padder>,<nod>
dispcnt = 0
        ELSE
        IFIDN <padder>,<d8>
dispcnt = 1
        ELSE
        IFIDN <padder>,<d16>
dispcnt = 2
        ELSE
        IFIDN <padder>,<d32>
dispcnt = 4
        ELSE
        IFIDN <padder>,<sib>
dispcnt = 1
        ELSE
        IFIDN <padder>,<sibd8>
dispcnt = 2
        ELSE
        IFIDN <padder>,<sibd32>
dispcnt = 5
        ELSE
        .ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
;;figure out register substitution
        IFIDN <preg>,<al>
sreg    EQU     <EAX>
        ELSE
        IFIDN <preg>,<bl>
sreg    EQU     <EBX>
        ELSE
        IFIDN <preg>,<cl>
sreg    EQU     <ECX>
        ELSE
        IFIDN <preg>,<dl>
sreg    EQU     <EDX>
        ELSE
        IFIDN <preg>,<ah>
sreg    EQU     <ESP>
        ELSE
        IFIDN <preg>,<bh>
sreg    EQU      <EDI>
```

```
        ELSE
        IFIDN <preg>,<ch>
sreg    EQU     <EBP>
        ELSE
        IFIDN <preg>,<dh>
sreg    EQU     <ESI>
        ELSE
        .ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
;;start code generation
istart = $
        lar     sreg,pmem
inext  = $
isize  = $-istart
iprefix = isize - dispcnt - 3
iopcode = istart + iprefix + 1
        org iopcode     ;;locate for overlay opcode
        db      12H     ;;umove opcode
        org inext       ;;go back to next
        endm

;************************************************
;*
;* UMOV WRITE WIDE (16/32) TO MEMORY FROM REGISTER
;*
;************************************************
umovwrw macro pmem,preg,padder
        local istart,inext,dispcnt,iprefix,isize,iopcode
;;figure out displacement field size
        IFIDN <padder>,<nod>
dispcnt = 0
        ELSE
        IFIDN <padder>,<d8>
dispcnt = 1
        ELSE
        IFIDN <padder>,<d16>
dispcnt = 2
        ELSE
        IFIDN <padder>,<d32>
dispcnt = 4
        ELSE
        IFIDN <padder>,<sib>
dispcnt = 1
        ELSE
        IFIDN <padder>,<sibd8>
dispcnt = 2
        ELSE
        IFIDN <padder>,<sibd32>
dispcnt = 5
        ELSE
        .ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
```

```
;;start code generation
istart = $
       lar      preg,pmem
inext  = $
isize  = $-istart
iprefix = isize - dispcnt - 3
iopcode = istart + iprefix + 1
       org iopcode     ;;locate for overlay opcode
       db       11H    ;;umove opcode
       org inext       ;;go back to next
       endm

;*****************************************************************
;*
;* UMOV WRITE BYTE (8 BIT) TO MEMORY FROM REGISTER, 16-BIT CS
;*
;*****************************************************************
umovwr8 macro pmem,preg,padder
       local istart,inext,dispcnt,iprefix,isize,iopcode,sreg
;;figure out displacement field size
       IFIDN <padder>,<nod>
dispcnt = 0
       ELSE
       IFIDN <padder>,<d8>
dispcnt = 1
       ELSE
       IFIDN <padder>,<d16>
dispcnt = 2
       ELSE
       IFIDN <padder>,<d32>
dispcnt = 4
       ELSE
       IFIDN <padder>,<sib>
dispcnt = 1
       ELSE
       IFIDN <padder>,<sibd8>
dispcnt = 2
       ELSE
       IFIDN <padder>,<sibd32>
dispcnt = 5
       ELSE
       .ERR
       ENDIF
       ENDIF
       ENDIF
       ENDIF
       ENDIF
       ENDIF
       ENDIF

;;figure out register substitution
       IFIDN <preg>,<al>
sreg   EQU      <AX>
       ELSE
       IFIDN <preg>,<bl>
sreg   EQU      <BX>
       ELSE
       IFIDN <preg>,<cl>
sreg   EQU      <CX>
       ELSE
       IFIDN <preg>,<dl>
sreg   EQU     <DX>
       ELSE
       IFIDN <preg>,<ah>
sreg   EQU     <SP>
       ELSE
       IFIDN <preg>,<bh>
sreg   EQU     <DI>
       ELSE
       IFIDN <preg>,<ch>
sreg   EQU     <BP>
```

```
        ELSE
        IFIDN <preg>,<dh>
sreg    EQU     <SI>
        ELSE
        .ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
;;start code generation
istart    = $
          lar    sreg,pmem
inext     = $
isize     = $-istart
iprefix   = isize - dispcnt - 3
iopcode   = istart + iprefix + 1
        org iopcode     ;;locate for overlay opcode
        db      10H     ;;umove opcode
        org inext       ;;go back to next
        endm


;*****************************************************************
;*
;* UMOV WRITE BYTE (8 BIT) TO MEMORY FROM REGISTER, 32-BIT CS
;*
;*****************************************************************
umovwr8p macro pmem,preg,padder
         local istart,inext,dispcnt,iprefix,isize,iopcode,sreg
;;figure out displacement field size
        IFIDN <padder>,<nod>
dispcnt = 0
        ELSE
        IFIDN <padder>,<d8>
dispcnt = 1
        ELSE
        IFIDN <padder>,<d16>
dispcnt = 2
        ELSE
        IFIDN <padder>,<d32>
dispcnt = 4
        ELSE
        IFIDN <padder>,<sib>
dispcnt = 1
        ELSE
        IFIDN <padder>,<sibd8>
dispcnt = 2
        ELSE
        IFIDN <padder>,<sibd32>
dispcnt = 5
        ELSE
        .ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
```

```
;;figure out register substitution
        IFIDN <preg>,<al>
sreg    EQU     <EAX>
        ELSE
        IFIDN <preg>,<bl>
sreg    EQU     <EBX>
        ELSE
        IFIDN <preg>,<cl>
sreg    EQU     <ECX>
        ELSE
        IFIDN <preg>,<dl>
sreg    EQU     <EDX>
        ELSE
        IFIDN <preg>,<ah>
sreg    EQU     <ESP>
        ELSE
        IFIDN <preg>,<bh>
sreg    EQU     <EDI>
        ELSE
        IFIDN <preg>,<ch>
sreg    EQU     <EBP>
        ELSE
        IFIDN <preg>,<dh>
sreg    EQU     <ESI>
        ELSE
        ERR
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF
        ENDIF

;;start code generation
istart   = $
         lar    sreg,pmem
inext    = $
isize    = $-istart
iprefix  = isize - dispcnt - 3
iopcode  = istart + iprefix + 1
        org iopcode     ;;locate for overlay opcode
        db      10H     ;;umove opcode
        org inext       ;;go back to next
        endm
```

APPENDIX C

SMM PROTECTED MODE SAMPLE

Listing 1 **Protected Mode Entry While In SMM**

```
page ,132
NAME protcode
        .386P

;****extern decls ****
        EXTRN  SEG_STARTUP_REALSEG:ABS
;link time "define="
        EXTRN  _SEG_SYS_GDT_LIMIT:ABS
;GDT LIMIT
        EXTRN  _SEG_SYS_GDT_PADDR:ABS
;GDT PHYSICAL ADDRESS
        EXTRN  _SEG_SYS_IDT_LIMIT:ABS
;IDT LIMIT
        EXTRN  _SEG_SYS_IDT_PADDR:ABS
;IDT PHYSICAL ADDRESS
        EXTRN  _SEG_FLATDATA_BEGIN:PWORD
;for flatdata selector
;************************ MACROS ***************************************

;************************ MAIN PROGRAM *********************************

smidata  segment use32 rw 'DATA' ; should be located at 60000h
;****** smi save state data
        org 0h
sscr0   dd      ?
sseflg  dd      ?
sseip   dd      ?
ssedi   dd      ?
ssesi   dd      ?
ssebp   dd      ?
ssesp   dd      ?
ssebx   dd      ?
ssedx   dd      ?
ssecx   dd      ?
sseax   dd      ?

ssdr6   dd      ?
ssdr7   dd      ?

sstr    dw      ?
ssdum1  dw      ?

ssldt   dw      ?
ssdum2  dw      ?

ssgs    dw      ?
ssdum3  dw      ?

ssfs    dw      ?
ssdum4  dw      ?

ssds    dw      ?
ssdum5  dw      ?

ssss    dw      ?
ssdum6  dw      ?

sscs    dw      ?
ssdum7  dw      ?
```

Listing 1 Protected Mode Entry While In SMM (continued)

```
sses    dw      ?
ssdum8  dw      ?

sstssa  dd      ?
sstssb  dd      ?
sstssl  dd      ?

ssidta  dd      ?
ssidtb  dd      ?
ssidtl  dd      ?

ssgdta dd       ?
ssgdtb dd       ?
ssgdtl dd       ?

ssldta dd       ?
ssldtb dd       ?
ssldtl dd       ?

ssgsa   dd      ?
ssgsb   dd      ?
ssgsl   dd      ?

ssfsa   dd      ?
ssfsb   dd      ?
ssfsl   dd      ?

ssdsa   dd      ?
ssdsb   dd      ?
ssdsl   dd      ?

ssssa   dd      ?
ssssb   dd      ?
ssssl   dd      ?

sscsa   dd      ?
sscsb   dd      ?
sscsl   dd      ?

ssesa   dd      ?
ssesb   dd      ?
ssesl   dd      ?

        org 100h
ssptu   dd      ?
ssr22   dd      ?
ssr26   dd      ?
ssr27   dd      ?
ssr28   dd      ?
ssr29   dd      ?
ssr210  dd      ?
ssr41   dd      ?
ssr42   dd      ?
ssr43   dd      ?

        org 200h
stdata  label   byte

        org 1FFFh
lastone db      ?  ;end of 8k data segment
smidata ends
```

Listing 1 Protected Mode Entry While In SMM (continued)

```
;******* stack segment *******
stack    segment para STACK use32 rw 'STACK' ;locate at 62000h
stkbot   db  0fffh dup (?)
stktop   db  ?
stack    ends

;******* Protected Mode code ********
pcode    segment use32 er 'CODE' ;locate at 64000h
pstart:
        assume cs:pcode,ds:smidata,ss:stack,es:nothing,
        fs:nothing,gs:nothing
        mov     ebx,0F0F0F0F0h      ;very large address
        mov     eax,05A5A5A5AH
;       umov    es:[ebx],eax        ;example write to upper memory
        db      026h,0Fh,11h,00h

;clear text screen to prove in Protected Mode
        mov  ebx,0B8000h            ;screen address for cga style screen
        mov  ecx,2048               ;guess at number bytes
        mov  ax,0700h+' '           ;guess at char value

clrtop:
;       umov    es:[ebx],ax
        db      066h,026h,0Fh,11h,00h
        add     ebx,2
        loop32 clrtop

;*** Protected Mode resume
        mov edi,0
        mov ax,ds                   ;ds has smidata segment
        mov es,ax
        assume es:smidata
prezume  db 0fh                     ;protect mode loadall from es:[edi]
         db 07h
         nop
         nop
         nop
         nop

         hlt

pcode    ends

;******* startup segment ***************
startup segment use16 er 'CODE'
FFFF0    equ 0ff0h                  ;ROM location 0fh bytes from the top
         assume cs:startup          ;will be set at load time

;******* CS BASED VARS *******

gdt_p    label    pword
         dw       _SEG_SYS_GDT_LIMIT
         dd       _SEG_SYS_GDT_PADDR
idt_p    label    pword
         dw  _    _SEG_SYS_IDT_LIMIT
         dd  _    _SEG_SYS_IDT_PADDR
null_p   label    pword
         dw       0h
         dd       0h

;******* ROM CODE START *******
         org 100h                    ;bottom of 4k segment at 67000h +
100H
start:
         in al,080h                  ;increment port 80h
         inc al
         inc al
         out 080h,al
```

Listing 1 Protected Mode Entry While In SMM (continued)

```
;******* PROTECT MODE ENTRY ******

        cli
        lidte   cs:null_p           ;shutdown if error
        lgdte   cs:gdt_p            ;load gdt pointer

        mov     eax,cr0             ;set protect bit
        or      eax,01h
        mov     cr0,eax

        jmp     short flush         ;flush prefetch queue
flush:
        mov     bx,seg stack        ;setup stack
        mov     ss,bx
        assume  ss:stack
        mov     esp,offset stktop

        mov     bx,seg smidata      ;setup ds
        mov     ds,bx
        assume  ds:smidata
        mov     bx,seg_SEG_FLATDATA_BEGIN ;setup es,fs,gs
        mov     es,bx
        mov     fs,bx
        mov     gs,bx
        lidte   cs:idt_p            ;setup real idt
        jmp     far ptr pstart      ;far intersegment jump

;******* END PROTECT MODE ENTRY *****
        ; REAL MODE RESUME ONLY!!!
        ; Setup resume from data at 60000h assumes EDI=0
        mov edi,0
        mov ax,6000h
        mov es,ax
rezume  db 0fh
        db 07h

        nop
        nop
        nop
        nop

        hlt

        ;-------------------------------------------------------
        ;         BOOTSTRAP RESET ROUTINE
        ;-------------------------------------------------------
        org FFFF0 ;reset location 0ffff0h remapped to (ROM size-10h)
reset:
;       jmp far ptr start
        DB    0EAh  ;FAR JUMP OPCODE
        DW    offset start
        DW    SEG_STARTUP_REALSEG
        nop
        nop

        org FFFF0 + 000Fh
lastfill db 0AAh

startup ends      ;end of startup code

        end start ;end of code, start desig is nop
```

Listing 2 PHAR Lap Link File

```
protcode
! object file
!-omfboot protcode.omf
!for software debugger
!-symbols
!-isymbols
!-locmap
-binary protcode.bin 060000h
!-hex protcode.hex
-386
! 386 target
-multiseg
-segsyms
-nosshide
-mapnames 20
-mapwidth 132
-build gdt,idt
-locate seg startup 067000h
-define SEG_STARTUP_REALSEG=6700h
-locate seg smidata
060000h
-locate seg stack
062000h
-locate seg pcode
064000h
-locate seg sys_gdt
063000h
-locate seg sys_idt
063100h
-segment FLATDATA limit=0FFFFFFFFh rw
-locate seg FLATDATA 00000000h
```

Chapter 2

SCSI Products

CHAPTER 2
Small Computer System Interface (SCSI) Products

PRELIMINARY

Am53C94LV

Low-Voltage, High-Performance SCSI Controller

Advanced Micro Devices

DISTINCTIVE CHARACTERISTICS

- Functionally compatible with Am53C94
- Supports Low Voltage operation at 3.3 V. Conforms to JEDEC baseline specification
- AMD patented GLITCH EATER™ circuitry
- 5 Mbit per second SCSI transfer rate
- 20 Mbit per second DMA transfer rate
- 16-bit DMA interface plus 2 bits of parity
- Flexible bus architecture, supports a three bus architecture
- Supports single ended SCSI bus
- Selection of multiplexed or non-multiplexed address and data bus
- High current drivers (48 mA) for direct connection to the single ended SCSI bus
- Supports Disconnect and Reselect commands
- Supports burst mode DMA operation with a threshold of 8
- Supports 3 byte tagged queuing as per the SCSI-2 specification
- Supports group 2 and 5 command recognition as per the SCSI-2 specification
- Advanced CMOS process for low power consumption
- Available in 100-pin PQFP package

GENERAL DESCRIPTION

The Low-Voltage, High-Performance SCSI Controller (LVHPSC) has a flexible three bus architecture. The LVHPSC has a 16-bit DMA interface, an 8-bit host data interface and an 8-bit SCSI data interface. The LVHPSC is designed to minimize host intervention by implementing common SCSI sequences in hardware. An on-chip state machine reduces protocol overheads by performing the required sequences in response to a single command from the host. Selection, Reselection, Information Transfer and Disconnection commands are directly supported.

The 16-byte internal FIFO further assists in minimizing host involvement. The FIFO provides a temporary storage for all command, data, status and message bytes as they are transferred between the 16 bit host data bus and the 8 bit SCSI data bus. During DMA operations the FIFO acts as a buffer to allow greater latency in the DMA channel. This permits the DMA channel to be suspended for higher priority operations such as DRAM refresh or reception of an ISDN packet.

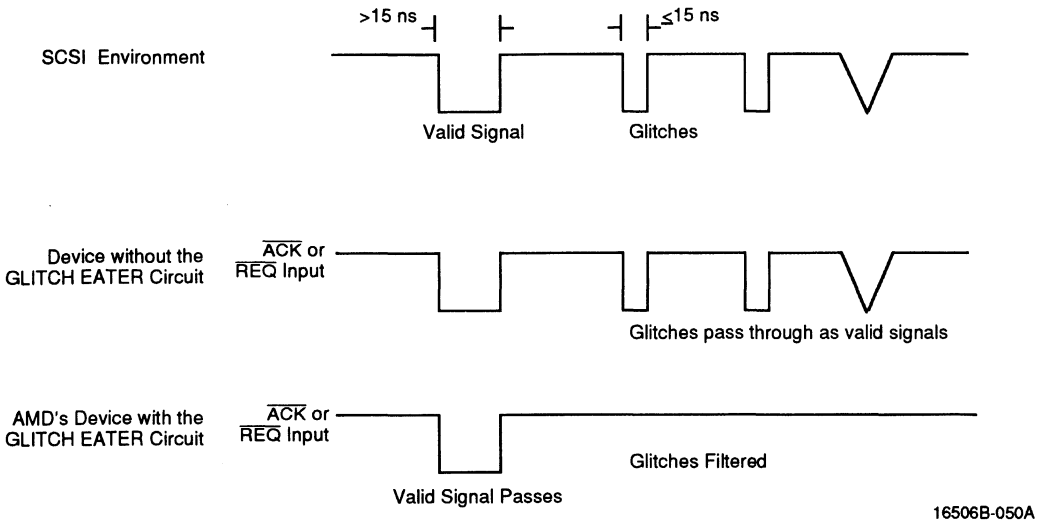
Parity on the DMA bus is optional. Parity can either be generated and checked or it can be simply passed through.

The patented GLITCH EATER Circuitry in the High Performance SCSI Controller detects signal changes that are less than or equal to 15 ns and filters them out. It is designed to dramatically increase system performance and reliability by detecting and filtering glitches that can cause system failure.

The GLITCH EATER Circuitry is implemented on the $\overline{ACK}$ and $\overline{REQ}$ lines only. These lines often encounter many electrical anomalies which degrade system performance and reliability. The two most common are Reflections and Voltage Spikes. Reflections are a result of high current SCSI signals that are mismatched by stubs, cables and terminators. These reflections vary from application to application and can trigger false handshake signals on the $\overline{ACK}$ and $\overline{REQ}$ lines if the voltage amplitude is at the TTL threshold levels. Spikes are generated by high current SCSI signals switching concurrently. On the control signals ($\overline{ACK}$ and $\overline{REQ}$) they can trigger false data transfers which result in loss of data, addition of random data, double clocking and reduced system reliability. AMD's GLITCH EATER Circuitry helps maintain excellent system performance by treating the glitches. Refer to the diagrams GLITCH EATER Circuitry and System Block Diagram.

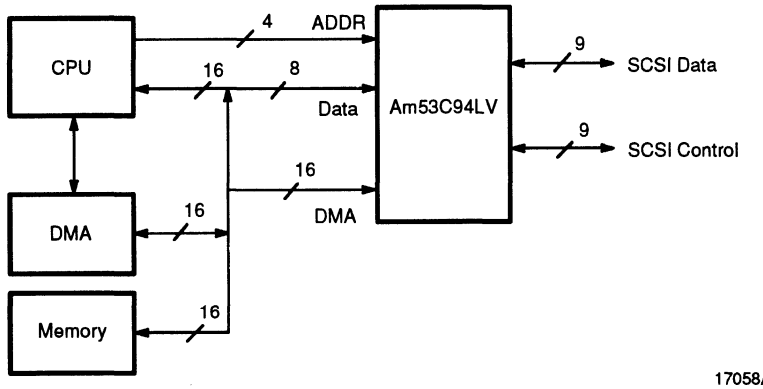
Publication# 17058 Rev. B Amendment /0
Issue Date: January 1993

GLITCH EATER Circuitry in SCSI Environment



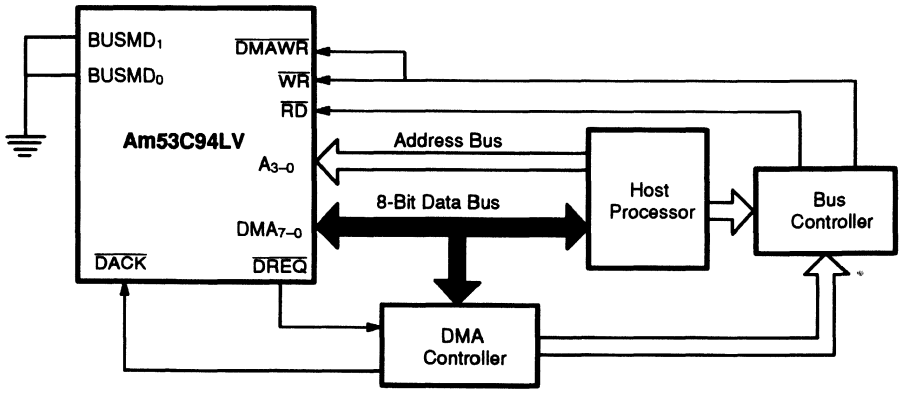
16506B-050A

SYSTEM BLOCK DIAGRAM



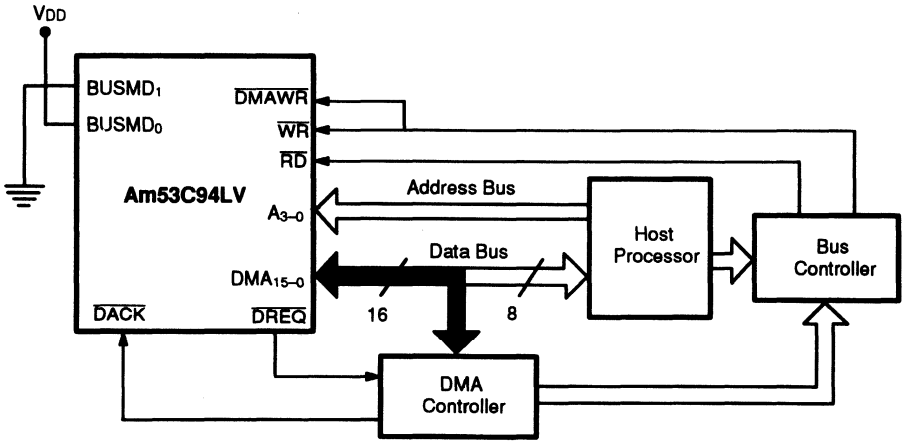
17058A-001A

SYSTEM BUS MODE DIAGRAMS



17058A-002A

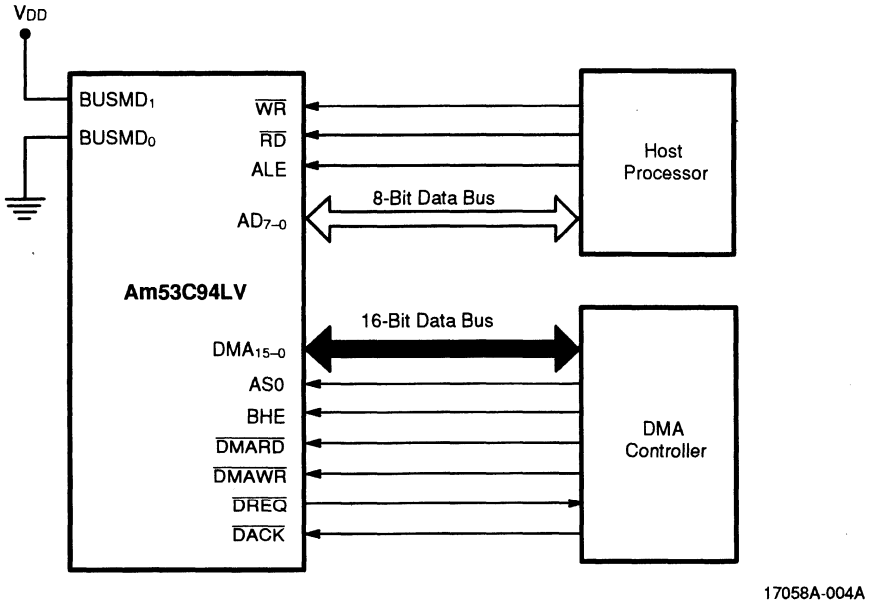
Bus Mode 0



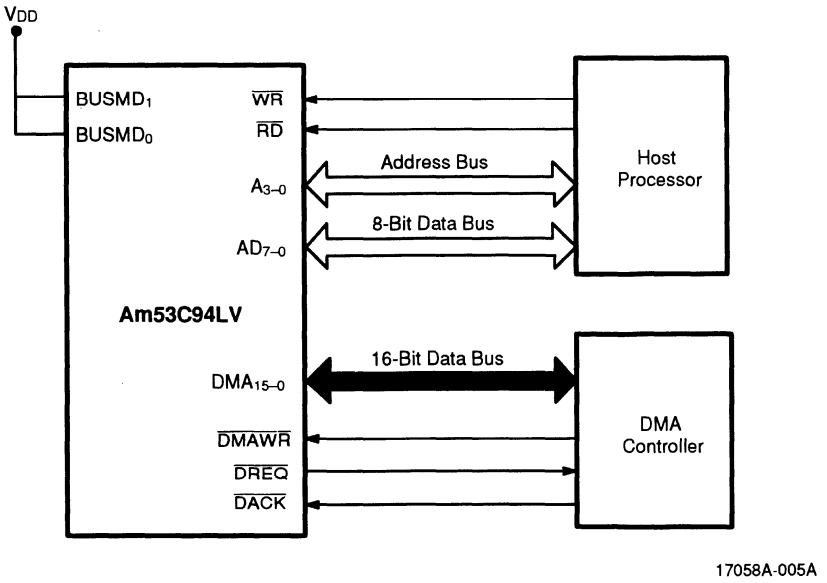
17058A-003A

Bus Mode 1

SYSTEM BUS MODE DIAGRAMS

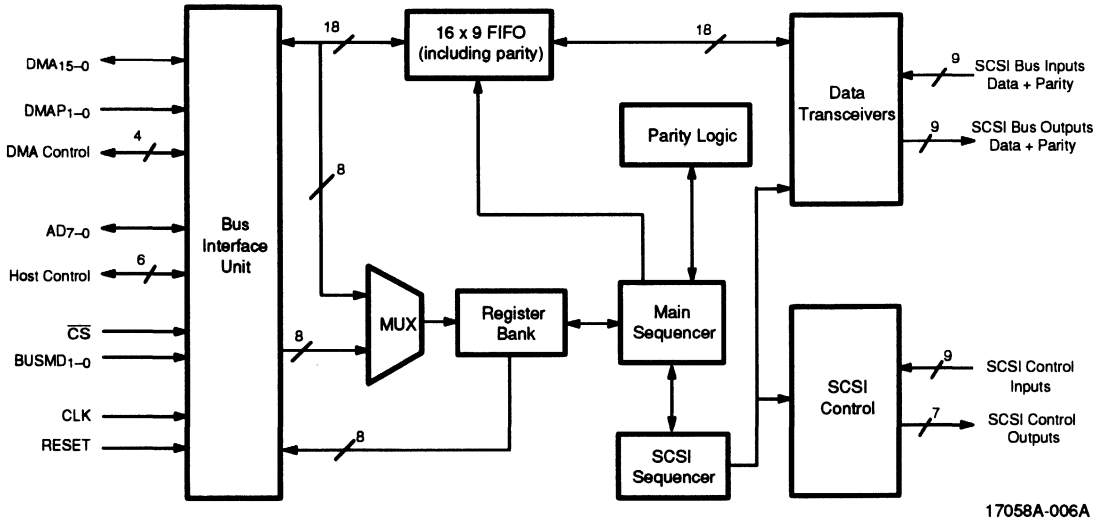


Bus Mode 2



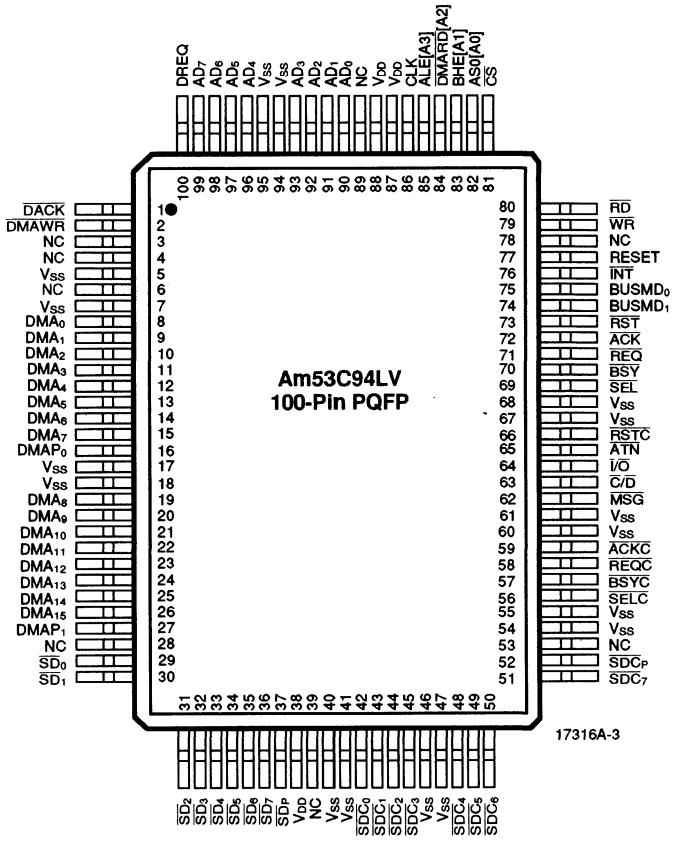
Bus Mode 3

BLOCK DIAGRAM



CONNECTION DIAGRAM
Am53C94LV (Top View)

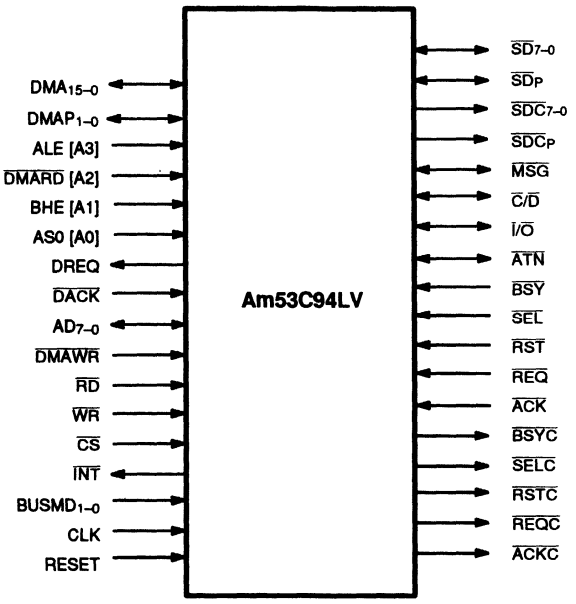
PQFP



RELATED AMD PRODUCTS

Part Number	Description
Am33C93A	5 Mbytes/sec Async/Synchronous CMOS SCSI Controller
Am53C80A	4 Mbytes/sec Asynchronous CMOS SCSI Controller
Am53C94	High-Performance CMOS SCSI Controller (Single-Ended)
Am53CF94	CMOS Fast SCSI-2 Chip (Single-Ended)
Am53CF94LV	Low Voltage Fast SCSI-2 Controller
Am53C96	High-Performance CMOS SCSI Controller (Single-Ended and Differential)
Am53CF96	CMOS Fast SCSI-2 Controller (Single-Ended and Differential)
Am85C30	Enhanced Serial Communications Controller (ESCC)
Am85C80	Combination SCSI Controller (Am53C80A) and ESCC (Am85C30)

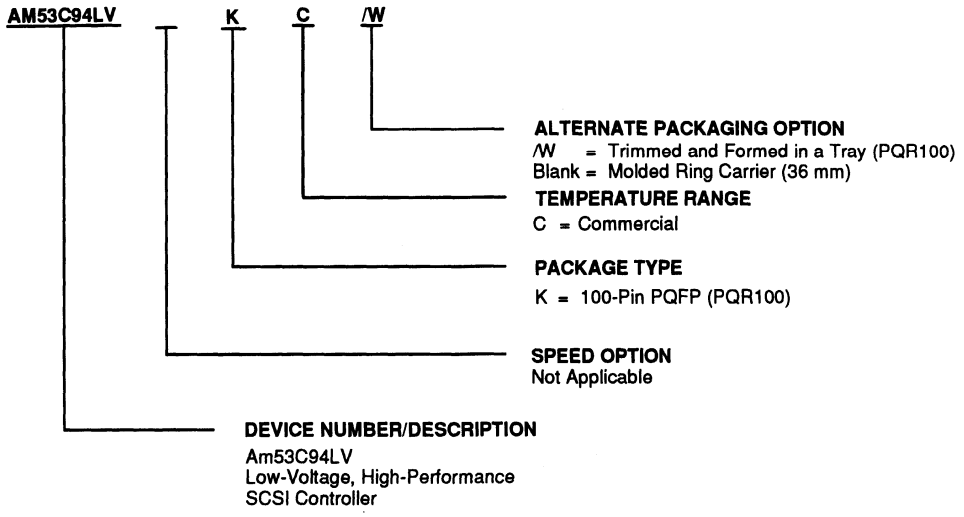
LOGIC SYMBOL



17058A-008A

ORDERING INFORMATION
Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:

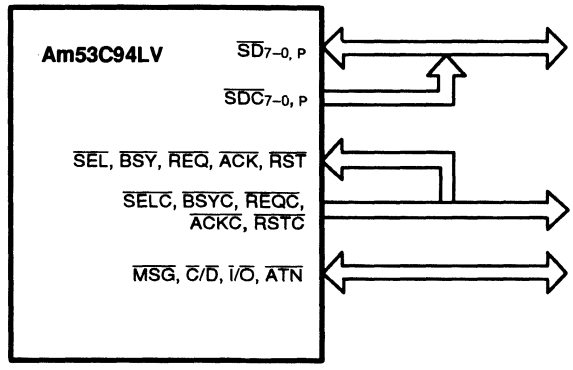


Valid Combinations	
AM53C94LV	KC, KC/W

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, to check on newly released combinations, and to obtain additional data on AMD's standard military grade products.

SCSI OUTPUT CONNECTIONS



Am53C94LV Single Ended SCSI Bus Configuration

16506B-048A

PIN DESCRIPTION

Host Interface Signals

DMA_{15-0}
Data/DMA Bus (Input/Output, Active High, Internal Pullup)

The configuration of this bus depends on the Bus Mode 1–0 ($BUSMD_{1-0}$) inputs. When the device is configured for single bus operation, the host can access the internal register set on the lower eight lines and the DMA accesses can be made to the FIFO using the entire bus. When using the Byte Mode via the BHE and A0 inputs the data can be transferred on either the upper or lower half of the DMA_{15-0} bus.

$DMAP_{1-0}$
Data/DMA Parity Bus (Input/Output, Active High, Internal Pullup)

These lines are odd parity for the DMA_{15-0} bus. DMAP 1 is the parity for the upper half of the bus (DMA_{15-8}) and DMAP 0 is the parity for the lower half of the bus (DMA_{7-0}).

ALE [A3]
Address Latch Enable [Address 3] (Input, Active High)

This is a dual function input. When the device is configured for single bus operation this input acts as ALE. As ALE, this input latches the address on the AD_{7-0} bus on its Low going edge. When the device is configured for dual bus operation this input acts as A3. As A3, this input is the third bit of the address bus.

$\overline{DMARD}$ [A2]
DMA Read [Address 2] (Input, Active Low [Active High])

This is a dual function input. When the device is configured for single bus operation this input acts as $\overline{DMARD}$. As $\overline{DMARD}$, this input is the read signal for the DMA_{15-0} bus. When the device is configured for dual bus operation this input acts as A2. As A2, this input is the second bit of the address bus.

BHE [A1]
Bus High Enable [Address 1] (Input, Active High)

This is a dual function input. When the device is configured for single bus operation this input acts as BHE. As BHE, this input along with AS0 indicates on which lines the data transfer is to take place. When the device is configured for dual bus operation this input acts as A1. As A1, this input is the first bit of the address bus.

The following is the decoding for the BHE and AS0 inputs:

BHE	AS0	Bus Used
1	1	Upper Bus – DMA_{15-8}, $DMAP_1$
1	0	Full Bus – DMA_{15-0}, $DMAP_{1-0}$
0	1	Reserved
0	0	Lower Bus – DMA_{7-0}, $DMAP_0$

AS0 [A0]
Address Status [Address 0] (Input, Active High)

This is a dual function input. When the device is configured for single bus operation this input acts as AS0. As AS0, this input along with BHE indicates on which lines the data transfer is to take place. When the device is configured for dual bus operation this input acts as A0. As A0, this input is the zeroth bit of the address bus.

DREQ
DMA Request (Output, Active High, HI-Z)

This output signal to the DMA controller will be active during DMA read and write cycles. During a DMA read cycle it will be active as long as there is a word (or a byte in the byte mode) in the FIFO to be transferred to memory. During a DMA write cycle it will be active as long as there is an empty space for a word (or a byte in the byte mode) in the FIFO.

$\overline{DACK}$
DMA Acknowledge (Input, Active Low)

This input signal from the DMA controller will be active during DMA read and write cycles. The $\overline{DACK}$ signal is used to access the DMA FIFO only and should never be active simultaneously with the $\overline{CS}$ signal, which accesses the registers only.

AD_{7-0}
Host Address Data Bus (Input/Output, Active High, Internal Pullup)

This bus is used only in the dual bus mode. This bus allows the host processor to access the device's internal registers while the DMA bus is transferring data. When using the multiplexed bus mode, these lines can be used for address and data. When using a non-multiplexed bus mode these lines can be used for the data only.

$\overline{DMAWR}$
DMA Write (Input, Active Low)

This signal writes the data on the DMA_{15-0} bus into the internal FIFO when DACK is also active. When in the single bus mode this signal must be tied to the WR signal.

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

This signal reads the internal device registers and places their contents on the data bus, when either $\overline{CS}$ signal or $\overline{DACK}$ signal is active.

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

This signal writes the internal device registers with the value present on the data bus, when the $\overline{CS}$ signal is also active.

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

This signal enables the read and write of the device registers. $\overline{CS}$ enables access to any register (including the FIFO) while the $\overline{DACK}$ enables access only to the FIFO. $\overline{CS}$ and $\overline{DACK}$ should never be active simultaneously in the single bus mode, they may however be active simultaneously in the dual bus mode provided the $\overline{CS}$ signal is not enabling access to the FIFO.

$\overline{INT}$
Interrupt (Output, Active Low, Open Drain)

This signal is a non-maskable interrupt flag to the host processor. This signal is latched on the output on the high going edge of the clock. This flag may be cleared by reading the Interrupt Status Register (ISTAT) or by performing a device reset (hard or soft). This flag is not cleared by a SCSI reset.

$BUSMD_{1-0}$
Bus Mode (Input, Active High)

These inputs configure the device for single bus or dual bus operation and the DMA width.

BUSMD1	BUSMD0	Bus Configuration
1	1	Two buses: 8-bit Host Bus and 16-bit DMA Bus Register Address on A_{3-0} and Data on AD Bus
1	0	Two buses: Multiplexed and byte control Register Address on AD_{3-0} and Data on AD Bus
0	1	Single bus: 8-bit Host Bus and 16-bit DMA Bus Register Address on A_{3-0} and Data on DMA Bus
0	0	Single bus: 8-bit Host Bus and 8-bit DMA Bus Register Address on A_{3-0} and Data on DMA Bus

CLK
Clock (Input)

Clock input used to generate all the internal device timings. The maximum frequency of this input is 25 MHz. A minimum of 10 MHz is required to maintain the SCSI bus timings.

RESET
Reset (Input, Active High)

This input when active resets the device. The RESET input must be active for at least two CLK periods after the voltage on the power inputs has reached Vcc minimum.

SCSI Interface Signals

$\overline{SD}_{7-0}$
SCSI Data (Input, Active Low, Schmitt Trigger)

These are SCSI data input pins.

$\overline{SD}_P$
SCSI Data Parity (Input, Active Low, Schmitt Trigger)

This is the SCSI data parity input pin.

$\overline{SDC}_{7-0}$
SCSI Data Control (Output, Active Low, Open Drain)

These are SCSI data output pins.

$\overline{SDC}_P$
SCSI Data Control Parity (Output, Active Low, Open Drain)

This is the SCSI data parity output pin.

$\overline{MSG}$
Message (Input/Output, Active Low, Schmitt Trigger)

This is a bidirectional signal with 48 mA output driver. It is an output in the target mode and a Schmitt trigger input in the initiator mode.

$\overline{C}/\overline{D}$
Command/Data (Input/Output, Schmitt Trigger)

This is a bidirectional signal with 48 mA output driver. It is an output in the target mode and a Schmitt trigger input in the initiator mode.

$\overline{I}/\overline{O}$
Input/Output (Input/Output, Schmitt Trigger)

This is a bidirectional signal with 48 mA output driver. It is an output in the target mode and a Schmitt trigger input in the initiator mode.

$\overline{\text{ATN}}$
Attention (Input/Output, Active Low, Schmitt Trigger)

This signal is a 48 mA output in the initiator mode and a Schmitt trigger input in the target mode. This signal will be asserted when the initiator detects a parity error or it can be asserted via certain initiator commands.

$\overline{\text{BSY}}$
Busy (Input, Active Low, Schmitt Trigger)

This is a SCSI input signal with a Schmitt trigger.

$\overline{\text{SEL}}$
Select (Input, Active Low, Schmitt Trigger)

This is a SCSI input signal with a Schmitt trigger.

$\overline{\text{RST}}$
Reset (Input, Active Low, Schmitt Trigger)

This is a SCSI input signal with a Schmitt trigger.

$\overline{\text{REQ}}$
Request (Input, Active Low, Schmitt Trigger)

This is a SCSI input signal with a Schmitt trigger.

$\overline{\text{ACK}}$
Acknowledge (Input, Active Low, Schmitt Trigger). This is a SCSI input signal with a Schmitt trigger.

$\overline{\text{BSYC}}$
Busy Control (Output, Active Low, Open Drain)

This is a SCSI output with 48 mA drive. This pin is the $\overline{\text{BSY}}$ output for the SCSI bus.

$\overline{\text{SELC}}$
Select Control (Output, Active Low, Open Drain)

This is a SCSI output with 48 mA drive. This pin is the $\overline{\text{SEL}}$ output for the SCSI bus.

$\overline{\text{RSTC}}$
Reset Control (Output, Active Low, Open Drain)

This is a SCSI output with 48 mA drive. The Reset SCSI command will cause the device to drive $\overline{\text{RSTC}}$ active for 25–40 microseconds, which will depend on the CLK frequency and the conversion factor. This pin is the $\overline{\text{RST}}$ output for the SCSI bus.

$\overline{\text{REQC}}$
Request Control (Output, Active Low, Open Drain)

This is a SCSI output with 48 mA drive. This signal is asserted only in the target mode.

$\overline{\text{ACKC}}$
Acknowledge Control (Output, Active Low, Open Drain)

This is a SCSI output with 48 mA drive. This signal is asserted only in the initiator mode.

FUNCTIONAL DESCRIPTION

Register Map

Address (Hex)	Operation	Register
00	Read	Current Transfer Count Register LSB
00	Write	Start Transfer Count Register LSB
01	Read	Current Transfer Count Register MSB
01	Write	Start Transfer Count Register MSB
02	Read/Write	FIFO Register
03	Read/Write	Command Register
04	Read	Status Register
04	Write	SCSI Destination ID Register
05	Read	Interrupt Status Register
05	Write	SCSI Timeout Register
06	Read	Internal State Register
06	Write	Synchronous Transfer Period Register
07	Read	Current FIFO Internal State Register
07	Write	Synchronous Offset Register
08	Read/Write	Control Register 1
09	Write	Clock Factor Register
0A	Write	Forced Test Mode Register
0B	Read/Write	Control Register 2
0C	Read/Write	Control Register 3
0F	Write	Data Alignment Register

Note:

Not all registers in this device are both readable and writable. Some read only registers share the same address with write only registers. The registers can be accessed by asserting the $\overline{CS}$ signal and then asserting either $\overline{RD}$ or $\overline{WR}$ signal depending on the operation to be performed. Only the FIFO Register can be accessed by asserting either $\overline{CS}$ or $\overline{DACK}$ in conjunction with $\overline{RD}$ and $\overline{WR}$ signals or $\overline{DMARD}$ and $\overline{DMAWR}$ signals. The register address inputs are ignored when $\overline{DACK}$ is used but must be valid when $\overline{CS}$ is used.

Current Transfer Count Register (00H–01H) Read Only

Current Transfer Count Register
CTCREG

Address: 00H–01H
Type: Read

15	14	13	12	11	10	9	8
CRVL15	CRVL14	CRVL13	CRVL12	CRVL11	CRVL10	CRVL9	CRVL8
X	X	X	X	X	X	X	X

7	6	5	4	3	2	1	0
CRVL7	CRVL6	CRVL5	CRVL4	CRVL3	CRVL2	CRVL1	CRVL0
X	X	X	X	X	X	X	X

16506B-013A

CTCREG – Bits 15:0 – CRVL 15:0 – Current Value 15:0

This is a two byte register. It counts down to keep track of the number of DMA transfers. Reading this registers will return the current value of the counter. The counter will decrement by one for every byte transferred and two for every word transferred over the SCSI bus. The transaction is complete when the count reaches zero. These registers are automatically loaded with the values in the Start Transfer Count Register every time a DMA command is issued.

In the target mode, this counter is decremented by the active edge of $\overline{DACK}$ during the Data-In phase and by $\overline{REQC}$ during the Data-Out phase.

In the initiator mode, the counter is decremented by the active edge of $\overline{DACK}$ during the Synchronous Data-In phase or by $\overline{ACKC}$ during the Asynchronous Data-In phase and by $\overline{DACK}$ during the Data-Out phase.

Start Transfer Count Register (00H–01H) Write Only

Start Transfer Count Register
STCREG

Address: 00H–01H
Type: Write

15	14	13	12	11	10	9	8
STVL15	STVL14	STVL13	STVL12	STVL11	STVL10	STVL9	STVL8
X	X	X	X	X	X	X	X

7	6	5	4	3	2	1	0
STVL7	STVL6	STVL5	STVL4	STVL3	STVL2	STVL1	STVL0
X	X	X	X	X	X	X	X

16506A-014A

STCREG – Bits 15:0 – STVL 15:0 – Start Value 15:0

This is a two byte register. It contains the number of bytes to be transferred during a DMA operation. The value of this register is set to the number of bytes to be transferred prior to a DMA transfer command. This register retains its programmed value until it is overwritten and is not affected by hardware or software reset. Therefore, it is not necessary to reprogram the count for subsequent DMA transfers of the same size. Writing a zero to this register sets a maximum transfer count of 65536 bytes. The value in this register is undefined at power-up.

FIFO Register (02H) Read/Write

FIFO Register
FFREG

Address: 02H
Type: Read/Write

7	6	5	4	3	2	1	0
FF7	FF6	FF5	FF4	FF3	FF2	FF1	FF0
0	0	0	0	0	0	0	0

16506A-015A

FFREG – Bits 7:0 – FF 7:0 – FIFO 7:0

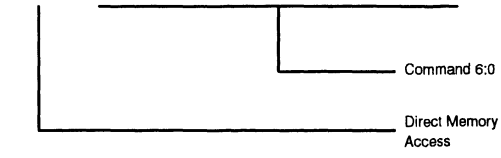
The bottom of the 16x9 FIFO is mapped into the FIFO Register address. By reading and writing this register the bottom of the FIFO can be read or written. This is the only register that can also be accessed by $\overline{DACK}$ along with $\overline{DMARD}$ or $\overline{DMAWR}$. This register is reset to zero by hardware or software reset and also at the start of a selection or reselection sequence.

Command Register (03H) Read/Write

Command Register
CMDREG

Address: 03H
Type: Read/Write

7	6	5	4	3	2	1	0
DMA	CMD6	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0
X	X	X	X	X	X	X	X



16506A-016A

Commands to the device are issued by writing to this register. This register is two deep which allows for command queuing. The second command can be issued before the first one is completed. The Reset command and the Stop DMA command are not queued and are executed immediately. Reading this register will return the command currently being executed (or the last command executed if there are no pending commands).

CMDREG – Bit 7 – DMA – Direct Memory Access

The DMA bit when set notifies the device that the command is a DMA instruction, when reset it is a non-DMA instruction. For DMA instructions the Current Transfer Count Register (CTCREG) will be loaded with the contents of the Start Transfer Count Register (STCREG). The data is then transferred and the CTCREG is decremented for each byte until it reaches zero.

CMDREG – Bits 6:0 – CMD 6:0 – Command 6:0

These command bits decode the commands that the device needs to perform. There are a total of 29 commands grouped into four categories. The groups are Initiator Commands, Target Commands, Selection/ Reselection Commands and General Purpose Commands.

Initiator Commands

CMD6	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0	Command
0	0	1	0	0	0	0	Information Transfer
0	0	1	0	0	0	1	Initiator Command Complete Steps
0	0	1	0	0	1	0	Message Accepted
0	0	1	1	0	0	0	Transfer Pad Bytes
0	0	1	1	0	1	0	Set ATN
0	0	1	1	0	1	1	Reset ATN

Target Commands

CMD6	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0	Command
0	1	0	0	0	0	0	Send Message
0	1	0	0	0	0	1	Send Status
0	1	0	0	0	1	0	Send Data
0	1	0	0	0	1	1	Disconnect Steps
0	1	0	0	1	0	0	Terminate Steps
0	1	0	0	1	0	1	Target Command Complete Steps
0	1	0	0	1	1	1	Disconnect
0	1	0	1	0	0	0	Receive Message Steps
0	1	0	1	0	0	1	Receive Command
0	1	0	1	0	1	0	Receive Data
0	1	0	1	0	1	1	Receive Command Steps
0	0	0	0	1	0	0	DMA Stop

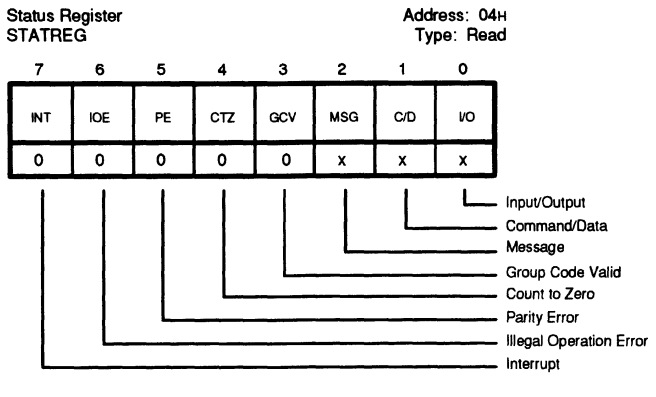
Idle Commands

CMD6	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0	Command
1	0	0	0	0	0	0	Reselect Steps
1	0	0	0	0	0	1	Select without ATN Steps
1	0	0	0	0	1	0	Select with ATN Steps
1	0	0	0	0	1	1	Select with ATN and Stop Steps
1	0	0	0	1	0	0	Enable Selection/Reselection
1	0	0	0	1	0	1	Disable Selection/Reselection
1	0	0	0	1	1	0	Select with ATN3 Steps

General Commands

CMD6	CMD5	CMD4	CMD3	CMD2	CMD1	CMD0	Command
0	0	0	0	0	0	0	No Operation
0	0	0	0	0	0	1	Clear FIFO
0	0	0	0	0	1	0	Reset Device
0	0	0	0	0	1	1	Reset SCSI Bus

Status Register (04H) Read



16506A-017A

This read only register contains flags to indicate the status and phase of the SCSI transactions. It indicates whether an interrupt or error condition exists. It should be read every time the host is interrupted to determine which device is asserting an interrupt. The data is latched until the Interrupt Status Register is read. The phase bits will be latched only if latching is enabled in the Control Register 2, otherwise, it will indicate the current SCSI phase. If command stacking is used, two interrupts might occur. Reading this register will clear the status information for the first interrupt and update the Status Register for the second interrupt.

STATREG – Bit 7 – INT – Interrupt

The INT bit is set when the device asserts the interrupt output. This bit will be cleared by a hardware or software reset. Reading the Interrupt Status Register will deassert the interrupt output and also clear this bit.

STATREG – Bit 6 – IOE – Illegal Operation Error

The IOE bit is set when an illegal operation is attempted. This condition will not cause an interrupt, it will be detected by reading the status register while servicing another interrupt. The following conditions will cause the IOE bit to be set:

- DMA and SCSI transfer directions are opposite.
- FIFO overflows.
- In initiator mode an unexpected phase change detected during synchronous data transfer.
- Command Register overwritten.

This bit will be cleared by reading the Interrupt Status Register or by a hard or soft reset.

STATREG – Bit 5 – PE – Parity Error

The PE bit is set if the parity checking option is enabled in Control Register 1 and the device detects a parity error on incoming SCSI data, command, status or message bytes. Detection of a parity error condition will not cause an interrupt but will be reported with other interrupt causing conditions. When a parity error is detected in the information phase of the initiator mode $\overline{ATN}$ is asserted on the SCSI bus.

This bit will be cleared by reading the Interrupt Status Register or by a hard or soft reset.

STATREG – Bit 4 – CTZ – Count To Zero

The CTZ bit is set when the Current Transfer Count Register (CTCREG) has counted down to zero. This bit will be reset when the CTCREG is written.

Reading the Interrupt Status Register will not affect this bit. This bit will however be cleared by a hard or soft reset.

Note:

A non-DMA NOP will not reset the CTZ bit since it does not load the CTCREG but a DMA NOP will reset this bit since it loads the CTCREG.

STATREG – Bit 3 – GCV – Group Code Valid

The GCV bit is set if the group code field in the Command Descriptor Block (CDB) is one that is defined by the ANSI Committee in their document X3.131 – 1986. If the SCSI-2 Feature Enable (S2FE) bit in the Control Register 2 (CNTLREG2) is set, Group 2 commands will be treated as ten byte commands and the GCV bit will be set. If S2FE is reset then Group 2 commands will be treated as reserved commands. Group 3 and 4 command will always be considered as reserved commands. The device will treat all reserved commands as six byte commands. Group 6 commands will always be treated as vendor unique six byte commands and Group 7 commands will always be treated as vendor unique ten byte commands.

The GCV bit is cleared by reading the Interrupt Status Register (INSTREG) or by a hard or soft reset.

STATREG – Bit 2 – MSG – Message

STATREG – Bit 1 – C/D – Command/Data

STATREG – Bit 0 – I/O – Input/Output

Bit2 MSG	Bit1 C/D	Bit0 I/O	SCSI Phase
1	1	1	Message In
1	1	0	Message Out
1	0	1	Reserved
1	0	0	Reserved
0	1	1	Status
0	1	0	Command
0	0	1	Data_In
0	0	0	Data_Out

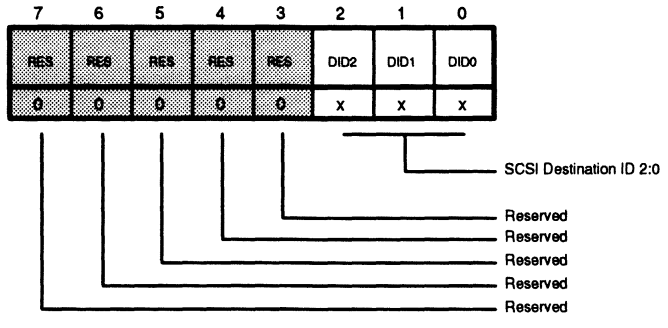
The MSG, C/D and I/O bits together can be referred to as the SCSI Phase bits. They indicate the phase of the SCSI bus. These bits may be latched or unlatched depending on the option selected in Control Register 2 (CNTLREG2) by the Latch SCSI Phase (LSP) bit.

In the latched mode the SCSI phase bits are latched at the end of a command and the latch is opened when the Interrupt Status Register (INSTREG) is read. In the unlatched mode, they indicate the phase of the SCSI bus when this register is read.

SCSI Destination ID Register (04H) Write

SCSI Destination ID Register
SDIDREG

Address: 04H
Type: Write



16506A-018A

SDIDREG – Bits 7:3 – RES – Reserved

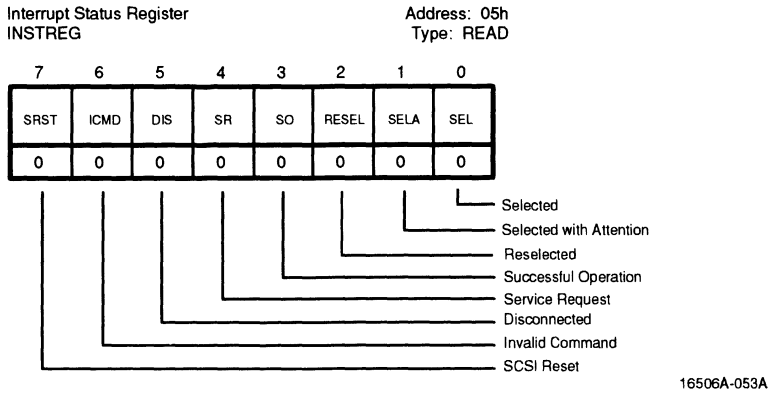
At power-up the state of these bits is undefined. The DID 2:0 bits are not affected by reset.

SDIDREG – Bits 2:0 – DID 2:0 – Destination ID 2:0

The DID 2:0 bits are the encoded SCSI ID of the device on the SCSI bus which needs to be selected or reselected.

DID2	DID1	DID0	SCSI ID
1	1	1	7
1	1	0	6
1	0	1	5
1	0	0	4
0	1	1	3
0	1	0	2
0	0	1	1
0	0	0	0

Interrupt Status Register (05H) Read



The Interrupt Status Register (INSTREG) will indicate the reason for the interrupt. This register is used with the Status Register (STATREG) and Internal Status Register (ISREG) to determine the reason for the interrupt. Reading the INSTREG will clear all three registers.

INSTREG – Bit 7 – SRST – SCSI Reset

The SRST bit will be set if a SCSI Reset is detected and SCSI reset reporting is enabled via the DISR (bit 6) of the CNTLREG1.

INSTREG – Bit 6 – ICMD – Invalid Command

The ICMD bit will be set if the device detects an illegal command code. This bit is also set if a command code from a different mode is detected than the mode the device is currently in.

INSTREG – Bit 5 – DIS – Disconnected

The DIS bit can be set in the target or the initiator mode when the device disconnects from the SCSI bus. In the target mode this bit will be set if a terminate or a command complete sequence causes the device to disconnect from the SCSI bus. In the Initiator mode this bit will be set if the target disconnects or a selection or reselection timeout occurs.

INSTREG – Bit 4 – SR – Service Request

The SR bit can be set in the target or the initiator mode when another device on the SCSI bus has a service request. In the target mode this bit will be set when the initiator asserts the $\overline{ATN}$ signal. In the Initiator mode this bit is set whenever the target requests an information transfer phase.

INSTREG – Bit 3 – SO – Successful Operation

The SO bit can be set in the target or the initiator mode when an operation is successfully complete. In the target mode this bit will be set when any target mode command is completed. In the initiator mode this bit is set after a target has been successfully selected, after a command is successfully completed and after an information transfer command when the target requests a message in phase.

INSTREG – Bit 2 – RESEL – Reselected

The RESEL bit is set at the end of the reselection phase indicating that the device has been reselected as an initiator.

INSTREG – Bit 1 – SELA – Selected with Attention

The SELA bit is set at the end of the selection phase indicating that the device has been selected and that the $\overline{ATN}$ signal was active during the selection.

INSTREG – Bit 0 – SEL – Selected

The SEL bit is set at the end of the selection phase indicating that the device has been selected and that the $\overline{ATN}$ signal was inactive during the selection.

SCSI Timeout Register (05H) Write

SCSI Timeout Register
STIMREG

Address: 05H
Type: Write

7	6	5	4	3	2	1	0
STIM7	STIM6	STIM5	STIM4	STIM3	STIM2	STIM1	STIM0
X	X	X	X	X	X	X	X

16506A-020A

This register determines how long the initiator (target) will wait for a target to respond to a selection (reselection) before timing out. It should be set to yield 250 ms to comply with ANSI standards for SCSI.

STIMREG – Bits 7:0 – STIM 7:0 – SCSI Timer 7:0

The value loaded in STIM 7:0 can be calculated from the following formula:

STIM 7:0 =

[(SCSI Time Out) (Clock Frequency) / (8192 (Clock Factor))]

Example:

SCSI Time Out (in seconds): 250 ms. (Recommended by the ANSI Standard) = 250 x 10–3 s.

Clock Frequency: 20 MHz. (assume) = 20 x 106 Hz.

Clock Factor: CLKF 2:0 from Clock Conversion Register (09H) = 5

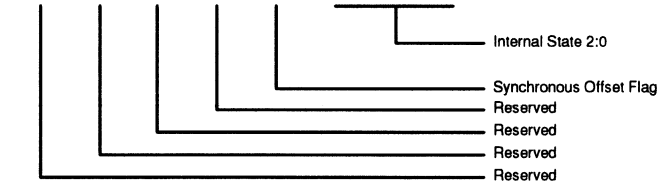
STIM 7:0 = (250 x 10–3) X (20 x 106) / (8192 (5)) = 122 decimal

Internal State Register (06H) Read

Internal State Register
ISREG

Address: 06H
Type: Read

7	6	5	4	3	2	1	0
RES	RES	RES	RES	$\overline{SOF}$	IS2	IS1	IS0
X	X	X	X	0	0	0	0



16506A-019A

The Internal State Register (ISREG) tracks the progress of a sequence-type command. It is updated after each successful completion of an intermediate operation. If an error occurs, the host can read this register to determined at where the command failed and take the necessary procedure for recovery. Reading the Interrupt Status Register will clear this register.

ISREG – Bits 7:4 – RES – Reserved

ISREG – Bit 3 – $\overline{SOF}$ – Synchronous Offset Flag

The SOF is reset when the Synchronous Offset Register (SOFREG) has reached its maximum value.

Note:
The SOF bit is active Low.

ISREG – Bits 2:0 – IS 2:0 – Internal State 2:0

The IS 2:0 bits along with the Interrupt Status Register (INSTREG) indicates the status of the successfully completed intermediate operation. Refer to the Status Decode section for more details.

Initiator Select without ATN Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
0	20	Arbitration steps completed or disconnected or selection time–out
4	18	Selection with ATN steps fully executed
3	18	Sequence halted during command transfer due to premature phase change (target)
2	18	Arbitration and selection completed; sequence halted because target failed to assert command phase

Initiator Select with ATN Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
4	18	Selection with ATN steps fully executed
3	18	Sequence halted during command transfer due to premature phase change; some CDB bytes may not have been sent; check FIFO flags
2	18	Message out completed; sent one message byte with ATN true, then released ATN; sequence halted because target failed to assert command phase after message byte was sent
0	18	Arbitration and selection completed; sequence halted because target did not assert message out phase; ATN still driven by HPSC

Initiator Select with ATN3 Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
0	20	Arbitration steps completed or disconnected or selection time–out
4	18	Selection with ATN3 steps fully executed
3	18	Sequence halted during command transfer due to premature phase change; some CDB bytes may not have been sent; check FIFO flags
2	18	One, two, or three message bytes sent; sequence halted because target failed to assert command phase after third message byte, or prematurely released message out phase; ATN released only if third message byte was sent
0	18	Arbitration and selection completed; sequence halted because target failed to assert message out phase; ATN still driven by HPSC

Initiator Select with ATN and Stop Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
0	20	Arbitration steps completed or disconnected or selection time–out
0	18	Arbitration and selection completed; sequence halted because target failed to assert message out phase; ATN still asserted by HPSC
1	18	Message out completed; one message byte sent; ATN on

Target Selected without ATN Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
2	11	Selected; received entire CDB; check valid group status bit; initiator asserted ATN in command phase
1	11	Sequence halted in command phase due to parity error; some CDB bytes may not have been received; check FIFO flags; initiator asserted ATN in command phase
2	01	Selected; received entire CDB; check valid group status bit
1	01	Sequence halted in command phase because of parity error; some CDB bytes may not have been received; check FIFO flags
0	01	Selected; loaded bus ID into FIFO; null–byte message loaded into FIFO

Target Select with ATN Steps, SCSI–2 Bit NOT SET

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
2	12	Selection complete; received one message byte and entire CDB; initiator asserted ATN during command phase
1	12	Halted in command phase; parity error and ATN true
0	12	Selected with ATN; stored bus ID and one message byte; sequence halted because ATN remained true after first message byte
2	02	Selection completed; received one message byte and the entire CDB
1	02	Sequence halted in command phase because of parity error; some CDB bytes not received; check valid group code bit and FIFO flags
0	02	Selected with ATN; stored bus ID and one message byte; sequence halted because of parity error or invalid ID message

Target Select with ATN Steps, SCSI–2 Bit SET

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
5	12	Halted in command phase; parity error and ATN true
4	12	ATN remained true after third message byte
0	02	Selected with ATN; stored bus ID and one message byte; sequence halted because of parity error or invalid ID message
6	02	Selection completed; received three message bytes and the entire CDB
5	02	Received three message bytes then halted in command phase because of parity error; some CDB bytes not received; check valid group code bit and FIFO flags
402Parity error during second or third message byte		

Target Receive Command Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
2	18	Received entire CDB; initiator asserted ATN
1	18	Sequence halted during command transfer due to parity error; ATN asserted by initiator
2	08	Received entire CDB
1	08	Sequence halted during command transfer due to parity error; check FIFO flags

Target Disconnect Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
2	28	Disconnect steps fully executed; disconnected; bus is free
1	18	Two message bytes sent; sequence halted because initiator asserted ATN
0	18	One message byte sent; sequence halted because initiator asserted ATN

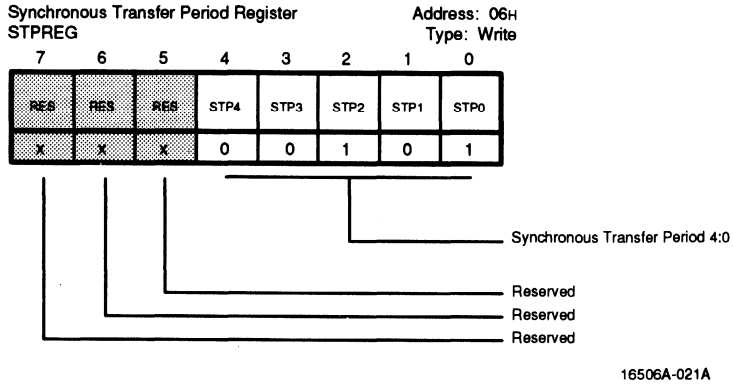
Target Terminate Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
2	28	Terminate steps fully executed; disconnected; bus is free
1	18	Status and message bytes sent; sequence halted because initiator asserted ATN
0	18	Status byte sent; sequence halted because initiator asserted ATN

Target Command Complete Steps

Internal State Register (06H) Bits 2:0 (Hex)	Interrupt Status Register (05H) Bits 7:0 (Hex)	Explanation
1	18	Status and message bytes sent; sequence halted because initiator set ATN
0	18	Status byte sent; sequence halted because initiator set ATN
2	08	Command complete steps fully executed

Synchronous Transfer Period Register (06H) Write



The Synchronous Transfer Period Register (STPREG) contains a 5-bit value indicating the number of clock cycles each byte will take to be transferred over the SCSI bus in synchronous mode. The minimum value allowed is 5. The STPREG defaults to five after a hard or soft reset.

STPREG – Bits 7:5 – RES – Reserved

STPREG – Bits 4:0 – STP 4:0 – Synchronous Transfer Period 4:0

The STP 4:0 bits are programmed to specify the synchronous transfer period or the number of clock cycles for each byte transfer in the synchronous mode. The minimum value for STP 4:0 is five. Missing table entries follow the binary code.

STP4	STP3	STP2	STP1	STP0	Clocks/ Byte
0	0	1	0	0	5
0	0	1	0	1	5
0	0	1	1	0	6
0	0	1	1	1	7
•	•	•	•	•	•
•	•	•	•	•	•
1	1	1	1	1	31
0	0	0	0	0	32
0	0	0	0	1	33
0	0	0	1	0	34
0	0	0	1	1	35

Current FIFO/Internal State Register (07H) Read

Current FIFO/Internal State Register
CFISREG

Address: 07H
Type: Read

7	6	5	4	3	2	1	0
IS2	IS1	IS0	CF4	CF3	CF2	CF1	CF0
0	0	0	0	0	0	0	0

Current FIFO 4:0

Internal State 2:0

16506A-022A

This register has two fields, the Current FIFO field and the Internal State field.

CFISREG – Bits 7:5 – IS 2:0 – Internal State 2:0

The Internal State Register (ISREG) tracks the progress of a sequence-type command.

The IS 2:0 bits are duplicated from the IS 2:0 field in the Internal State Register (ISREG) in the normal mode. If the device is in the test mode, IS 0 is set to indicate that the offset value is non zero. A non zero value indicates that synchronous data transfer can continue. A zero value indicates that the synchronous offset count has been reached and no more data can be transferred until an acknowledge is received.

CFISREG – Bits 4:0 – CF 4:0 – Current FIFO 4:0

The CF 4:0 bits are the binary coded value of the number of bytes in the FIFO. These bits should not be read when the device is transferring data since this count may not be stable.

Synchronous Offset Register (07H) Write

Synchronous Offset Register
SOFREG

Address: 07H
Type: Write

7	6	5	4	3	2	1	0
RES	RES	RES	RES	SO3	SO2	SO1	SO0
X	X	X	X	0	0	0	0

Synchronous Offset 3:0

Reserved

Reserved

Reserved

Reserved

16506A-023A

The Synchronous Offset Register (SOFREG) contains a 4-bit count of the number of bytes that can be sent to (or received from) the SCSI bus without an $\overline{ACK}$ (or $\overline{REQ}$). Bytes exceeding the threshold will be sent one byte at a time (asynchronously). That is, each byte will require an $\overline{ACK}/\overline{REQ}$ handshake. To set up an asynchronous transfer, the SOFREG is set to zero. The SOFREG is set to zero after a hard or soft reset.

SOFREG – Bits 7:4 – RES – Reserved

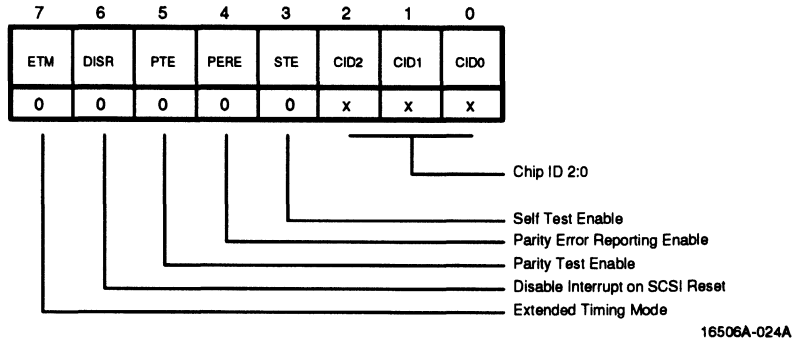
SOFREG – Bits 3:0 – SO 3:0 – Synchronous Offset 3:0

The SO 4:0 bits are the binary coded value of the number of bytes that can be sent to (or received from) the SCSI bus without an $\overline{ACK}$ (or $\overline{REQ}$) signal.

Control Register One (08H) Read/Write

Control Register One
CNTLREG1

Address: 08H
Type: Read/Write



16506A-024A

The Control Register 1 (CNTLREG1) sets up the device with various operating parameters.

CNTLREG1 – Bit 7 – ETM – Extended Timing Mode

The ETM bit is set if an extra clock period is required between the data being driven on the bus and the $\overline{REQ}$ or $\overline{ACK}$ being asserted. This is some times necessary in high capacitive loading environments. The ETM bit is reset to zero by a hard or soft reset.

CNTLREG1 – Bit 6 – DISR – Disable Interrupt on SCSI Reset

The DISR bit masks the reporting of the SCSI reset. When the DISR bit is set and a SCSI reset is asserted, the device will disconnect from the SCSI bus and remain idle without interrupting the host processor. When the DISR bit is reset and a SCSI reset is asserted the device will respond by interrupting the host processor. The DISR bit is reset to zero by a hard or soft reset.

CNTLREG1 – Bit 5 – PTE – Parity Test Enable

The PTE bit is for test use only. When the PTE bit is set the parity on the output (SCSI or host processor) bus is forced to the value of the MSB (bit 7) of the output data from the internal FIFO. This allows parity errors to be created to test the hardware and software. The PTE bit is reset to zero by a hard or soft reset.

CNTLREG1 – Bit 4 – PERE – Parity Error Reporting Enable

The PERE bit enables the checking and reporting of parity errors on incoming SCSI bytes during the information transfer phase. When the PERE bit set and a bad parity is detected, the PE bit in the STATREG is will be set but an interrupt will not be generated. In the initiator mode the $\overline{ATN}$ signal will also be asserted on the SCSI bus. When the PERE bit is reset and a bad parity occurs it is not detected and no action is taken.

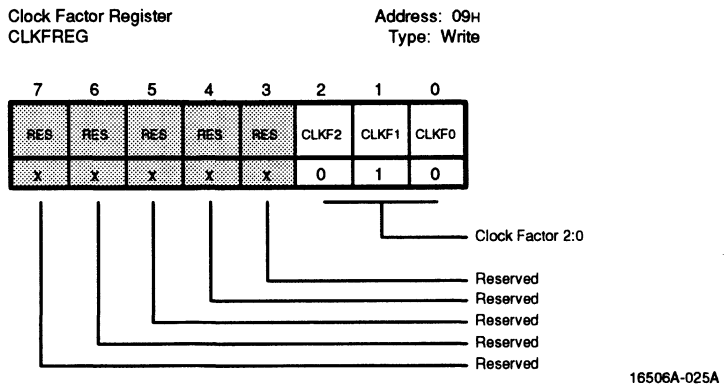
CNTLREG1 – Bit 3 – STE – Self Test Enable

The STE bit is for test use only. When the STE bit is set the device is placed in a test mode which enables the device to access the test register at address 0AH. To reset this bit and to resume normal operation the device must be issued a hard or soft reset.

CNTLREG1 – Bit 2:0 – CID 2:0 – Chip ID 2:0

The Chip ID 2:0 bits specify the binary coded value of the device ID on the SCSI bus. The device will arbitrate with this ID and will respond to selection or reselection to this ID. At power-up the state of these bit are undefined. These bits are not affected by hard or soft reset.

Clock Factor Register (09H) Write



The Clock Factor Register (CLKFREG) must be set to indicate the input frequency range of the device. This value is crucial for controlling various timings to meet the SCSI specification. The selector can be calculated by rounding off the quotient of (Input Clock Frequency in MHz)/(5 MHz). The device has a frequency range of 10 to 25 MHz.

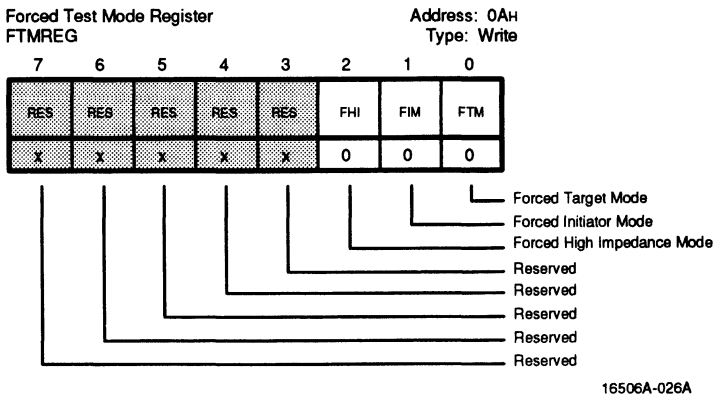
CLKF2	CLKF1	CLKF0	Input Clock Frequency in MHz
0	1	0	10
0	1	1	10.01 to 15
1	0	0	15.01 to 20
1	0	1	20.01 to 25

CLKFREG – Bits 7:3 – RES – Reserved

CLKFREG – Bits 2:0 – CLKF 2:0 – Clock Factor 2:0

The CLKF 2:0 bits specify the binary coded value of the clock factor. The CLKF 2:0 bits will default to a value of 2 by a hard or soft reset.

Forced Test Mode Register (0AH) Write



The Forced Test Mode Register (FTMREG) is for test use only. The STE bit in the CNTLREG1 must be set for the FTMREG to operate.

FTMREG – Bits 7:3 – RES – Reserved

FTMREG – Bit 2 – FHI – Forced High Impedance Mode

The FHI bit when set places all the output and bidirectional pins into a high impedance state.

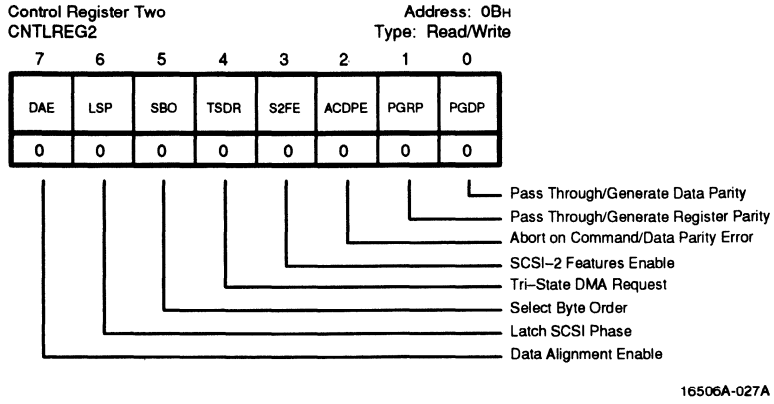
FTMREG – Bit 1 – FIM – Forced Initiator Mode

The FIM bit when set forces the device into the initiator mode. The device will then execute all initiator commands irrespective of the SCSI bus status.

FTMREG – Bit 0 – FTM – Forced Target Mode

The FTM bit when set forces the device into the target mode. The device will then execute all target commands irrespective of the SCSI bus status.

Control Register Two (0BH) Read/Write



The Control Register 2 (CNTLREG2) sets up the device with various operating parameters.

CNTLREG2 – Bit 7 – DAE – Data Alignment Enable

The DAE bit is used in the initiator Synchronous Data-In phase only. When the DAE bit is set one byte is reserved at the end of the FIFO when the phase changes to the Synchronous Data-In phase. The contents of this byte will become the lower byte of the DMA word (16-bit) transfer to the memory, the upper byte being the first byte of the first word received from the SCSI bus.

Note:

If an interrupt is received for a misaligned boundary on a phase change to synchronous data the following recovery procedure may be followed. The host processor should copy the byte at the start address in the host memory to the Data Alignment Register 0FH (DALREG) and then issue an information transfer command. The first word the device will write to the memory (via DMA) will consists of the lower byte from the DALREG and the upper byte from the first byte received from the SCSI bus.

The DAE bit must be set before the phase changes to the Synchronous Data-In. The DAE bit is reset to zero by a hard or soft reset or by writing the DALREG when interrupted in the Synchronous Data-In phase.

CNTLREG2 – Bit 6 – LSP – Latch SCSI Phase

The LSP bit is used to enable or disable the latching of the SCSI phase bits (MSG, C/D and I/O) in the Status Register (STATREG) 04H.

When the LSP bit is set the phase bits STSTREG – Bits 2:0 are latched at the end of each command. This simplifies software for stacked commands. When the LSP bit is reset the phase bits STATREG – Bits 2:0 reflect the actual state of the SCSI phase lines at any instant of time. The LSP bit is reset by a hard or soft reset.

CNTLREG2 – Bit 5 – SBO – Select Byte Order

The SBO bit is used only when the BUSMD 1:0 = 10 to enable or disable the byte control on the DMA interface. When SBO is set and the BUSMD 1:0 = 10, the byte control inputs BHE and AS0 control the byte positions. When SBO is reset the byte control inputs BHE and AS0 are ignored.

CNTLREG2 – Bit 4 – TSDR – Tri-State DMA Request

The TSDR bit when set sends the DREQ output signal to high impedance state and the device ignores all activity on the DMA request (DREQ) input. This is useful for wiring-OR several devices that share a common DMA request line. When the TSDR bit is reset the DREQ output is driven to TTL levels.

CNTLREG2 – Bit 3 – S2FE – SCSI–2 Features Enable

The S2FE bit allows the device to recognize two SCSI-2 features. The two features are extended message feature and the Group 2 command recognition.

Extended Message Feature: When the S2FE bit is set and the device is selected with attention, the device will monitor the $\overline{ATN}$ signal at the end of the first message byte. If the $\overline{ATN}$ signal is active, the device will request two more message bytes before switching to the command phase. If the $\overline{ATN}$ signal is inactive the device will switch to the command phase. When the S2FE bit is reset the device as a target will request a single message byte. As an initiator, the device will abort the selection

sequence if the target does not switch to the command phase after receiving a single message byte.

Group 2 Command Recognition: When the S2FE bit is set the group 2 commands are recognized as 10 byte commands. The GCV (Group Code Valid) bit in the STATREG (04H) is set. When the S2FE bit is reset, the device will interpret the group 2 commands as reserved commands and will request 6 byte commands. The GCV bit in the STATREG will not be set in this case.

CNTLREG2 – Bit 2 – ACDPE – Abort on Command/ Data Parity Error

The ACDPE bit when set allows the device to abort a command or data transfer when a parity error is detected. When the ACDPE bit is reset parity error is ignored.

CNTLREG2 – Bit 1 – PGRP – Pass Through/Generate Register Parity

The PGRP bit when set causes the data along with the parity from the host to pass through to the FIFO under the control of the $\overline{CS}$ and the $\overline{WR}$ signals. When the PGRP bit is reset, the device generates the parity on the data from the host before writing it to the FIFO.

When the device is placing the data on the SCSI bus, it will check for an outgoing parity error if either the PGRP bit is set or the PGDP (Pass Through/Generate Data Parity) bit is set.

CNTLREG2 – Bit 0 – PGDP – Pass Through/Generate Data Parity

The PGDP bit when set causes the data along with the parity from the host to pass through to the FIFO under the control of the $\overline{DACK}$ and the $\overline{WR}$ signals. When the PGDP bit is reset, the device generates the parity on the data from the host before writing it to the FIFO.

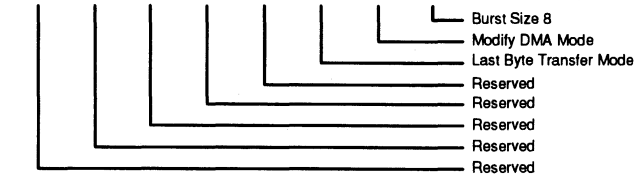
When the device is placing the data on the SCSI bus, it will check for an outgoing parity error if either the PGDP bit is set or the PGRP (Pass Through/Generate Register Parity) bit is set.

Control Register Three (0CH) Read/Write

Control Register Three
CNTLREG3

Address: 0CH
Type: Read/Write

7	6	5	4	3	2	1	0
RES	RES	RES	RES	RES	LBTM	MDM	BS8
0	0	0	0	0	0	0	0



16506A-028A

CNTLREG3 – Bits 7:3 – RES – Reserved

CNTLREG3 – Bit 2 – LBTM – Last Byte Transfer Mode

The LBTM bit specifies how the last byte in an odd byte transfer is handled during 16-bit DMA transfers. This mode is not used if byte control is selected via BUSMD 1:0 inputs and BSO (Byte Select Order) bit in the CNTLREG2. This mode has no affect during 8-bit DMA transfers and on transfers on the SCSI bus.

When the LBTM bit is set the DREQ signal will not be asserted for the last byte, instead the host will read or write the last byte from or to the FIFO. When the LBTM bit is reset the DREQ signal will be asserted for the last byte and the following 16-bit DMA transfer will contain the last byte on the lower bus. If the transfer is a DMA read the upper bus will be all ones.

The LBTM bit is reset by hard or soft reset.

CNTLREG3 – Bit 1 – MDM – Modify DMA Mode

The MDM bit is used to modify the timing of the $\overline{DACK}$ signal with respect to the $\overline{DMARD}$ and $\overline{DMAWR}$ signals. The MDM bit is used in conjunction with the Burst Size 8 (BS8) bit in the CNTLREG3. Both bits have to be configured for proper device operation.

When the MDM bit is set and the device is in a DMA read or write mode the $\overline{DACK}$ signal will remain asserted while the data is strobed by the $\overline{DMARD}$ or $\overline{DMAWR}$ signals. In the DMA read mode when BUSMD 1:0 = 11 the $\overline{DACK}$ signal will toggle for every DMA read.

When the MDM bit is reset and the device is in a DMA read or write mode the $\overline{DACK}$ signal will toggle every time the data is strobed by the $\overline{DMARD}$ or $\overline{DMAWR}$ signals.

CNTLREG3 – Bit 0 – BS8 – Burst Size 8

The BS8 bit is used to modify the timing of the DREQ signal with respect to the $\overline{\text{DMARD}}$ and $\overline{\text{DMAWR}}$ signals. The BS8 bit may be used in conjunction with the Modify DMA Mode (MDM) bit in the CNTLREG3 to enable the Burst PMA mode. Both bits have to be set for proper operation.

When the BS8 bit is set the device delays the assertion of the DREQ signal until 8 bytes or 4 words transfer is possible.

When the BS8 bit is set and the device is in a DMA write mode the DREQ signal will be asserted only when 8 byte locations are available for writing. In the DMA read mode the DREQ signal will go active under the following circumstances:

At the end of a transfer,

- In the target mode,
  - when the transfer is complete

  or
  - when the $\overline{\text{ATN}}$ signal is active
- In the initiator mode,
  - when the Current Transfer Register is decremented to zero

  or
  - after any phase change

In the middle of a transfer

- In the initiator mode,
  - when the last 8 bytes of the FIFO are full
  - during Synchronous Data-In transfer when the Event Transfer Count Register is greater than 7 and the last 8 bytes of the FIFO are full.

When the BS8 bit is reset and the device is in a DMA read or write mode the DREQ signal will toggle every time the data is strobed by the $\overline{\text{DMARD}}$ or $\overline{\text{DMAWR}}$ signals.

Using Bit 0 (BS8) and Bit 1 (MDM) of Control Register 3, one can enable the different combination modes shown in the table below.

(MDM) Bit 1	(BS8) Bit 0	Function	Maximum Synchronous Offset
0	0	Normal DMA Mode	15
0	1	Burst Size 8 Mode	7
1	0	Reserved	–
1	1	Modified DMA Mode	15

Data Alignment Register (0FH) Write

Data Alignment Register
DALREG

Address: 0FH
Type: Write

7	6	5	4	3	2	1	0
DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
0	0	0	0	0	0	0	0

16506A-029A

The Data Alignment Register (DALREG) is used if the first byte of a 16-bit DMA transfer from the SCSI bus to the host processor is misaligned. Prior to issuing an information transfer command, the host processor must set the Data Alignment Enable (DAE) bit in the CNTLREG2.

DALREG – Bits 7:0 – DA 7:0 – Data Alignment 7:0

COMMANDS

The device commands can be broadly divided into two categories, DMA commands and non-DMA commands. DMA commands are those which cause data movement between the host memory and the SCSI bus while non-DMA commands are those that cause data movement between the device FIFO and the SCSI bus. The MSB of the command byte differentiates DMA commands from non-DMA commands.

Summary of Commands

Command	Command Code (Hex)	
	Non-DMA Mode	DMA Mode
Initiator Commands		
Information Transfer	10	90
Initiator Command Complete Steps	11	91
Message Accepted	12	-
Transfer Pad Bytes	18	98
Set $\overline{ATN}$	1A	-
Reset $\overline{ATN}$	1B	-
Target Commands		
Send Message	20	A0
Send Status	21	A1
Send Data	22	A2
Disconnect Steps	23	A3
Terminate Steps	24	A4
Target Command Complete Steps	25	A5
Disconnect	27	A7
Receive Message	28	A8
Receive Command Steps	29	A9
Receive Data	2A	AA
Receive Command Steps	2B	AB
Target Abort DMA	04	84

Command	Command Code (Hex)	
	Non-DMA Mode	DMA Mode
Idle State Commands		
Reselect Steps	40	C0
Select without $\overline{ATN}$ Steps	41	C1
Select with $\overline{ATN}$ Steps	42	C2
Select with $\overline{ATN}$ and Stop Steps	43	C3
Enable Selection/Reselection	44	C4
Disable Selection/Reselection		45
Select with $\overline{ATN3}$	46	C6
General Commands		
No Operation	00	80
Clear FIFO	01	81
Reset Device	02	82
Reset SCSI bus	03	83

COMMAND DESCRIPTION

Initiator Commands

Initiator commands are executed by the device when it is in the initiator mode. If the device is not in the initiator mode and an initiator command is received the device will ignore the command, generate an illegal command interrupt and clear the Command Register (CMDREG) 03H.

Information Transfer Command (Command Code 10H/90H)

The Information Transfer command is used to transfer information bytes over the SCSI bus. This command may be issued during any SCSI Information Transfer phase. Information transfer for synchronous data must use the DMA mode.

The device will continue to transfer information until it is terminated by any one of the following conditions:

- The target changes the SCSI bus phase before the expected number of bytes are transferred. The device clears the Command Register (CMDREG) 03H, and generates a service interrupt when the target asserts $\overline{REQ}$.
- Transfer is successfully complete. If the phase is Message Out, the device deasserts $\overline{ATN}$ before asserting $\overline{ACK}$ for the last byte of the message. When the target asserts $\overline{REQ}$, a service interrupt is generated.
- In the Message In phase when the device receives the last byte. The device keeps the $\overline{ACK}$ signal asserted and generates a Successful Operation interrupt.

During synchronous data transfers the target may send up to the maximum synchronous threshold number of $\overline{REQ}$ pulses to the initiator. If it is the Synchronous Data-In phase then the target sends the data and the $\overline{REQ}$ pulses. These bytes are stored by the initiator in the FIFO as they are received.

Information Transfer Command when issued during the following SCSI phases and terminating in synchronous data phases, is handled as described below:

- Message In/Status Phase —When a phase change to Synchronous Data-In or Synchronous Data-Out is detected by the device, the Command Register (CMDREG) 03H is cleared and the DMA interface is disabled to disallow any transfer of data phase bytes. If the phase change is to Synchronous Data-In and bad parity is detected on the data bytes coming in, it is not reported since the Status Register (STATREG) 04H will report the status of the command just completed. The parity error flag and the $\overline{ATN}$ signal will be asserted when the Transfer Information command begins execution.
- Message Out/Command Phase – When a phase change to Synchronous Data-In or Synchronous Data-Out is detected by the device, the Command Register (CMDREG) 03H is cleared and the DMA interface is disabled to allow any transfer of data phase bytes. If the phase change is to Synchronous Data-In and bad parity is detected on the data bytes coming in, it is not reported since the Status Register (STATREG) 04H will report the status of the command just completed. The parity error flag and the ATN signal will be asserted when the Transfer Information command begins execution. The FIFO Register29 (FFREG) 02H will be latched and will remain in that condition until the next command begins execution. The value in the FFREG indicates the number of bytes in the FIFO when the phase changed to Synchronous Data-In. These bytes are cleared from the FIFO, which now contains only the incoming data bytes.
- In the Synchronous Data-Out phase, the threshold counter is incremented as $\overline{REQ}$ pulses are received. The transfer is completed when the FIFO is empty and the Current Transfer Count Register (CTCREG) 00H–01H is zero. The threshold counter will not be zero.
- In the Synchronous Data-In phase, the Current Transfer Count Register (CTCREG) is decremented as bytes are read from the FIFO rather than being decremented when the bytes are being written to the FIFO. The transfer is completed when Current Transfer Count Register (CTCREG) is zero but the FIFO may not be empty.

Initiator Command Complete Steps (Command Code 11H/91H)

The Initiator Command Complete Steps command is normally issued when the SCSI bus is in the Status In phase. One Status byte followed by one Message byte is transferred if this command completes normally. After receiving the message byte the device will keep the $\overline{ACK}$ signal asserted to allow the initiator to examine the message and assert the $\overline{ATN}$ signal if it is unacceptable. The command terminates early if the target does not switch to the Message In phase or if the target disconnects from the SCSI bus.

Message Accepted Command (Command Code 12H)

The Message Accepted Command is used to release the $\overline{ACK}$ signal. This command is normally used to complete a Message In handshake. Upon execution of this command the device generates a service request interrupt after $\overline{REQ}$ is asserted by the target.

After the device has received the last byte of message, it keeps the $\overline{ACK}$ signal asserted. This allows the device to either accept or reject the message. To accept the message, Message Accepted Command is issued. To reject the message the $\overline{ATN}$ signal must be asserted (with the help of the Set $\overline{ATN}$ Command) before issuing the Message Accepted Command. In either case the Message Accepted Command has to be issued to release the $\overline{ACK}$ signal.

Transfer Pad Bytes Command (Command Code 18H/98H)

The Transfer Pad Bytes Command is used to recover from an error condition. This command is similar to the Information Transfer Command, only the information bytes consists of null data. It is used when the target expects more data bytes than the initiator has to send. It is also used when the initiator receives more information than it expected from the target.

When sending data to the SCSI bus, the FIFO is loaded with null bytes and these bytes are sent out to the SCSI bus. DMA has to be enabled when pad bytes are transferred to the SCSI bus. No actual DMA requests are made but the device uses the Current Transfer Count Register (CTCREG) 00H–01H to terminate the transfer.

When receiving data from the SCSI bus, the device will receive the pad bytes and place them on the top of the FIFO and unload them from the bottom of the FIFO.

The command terminates under the same condition as the Information Transfer Command, only the device does not keep the $\overline{ACK}$ signal asserted during the last byte of the Message In phase. If this command terminates prematurely, due to a disconnect or a phase change, before the CTCREG decrements to zero, the FIFO may contain residual pad bytes.

Set $\overline{ATN}$ Command (Command Code 1AH)

The Set $\overline{ATN}$ Command is used to drive the $\overline{ATN}$ signal active on the SCSI bus. An interrupt is not generated at the end of this command. The $\overline{ATN}$ signal is deasserted before asserting the $\overline{ACK}$ signal during the last byte of the Message Out phase.

Note: *The $\overline{ATN}$ signal is asserted by the device without this command in the following cases:*

- If any select with $\overline{ATN}$ command is issued and the arbitration is won.
- An initiator needs the target's attention to send a message. The $\overline{ATN}$ signal is asserted before deasserting the $\overline{ACK}$ signal.

Reset $\overline{ATN}$ Command (Command Code 1BH)

The Reset $\overline{ATN}$ Command is used to deassert the $\overline{ATN}$ signal on the SCSI bus. An interrupt is not generated at the end of this command. This command is used only when interfacing with devices that do not support the Common Command Set (CCS). These older devices do not deassert their $\overline{ATN}$ signal automatically on the last byte of the Message Out phase. This device does deassert its $\overline{ATN}$ signal automatically on the last byte of the Message Out phase.

Target Commands

Target commands are executed by the device when it is in the target mode. If the device is not in the target mode and a target command is received the device will ignore the command, generate an illegal command interrupt and clear the Command Register (CMDREG) 03H.

A SCSI bus reset during any target command will cause the device to abort the command sequence , flag a SCSI bus reset interrupt (if the interrupt is enabled) and disconnect from the SCSI bus.

Normal or successful completion of a target command will cause a Successful Operation interrupt to be flagged. If the $\overline{ATN}$ signal is asserted during a target command sequence the Service Request bit is asserted in the Interrupt Status Register (INSTREG) 05H. If the $\overline{ATN}$ signal is asserted when the device is in an idle state a Service Request interrupt will be generated, the Successful Operation bit in the INSTREG will be reset and the CMDREG cleared.

Send Message Command (Command Code 20H/A0H)

The Send Message Command is used by the target to inform the initiator to receive a message. The SCSI bus phase lines are set to the Message In Phase and message bytes are transferred from the device FIFO to the buffer memory.

Send Status Command (Command Code 21H/A1H)

The Send Status Command is used by the target to inform the initiator to receive status information. The SCSI bus phase lines are set to the Status Phase and status bytes are transferred from the target device to the initiator device.

Send Data Command (Command Code 22H/A2H)

The Send Data Command is used by the target to inform the initiator to receive data bytes. The SCSI bus phase lines are set to the Data-In Phase and data bytes are transferred from the target device to the initiator device.

Disconnect Steps Command (Command Code 23H/A3H)

The Disconnect Steps Command is used by the target to disconnect from the SCSI bus. This command consists of two steps. The first step consists of sending two bytes of the Save Data Pointers commands by the target in the Message In Phase. In the second step the target disconnects from the SCSI bus. Successful Operation and Disconnected bits are set in the Interrupt Status Register (INSTREG) 05H upon command completion. If $\overline{ATN}$ signal is asserted by the initiator then Successful Operation and Service Request bits are set in the INSTREG, the CMDREG is cleared and Disconnect Steps Command terminates without disconnecting.

Terminate Steps Command (Command Code 24H/A4H)

The Terminate Steps Command is used by the target to disconnect from the SCSI bus. This command consists of three steps. The first step consists of sending one status byte by the target in the Status Phase. The second step consists of sending one message byte by the target in the Message In Phase. As the third step the target disconnects from the SCSI bus. Successful Operation and Disconnected bits are set in the Interrupt Status Register (INSTREG) 05H upon command completion. If $\overline{ATN}$ signal is asserted by the initiator then Successful Operation and Service Request bits are set in the INSTREG, the CMDREG is cleared and Terminate Steps Command terminates without disconnecting.

Target Command Complete Steps Command (Command Code 25H/A5H)

The Target Command Complete Steps Command is used by the target to inform the initiator of a linked command completion. This command consists of two steps. The first step consists of sending one status byte by the target in the Status Phase. The second step consists of sending one message byte by the target in the Message In Phase. The Successful Operation bit is set in the Interrupt Status Register (INSTREG) 05H upon command completion. If $\overline{ATN}$ signal is asserted by the initiator then Successful Operation and Service Request bits are set in the INSTREG, the CMDREG is cleared and Target Command Complete Steps Command terminates prematurely.

Disconnect Command (Command Code 27H/A7H)

The Disconnect Command is used by the target to disconnect from the SCSI bus. All SCSI bus signals except $\overline{RSTC}$ are released and the device returns to the Disconnected state. The $\overline{RSTC}$ signal is driven active for about 25 micro seconds (depending on clock frequency and clock factor). Interrupt is not generated to the microprocessor.

Receive Message Steps Command (Command Code 28H A8H)

The Receive Message Steps Command is used by the target to request message bytes from the initiator. During this command the target receives the message bytes from the initiator while the SCSI bus is in the Message Out Phase. The Successful Operation bit is set in the Interrupt Status Register (INSTREG) 05H upon command completion. If $\overline{ATN}$ signal is asserted by the initiator then Successful Operation and Service Request bits are set in the INSTREG, the CMDREG is cleared. If a parity error is detected, the device ignores the received message bytes until $\overline{ATN}$ signal is asserted, the Successful Operation bit is set in the INSTREG, and the CMDREG is cleared.

Receive Commands Command (Command Code 29H/A9H)

The Receive Commands Command is used by the target to request the initiator for command bytes. During this command the target receives the command bytes from the initiator while the SCSI bus is in the Command Phase. The Successful Operation bit is set in the Interrupt Status Register (INSTREG) 05H upon command completion. If $\overline{ATN}$ signal is asserted by the initiator then Successful Operation and Service Request bits are set in the INSTREG, the CMDREG is cleared and the command terminates prematurely. If a parity error is detected, the device continues to receive command bytes until the transfer is complete if the Abort on Command/ Data Parity Error (ACDPE) bit in the Control Register (CNTLREG2) 0BH is reset. If the ACDPE bit is set, the command is terminated immediately. The Parity Error (PE) bit in the Status Register (STATREG) 04H is set and CMDREG is cleared.

Receive Data Command (Command Code 2AH/AAH)

The Receive Data Command is used by the target to request the initiator for data bytes. During this command the target receives the data bytes from the initiator while the SCSI bus is in the Data-Out Phase. The Successful Operation bit is set in the Interrupt Status Register (INSTREG) 05H upon command completion. If $\overline{ATN}$ signal is asserted by the initiator then Successful Operation and Service Request bits are set in the INSTREG, the CMDREG is cleared and the command terminates prematurely. If a parity error is detected, the device continues to receive data bytes until the transfer is complete if the Abort on Command/Data Parity Error (ACDPE) bit in the Control Register (CNTLREG2) 0BH is reset. If the ACDPE bit is set, the command is terminated immediately. The Parity Error (PE) bit in the Status Register (STATREG) 04H is set and CMDREG is cleared.

Receive Command Steps Command (Command Code 2BH/ABH)

The Receive Command Steps Command is used by the target to request the initiator for command information bytes. During this command the target receives the command information bytes from the initiator while the SCSI bus is in the Command Phase.

The target device determines the command byte length from the first command byte. If an unknown length is received, the Start Transfer Count Register (STCREG) 00H–01H is loaded with 5 and the Group Code Valid (GCV) bit in the Status Register (STATREG) 04H is reset. If a valid length is received, the STCREG is loaded with the appropriate value and the GCV bit in the STATREG is set. If $\overline{ATN}$ signal is asserted by the initiator then the Service Request bit is set in the INSTREG, and the CMDREG is cleared If a parity error is detected, the command is terminated prematurely and the CMDREG is cleared.

DMA Stop Command (Command Code 04H/84H)

The DMA Stop Command is used by the target to allow the microprocessor to discontinue data transfers due to a lack of activity on the DMA channel. This command is executed from the top of the command queue. If there is a queued command waiting execution, it will be overwritten and the Illegal Operation Error (IOE) bit in the

Status Register (STATREG) 04H will be set. This command is cleared from the command queue once it is decoded.

Caution must be exercised when using this command.

The DMA Stop Command can be used only during a DMA Target Send Data Command or DMA Target Receive Data Command execution. In both cases the DMA controller has to be in the idle state.

During a DMA Target Send Data Command the FIFO has to be empty or the Current FIFO (CF 4:0) bits in the Current FIFO/Internal State Register (CFISREG) 07H are zero.

During a DMA Synchronous Target Receive Data Command the Current Transfer Count Register (CTCREG) 00–01H is zero, which is indicated by the Count to Zero (CTZ) bit of the Status Register (STATREG) 04H. or when the Synchronous Offset Register (SOFREG) 07H has reached its maximum value which is indicated by the Synchronous Offset Flag (SOF) bit of the Internal State Register (ISREG) 06H.

During a DMA Asynchronous Target Receive Data Command the FIFO is full which is indicated by the Current FIFO (CF 4:0) bits in the Current FIFO/Internal State Register (CFISREG) 07H being all high or Current Transfer Count Register (CTCREG) 00–01H is zero, which is indicated by the Count to Zero (CTZ) bit of the Status Register (STATREG) 04H.

Idle State Commands

The Idle State Commands can be issued to the device only when the device is disconnected from the SCSI bus. If these commands are issued to the device when it is logically connected to the SCSI bus, the commands are ignored, and the device will generate an illegal command interrupt and clear the Command Register (CMDREG) 03H.

Reselect Steps Command (Command Code 40H/C0H)

The Reselect Steps Command is used by the target device to reselect an initiator device. When this command is issued the device arbitrates for the control of the SCSI bus. If the device wins arbitration, it reselects the initiator device and transfers a single byte identify message. Before issuing this command the SCSI Timeout Register (STIMREG) 05H, the Control Register One (CNTLREG1) 08H and the SCSI Destination ID Register (SDIDREG) 04H must be set to the proper values. If DMA is enabled, the Start Transfer Count Register (STCREG) 00H–01H must be set to one. If DMA is not enabled, the single byte identify message must be loaded into the FIFO before issuing this command. This command will be terminated early if the SCSI Timeout Register times out. This command also resets the Internal State Register (ISREG) 06H.

Select without ATN Steps Command (Command Code 41H/C1H)

The Select without ATN Steps Command is used by the initiator to select a target. When this command is issued the device arbitrates for the control of the SCSI bus. When the device wins arbitration, it selects the target device and transfers the Command Descriptor Block (CDB). Before issuing this command the SCSI Timeout Register (STIMREG) 05H, the Control Register One (CNTLREG1) 08H and the SCSI Destination ID Register (SDIDREG) 04H must be set to the proper values. If DMA is enabled, the Start Transfer Count Register (STCREG) 00H–01H must be set to the total length of the command. If DMA is not enabled, the data must be loaded into the FIFO before issuing this command. This command will be terminated early if the SCSI Timeout Register times out or if the target does not go to the Command Phase following the Selection Phase or if the target exits the Command Phase early.

Select with ATN Steps Command (Command Code 42H/C2H)

The Select with ATN Steps Command is used by the initiator to select a target. When this command is issued the device arbitrates for the control of the SCSI bus. When the device wins arbitration, it selects the target device with the $\overline{ATN}$ signal asserted and transfers the Command Descriptor Block (CDB) and a one byte message. Before issuing this command the SCSI Timeout Register (STIMREG) 05H, the Control Register One (CNTLREG1) 08H and the SCSI Destination ID Register (SDIDREG) 04H must be set to the proper values. If DMA is enabled, the Start Transfer Count Register (STCREG) 00H–01H must be set to the total length of the command. If DMA is not enabled, the data must be loaded into the FIFO before issuing this command. This command will be terminated early in the following situations:

- The SCSI Timeout Register times out
- The target does not go to the Message Out Phase following the Selection Phase
- The target exits the Message Phase early
- The target does not go to the Command Phase following the Selection Phase
- The target exits the Command Phase early

Select with ANT and Stop Steps Command (Command Code 43H/C3H)

The Select with ATN and Stop Steps Command is used by the initiator to select a target. When this command is issued the device arbitrates for the control of the SCSI bus. When the device wins arbitration, it selects the target device with the $\overline{ATN}$ signal asserted and transfers the Command Descriptor Block (CDB) and stops after one message byte is sent, but the $\overline{ATN}$ signal is not

deasserted at the end of the command which allows the initiator to send other messages after the ID message is sent out. Before issuing this command the SCSI Timeout Register (STIMREG) 05H, the Control Register One (CNTLREG1) 08H and the SCSI Destination ID Register (SDIDREG) 04H must be set to the proper values. This command will be terminated early if the SCSI Timeout Register times out or if the target does not go to the Message Out Phase following the Selection Phase.

Enable Selection/Reselection Command (Command Code 44H/C4H)

The Enable Selection/Reselection Command is used by the target to respond to a bus-initiated reselection. Upon disconnecting from the bus the Selection/Reselection circuit is automatically disabled by device. This circuit has to be enabled for the device to respond to subsequent reselection attempts and the Enable Selection/ Reselection Command is issued to do that. This command is normally issued within 250 ms (select/reselect timeout) after the device disconnects from the bus. If DMA is enabled the device loads the received data to the buffer memory, but if the DMA is disabled, the received data stays in the FIFO.

Disable Selection/Reselection Command (Command Code 45H/C5H)

The Disable Selection/Reselection Command is used by the target to disable response to a bus-initiated reselection. When this command is issued before a bus initiated selection or reselection is initiated, it resets the internal mode bits previously set by the Enable Selection/Reselection Command. The device also generates a function complete interrupt to the processor. If however, this command is issued after a bus initiated selection/reselection has already begun the command is ignored since the Command Register is held reset and all incoming commands are ignored. The device generates a selected or reselected interrupt when the sequence is complete.

Select with ATN3 Steps Command (Command Code 46H/C6H)

The Select with ATN3 Steps Command is used by the initiator to select a target. This command is similar to the Select with ATN Steps Command, except that it sends exactly three message bytes. When this command is issued the device arbitrates for the control of the SCSI bus. When the device wins arbitration, it selects the target device with the $\overline{ATN}$ signal asserted and transfers the Command Descriptor Block (CDB) and three message bytes. Before issuing this command the SCSI Timeout Register (STIMREG) 05H, the Control Register One (CNTLREG1) 08H and the SCSI Destination ID Register (SDIDREG) 04H must be set to the proper values. If DMA is enabled, the Start Transfer Count Register (STCREG) 00H–01H must be set to the total length of the command. If DMA is not enabled, the data must be loaded into the FIFO before issuing this command. This command will be terminated early in the following situations:

- The SCSI Timeout Register times out
- The target does not go to the Message Out Phase following the Selection Phase
- The target exits the Message Phase early
- The target does not go to the Command Phase following the Selection Phase
- The target exits the Command Phase early

General Commands

No Operation Command (Command Code 00H/80H)

The No Operation Command is used to perform no operation and no interrupt is generated at the end of this command. This command is issued after the Reset Device Command to enable the Command Register. A No Operation Command in the DMA mode may be used to verify the contents of the Start Transfer Count Register (STCREG) 00H – 01H. After the STCREG is loaded with the transfer count and a No Operation Command is issued, reading the Current Transfer Count Register (CTCREG) 00H–01H will give the transfer count value.

Clear FIFO Command (Command Code 01H/81H)

The Clear FIFO Command is used to initialize the FIFO to the empty condition. The Current FIFO Register (CFISREG) 07H reflects the empty FIFO status and the bottom of the FIFO is set to zero. No interrupt is generated at the end of this command.

Reset Device Command (Command Code 02H/82H)

The Reset Device Command immediately stops any device operation and resets all the functions of the device. It returns the device to the disconnected state and it also generates a hard reset. The Reset Device Command remains on the top of the Command Register FIFO holding the device in the reset state until the No Operation Command is loaded. The No Operation command serves to enable the Command Register.

Reset SCSI Bus Command (Command Code 03H/83H)

The Reset SCSI Bus Command is used to assert the $\overline{RSTC}$ signal for approximately 25 ms. This command causes the device to go to the disconnected state. No interrupt is generated upon command completion. A SCSI reset interrupt is however generated upon command completion if the interrupt is not disabled in the Control Register One (CNTLREG1) 08H.

ABSOLUTE MAXIMUM RATINGS

Storage Temperature . . . . . –65°C to +150°C
Ambient Temperature Under Bias . . –55°C to +125°C
Maximum V_{DD} . . . . . –0.5 V to +7.0 V
DC Voltage Applied to Any Pin . –0.5 to (V_{DD} + 0.3) V
Input Static Discharge Protection . . 3000 V pin to pin
(Human body model: 100 pF at 1.5 KΩ)

Stresses above those listed under Absolute Maximum Ratings may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

OPERATING RANGES

Commercial (C) Devices

Ambient Temperature (T_A) . . . . . 0°C to +70°C
Supply Voltage (V_{DD}) . . . . . 3.3 V ±10%

Operating ranges define those limits between which the functionality of the device is guaranteed.

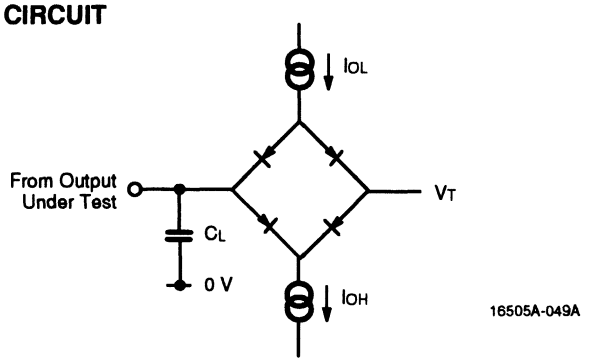
DC OPERATING CHARACTERISTICS V_{DD} = 2.9 V to 3.7 V; T_{CASE} = 0°C to + 100°C

Parameter Symbol	Parameter Description	Pin Names	Test Condition	Min	Max	Unit
I_{CCS}	Static Supply Current		$V_{DD\ MAX}$		2.0	mA
I_{CCD}	Dynamic Supply Current		$V_{DD\ MAX}$		25	mA
I_{LU}	Latch Up Current	All I/O	$V_{LU} \le 10$ V	–100	+100	mA
SCSI Pins						
V_{IH}	Input High Voltage	All SCSI Inputs		2.0	V_{DD} +0.3	V
V_{IL}	Input Low Voltage	All SCSI Inputs		V_{SS}–0.3	0.4	V
V_{IHST}	Input Hysterisis	All SCSI Inputs	2.9 V < V_{DD} < 3.7 V	200		mV
V_{OH}	Output High Voltage	$\overline{SDC}_{7-0}$, $\overline{SDC}_P$, $\overline{REQC}$, $\overline{ACKC}$		2.2		V
V_{OL1}	SCSI Output Low Voltage	$\overline{SD}_{7-0}$ and $\overline{SD}_P$	I_{OL}= 4 mA		0.4	V
V_{OL2}	SCSI Output Low Voltage	$\overline{SDC}_{7-0}$, $\overline{SDC}_P$, $\overline{MSG}$, $\overline{C/D}$, $\overline{I/O}$, $\overline{ATN}$, $\overline{RSTC}$, $\overline{SELC}$, $\overline{BSYC}$, $\overline{ACKC}$, $\overline{REQC}$	I_{OL}= 48 mA		0.5	V
I_{IL}	Input Low Leakage		0.0 V ≤ V_{IN} ≤ 2.7 V	–10	+10	µA
I_{IH}	Input High Leakage		2.7 V ≤ V_{IN} ≤ V_{DD}	–10	+10	µA
I_{OZ}	High Impedance Leakage		0 V ≤ V_{OUT} ≤ V_{DD}	–10	+10	µA
Host Bidirectional Pins						
V_{IH}	Input High Voltage	DMA_{15-0}, $DMAP_{1-0}$ and AD_{7-0}		2.0	V_{DD} +0.3	V
V_{IL}	Input Low Voltage	DMA_{15-0}, $DMAP_{1-0}$ and AD_{7-0}		V_{SS}–0.3	0.4	V
V_{OH}	Output High Voltage	DMA_{15-0}, $DMAP_{1-0}$ and AD_{7-0}		2.2		V
V_{OL}	Output Low Voltage	DMA_{15-0}, $DMAP_{1-0}$ and AD_{7-0}	I_{OL}= 0.5 mA		0.2	V
			I_{OL}= 2 mA		0.45	V
I_{IL}	Input Leakage	DMA_{15-0}, $DMAP_{1-0}$ and AD_{7-0}	0.0 V ≤ V_{IN} ≤ V_{DD}	–15	+15	µA
I_{OZ}	High Impedance Leakage	DMA_{15-0}, $DMAP_{1-0}$ and AD_{7-0}	0 V ≤ V_{OUT} ≤ V_{DD}	–15	+15	µA

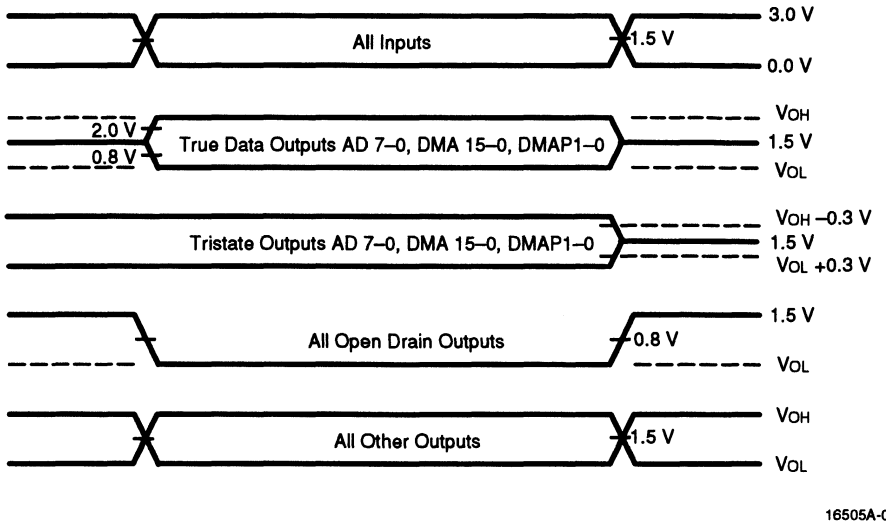
DC OPERATING CHARACTERISTICS (continued)

Parameter Symbol	Parameter Description	Pin Names	Test Condition	Min	Max	Unit
Host Output Pins						
V_{OH}	Output High Voltage	DREQ and $\overline{INT}$		2.2		V
V_{OL}	Output Low Voltage	DREQ and $\overline{INT}$	I_{OL}= 0.5 mA		0.2	V
			I_{OL}= 2.5 mA		0.45	V
I_{OZ}	High Impedance Leakage	DREQ and $\overline{INT}$	$0 \le V_{OUT} \le V_{DD}$	–10	+10	µA
Host Input Pins						
V_{IH}	Input High Voltage	ALE[A3], $\overline{DMARD}$[A2], BHE[A1], AS0[A0] $\overline{DACK}$, $\overline{DMAWR}$, $\overline{RD}$, $\overline{WR}$, $\overline{CS}$, $BUSMD_{1-0}$, CLK, RESET		2.0	V_{DD} + 0.3	V
V_{IL}	Input Low Voltage	ALE[A3], $\overline{DMARD}$[A2], BHE[A1], AS0[A0] $\overline{DACK}$, $\overline{DMAWR}$, $\overline{RD}$, $\overline{WR}$, $\overline{CS}$, $BUSMD_{1-0}$, CLK, RESET		V_{SS} – 0.3	0.4	V

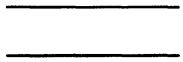
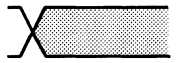
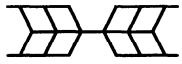
SWITCHING TEST CIRCUIT



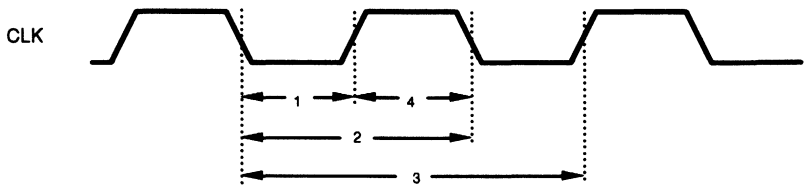
SWITCHING TEST WAVEFORMS



KEY TO SWITCHING WAVEFORMS

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
	Don't Care, Any Change Permitted	Changing, State Unknown
	Does Not Apply	Center Line is High-Impedance "Off" State

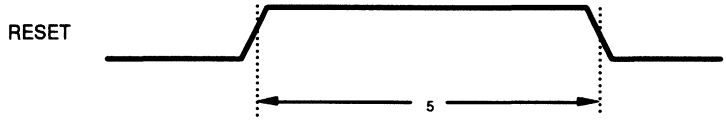
KS000010



Clock Input 16505A-030A

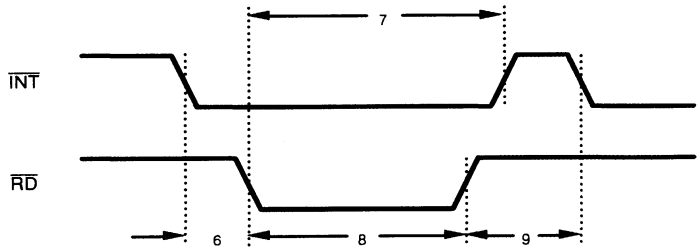
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
1	t_{PWL}	Clock Pulse Width Low		14.58		ns
2	t_{CP}	Clock period		40	100	ns
3	t_L	Synchronization latency (parameter 2 + parameter 1)		54.58	185.42	ns
4	t_{PWH}	Clock Pulse Width High		14.58		ns

Note:
Clock Frequency Range = 10 MHz to 25 MHz for Asynchronous SCSI Bus
= 12 MHz to 25 MHz for Synchronous SCSI Bus



Reset Input 16505A-031A

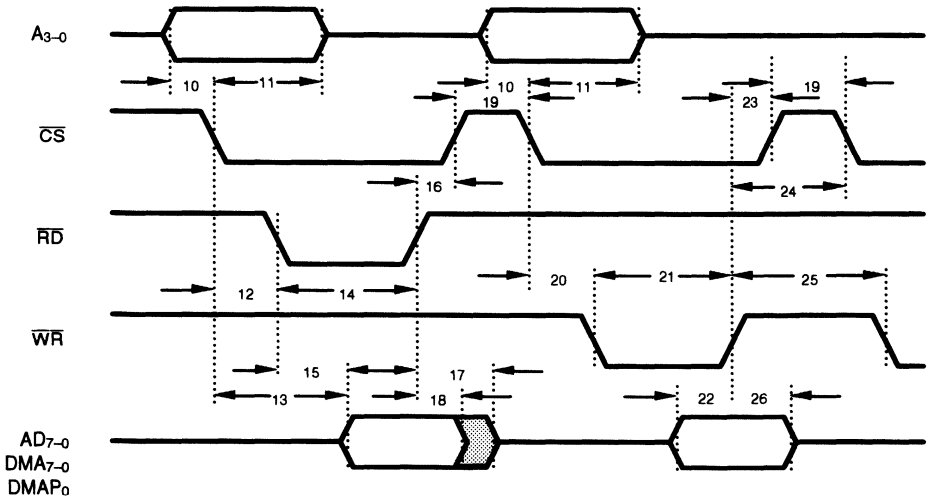
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
5	t_{PWH}	Reset Pulse Width High		500		ns



16505A-032A

Interrupt Output

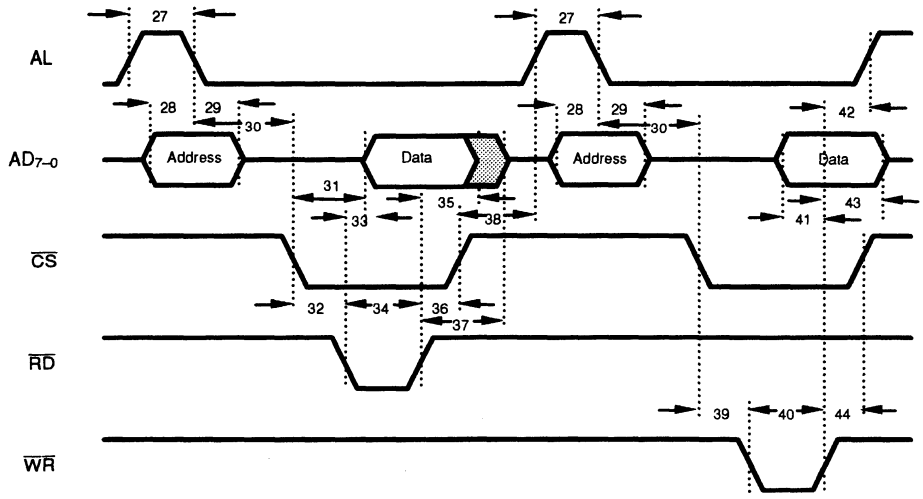
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
6	t_S	$\overline{INT}$ ↓ to $\overline{RD}$ ↓ Set Up Time		0		ns
7	t_{PD}	$\overline{RD}$ ↓ to $\overline{INT}$ ↑ Delay		0	100	ns
8	t_{PWL}	$\overline{RD}$ Pulse Width Low		50		ns
9	t_{PD}	$\overline{RD}$ ↑ to $\overline{INT}$ ↓ Delay				ns



Register Read/Write with Non-Multiplexed Address Data Bus

16505A-033A

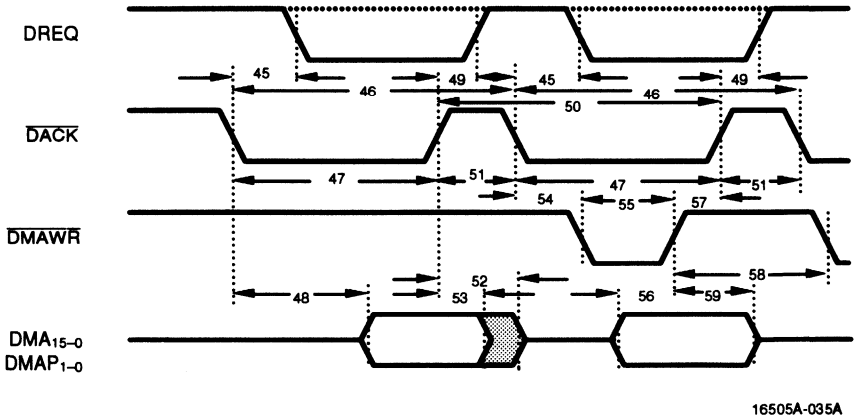
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
10	t_S	Address to $\overline{CS}$ ↓ Set Up Time		0		ns
11	t_H	Address to $\overline{CS}$ ↓ Hold Time		50		ns
12	t_S	$\overline{CS}$ ↓ to $\overline{RD}$ ↓ Set Up Time		11		ns
13	t_{PD}	$\overline{CS}$ ↓ to Data Valid Delay			90	ns
14	t_{PWL}	$\overline{RD}$ Pulse Width Low		50		ns
15	t_{PD}	$\overline{RD}$ ↓ to Data Valid Delay			50	ns
16	t_H	$\overline{RD}$ ↑ to $\overline{CS}$ ↑ Hold Time		0		ns
17	t_Z	$\overline{RD}$ ↑ to Data High Impedance			50	ns
18	t_H	$\overline{RD}$ ↑ to Data Hold Time		2		ns
19	t_{PWH}	$\overline{CS}$ Pulse Width High		40		ns
20	t_S	$\overline{CS}$ ↓ to $\overline{WR}$ ↓ Set Up Time		0		ns
21	t_{PWL}	$\overline{WR}$ Pulse Width Low		40		ns
22	t_S	Data to $\overline{WR}$ ↑ Set Up Time		18		ns
23	t_H	$\overline{WR}$ ↑ to $\overline{CS}$ ↑ Hold Time		0		ns
24	t_S	$\overline{WR}$ ↑ to $\overline{CS}$ ↓ Set Up Time		60		ns
25	t_{PWH}	$\overline{WR}$ Pulse Width High		60		ns
26	t_H	Data to $\overline{WR}$ ↑ Hold Time		0		ns



16506B-055A

Register Read/Write with Multiplexed Address Data Bus

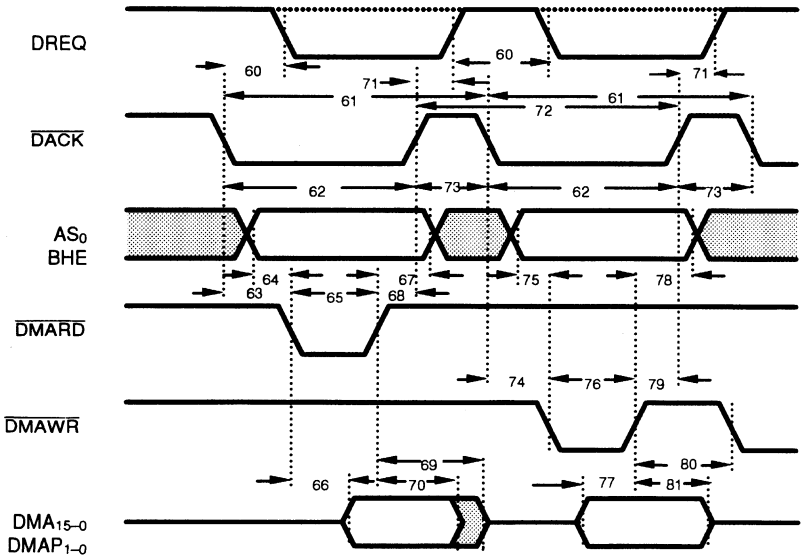
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
27	t_{PWH}	ALE Pulse Width High		20		ns
28	t_S	Address to ALE ↓ Set Up Time		10		ns
29	t_H	Address to ALE ↓ Hold Time		10		ns
30	t_S	ALE ↓ to $\overline{CS}$ ↓ Set Up Time		10		ns
31	t_{PD}	$\overline{CS}$ ↓ to Data Valid Delay			90	ns
32	t_S	$\overline{CS}$ ↓ to $\overline{RD}$ ↓ Set Up Time		0		ns
33	t_{PD}	$\overline{RD}$ ↓ to Data Valid Delay			50	ns
34	t_{PWL}	$\overline{RD}$ Pulse Width Low		50		ns
35	t_H	$\overline{RD}$ ↑ to Data Hold Time		2		ns
36	t_H	$\overline{RD}$ ↑ to $\overline{CS}$ ↑ Hold Time		0		ns
37	t_Z	$\overline{RD}$ ↑ to Data High Impedance			40	ns
38	t_S	$\overline{CS}$ ↑ to ALE ↑ Set Up Time		50		ns
39	t_S	$\overline{CS}$ ↓ to $\overline{WR}$ ↓ Set Up Time		0		ns
40	t_{PWL}	$\overline{WR}$ Pulse Width Low		40		ns
41	t_S	Data to $\overline{WR}$ ↑ Set Up Time		15		ns
42	t_S	$\overline{WR}$ ↑ to ALE ↑ Set Up Time		50		ns
43	t_H	Data to $\overline{WR}$ ↑ Hold Time		0		ns
44	t_H	$\overline{WR}$ ↑ to $\overline{CS}$ ↑ Hold Time		0		ns



16505A-035A

DMA Read/Write without Byte Control

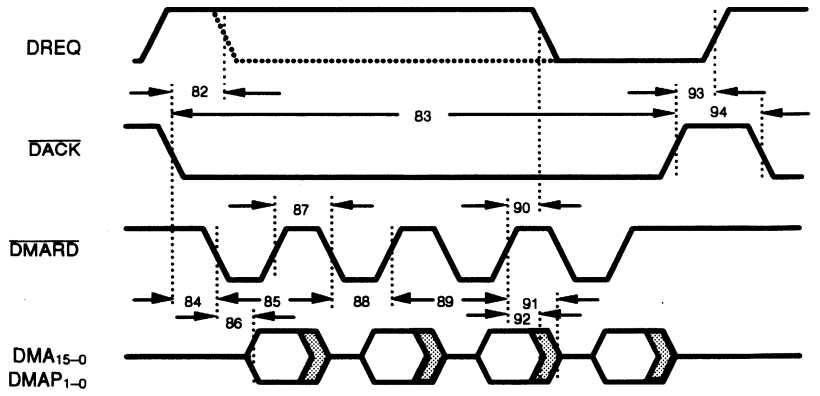
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
45	t_{PD}	$\overline{DACK}$ ↓ to DREQ ↓ Valid Delay			38	ns
46	t_P	$\overline{DACK}$ ↓ to $\overline{DACK}$ ↓ period		100		ns
47	t_{PWL}	$\overline{DACK}$ Pulse Width Low		60		ns
48	t_{PD}	$\overline{DACK}$ ↓ to Data Valid Delay			41	ns
49	t_{PD}	$\overline{DACK}$ ↑ to DREQ ↑ Valid Delay			40	ns
50	t_P	$\overline{DACK}$ ↑ to $\overline{DACK}$ ↑ period		t_3+50-t_{51}		ns
51	t_{PWH}	$\overline{DACK}$ Pulse Width High		12		ns
52	t_Z	$\overline{DACK}$ ↑ to Data High Impedance			50	ns
53	t_H	$\overline{DACK}$ ↑ to Data Hold Time		2		ns
54	t_S	$\overline{DACK}$ ↓ to $\overline{DMAWR}$ ↓ Set Up Time		0		ns
55	t_{PWL}	$\overline{DMAWR}$ Pulse Width Low		50		ns
56	t_S	Data to $\overline{DMAWR}$ ↑ Set Up Time		18		ns
57	t_H	$\overline{DMAWR}$ ↑ to $\overline{DACK}$ ↑ Hold Time		0		ns
58	t_{PWH}	$\overline{DMAWR}$ Pulse Width High		40		ns
59	t_H	Data to $\overline{DMAWR}$ ↑ Hold Time		0		ns



DMA Read/Write with Byte Control

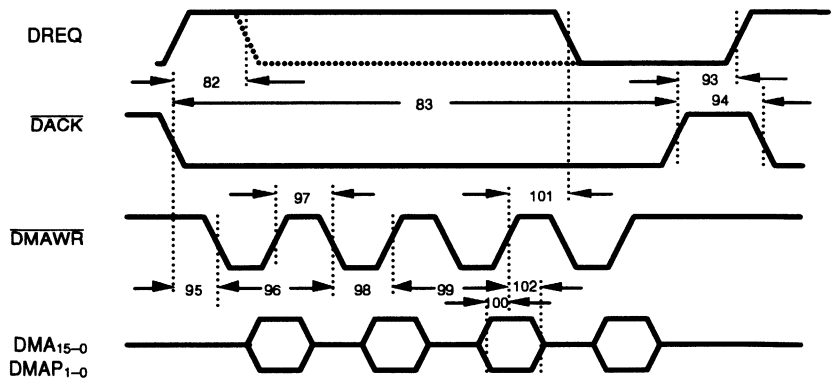
16505A-036A

No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
60	t_{PD}	$\overline{DACK}$ ↓ to DREQ ↓ Valid Delay			38	ns
61	t_P	$\overline{DACK}$ ↓ to $\overline{DACK}$ ↓ period		100		ns
62	t_{PWL}	$\overline{DACK}$ Pulse Width Low		60		ns
63	t_S	$\overline{DACK}$ ↓ to $\overline{DMARD}$ ↓ Set Up Time		0		ns
64	t_S	BHE, AS0 to $\overline{DMARD}$ ↓ Set Up Time		20		ns
65	t_{PWL}	$\overline{DMARD}$ Pulse Width Low		60		ns
66	t_{PD}	$\overline{DMARD}$ ↓ to Data Valid Delay			51	ns
67	t_H	BHE, AS0 to $\overline{DMARD}$ ↑ Hold Time		20		ns
68	t_H	$\overline{DMARD}$ ↑ to $\overline{DACK}$ ↑ Hold Time		0		ns
69	t_Z	$\overline{DMARD}$ ↑ to Data High Impedance			40	ns
70	t_H	$\overline{DMARD}$ ↑ to Data Hold Time		2		ns
71	t_{PD}	$\overline{DACK}$ ↑ to DREQ ↑ Valid Delay			40	ns
72	t_P	$\overline{DACK}$ ↑ to $\overline{DACK}$ ↑ period		100		ns
73	t_{PWH}	$\overline{DACK}$ Pulse Width High		12		ns
74	t_S	$\overline{DACK}$ ↓ to $\overline{DMAWR}$ ↓ Set Up Time		0		ns
75	t_S	BHE, AS0 to $\overline{DMAWR}$ ↓ Set Up Time		20		ns
76	t_{PWL}	$\overline{DMAWR}$ Pulse Width Low		50		ns
77	t_S	Data to $\overline{DMAWR}$ ↑ Set Up Time		18		ns
78	t_H	BHE, AS0 to $\overline{DMAWR}$ ↑ Hold Time		20		ns
79	t_H	$\overline{DMAWR}$ ↑ to $\overline{DACK}$ ↑ Hold Time		0		ns
80	t_{PWH}	$\overline{DMAWR}$ Pulse Width High		40		ns
81	t_H	Data to $\overline{DMAWR}$ ↑ Hold Time		0		ns



16506B-056A

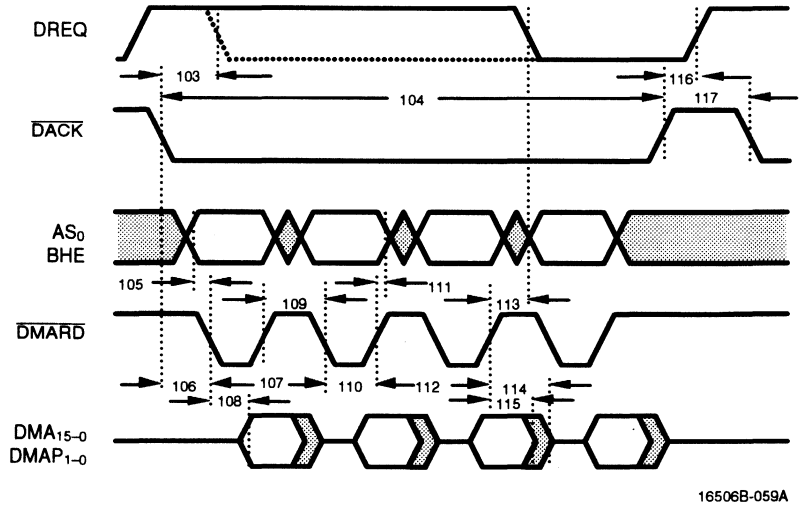
Burst DMA Read without Byte Control



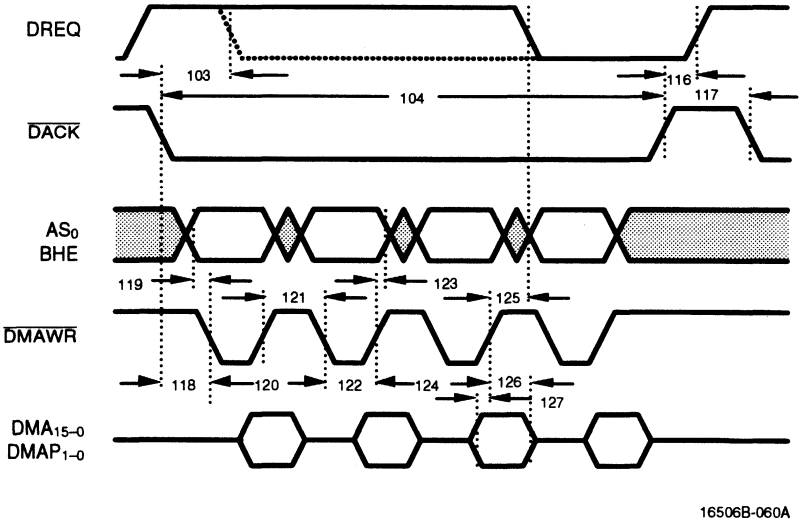
16506B-057A

Burst DMA Write without Byte Control

No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
82	t_{PD}	$\overline{DACK}$↓ to DREQ↓ Valid Delay			45	ns
83	t_{PWL}	$\overline{DACK}$ Pulse Width Low		100		ns
84	t_S	$\overline{DACK}$ ↓ to $\overline{DMARD}$↓ Set Up Time		0		ns
85	t_P	$\overline{DMARD}$↓ to $\overline{DMARD}$↓ period		130		ns
86	t_{PD}	$\overline{DMARD}$↓ to Data Valid Delay			70	ns
87	t_{PWH}	$\overline{DMARD}$ Pulse Width High		60		ns
88	t_{PWL}	$\overline{DMARD}$ Pulse Width Low		70		ns
89	t_P	$\overline{DMARD}$↑ to $\overline{DMARD}$ ↑ period		$t_3 + 50$		ns
90	t_{PD}	$\overline{DMARD}$↑ to DREQ↓ Valid Delay			140	ns
91	t_Z	$\overline{DMARD}$↑ to Data High Impedance			50	ns
92	t_H	$\overline{DMARD}$↑ to Data Hold Time		2		ns
93	t_{PD}	$\overline{DACK}$↑ to DREQ ↑ Valid Delay			40	ns
94	t_{PWH}	$\overline{DACK}$ Pulse Width High		60		ns
95	t_S	$\overline{DACK}$↓ to $\overline{DMAWR}$↓ Set Up Time		0		ns
96	t_P	$\overline{DMAWR}$↓ to $\overline{DMAWR}$↓ period		160		ns
97	t_{PWH}	$\overline{DMAWR}$ Pulse Width High		60		ns
98	t_{PWL}	$\overline{DMAWR}$ Pulse Width Low		100		ns
99	t_P	$\overline{DMAWR}$ ↑ to $\overline{DMAWR}$ ↑ period		$t_3 + 50$		ns
100	t_S	Data to $\overline{DMAWR}$↑ Set Up Time		18		ns
101	t_{PD}	$\overline{DMAWR}$↑ to DREQ↓ Valid Delay			140	ns
102	t_H	Data to $\overline{DMAWR}$ ↑ Hold Time		0		ns

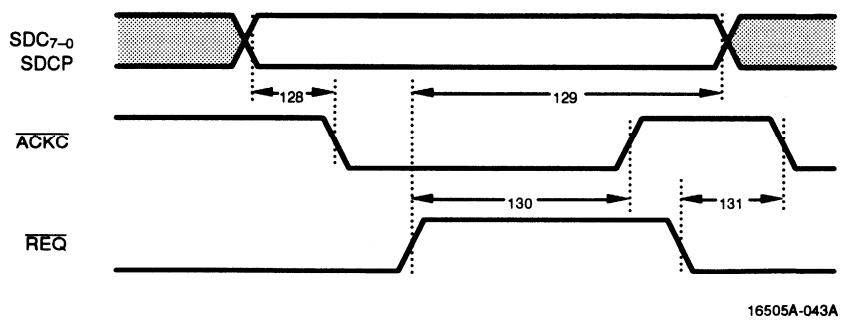


Burst DMA Read with Byte Control



Burst DMA Write with Byte Control

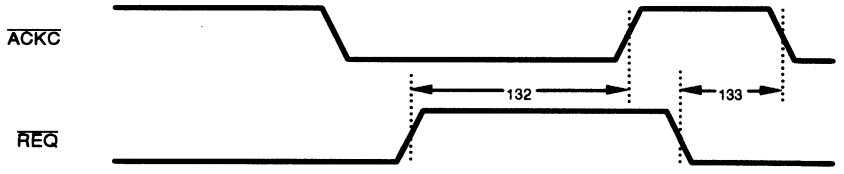
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
103	t_{PD}	$\overline{DACK}$ ↓ to DREQ ↓ Valid Delay			45	ns
104	t_{PWL}	$\overline{DACK}$ Pulse Width Low		100		ns
105	t_S	BHE, AS0 to $\overline{DMARD}$ ↓ Set Up Time		20		ns
106	t_S	$\overline{DACK}$ ↓ to $\overline{DMARD}$ ↓ Set Up Time		0		ns
107	t_P	$\overline{DMARD}$ ↓ to $\overline{DMARD}$ ↓ period		130		ns
108	t_{PD}	$\overline{DMARD}$ ↓ to Data Valid Delay			70	ns
109	t_{PWH}	$\overline{DMARD}$ Pulse Width High		60		ns
110	t_{PWL}	$\overline{DMARD}$ Pulse Width Low		70		ns
111	t_H	BHE, AS0 to $\overline{DMARD}$ ↑ Hold Time		20		ns
112	t_P	$\overline{DMARD}$ ↑ to $\overline{DMARD}$ ↑ period		$t_3 + 50$		ns
113	t_{PD}	$\overline{DMARD}$ ↑ to DREQ ↓ Valid Delay			140	ns
114	t_Z	$\overline{DMARD}$ ↑ to Data High Impedance			50	ns
115	t_H	$\overline{DMARD}$ ↑ to Data Hold Time		2		ns
116	t_{PD}	$\overline{DACK}$ ↑ to DREQ ↑ Valid Delay			50	ns
117	t_{PWH}	$\overline{DACK}$ Pulse Width High		60		ns
118	t_S	$\overline{DACK}$ ↓ to $\overline{DMAWR}$ ↓ Set Up Time		0		ns
119	t_S	BHE, AS0 to $\overline{DMAWR}$ ↓ Set Up Time		20		ns
120	t_P	$\overline{DMAWR}$ ↓ to $\overline{DMAWR}$ ↓ period		160		ns
121	t_{PWH}	$\overline{DMAWR}$ Pulse Width High		60		ns
122	t_{PWL}	$\overline{DMAWR}$ Pulse Width Low		100		ns
123	t_H	BHE, AS0 to $\overline{DMAWR}$ ↑ Hold Time		20		ns
124	t_P	$\overline{DMAWR}$ ↑ to $\overline{DMAWR}$ ↑ period		$t_3 + 50$		ns
125	t_{PD}	$\overline{DMAWR}$ ↑ to DREQ ↓ Valid Delay			140	ns
126	t_H	Data to $\overline{DMAWR}$ ↑ Hold Time		0		ns
127	t_S	Data to $\overline{DMAWR}$ ↑ Set Up Time		18		ns



16505A-043A

Asynchronous Initiator Send

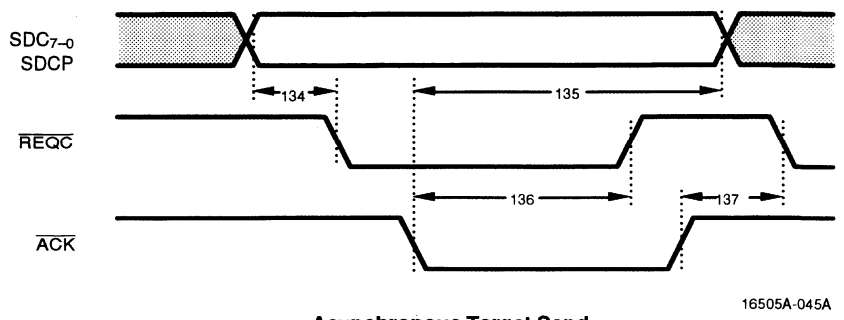
No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
128	tS	Data to ACKC ↓ Set Up Time		55		ns
129	tPD	REQ ↓ to Data Delay			80	ns
130	tPD	REQ ↓ to ACKC ↓ Delay			46	ns
131	tPD	REQ ↑ to ACKC ↑ Delay			72	ns



16505A-044A

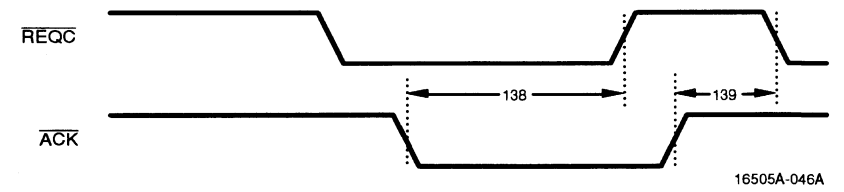
Asynchronous Initiator Receive

No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
132	tPD	REQ ↓ to ACKC ↓ Delay			43	ns
133	tPD	REQ ↑ to ACKC ↑ Delay			72	ns



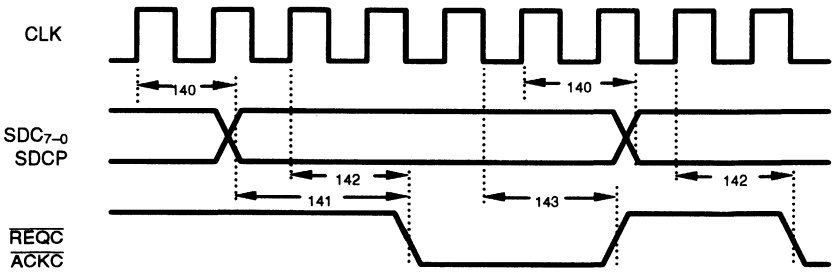
Asynchronous Target Send

No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
134	tS	Data to REQC ↓ Set Up Time		55		ns
135	tPD	ACK ↓ to Data Delay			88	ns
136	tPD	ACK ↓ to REQC ↓ Delay			75	ns
137	tPD	ACK ↓ to REQC ↓ Delay			45	ns



Asynchronous Target Receive

No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
138	tPD	ACK ↓ to REQC ↓ Delay			75	ns
139	tPD	ACK ↓ to REQC ↓ Delay			45	ns



16505A-047A

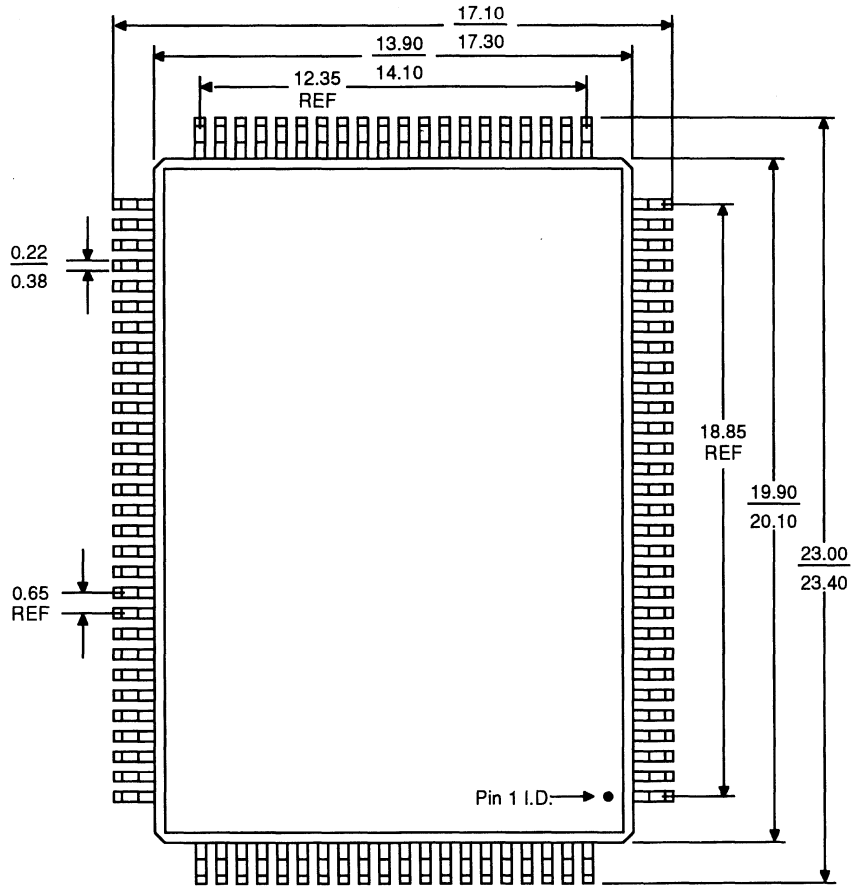
Synchronous Initiator Target Transmit

No.	Parameter Symbol	Parameter Description	Test Conditions	Min	Max	Unit
140	t_{PD}	CLK↓ to Data Delay		15*	90	ns
141	t_S	$\overline{ACKC}$ or $\overline{REQC}$↑ to Data Set Up Time		55		ns
142	t_{PD}	CLK↓ to $\overline{ACKC}$ or $\overline{REQC}$ ↑ Delay		13*	68	ns
143	t_{PD}	CLK↑ to $\overline{ACKC}$ or $\overline{REQC}$↓ Delay		17	70	ns

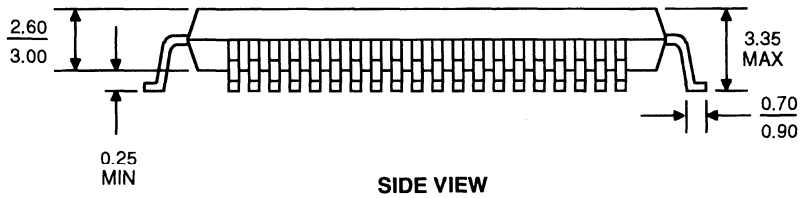
* *The minimum values have a wide range since they depend on the Synchronization latency. The synchronization latency, in turn, depends on the operating frequency of the device.*

PHYSICAL DIMENSIONS*

PQR100
Plastic Quad Flat Pack Trimmed and Formed (measured in millimeters)



TOP VIEW



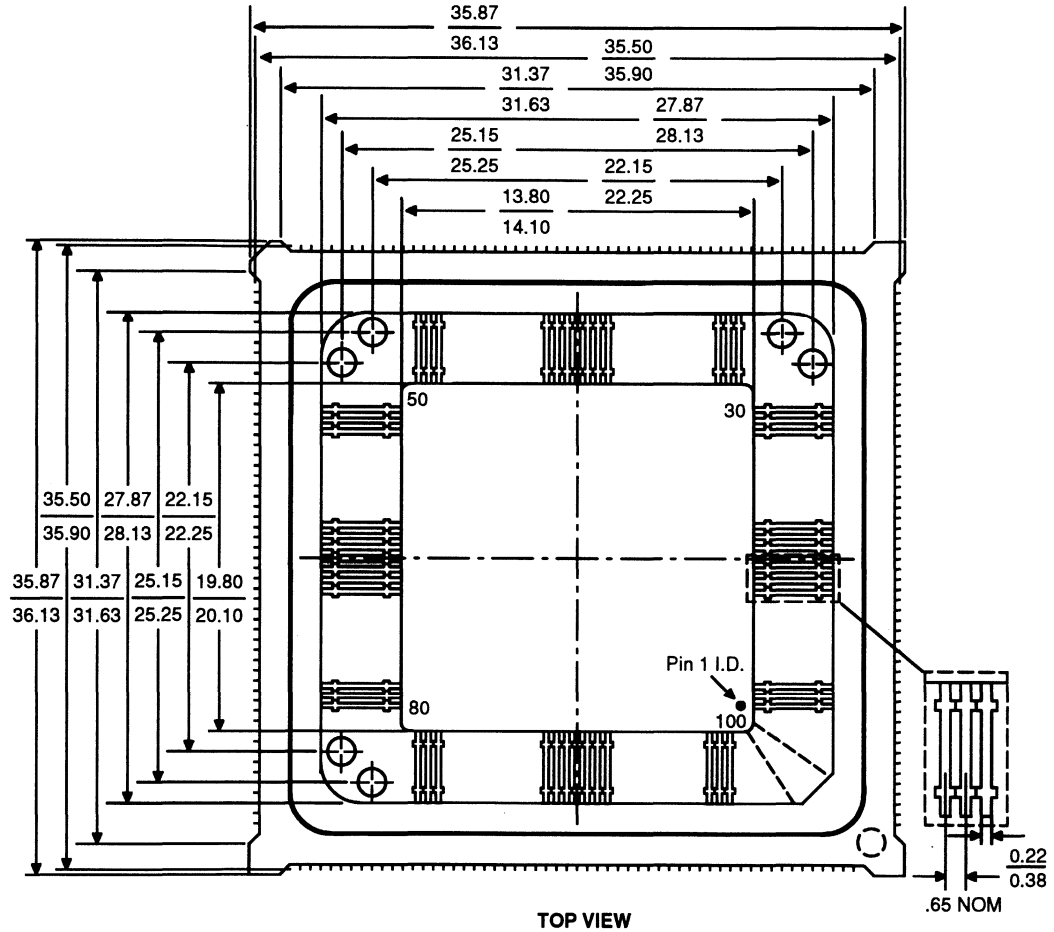
SIDE VIEW

15590D
BX 45
9/6/91 SG

** For reference only. BSC is an ANSI standard for Basic Space Centering. Not drawn to scale.*

PHYSICAL DIMENSIONS*

PQR100
Molded Carrier Ring Quad Flat Pack (measured in millimeters)



CB 48
6/25/92 SG

Chapter 3

EPROM Products

CHAPTER 3
EPROM Products

PRELIMINARY

Am27LV010/Am27LV010B

1 Megabit (131,072 x 8-Bit) CMOS EPROM

Advanced Micro Devices

DISTINCTIVE CHARACTERISTICS

- **Single +3.3 V power supply**
  - — Regulated power supply 3.0 V – 3.6 V
  - — Unregulated power supply 2.7 V – 3.6 V (for battery operated systems)
- **Low power consumption:**
  - — 10 µA typical CMOS standby current
  - — 90 µW maximium standby power
  - — 54 mW maximum power at 5 MHz
- **Fast access time—120 ns**
- **JEDEC-approved pinout**
  - — Pin compatible with 5.0 V 1 Mbit EPROM
  - — Easy upgrade from 28-pin EPROMs
- **Fast Flashrite™ programming**
  - — Typical programming time of 16 seconds
- **Latch-up protected to 100 mA from –1 V to Vcc + 1 V**
- **High noise immunity**
- **Compact 32-pin DIP package requires no hardware change for upgrades to 8 Mbit**
- **Versatile features for simple interfacing**
  - — Both CMOS and TTL input/output compatibility
  - — Two line control functions

GENERAL DESCRIPTION

The Am27LV010 is a low voltage, low power megabit, ultraviolet erasable, progammable read-only memory, organized as 128K words by 8 bits per word.

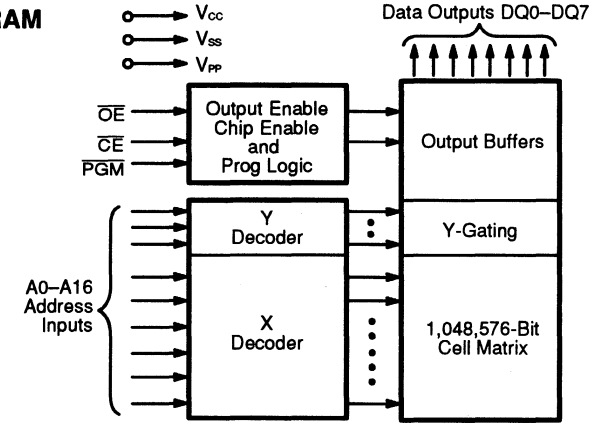
The Am27LV010 operates from a single power supply of 3.3 V and is offered with two power supply tolerances. The Am27LV010 has a V_{CC} tolerance range of 3.3 V ± 10% making it suitable for use in systems that have regulated power supplies. The Am27LV010B has a voltage supply range of 2.7 V – 3.6 V making it an ideal part for battery operated systems.

Maximum power consumption of the Am27LV010 in standby mode is only 90 µW. If the device is constantly accessed at 5 MHz, then the maximum power consumption increases to 54 mW. These power ratings are significantly lower than typical EPROMs. Also, as power consumption is proportional to voltage squared, 3.3 V devices consume at least 57% less power than their 5.0 V counterparts. Due to its lower current and voltage, the Am27LV010 is well-suited for battery operated and portable systems as it extends the battery life in these systems. Typical applications are notebook and handheld computers as well as cellular phones.

The Am27LV010 is packaged in the industry standard 32-pin windowed ceramic DIP and LCC packages, as well as one-time programmable (OTP) packages. This device is pin-compatible with the 5.0 V devices.

The Am27LV010 uses AMD's Flashrite™ programming algorithm (100 µs pulses) resulting in typical programming time of 16 seconds. This device is manufactured on AMD's sub-micron process technology which provides high speed, low power and high noise immunity.

BLOCK DIAGRAM



17341A-1

Publication# 17341 Rev. A Amendment /0
Issue Date: February 1993

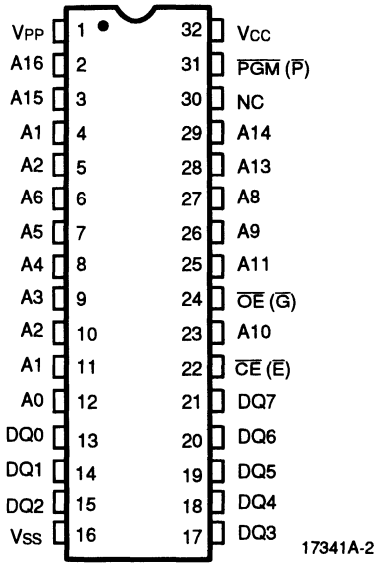
PRODUCT SELECTOR GUIDE

Family Part No.	Am27LV010				
Ordering Part No:					
Am27LV010 (3.0 V – 3.6 V)	**-120**	**-150**	**-200**	**-250**	**-300**
Am27LV010 B (2.7 V – 3.6 V)			**-200**	**-250**	**-300**
Max Access Time (ns)	120	150	200	250	300
$\overline{CE}$ ($\overline{E}$) Access (ns)	120	150	200	250	300
$\overline{OE}$ ($\overline{G}$) Access (ns)	50	65	75	100	120

CONNECTION DIAGRAMS

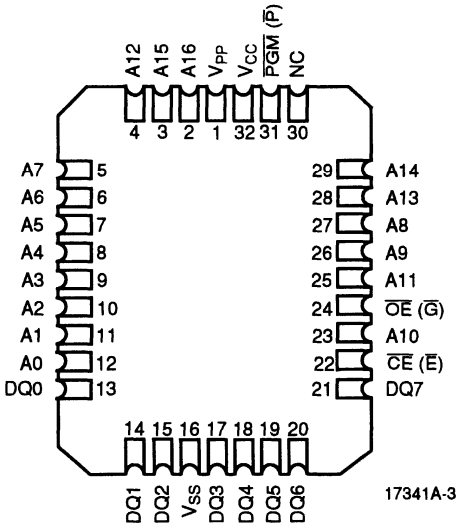
Top View

DIP



17341A-2

PLCC/LCC



17341A-3

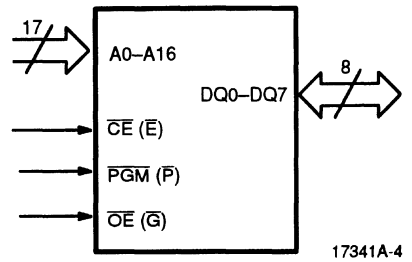
Notes:

1. *JEDEC nomenclature is in parenthesis.*
2. *The 32-pin DIP to 32-pin LCC configuration varies from the JEDEC 28-pin DIP to 32-pin LCC configuration.*

PIN DESCRIPTION

A0–A16 = Address Inputs
$\overline{CE}$ ($\overline{E}$) = Chip Enable Input
DQ0–DQ7 = Data Input/Outputs
V_{SS} = Ground
NC = No Internal Connect
$\overline{OE}$ ($\overline{G}$) = Output Enable Input
$\overline{PGM}$ ($\overline{P}$) = Program Enable Input
V_{CC} = V_{CC} Supply Voltage
V_{PP} = Program Supply Voltage

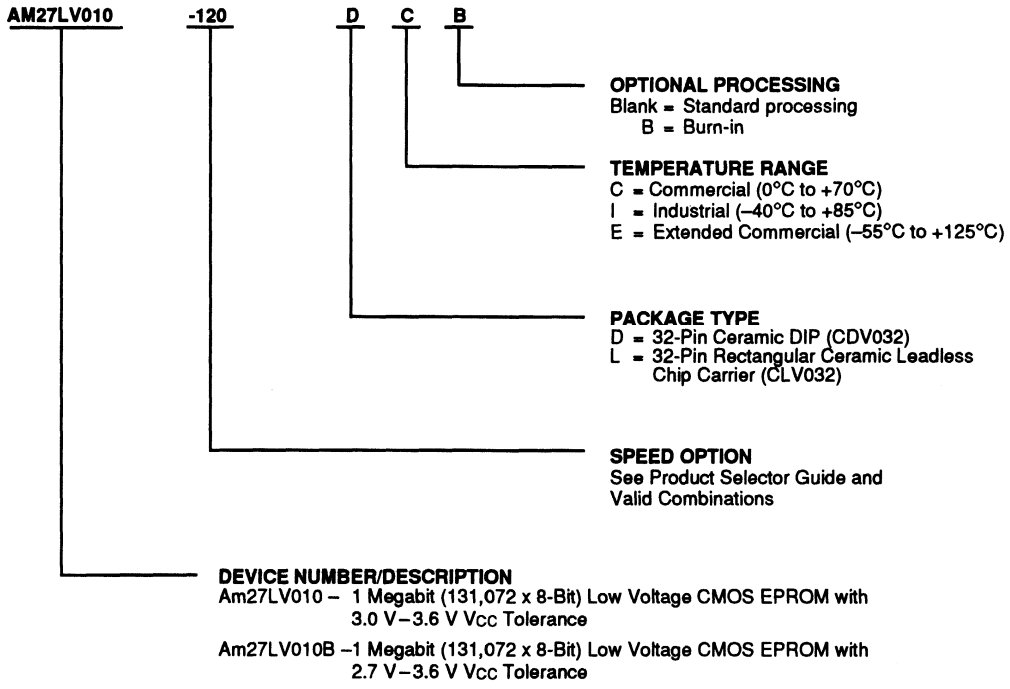
LOGIC SYMBOL



17341A-4

ORDERING INFORMATION
EPROM Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:



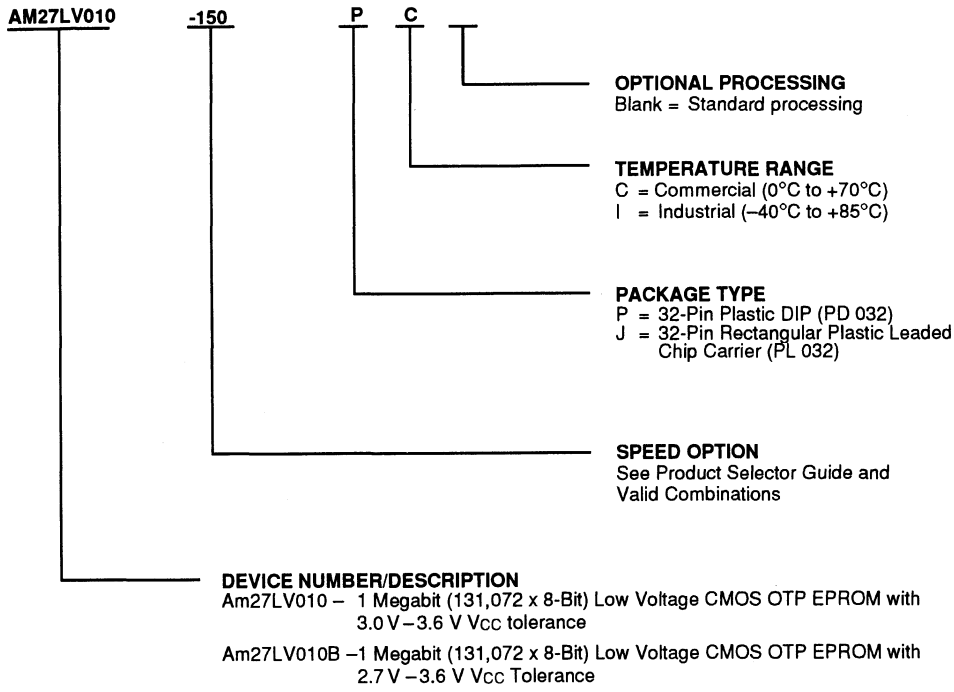
Valid Combinations	
AM27LV010-120	DC, DCB, LC, LCB
AM27LV010-150	DC, DCB, DE, DEB, DI, DIB, LC, LI, LE, LEB
AM27LV010-200	
AM27LV010-250	
AM27LV010-300	
AM27LV010B-150	
AM27LV010B-200	
AM27LV010B-250	
AM27LV010B-300	

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check on newly released combinations.

ORDERING INFORMATION
OTP Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:



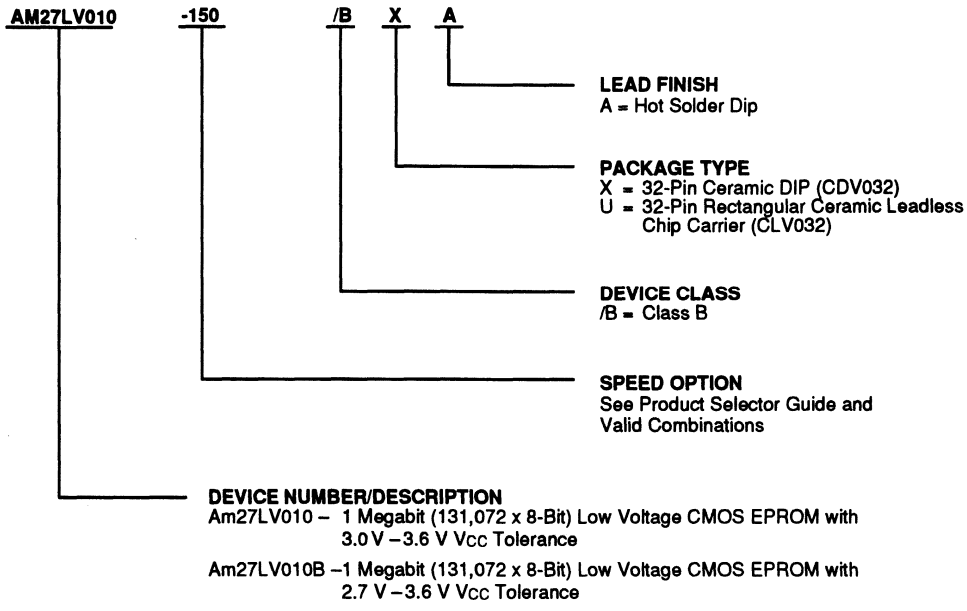
Valid Combinations	
AM27LV010-150	PC, JC, PI, JI
AM27LV010-200	
AM27LV010-250	
AM27LV010-300	
AM27LV010B-200	
AM27LV010B-250	
AM27LV010B-300	

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check on newly released combinations.

MILITARY ORDERING INFORMATION
APL Products

AMD products for Aerospace and Defense applications are available in several packages and operating ranges. APL (Approved Products List) products are fully compliant with MIL-STD-883 requirements. The order number (Valid Combination) is formed by a combination of:



Valid Combinations	
AM27LV010-150	/BXA, /BUA
AM27LV010-200	
AM27LV010-250	
AM27LV010-300	
AM27LV010B-250	
AM27LV010B-300	

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, or to check on newly released combinations.

Group A Tests

Group A tests consist of Subgroups 1, 2, 3, 7, 8, 9, 10, 11.

FUNCTIONAL DESCRIPTION

Erasing the Am27LV010

In order to clear all locations of their programmed contents, it is necessary to expose the Am27LV010 to an ultraviolet light source. A dosage of 15 W seconds/cm^2 is required to completely erase an Am27LV010. This dosage can be obtained by exposure to an ultraviolet lamp —wavelength of 2537 Angstroms (Å) —with intensity of 12,000 µW/cm^2 for 15 to 20 minutes. The Am27LV010 should be directly under and 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 Am27LV010, and similar devices, will erase with light sources having wavelengths shorter than 4000 Å. Although erasure times will be much longer than with UV sources at 2537Å, nevertheless the exposure to fluorescent light and sunlight will eventually erase the Am27LV010 and exposure to them should be prevented to realize maximum system reliability. If used in such an environment, the package window should be covered by an opaque label or substance.

Programming the Am27LV010

Upon delivery, or after each erasure, the Am27LV010 has all 1,048,576 bits in the "ONE", or HIGH state. "ZEROs" are loaded into the Am27LV010 through the procedure of programming.

The programming mode is entered when 12.75 V ± 0.25 V is applied to the V_{PP} pin, $\overline{CE}$ and $\overline{PGM}$ are at V_{IL} and $\overline{OE}$ is at V_{IH}.

For programming, the data to be programmed is applied 8 bits in parallel to the data output pins.

The Flashrite programming algorithm reduces programming time by using initial 100 µs pulses followed by a byte verification to determine whether the byte has been successfully programmed. If the data does not verify, an additional pulse is applied for a maximum of 25 pulses. This process is repeated while sequencing through each address of the EPROM.

The Flashrite programming algorithm programs and verifies at V_{CC} = 6.25 V and V_{PP} = 12.75 V. After the final address is completed, all bytes are compared to the original data with V_{CC} = V_{PP} = 5.25 V. Am27LV010 can be programmed using the same algorithm as the 5 V counterpart 27C010.

Program Inhibit

Programming of multiple Am27LV010s in parallel with different data is also easily accomplished. Except for $\overline{CE}$, all like inputs of the parallel Am27LV010 may be common. A TTL low-level program pulse applied to an Am27LV010 $\overline{CE}$ input with V_{PP} = 12.75 ± 0.25 V, $\overline{PGM}$ LOW, and $\overline{OE}$ HIGH will program that Am27LV010. A high-level $\overline{CE}$ input inhibits the other Am27LV010s from being programmed.

Program Verify

A verify should be performed on the programmed bits to determine that they were correctly programmed. The verify should be performed with $\overline{OE}$ and $\overline{CE}$ at V_{IL}, $\overline{PGM}$ at V_{IH}, and V_{PP} between 12.5 V and 13.0 V.

Auto Select Mode

The auto select mode allows the reading out of a binary code from an EPROM that will identify its manufacturer and type. This mode is intended for use by programming equipment for the purpose of automatically matching the device to be programmed with its corresponding programming algorithm. This mode is functional in the 25°C ± 5°C ambient temperature range that is required when programming the Am27LV010.

To activate this mode, the programming equipment must force 12.0 ± 0.5 V on address line A9 of the Am27LV010. Two identifier bytes may then be sequenced from the device outputs by toggling address line A0 from V_{IL} to V_{IH}. All other address lines must be held at V_{IL} during auto select mode.

Byte 0 (A0 = V_{IL}) represents the manufacturer code, and Byte 1 (A0 = V_{IH}), the device identifier code. For the Am27LV010, these two identifier bytes are given in the Mode Select table. All identifiers for manufacturer and device codes will possess odd parity, with the MSB (DQ7) defined as the parity bit.

Read Mode

The Am27LV010 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 (t_{ACC}) is equal to the delay from $\overline{CE}$ to output (t_{CE}). Data is available at the outputs t_{OE} after the falling edge of $\overline{OE}$, assuming that $\overline{CE}$ has been LOW and addresses have been stable for at least $t_{ACC} - t_{OE}$.

Standby Mode

The Am27LV010 has a CMOS standby mode which reduces the maximum V_{CC} current to 25 µA. It is placed in CMOS-standby when $\overline{CE}$ is at V_{CC} ± 0.3 V. The Am27LV010 also has a TTL-standby mode which reduces the maximum V_{CC} current to 0.6 mA. It is placed in TTL-standby when $\overline{CE}$ is at V_{IH}. When in standby mode, the outputs are in a high-impedance state, independent of the $\overline{OE}$ input.

Mixed Power Supply System

Am27LV020 (in 3.0 V to 3.6 V regulated power supply) can be intefaced with 5 V system only when the I/O pins (DQ0–DQ7) are not driven by the 5 V system. V_{IHmax} = V_{CCLV} + 2.2 V for address and clock pins and V_{IHmax} = V_{CCLV} + 0.5 V for I/O pins should be followed to avoid CMOS latch-up condition

Output OR-Tieing

To accommodate multiple memory connections, a two-line control function is provided to allow for:

- Low memory power dissipation
- Assurance that output bus contention will not occur

It is recommended that $\overline{CE}$ be decoded and used as the primary device-sele cting 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.

System Applications

During the switch between active and standby conditions, transient current peaks are produced on the rising and falling edges of Chip Enable. The magnitude of these transient current peaks is dependent on the output capacitance loading of the device. At a minimum, a 0.1 µF ceramic capacitor (high frequency, low inherent inductance) should be used on each device between V_{CC} and V_{SS} to minimize transient effects. In addition, to overcome the voltage drop caused by the inductive effects of the printed circuit board traces on EPROM arrays, a 4.7 µF bulk electrolytic capacitor should be used between V_{CC} and V_{SS} for each eight devices. The location of the capacitor should be close to where the power supply is connected to the array.

Mode Select Table

Mode \ Pins		$\overline{CE}$	$\overline{OE}$	$\overline{PGM}$	A0	A9	V_{PP}	Outputs
Read		V_{IL}	V_{IL}	X	X	X	X	D_{OUT}
Output Disable		V_{IL}	V_{IH}	X	X	X	X	High Z
Standby (TTL)		V_{IH}	X	X	X	X	X	High Z
Standby (CMOS)		V_{CC} ± 0.3 V	X	X	X	X	X	High Z
Program		V_{IL}	V_{IH}	V_{IL}	X	X	V_{PP}	D_{IN}
Program Verify		V_{IL}	V_{IL}	V_{IH}	X	X	V_{PP}	D_{OUT}
Program Inhibit		V_{IH}	X	X	X	X	V_{PP}	High Z
Auto Select (Note 3)	Manufacturer Code	V_{IL}	V_{IL}	X	V_{IL}	V_H	X	01H
	Device Code	V_{IL}	V_{IL}	X	V_{IH}	V_H	X	0EH

Notes:

1. *X can be either V_{IL} or V_{IH}*
2. *V_H = 12.0 V ± 0.5 V*
3. *A1–A8 = A10–A16 = V_{IL}*
4. *See DC Programming Characteristics for V_{PP} voltage during programming.*

ABSOLUTE MAXIMUM RATINGS

Storage Temperature:
- OTP Products ..... −65°C to +125°C
- All Other Products ..... −65°C to +150°C

Ambient Temperature with Power Applied ..... −55°C to +125°C

Voltage with Respect to V_{SS}:
- All pins except A9, V_{PP}, and V_{CC} (Note 1) ..... −0.6 V to V_{CC} +0.6 V
- A9 and V_{PP} (Note 2) ..... −0.6 V to 13.5 V
- V_{CC} ..... −0.6 V to 7.0 V

Notes:

1. *During transitions, the input may overshoot V_{SS} to −2.0 V for periods of up to 20 ns. Maximum DC voltage on input and I/O may overshoot to V_{CC} + 2.0 V for periods up to 20 ns.*
2. *During transitions, A9 and V_{PP} may overshoot V_{SS} to −2.0 V for periods of up to 20 ns. A9 and V_{PP} must not exceed 13.5 V for any period of time.*

Stresses above those listed under Absolute Maximum Ratings may cause permanent device failure. This is a stress rating only; functional operation of the devices at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

OPERATING RANGES

Commercial (C) Devices
- Case Temperature (T_C) ..... 0°C to +70°C

Industrial (I) Devices
- Case Temperature (T_C) ..... −40°C to +85°C

Extended Commercial (E) Devices
- Case Temperature (T_C) ..... −55°C to +125°C

Military (M) Devices
- Case Temperature (T_C) ..... −55°C to +125°C

Supply Read Voltages:
- V_{CC} for Am27LV010 ..... +3.0 V to +3.6 V
- V_{CC} for Am27LV010B ..... +2.7 V to +3.6 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over operating ranges unless otherwise specified (Notes 1, 4, 5 and 7) (for APL products, Group A, Subgroups 1, 2, 3, 7 and 8 are tested unless otherwise noted)

Parameter Symbol	Parameter Description	Test Conditions		Min	Max	Unit
TTL and CMOS Inputs for V_{CC} = 3.0 V to 3.6 V						
V_{OH}	Output HIGH Voltage	I_{OH} = –2.0 mA		2.4		V
V_{OL}	Output LOW Voltage	I_{OL} = 2.0 mA			0.4	V
V_{IH}	Input HIGH Voltage			2.0	V_{CC} + 0.3	V
V_{IL}	Input LOW Voltage			–0.3	+0.8	V
I_{LI}	Input Load Current	V_{IN} = 0 V to V_{CC}	C/I Devices		1.0	µA
			E/M Devices		1.0	
I_{LO}	Output Leakage Current	V_{OUT} = 0 V to V_{CC}	C/I Devices		5	µA
			E/M Devices		5	
I_{CC1}	V_{CC} Active Current (Note 5)	$\overline{CE}$ = V_{IL}, f = 5 MHz, I_{OUT} = 0 mA (Open Outputs)	C/I Devices		15	mA
			E/M Devices		20	
I_{CC2}	V_{CC} TTL Standby Current	$\overline{CE}$ = V_{IH}	TTL		0.6	mA
I_{CC3}	V_{CC} CMOS Standby Current	$\overline{CE}$ = V_{CC} ± 0.3 V	CMOS		25	µA
I_{PP1}	V_{PP} Current During Read	$\overline{CE}$ = $\overline{OE}$ = V_{IL}, V_{PP} = V_{CC}			100	µA

Parameter Symbol	Parameter Description	Test Conditions		Min	Max	Unit
CMOS Inputs for V_{CC} = 2.7 V to 3.6 V						
V_{OH}	Output HIGH Voltage	I_{OH} = –20 µA		V_{CC} – 0.1		V
V_{OL}	Output LOW Voltage	I_{OL} = 20 µA			0.1	V
V_{IH}	Input HIGH Voltage			0.7 V_{CC}	V_{CC} + 0.3	V
V_{IL}	Input LOW Voltage			–0.3	0.2 V_{CC}	V
I_{LI}	Input Load Current	V_{IN} = 0 V to +V_{CC}	C/I Devices		1.0	µA
			E/M Devices		1.0	
I_{LO}	Output Leakage Current	V_{OUT} = 0 V to +V_{CC}	C/I Devices		5	µA
			E/M Devices		5	
I_{CC1}	V_{CC} Active Current (Note 5 and 8)	$\overline{CE}$ = V_{IL}, f = 5 MHz, I_{OUT} = 0 mA (Open Outputs)	C/I Devices		15	mA
			E/M Devices		20	
I_{CC3}	V_{CC} CMOS Standby Current	$\overline{CE}$ = V_{CC} + 0.3 V			25	µA
I_{PP1}	V_{PP} Current During Read	$\overline{CE}$ = $\overline{OE}$ = V_{IL}, V_{PP} = V_{CC}			100	µA

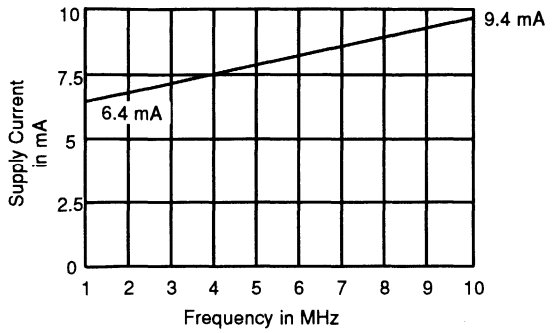


Figure 1. Typical Supply Current vs. Frequency
V_{CC} = 3.6 V, T = 25°C

17341A-5

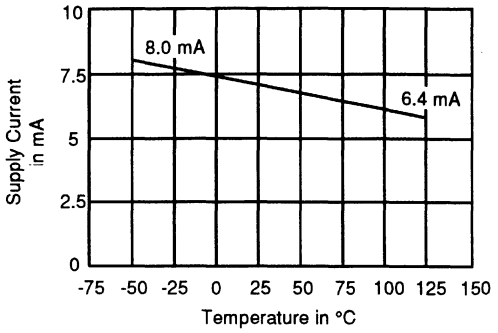


Figure 2. Typical Supply Current vs. Temperature
V_{CC} = 3.6 V, f = 5 MHz

17341A-6

CAPACITANCE (Notes 2, 3, and 7)

Parameter Symbol	Parameter Description	Test Conditions	CDV032		CLV032		Unit
			Typ	Max	Typ	Max	
C_{IN}	Input Capacitance	V_{IN} = 0 V	10	12	8	10	pF
C_{OUT}	Output Capacitance	V_{OUT} = 0 V	12	15	9	12	pF

Notes:

1. *V_{CC} must be applied simultaneously or before V_{PP}, and removed simultaneously or after V_{PP}.*
2. *Typical values are for nominal supply voltages.*
3. *This parameter is only sampled and not 100% tested.*
4. ***Caution**: The Am27LV010 must not be removed from, or inserted into, a socket or board when V_{CC} or V_{PP} is applied.*
5. *I_{CC1} is tested with $\overline{OE}$ = V_{IH} to simulate open outputs.*
6. *T_A = +25°C, f = 1 MHz.*
7. *During transitions, the inputs may overshoot to –2.0 V for periods less than 20 ns. Maximum DC voltage on output pins may overshoot to V_{CC} + 2.0 V for periods less than 20 ns.*
8. *For typical supply current valves at various frequencies, refer to Figure 1. For temperature, refer to Figure 2.*

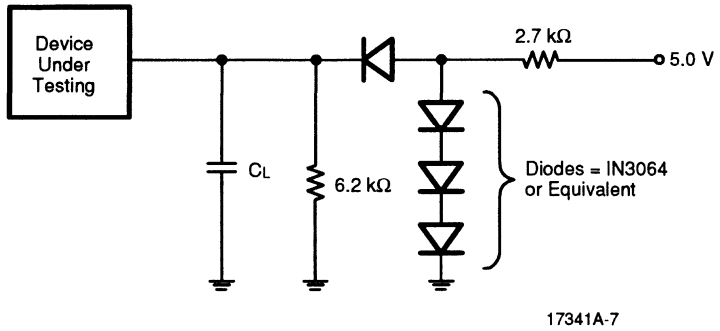
SWITCHING CHARACTERISTICS over operating ranges unless otherwise specified (for APL products, Group A, Subgroups 9, 10 (Notes 1, 3, and 4) and 11 are tested unless otherwise noted)

JEDEC	Standard	Parameter Description	Test Conditions		Am27LV010 -120	-150	-200	-250	-300	Unit
t_{AVQV}	t_{ACC}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$	Min						ns
				Max	120	150	200	250	300	
t_{ELQV}	t_{CE}	Chip Enable to Output Delay	$\overline{OE} = V_{IL}$	Min						ns
				Max	120	150	200	250	300	
t_{GLQV}	t_{OE}	Output Enable to Output Delay	$\overline{CE} = V_{IL}$	Min	–	–	–	–	–	ns
				Max	50	65	75	100	120	
t_{EHQZ}, t_{GHQZ}	t_{DF}	Chip Enable HIGH or Output Enable HIGH, whichever comes first, to Output Float (Note 2)		Min	0	0	0	0	0	ns
				Max	40	50	60	60	60	
t_{AXQX}	t_{OH}	Output Hold from Addresses, $\overline{CE}$, or $\overline{OE}$, whichever occurred first		Min	0	0	0	0	0	ns
				Max	–	–	–	–	–	

Notes:

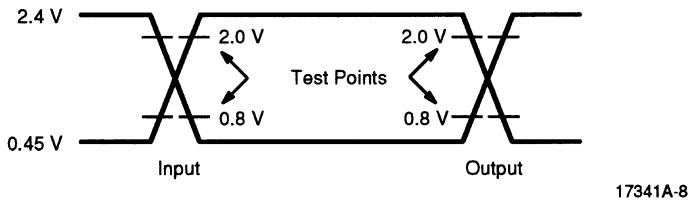
1. *V_{CC} must be applied simultaneously or before V_{PP}, and removed simultaneously or after V_{PP}.*
2. *This parameter is only sampled, not 100% tested.*
3. ***Caution**: The Am27LV010 must not be removed from, or inserted into, a socket when V_{PP} or V_{CC} is applied.*
4. *Output Load: 1 TTL gate and C_L = 100 pF*
   Input Rise and Fall Times: 20 ns
   Input Pulse Levels: 0.45 V to 2.4 V
   Timing Measurement Reference Level— Inputs: 0.8 V and 2.0 V
   Outputs: 0.8 V and 2.0 V.

SWITCHING TEST CIRCUIT



C_L = 100 pF including jig capacitance

SWITCHING TEST WAVEFORM



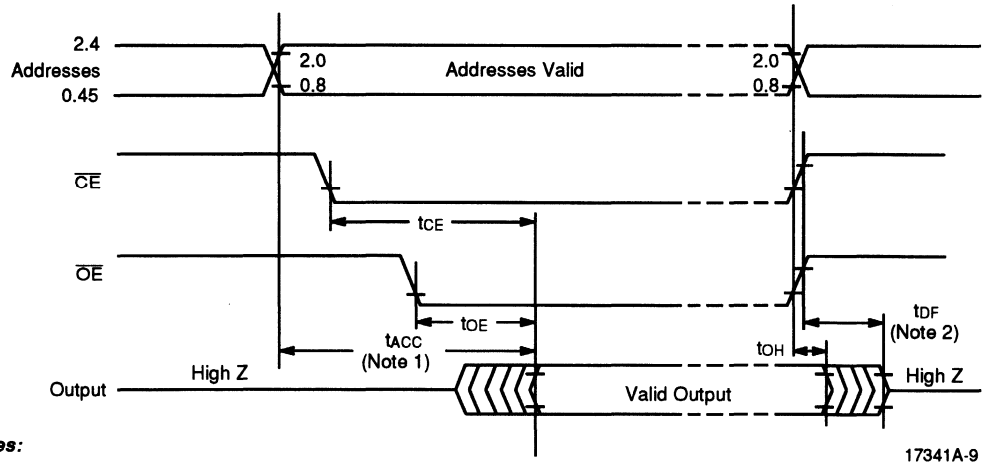
AC Testing: Inputs are driven at 2.4 V for a Logic "1" and 0.45 V for a Logic "0". Input pulse rise and fall times are ≤ 20 ns.

KEY TO SWITCHING WAVEFORMS

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
	Don't Care, Any Change Permitted	Changing, State Unknown
	Does Not Apply	Center Line is High Impedance "Off" State

KS000010

SWITCHING WAVEFORM



17341A-9

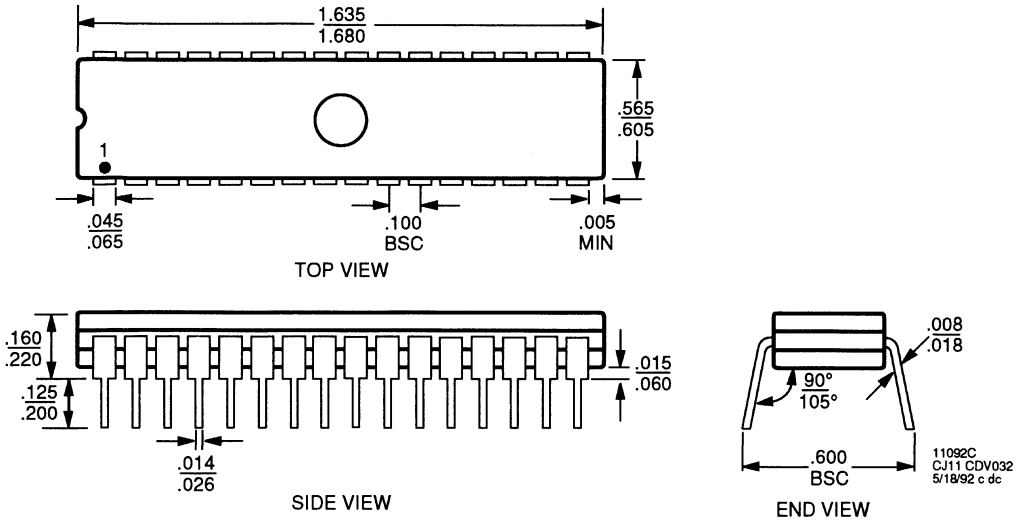
Notes:

1. $\overline{OE}$ *may be delayed up to* $t_{ACC} - t_{OE}$ *after the falling edge of addresses without impact on* t_{ACC}.
2. t_{DF} *is specified from* $\overline{OE}$ *or* $\overline{CE}$, *whichever occurs first.*

PHYSICAL DIMENSIONS*

CDV032

32-Pin Ceramic DIP (measured in inches)

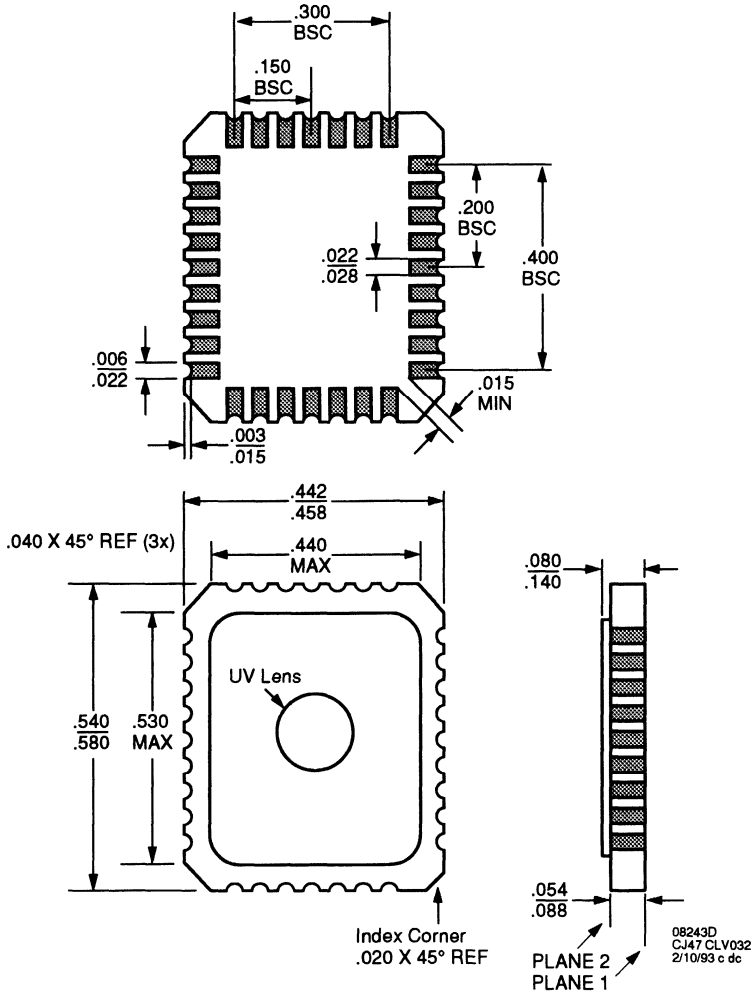


For reference only. BSC is an ANSI standard for Basic Space Centering.

PHYSICAL DIMENSIONS*

CLV032

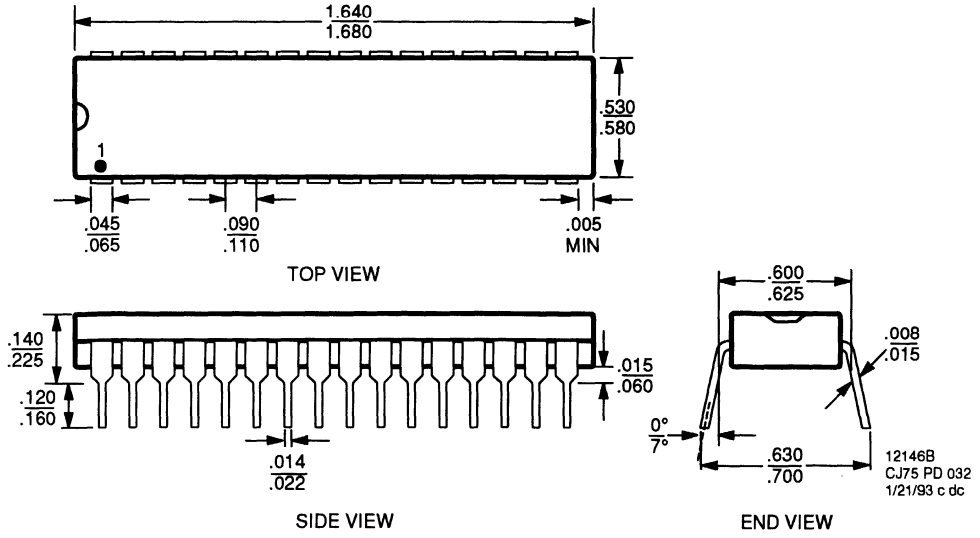
32-Pin Rectangular Ceramic Leadless Chip Carrier (measured in inches)



PHYSICAL DIMENSIONS*

PD 032

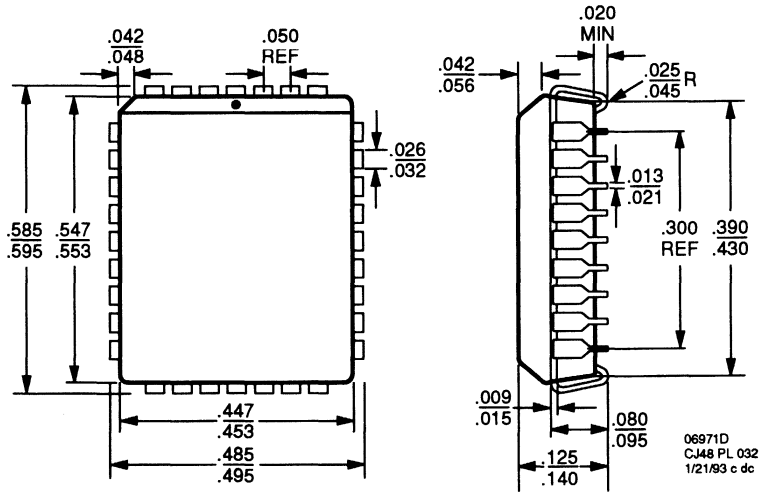
32-Pin Plastic DIP (measured in inches)



PHYSICAL DIMENSIONS*

PL 032

32-Pin Rectangular Plastic Leaded Chip Carrier (measured in inches)



PRELIMINARY

Am27LV020/Am27LV020B

2 Megabit (262,144 x 8-Bit) Low Voltage CMOS EPROM

Advanced Micro Devices

DISTINCTIVE CHARACTERISTICS

- **Single 3.3 V power supply**
  - — Regulated power supply 3.0 V–3.6 V
  - — Unregulated power supply 2.7 V–3.6 V (battery-operated systems)
- **Low power consumption:**
  - — 10 µA typical CMOS standby current
  - — 90 µW maximum standby power
  - — 54 mW power at 5 MHz maximum
- **Fast access time**
  - — 150 ns
- **JEDEC-approved pinout**
  - — Pin compatible with 5.0 V 2-Mbit EPROM
  - — Easy upgrade from 28-pin JEDEC EPROMs
- **100% Flashrite™ programming**
  - — Typical programming time of 32 seconds
- **Latch-up protected to 100 mA from –1 V to Vcc + 1 V**
- **High noise immunity**
- **Compact 32-pin DIP package requires no hardware change for upgrades to 8 Mbit**
- **Versatile features for simple interfacing**
  - — Both CMOS and TTL input/output compatibility
  - — Two line control functions

GENERAL DESCRIPTION

The Am27LV020 is a low voltage, low power megabit, ultraviolet erasable, progammable read-only memory organized as 256K words by 8 bits per word.

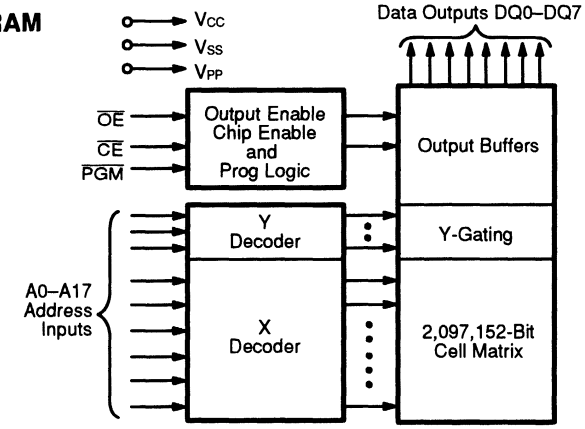
The Am27LV020 operates from a single power supply of 3.3 V and is offered with two power supply tolerances. The Am27LV020 has a V_{CC} tolerance range of 3.3 V ± 10% making it suitable for use in systems that have regulated power supplies. The Am27LV020B has a voltage supply range of 2.7 V–3.6 V making it an ideal part for battery operated systems.

Maximum power consumption of the Am27LV020 in standby mode is only 90 µW. If the device is constantly accessed at 5 MHz, then the maximum power consumption increases to 54 mW. These power ratings are significantly lower than typical EPROMs. Also, as power consumption is proportional to voltage squared, 3.3 V devices consume at least 57% less power than their 5.0 V counterparts. Due to its lower current and voltage, the Am27LV020 is well-suited for battery operated and portable systems as it extends the battery life in these systems. Typical applications are notebook and handheld computers as well as cellular phones.

The Am27LV020 is packaged in the industry standard 32-pin windowed ceramic DIP and LCC packages, as well as one-time programmable (OTP) packages. This device is pin-compatible with the 5.0 V devices.

The Am27LV020 uses AMD's Flashrite™ programming algorithm (100 µs pulses) resulting in typical programming times of 32 seconds. This device is manufactured on AMD's sub-micron process technology which provides high speed, low power and high noise immunity.

BLOCK DIAGRAM



17342A-1

Publication# 17342 Rev. A Amendment /0
Issue Date: February 1993

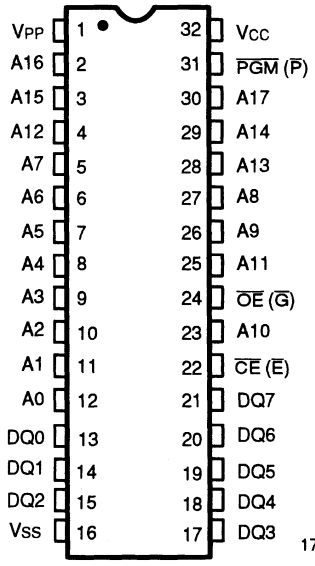
PRODUCT SELECTOR GUIDE

Family Part No.	Am27LV020			
Ordering Part No:				
Am27LV020 (3.0 V–3.6 V)	**-150**	**-200**	**-250**	**-300**
Am27LV020B (2.7 V–3.6 V)		**-200**	**-250**	**-300**
Max Access Time (ns)	150	200	250	300
$\overline{CE}$ ($\overline{E}$) Access (ns)	150	200	250	300
$\overline{OE}$ ($\overline{G}$) Access (ns)	65	75	100	120

CONNECTION DIAGRAMS

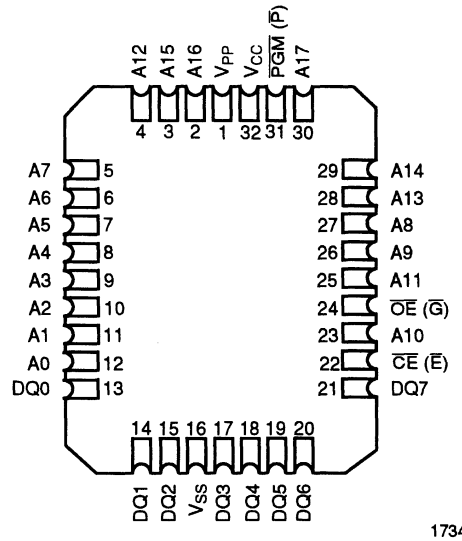
Top View

PDIP



17342A-2

PLCC/LCC



17342A-3

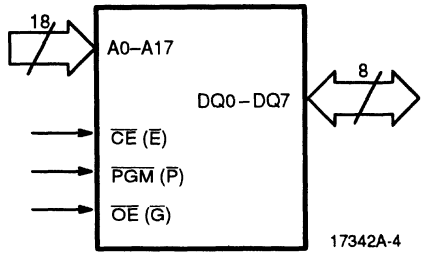
Notes:

1. *JEDEC nomenclature is in parenthesis.*
2. *The 32-pin DIP to 32-pin LCC configuration varies from the JEDEC 28-pin DIP to 32-pin LCC configuration.*

PIN DESCRIPTION

A0–A17	=	Address Inputs
$\overline{CE}$ ($\overline{E}$)	=	Chip Enable Input
VSS	=	Ground
DQ0–DQ7	=	Data Input/Outputs
$\overline{OE}$ ($\overline{G}$)	=	Output Enable Input
$\overline{PGM}$ ($\overline{P}$)	=	Program Enable Input
VCC	=	VCC Supply Voltage
VPP	=	Program Supply Voltage

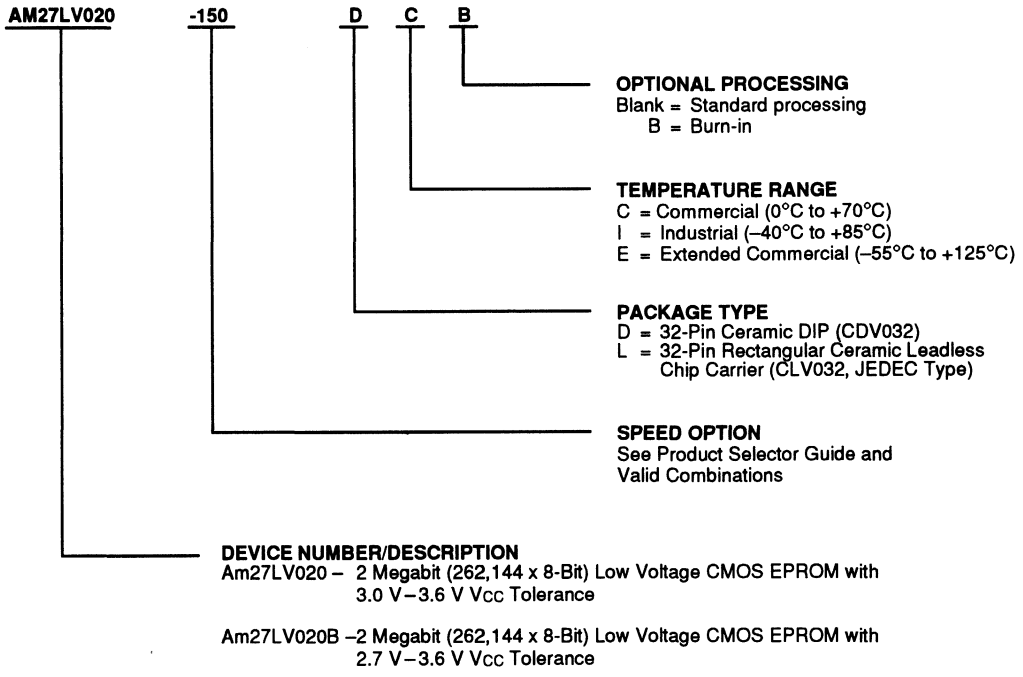
LOGIC SYMBOL



17342A-4

ORDERING INFORMATION
EPROM Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:



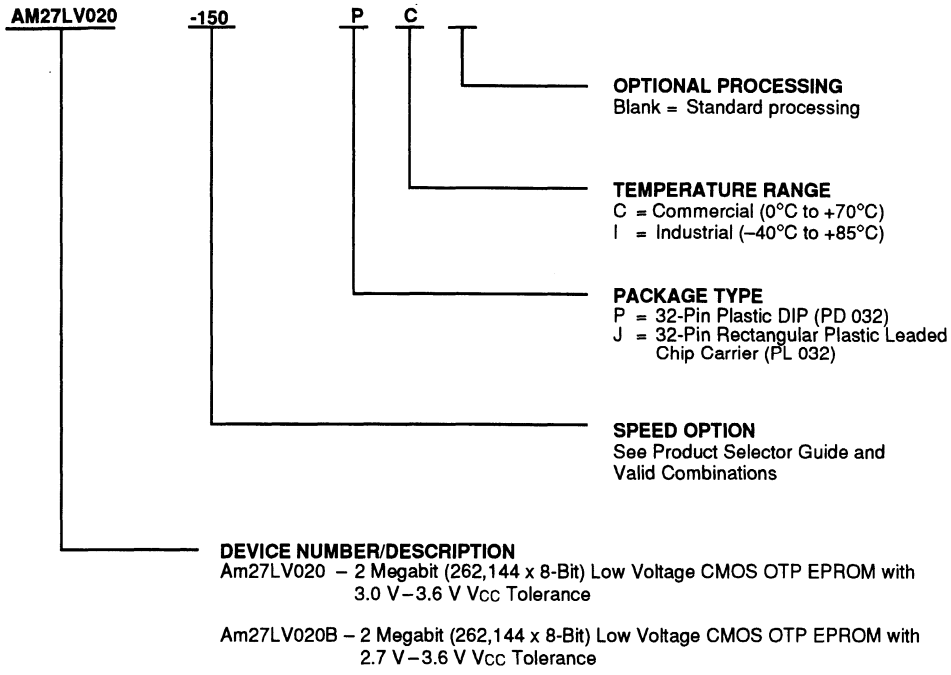
Valid Combinations	
AM27LV020-150	DC, DCB, DI, DIB, LC, LCB, LI, LIB
AM27LV020-200	DC, DCB, DE, DEB, DI, DIB, LC, LCB, LI, LIB, LE, LEB
AM27LV020-250	
AM27LV020-300	
AM27LV020B-200	
AM27LV020B-250	
AM27LV020B-300	

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check on newly released combinations.

ORDERING INFORMATION
OTP Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of:



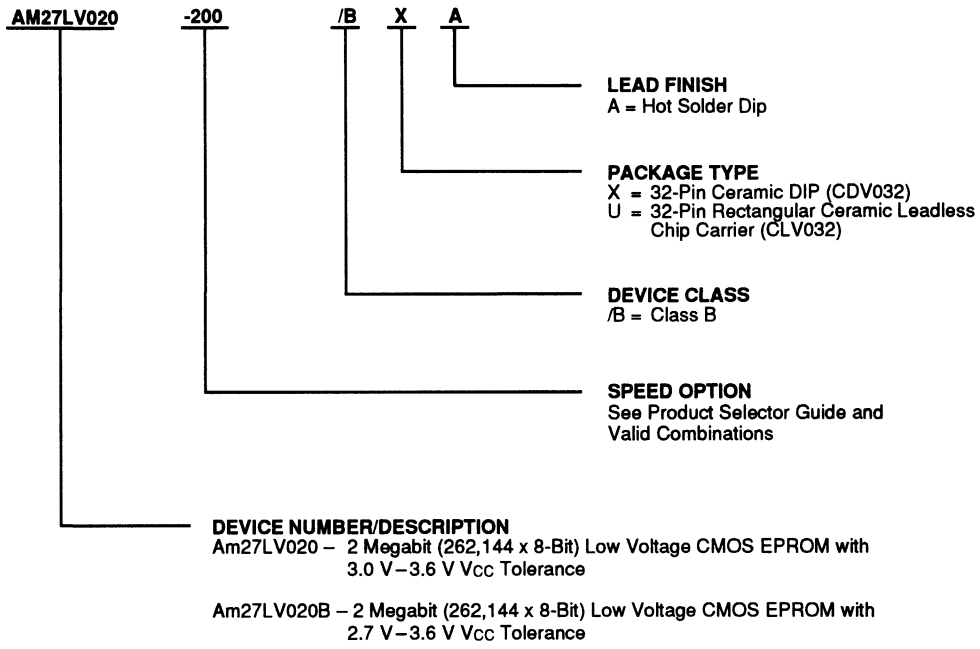
Valid Combinations	
AM27LV020-150	PC, JC, PI, JI
AM27LV020-200	
AM27LV020-250	
AM27LV020-300	
AM27LV020B-200	
AM27LV020B-250	
AM27LV020B-300	

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check on newly released combinations.

MILITARY ORDERING INFORMATION
APL Products

AMD products for Aerospace and Defense applications are available in several packages and operating ranges. APL (Approved Products List) products are fully compliant with MIL-STD-883 requirements. The order number (Valid Combination) is formed by a combination of:



Valid Combinations	
AM27LV020-200	/BXA, /BUA
AM27LV020-250	
AM27LV020-300	
AM27LV020B-250	
AM27LV020B-300	

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, or to check on newly released combinations.

Group A Tests

Group A tests consist of Subgroups 1, 2, 3, 7, 8, 9, 10, 11.

FUNCTIONAL DESCRIPTION

Erasing the Am27LV020

In order to clear all locations of their programmed contents, it is necessary to expose the Am27LV020 to an ultraviolet light source. A dosage of 15 W seconds/cm^2 is required to completely erase an Am27LV020. This dosage can be obtained by exposure to an ultraviolet lamp —wavelength of 2537 Angstroms (Å) —with intensity of 12,000 µW/cm^2 for 15 to 20 minutes. The Am27LV020 should be directly under and 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 Am27LV020, and similar devices, will erase with light sources having wavelengths shorter than 4000 Å. Although erasure times will be much longer than with UV sources at 2537Å, nevertheless the exposure to fluorescent light and sunlight will eventually erase the Am27LV020 and exposure to them should be prevented to realize maximum system reliability. If used in such an environment, the package window should be covered by an opaque label or substance.

Programming the Am27LV020

Upon delivery, or after each erasure, the Am27LV020 has all 2,097,152 bits in the "ONE", or HIGH state. "ZEROs" are loaded into the Am27LV020 through the procedure of programming.

The programming mode is entered when 12.75 V ± 0.25 V is applied to the V_{PP} pin, $\overline{CE}$ and $\overline{PGM}$ are at V_{IL} and $\overline{OE}$ is at V_{IH}.

For programming, the data to be programmed is applied 8 bits in parallel to the data output pins.

The Flashrite algorithm reduces programming time by using 100 µs programming pulse and by giving each address only as many pulses as are necessary in order to reliably program the data. After each pulse is applied to a given address, the data in that address is verified. If the data does not verify, additional pulses are given until it verifies or the maximum is reached. This process is repeated while sequencing through each address of the Am27LV020. This part of the algorithm is done at V_{CC} = 6.25 V to assure that each EPROM bit is programmed to a sufficiently high threshold voltage. After the final address is completed, the entire EPROM memory is verified at $V_{CC} = V_{PP} = 5.25$ V. Am27LV020 can be programmed using the same algorithm as the 5 V counterpart 27C020.

Program Inhibit

Programming of multiple Am27LV020s in parallel with different data is also easily accomplished. Except for $\overline{CE}$, all like inputs of the parallel Am27LV020 may be common. A TTL low-level program pulse applied to an Am27LV020 $\overline{CE}$ input with V_{PP} = 12.75 ± 0.25 V, $\overline{PGM}$ LOW, and $\overline{OE}$ HIGH will program that Am27LV020. A high-level $\overline{CE}$ input inhibits the other Am27LV020s from being programmed.

Program Verify

A verify should be performed on the programmed bits to determine that they were correctly programmed. The verify should be performed with $\overline{OE}$ and $\overline{CE}$ at V_{IL}, $\overline{PGM}$ at V_{IH}, and V_{PP} between 12.5 V and 13.0 V.

Auto Select Mode

The auto select mode allows the reading out of a binary code from an EPROM that will identify its manufacturer and type. This mode is intended for use by programming equipment for the purpose of automatically matching the device to be programmed with its corresponding programming algorithm. This mode is functional in the 25°C ± 5°C ambient temperature range that is required when programming the Am27LV020.

To activate this mode, the programming equipment must force 12.0 ± 0.5 V on address line A9 of the Am27LV020. Two identifier bytes may then be sequenced from the device outputs by toggling address line A_0 from V_{IL} to V_{IH}. All other address lines must be held at V_{IL} during auto select mode.

Byte 0 ($A0 = V_{IL}$) represents the manufacturer code, and Byte 1 ($A0 = V_{IH}$), the device identifier code. For the Am27LV020, these two identifier bytes are given in the Mode Select table. All identifiers for manufacturer and device codes will possess odd parity, with the MSB (DQ7) defined as the parity bit.

Read Mode

The Am27LV020 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 (t_{ACC}) is equal to the delay from $\overline{CE}$ to output (t_{CE}). Data is available at the outputs t_{OE} after the falling edge of $\overline{OE}$, assuming that $\overline{CE}$ has been LOW and addresses have been stable for at least $t_{ACC} - t_{OE}$.

Standby Mode

The Am27LV020 has a CMOS standby mode which reduces the maximum V_{CC} current to 25 µA. It is placed in CMOS-standby when $\overline{CE}$ is at V_{CC} ± 0.3 V. The Am27LV020 also has a TTL-standby mode which reduces the maximum V_{CC} current to 0.6 mA. It is placed in TTL-standby when $\overline{CE}$ is at V_{IH}. When in standby mode, the outputs are in a high-impedance state, independent of the $\overline{OE}$ input.

Mixed Power Supply System

Am27LV020 (in 3.0 V to 3.6 V regulated power supply) can be intefaced with 5 V system only when the I/O pins (DQ0–DQ7) are not driven by the 5 V system. $V_{IHmax} = V_{CCLV}$ + 2.2 V for address and clock pins and $V_{IHmax} = V_{CCLV}$ + 0.5 V for I/O pins should be followed to avoid CMOS latch-up condition.

Output OR-Tieing

To accommodate multiple memory connections, a two-line control function is provided to allow for:

- Low memory power dissipation
- 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.

System Applications

During the switch between active and standby conditions, transient current peaks are produced on the rising and falling edges of Chip Enable. The magnitude of these transient current peaks is dependent on the output capacitance loading of the device. At a minimum, a 0.1 µF ceramic capacitor (high frequency, low inherent inductance) should be used on each device between V_{CC} and V_{SS} to minimize transient effects. In addition, to overcome the voltage drop caused by the inductive effects of the printed circuit board traces on EPROM arrays, a 4.7 µF bulk electrolytic capacitor should be used between V_{CC} and V_{SS} for each eight devices. The location of the capacitor should be close to where the power supply is connected to the array.

Mode Select Table

Mode \ Pins		$\overline{CE}$	$\overline{OE}$	$\overline{PGM}$	A0	A9	V_{PP}	Outputs
Read		V_{IL}	V_{IL}	X	X	X	X	D_{OUT}
Output Disable		V_{IL}	V_{IH}	X	X	X	X	High Z
Standby (TTL)		V_{IH}	X	X	X	X	X	High Z
Standby (CMOS)		V_{CC} ± 0.3 V	X	X	X	X	X	High Z
Program		V_{IL}	V_{IH}	V_{IL}	X	X	V_{PP}	D_{IN}
Program Verify		V_{IL}	V_{IL}	V_{IH}	X	X	V_{PP}	D_{OUT}
Program Inhibit		V_{IH}	X	X	X	X	V_{PP}	High Z
Auto Select (Note 3)	Manufacturer Code	V_{IL}	V_{IL}	X	V_{IL}	V_H	X	01H
	Device Code	V_{IL}	V_{IL}	X	V_{IH}	V_H	X	97H

Notes:

1. *V_H = 12.0 V± 0.5 V*
2. *X can be either V_{IL} or V_{IH}*
3. *A1–A8 = A10–A17 = V_{IL}*
4. *See DC Programming Characteristics for V_{PP} voltage during programming.*

ABSOLUTE MAXIMUM RATINGS

Storage Temperature:
- OTP Products ..... −65°C to +125°C
- All Other Products ..... −65°C to +150°C

Ambient Temperature with Power Applied ..... −55°C to +125°C

Voltage with Respect to V_{SS}:
- All pins except A9, V_{PP}, and V_{CC} (Note 1) ..... −0.6 V to V_{CC} +0.6 V
- A9 and V_{PP} (Note 2) ..... −0.6 V to 13.5 V
- V_{CC} ..... −0.6 V to 7.0 V

Notes:

1. *During transitions, the input may overshoot V_{SS} to −2.0 V for periods of up to 20 ns. Maximum DC voltage on input and I/O may overshoot to V_{CC} + 2.0 V for periods of up to 20 ns.*
2. *During transitions, A9 and V_{PP} may overshoot V_{SS} to −2.0 V for periods of up to 20 ns. A9 and V_{PP} must not exceed 13.5 V for any period of time.*

Stresses above those listed under Absolute Maximum Ratings may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

OPERATING RANGES

Commercial (C) Devices
- Case Temperature (T_C) ..... 0°C to +70°C

Industrial (I) Devices
- Case Temperature (T_C) ..... −40°C to +85°C

Extended Commercial (E) Devices
- Case Temperature (T_C) ..... −55°C to +125°C

Military (M) Devices
- Case Temperature (T_C) ..... −55°C to +125°C

Supply Read Voltages:
- V_{CC} for Am27LV020 ..... +3.0 V to +3.6 V
- V_{CC} for Am27LV020B ..... +2.7 V to +3.6 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over operating ranges unless otherwise specified (Notes 1, 4, 5 and 7) (for APL products, Group A, Subgroups 1, 2, 3, 7 and 8 are tested unless otherwise noted)

Parameter Symbol	Parameter Description	Test Conditions		Min	Max	Unit
TTL and CMOS Inputs for V_{CC} = 3.0 V to 3.6 V						
V_{OH}	Output HIGH Voltage	I_{OH} = –2.0 mA		2.4		V
V_{OL}	Output LOW Voltage	I_{OL} = 2.0 mA			0.4	V
V_{IH}	Input HIGH Voltage			2.0	V_{CC} + 0.3	V
V_{IL}	Input LOW Voltage			–0.3	+0.8	V
I_{LI}	Input Load Current	V_{IN} = 0 V to V_{CC}	C/I Devices		1.0	µA
			E/M Devices		1.0	
I_{LO}	Output Leakage Current	V_{OUT} = 0 V to V_{CC}	C/I Devices		5.0	µA
			E/M Devices		5.0	
I_{CC1}	V_{CC} Active Current (Notes 5 and 8)	$\overline{CE}$ = V_{IL}, f = 5 MHz I_{OUT} = 0 mA (Open Outputs)	C/I Devices		15	mA
			E/M Devices		20	
I_{CC2}	V_{CC} TTL Standby Current	$\overline{CE}$ = V_{IH}, $\overline{OE}$ = V_{IL}	TTL		0.6	mA
I_{CC3}	V_{CC} CMOS Standby Current	$\overline{CE}$ = V_{CC} ± 0.3 V	CMOS		25	µA
I_{PP1}	V_{PP} Supply Current (Read)	$\overline{CE}$ = $\overline{OE}$ = V_{IL}, V_{PP} = V_{CC}			100	µA

Parameter Symbol	Parameter Description	Test Conditions		Min	Max	Unit
CMOS for V_{CC} = 2.7 V to 3.6 V						
V_{OH}	Output HIGH Voltage	I_{OH} = –20 µA		V_{CC} – 0.1		V
V_{OL}	Output LOW Voltage	I_{OL} = 20 µA			0.1	V
V_{IH}	Input HIGH Voltage			0.7 V_{CC}	V_{CC} + 0.3	V
V_{IL}	Input LOW Voltage			–0.3	0.2 V_{CC}	V
I_{LI}	Input Load Current	V_{IN} = 0 V to +V_{CC}	C/I Devices		1.0	µA
			E/M Devices		1.0	
I_{LO}	Output Leakage Current	V_{OUT} = 0 V to +V_{CC}	C/I Devices		5.0	µA
			E/M Devices		5.0	
I_{CC1}	V_{CC} Active Current (Notes 5 and 8)	$\overline{CE}$ = V_{IL}, f = 5 MHz, I_{OUT} = 0 mA (Open Outputs)	C/I Devices		15	mA
			E/M Devices		20	
I_{CC3}	V_{CC} CMOS Standby Current	$\overline{CE}$ = V_{CC} ± 0.3 V			25	µA
I_{PP1}	V_{PP} Supply Current (Read)	$\overline{CE}$ = $\overline{OE}$ = V_{IL}, V_{PP} = V_{CC}			100	µA

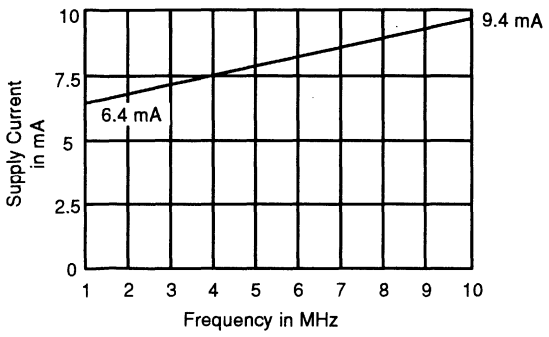


Figure 1. Typical Supply Current vs. Frequency
V_{CC} = 3.6 V, T = 25°C

17342A-5

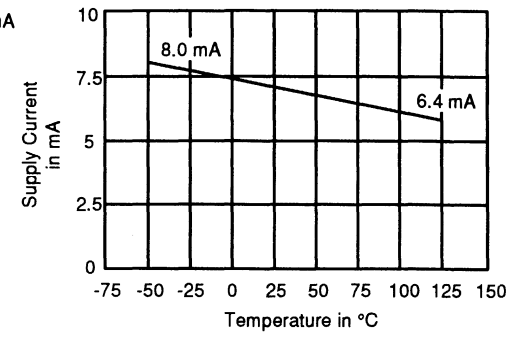


Figure 2. Typical Supply Current vs. Temperature
V_{CC} = 3.6 V, f = 5 MHz

17342A-6

CAPACITANCE (Notes 2, 3 and 7)

Parameter Symbol	Parameter Description	Test Conditions	CDV032		CLV032		PD032		PL032		Unit
			Typ	Max	Typ	Max	Typ	Max	Typ	Max	
C_{IN}	Input Capacitance	V_{IN} = 0 V	10	12	8	10	10	12	8	10	pF
C_{OUT}	Output Capacitance	V_{OUT} = 0 V	12	15	9	12	12	15	9	12	pF

Notes:

1. *V_{CC} must be applied simultaneously or before V_{PP}, and removed simultaneously or after V_{PP}.*
2. *Typical values are for nominal supply voltages.*
3. *This parameter is only sampled and not 100% tested.*
4. ***Caution**: The Am27LV020 must not be removed from, or inserted into, a socket or board when V_{CC} or V_{PP} is applied.*
5. *I_{CC1} is tested with $\overline{OE}$ = V_{IH} to simulate open outputs.*
6. *T_A = +25°C, f = 1 MHz.*
7. *During transitions, the inputs may overshoot to –2.0 V for periods less than 20 ns. Maximum DC voltage on output pins may overshoot to V_{CC} + 2.0 V for periods less than 20 ns.*
8. *For typical supply current values at various frequencies, refer to Figure 1. For temperature, refer to Figure 2.*

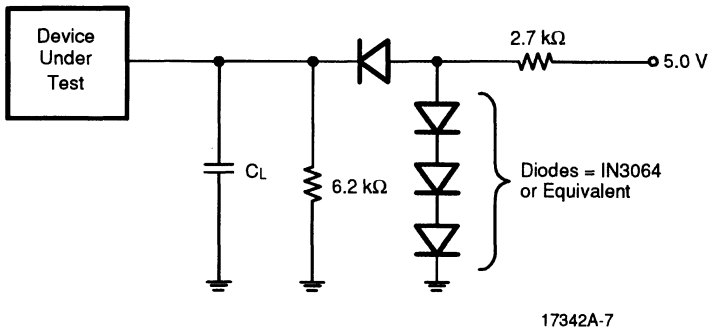
SWITCHING CHARACTERISTICS over operating ranges unless otherwise specified (Notes 1, 3 and 4) (for APL products, Group A, Subgroups 9, 10 and 11 are tested unless otherwise noted)

PRELIMINARY									
Parameter Symbols		Parameter Description	Test Conditions		Am27LV020/Am27LV020B				
JEDEC	Standard				-150	-200	-250	-300	Unit
t_{AVQV}	t_{ACC}	Address to Output Delay	$\overline{CE} = \overline{OE} = V_{IL}$	Min					ns
				Max	150	200	250	300	
t_{ELQV}	t_{CE}	Chip Enable Output Delay	$\overline{OE} = V_{IL}$	Min					ns
				Max	150	200	250	300	
t_{GLQV}	t_{OE}	Output Enable to Output Delay	$\overline{CE} = V_{IL}$	Min					ns
				Max	65	75	100	120	
t_{EHQZ}, t_{GHQZ}	t_{DF} (Note 2)	Chip Enable HIGH or Output Enable HIGH, whichever comes first, to Output Float		Min	0	0	0	0	ns
				Max	50	60	60	60	
t_{AXQX}	t_{OH}	Output Hold from Addresses, $\overline{CE}$, or $\overline{OE}$, whichever occurred first		Min	0	0	0	0	ns
				Max					

Notes:

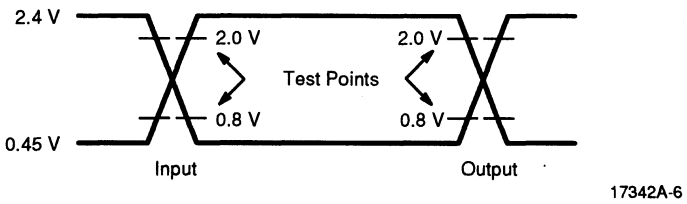
1. *V_{CC} must be applied simultaneously or before V_{PP}, and removed simultaneously or after V_{PP}.*
2. *This parameter is only sampled and not 100% tested.*
3. ***Caution**: The Am27LV020 must not be removed from, or inserted into a socket or board when V_{PP} or V_{CC} is applied.*
4. *Output Load: 1 TTL gate and C_L = 100 pF,*
   Input Rise and Fall Times: 20 ns,
   Input Pulse Levels: 0.45 V to 2.4 V,
   Timing Measurement Reference Level—Inputs: 0.8 V and 2.0 V,
   Outputs: 0.8 V and 2.0 V

SWITCHING TEST CIRCUIT



C_L = 100 pF including jig capacitance

SWITCHING TEST WAVEFORM



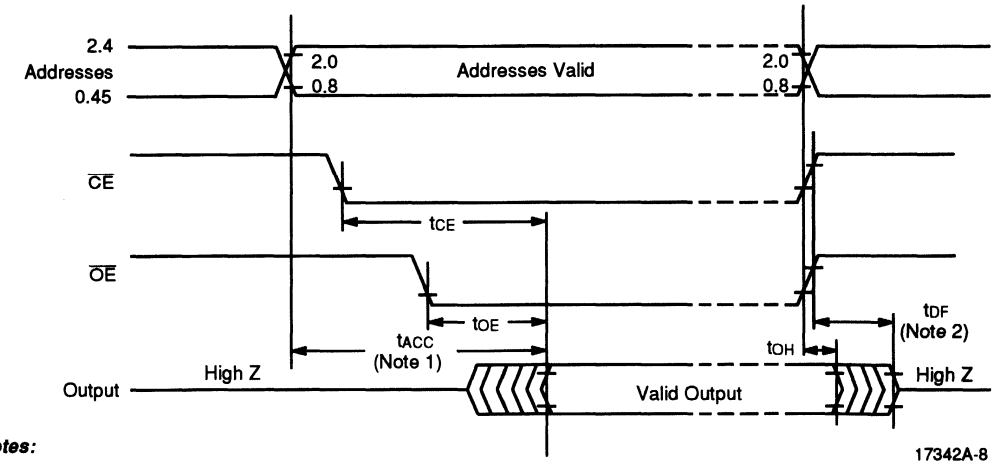
AC Testing: Inputs are driven at 2.4 V for a Logic "1" and 0.45 V for a Logic "0". Input pulse rise and fall times are ≤ 20 ns.

KEY TO SWITCHING WAVEFORMS

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
	Don't Care, Any Change Permitted	Changing, State Unknown
	Does Not Apply	Center Line is High Impedance "Off" State

KS000010

SWITCHING WAVEFORM



17342A-8

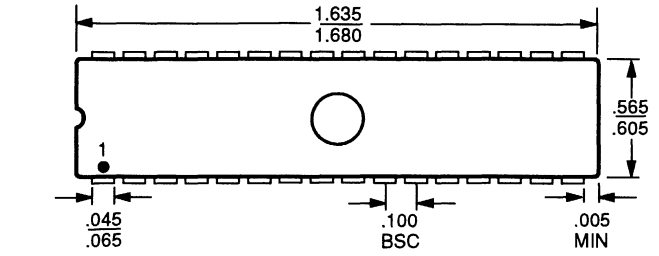
Notes:

1. $\overline{OE}$ *may be delayed up to* $t_{ACC} - t_{OE}$ *after the falling edge of addresses without impact on* t_{ACC}.
2. t_{DF} *is specified from* $\overline{OE}$ *or* $\overline{CE}$, *whichever occurs first.*

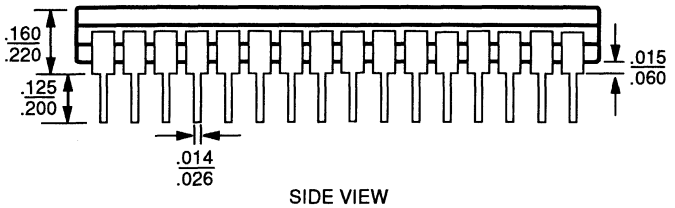
PHYSICAL DIMENSIONS*

CDV032

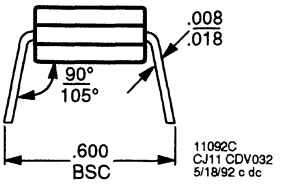
32-Pin Ceramic DIP (measured in inches)



TOP VIEW



SIDE VIEW



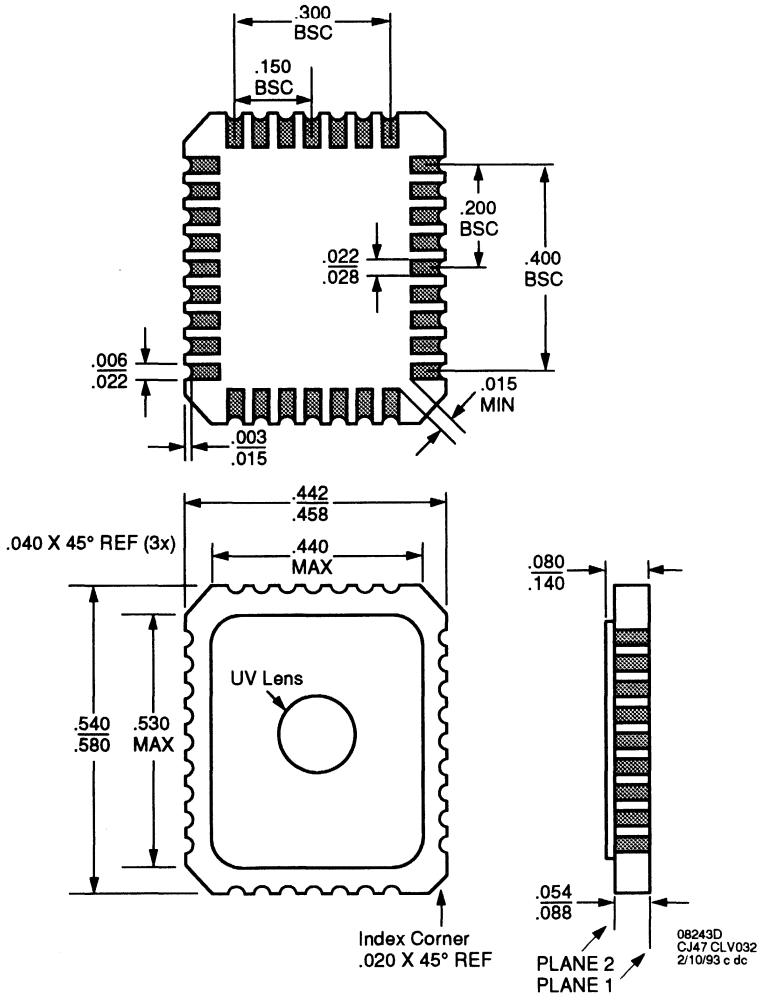
END VIEW

**For reference only. BSC is an ANSI standard for Basic Space Centering.*

PHYSICAL DIMENSIONS*

CLV032

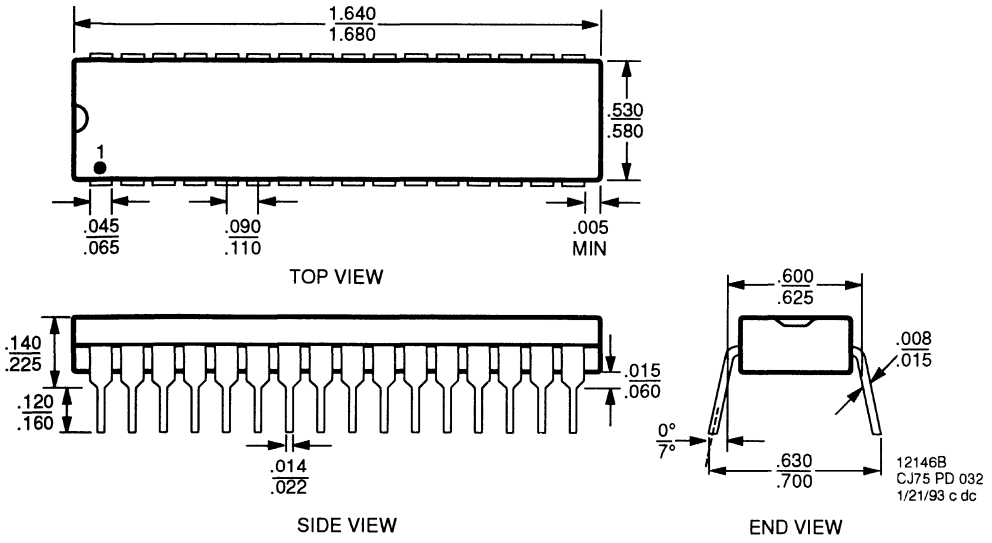
32-Pin Rectangular Ceramic Leadless Chip Carrier (measured in inches)



PHYSICAL DIMENSIONS*

PD 032

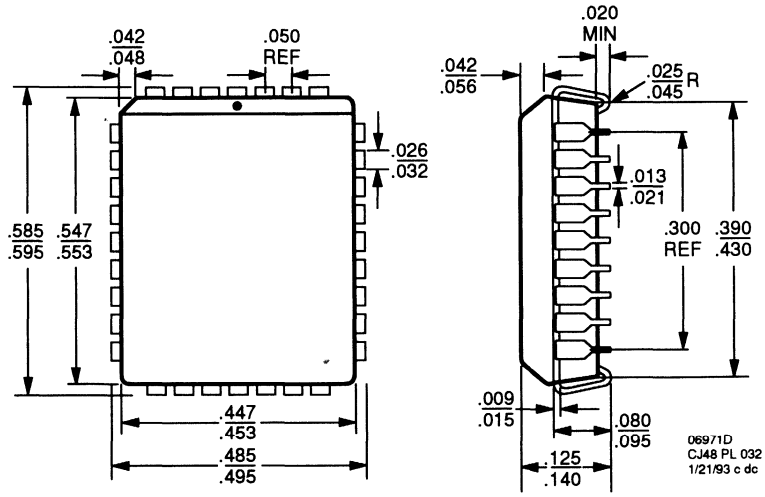
32-Pin Plastic DIP (measured in inches)



PHYSICAL DIMENSIONS*

PL 032

32-Pin Rectangular Plastic Leaded Chip Carrier (measured in inches)



Chapter 4

Programmable Array Logic (PAL) Products

CHAPTER 4
Programmable Array Logic (PAL) Products

FINAL | IND: -30

PALLV16V8Z-30

Low-Voltage, Zero-Power 20-Pin EE CMOS Universal Programmable Array Logic

Advanced Micro Devices

DISTINCTIVE CHARACTERISTICS

- **Low-voltage operation, 3.3 V JEDEC compatible**
  - V_{CC} = +3.0 V to +3.6 V
- **Zero-power CMOS technology**
  - 15 µA standby current
  - 30 ns propagation delay
- **Industrial operating temperature range**
  - T_C = –40°C to +85°C
- **Unused product term disable for reduced power consumption**
- **Pin, function and fuse-map compatible with all 20-pin GAL® devices**
- **Electrically-erasable CMOS technology provides reconfigurable logic and full testability**
- **Direct plug-in replacement for the PAL16R8 series and most of the PAL10H8 series**
- **Outputs programmable as registered or combinatorial in any combination**
- **Programmable output polarity**
- **Programmable enable/disable control**
- **Preloadable output registers for testability**
- **Automatic register reset on power up**
- **Cost-effective 20-pin plastic DIP and PLCC packages**
- **Extensive third-party software and programmer support through FusionPLD℠ partners**
- **Fully tested for 100% programming and functional yields and high reliability**

GENERAL DESCRIPTION

The PALLV16V8Z is an advanced PAL® device built with low-voltage, zero-power, high-speed, electrically-erasable CMOS technology. It is functionally compatible with all 20-pin GAL devices. The macrocells provide a universal device architecture. The PALLV16V8Z will directly replace the PAL16R8 and PAL10H8 series devices, with the exception of the PAL16C1.

The PALLV16V8Z provides zero standby power and high speed. At 15 µA maximum standby current, the PALLV16V8Z allows battery powered operation for an extended period.

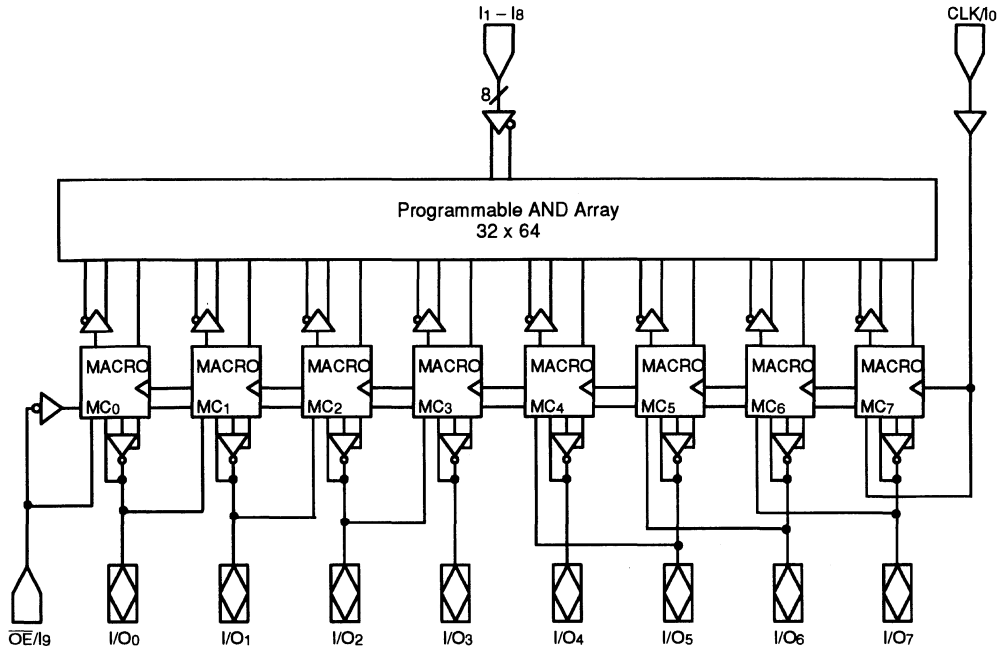
The PALLV16V8Z utilizes the familiar sum-of-products (AND/OR) architecture that allows users to implement complex logic functions easily and efficiently. Multiple levels of combinatorial logic can always be reduced to sum-of-products form, taking advantage of the very wide input gates available in PAL devices. The equations are programmed into the device through floating-gate cells in the AND logic array that can be erased electrically.

The fixed OR array allows up to eight data product terms per output for logic functions. The sum of these products feeds the output macrocell. Each macrocell can be programmed as registered or combinatorial with an active-high or active-low output. The output configuration is determined by two global bits and one local bit controlling four multiplexers in each macrocell.

AMD's FusionPLD program allows PALLV16V8Z designs to be implemented using a wide variety of popular industry-standard design tools. By working closely with the FusionPLD partners, AMD certifies that the tools provide accurate, quality support. By ensuring that third-party tools are available, costs are lowered because a designer does not have to buy a complete set of new tools for each device. The FusionPLD program also greatly reduces design time since a designer can use a tool that is already installed and familiar. Please refer to page 4-20 or the Software Reference Guide to PLD Compilers for certified development systems, and page 4-22 for the Programmer Reference Guide for approved programmers.

Publication# 17422 Rev. A Amendment /0
Issue Date: November 1992

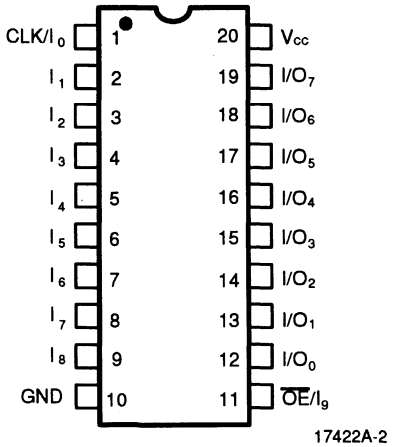
BLOCK DIAGRAM



17422A-1

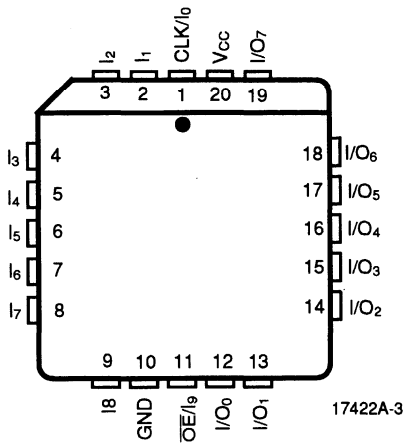
CONNECTION DIAGRAMS (Top View)

DIP



17422A-2

PLCC



17422A-3

PIN DESIGNATIONS

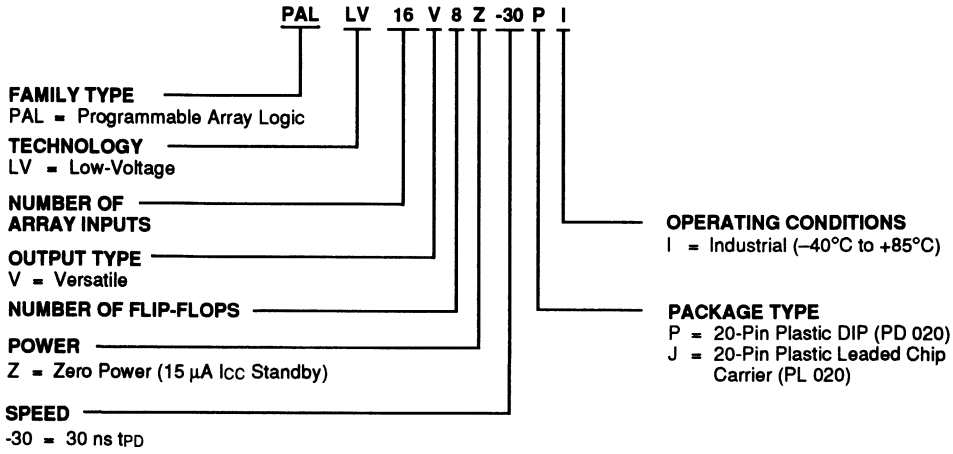
CLK = Clock
GND = Ground
I = Input
I/O = Input/Output
$\overline{OE}$ = Output Enable
V_{CC} = Supply Voltage

Note:
Pin 1 is marked for orientation.

ORDERING INFORMATION

Industrial Products

AMD programmable logic products for industrial applications are available with several ordering options. The order number (Valid Combination) is formed by a combination of:



Valid Combinations	
PALLV16V8Z-30	PI, JI

Valid Combinations

The Valid Combinations table lists configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations and to check on newly released combinations.

FUNCTIONAL DESCRIPTION

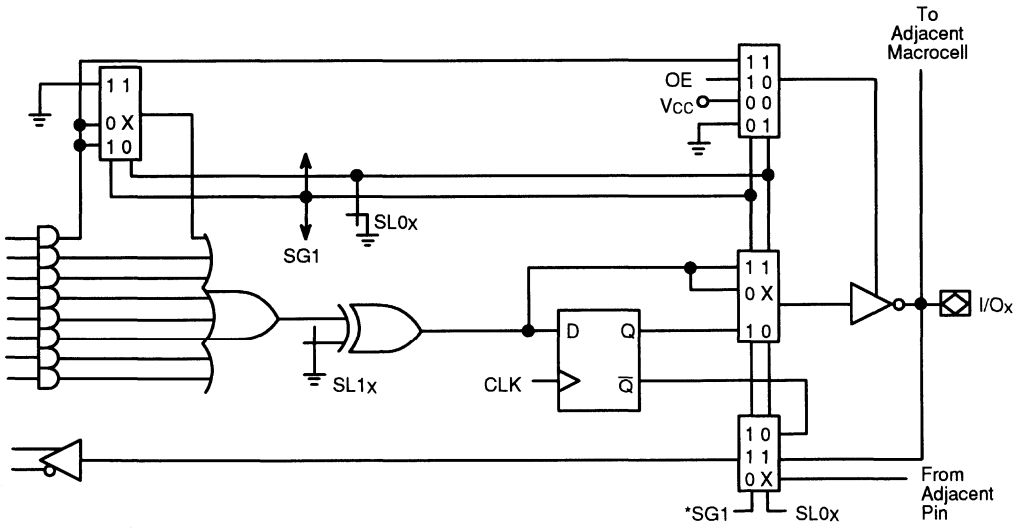
The PALLV16V8Z is a low-voltage, EE CMOS version of the PALCE16V8. In addition, the PALLV16V8Z has zero standby power and unused product term disable.

The PALLV16V8Z is a universal PAL device. It has eight independently configurable macrocells (MC_0–MC_7). Each macrocell can be configured as registered output, combinatorial output, combinatorial I/O or dedicated input. The programming matrix implements a programmable AND logic array, which drives a fixed OR logic array. Buffers for device inputs have complementary outputs to provide user-programmable input signal polarity. Pins 1 and 11 serve either as array inputs or as clock (CLK) and output enable ($\overline{OE}$), respectively, for all flip-flops.

Unused input pins should be tied directly to V_{CC} or GND. Product terms with all bits unprogrammed (disconnected) assume the logical HIGH state and product terms with both true and complement of any input signal connected assume a logical LOW state.

The programmable functions on the PALLV16V8Z are automatically configured from the user's design specification, which can be in a number of formats. The design specification is processed by development software to verify the design and create a programming file. This file, once downloaded to a programmer, configures the device according to the user's desired function.

The user is given two design options with the PALLV16V8Z. First, it can be programmed as a standard PAL device from the PAL16R8 and PAL10H8 series. The PAL programmer manufacturer will supply device codes for the standard PALdevice architectures to be used with the PALLV16V8Z. The programmer will program the PALLV16V8Z in the corresponding architecture.This allows the user to use existing standard PAL device JEDEC files without making any changes to them. Alternatively, the device can be programmed as a PALLV16V8Z. Here the user must use the PALLV16V8Z device code.This option allows full utilization of the macrocell.



**In macrocells MC_0 and MC_7, SG1 is replaced by $\overline{SG0}$ on the feedback multiplexer.*

17422A-4

Figure 1. PALLV16V8 Macrocell

Configuration Options

Each macrocell can be configured as one of the following: registered output, combinatorial output, combinatorial I/O, or dedicated input. In the registered output configuration, the output buffer is enabled by the $\overline{OE}$ pin. In the combinatorial configuration, the buffer is either controlled by a product term or always enabled. In the dedicated input configuration, it is always disabled. With the exception of MC_0 and MC_7, a macrocell configured as a dedicated input derives the input signal from an adjacent I/O. MC_0 derives its input from pin 11 ($\overline{OE}$) and MC_7 from pin 1 (CLK).

The macrocell configurations are controlled by the configuration control word. It contains 2 global bits (SG0 and SG1) and 16 local bits ($SL0_0$ through $SL0_7$ and $SL1_0$ through $SL1_7$). SG0 determines whether registers will be allowed. SG1 determines whether the PALLV16V8Z will emulate a PAL16R8 family or a PAL10H8 family device. Within each macrocell, $SL0_x$, in conjunction with SG1, selects the configuration of the macrocell, and $SL1_x$ sets the output as either active low or active high for the individual macrocell.

The configuration bits work by acting as control inputs for the multiplexers in the macrocell. There are four multiplexers: a product term input, an enable select, an output select, and a feedback select multiplexer. SG1 and $SL0_x$ are the control signals for all four multiplexers. In MC_0 and MC_7, $\overline{SG0}$ replaces SG1 on the feedback multiplexer. This accommodates CLK being the adjacent pin for MC_7 and $\overline{OE}$ the adjacent pin for MC_0.

Registered Output Configuration

The control bit settings are SG0 = 0, SG1 = 1 and $SL0_x = 0$. There is only one registered configuration. All eight product terms are available as inputs to the OR gate. Data polarity is determined by $SL1_x$. The flip-flop is loaded on the LOW-to-HIGH transition of CLK. The feedback path is from $\overline{Q}$ on the register. The output buffer is enabled by $\overline{OE}$.

Combinatorial Configurations

The PALLV16V8Z has three combinatorial output configurations: dedicated output in a non-registered device, I/O in a non-registered device and I/O in a registered device.

Dedicated Output In a Non-Registered Device

The control bit settings are SG0 = 1, SG1 = 0 and $SL0_x = 0$. All eight product terms are available to the OR gate. Although the macrocell is a dedicated output, the feedback is used, with the exception of MC_3 and MC_4. MC_3 and MC_4 do not use feedback in this mode. Because CLK and $\overline{OE}$ are not used in a non-registered device, pins 1 and 11 are available as input signals. Pin 1 will use the feedback path of MC_7 and pin 11 will use the feedback path of MC_0.

Combinatorial I/O In a Non-Registered Device

The control bit settings are SG0 = 1, SG1 = 1, and $SL0_x = 1$. Only seven product terms are available to the OR gate. The eighth product term is used to enable the output buffer. The signal at the I/O pin is fed back to the AND array via the feedback multiplexer. This allows the pin to be used as an input.

Because CLK and $\overline{OE}$ are not used in a non-registered device, pins 1 and 11 are available as inputs. Pin 1 will use the feedback path of MC_7 and pin 11 will use the feedback path of MC_0.

Combinatorial I/O in a Registered Device

The control bit settings are SG0 = 0, SG1 = 1 and $SL0_x = 1$. Only seven product terms are available to the OR gate. The eighth product term is used as the output enable. The feedback signal is the corresponding I/O signal.

Dedicated Input Configuration

The control bit settings are SG0 = 1, SG1 = 0 and $SL0_x = 1$. The output buffer is disabled. Except for MC_0 and MC_7 the feedback signal is an adjacent I/O. For MC_0 and MC_7 the feedback signals are pins 1 and 11. These configurations are summarized in Table 1 and illustrated in Figure 2.

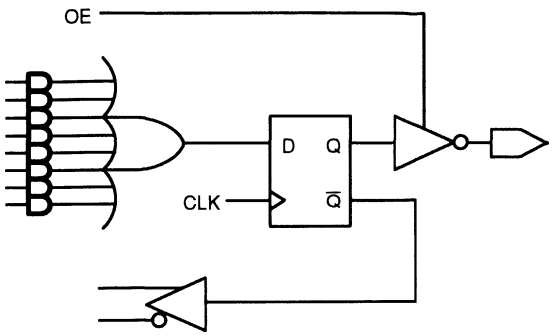
Table 1. Macrocell Configuration

SG0	SG1	SL0x	Cell Configuration	Devices Emulated
Device Uses Registers				
0	1	0	Registered Output	PAL16R8, 16R6, 16R4
0	1	1	Combinatorial I/O	PAL16R6, 16R4
Device Uses No Registers				
1	0	0	Combinatorial Output	PAL10H8, 12H6, 14H4, 16H2, 10L8, 12L6, 14L4, 16L2
1	0	1	Input	PAL12H6, 14H4, 16H2, 12L6, 14L4, 16L2
1	1	1	Combinatorial I/O	PAL16L8

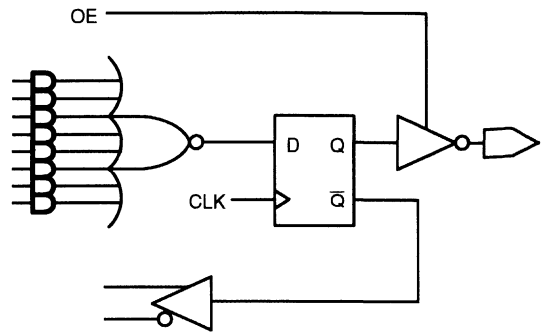
Programmable Output Polarity

The polarity of each macrocell can be active-high or active-low, either to match output signal needs or to reduce product terms. Programmable polarity allows Boolean expressions to be written in their most compact form (true or inverted), and the output can still be of the desired polarity. It can also save "DeMorganizing" efforts.

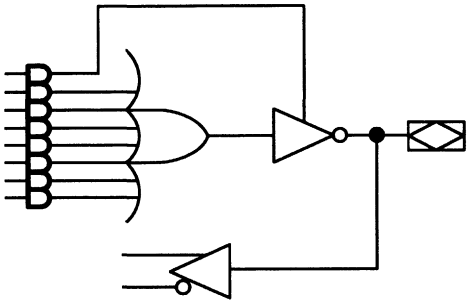
Selection is through a programmable bit $SL1_x$ which controls an exclusive-OR gate at the output of the AND/OR logic. The output is active high if $SL1_x$ is 1 and active low if $SL1_x$ is 0.



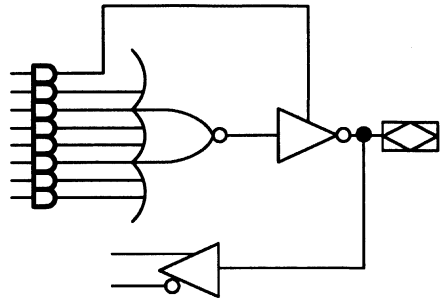
Registered Active Low



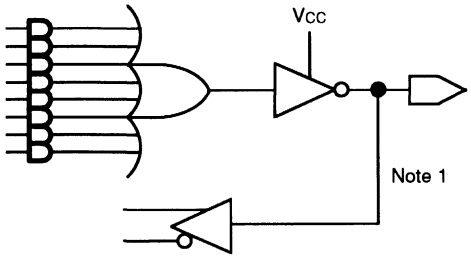
Registered Active High



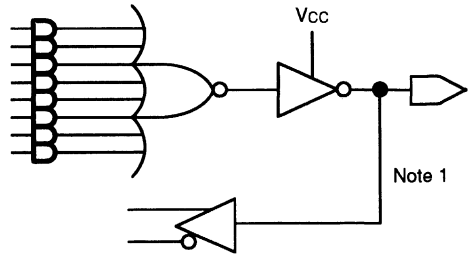
Combinatorial I/O Active Low



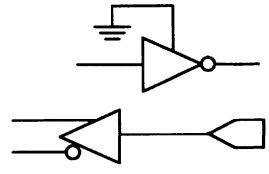
Combinatorial I/O Active High



Combinatorial Output Active Low



Combinatorial Output Active High



Dedicated Input

17422A-5

Notes:

1. *Feedback is not available on pins 15 and 16 in the combinatorial output mode.*
2. *The dedicated-input configuration is not available on pins 15 and 16.*

Figure 2. Macrocell Configurations

Benefits of Lower Operating Voltage

The PALLV16V8Z has an operating voltage range of 3.0 V to 3.6 V. Low voltage allows for lower operating power consumption, longer battery life, and/or smaller batteries for notebook applications.

Because power is proportional to the square of the voltage, reduction of the supply voltage from 5.0 V to 3.3 V significantly reduces power consumption. This directly translates to longer battery life for portable applications. Lower power consumption can also be used to reduce the size and weight of the battery. Thus, 3.3-V designs facilitate a reduction in the form factor.

A lower operating voltage results in a reduction of I/O voltage swings. This reduces noise generation and provides a less hostile environment for board design. Lower Operating Voltage also reduces electromagnetic radiation noise and makes obtaining FCC approval easier.

Zero-Standby Power Mode

The PALLV16V8Z features a zero-standby power mode. When none of the inputs switch for an extended period (typically 50 ns), the PALLV16V8Z will go into standby mode, shutting down most of its internal circuitry. The current will go to almost zero (I_{CC} < 15 µA). The outputs will maintain the states held before the device went into the standby mode.

When any input switches, the internal circuitry is fully enabled and power consumption returns to normal. This feature results in considerable power savings for operation at low to medium frequencies. This savings is illustrated in the I_{CC} vs. frequency graph on page 4-17.

Product-Term Disable

On a programmed PALLV16V8Z, any product terms that are not used are disabled. Power is cut off from these product terms so that they do not draw current. As shown in the I_{CC} vs frequency graph on page 4-17, product-term disabling results in considerable power savings. This savings is greater at the higher frequencies.

Further hints on minimizing power consumption can be found in the Application Note "Minimizing Power Consumption with Zero-Power PLDs," order #16948.

Power-Up Reset

All flip-flops power up to a logic LOW for predictable system initialization. Outputs of the PALLV16V8Z will depend on whether they are selected as registered or combinatorial. If registered is selected, the output will be HIGH. If combinatorial is selected, the output will be a function of the logic. Details on power-up reset can be found on page 4-19.

Register Preload

The register on the PALLV16V8Z can be preloaded from the output pins to facilitate functional testing of complex state machine designs. This feature allows direct loading of arbitrary states, making it unnecessary to cycle through long test vector sequences to reach a desired state. In addition, transitions from illegal states can be verified by loading illegal states and observing proper recovery.

The preload function is not disabled by the security bit. This allows functional testing after the security bit is programmed.

Security Bit

A security bit is provided on the PALLV16V8Z as a deterrent to unauthorized copying of the array configuration patterns. Once programmed, this bit defeats readback of the programmed pattern by a device programmer, securing proprietary designs from competitors. However, programming and verification are also defeated by the security bit. The bit can only be erased in conjunction with the array during an erase cycle.

Electronic Signature Word

An electronic signature word is provided in the PALLV16V8Z device. It consists of 64 bits of programmable memory that can contain user-defined data. The signature data is always available to the user independent of the security bit.

Programming and Erasing

The PALLV16V8Z can be programmed on standard logic programmers. It also may be erased to reset a previously configured device back to its unprogrammed state. Erasure is automatically performed by the programming hardware. No special erase operation is required. Approved programmers are listed on page 4-22.

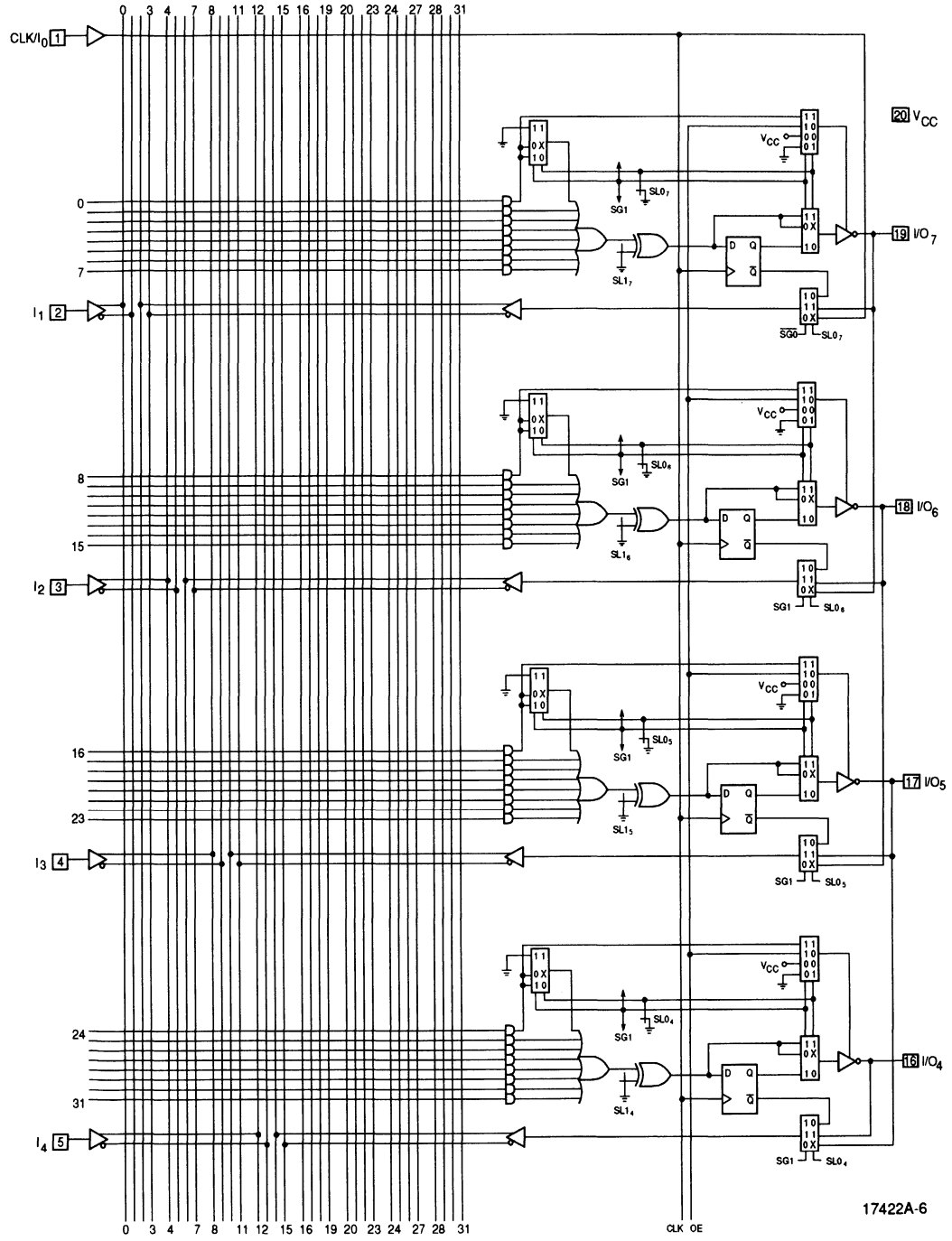
Quality and Testability

The PALLV16V8Z offers a very high level of built-in quality. The erasability if the device provides a direct means of verifying performance of all the AC and DC parameters. In addition, this verifies complete programmability and functionality of the device to yield the highest programming yields and post-programming function yields in the industry.

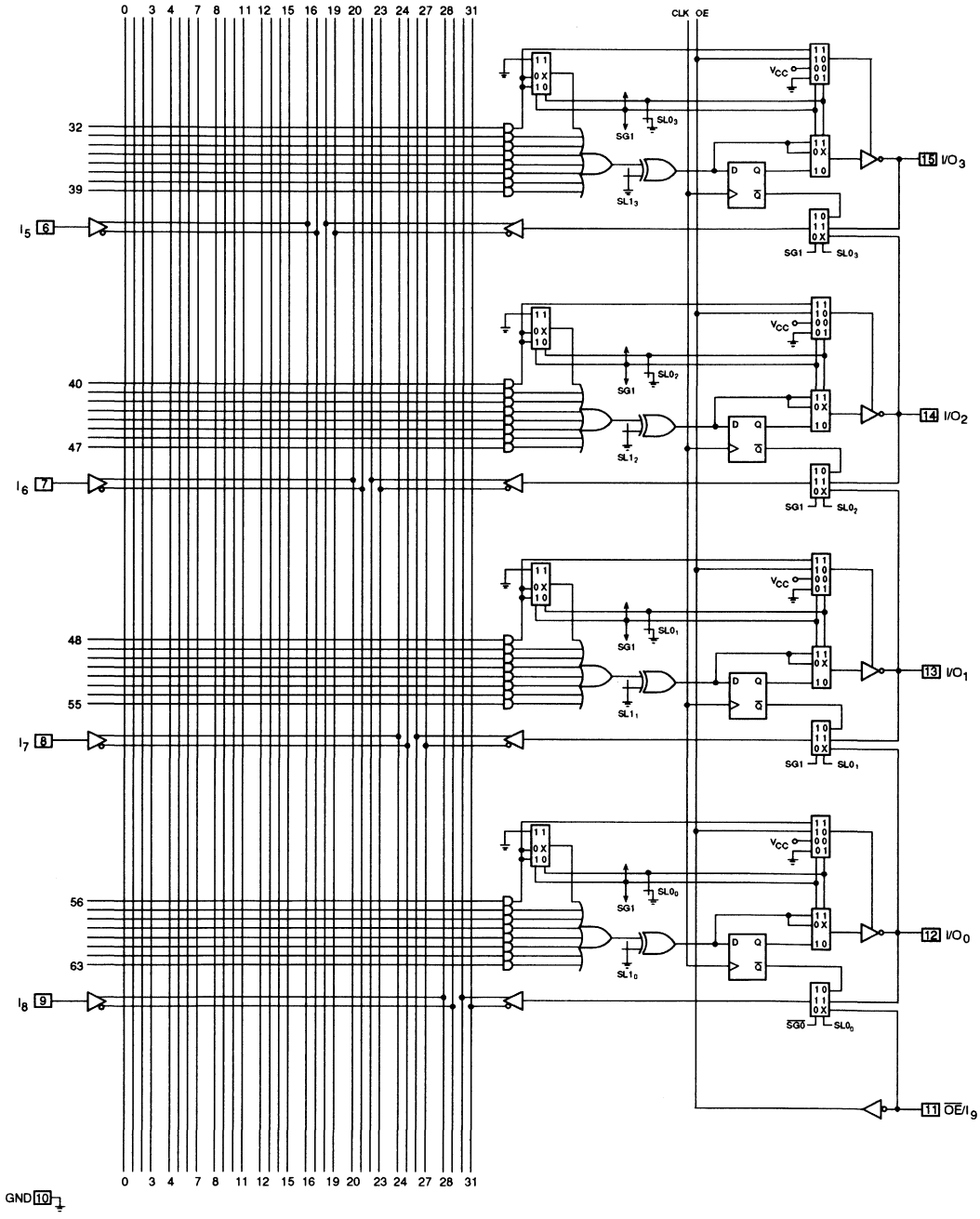
Technology

The high-speed PALLV16V8Z is fabricated with AMD's advanced electrically-erasable (EE) CMOS process. The array connections are formed with proven EE cells. This technology provides strong input-clamp diodes, output slew-rate control, and a grounded substrate for clean switching.

LOGIC DIAGRAM



LOGIC DIAGRAM (continued)



17422A-6
continued

ABSOLUTE MAXIMUM RATINGS

Storage Temperature ..... –65°C to +150°C

Ambient Temperature with Power Applied ..... –55°C to +125°C

Supply Voltage with Respect to Ground ..... –0.5 V to +7.0 V

DC Input Voltage ..... –0.5 V to V_{CC} + 0.5 V

DC Output or I/O Pin Voltage ..... –0.5 V to V_{CC} + 0.5 V

Static Discharge Voltage ..... 2001 V

Latchup Current (T_A = –40°C to 85°C) ..... 100 mA

Stresses above those listed under Absolute Maximum Ratings may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to Absolute Maximum Ratings for extended periods may affect device reliability. Programming conditions may differ.

OPERATING RANGES

Industrial (I) Devices

Operating Case Temperature (T_C) ..... –40°C to +85°C

Supply Voltage (V_{CC}) with Respect to Ground ..... +3.0 V to +3.6 V

Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over INDUSTRIAL operating ranges unless otherwise specified

Parameter Symbol	Parameter Description	Test Conditions		Min	Max	Unit
V_{OH}	Output HIGH Voltage	V_{IN} = V_{IH} or V_{IL}	I_{OH} = –2 mA	2.4		V
		V_{CC} = Min	I_{OH} = –100 µA	V_{CC} – 0.1 V		V
V_{OL}	Output LOW Voltage	V_{IN} = V_{IH} or V_{IL}	I_{OL} = 2 mA		0.4	V
		V_{CC} = Min	I_{OL} = 100 µA		0.1	V
V_{IH}	Input HIGH Voltage	Guaranteed Input Logical HIGH Voltage for all Inputs (Note 1)		2.0	V_{CC} + 0.3	V
V_{IL}	Input LOW Voltage	Guaranteed Input Logical LOW Voltage for all Inputs (Note 1)			0.8	V
I_{IH}	Input HIGH Leakage Current	V_{IN} = V_{CC}, V_{CC} = Max (Note 2)			10	µA
I_{IL}	Input LOW Leakage Current	V_{IN} = 0 V, V_{CC} = Max (Note 2)			–10	µA
I_{OZH}	Off-State Output Leakage Current HIGH	V_{OUT} = V_{CC}, V_{CC} = Max V_{IN} = V_{IH} or V_{IL} (Note 2)			10	µA
I_{OZL}	Off-State Output Leakage Current LOW	V_{OUT} = 0 V, V_{CC} = Max V_{IN} = V_{IH} or V_{IL} (Note 2)			–10	µA
I_{SC}	Output Short-Circuit Current	V_{OUT} = 0.5 V, V_{CC} = Max (Note 3)		–30	–150	mA
I_{CC}	Supply Current	Outputs Open (I_{OUT} = 0 mA)	f = 0 MHz		15	µA
		V_{CC} = Max	f = 25 MHz		55	mA

Notes:

1. *These are absolute values with respect to device ground and all overshoots due to system or tester noise are included.*
2. *I/O pin leakage is the worst case of I_{IL} and I_{OZL} (or I_{IH} and I_{OZH}).*
3. *Not more than one output should be shorted at a time and duration of the short-circuit should not exceed one second. V_{OUT} = 0.5 V has been chosen to avoid test problems caused by tester ground degradation.*

CAPACITANCE (Note 1)

Parameter Symbol	Parameter Description	Test Condition		Typ	Unit
C_{IN}	Input Capacitance	V_{IN} = 2.0 V	V_{CC} = 5.0 V, T_A = 25°C, f = 1 MHz	5	pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0 V		8	pF

Note:

1. *These parameters are not 100% tested, but are evaluated at initial characterization and at any time the design is modified where capacitance may be affected.*

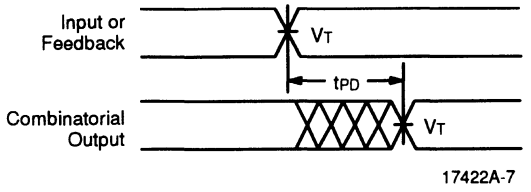
SWITCHING CHARACTERISTICS over INDUSTRIAL operating ranges (Note 2)

Parameter Symbol	Parameter Description			Min	Max	Unit
t_{PD}	Input or Feedback to Combinatorial Output (Note 3)				30	ns
t_S	Setup Time from Input or Feedback to Clock			30		
t_H	Hold Time			0		ns
t_{CO}	Clock to Output				15	ns
t_{WL}	Clock Width	LOW		13		ns
t_{WH}		HIGH		13		ns
f_{MAX}	Maximum Frequency (Note 4)	External Feedback	$1/(t_S+t_{CO})$	22		MHz
		Internal Feedback (f_{CNT})		38.4		MHz
		No Feedback	$1/(t_S+t_H)$	38.4		MHz
t_{PZX}	$\overline{OE}$ to Output Enable				25	ns
t_{PXZ}	$\overline{OE}$ to Output Disable				25	ns
t_{EA}	Input to Output Enable Using Product Term Control				30	ns
t_{ER}	Input to Output Disable Using Product Term Control				30	ns

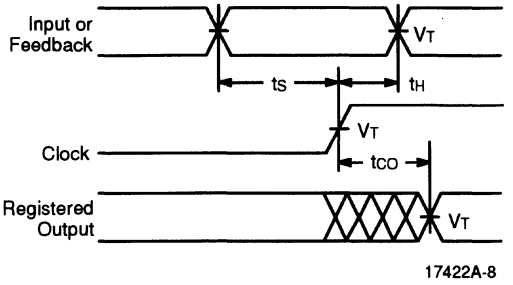
Notes:

2. *See Switching Test Circuit for test conditions.*
3. *This parameter is tested in Standby Mode. When the device is not in Standby Mode, the t_{PD} will typically be about 2 ns faster.*
4. *These parameters are not 100% tested, but are evaluated at initial characterization and at any time the design is modified where frequency may be affected.*

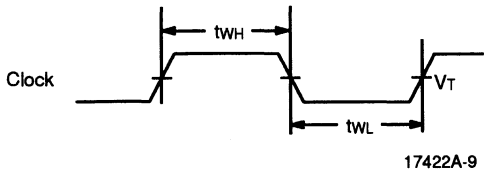
SWITCHING WAVEFORMS



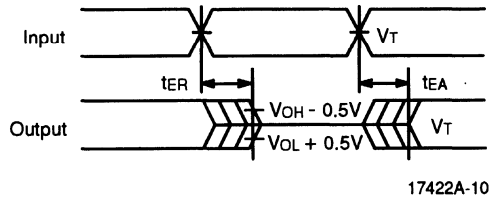
Combinatorial Output



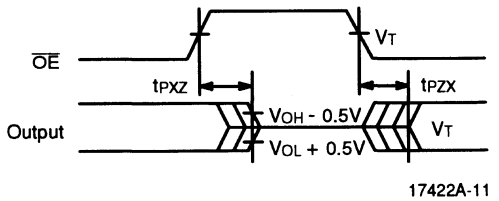
Registered Output



Clock Width



Input to Output Disable/Enable



$\overline{OE}$ to Output Disable/Enable

Notes:

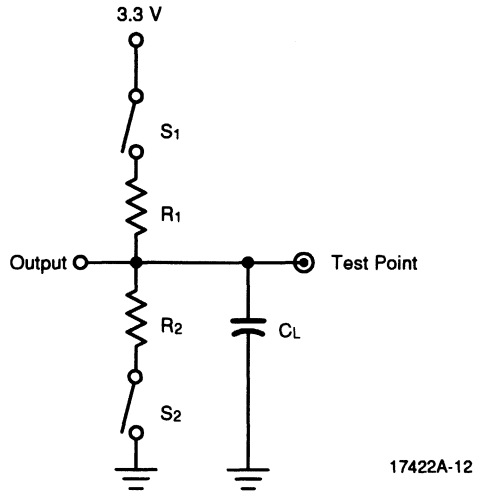
1. *V_T = 1.5 V for Input Signals and 1.65 V for Output Signals.*
2. *Input pulse amplitude 0 V to 3.0 V.*
3. *Input rise and fall times 2–5 ns typical.*

KEY TO SWITCHING WAVEFORMS

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
	Don't Care, Any Change Permitted	Changing, State Unknown
	Does Not Apply	Center Line is High-Impedance "Off" State

KS000010-PAL

SWITCHING TEST CIRCUIT



17422A-12

Specification	S_1	S_2	C_L	R_1	R_2	Measured Output Value
t_{PD}, t_{CO}	Closed	Closed	30 pF	1.6K	1.6K	1.65 V
t_{PZX}, t_{EA}	Z → H: Open Z → L: Closed	Z → H: Closed Z → L: Open				1.65 V
t_{PXZ}, t_{ER}	H → Z: Open L → Z: Closed	H → Z: Closed L → Z: Open	5 pF			H → Z: V_{OH} – 0.5 V L → Z: V_{OL} + 0.5 V

ENDURANCE CHARACTERISTICS

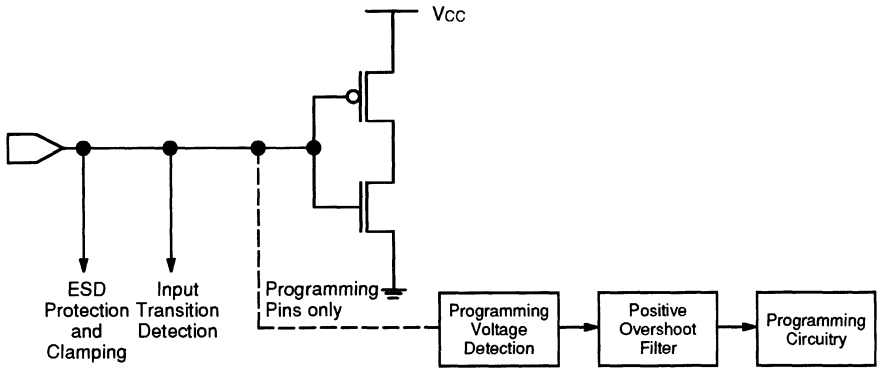
The PALLV16V8Z is manufactured using AMD's advanced Electrically Erasable process. This technology uses an EE cell to replace the fuse link used in bipolar parts. As a result, the device can be erased ar reprogrammed – a feature which allows 100% testing the factory.

Symbol	Parameter	Min	Units	Test Conditions
t_{DR}	Min Pattern Data Retention Time	10	Years	Max Storage Temperature
		20	Years	Max Operating Temperature
N	Min Reprogramming Cycles	100	Cycles	Normal Programming Conditions

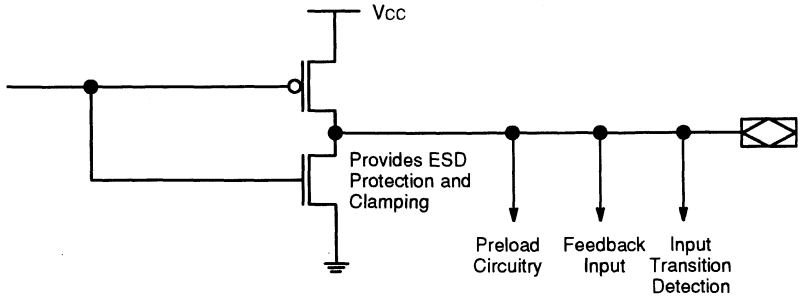
ROBUSTNESS FEATURES

The PALLV16V8Z-30 has some unique features that make it extremely robust, especially when operating in high-speed design environments. Input clamping circuitry limits negative overshoot, eliminating the possibility of false clocking caused by subsequent ringing. special noise filter makes the programming circuit completely insensitive to any positive overshoot th has a pulse width of less than about 100 ns.

INPUT/OUTPUT EQUIVALENT SCHEMATICS



Typical Input



Typical Output

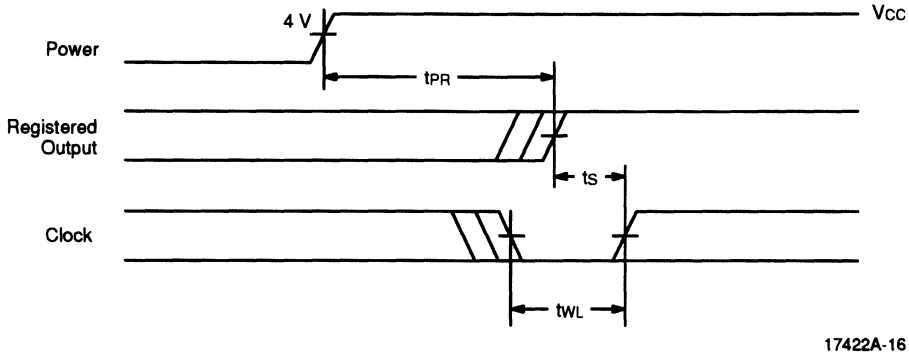
17422A-15

POWER-UP RESET

The PALLV16V8 has been designed with the capability to reset during system power-up. Following power-up, all flip-flops will be reset to LOW. The output state will be HIGH independent of the logic polarity. This feature provides extra flexibility to the designer and is especially valuable in simplifying state machine initialization. A timing diagram and parameter table are shown below. Due to the synchronous operation of the power-up reset and the wide range of ways V_{CC} can rise to its steady state, two conditions are required to insure a valid power-up reset. These conditions are:

1. The V_{CC} rise must be monotonic.
2. Following reset, the clock input must not be driven from LOW to HIGH until all applicable input and feedback setup times are met.

Parameter Symbol	Parameter Descriptions	Min	Max	Unit
t_{PR}	Power-Up Reset Time		1000	ns
t_S	Input or Feedback Setup Time	See Switching Characteristics		
t_{WL}	Clock Width LOW			



17422A-16

DEVELOPMENT SYSTEMS (subject to change)

For more information on the products listed below, please consult the AMD FusionPLD Catalog.

MANUFACTURER	COMPILERS
Advanced Micro Devices, Inc. 901 Thompson Place MS 1028 Sunnyvale, CA 94088-3543 (800) 222-9323 or (408) 732-2400	PALASM® Software Rev. 1.0
Cadence (Valid) Design Systems, Inc. 555 River Oaks Parkway San Jose, CA 95134 (408) 943-1234	Contact Cadence
Capilano Computing Systems, Ltd. 960 Quayside Dr., Suite 406 New Westminster, B.C. Canada V3M 6G2 (800) 444-9064 or (604) 522-6200	Contact Capilano
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 332-8246 or (206) 881-6444	ABEL™-4 Software Rev. 2.0
ISDATA GmbH Daimlerstr. 51 D-7500 Karlsruhe 21 Germany 0721/75 10 87 or (408) 373-7359 (U.S.)	LOG/iC™ Software Rev. 3.2
Logical Devices, Inc. 1201 E. Northwest 65th Pl. Fort Lauderdale, FL 33309 (800) 331-7766 or (305) 974-0967	CUPL™ Software Rev. 4.0
Mentor Graphics Corp. 8005 S.W. Beckman Rd. Wilsonville, OR 97070-7777 (800) 345-2308	Contact Mentor Graphics
MINC Incorporated 6755 Earl Drive, Suite 200 Colorado Springs, CO 80918 (719) 590-1155	PLDesigner® Software Rev. 2.1
OrCAD 3175 N.W. Aloclek Dr. Hillsboro, OR 97124 (503) 690-9881	Programmable Logic Design Tools Rev. 4.0
Viewlogic Systems, Inc. 293 Boston Post Road West Marlboro, MA 01752 (800) 422-4660 or (508) 480-0881	Contact Viewlogic
MANUFACTURER	**SCHEMATIC EDITORS AND LIBRARIES**
OrCAD 3175 N.W. Aloclek Dr. Hillsboro, OR 97124 (503) 690-9881	Contact OrCAD
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 32-8246 or (206) 881-6444	Contact Data I/O

DEVELOPMENT SYSTEMS (subject to change)

For more information on the products listed below, please consult the AMD FusionPLD Catalog.

MANUFACTURER	SIMULATORS
ALDEC Company, Inc. 3525 Old Conejo Rd., Suite 111 Newbury Park, CA 91320 (805) 499-6867	Contact ALDEC
Cadence (Valid) Design Systems, Inc. 555 River Oaks Parkway San Jose, CA 95134 (408) 943-1234	Contact Cadence
iNt GmbH Bunsenstrasse 6 D-8033 Martinsreid/Munich Germany (89) 857-6667	Contact iNt
Logic Automation, Inc. 19500 NW Gibbs Dr. P.O. Box 310 Beaverton, OR 97075 (503) 690-6900	Contact Logic Automation
OrCAD 3175 N.W. Aloclek Dr. Hillsboro, OR 97124 (503) 690-9881	Contact OrCAD
Viewlogic Systems, Inc. 293 Boston Post Road West Marlboro, MA 01752 (800) 422-4660 or (508) 480-0881	Contact Viewlogic
MANUFACTURER	**TEST GENERATION SYSTEM**
Acugen Software, Inc. 427-3 Amherst St., Suite 391 Nashua, NH 03063 (603) 891-1995	Contact Acugen
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 332-8246 or (206) 881-6444	Contact Data I/O
iNt GmbH Bunsenstrasse 6 D-8033 Martinsreid/Munich Germany (89) 857-6667	Contact iNt

Advanced Micro Devices is not responsible for any information relating to the products of third parties. The inclusion of such information is not a representation nor an endorsement by AMD of these products.

APPROVED PROGRAMMERS (subject to change)

For more information on the products listed below, please consult the AMD FusionPLD Catalog.

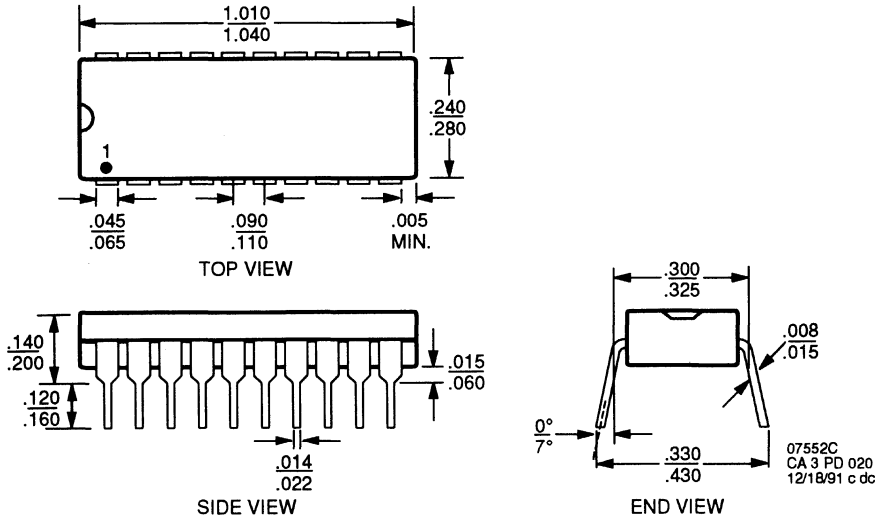
Manufacturer	Programmer Configuration
Advanced Micro Devices, Inc. 901 Thompson Place Sunnyvale, CA 94088 (800) 222-9323 or (408) 732-2400	LabPro™ Rev. A1.3
Advin Systems, Inc. 1050-L East Duane Avenue Sunnyvale, CA 94086 (408) 243-7000	U40 Rev. 10.36 U80 Rev. 10.36
BP Microsystems 10681 Haddington, Suite #190 Houston, TX 77043 (800) 225-2102 or (713) 461-9430	PLD-1128 Rev. 1.86
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 332-8246 or (206) 881-6444	UniSite™ DIP: Rev. 3.6 Model 2900 Rev. 1.9 M3900 Rev. 1.3 Family/Pinout Codes: 80-E1
Digelec, Inc. 20144 Plummer St. Chatsworth, CA 91311 (800) 367-8750 or (818) 701-9677 or Digitronics 25 Galgaley Haplada St. Herzliya B46722, Israel 52-55-9615	ALLPRO88 Rev. 2.2
Logical Devices, Inc. 1201 E. Northwest 65th Pl. Fort Lauderdale, FL 33309 (800) 331-7766 or (305) 974-0967	Contact Logical Devices
Microprocess Parc d'Activite des Pres 5, rue Denis-Papin 59650 Villeneuve-d'Ascq, France (20) 47.90.40	Contact Microprocess

APPROVED PROGRAMMERS (continued) (subject to change)

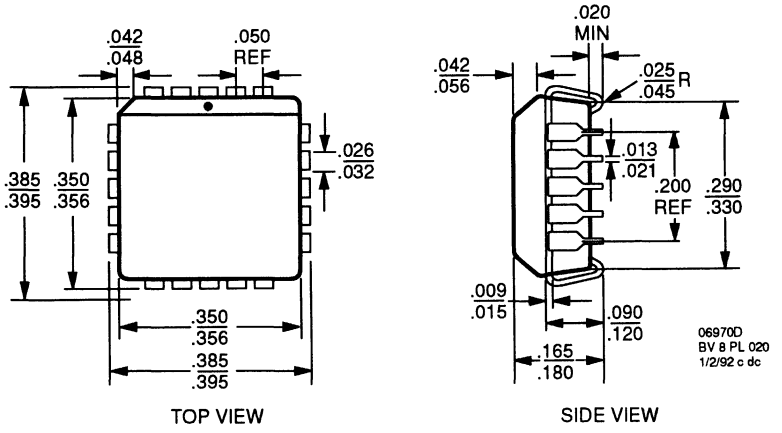
Manufacturer	Programmer Configuration
SMS North America, Inc. 16552 NE 135th Pl. Redmond, WA 98052 (800) 722-4122 or (206) 883-8447 or SMS Im Morgental 13 D-8994 Hergatz, Germany 07522-5018	Contact SMS
Stag Microsystems 1600 Wyatt Dr. Suite 3 Santa Clara, CA 95054 (408) 988-1118 or Stag House Martinfield, Welwyn Garden City Herfordshire UK AL7 1JT 707-332148	Contact Stag
System General Corp. 510 S. Park Victoria Dr. Milpitas, CA 95035 (408) 263-6667 or System General Corp. 3F, No. 1, Alley 8, Lane 45 Bao Shing Rd., Shin Diau Taipei, Taiwan 2-917-3005	Contact System General

PHYSICAL DIMENSIONS*

PD 020
20-Pin Plastic DIP (measured in inches)



PL 020
20-Pin Plastic Leaded Chip Carrier (measured in inches)



**For reference only. BSC is an ANSI standard for Basic Space Centering.*

PRELIMINARY

PALLV22V10Z-25

Low-Voltage, Zero-Power 24-Pin EE CMOS Versatile PAL Device

Advanced Micro Devices

DISTINCTIVE CHARACTERISTICS

- **Low-voltage operation, 3.3 V JEDEC compatible**
- **Zero-power CMOS technology**
  - — 15 µA standby current
  - — 25 ns first-access propagation delay
- **Unused product term disable for reduced power consumption**
- **Industrial operating temperature range**
  - — T_C = –40°C to +85°C
- **3.3 V (CMOS) and 5 V (CMOS and TTL)-compatible inputs and I/O**
- **Electrically-erasable technology provides reconfigurable logic and full testability**
- **10 macrocells programmable as registered or combinatorial, and active high or active low to match application needs**
- **Varied product term distribution allows up to 16 product terms per output for complex functions**
- **Global asynchronous reset and synchronous preset for initialization**
- **Power-up reset for initialization and register preload for testability**
- **Extensive third-party software and programmer support through FusionPLD℠ partners**
- **24-pin SKINNYDIP®, 24-pin SOIC and 28-pin PLCC packages save space**

GENERAL DESCRIPTION

The PALLV22V10Z is an advanced PAL device built with low-voltage, zero-power, high-speed, electrically-erasable CMOS technology. It provides user-programmable logic for replacing conventional zero-power CMOS SSI/MSI gates and flip-flops at a reduced chip count.

The PALLV22V10Z provides high speed at low voltage and zero standby power. At 15 µA maximum standby current, the PALLV22V10Z allows battery powered operation for an extended period.

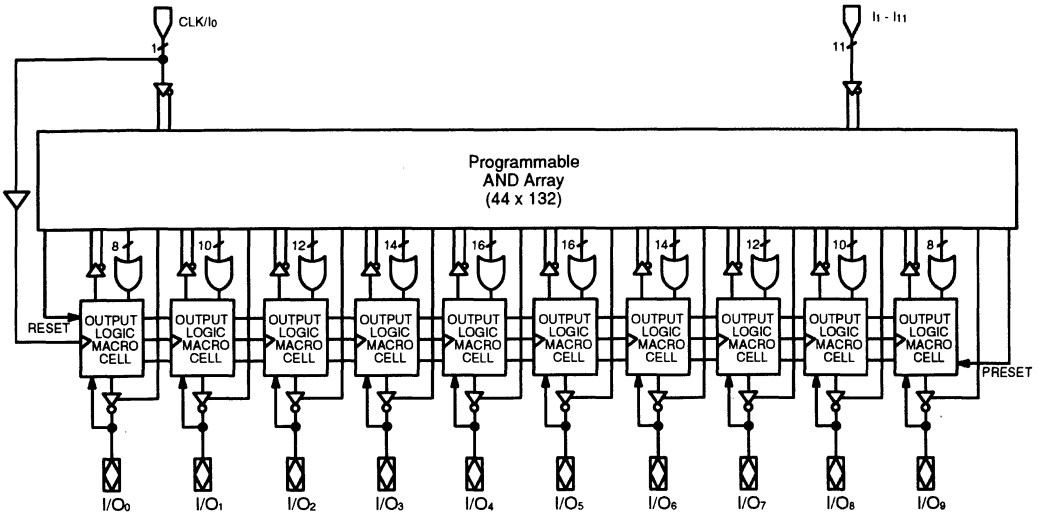
The ZPAL device implements the familiar Boolean logic transfer function, the sum of products. The PAL device is a programmable AND array driving a fixed OR array. The AND array is programmed to create custom product terms, while the OR array sums selected terms at the outputs.

The product terms are connected to the fixed OR array with a varied distribution from 8 to16 across the outputs (see Block Diagram). The OR sum of the products feeds the output macrocell. Each macrocell can be programmed as registered or combinatorial, and active high or active low. The output configuration is determined by two bits controlling two multiplexers in each macrocell.

AMD's FusionPLD program allows PALLV22V10Z designs to be implemented using a wide variety of popular industry-standard design tools. By working closely with the FusionPLD partners, AMD certifies that the tools provide accurate, quality support. By ensuring that third-party tools are available, costs are lowered because a designer does not have to buy a complete set of new tools for each device. The FusionPLD program also greatly reduces design time since a designer can use a tool that is already installed and familiar. Please refer to page 4-39 or the Software Reference Guide to PLD Compliers for certified development systems, and page 4-41 or the Programmer Reference Guide for approved programmers.

Publication# 17661 Rev. A Amendment /0
Issue Date: March 1993

BLOCK DIAGRAM

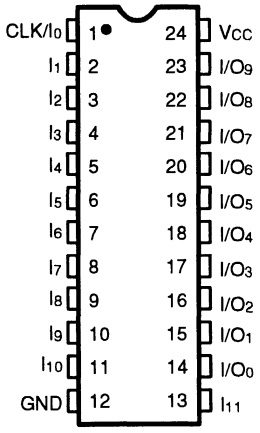


17661A-1

CONNECTION DIAGRAMS

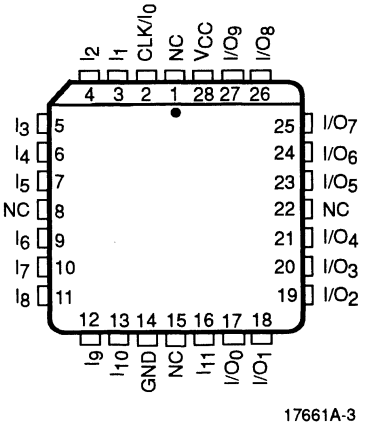
Top View

SKINNYDIP/SOIC



17661A-2

PLCC



17661A-3

Note:

Pin 1 is marked for orientation.

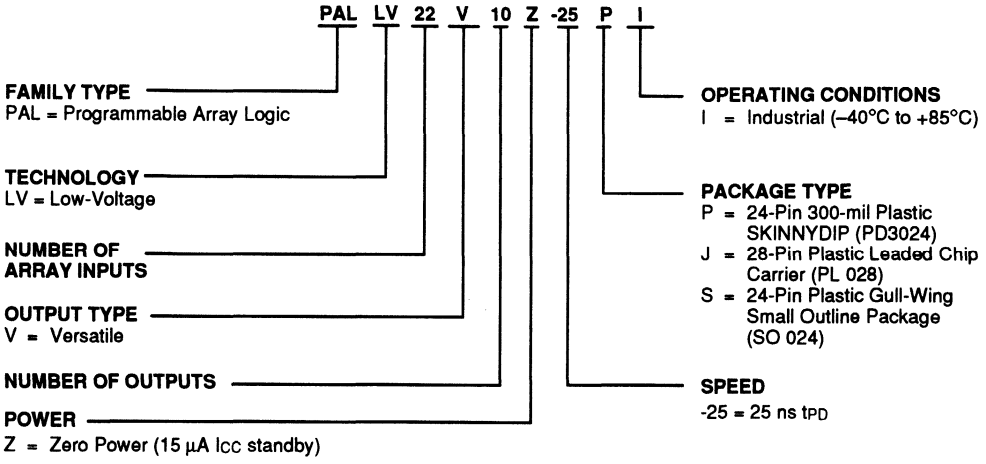
PIN DESCRIPTION

CLK	=	Clock
GND	=	Ground
I	=	Input
I/O	=	Input/Output
NC	=	No Connect
V_{CC}	=	Supply Voltage

ORDERING INFORMATION

Industrial Products

AMD programmable logic products for industrial applications are available with several ordering options. The order number (Valid Combination) is formed by a combination of these elements:



Valid Combinations	
PALLV22V10Z-25	PI, JI, SI

Valid Combinations

The Valid Combinations table lists configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, and to check on newly released combinations.

FUNCTIONAL DESCRIPTION

The PALLV22V10Z is the low-voltage, zero-power version of the PALCE22V10. It has all the architectural features of the PALCE22V10. In addition, the PALLV22V10Z has zero standby power and unused product term disable.

The PALLV22V10Z allows the systems engineer to implement the design on-chip, by programming EE cells to configure AND and OR gates within the device, according to the desired logic function. Complex interconnections between gates, which previously required time-consuming layout, are lifted from the PC board and placed on silicon, where they can be easily modified during prototyping or production.

Product terms with all connections opened assume the logical HIGH state; product terms connected to both true and complement of any single input assume the logical LOW state.

The PALLV22V10Z has 12 inputs and 10 I/O macrocells. The macrocell (Figure 1) allows one of four potential output configurations; registered output or combinatorial I/O, active high or active low (see Figure 2). The configuration choice is made according to the user's design specification and corresponding programming of the configuration bits S_0 - S_1. Multiplexer controls are connected to ground (0) through a programmable bit, selecting the "0" path through the multiplexer. Erasing the bit disconnects the control line from GND and it floats to V_{CC} (1), selecting the "1" path.

The device is produced with a EE cell link at each input to the AND gate array, and connections may be selectively removed by applying appropriate voltages to the circuit. Utilizing an easily-implemented programming algorithm, these products can be rapidly programmed to any customized pattern.

Variable Input/Output Pin Ratio

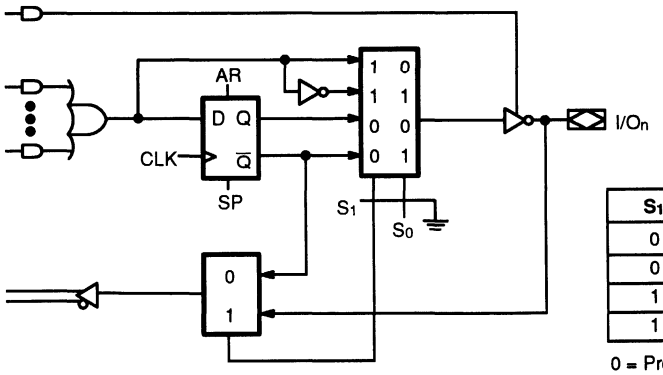
The PALLV22V10Z has twelve dedicated input lines, and each macrocell output can be an I/O pin. Buffers for device inputs have complementary outputs to provide user-programmable input signal polarity. Unused input pins should be tied to V_{CC} or GND.

Registered Output Configuration

Each macrocell of the PALLV22V10Z includes a D-type flip-flop for data storage and synchronization. The flip-flop is loaded on the LOW-to-HIGH transition of the clock input. In the registered configuration ($S_1 = 0$), the array feedback is from $\overline{Q}$ of the flip-flop.

Combinatorial I/O Configuration

Any macrocell can be configured as combinatorial by selecting the multiplexer path that bypasses the flip-flop ($S_1 = 1$). In the combinatorial configuration the feedback is from the pin.



S_1	S_0	Output Configuration
0	0	Registered/Active Low
0	1	Registered/Active High
1	0	Combinatorial/Active Low
1	1	Combinatorial/Active High

0 = Programmed EE bit
1 = Erased (charged) EE bit

17661A-4

Figure 1. Output Logic Macrocell

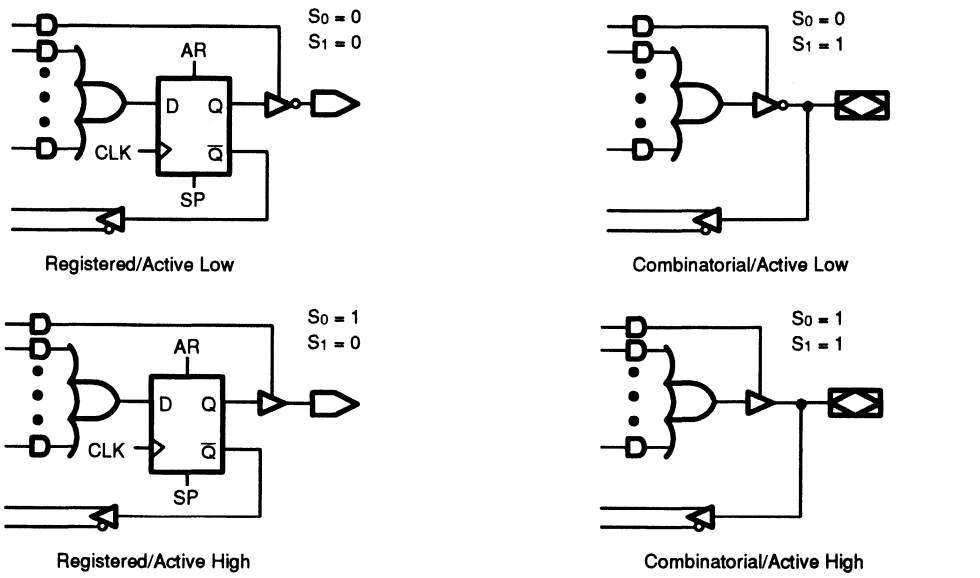


Figure 2. Macrocell Configuration Options

Programmable Three-State Outputs

Each output has a three-state output buffer with three-state control. A product term controls the buffer, allowing enable and disable to be a function of any product of device inputs or output feedback. The combinatorial output provides a bidirectional I/O pin, and may be configured as a dedicated input if the buffer is always disabled.

Programmable Output Polarity

The polarity of each macrocell output can be active high or active low, either to match output signal needs or to reduce product terms. Programmable polarity allows Boolean expressions to be written in their most compact form (true or inverted), and the output can still be of the desired polarity. It can also save "DeMorganizing" efforts.

Selection is controlled by programmable bit S_0 in the output macrocell, and affects both registered and combinatorial outputs. Selection is automatic, based on the design specification and pin definitions. If the pin definition and output equation have the same polarity, the output is programmed to be active high ($S_0 = 1$).

Preset/Reset

For initialization, the PALLV22V10Z has additional Preset and Reset product terms. These terms are connected to all registered outputs. When the Synchronous Preset (SP) product term is asserted high, the output registers will be loaded with a HIGH on the next LOW-to-HIGH clock transition. When the Asynchronous Reset (AR) product term is asserted high, the output registers will be immediately loaded with a LOW independent of the clock.

Note that preset and reset control the flip-flop, not the output pin. The output level is determined by the output polarity selected.

Benefits of Lower Operating Voltage

The PALLV22V10Z has an operating voltage range of 3.0 V to 3.6 V. Low voltage allows for lower operating power consumption, longer battery life, and/or smaller batteries for notebook applications.

Because power is proportional to the square of the voltage, reduction of the supply voltage from 5.0 V to 3.3 V significantly reduces power consumption. This directly translates to longer battery life for portable applications. Lower power consumption can also be used to reduce the size and weight of the battery. Thus, 3.3 V designs facilitate a reduction in the form factor.

A lower operating voltage results in a reduction of I/O voltage swings. This reduces noise generation and provides a less hostile environment for board design. Lower operating voltage also reduces electromagnetic radiation noise and makes obtaining FCC approval easier.

Zero-Standby Power Mode

The PALLV22V10Z features a zero-standby power mode. When none of the inputs switch for an extended period (typically 30 ns), the PALLV22V10Z will go into standby mode, shutting down most of its internal circuitry. The current will go to almost zero (I_{CC} < 15 μA). The outputs will maintain the states held before the device went into the standby mode.

If a macrocell is used in registered mode, switching pin CLK/I_0 will not affect standby mode status for that macrocell. If a macrocell is used in combinatorial mode, switching pin CLK/I_0 will affect standby mode status for that macrocell.

This feature reduces dynamic I_{CC} proportional to the number of registered macrocells used. If all macrocells are used as registers, and only CLK/I_0 is switching, the device will not be in standby mode but dynamic I_{CC} will typically be < 2 mA. This is because only the CLK/I_0 buffer will draw current.

When any input switches, the internal circuitry is fully enabled and power consumption returns to normal. This feature results in considerable power savings for operation at low to medium frequencies.

Product-Term Disable

On a programmed PALLV22V10Z, any product terms that are not used are disabled. Power is cut off from these product terms so that they do not draw current. Product-term disabling results in considerable power savings. This savings is greater at the higher frequencies.

Further hints on minimizing power consumption can be found in the Application Note "Minimizing Power Consumption with Zero-Power PLDs," order #16948.

3.3 V (CMOS) and 5 V (CMOS and TTL) Compatible Inputs and I/O

Input voltages can be at TTL levels without the device drawing more current than true 3.3 V CMOS levels. Additionally, the PALLV22V10Z can be driven with true 5 V CMOS levels due to special input and I/O buffer circuitry.

Power-Up Reset

All flip-flops power-up to a logic LOW for predictable system initialization. Outputs of the PALLV22V10Z will depend on the programmed output polarity. The V_{CC} rise must be monotonic and the reset delay time is 1000 ns maximum. Details on power-up reset can be found on page 4-38.

Register Preload

The registers on the PALLV22V10Z can be preloaded from the output pins to facilitate functional testing of complex state machine designs. This feature allows direct loading of arbitrary states, making it unnecessary to cycle through long test vector sequences to reach a desired state. In addition, transitions from illegal states can be verified by loading illegal states and observing proper recovery.

Security Bit

After programming and verification, a PALLV22V10Z design can be secured by programming the security EE bit. Once programmed, this bit defeats readback of the internal programmed pattern by a device programmer, securing proprietary designs from competitors. When the security bit is programmed, the array will read as if every bit is erased, and preload will be disabled.

The bit can only be erased in conjunction with erasure of the entire pattern.

Programming and Erasing

The PALLV22V10Z can be programmed on standard logic programmers. It also may be erased to reset a previously configured device back to its virgin state. Erasure is automatically performed by the programming hardware. No special erase operation is required. Approved programmers are listed on page 4-41.

Quality and Testability

The PALLV22V10Z offers a very high level of built-in quality.

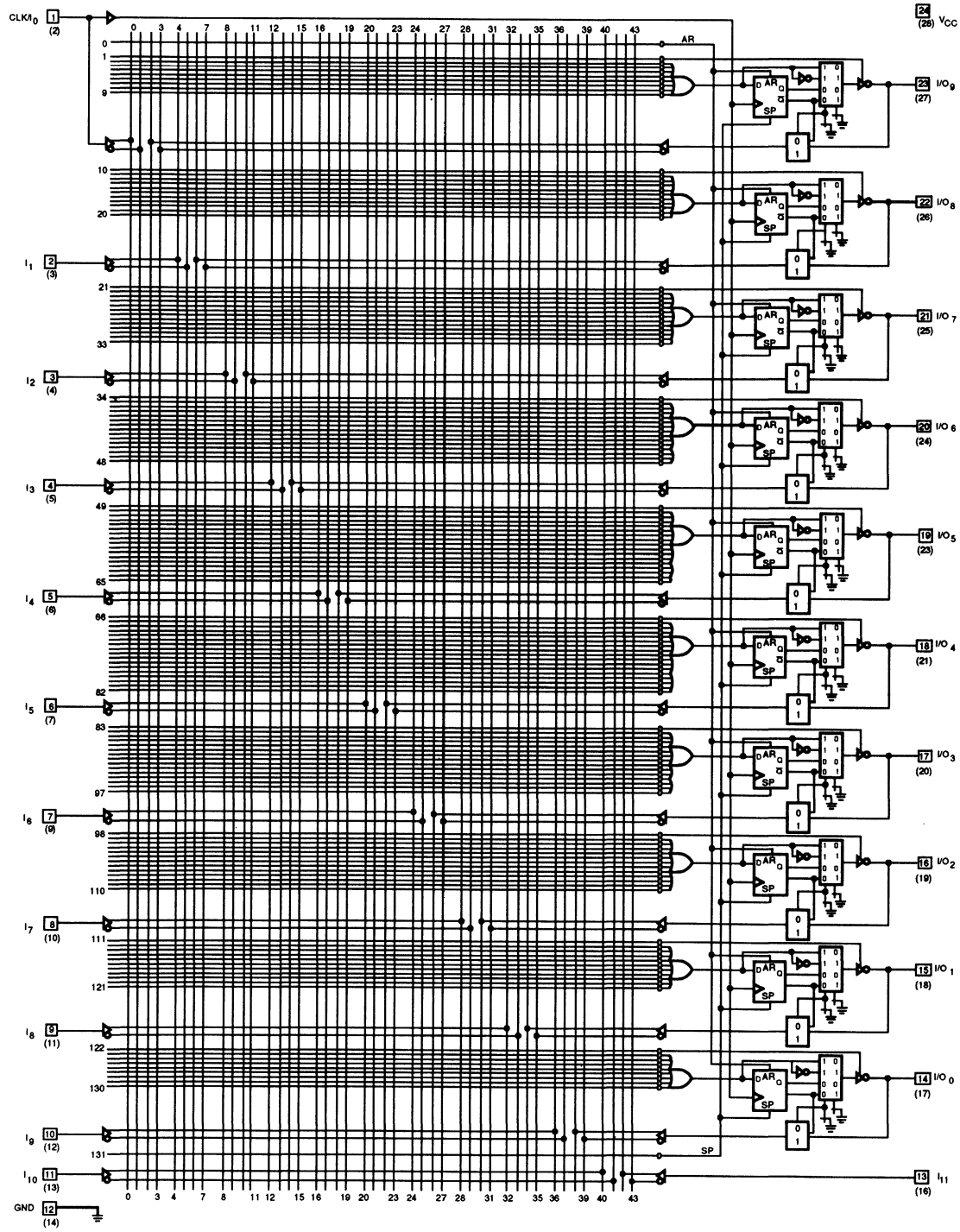
The erasability of the CMOS PALLV22V10Z allows direct testing of the device array to guarantee 100% programming and functional yields.

Technology

The high-speed PALLV22V10Z is fabricated with AMD's advanced electrically-erasable (EE) CMOS process. The array connections are formed with proven EE cells. Inputs and outputs are designed to be 3.3 and 5 V device compatible. This technology provides strong input-clamp diodes, output slew-rate control, and a grounded substrate for clean switching.

LOGIC DIAGRAM

SKINNYDIP (PLCC) Pinouts



ABSOLUTE MAXIMUM RATINGS

Storage Temperature ..... –65°C to +150°C

Ambient Temperature with Power Applied ..... –55°C to +125°C

Supply Voltage with Respect to Ground ..... –0.5 V to +7.0 V

DC Input Voltage ..... –0.5 V to +5.5 V

DC Output or I/O Pin Voltage ..... –0.5 V to +5.5 V

Static Discharge Voltage ..... 2001 V

Latchup Current (T_C = –40°C to +85°C) .... 100 mA

Stresses above those listed under Absolute Maximum Ratings may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to Absolute Maximum Ratings for extended periods may affect device reliability. Programming conditions may differ.

OPERATING RANGES

Industrial (I) Devices

Operating Case Temperature (T_C) ..... –40°C to +85°C

Supply Voltage (V_{CC}) with Respect to Ground ..... +3.0 V to +3.6 V

Operating Ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS over INDUSTRIAL operating ranges unless otherwise specified

Parameter Symbol	Parameter Description	Test Conditions		Min	Max	Unit
V_{OH}	Output HIGH Voltage	V_{IN} = V_{IH} or V_{IL}	I_{OH} = –2 mA	2.4		V
		V_{CC} = Min	I_{OH} = 100 µA	V_{CC}– 0.1		V
V_{OL}	Output LOW Voltage	V_{IN} = V_{IH} or V_{IL}	I_{OL} = 2 mA		0.4	V
			I_{OL} = 100 µA		0.1	V
V_{IH}	Input HIGH Voltage	Guaranteed Input Logical HIGH Voltage for all Inputs (Notes 1, 2)		2.0	5.5	V
V_{IL}	Input LOW Voltage	Guaranteed Input Logical LOW Voltage for all Inputs (Notes 1, 2)			0.8	V
I_{IH}	Input HIGH Leakage Current	V_{IN} = V_{CC}, V_{CC} = Max (Note 3)			10	µA
I_{IL}	Input LOW Leakage Current	V_{IN} = 0 V, V_{CC} = Max (Note 3)			–10	µA
I_{OZH}	Off-State Output Leakage Current HIGH	V_{OUT} = V_{CC}, V_{CC} = Max V_{IN} = V_{IH} or V_{IL} (Note 3)			10	µA
I_{OZL}	Off-State Output Leakage Current LOW	V_{OUT} = 0 V, V_{CC} = Max V_{IN} = V_{IH} or V_{IL} (Note 3)			–10	µA
I_{SC}	Output Short-Circuit Current	V_{OUT} = 0.5 V, V_{CC} = Max (Note 4)		–15	–75	mA
I_{CC}	Supply Current	Outputs Open (I_{OUT} = 0 mA)	f = 0 MHz		15	µA
		V_{CC} = Max	f = 25 MHz		55	mA

Notes:

1. *These are absolute values with respect to device ground and all overshoots due to system or tester noise are included.*
2. *I/O pin leakage is the worst case of I_{IL} and I_{OZL} (or I_{IH} and I_{OZH}).*
3. *Not more than one output should be shorted at a time and duration of the short-circuit should not exceed one second. V_{OUT} = 0.5 V has been chosen to avoid test problems caused by tester ground degradation.*

CAPACITANCE (Note 1)

Parameter Symbol	Parameter Description	Test Condition		Typ	Unit
C_{IN}	Input Capacitance	V_{IN} = 2.0 V	V_{CC} = 3.3 V, T_A = 25°C, f = 1 MHz		pF
C_{OUT}	Output Capacitance	V_{OUT} = 2.0 V			

Note:

1. *These parameters are not 100% tested, but are evaluated at initial characterization and at any time the design is modified where capacitance may be affected.*

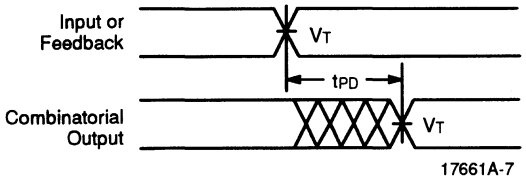
SWITCHING CHARACTERISTICS over INDUSTRIAL operating ranges (Note 2)

Parameter Symbol	Parameter Description			Min	Max	Unit
t_{PD}	Input or Feedback to Combinatorial Output (Note 3)				25	ns
t_S	Setup Time from Input, Feedback or SP to Clock			15		ns
t_H	Hold Time			0		ns
t_{CO}	Clock to Output				15	ns
t_{AR}	Asynchronous Reset to Registered Output				25	ns
t_{ARW}	Asynchronous Reset Width			25		ns
t_{ARR}	Asynchronous Reset Recovery Time			25		ns
t_{SPR}	Synchronous Preset Recovery Time			25		ns
t_{WL}	Clock Width	LOW		10		ns
t_{WH}		HIGH		10		ns
f_{MAX}	Maximum Frequency (Note 4)	External Feedback	$1/(t_S + t_{CO})$	33.3		MHz
		Internal Feedback (f_{CNT})		35.7		MHz
		No Feedback	$1/(t_{WH} + t_{WL})$	50		MHz
t_{EA}	Input to Output Enable Using Product Term Control				25	ns
t_{ER}	Input to Output Disable Using Product Term Control				25	ns

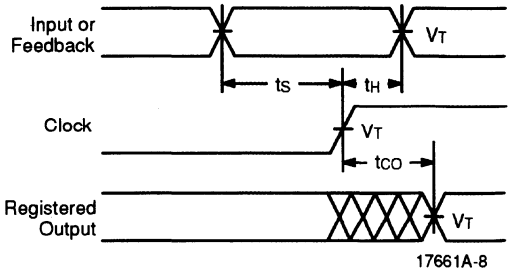
Notes:

2. *See Switching Test Circuit for test conditions.*
3. *This parameter is tested in Standby Mode. When the device is not in Standby Mode, the t_{PD} may be slightly faster.*
4. *These parameters are not 100% tested, but are evaluated at initial characterization and at any time the design is modified where frequency may be affected.*

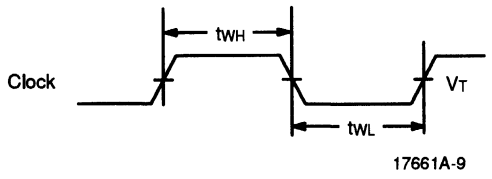
SWITCHING WAVEFORMS



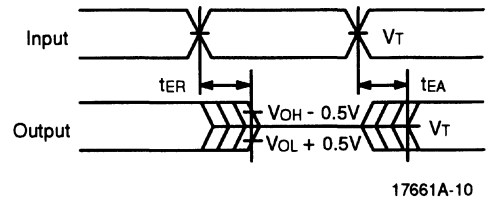
Combinatorial Output



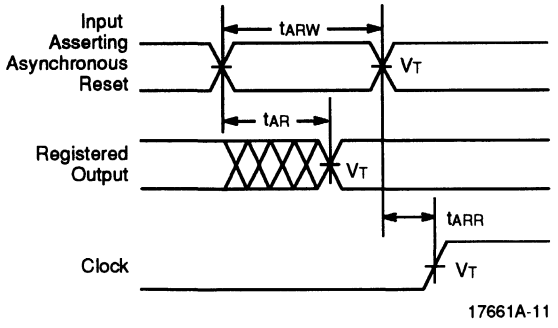
Registered Output



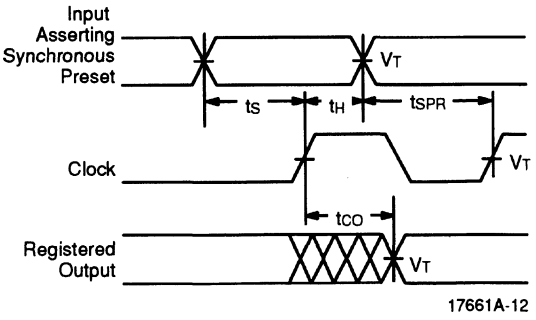
Clock Width



Input to Output Disable/Enable



Asynchronous Reset



Synchronous Preset

Notes:

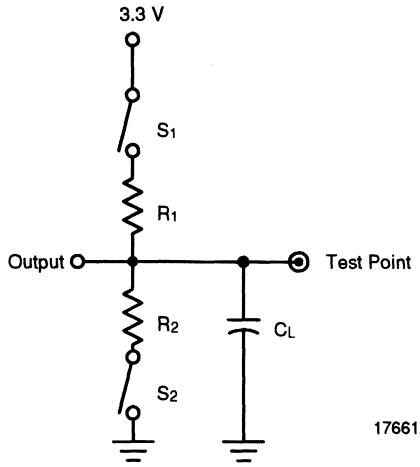
1. *V_T = 1.5 V for Input Signals and 1.65 V for Output Signals.*
2. *Input pulse amplitude 0 V to 3.0 V.*
3. *Input rise and fall times 2 ns – 5 ns typical.*

KEY TO SWITCHING WAVEFORMS

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
	Don't Care, Any Change Permitted	Changing, State Unknown
	Does Not Apply	Center Line is High-Impedance "Off" State

KS000010-PAL

SWITCHING TEST CIRCUIT



17661A-13

Specification	S_1	S_2	C_L	R_1	R_2	Measured Output Value
t_{PD}, t_{CO}	Closed	Closed	30 pF	1.6K Ω	1.6K Ω	1.65 V
t_{EA}	Z → H: Open Z → L: Closed	Z → H: Closed Z → L: Open				1.65 V
t_{ER}	H → Z: Open L → Z: Closed	H → Z: Closed L → Z: Open	5 pF			H → Z: V_{OH} – 0.5 V L → Z: V_{OL} + 0.5 V

f_{MAX} Parameters

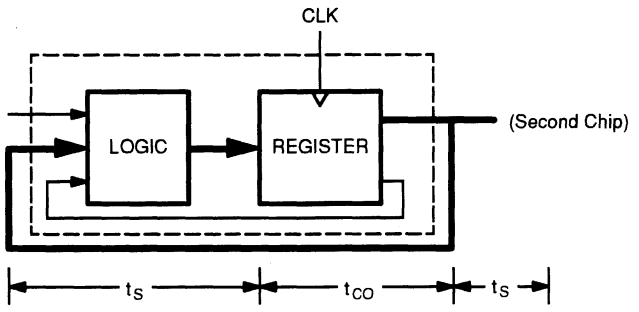
The parameter f_{MAX} is the maximum clock rate at which the device is guaranteed to operate. Because the flexibility inherent in programmable logic devices offers a choice of clocked flip-flop designs, f_{MAX} is specified for three types of synchronous designs.

The first type of design is a state machine with feedback signals sent off-chip. This external feedback could go back to the device inputs, or to a second device in a multi-chip state machine. The slowest path defining the period is the sum of the clock-to-output time and the input setup time for the external signals ($t_S + t_{CO}$). The reciprocal, f_{MAX}, is the maximum frequency with external feedback or in conjunction with an equivalent speed device. This f_{MAX} is designated "f_{MAX} external."

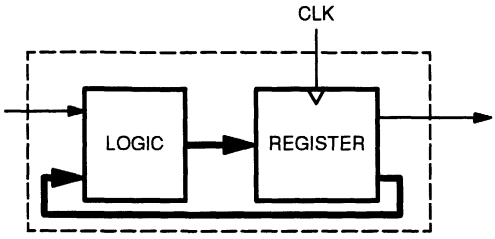
The second type of design is a single-chip state machine with internal feedback only. In this case, flip-flop inputs are defined by the device inputs and flip-flop outputs. Under these conditions, the period is limited by the internal delay from the flip-flop outputs through the internal feedback and logic to the flip-flop inputs. This f_{MAX} is designated "f_{MAX} internal". A simple internal counter is a good example of this type of design, therefore, this parameter is sometimes called "f_{CNT}."

The third type of design is a simple data path application. In this case, input data is presented to the flip-flop and clocked through; no feedback is employed. Under these conditions, the period is limited by the sum of the data setup time and the data hold time ($t_S + t_H$). However, a lower limit for the period of each f_{MAX} type is the minimum clock period ($t_{WH} + t_{WL}$). Usually, this minimum clock period determines the period for the third f_{MAX}, designated "f_{MAX} no feedback."

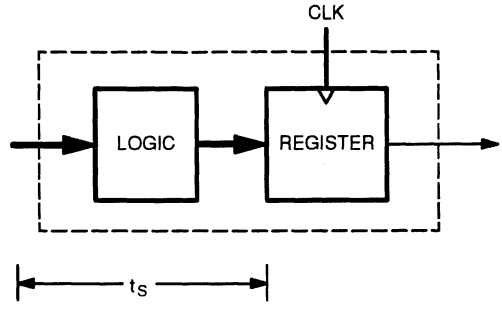
f_{MAX} external and f_{MAX} no feedback are calculated parameters. f_{MAX} external is calculated from t_S and t_{CO}, and f_{MAX} no feedback is calculated from t_{WL} and t_{WH}. f_{MAX} internal is measured.



f_{MAX} External; $1/(t_S + t_{CO})$



f_{MAX} Internal (f_{CNT})



f_{MAX} No Feedback; $1/(t_S + t_H)$ or $1/(t_{WH} + t_{WL})$

17661A-14

ENDURANCE CHARACTERISTICS

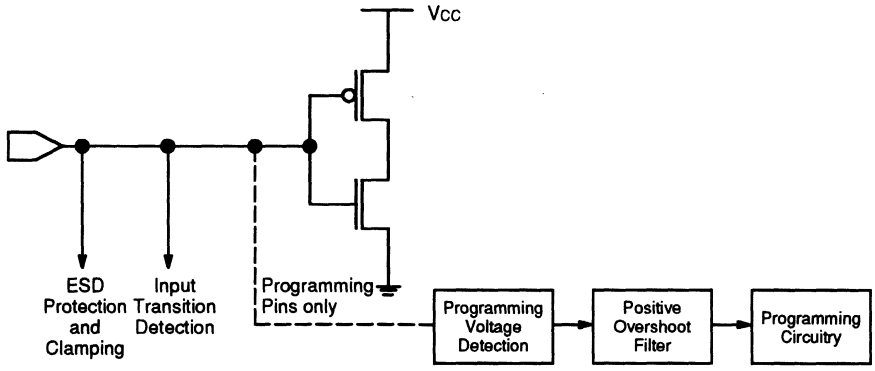
The PALLV22V10Z is manufactured using AMD's advanced Electrically Erasable process. This technology uses an EE cell to replace the fuse link used in bipolar parts. As a result, the device can be erased and reprogrammed—a feature which allows 100% testing at the factory.

Symbol	Parameter	Test Conditions	Min	Unit
t_{DR}	Min Pattern Data Retention Time	Max Storage Temperature	10	Years
		Max Operating Temperature	20	Years
N	Min Reprogramming Cycles	Normal Programming Conditions	100	Cycles

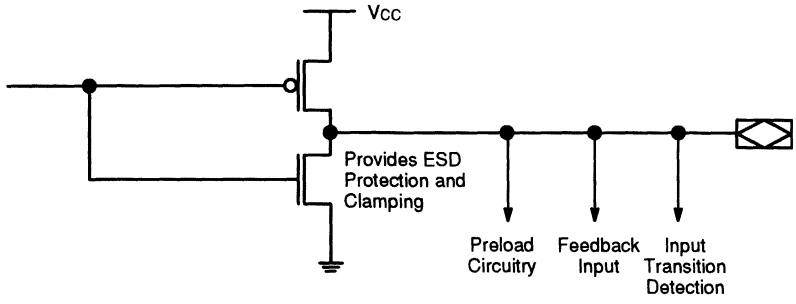
ROBUSTNESS FEATURES

The PALLV22V10Z-25 has some unique features that make it extremely robust, especially when operating in high speed design environments. Input clamping circuitry limits negative overshoot, eliminating the possibility of false clocking caused by subsequent ringing. A special noise filter makes the programming circuitry completely insensitive to any positive overshoot that has a pulse width of less than about 100 ns.

INPUT/OUTPUT EQUIVALENT SCHEMATICS



Typical Input



Typical Output

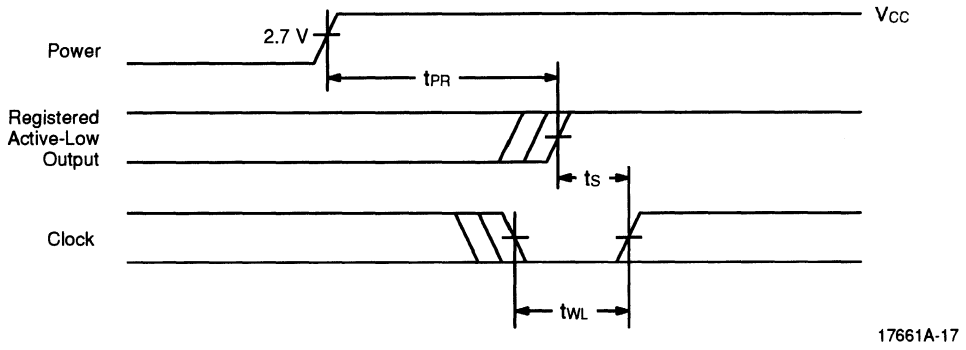
17661A-16

POWER-UP RESET

The power-up reset feature ensures that all flip-flops will be reset to LOW after the device has been powered up. The output state will depend on the programmed pattern. This feature is valuable in simplifying state machine initialization. A timing diagram and parameter table are shown below. Due to the synchronous operation of the power-up reset and the wide range of ways V_{CC} can rise to its steady state, two conditions are required to ensure a valid power-up reset. These conditions are:

- The V_{CC} rise must be monotonic.
- Following reset, the clock input must not be driven from LOW to HIGH until all applicable input and feedback setup times are met.

Parameter Symbol	Parameter Description	Max	Unit
t_{PR}	Power-Up Reset Time	1000	ns
t_S	Input or Feedback Setup Time	See Switching Characteristics	
t_{WL}	Clock Width LOW		



Power-Up Reset Waveform

DEVELOPMENT SYSTEMS (subject to change)

For more information on the products listed below, please consult the AMD FusionPLD Catalog.

MANUFACTURER	COMPILERS
Advanced Micro Devices, Inc. 901 Thompson Place MS 1028 Sunnyvale, CA 94088-3543 (800) 222-9323 or (408) 732-2400	PALASM® Software Rev. 2.2 or later
Cadence (Valid) Design Systems, Inc. 555 River Oaks Parkway San Jose, CA 95134 (408) 943-1234	Contact Cadence
Capilano Computing Systems, Ltd. 960 Quayside Dr., Suite 406 New Westminster, B.C. Canada V3M 6G2 (800) 444-9064 or (604) 522-6200	Contact Capilano
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 332-8246 or (206) 881-6444	ABEL™-4 Software Rev. 2.0 or later
ISDATA GmbH Daimlerstr. 51 D-7500 Karlsruhe 21 Germany 0721/75 10 87 or (408) 373-7359 (U.S.)	LOG/iC™ Software
Logical Devices, Inc. 1201 E. Northwest 65th Pl. Fort Lauderdale, FL 33309 (800) 331-7766 or (305) 974-0967	CUPL™ Software Rev. 2.11 or later
Mentor Graphics Corp. 8005 S.W. Beckman Rd. Wilsonville, OR 97070-7777 (800) 345-2308	Contact Mentor Graphics
MINC Incorporated 6755 Earl Drive, Suite 200 Colorado Springs, CO 80918 (719) 590-1155	PLDesigner® Software
OrCAD 3175 N.W. Aloclek Dr. Hillsboro, OR 97124 (503) 690-9881	Programmable Logic Design Tools
Viewlogic Systems, Inc. 293 Boston Post Road West Marlboro, MA 01752 (800) 422-4660 or (508) 480-0881	Contact Viewlogic
MANUFACTURER	**SCHEMATIC EDITORS AND LIBRARIES**
OrCAD 3175 N.W. Aloclek Dr. Hillsboro, OR 97124 (503) 690-9881	Contact OrCAD
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 332-8246 or (206) 881-6444	Contact Data I/O

DEVELOPMENT SYSTEMS (subject to change)

For more information on the products listed below, please consult the AMD FusionPLD Catalog.

MANUFACTURER	SIMULATORS
ALDEC Company, Inc. 3525 Old Conejo Rd., Suite 111 Newbury Park, CA 91320 (805) 499-6867	Contact ALDEC
Cadence (Valid) Design Systems, Inc. 555 River Oaks Parkway San Jose, CA 95134 (408) 943-1234	Contact Cadence
iNt GmbH Bunsenstrasse 6 D-8033 Martinsreid/Munich Germany (89) 857-6667	Contact iNt
Logic Automation Inc. 19500 NW Gibbs Dr. P.O. Box 310 Beaverton, OR 97075 (503) 690-6900	Contact Logic Automation
OrCAD 3175 N.W. Aloclek Dr. Hillsboro, OR 97124 (503) 690-9881	Contact OrCAD
Viewlogic Systems, Inc. 293 Boston Post Road West Marlboro, MA 01752 (800) 422-4660 or (508) 480-0881	Contact Viewlogic
MANUFACTURER	**TEST GENERATION SYSTEM**
Acugen Software, Inc. 427-3 Amherst St., Suite 391 Nashua, NH 03063 (603) 891-1995	ATGEN™ Software
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 332-8246 or (206) 881-6444	PLDtest™ Plus Software
iNt GmbH Bunsenstrasse 6 D-8033 Martinsreid/Munich Germany (89) 857-6667	Contact iNt

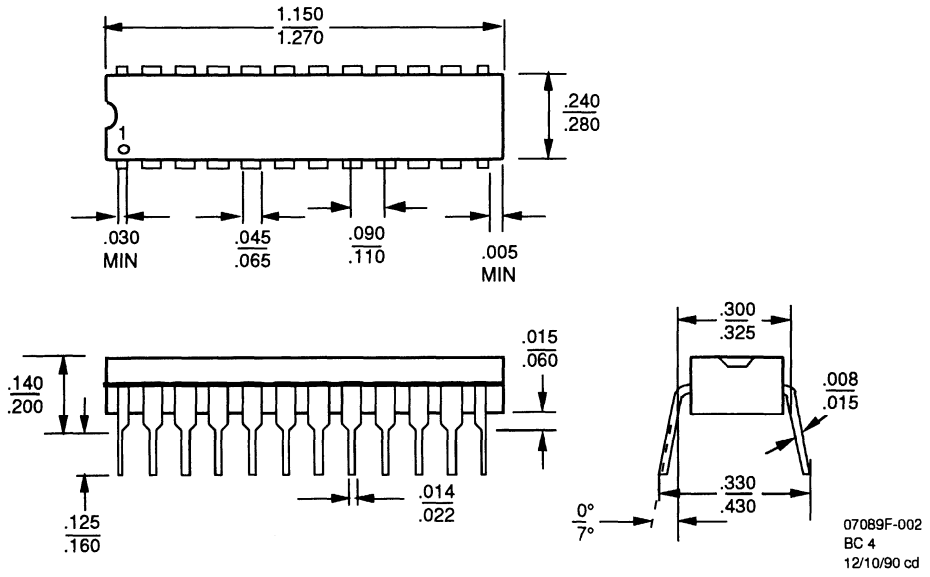
Advanced Micro Devices is not responsible for any information relating to the products of third parties. The inclusion of such information is not a representation nor an endorsement by AMD of these products.

APPROVED PROGRAMMERS (subject to change)

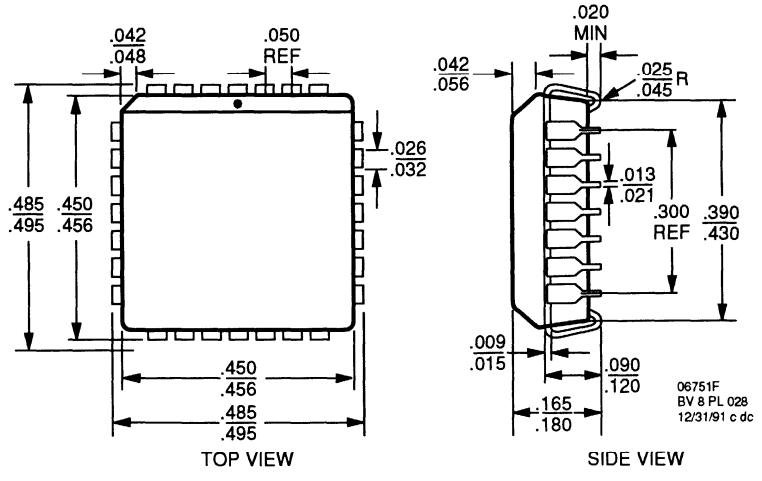
For more information on the products listed below, please consult the AMD FusionPLD Catalog.

Manufacturer	Programmer Configuration
Advin Systems, Inc. 1050-L East Duane Avenue Sunnyvale, CA 94086 (408) 243-7000	U40 Rev. 10.36 U84 Rev. 10.36
BP Microsystems 10681 Haddington, Suite #190 Houston, TX 77043 (800) 225-2102 or (713) 461-9430	PLD-1128 Rev. 1.55
Data I/O 10525 Willows Road N.E. P.O. Box 97046 Redmond, WA 98073-9746 (800) 332-8246 or (206) 881-6444	UniSite™ DIP: Rev. 3.4 PLCC: Rev. 3.4 Family/Pinout Codes: 80-E0
Logical Devices, Inc. 1201 E. Northwest 65th Pl. Fort Lauderdale, FL 33309 (800) 331-7766 or (305) 974-0967	Contact Logical Devices
Microprоss Parc d'Activite des Pres 5, rue Denis-Papin 59650 Villeneuve-d'Ascq, France (20) 47.90.40	Contact Microprоss
SMS North America, Inc. 16552 NE 135th Pl. Redmond, WA 98052 (800) 722-4122 or (206) 883-8447 or SMS Im Morgental 13 D-8994 Hergatz, Germany 07522-5018	Contact SMS
Stag Microsystems 1600 Wyatt Dr. Suite 3 Santa Clara, CA 95054 (408) 988-1118 or Stag House Martinfield, Welwyn Garden City Herfordshire UK AL7 1JT 707-332148	Contact Stag
System General Corp. 510 S. Park Victoria Dr. Milpitas, CA 95035 (408) 263-6667 or System General Corp. 3F, No. 1, Alley 8, Lane 45 Bao Shing Rd., Shin Diau Taipei, Taiwan 2-917-3005	Turpro-1 Rev. 1.50

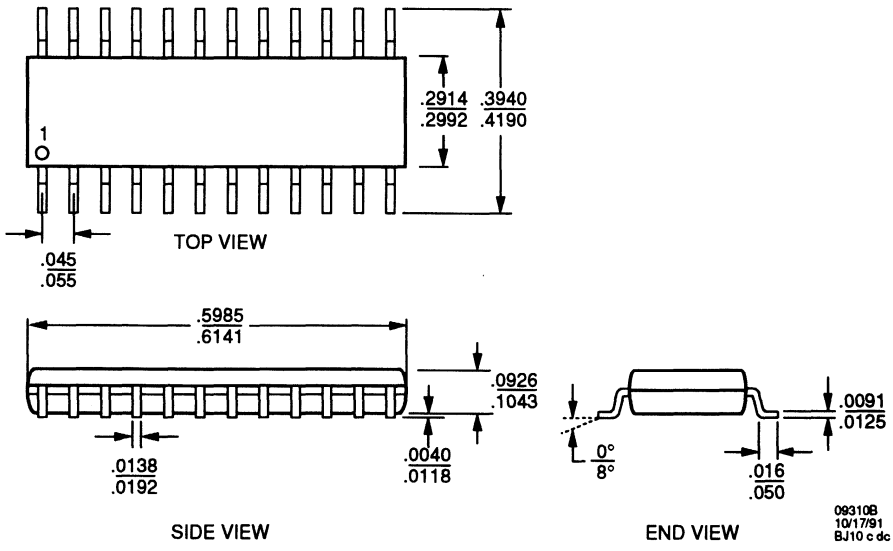
PHYSICAL DIMENSIONS*
PD3024
24-Pin 300-mil Plastic SKINNYDIP (measured in inches)



PL 028
28-Pin Plastic Leaded Chip Carrier (measured in inches)



PHYSICAL DIMENSIONS*
SO 024
24-Pin Plastic Gull-Wing Small Outline Package (measured in inches)



For reference only. BSC is an ANSI standard for Basic Space Centering.

Sales Offices

North American

ALABAMA		(205) 882-9122
ARIZONA		(602) 242-4400
CALIFORNIA,		
Culver City		(310) 645-1524
Newport Beach		(714) 752-6262
Sacramento(Roseville)		(916) 786-6700
San Diego		(619) 560-7030
San Jose		(408) 452-0500
Woodland Hills		(818) 992-4155
CANADA, Ontario,		
Kanata		(613) 592-0060
Willowdale		(416) 222-7800
COLORADO		(303) 741-2900
CONNECTICUT		(203) 264-7800
FLORIDA,		
Clearwater		(813) 530-9971
Ft. Lauderdale		(407) 361-0050
Orlando (Longwood)		(407) 862-9292
GEORGIA		(404) 449-7920
IDAHO		(208) 377-0393
ILLINOIS,		
Chicago (Itasca)		(708) 773-4422
Naperville		(708) 505-9517
MARYLAND		(301) 381-3790
MASSACHUSETTS		(617) 273-3970
MINNESOTA		(612) 938-0001
NEW JERSEY,		
Cherry Hill		(609) 662-2900
Parsippany		(201) 299-0002
NEW YORK,		
Brewster		(914) 279-8323
Rochester		(716) 425-8050
NORTH CAROLINA		
Charlotte		(704) 875- 3091
Raleigh		(919) 878- 8111
OHIO,		
Columbus (Westerville)		(614) 891-6455
Dayton		(513) 439-0268
OREGON		(503) 245-0080
PENNSYLVANIA		(215) 398-8006
TEXAS,		
Austin		(512) 346-7830
Dallas		(214) 934-9099
Houston		(713) 376-8084

International

BELGIUM, Antwerpen	TEL	(03) 248 43 00
	FAX	(03) 248 46 42
FRANCE, Paris	TEL	(1) 49-75-10-10
	FAX	(1) 49-75-10-13
GERMANY,		
Bad Homburg	TEL	(06172)-24061
	FAX	(06172)-23195
München	TEL	(089) 45053-0
	FAX	(089) 406490
HONG KONG,	TEL	(852) 865-4525
Wanchai	FAX	(852) 865-4335
ITALY, Milano	TEL	(02) 3390541
	FAX	(02) 38103458
JAPAN,		
Atsugi	TEL	(0462) 29-8460
	FAX	(0462) 29-8458
Kanagawa	TEL	(0462) 47-2911
	FAX	(0462) 47-1729
Tokyo	TEL	(03) 3346-7550
	FAX	(03) 3342-5196
Osaka	TEL	(06) 243-3250
	FAX	(06) 243-3253
KOREA, Seoul	TEL	(82) 2-784-0030
	FAX	(82) 2-784-8014

International (Continued)

LATIN AMERICA,		
Ft. Lauderdale	TEL	(305) 484-8600
	FAX	(305) 485-9736
SINGAPORE	TEL	(65) 3481188
	FAX	(65) 3480161
SWEDEN,		
Stockholm area	TEL	(08) 98 61 80
(Bromma)	FAX	(08) 98 09 06
TAIWAN, Taipei	TEL	(886) 2-7153536
	FAX	(886) 2-7122183
UNITED KINGDOM,		
Manchester area	TEL	(0925) 830380
(Warrington)	FAX	(0925) 830204
London area	TEL	(0483) 740440
(Woking)	FAX	(0483) 756196

North American Representatives

CANADA	
Burnaby, B.C. – DAVETEK MARKETING	(604) 430-3680
Kanata, Ontario – VITEL ELECTRONICS	(613) 592-0060
Mississauga, Ontario – VITEL ELECTRONICS	(416) 564-9720
Lachine, Quebec – VITEL ELECTRONICS	(514) 636-5951
ILLINOIS	
Skokie – INDUSTRIAL REPRESENTATIVES,INC	(708) 967-8430
INDIANA	
Huntington – ELECTRONIC MARKETING CONSULTANTS, INC	(317) 921-3450
Indianapolis – ELECTRONIC MARKETING CONSULTANTS, INC	(317) 921-3450
IOWA	
LORENZ SALES	(319) 377-4666
KANSAS	
Merriam – LORENZ SALES	(913) 469-1312
Wichita – LORENZ SALES	(316) 721-0500
KENTUCKY	
ELECTRONIC MARKETING CONSULTANTS, INC	(317) 921-3452
MICHIGAN	
Holland – COM-TEK SALES, INC	(616) 335-8418
Brighton – COM-TEK SALES, INC	(313) 227-0007
MINNESOTA	
Mel Foster Tech. Sales, Inc.	(612) 941-9790
MISSOURI	
LORENZ SALES	(314) 997-4558
NEBRASKA	
LORENZ SALES	(402) 475-4660
NEW MEXICO	
THORSON DESERT STATES	(505) 883-4343
NEW YORK	
East Syracuse – NYCOM, INC	(315) 437-8343
Hauppauge – COMPONENT CONSULTANTS, INC	(516) 273-5050
OHIO	
Centerville – DOLFUSS ROOT & CO	(513) 433-6776
Columbus – DOLFUSS ROOT & CO	(614) 885-4844
Westlake – DOLFUSS ROOT & CO	(216) 899-9370
OREGON	
ELECTRA TECHNICAL SALES, INC	(503) 643-5074
PENNSYLVANIA	
RUSSELL F. CLARK CO.,INC.	(412) 242-9500
PUERTO RICO	
COMP REP ASSOC, INC	(809) 746-6550
UTAH	
Front Range Marketing	(801) 288-2500
WASHINGTON	
ELECTRA TECHNICAL SALES	(206) 821-7442
WISCONSIN	
Brookfield – INDUSTRIAL REPRESENTATIVES,INC	(414) 789-9393

Advanced Micro Devices, Inc. 901 Thompson Place, P.O. Box 3453, Sunnyvale, CA 94088, USA
Tel: (408) 732-2400 • TWX: 910-339-9280 • TELEX: 34-6306 • TOLL FREE: (800) 538-8450
APPLICATIONS HOTLINE & LITERATURE ORDERING • TOLL FREE: (800) 222-9323 • (408) 749-5703

17028B 2/3/93
Ban-15M-3/93-0 Printed in USA