8086

16-Bit Microprocessor
iAPX86 Family

DISTINCTIVE CHARACTERISTICS

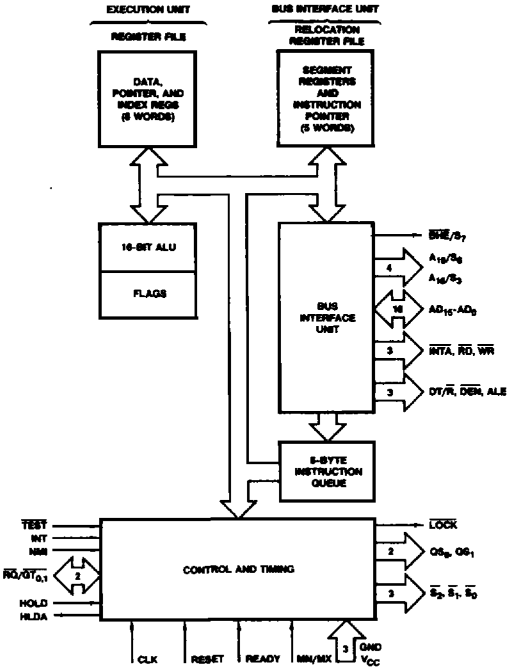
- Directly addresses up to 1 Mbyte of memory
- 24 operand addressing modes
- Efficient implementation of high level languages
- Instruction set compatible with 8080 software
- Bit, byte, word, and block operations
- 8 and 16-bit signed and unsigned arithmetic in binary or decimal
- Multibus* system interface
- Three speed options
  - 5MHz for 8086
  - 8MHz for 8086-2
  - 10MHz for 8086-1

GENERAL DESCRIPTION

The 8086 is a general purpose 16-bit microprocessor CPU. Its architecture is built around thirteen 16-bit registers and nine 1-bit flags. The CPU operates on 16-bit address spaces and can directly address up to 1 megabyte using offset addresses within four distinct memory segments, designated as code, data, stack and extra code. The 8086 implements a powerful instruction set with 24 operand addressing modes. This instruction set is compatible with that of the 8080 and 8085. In addition, the 8086 is particularly effective in executing high level languages.

The 8086 can operate in minimum and maximum modes. Maximum mode offloads certain bus control functions to a peripheral device and allows the CPU to operate efficiently in a multi-processor system. The CPU and its high performance peripherals are Multibus* compatible. The 8086 is implemented in N-channel, depletion load, silicon gate technology and is contained in a 40-pin CERDIP package, Molded DIP package, or Plastic Leaded Chip Carrier.

BLOCK DIAGRAM



BD003740

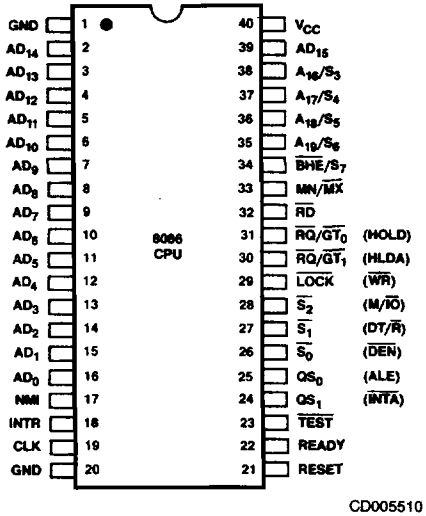
Figure 1.

Multibus is a registered trademark of Intel Corp.

01966B

Refer to page 7-1 for Essential Information on Military Devices

CONNECTION DIAGRAM
Top View
D-40-1, P-40-1

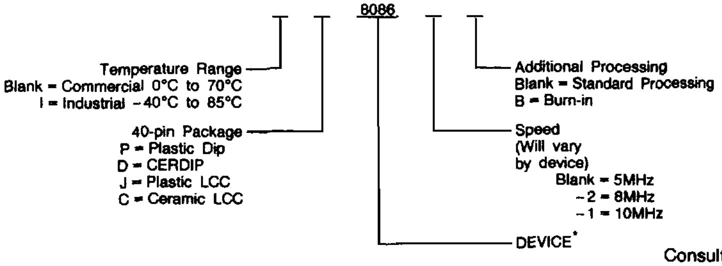


Note: Pin 1 is marked for orientation
Also available in PLCC. See Section 7 for pinout details.

Figure 2.

ORDERING INFORMATION

AMD products are available in several packages and operating ranges. The order number is formed by a combination of the following: Device number, speed option (if applicable), package type, operating range and screening option (if desired).



Valid Combinations	
8086 8086-2 8086B 8086-2B	P, D, ID, J
8086-1 8086-1B	P, D, J
8086-2 8086	/BQA

*A "C" in the middle of the device type denotes CMOS version of the product.

Valid Combinations

Consult the local AMD sales office to confirm availability of specific valid combinations, check for newly released valid combinations and/or obtain additional data on AMD's standard military grade product.

Refer to page 7-1 for Essential Information on Military Devices

PIN DESCRIPTION

The following pin function descriptions are for 8086 systems in either minimum or maximum mode. The "Local Bus" in these descriptions is the direct multiplexed bus interface connection to the 8086 (without regard to additional bus buffers).

<table>
<tr><th>Pin No.</th><th>Name</th><th>I/O</th><th>Description</th></tr>
<tr><td>39, 2-16</td><td>AD_{15}-AD_0</td><td>I/O</td><td>Address Data Bus. These lines constitute the time multiplexed memory/IO address (T_1) and data (T_2, T_3, T_W, T_4) bus. A_0 is analogous to $\overline{BHE}$ for the lower byte of the data bus, pins D_7-D_0. It is LOW during T_1 when a byte is to be transferred on the lower portion of the bus in memory or I/O operations. Eight-bit oriented devices tied to the lower half would normally use A_0 to condition chip select functions. (See $\overline{BHE}$.) These lines are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus "hold acknowledge."</td></tr>
<tr><td>35-38</td><td>A_{19}/S_6,
A_{18}/S_5,
A_{17}/S_4,
A_{16}/S_3</td><td>O</td><td>Address/Status. During T_1 these are the four most significant address lines for memory operations. During I/O operations these lines are LOW. During memory and I/O operations, status information is available on these lines during T_2, T_3, T_W, and T_4. The status of the interrupt enable FLAG bit (S_5) is updated at the beginning of each CLK cycle. A_{17}/S_4 and A_{16}/S_3 are encoded as shown.
This information indicates which relocation register is presently being used for data accessing.
These lines float to 3-state OFF during local bus "hold acknowledge."
<table>
<tr><th>A_{17}/S_4</th><th>A_{16}-S_3</th><th>Characteristics</th></tr>
<tr><td>0 (LOW)</td><td>0</td><td>Alternate Data</td></tr>
<tr><td>0</td><td>1</td><td>Stack</td></tr>
<tr><td>1 (HIGH)</td><td>0</td><td>Code or None</td></tr>
<tr><td>1</td><td>1</td><td>Data</td></tr>
<tr><td>S_6 is 0 (LOW)</td><td></td><td></td></tr>
</table></td></tr>
<tr><td>34</td><td>$\overline{BHE}/S_7$</td><td>O</td><td>Bus High Enable/Status. During T_1 the bus high enable signal ($\overline{BHE}$) should be used to enable data onto the most significant half of the data bus, pins D_{15}-D_8. Eight-bit oriented devices tied to the upper half of the bus would normally use $\overline{BHE}$ to condition chip select functions. $\overline{BHE}$ is LOW during T_1 for read, write, and interrupt acknowledge cycles when a byte is to be transferred on the high portion of the bus. The S_7 status information is available during T_2, T_3, and T_4. The signal is active LOW and floats to 3-state OFF in "hold." It is LOW during T_1 for the first interrupt acknowledge cycle.
<table>
<tr><th>$\overline{BHE}$</th><th>A_0</th><th>Characteristics</th></tr>
<tr><td>0</td><td>0</td><td>Whole word</td></tr>
<tr><td>0</td><td>1</td><td>Upper byte from/ to odd address</td></tr>
<tr><td>1</td><td>0</td><td>Lower byte from/ to even address</td></tr>
<tr><td>1</td><td>1</td><td>None</td></tr>
</table></td></tr>
<tr><td>32</td><td>$\overline{RD}$</td><td>O</td><td>Read. Read strobe indicates that the processor is performing a memory of I/O read cycle, depending on the state of the S_2 pin. This signal is used to read devices which reside on the 8086 local bus. $\overline{RD}$ is active LOW during T_2, T_3 and T_W of any read cycle and is guaranteed to remain HIGH in T_2 until the 8086 local bus has floated.
This signal floats to 3-state OFF in "hold acknowledge."</td></tr>
<tr><td>22</td><td>READY</td><td>I</td><td>READY. Is the acknowledgment from the addressed memory or I/O device that it will complete the data transfer. The READY signal from memory/IO is synchronized by the 8284A Clock Generator to form READY. This signal is active HIGH. The 8086 READY input is not synchronized. Correct operation is not guaranteed if the set-up and hold times are not met.</td></tr>
<tr><td>18</td><td>INTR</td><td>I</td><td>Interrupt Request. Is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup table located in system memory. It can be internally masked by software resetting the interrupt enable bit. INTR is internally synchronized. This signal is active HIGH.</td></tr>
<tr><td>23</td><td>$\overline{TEST}$</td><td>I</td><td>$\overline{TEST}$. Input is examined by the "Wait" instruction. If the $\overline{TEST}$ input is LOW, execution continues; otherwise, the processor waits in an "Idle" state. This input is synchronized internally during each clock cycle on the leading edge of CLK.</td></tr>
<tr><td>17</td><td>NMI</td><td>I</td><td>Non-Maskable Interrupt. An edge-triggered input which causes a type 2 interrupt. A subroutine is vectored to via an interrupt vector lookup table located in system memory. NMI is not maskable internally by software. A transition from a LOW to HIGH initiates the interrupt at the end of the current instruction. This input is internally synchronized.</td></tr>
<tr><td>21</td><td>RESET</td><td>I</td><td>Reset. Causes the processor to immediately terminate its present activity. The signal must be active HIGH for at least four clock cycles. It restarts execution, as described in the Instruction Set description, when RESET returns LOW. RESET is internally synchronized.</td></tr>
<tr><td>19</td><td>CLK</td><td>I</td><td>Clock. Provides the basic timing for the processor and bus controller. It is asymmetric with a 33% duty cycle to provide optimized internal timing.</td></tr>
<tr><td>40</td><td>V_{CC}</td><td></td><td>V_{CC}. The +5V power supply pin.</td></tr>
<tr><td>1, 20</td><td>GND</td><td></td><td>Ground. The ground pin.</td></tr>
<tr><td>33</td><td>MN/$\overline{MX}$</td><td>I</td><td>Minimum/Maximum. Indicates what mode the processor is to operate in. The two modes are discussed in the following sections.</td></tr>
</table>

PIN DESCRIPTION (Cont.)

<table>
<tr><th>Pin No.</th><th>Name</th><th>I/O</th><th>Description</th></tr>
<tr><td>28-26</td><td>$\overline{S_2}$, $\overline{S_1}$, $\overline{S_0}$</td><td>O</td><td>Status. Active during T_4, T_1, and T_2 and is returned to the passive state (1, 1, 1) during T_3 or during T_W when READY is HIGH. This status is used by the 8288 Bus Controller to generate all memory and I/O access control signals. Any change by S_2, S_1, or S_0 during T_4 is used to indicate the beginning of a bus cycle, and the return to the passive state in T_3 or T_W is used to indicate the end of a bus cycle. These signals float to 3-state OFF in "hold acknowledge." These status lines are encoded as shown.
<table>
<tr><th>$\overline{S_2}$</th><th>$\overline{S_1}$</th><th>$\overline{S_0}$</th><th>Characteristics</th></tr>
<tr><td>0 (LOW)</td><td>0</td><td>0</td><td>Interrupt Acknowledge</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Read I/O Port</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Write I/O Port</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Halt</td></tr>
<tr><td>1 (HIGH)</td><td>0</td><td>0</td><td>Code Access</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Read Memory</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Write Memory</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Passive</td></tr>
</table>
</td></tr>
<tr><td>31, 30</td><td>$\overline{RQ}/\overline{GT_0}$, $\overline{RQ}/\overline{GT_1}$</td><td>I/O</td><td>Request/Grant. Pins are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle. Each pin is bidirectional with $\overline{RQ}/\overline{GT_0}$ having higher priority than $\overline{RQ}/\overline{GT_1}$. $\overline{RQ}/\overline{GT}$ has an internal pull-up resistor so it may be left unconnected. The request/grant sequence is as follows:
1. A pulse of 1 CLK wide from another local bus master indicates a local bus request ("hold") to the 8086 (pulse 1).
2. During a T_4 or T_1 clock cycle, a pulse 1 CLK wide from the 8086 to the requesting master (pulse 2), indicates that the 8086 has allowed the local bus to float and that it will enter the "hold acknowledge" state at the next CLK. The CPU's bus interface unit is disconnected logically from the local bus during "hold acknowledge."
3. A pulse 1 CLK wide from the requesting master indicates to the 8086 (pulse 3) that the "hold" request is about to end and that the 8086 can reclaim the local bus at the next CLK.
Each master-master exchange of the local bus is a sequence of 3 pulses. There must be one dead CLK cycle after each bus exchange. Pulses are active LOW.
If the request is made while the CPU is performing a memory cycle, it will release the local bus during T_4 of the cycle when all the following conditions are met:
1. Request occurs on or before T_2.
2. Current cycle is not the low byte of a word (on an odd address).
3. Current cycle is not the first acknowledge of an interrupt acknowledge sequence.
4. A locked instruction is not currently executing.
If the local bus is idle when the request is made, two possible events will follow:
1. Local bus will be released during the next clock.
2. A memory cycle will start within 3 clocks. Now the four rules for a currently active memory cycle apply with condition number 1 already satisfied.</td></tr>
<tr><td>29</td><td>$\overline{LOCK}$</td><td>O</td><td>$\overline{LOCK}$. Output indicates that other system bus masters are not to gain control of the system bus while $\overline{LOCK}$ is active LOW. The $\overline{LOCK}$ signal is activated by the "LOCK" prefix instruction and remains active until the completion of the next instruction. This signal is active LOW, and floats to 3-state OFF in "hold acknowledge."</td></tr>
<tr><td>24, 25</td><td>QS_1, QS_0</td><td>O</td><td>Queue Status. The queue status is valid during the CLK cycle after which the queue operation is performed.
QS_1 and QS_0 provide status to allow external tracking of the internal 8086 instruction queue.</td></tr>
<tr><td>28</td><td>$M/\overline{IO}$</td><td>O</td><td>Status line. Logically equivalent to S_2 in the maximum mode. It is used to distinguish a memory access from an I/O access. M/IO becomes valid in the T_4 preceding a bus cycle and remains valid until the final T_4 of the cycle (M = HIGH, IO = LOW). $M/\overline{IO}$ floats to 3-state OFF in local bus ""hold acknowledge.""</td></tr>
<tr><td>29</td><td>$\overline{WR}$</td><td>O</td><td>Write. Indicates that the processor is performing a write memory or write I/O cycle, depending on the state of M/IO signal. WR is active for T_2, T_3 and T_W of any write cycle. It is active LOW, and floats to 3-state OFF in local bus "hold acknowledge."</td></tr>
<tr><td>24</td><td>$\overline{INTA}$</td><td>O</td><td>$\overline{INTA}$. Is used as a read strobe for interrupt acknowledge cycles. It is active LOW during T_2, T_3 and T_W of each interrupt acknowlegde cycle.</td></tr>
<tr><td>25</td><td>ALE</td><td>O</td><td>Address Latch Enable. Provided by the processor to latch the address into 8282/8283 address latch. It is a HIGH pulse active during T_1 of any bus cycle. Note that ALE is never floated.</td></tr>
<tr><td>27</td><td>$DT/\overline{R}$</td><td>O</td><td>Data Transmit/Receive. Needed in minimum system that desires to use an 8286/8287 data bus transceiver. It is used to control the direction of data flow through the transceiver. Logically $DT/\overline{R}$ is equivalent to $\overline{S_1}$ in the maximum mode, and its timing is the same as for $M/\overline{IO}$. (T = HIGH, R = LOW.) This signal floats to 3-state OFF in local bus "hold acknowledge."</td></tr>
<tr><td>26</td><td>$\overline{DEN}$</td><td>O</td><td>Data Enable. Provided as an output enable for the 8286/8287 in a minimum system which uses the transceiver. $\overline{DEN}$ is active LOW during each memory and I/O access and for INTA cycles. For a read or $\overline{INTA}$ cycle, it is active from the middle of T_2 until the middle of T_4, while for a write cycle, it is active from the beginning of T_2 until the middle of T_4. $\overline{DEN}$ floats to 3-state OFF in local bus "hold acknowledge."</td></tr>
</table>

3

PIN DESCRIPTION (Cont.)

Pin No.	Name	I/O	Description
31, 30	HOLD, HLDA	I/O	HOLD. Indicates that another master is requesting a local bus "hold." To be acknowledged, HOLD must be active HIGH. The processor receiving the "hold" request will issue HLDA (HIGH) as an acknowledgement in the middle of a T_4 or T_1 clock cycle. Simultaneous with the issuance of HLDA, the processor will float the local bus and control lines. After HOLD is detected as being LOW, the processor will LOWer HLDA, and when the processor needs to run another cycle, it will again drive the local bus and control lines. The same rules as for $\overline{RQ}/\overline{GT}$ apply, regarding when the local bus will be released. HOLD is not asynchroneous input. External synchronization should be provided if the system cannot otherwise guarantee the set-up time.

DETAILED DESCRIPTION

The 8086 CPU is internally organized into two processing units. These two units are the Bus Interface Unit (BIU) and the Execution Unit (EU). A block diagram of this organization is shown in Figure 1.

The BIU performs instruction fetch and queuing, operand fetch and store, address relocation, and basic bus control. The EU receives operands and instructions from the BIU and processes them on a 16-bit ALU. The EU accesses memory and peripheral devices through requests to the BIU. The BIU generates physical addresses in memory using the 4 segment registers and offset values.

The BIU and EU usually operate asynchronously. This permits the 8086 to overlap execution fetch and execution. Up to 6 instruction bytes can be queued. The instruction queue acts as a FIFO buffer for instructions, from which the EU extracts instruction bytes as required.

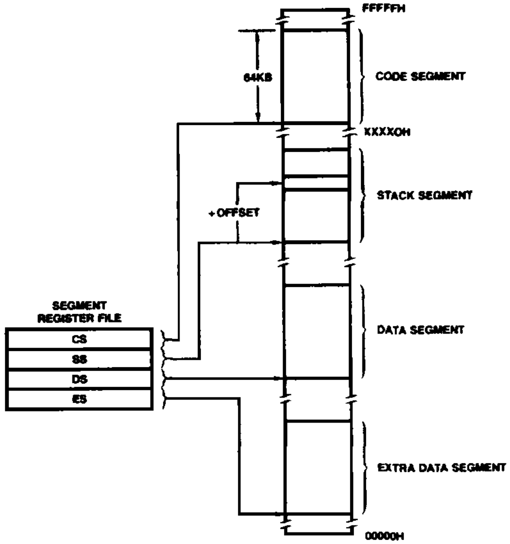
Memory Organization

The 8086 addresses up to 1 megabyte of memory. The address space is organized as a linear array, from 00000 to FFFFF in hexadecimal. Memory is subdivided into segments of 64K bytes each. There are 4 segments: code, stack, data, and extra (usually employed as an extra data segment). Each segment thus contains information of a similar type. Selection of a destination segment is automatically performed using the rules in the table below. This segmentation makes memory more easily relocatable and supports a more structured programming style.

Physical addresses in memory are generated by selecting the appropriate segment, obtaining the segment "base" address from the segment register, shifting the base address 4 digits to the left, and then adding this base to the "offset" address. For programming code, the offset address is obtained from the instruction pointer. For operands, the offset address is calculated in several ways, depending upon information contained in the addressing mode. Memory organization and address generation are shown in Figure 3a.

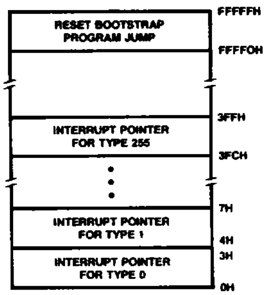
Certain memory locations are reserved for specific CPU operations. These are shown in Figure 3b. Addresses FFFF0H through FFFFFH are reserved for operations which include a jump to the initial program loading routine. After RESET, the CPU will always begin execution at location FFFF0H, where the jump must be located.

Addresses 00000H through 003FFH are reserved for interrupt operations. The service routine of each of the 256 possible interrupt types is signaled by a 4-byte pointer. The pointer elements must be stored in reserved memory addresses before the interrupts are invoked.



DF003310

Figure 3a. Memory Organization



DF003320

Figure 3b. Reserved Memory Locations

Memory Reference Need	Segment Register Used	Segment Selection Rule
Instructions	CODE (CS)	Automatic for all prefetching of instructions.
Stack	STACK (SS)	All stack pushes and pops, and all memory references relative to BP base register except data references.
Local Data	DATA (DS)	Data references which are relative to the stack, the destination of a string operation, or explicitly overriden.
External (Global) Data	EXTRA (ES)	Destination of string operations, when they are explicitly selected using a segment override.

Minimum and Maximum Modes

The 8086 has two system configurations, minimum and maximum mode. The CPU has a strap pin, MN/$\overline{MX}$, which defines the system configuration. The status of this strap pin defines the function of pin numbers 24 through 31.

When MN/$\overline{MX}$ is strapped to GND, the 8086 operates in maximum mode. The operations of pins 24 through 31 are redefined. In maximum mode, several bus timing and control functions are "off-loaded" to the 8288 bus controller, thus freeing up the CPU. The CPU communicates status information to the 8288 through pins S_0, S_1, and S_2. In maximum mode, the 8086 can operate in a multiprocessor system, using the LOCK signal within a Multibus format.

When MN/$\overline{MX}$ is strapped to V_{CC}, the 8086 operates in minimum mode. The CPU sends bus control signals itself through pins 24 through 31. This is shown in Figure 2 (in parentheses). Examples of minimum and maximum mode systems are shown in Figure 4.

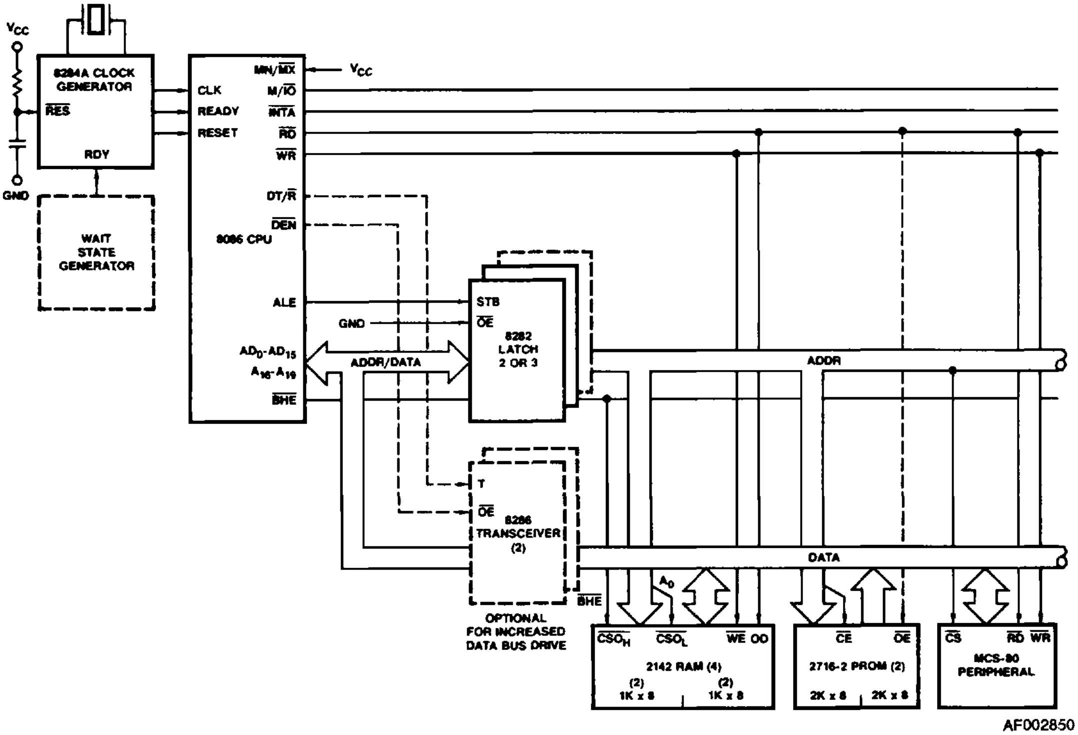


Figure 4a. Minimum Mode 8086 Typical Configuration

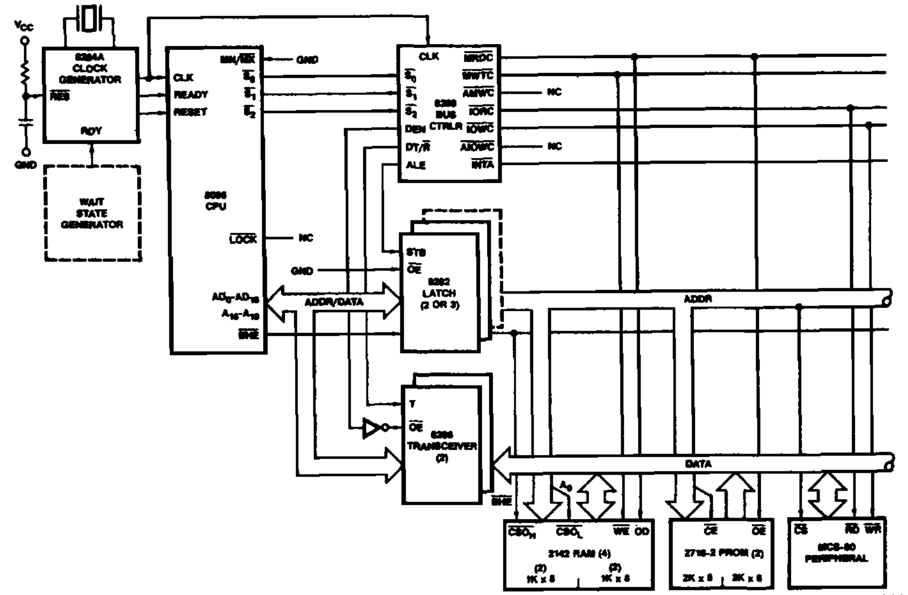


Figure 4b. Maximum Mode 8086 Typical Configuration

Bus Operation

The 8086 has a combined address and data bus, commonly referred to as "a time multiplexed bus." This technique provides the most efficient use of pins on the processor while permitting the use of a standard 40-lead package. This bus can be used throughout the system with address latching provided on memory and I/O modules. The bus can also be demultiplexed at the processor with a single set of address latches if a standard non-multiplexed bus is desired for the system.

Each bus cycle consists of at least four CLK cycles. These are referred to as T_1, T_2, T_3 and T_4 (see Figure 5). The address is sent from the processor during T_1. Data transfer occurs on the bus during T_3 and T_4. T_2 is used for changing the direction of the bus during read operations. In the event that a "NOT READY" indication is given by the addressed device, "Wait" states (T_W) are inserted between T_3 and T_4. Each inserted "Wait" state is of the same duration as a CLK cycle. "Idle" states (T_I) or inactive CLK cycles can occur between 8086 bus cycles. The processor uses these cycles for internal housekeeping.

During T_1 of any bus cycle, the ALE (Address Latch Enable) signal is emitted (by either the processor or the 8288 bus controller, depending on the MN/$\overline{\text{MX}}$ strap). At the trailing edge of this pulse, a valid address and certain status information for the cycle may be latched.

Status bits $\overline{S_0}$, $\overline{S_1}$, and $\overline{S_2}$ are used, in maximum mode, by the bus controller to identify the type of bus transaction according to the following table:

$\overline{S_2}$	$\overline{S_1}$	$\overline{S_0}$	Characteristics
0(LOW)	0	0	Interrupt Acknowledge
0	0	1	Read I/O
0	1	0	Write I/O
0	1	1	Halt
1(HIGH)	0	0	Instruction Fetch
1	0	1	Read Data from Memory
1	1	0	Write Data to Memory
1	1	1	Passive (no bus cycle)

Status bits S_3 through S_7 are multiplexed with high-order address bits and the $\overline{\text{BHE}}$ signal, and are therefore valid during T_2 through T_4. S_3 and S_4 indicate which segment register (see Instruction Set description) was used for this bus cycle in forming the address, according to the following table:

S_4	S_3	Characteristics
0(LOW)	0	Alternate Data (extra segment)
0	1	Stack
1(HIGH)	0	Code or None
1	1	Data

S_5 is a reflection of the PSW interrupt enable bit. $S_6 = 0$ and S_7 is a spare status bit.

I/O Addressing

8086 I/O operations can address up to a maximum of 64K I/O byte registers or 32K I/O word registers. The I/O address appears in the same format as the memory address on bus lines $A_{15}-A_0$. The address lines $A_{19}-A_{16}$ are zero in I/O operations. I/O instructions which use register DX as a pointer have full address capability. Direct I/O instructions directly address one or two of the 256 I/O byte locations in page 0 of the I/O address space. I/O ports are addressed in the same manner as memory locations.

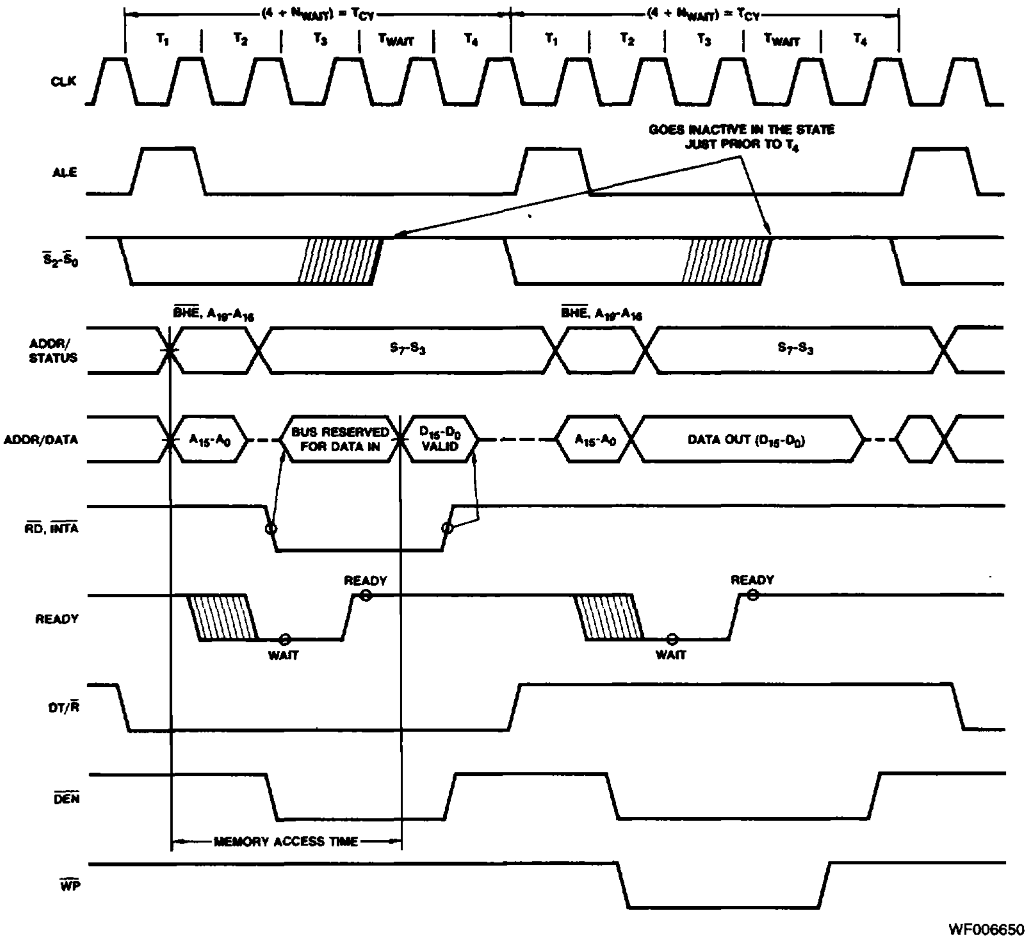


Figure 5. Basic System Timing

EXTERNAL INTERFACE

Processor Reset and Initialization

Processor initialization or start up is accomplished with activation (HIGH) of the RESET pin. The 8086 RESET is required to be HIGH for greater than 4 CLK cycles. The 8086 will terminate operations on the high-going edge of RESET and will remain dormant as long as RESET is HIGH. The low-going transition of RESET triggers an internal reset sequence for approximately 10 CLK cycles. After this interval the 8086 operates normally beginning with the instruction in absolute location FFFF0H (see Figure 3B). The details of this operation are explained in the Instruction Set description of the MCS-86 Family User's Manual. The RESET input is internally synchronized to the processor clock. At initialization the HIGH-to-LOW transition of RESET must occur no sooner than 50μs after power-up, to allow complete initialization of the 8086.

NMI may not be asserted prior to the 2nd CLK cycle following the end of RESET.

Interrupt Operations

Interrupt operations fall into two classes: software or hardware initiated. The software initiated interrupts and software aspects of hardware interrupts are described in the Instruction Set description. Hardware interrupts are either non-maskable or maskable.

Interrupts transfer control to a new program location. A 256-element table containing address pointers to the interrupt service program locations resides in absolute locations 0 through 3FFH (see Figure 3b), which are reserved for this purpose. Each element in the table is 4 bytes in size and corresponds to an interrupt "type." An interrupting device supplies an 8-bit type number during the interrupt acknowledge sequence, which is used to "vector" through the appropriate element to the new interrupt service program location.

Non-Maskable Interrupt (NMI)

The processor provides a single non-maskable interrupt pin (NMI) which has higher priority than the maskable interrupt request pin (INTR). A typical use would be to activate a power failure routine. The NMI is edge-triggered on a LOW-to-HIGH transition. The activation of this pin causes a type 2 interrupt. (See Instruction Set description.)

NMI is required to have a duration in the HIGH state of greater than two CLK cycles, but is not required to be synchronized to the clock. Any high-going transition of NMI is latched on-chip and will be serviced at the end of the current instruction or between whole moves of a block-type instruction. Worst case response to NMI would be to multiply, divide, and variable shift instructions. There is no specification on the occurrence of the low-going edge; it may occur before, during, or after the servicing of NMI. Another high-going edge triggers another response if it occurs after the start of the NMI procedure. The signal must be free of logical spikes in general and be free of bounces on the low-going edge to avoid triggering extraneous responses.

Maskable Interrupt (INTR)

The 86/10 provides a single interrupt request input (INTR) which can be masked internally by software with the resetting of the interrupt enable FLAG status bit. The interrupt request signal is level-triggered. It is internally synchronized during each clock cycle on the high-going edge of CLK. To be responded to, INTR must be present (HIGH) during the clock period preceding the end of the current instruction or the end of a whole move for a block-type instruction. During the interrupt response sequence, further interrupts are disabled. The enable bit is reset as part of the response to any interrupt (INTR, NMI, software interrupt, or single-step), although the FLAGS register, which is automatically pushed onto the stack, reflects the state of the processor prior to the Interrupt. Until the old FLAGS register is restored, the enable bit will be zero unless specifically set by an instruction.

During the response sequence (Figure 6), the processor executes two successive (back-to-back) interrupt acknowledge cycles. The 8086 emits the LOCK signal from T_2 of the first bus cycle until T_2 of the second. A local bus "hold" request will not be honored until the end of the second bus cycle. In the second bus cycle, a byte is fetched from the external interupt system (e.g., 8259A PIC) which identifies the source (type) of the interrupt. This byte is multiplied by four and used as a pointer into the interrupt vector lookup table. An INTR signal left HIGH will be continually responded to within the limitations of the enable bit and sample period. The INTERRUPT RETURN instruction includes a FLAGS pop, which returns the status of the original interrupt enable bit when it restores the FLAGS.

HALT

When a software "HALT" instruction is executed, the processor indicates that it is entering the "HALT" state in one of two ways depending upon which mode is strapped. In minimum mode, the processor issues one ALE with no qualfying bus control signals. In Maximum Mode, the processor issues appropiate HALT status on $\overline{S}_2\overline{S}_1\overline{S}_0$, and the 8288 bus controller issues one ALE. The 8086 will not leave the "HALT" state when a local bus "hold" is entered while in "HALT." In this case, the processor reissues the HALT indicator. An interrupt request or RESET will force the 8086 out of the "HALT" state.

Read/Modify/Write (Semaphore) Operation Via Lock

The $\overline{LOCK}$ status information is provided by the processor when directly consecutive bus cycles are required during the execution of an instruction. This provides the processor with the capability of performing read/modify/write operations on memory (via the Exchange Register With Memory Instruction, for example) without the possibility of another system bus master receiving intervening memory cycles. This is useful in multiprocessor system configurations to accomplish "test and set lock" operations. The $\overline{LOCK}$ signal is activated (forced LOW) in the clock cycle following the one in which the software "LOCK" prefix instruction is decoded by the EU. It is deactivated at the end of the last bus cycle of the instruction following the "LOCK" prefix instruction. While $\overline{LOCK}$ is active, a request on a RQ/GT pin will be recorded and then honored at the end of the LOCK.

External Synchronization Via Test

As an alternative to the interrupts and general I/O capabilities, the 8086 provides a single software-testable input known as the $\overline{TEST}$ signal. At any time, the program may execute a WAIT instruction. If at that time the $\overline{TEST}$ signal is inactive (HIGH), program execution becomes suspended while the processor waits for $\overline{TEST}$ to become active. It must remain active for at least 5 CLK cycles. The WAIT instruction is re-executed repeatedly until that time. This activity does not consume bus cycles. The processor remains in an idle state while waiting. All 8086 drivers go to 3-state OFF if bus "HOLD" is entered. If interrupts are enabled, they may occur while the processor is waiting. When this occurs, the processor fetches the WAIT instruction one extra time, processes the interrupt, and then re-fetches and re-executes the WAIT instruction upon returning from the interrupt.

Basic System Timing

Typical system configurations for the processor operating in minimum mode and in maximum mode are shown in Figures 4a and 4b, respectively. In minimum mode, the processor emits bus control signals in a manner similar to the 8085. In maximum mode, the processor emits coded status information which the 8288 bus controller uses to generate MULTIBUS compatible bus control signals. Figure 5 illustrates the signal timing relationships.

System Timing - Minimum System

The read cycle begins in T_1 with the assertion of the Address Latch Enable (ALE) signal. The trailing (low-going) edge of this signal is used to latch the address information, which is valid on the local bus at this time, into the 8282/8283 latch. The $\overline{BHE}$ and A_0 signals address the low, high, or both bytes. From T_1 to T_4, the M/$\overline{IO}$ signal indicates a memory or I/O operation. At T_2 the address is removed from the local bus and the bus goes to a high impedance state. The read control signal is also asserted at T_2. The read ($\overline{RD}$) signal causes the addressed device to enable its data bus drivers to the local bus. Some time later valid data will be available on the bus and the addressed device will drive the READY line HIGH. When the processor returns the read signal to a HIGH level, the addressed device will again 3-state its bus drivers. If a transceiver (8286/8287) is required to buffer the 8086 local bus, signals DT/$\overline{R}$ and $\overline{DEN}$ are provided by the 8086.

A write cycle also begins with the assertion of ALE and the emission of the address. The M/$\overline{IO}$ signal is again asserted to indicate a memory or I/O write operation. In the T_2 immediately following the address emission, the processor emits the data to be written into the addressed location. This data remains valid until the middle of T_4. During T_2, T_3, and T_W, the processor asserts the write control signal. The write ($\overline{WR}$) signal becomes active at the beginning of T_2 as opposed to the read which is delayed somewhat into T_2 to provide time for the bus to float.

The $\overline{BHE}$ and A_0 signals are used to select the proper byte(s) of the memory/IO word to be read or written according to the following table.

$\overline{BHE}$	A_0	Characteristics
0	0	Whole word
0	1	Upper byte from/to odd address
1	0	Lower byte from/to even address
1	1	None

I/O ports are addressed in the same manner as memory location. Even addressed bytes are transferred on the D_7-D_0 bus lines and odd addressed bytes on D_{15}-D_8.

The basic difference between the interrupt acknowledge cycle and a read cycle is that the interrupt acknowledge signal ($\overline{INTA}$) is asserted in place of the read ($\overline{RD}$) signal and the address bus is floated. (See Figure 6.) In the second of two successive INTA cycles, a byte of information is read from bus lines D_7-D_0 as supplied by the interrupt system logic (i.e., 8259A Priority Interrupt Controller). This byte identifies the source (type) of the interrupt. It is multiplied by four and used as a pointer into a interrupt vector lookup table, as described earlier.

Bus Timing – Medium Size Systems

For medium size systems, the MN/$\overline{MX}$ pin is connected to V_{SS}, and the 8288 Bus Controller is added to the system as well as an 8282/8283 latch for latching the system address and a 8286/8287 transceiver to allow for bus loading greater than the 8086 is capable of handling. Signals ALE, DEN, and DT/$\overline{R}$ are generated by the 8288 instead of the processor in this configuration, although their timing remains relatively the same. The 8086 status ($\overline{S}_2$, $\overline{S}_1$, and $\overline{S}_0$) provide type-of-cycle information and become 8288 inputs. This bus cycle information specifies read (code, data, or I/O), write (data or I/O), interrupt acknowledge, or software halt. The 8288 thus issues control signals specifying memory read or write, I/O read or write, or interrupt acknowledge. The 8288 provides two types of write strobes, normal and advanced, to be applied as required. The normal write strobes have data valid at the leading edge of write. The advanced write strobes have the same timing as read strobes, and hence, data isn't valid at the leading edge of write. The 8286/8287 transceiver receives the usual T and OE inputs from the 8288's DT/$\overline{R}$ and DEN.

The pointer into the interrupt vector table, which is passed during the second INTA cycle, can derive from an 8259A located on either the local bus or the system bus. If the master 8259A Priority Interrupt Controller is positioned on the local bus, a TTL gate is required to disable the 8286/8287 transceiver when reading from the master 8259A during the interrupt acknowledge sequence and software "poll."

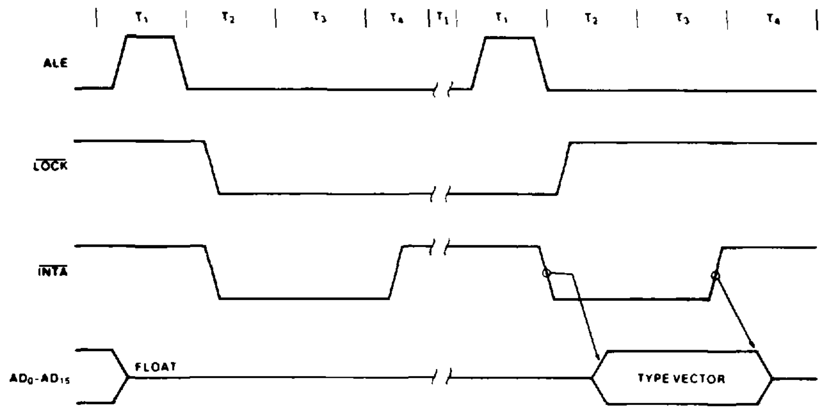


Figure 6. Interrupt Acknowledge Sequence

WF009370

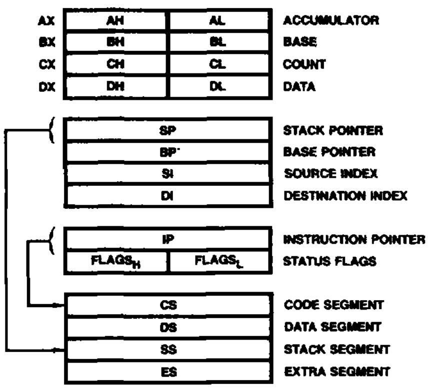


Figure 7. 8086 Register Model

DF003330

ABSOLUTE MAXIMUM RATINGS

Storage Temperature .... −65 to +150°C
Ambient Temperature Under Bias .... 0 to 70°C
Voltage on any Pin
with Respect to Ground .... −1 to +7.0V
Power Dissipation 2.5W

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

Part Number	T_A	V_{CC}
8086	0° to 70°C	5V ±10%
8086-2 8086-1	0° to 70°C	5V ±5%

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

DC CHARACTERISTICS

Parameters	Description	Test Conditions	Min	Max	Units
V_{IL}	Input Low Voltage		−0.5	+0.8	V
V_{IH}	Input High Voltage		2.0	V_{CC} + 0.5	V
V_{OL}	Output Low Voltage	I_{OL} = 2.5mA		0.45	V
V_{OH}	Output High Voltage	I_{OH} = −400μA	2.4		V
I_{CC}	Power Supply Current			340	mA
I_{LI}	Input Leakage Current	0V ≤ V_{IN} ≤ V_{CC}		±10	μA
I_{LO}	Output Leakage Current	0.45V ≤ V_{OUT} ≤ V_{CC}		±10	μA
V_{CL}	Clock Input Low Voltage		−0.5	+0.6	V
V_{CH}	Clock Input High Voltage		3.9	V_{CC} + 1.0	V
C_{IN}	Capacitance of Input Buffer (All input except AD_0-AD_{15}, $\overline{RQ}/\overline{GT}$)	fc = 1 MHz		15	pF
C_{IO}	Capacitance of I/O Buffer (AD_0-AD_{15}, $\overline{RQ}/\overline{GT}$)	fc = 1 MHz		15	pF

SWITCHING CHARACTERISTICS
MINIMUM COMPLEXITY SYSTEM TIMING REQUIREMENTS

Parameters	Description	Test Conditions	8086		8086-2		8086-1		Units
			Min	Max	Min	Max	Min	Max	
TCLCL	CLK Cycle Period		200	500	125	500	100	500	ns
TCLCH	CLK Low Time		118		68		53		ns
TCHCL	CLK High Time		69		44		39		ns
TCH1CH2	CLK Rise Time	From 1.0 to 3.5V		10		10		10	ns
TCL2CL1	CLK Fall Time	From 3.5 to 1.0V		10		10		10	ns
TDVCL	Data in Set-up Time		30		20		5		ns
TCLDX	Data in Hold Time		10		10		10		ns
TR1VCL	RDY Set-up Time into 8284A (See Notes 1, 2)		35		35		35		ns
TCLR1X	RDY Hold Time into 8284A (See Notes 1, 2)		0		0		0		ns
TRYHCH	READY Set-up Time into 8086		118		68		53		ns
TCHRYX	READY Hold Time into 8086		30		20		20		ns
TRYLCL	READY Inactive to CLK (See Note 3)		−8		−8		−10		ns
THVCH	HOLD Set-up Time		35		20		20		ns
TINVCH	INTR, NMI, $\overline{TEST}$ Set-up Time (See Note 2)		30		15		15		ns
TILIH	Input Rise Time (Except CLK)	From 0.8 to 2.0V		20		20		20	ns
TIHIL	Input Fall Time (Except CLK)	From 2.0 to 0.8V		12		12		12	ns

Notes: 1. Signal at 8284A shown for reference only.
2. Set-up requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T2 state (8ns into T3).

01966B

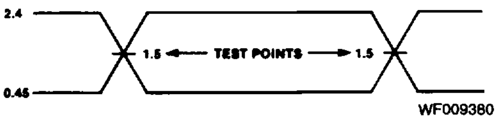
Refer to page 7-1 for Essential Information on Military Devices

SWITCHING CHARACTERISTICS (Cont.)
TIMING RESPONSES

Parameters	Description	Test Conditions	8086		8086-2		8086-1		Units
			Min	Max	Min	Max	Min	Max	
TCLAV	Address Valid Delay		10	110	10	60	10	50	ns
TCLAX	Address Hold Time		10		10		10		ns
TCLAZ	Address Float Delay		TCLAX	80	TCLAX	50	10	40	ns
TLHLL	ALE Width		TCLCH − 20		TCLCH − 10		TCLCH − 10		ns
TCLLH	ALE Active Delay			80		50		40	ns
TCHLL	ALE Inactive Delay			85		55		45	ns
TLLAX	Address Hold Time to ALE Inactive		TCHCL − 10		TCHCL − 10		TCHCL − 10		ns
TCLDV	Data Valid Delay		10	110	10	60	10	50	ns
TCHDX	Data Hold Time		10		10		10		ns
TWHDX	Data Hold Time After WR		TCLCH − 30		TCLCH − 30		TCLCH − 25		ns
TCVCTV	Control Active Delay 1		10	110	10	70	10	50	ns
TCHCTV	Control Active Delay 2	*C_L = 100pF for all 8086 Outputs (in addition to 8086 self-load)	10	110	10	60	10	45	ns
TCVCTX	Control Inactive Delay		10	110	10	70	10	50	ns
TAZRL	Address Float to READ active		0		0		0		ns
TCLRL	$\overline{RD}$ Active Delay		10	165	10	100	10	70	ns
TCLRH	$\overline{RD}$ Inactive Delay		10	150	10	80	10	60	ns
TRHAV	$\overline{RD}$ Inactive to Next Address Active		TCLCL − 45		TCLCL − 40		TCLCL − 35		ns
TCLHAV	HLDA Valid Delay		10	160	10	100	10	60	ns
TRLRH	$\overline{RD}$ Width		2TCLCL − 75		2TCLCL − 50		2TCLCL − 40		ns
TWLWH	$\overline{WR}$ Width		2TCLCL − 60		2TCLCL − 40		2TCLCL − 35		ns
TAVAL	Address Valid to ALE Low		TCLCH − 60		TCLCH − 40		TCLCH − 35		ns
TOLOH	Output Rise Time	From 0.8 to 2.0V		20		20		20	ns
TOHOL	Output Fall Time	From 2.0 to 0.8V		12		12		12	ns

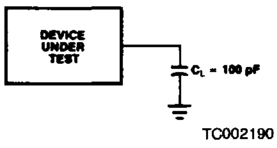
Notes: 1. Signal at 8284A shown for reference only.
2. Set-up requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T2 state (8ns into T3).

SWITCHING TEST INPUT/OUTPUT WAVEFORM



AC TESTING INPUTS ARE DRIVEN AT 2.4V FOR A LOGIC "1" AND 0.45V FOR A LOGIC "0." TIMING MEASUREMENTS ARE MADE AT 1.5V FOR BOTH A LOGIC "1" AND "0."

SWITCHING TEST LOAD CIRCUIT



C_L INCLUDES JIG CAPACITANCE

01966B

Refer to page 7-1 for Essential Information on Military Devices

SWITCHING CHARACTERISTICS (Cont.)
MAX MODE SYSTEM (USING 8288 BUS CONTROLLER)
TIMING REQUIREMENTS

Parameters	Description	Test Conditions	8086		8086-2		8086-1		Units
			Min	Max	Min	Max	Min	Max	
TCLCL	CLK Cycle Period		200	500	125	500	100	500	ns
TCLCH	CLK Low Time		118		68		53		ns
TCHCL	CLK High Time		69		44		39		ns
TCH1CH2	CLK Rise Time	From 1.0 to 3.5V		10		10		10	ns
TCL2CL1	CLK Fall Time	From 3.5 to 1.0V		10		10		10	ns
TDVCL	Data in Set-up Time		30		20		5		ns
TCLDX	Data in Hold Time		10		10		10		ns
TR1VCL	RDY Set-up Time into 8284A (See Notes 1, 2)		35		35		35		ns
TCLR1X	RDY Hold Time into 8284A (See Notes 1, 2)		0		0		0		ns
TRYHCH	READY Set-up Time into 8086		118		68		53		ns
TCHRYX	READY Hold Time into 8086		30		20		20		ns
TRYLCL	READY Inactive to CLK (See Note 4)		−8		−8		−10		ns
TINVCH	Set-up Time for Recognition (INTR, NMI, $\overline{TEST}$ (See Note 2)		30		15		15		ns
TGVCH	$\overline{RQ}/\overline{GT}$ Set-up Time		30		15		12		ns
TCHGX	$\overline{RQ}$ Hold Time into 8086		40		30		20		ns
TILIH	Input Rise Time (Except CLK)	From 0.8 to 2.0V		20		20		20	ns
TIHIL	Input Fall Time (Except CLK)	From 2.0 to 0.8V		12		12		12	ns

Notes: 1. Signal at 8284A or 8288 shown for reference only.
2. Set-up requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T3 and wait states.
4. Applies only to T2 state (8ns into T3).

01966B

Refer to page 7-1 for Essential Information on Military Devices

SWITCHING CHARACTERISTICS (Cont.)
TIMING RESPONSES

Parameters	Description	Test Conditions	8086		8086-2		8086-1		Units
			Min	Max	Min	Max	Min	Max	
TCLML	Command Active Delay (See Note 1)		10	35	10	35	10	35	ns
TCLMH	Command Inactive Delay (See Note 1)		10	35	10	35	10	35	ns
TRYHSH	READY Active to Status Passive (See Note 3)			110		65		45	ns
TCHSV	Status Active Delay		10	110	10	60	10	45	ns
TCLSH	Status Inactive Delay		10	130	10	70	10	55	ns
TCLAV	Address Valid Delay		10	110	10	60	10	50	ns
TCLAX	Address Hold Time		10		10		10		ns
TCLAZ	Address Float Delay		TCLAX	80	TCLAX	50	10	40	ns
TSVLH	Status Valid to ALE High (See Note 1)			15		15		15	ns
TSVMCH	Status Valid to MCE High (See Note 1)			15		15		15	ns
TCLLH	CLK Low to ALE Valid (See Note 1)			15		15		15	ns
TCLMCH	CLK Low to MCE High (See Note 1)			15		15		15	ns
TCHLL	ALE Inactive Delay (See Note 1)	C_L = 100pF for all 8086 Outputs (In addition to 8086 self-load)		15		15		15	ns
TCLMCL	MCE Inactive Delay (See Note 1)			15		15		15	ns
TCLDV	Data Valid Delay		10	110	10	60	10	50	ns
TCHDX	Data Hold Time		10		10		10		ns
TCVNV	Control Active Delay (See Note 1)		5	45	5	45	5	45	ns
TCVNX	Control Inactive Delay (See Note 1)		10	45	10	45	10	45	ns
TAZRL	Address Float to Read Active		0		0		0		ns
TCLRL	$\overline{RD}$ Active Delay		10	165	10	100	10	70	ns
TCLRH	$\overline{RD}$ Inactive Delay		10	150	10	80	10	60	ns
TRHAV	$\overline{RD}$ Inactive to Next Address Active		TCLCL − 45		TCLCL − 40		TCLCL − 35		ns
TCHDTL	Direction Control Active Delay (See Note 1)			50		50		50	ns
TCHDTH	Direction Control Inactive Delay (See Note 1)			30		30		30	ns
TCLGL	GT Active Delay		0	85	0	50	0	45	ns
TCLGH	GT Inactive Delay		0	85	0	50	0	45	ns
TRLRH	$\overline{RD}$ Width		2TCLCL − 75		2TCLCL − 50		2TCLCL − 40		ns
TOLOH	Output Rise Time	From 0.8 to 2.0V		20		20		20	ns
TOHOL	Output Fall Time	From 2.0 to 0.8V		12		12		12	ns

Notes: 1. Signal at 8284A or 8288 shown for reference only.
2. Set-up requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T3 and wait states.
4. Applies only to T2 state (8ns into T3).

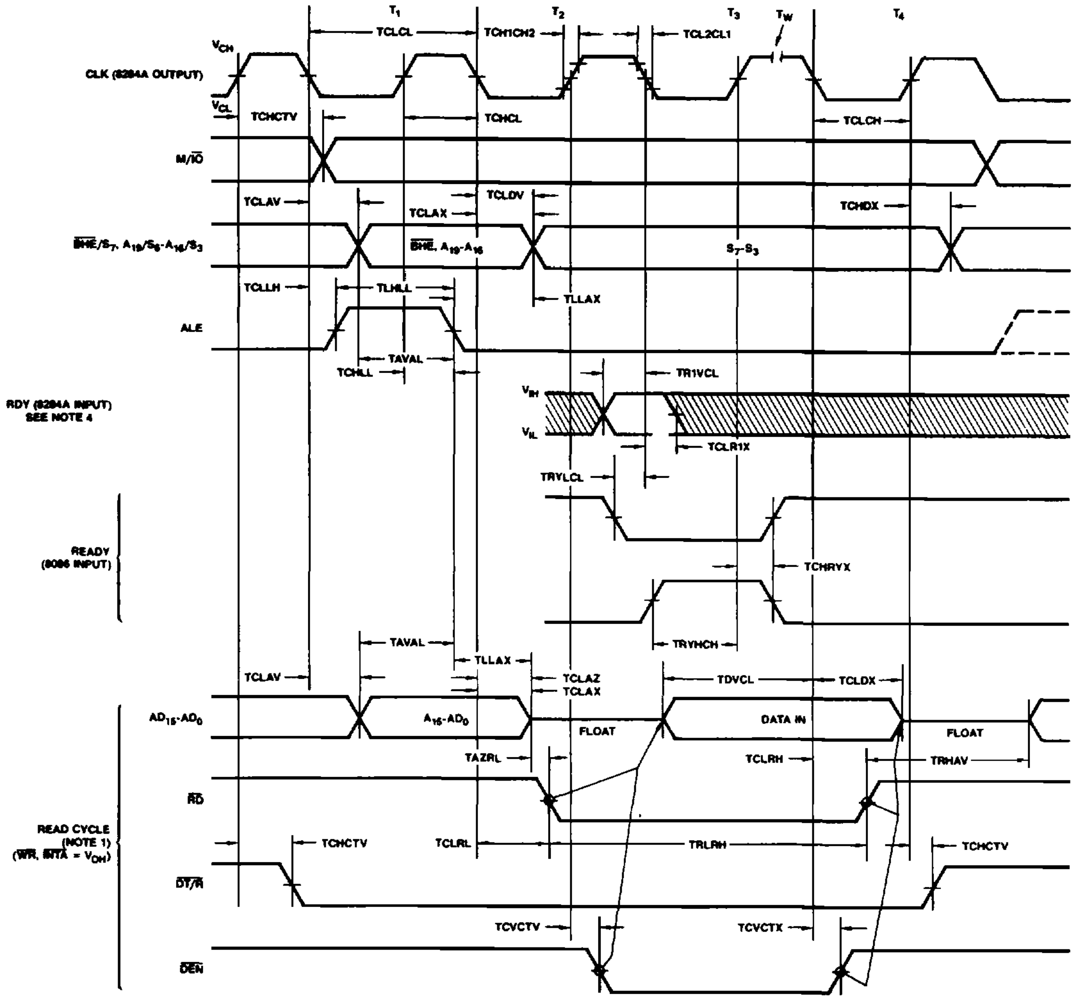
3

01966B

Refer to page 7-1 for Essential Information on Military Devices

SWITCHING WAVEFORMS

MINIMUM MODE

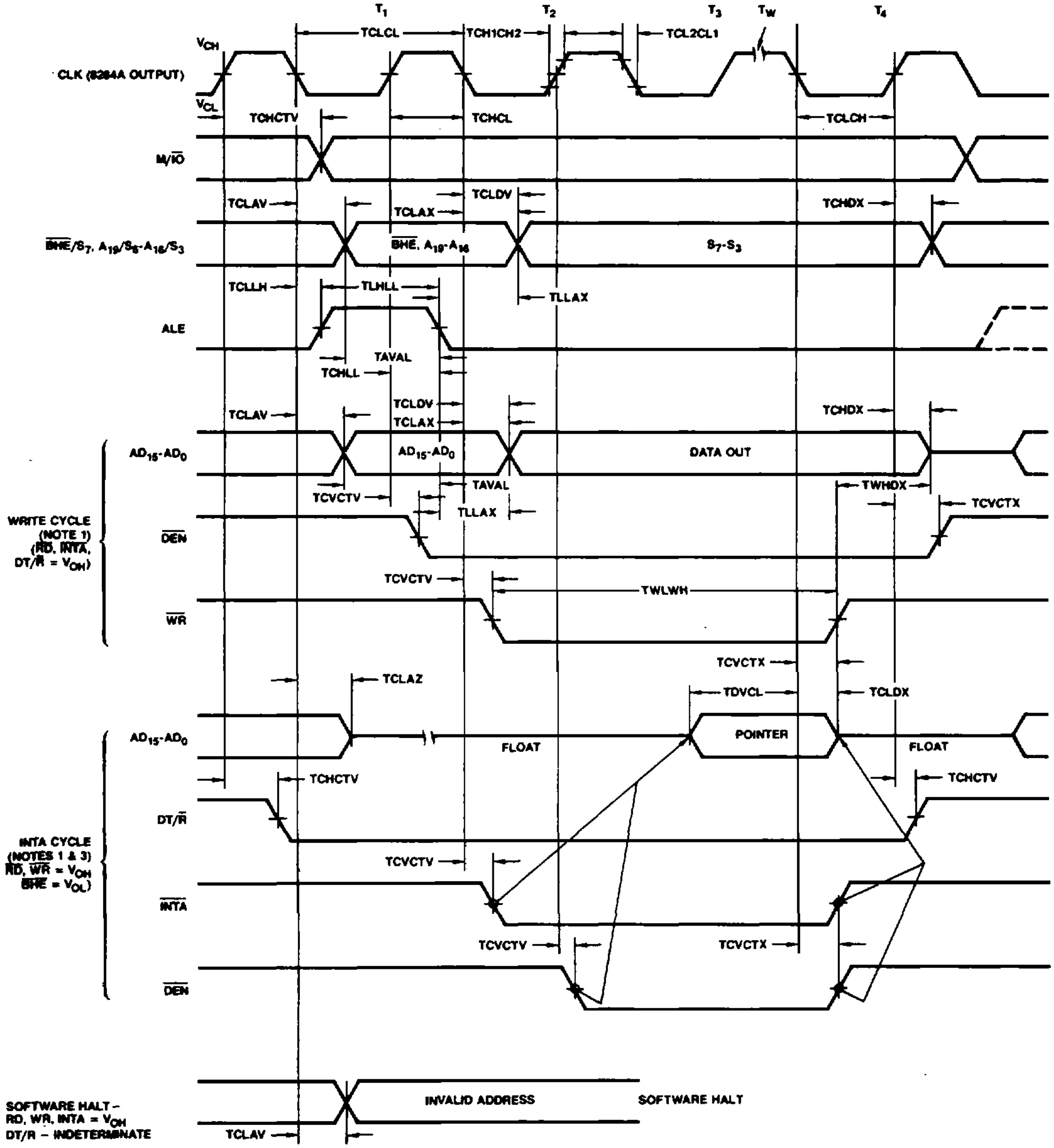


WF006660

01966B

SWITCHING WAVEFORMS (Cont.)

MINIMUM MODE



WF006670

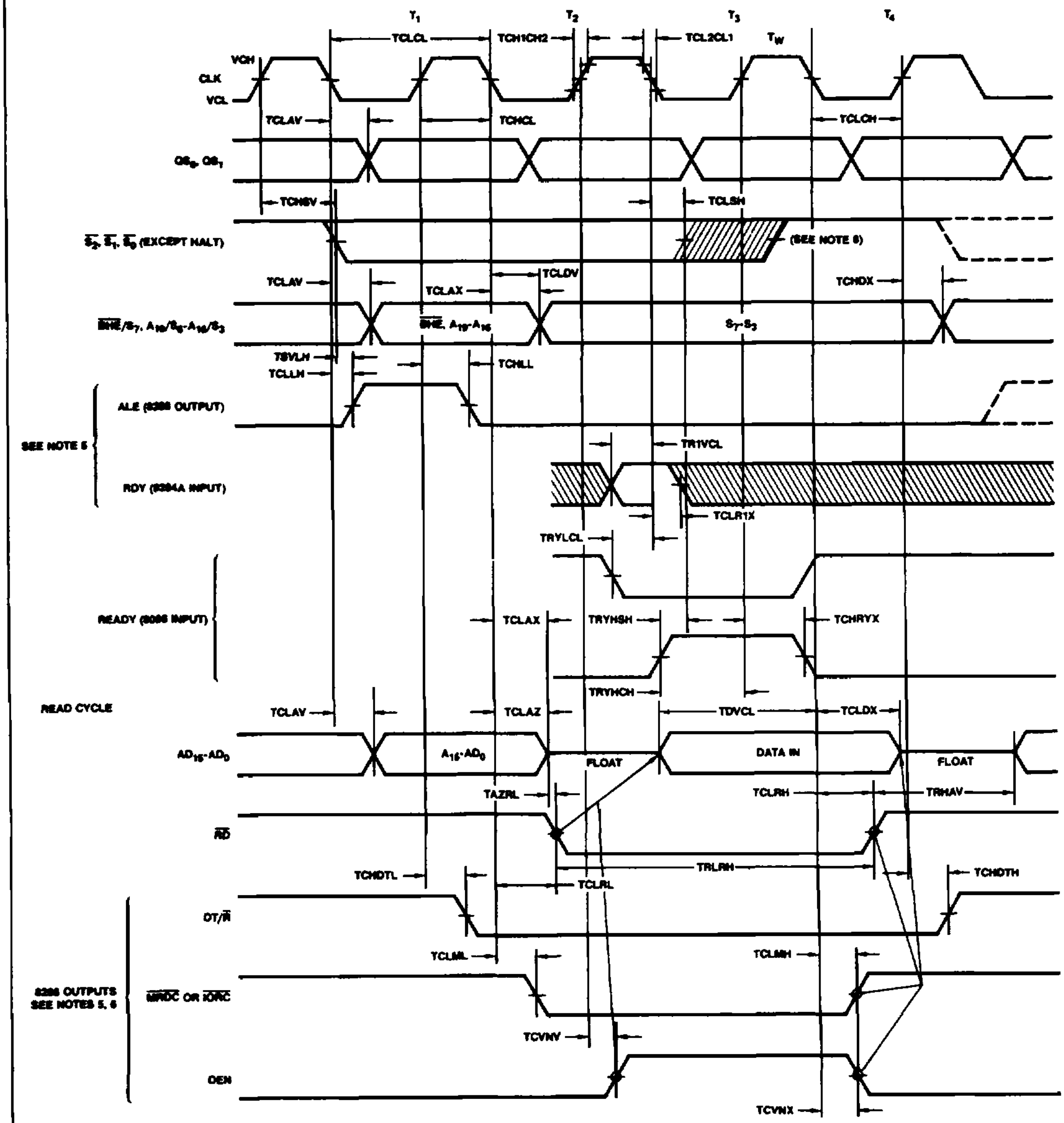
Notes: 1. All signals switch between V_{OH} and V_{OL} unless otherwise specified.
2. RDY is sampled near the end of T_2, T_3, T_W to determine if T_W machines states are to be inserted.
3. Two INTA cycles run back-to-back. The 8086 LOCAL ADDR/DATA BUS is floating during both INTA cycles. Control signals are shown for second INTA cycle.
4. Signals at 8284A are shown for reference only.
5. All timing measurements are made at 1.5V unless otherwise noted.

01966B

Refer to page 7-1 for Essential Information on Military Devices

SWITCHING WAVEFORMS (Cont.)

MAXIMUM MODE

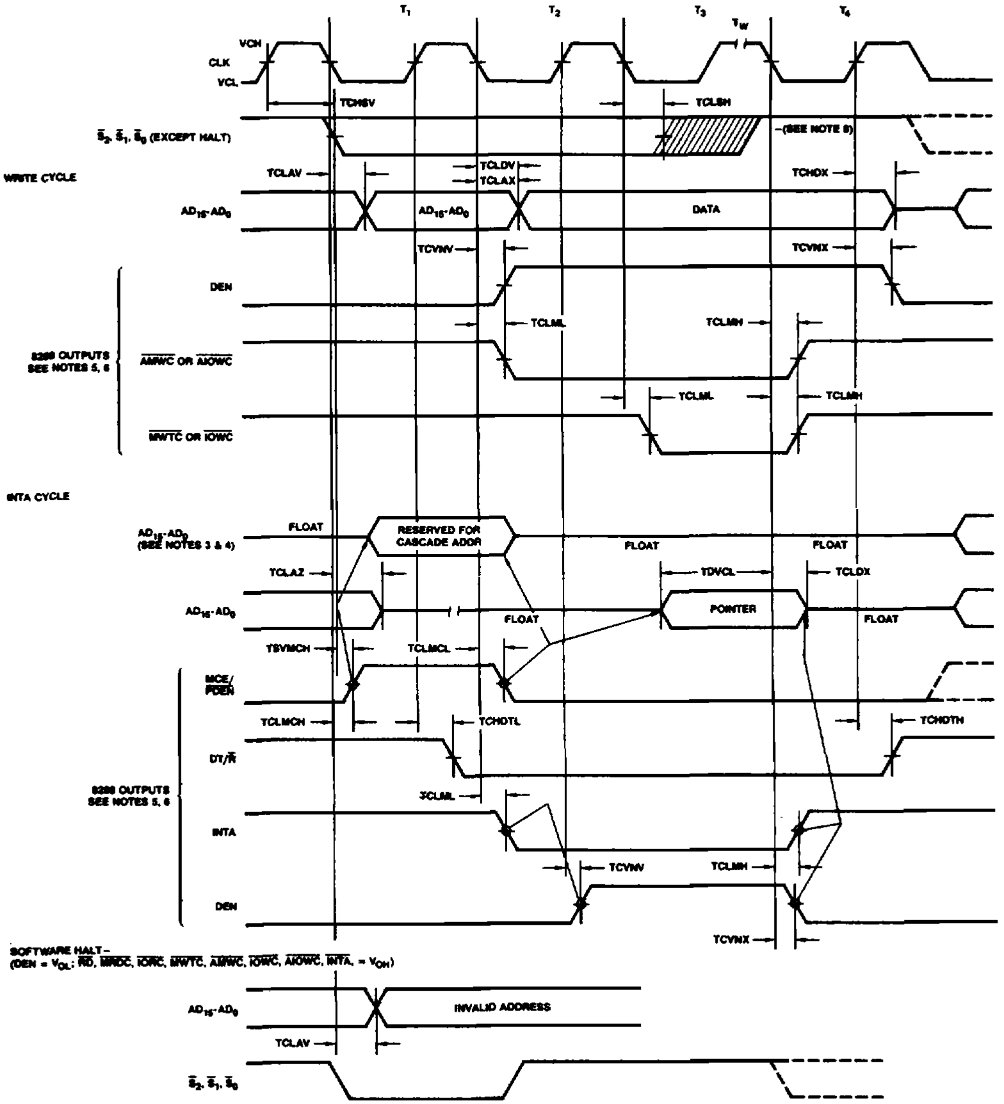


WF006680

01966B

SWITCHING WAVEFORMS (Cont.)

MAXIMUM MODE (Cont.)



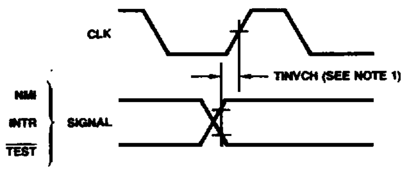
WF006730

Notes: 1. All signals switch between V_{OH} and V_{OL} unless otherwise specified.

2. RDY is sampled near the end of T_2, T_3, T_W to determine if T_W machines states are to be inserted.
3. Cascade address is valid between first and second INTA cycle.
4. Two INTA cycles run back-to-back. The 8086 LOCAL ADDR/DATA BUS is floating during both INTA cycles. Control for pointer address is shown for second INTA cycle.
5. Signals at 8284A or 8288 are shown for reference only.
6. The issuance of the 8288 command and control signals (MRDC, MWTC, AMWC, IORC, IOWC, AIOWC, INTA and DEN) lags the active high 8288 CEN.
7. All timing measurements are made at 1.5V unless otherwise noted.
8. Status inactive in state just prior to T_4.

SWITCHING WAVEFORMS (Cont.)

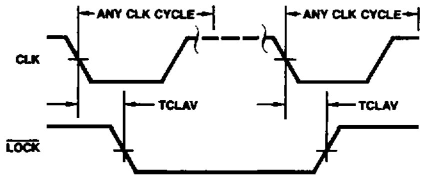
ASYNCHRONOUS SIGNAL RECOGNITION



WF006690

Note: 1. Set-up Requirements for Asynchronous signals only to guarantee recognition at next CLK.

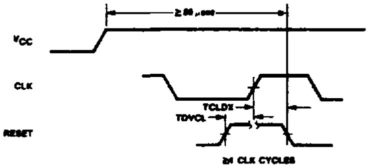
BUS LOCK SIGNAL TIMING



WF006700

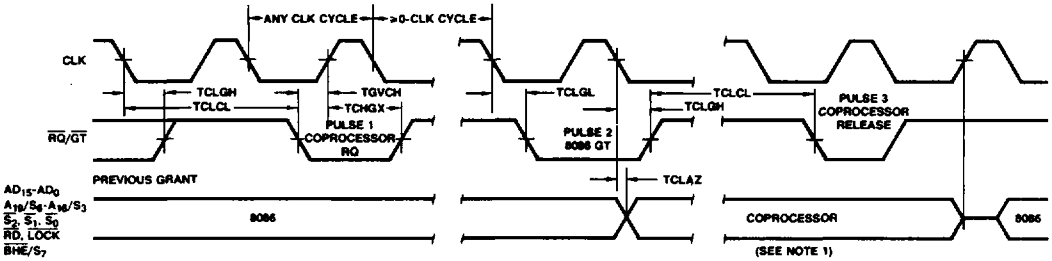
(MAXIMUM MODE ONLY)

RESET TIMING



WF009530

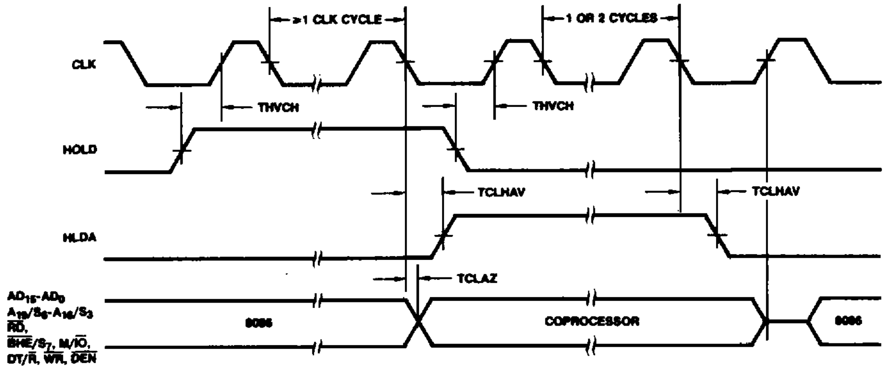
REQUEST/GRANT SEQUENCE TIMING (MAXIMUM MODE ONLY)



WF006710

Note: 1. The Coprocessor may not drive the buses outside the region shown without risking contention.

HOLD/HOLD ACKNOWLEDGE TIMING (MINIMUM MODE ONLY)



WF006720

01966B

Refer to page 7-1 for Essential Information on Military Devices

8086/8088 INSTRUCTION SET SUMMARY

DATA TRANSFER

MOV = **Move**	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
Register/memory to /from register	1 0 0 0 1 0 d w	mod reg r/m		
Immediate to register/memory	1 1 0 0 0 1 1 w	mod 0 0 0 r/m	data	data if w = 1
Immediate to register	1 0 1 1 w reg	data	data if w = 1	
Memory to accumulator	1 0 1 0 0 0 0 w	addr-low	addr-high	
Accumulator to memory	1 0 1 0 0 0 1 w	addr-low	addr-high	
Register/memory to segment register	1 0 0 0 1 1 1 0	mod 0 reg r/m		
Segment register to register/memory	1 0 0 0 1 1 0 0	mod 0 reg r/m		
PUSH = **Push:**				
Register/memory	1 1 1 1 1 1 1 1	mod 1 1 0 r/m		
Register	0 1 0 1 0 reg			
Segment register	0 0 0 reg 1 1 0			
POP = **Pop:**				
Register/memory	1 0 0 0 1 1 1 1	mod 0 0 0 r/m		
Register	0 1 0 1 1 reg			
Segment register	0 0 0 reg 1 1 1			
XCHG = **Exchange:**				
Register/memory with register	1 0 0 0 0 1 1 w	mod reg r/m		
Register with accumulator	1 0 0 1 0 reg			
IN = Input from:				
Fixed port	1 1 1 0 0 1 0 w	port		
Variable port	1 1 1 0 1 1 0 w			
OUT = Ouput to:				
Fixed port	1 1 1 0 0 1 1 w	port		
Variable port	1 1 1 0 1 1 1 w			
XLAT = Transtate byte to AL	1 1 0 1 0 1 1 1			
LEA = Load EA to register	1 0 0 0 1 1 0 1	mod reg r/m		
LDS = Load pointer to DS	1 1 0 0 0 1 0 1	mod reg r/m		
LES = Load pointer to ES	1 1 0 0 0 1 0 0	mod reg r/m		
LANF = Load AH with flags	1 0 0 1 1 1 1 1			
SANF = Store AH into flags	1 0 0 1 1 1 1 0			
PUSHF = Push flags	1 0 0 1 1 1 0 0			
POPF = Pop flags	1 0 0 1 1 1 0 1			

3

Refer to page 7-1 for Essential Information on Military Devices

INSTRUCTION SET SUMMARY (Cont.)

ARITHMETIC

	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
ADD = Add				
Reg/memory with register to either	0 0 0 0 0 0 d w	mod reg r/m		
Immediate to register / memory	1 0 0 0 0 0 s w	mod 0 0 0 r/m	data	data if s:w = 01
Immediate to accumulator	0 0 0 0 0 1 0 w	data	data if w = 1	
ADC = Add with carry:				
Reg/memory with register to either	0 0 0 1 0 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 s w	mod 0 1 0 r/m	data	data if s:w = 01
Immediate to accumulator	0 0 0 1 0 1 0 w	data	data if w = 1	
INC = Increment:				
Register/memory	1 1 1 1 1 1 1 w	mod 0 0 0 r/m		
Register	0 1 0 0 0 reg			
AAA = ASCII adjust for add	0 0 1 1 0 1 1 1			
DAA = Decimal adjust for add	0 0 1 0 0 1 1 1			
SUB = Subtract:				
Reg/memory and register to either	0 0 1 0 1 0 d w	mod reg r/m		
Immediate from register/memory	1 0 0 0 0 0 s w	mod 1 0 1 r/m	data	data if s:w = 01
Immediate from accumulator	0 0 1 0 1 1 0 w	data	data if w = 1	
SBB = Subtract with borrow:				
Reg/memory and register to either	0 0 0 1 1 0 d w	mod reg r/m		
Immediate from register/memory	1 0 0 0 0 0 s w	mod 0 1 1 r/m	data	data if s:w = 01
Immediate from accumulator	0 0 0 1 1 1 0 w	data	data if w = 1	
DEC = Decrement:				
Register/memory	1 1 1 1 1 1 1 w	mod 0 0 1 r/m		
Register	0 1 0 0 1 reg			
NEG Change sign	1 1 1 1 0 1 1 w	mod 0 1 1 r/m		
CMP = Compare:				
Register/memory with register	0 0 1 1 1 0 1 w	mod reg r/m		
Register with register/memory	0 0 1 1 1 0 0 w	mod reg r/m		
Immediate with register/memory	1 0 0 0 0 0 s w	mod 1 1 1 r/m	data	data if s:w = 01
Immediate with accumulator	0 0 1 1 1 1 0 w	data	data if w = 1	
AAS ASCII adjust for subtract	0 0 1 1 1 1 1 1			
DAS Decimal adjust for subtract	0 0 1 0 1 1 1 1			
MUL Mulitiply (unsigned)	1 1 1 1 0 1 1 w	mod 1 0 0 r/m		
IMUL Integer multiply (signed):	1 1 1 1 0 1 1 w	mod 1 0 1 r/m		
AAM ASCII adjust for multiply	1 1 0 1 0 1 0 0	0 0 0 0 1 0 1 0		
DIV Divide (unsigned):	1 1 1 1 0 1 1 w	mod 1 1 0 r/m		
IDIV Integer divide (signed)	1 1 1 1 0 1 1 w	mod 1 1 1 r/m		
AAD ASCH adjust for divide	1 1 0 1 0 1 0 1	0 0 0 0 1 0 1 0		
CBW Convert byte to word	1 0 0 1 1 0 0 0			
CWD Convert word to double word	1 0 0 1 1 0 0 1			

01966B

Refer to page 7-1 for Essential Information on Military Devices

INSTRUCTION SET SUMMARY (Cont.)

LOGIC

	76543210	76543210	76543210	76543210
NOT Invert	1 1 1 1 0 1 1 w	mod 0 1 0 r/m		
SHL/SAL Shift logical/arithmetic left	1 1 0 1 0 0 v w	mod 1 0 0 r/m		
SHR Shift logical right	1 1 0 1 0 0 v w	mod 1 1 1 r/m		
SAR Shift arithmetic right	1 1 0 1 0 0 v w	mod 1 1 1 r/m		
ROL Rotate left	1 1 0 1 0 0 v w	mod 0 0 0 r/m		
ROR Rotate right	1 1 0 1 0 0 v w	mod 0 0 1 r/m		
RCL Rotate through carry flag left	1 1 0 1 0 0 v w	mod 0 1 0 r/m		
RCR Rotate through carry right	1 1 0 1 0 0 v w	mod 0 1 1 r/m		
AND = And:				
Reg/memory and register to either	0 0 1 0 0 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 0 w	mod 1 0 0 r/m	data	data if w = 1
Immediate to accumulator	0 0 1 0 0 1 0 w	data	data if w = 1	
TEST = And function to flags, no result:				
Register/memory and register	1 0 0 0 0 1 0 w	mod reg r/m		
Immediate data and register/memory	1 1 1 1 0 1 1 w	mod 0 0 0 r/m	data	data if w = 1
Immediate data and accumulator	1 0 1 0 1 0 0 w	data	data if w = 1	
OR = Or:				
Reg/memory and register to either	0 0 0 0 1 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 0 w	mod 0 0 1 r/m	data	data if w = 1
Immediate to accumulator	0 0 0 0 1 1 0 w	data	data if w = 1	
XOR = Exclusive or:				
Reg/memory and register to either	0 0 1 1 0 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 0 w	mod 1 1 0 r/m	data	data if w = 1
Immediate to accumulator	0 0 1 1 0 1 0 w	data	data if w = 1	

STRING MANIPULATION:

REP = Repeat	1 1 1 1 0 0 1 z
MOVS = Move byte/word	1 0 1 0 0 1 0 w
CMPS = Compare byte/word	1 0 1 0 0 1 1 w
SCAS = Scan byte/word	1 0 1 0 1 1 1 w
LODS = Load byte/wd to AL/AX	1 0 1 0 1 1 0 w
STOS = Stor byte/wd from AL/A	1 0 1 0 1 0 1 w

01966B

Refer to page 7-1 for Essential Information on Military Devices

INSTRUCTION SET SUMMARY (Cont.)

CONTROL TRANSFER

	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
CALL = Call				
Direct within segment	1 1 1 0 1 0 0 0	disp-low	disp-high	
Indirect within segment	1 1 1 1 1 1 1 1	mod 0 1 0 r/m		
Direct intersegment	1 0 0 1 1 0 1 0	offset-low	offset-high	
		seg-low	seg-high	
Indirect intersegment	1 1 1 1 1 1 1 1	mod 0 1 1 r/m		
JMP = Unconditional jump:				
Direct within segment	1 1 1 0 1 0 0 1	disp-low	disp-high	
Direct within segment-short	1 1 1 0 1 0 1 1	disp		
Indirect within segment	1 1 1 1 1 1 1 1	mod 1 0 0 r/m		
Direct intersegment	1 1 1 0 1 0 1 0	offset-low	offset-high	
		seg-low	seg-high	
Indirect intersegment	1 1 1 1 1 1 1 1	mod 1 0 1 r/m		
RET = Return from CALL:				
Within segment	1 1 0 0 0 0 1 1			
Within seg adding immed to SP	1 1 0 0 0 0 1 0	data-low	data-high	
Intersegment	1 1 0 0 1 0 1 1			
Intersegment adding immediate to SP	1 1 0 0 1 0 1 0	data-low	data-high	
JE/JZ = Jump on equal/zero	0 1 1 1 0 1 0 0	disp		
JL/JNGE = Jump on less/not greater or equal	0 1 1 1 1 1 0 0	disp		
JLE/JNG = Jump on less or equal/not greater	0 1 1 1 1 1 1 0	disp		
JB/JNAE = Jump on below/not above or equal	0 1 1 1 0 0 1 0	disp		
JBE/JNA = Jump on below or equal/not above	0 1 1 1 0 1 1 0	disp		
JP/JPE = Jump on parity/parity even	0 1 1 1 1 0 1 0	disp		
JO = Jump on overflow	0 1 1 1 0 0 0 0	disp		
JS = Jump on sign	0 1 1 1 1 0 0 0	disp		
JNE/JNZ = Jump on not equal/not zero	0 1 1 1 0 1 0 1	disp		
JNL/JGE = Jump on not less/greater or equal	0 1 1 1 1 1 0 1	disp		
JNLE/JG = Jump on not less or equal/greater	0 1 1 1 1 1 1 1	disp		
JNB/JAE = Jump on not below/above or equal	0 1 1 1 0 0 1 1	disp		
JNBE/JA = Jump on not below or equal/above	0 1 1 1 0 1 1 1	disp		
JNP/JPO = Jump on not par/par odd	0 1 1 1 1 0 1 1	disp		
JNO = Jump on not overflow	0 1 1 1 0 0 0 1	disp		
JNS = Jump on not sign	0 1 1 1 1 0 0 1	disp		
LOOP = Loop CX times	1 1 1 0 0 0 1 0	disp		
LOOPZ/LOOPE = Loop while zero/equal	1 1 1 0 0 0 0 1	disp		
LOOPNZ/LOOPNE = Loop while not zero/equal	1 1 1 0 0 0 0 0	disp		
JCXZ = Jump on CX zero	1 1 1 0 0 0 1 1	disp		

Refer to page 7-1 for Essential Information on Military Devices

INSTRUCTION SET SUMMARY (Cont.)

CONTROL TRANSFER (Cont.)

Instruction	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
INT = **Interrupt**				
Type specified	1 1 0 0 1 1 0 1	type		
Type 3	1 1 0 0 1 1 0 0			
INTO = Interrupt on overflow	1 1 0 0 1 1 1 0			
IRET = Interrupt return	1 1 0 0 1 1 1 1			

PROCESSOR CONTROL

Instruction	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
CLC = Clear carry	1 1 1 1 1 0 0 0	
CMC = Complement carry	1 1 1 1 0 1 0 1	
STC = Set carry	1 1 1 1 1 0 0 1	
CLD = Clear direction	1 1 1 1 1 1 0 0	
STD = Set direction	1 1 1 1 1 1 0 1	
CLI = Clear interrupt	1 1 1 1 1 0 1 0	
STI = Set interrupt	1 1 1 1 1 0 1 1	
HLT = Halt	1 1 1 1 0 1 0 0	
WAIT = Wait	1 0 0 1 1 0 1 1	
ESC = Processor Extension Escape	1 0 0 1 1 x x x	mod x x x r/m
LOCK = Bus lock prefix	1 1 1 1 0 0 0 0	

3

Footnotes:

AL = 8-bit accumulator
AX = 16-bit accumulator
CX = Count register
DS = Data segment
ES = Extra segment
Above/below refers to unsigned value.
Greater = more positive.
Less = less positive (more negative) signed values
if d = 1 then "to" reg; if d = 0 then "from" reg
w = 1 then word instruction; if w = 0 then byte instruction

if mod = 11 then r/m is treated as a REG field
if mod = 00 then DISP = 0*, disp-low and disp-high are absent
if mod = 01 then DISP = disp-low sign-extended to 16-bits, disp-high is absent
if mod = 10 then DISP = disp-high: disp-low

if r/m = 000 then EA = (BX) + (SI) + DISP
if r/m = 001 then EA = (BX) + (DI) + DISP
if r/m = 010 then EA = (BP) + (SI) + DISP
if r/m = 011 then EA = (BP) + (DI) + DISP
if r/m = 100 then EA = (SI) + DISP
if r/m = 101 then EA = (DI) + DISP
if r/m = 110 then EA = (BP) + DISP*
if r/m = 111 then EA = (BX) + DISP

DISP follows 2nd byte of instruction (before data if required)

*except if mod = 00 and r/m = 110 then EA = disp-high: disp-low.

if s:w = 01 then 16 bits of immediate data form the operand.
if s:w = 11 then an immediate data byte is sign extended to form the 16-bit operand.
if v = 0 then "count" = 1; if v = 1 then "count" in (CL)
x = don't care
z is used for string primitives for comparison with ZF Flag.

SEGMENT OVERRIDE PREFIX

0	0	1	reg	1	1	0

REG is assigned according to the following table:

16-Bit (w = 1)	8-Bit (w = 0)	Segment
000 AX	000 AL	00 ES
001 CX	001 CL	01 CS
010 DX	010 DL	10 SS
011 BX	011 BL	11 DS
100 SP	100 AH	
101 BP	101 CH	
110 SI	110 DH	
111 DI	111 BH	

Instructions which reference the flag register files as a 16-bit object use the symbol FLAGS to represent the file:

FLAGS = X:X:X:X:(OF):(DF):(TF):(SF):(ZF):X:(AF):X:(PF):X:(CF)

Refer to page 7-1 for Essential Information on Military Devices