8086/8088 Hardware Specifications

CEN433

King Saud University

Dr. Mohammed Amer Arafah

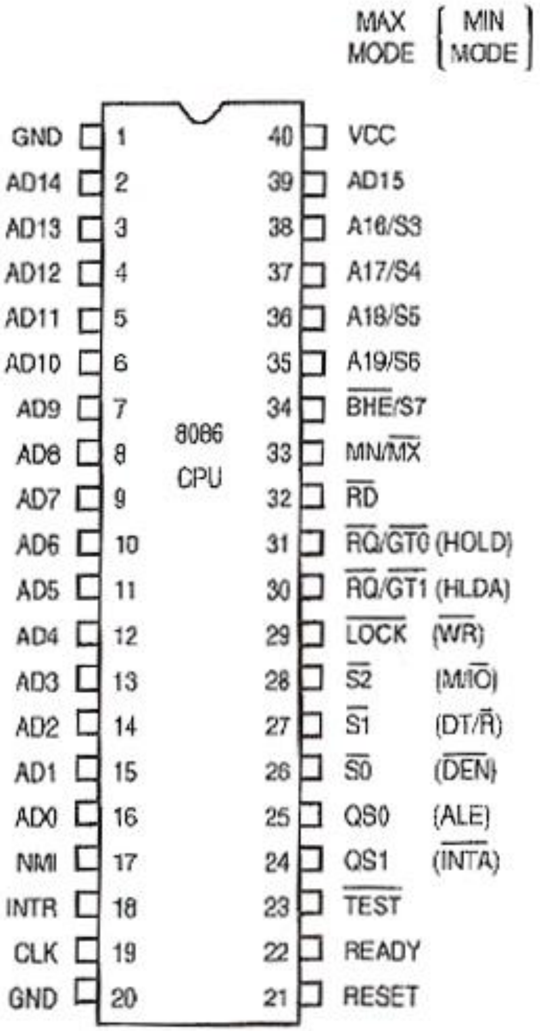
8088/8086 Microprocessors

- Fairly old microprocessors, but still considered a good way to introduce the Intel family
- Both microprocessors use 16-bit registers and 20-bit address bus (supporting 1 MB memory), but:

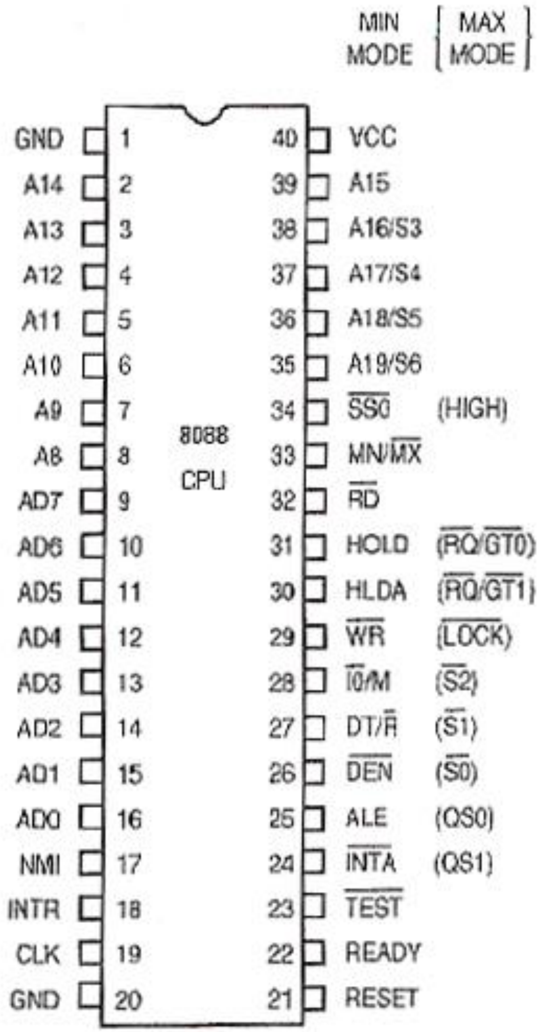
  - The 8086 (1978): 16-bit external data bus

  - The 8088 (1979): 8-bit external data bus
- Still used in embedded systems (cost is less than $1)

Pin Layout of the 8088/8086 Microprocessors

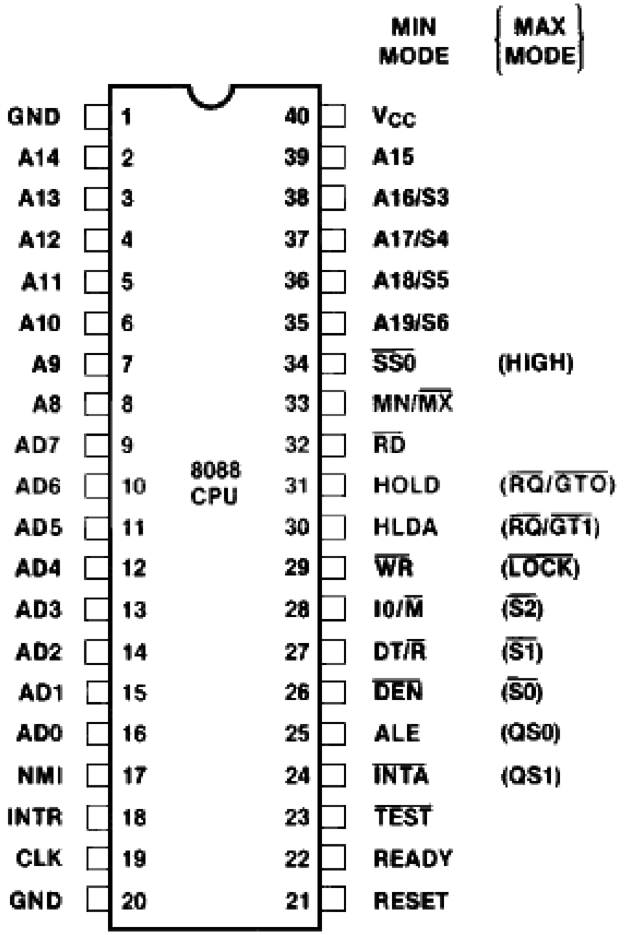

MAX MODE
[MIN MODE]
8086 CPU
GND 1
AD14 2
AD13 3
AD12 4
AD11 5
AD10 6
AD9 7
AD8 8
AD7 9
AD6 10
AD5 11
AD4 12
AD3 13
AD2 14
AD1 15
AD0 16
NMI 17
INTR 18
CLK 19
GND 20
40 VCC
39 AD15
38 A16/S3
37 A17/S4
36 A18/S5
35 A19/S6
34 $\overline{\text{BHE}}$/S7
33 MN/$\overline{\text{MX}}$
32 $\overline{\text{RD}}$
31 $\overline{\text{RQ}}$/$\overline{\text{GT0}}$ (HOLD)
30 $\overline{\text{RQ}}$/$\overline{\text{GT1}}$ (HLDA)
29 $\overline{\text{LOCK}}$ ($\overline{\text{WR}}$)
28 $\overline{\text{S2}}$ (M/$\overline{\text{IO}}$)
27 $\overline{\text{S1}}$ (DT/$\overline{\text{R}}$)
26 $\overline{\text{S0}}$ ($\overline{\text{DEN}}$)
25 QS0 (ALE)
24 QS1 ($\overline{\text{INTA}}$)
23 $\overline{\text{TEST}}$
22 READY
21 RESET

(a)


MIN MODE
[MAX MODE]
8088 CPU
GND 1
A14 2
A13 3
A12 4
A11 5
A10 6
A9 7
A8 8
AD7 9
AD6 10
AD5 11
AD4 12
AD3 13
AD2 14
AD1 15
AD0 16
NMI 17
INTR 18
CLK 19
GND 20
40 VCC
39 A15
38 A16/S3
37 A17/S4
36 A18/S5
35 A19/S6
34 $\overline{\text{SS0}}$ (HIGH)
33 MN/$\overline{\text{MX}}$
32 $\overline{\text{RD}}$
31 HOLD ($\overline{\text{RQ}}$/$\overline{\text{GT0}}$)
30 HLDA ($\overline{\text{RQ}}$/$\overline{\text{GT1}}$)
29 $\overline{\text{WR}}$ ($\overline{\text{LOCK}}$)
28 $\overline{\text{IO}}$/M ($\overline{\text{S2}}$)
27 DT/$\overline{\text{R}}$ ($\overline{\text{S1}}$)
26 $\overline{\text{DEN}}$ ($\overline{\text{S0}}$)
25 ALE (QS0)
24 $\overline{\text{INTA}}$ (QS1)
23 $\overline{\text{TEST}}$
22 READY
21 RESET

(b)

Pin Layout of the 8088 Microprocessors


MIN MODE
MAX MODE
GND 1
A14 2
A13 3
A12 4
A11 5
A10 6
A9 7
A8 8
AD7 9
AD6 10
AD5 11
AD4 12
AD3 13
AD2 14
AD1 15
AD0 16
NMI 17
INTR 18
CLK 19
GND 20
8088 CPU
40 V_{CC}
39 A15
38 A16/S3
37 A17/S4
36 A18/S5
35 A19/S6
34 $\overline{SS0}$ (HIGH)
33 MN/$\overline{MX}$
32 $\overline{RD}$
31 HOLD ($\overline{RQ}/\overline{GT0}$)
30 HLDA ($\overline{RQ}/\overline{GT1}$)
29 $\overline{WR}$ ($\overline{LOCK}$)
28 IO/$\overline{M}$ ($\overline{S2}$)
27 DT/$\overline{R}$ ($\overline{S1}$)
26 $\overline{DEN}$ ($\overline{S0}$)
25 ALE (QS0)
24 $\overline{INTA}$ (QS1)
23 $\overline{TEST}$
22 READY
21 RESET

Pin budget:
8088, Min mode:

20 Address
8 Data
20 Control & Status
3 Power

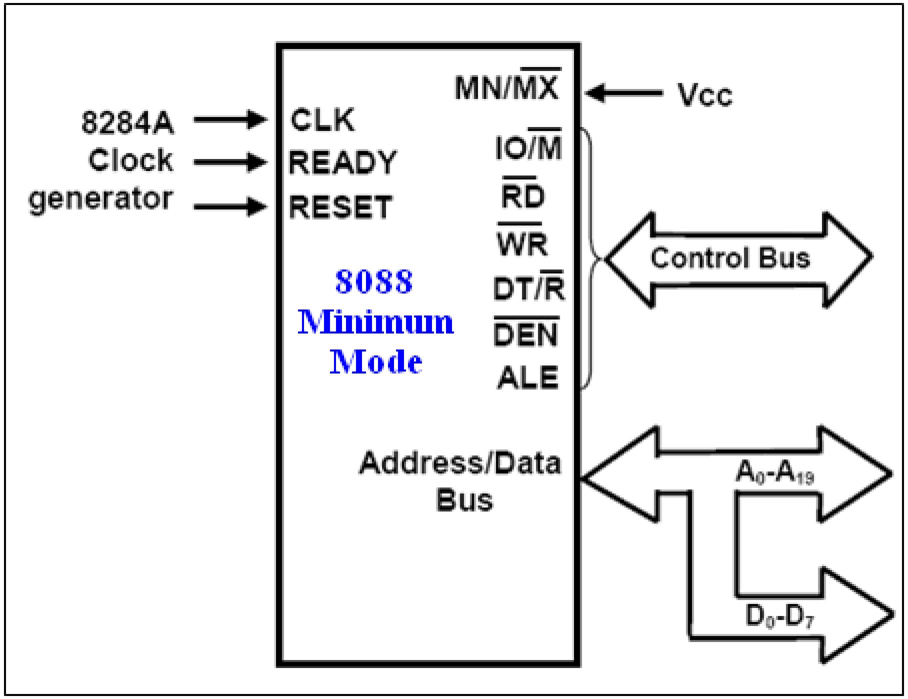
51 Total
> 40 pins available

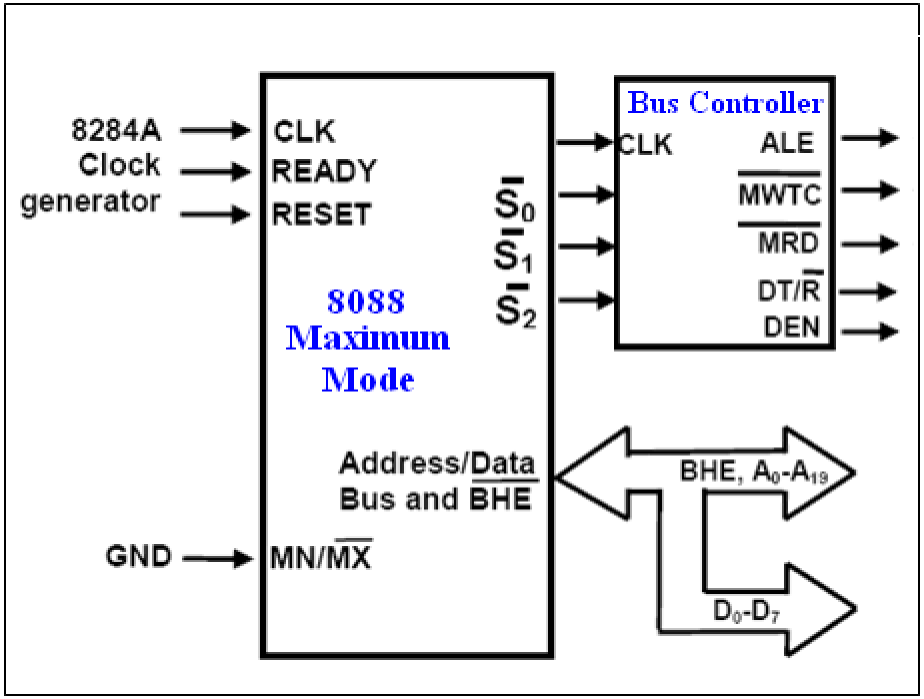
→ Use multiplexing

The Modes of Operation

- The microprocessors 8086 and 8088 can be configured to work in two modes: The Minimum mode and the Maximum mode.
  - The Minimum mode is used for single processor system, where 8086/8088 directly generates all the necessary control signals.
  - The Maximum mode is designed for multiprocessor systems, where an additional “Bus-controller” IC is required to generate the control signals. The processors control the Bus-controller using status-codes.

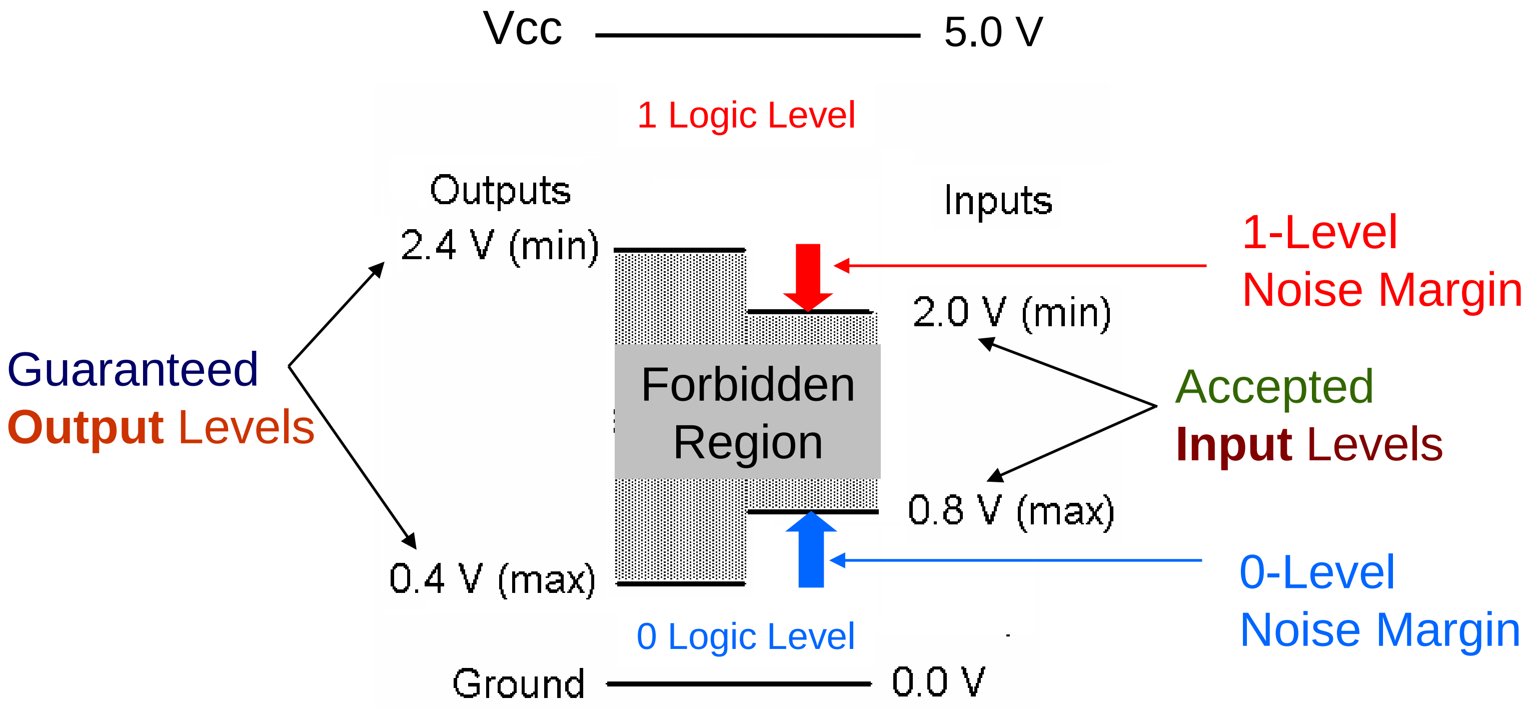
The Modes of Operation





DC Pin Characteristics: Voltages

Standard TTL Output and Inputs Voltage Levels

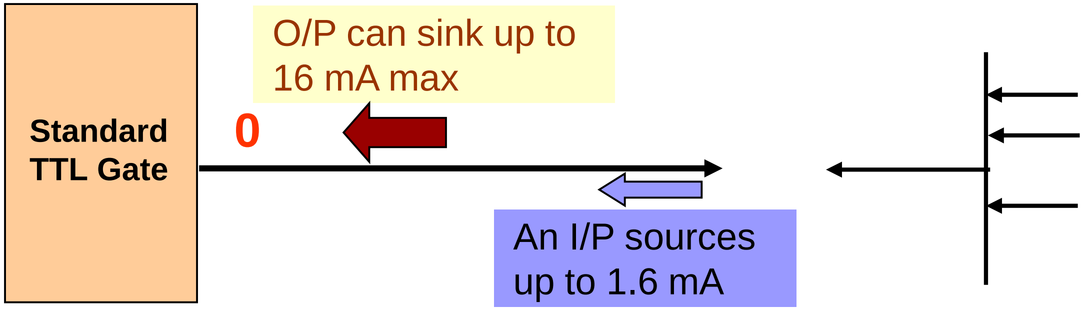


DC Pin Characteristics: Currents

Fan out for a standard TTL output

How many inputs can an output support?

For the 0 logic Level: (output “sinks” current)



Standard TTL Gate

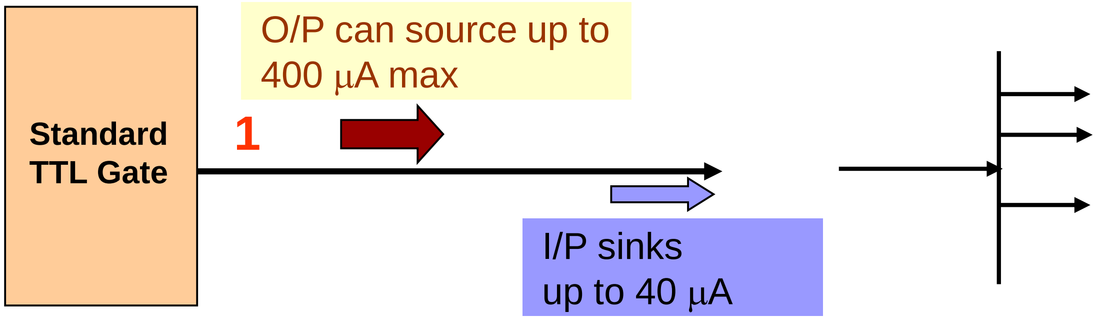
O/P can sink up to 16 mA max

0

An I/P sources up to 1.6 mA

0-level Fanout = Maximum number of inputs that the output can support = 16 mA/1.6 mA = 10

For the 1 logic Level: (output “sources” current)



Standard TTL Gate

O/P can source up to 400 μA max

1

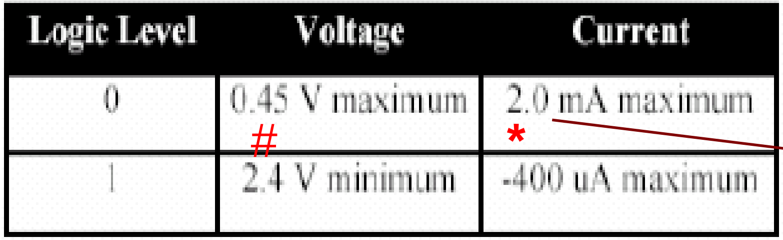
I/P sinks up to 40 μA

1-level Fanout = Maximum number of inputs that the output can support = 400 μA/40 μA = 10

8088/86 Pin Characteristics: DC

Output pins:

Guaranteed
Output levels

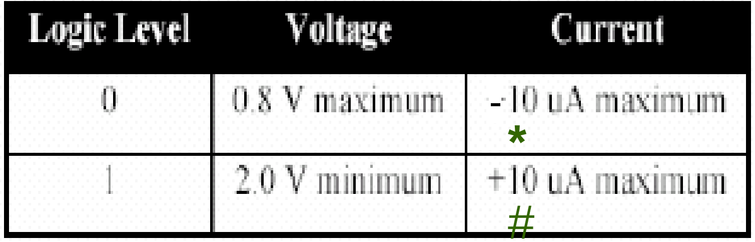


Logic Level	Voltage	Current
0	0.45 V maximum #	2.0 mA maximum *
1	2.4 V minimum	-400 uA maximum

* = 16 mA for standard 74 TTL
# = 0.40 V for standard 74 TTL

Input pins:

Accepted
Input levels



Logic Level	Voltage	Current
0	0.8 V maximum	-10 uA maximum *
1	2.0 V minimum	+10 uA maximum #

* = 1.6 mA for standard 74 TTL
= 40 µA for standard 74 TTL

0 level fan-out to TTL gate = 2 ÷ 1.6 ≈ 1 (8086/88 µP)
=16 ÷ 1.6 = 10 (for standard 74 TTL O/P)

0 level noise margin = 0.8 – 0.45 = 0.35 V (8086/88 µP)
= 0.8 – 0.40 = 0.40 V (for standard 74 TTL O/P)

8088/86 Pin Characteristics: DC

- **Input pins** are TTL compatible and require only ±10μA of current (actually better than TTL)

- **Output pins** are nearly TTL compatible, but have problems at logic 0:
  - □ A higher maximum logic 0 voltage of 0.45 V (instead of the TTL standard of 0.4 V)
  - □ This reduces logic 0 noise margin from 400 mV to 350 mV…

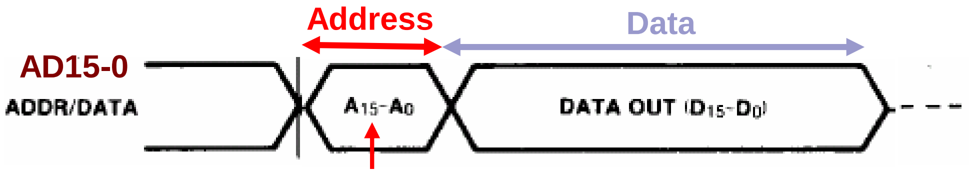
  → be careful with **long wiring** from output pins

- A lower logic 0 sink current of 2.0 mA (instead 16 mA for the standard 74 TTL)
  - □ This reduces fan out capability…

  → better **use 74LS**, or **AL**, for interfacing, or
  → **use buffers**

Multiplexing

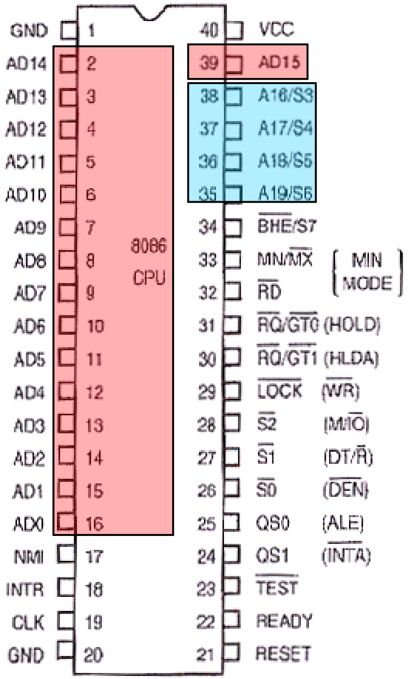
- Some functions are multiplexed on the same pins to reduce chip pin count

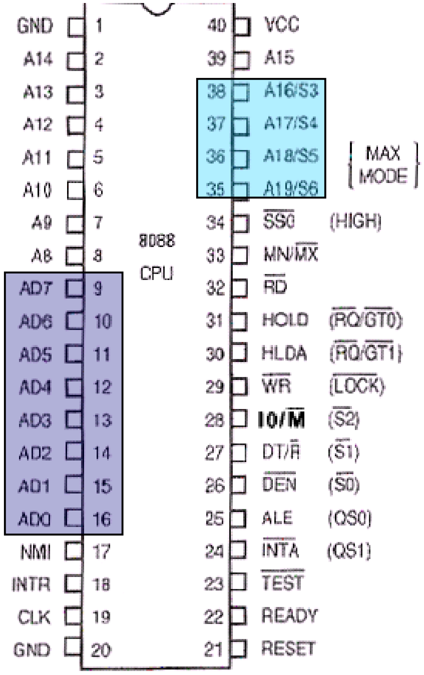


Write Cycle

ALE: Latch address

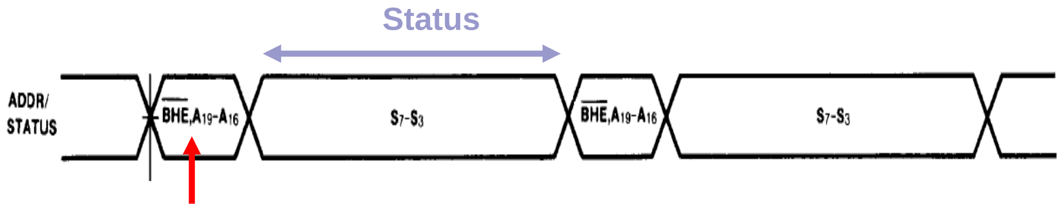
- **For both microprocessors, Address bus signals are A0-A19 (20 lines) for 1M byte of addressing space**
- **Data bus signals are**
  - D0-D7 for the 8088
  - D0-D15 for the 8086
- **The address & data pins are multiplexed as:**
  - AD0-AD7 (8088)
  - or AD0-AD15 (8086)
- **Address/Status pins are MUXed**
  - A/S for A16-19
- **The ALE O/P signal is used to demultiplex the address/data (AD) bus and also the address/status (A/S) bus.**





The Status (S) Bus

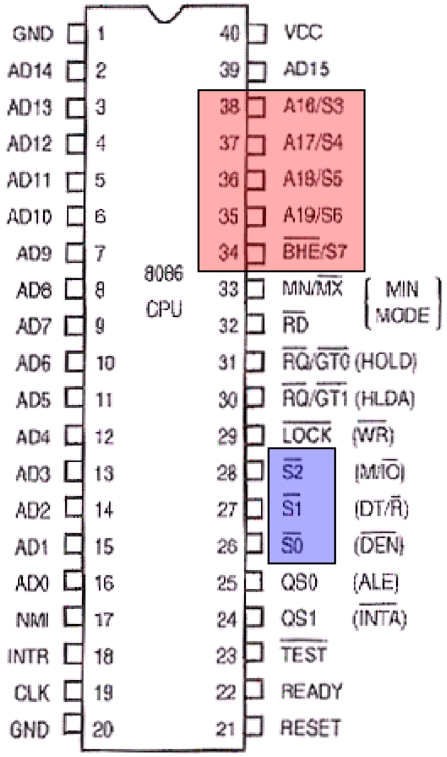
- 8086: Address bits A16-A19 & BHE/ are muxed with the status bits S3-S7.

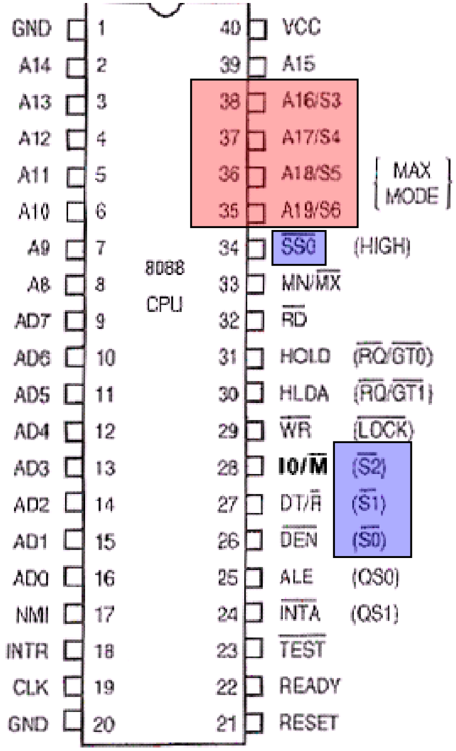


ALE: Latch address and BHE/

- S3 & S4 indicate which segment register is used with the current instruction:

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



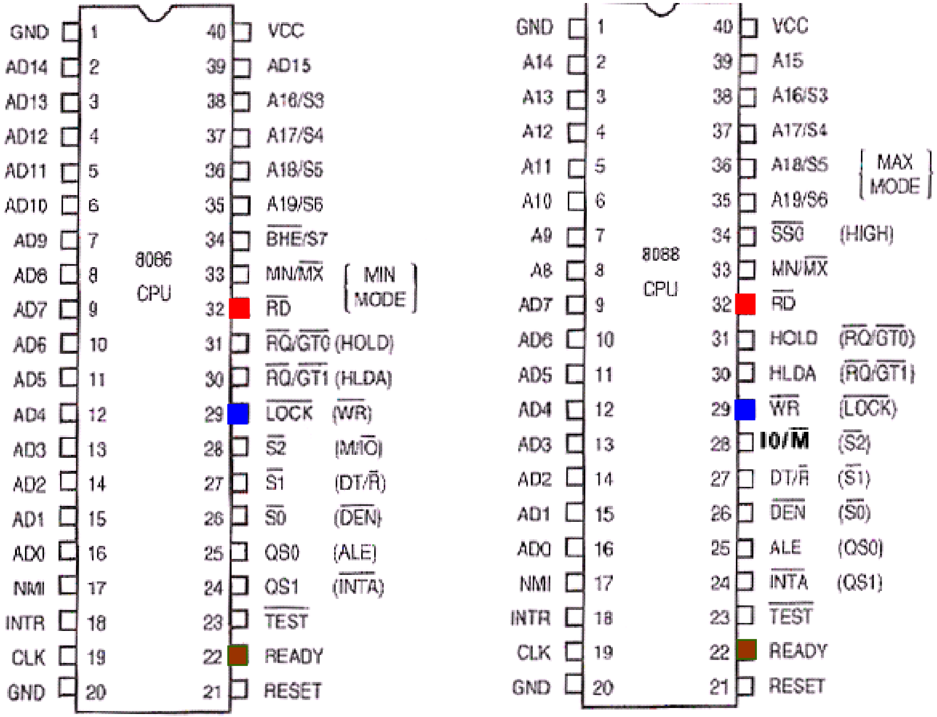


- S5 = IF (Interrupt flag)
- S6 = 0 } Spare
- S7 = 1 } Spare
- S0/, S1/, S2/ are not MUXed. They encode bus status (current bus cycle)
- Available only in the MAX mode for use by a bus controller chip
- SS0: Not Muxed, Min mode

Main Control Signals

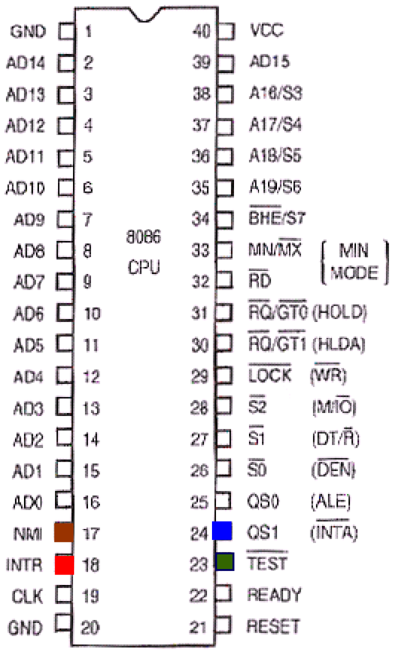
Common Signals for both MIN and MAX modes:

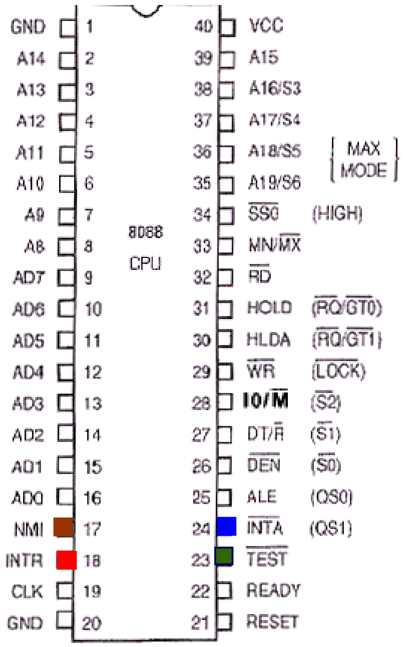
- S0/, The read output (**RD/**): indicates a read operation
- The write output (**WR/**) : indicates a write
- The **READY** input: when low (not ready), forces the processor to enter a wait state. Facilitates interfacing the processor with slow memory chips

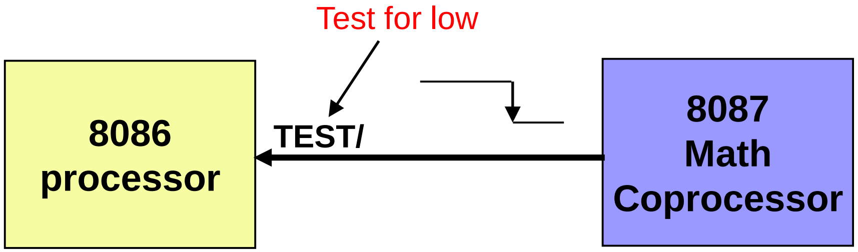


Main Control Signals (cont’d)

- **Two hardware interrupt inputs:**
  - □ **INTR input:** Hardware interrupt request. Honored only if the **IF flag** is set. The microprocessor enters an interrupt ACK cycle by lowering the **INTA/** output
  - □ **NMI input:** Hardware non-maskable interrupt request. Honored regardless of the status of the IF flag. Uses interrupt vector 2
- **TEST/ input:** Example: interfacing the microprocessor with the 8087 math coprocessor. Checked by the WAIT instruction that precedes each floating point instruction. If high, the instruction waits till input signal goes low and then gives FP instruction to the math processor






Test for low

8086 processor

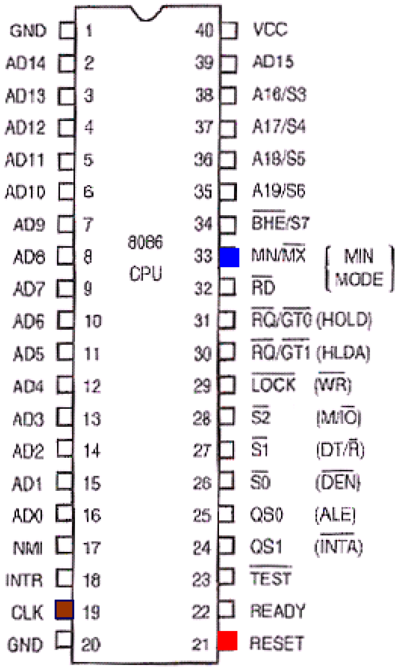
TEST/

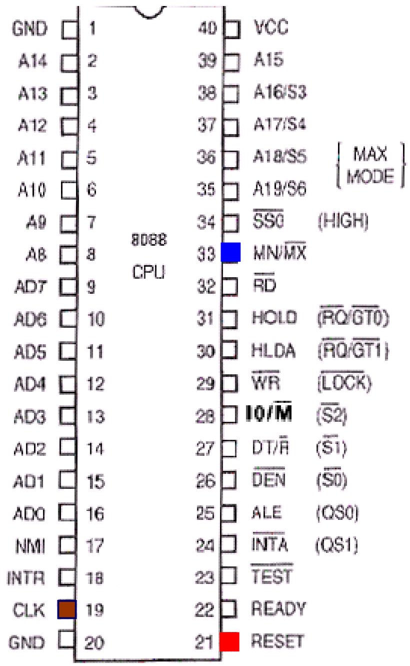
8087 Math Coprocessor

Synchronizes processor execution to external events

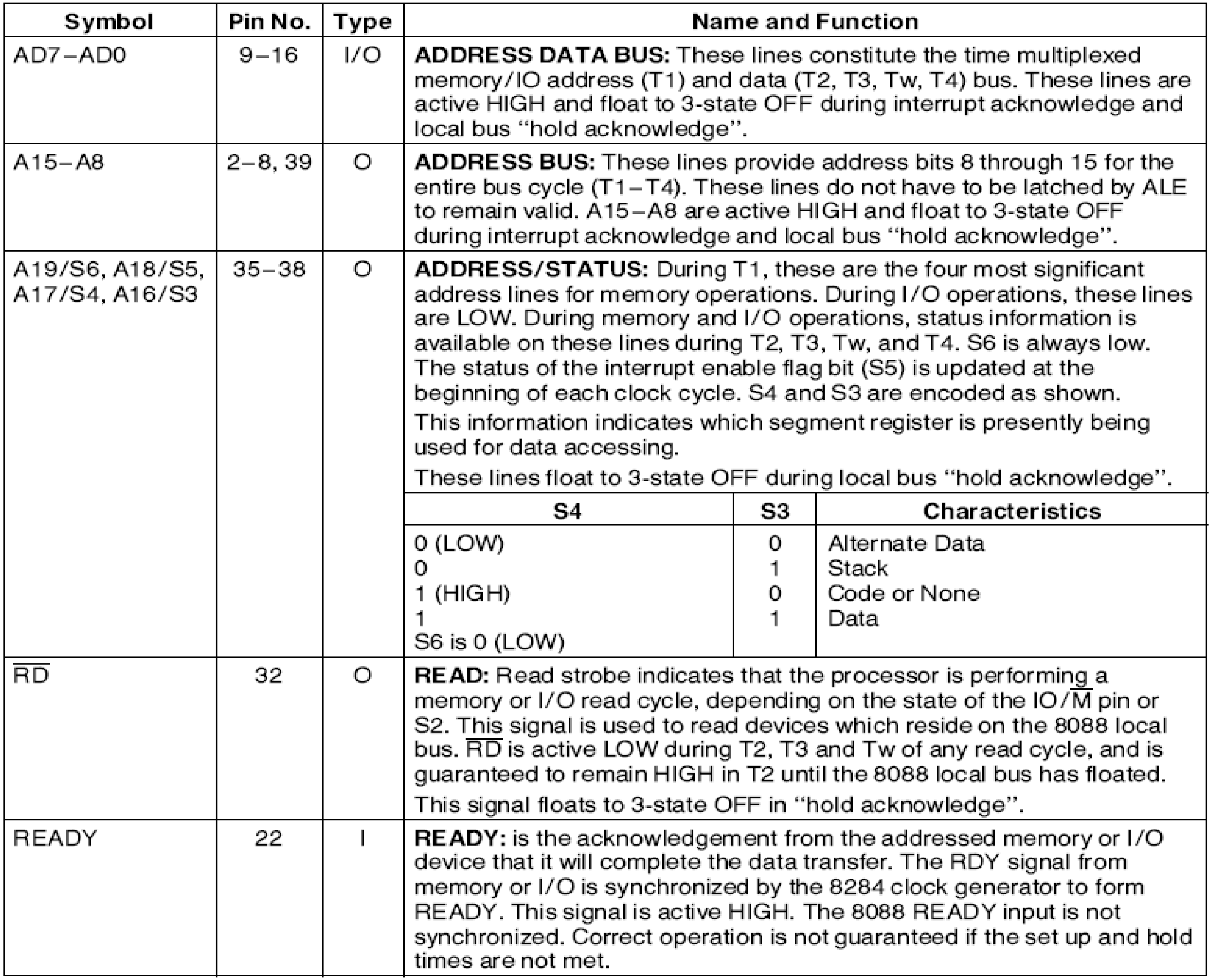
Main Control Signals (cont'd)

- **CLK input:** Basic timing clock for the processor (Duty cycle= 1/3)
- **MN/#MX input:** Selects either Minimum (+ 5V directly) or Maximum mode (GND)
- **#BHE/S7 output** (MUXed):
  - BHE/: (Bus High Enable) Enables writing to the high byte of the 16-bit data bus on the 8086
  - Not on 8088 (has an 8-bit data bus- no high byte!)
- **RESET input:** resets the microprocessor (reboots the computer). Causes the processor to start executing at address FFFF0H (Start of last 16 bytes of ROM at the top of the 1MB memory) after disabling the INTR input interrupts (CLR IF flag). Input must be kept high for at least 50 ms. Sampled by the processor at the + ive clock edge



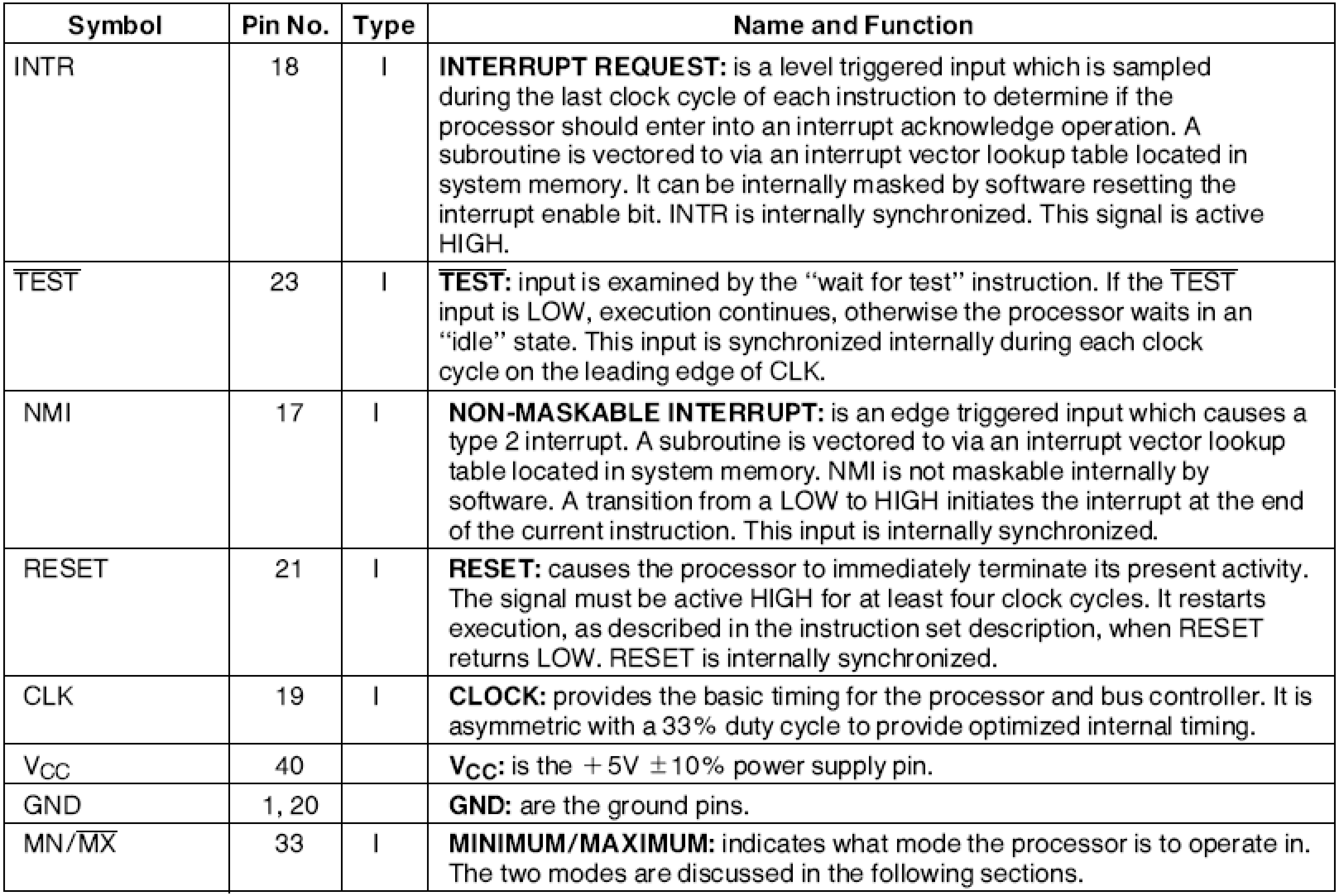


Main Control Signals (cont'd)



<table>
<tr><th>Symbol</th><th>Pin No.</th><th>Type</th><th colspan="3">Name and Function</th></tr>
<tr><td>AD7–AD0</td><td>9–16</td><td>I/O</td><td colspan="3"><b>ADDRESS DATA BUS:</b> These lines constitute the time multiplexed memory/IO address (T1) and data (T2, T3, Tw, T4) bus. These lines are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus "hold acknowledge".</td></tr>
<tr><td>A15–A8</td><td>2–8, 39</td><td>O</td><td colspan="3"><b>ADDRESS BUS:</b> These lines provide address bits 8 through 15 for the entire bus cycle (T1–T4). These lines do not have to be latched by ALE to remain valid. A15–A8 are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus "hold acknowledge".</td></tr>
<tr><td rowspan="3">A19/S6, A18/S5, A17/S4, A16/S3</td><td rowspan="3">35–38</td><td rowspan="3">O</td><td colspan="3"><b>ADDRESS/STATUS:</b> During T1, 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 T2, T3, Tw, and T4. S6 is always low. The status of the interrupt enable flag bit (S5) is updated at the beginning of each clock cycle. S4 and S3 are encoded as shown.
This information indicates which segment register is presently being used for data accessing.
These lines float to 3-state OFF during local bus "hold acknowledge".</td></tr>
<tr><th>S4</th><th>S3</th><th>Characteristics</th></tr>
<tr><td>0 (LOW)
0
1 (HIGH)
1
S6 is 0 (LOW)</td><td>0
1
0
1</td><td>Alternate Data
Stack
Code or None
Data</td></tr>
<tr><td>$\overline{RD}$</td><td>32</td><td>O</td><td colspan="3"><b>READ:</b> Read strobe indicates that the processor is performing a memory or I/O read cycle, depending on the state of the IO/$\overline{M}$ pin or S2. This signal is used to read devices which reside on the 8088 local bus. $\overline{RD}$ is active LOW during T2, T3 and Tw of any read cycle, and is guaranteed to remain HIGH in T2 until the 8088 local bus has floated.
This signal floats to 3-state OFF in "hold acknowledge".</td></tr>
<tr><td>READY</td><td>22</td><td>I</td><td colspan="3"><b>READY:</b> is the acknowledgement from the addressed memory or I/O device that it will complete the data transfer. The RDY signal from memory or I/O is synchronized by the 8284 clock generator to form READY. This signal is active HIGH. The 8088 READY input is not synchronized. Correct operation is not guaranteed if the set up and hold times are not met.</td></tr>
</table>

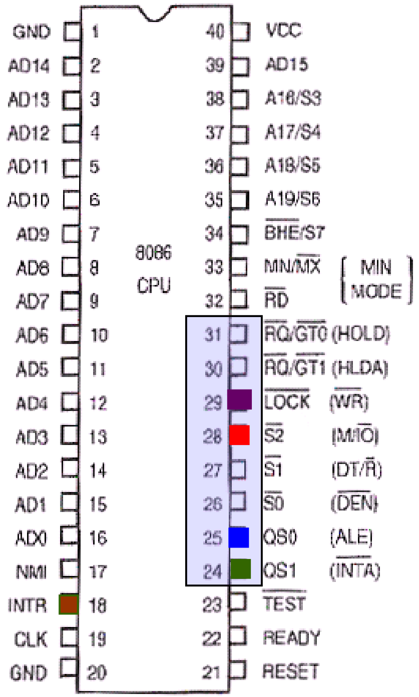
Main Control Signals (cont’d)

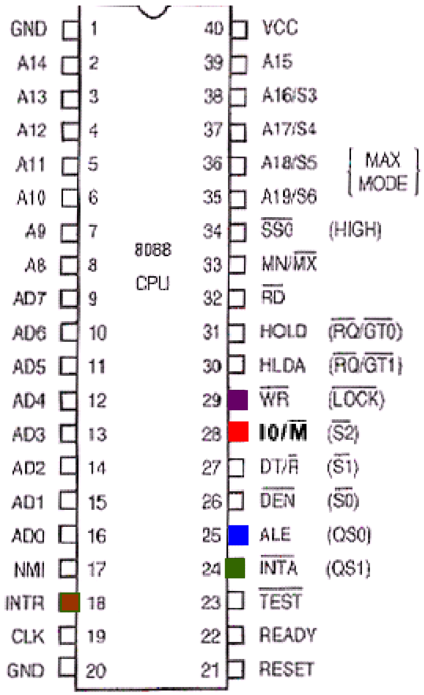


Symbol	Pin No.	Type	Name and Function
INTR	18	I	**INTERRUPT REQUEST:** is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup 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.
$\overline{TEST}$	23	I	**$\overline{TEST}$:** input is examined by the “wait for test” 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.
NMI	17	I	**NON-MASKABLE INTERRUPT:** is 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.
RESET	21	I	**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.
CLK	19	I	**CLOCK:** provides the basic timing for the processor and bus controller. It is asymmetric with a 33% duty cycle to provide optimized internal timing.
V_{CC}	40		**V_{CC}:** is the +5V ±10% power supply pin.
GND	1, 20		**GND:** are the ground pins.
MN/$\overline{MX}$	33	I	**MINIMUM/MAXIMUM:** indicates what mode the processor is to operate in. The two modes are discussed in the following sections.

Minimum Mode Control Signals

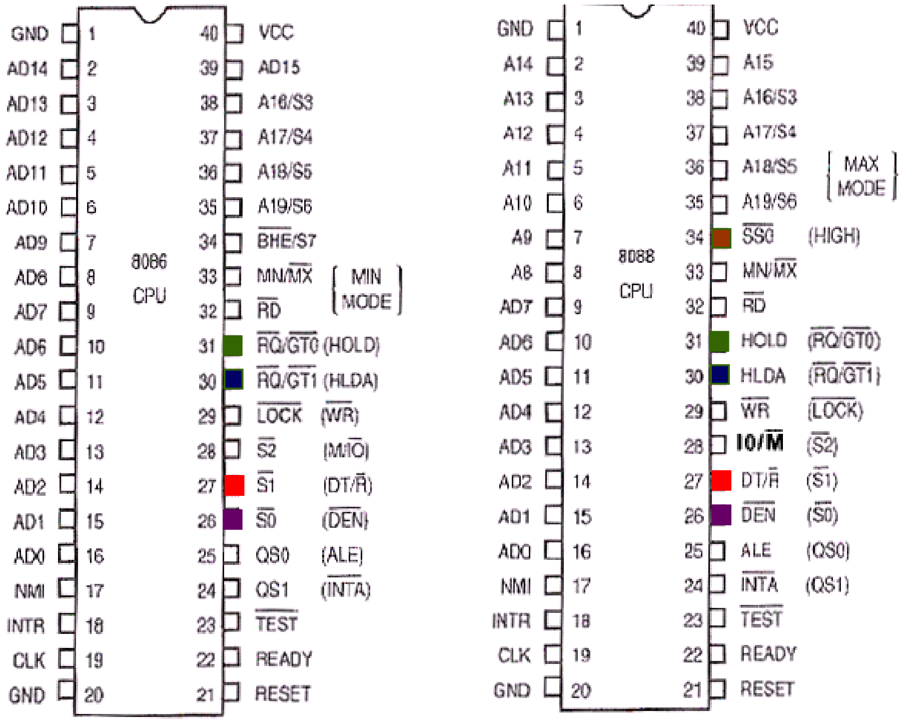
- For the processor to operate in the minimum mode, connect **MN/#MX** input to +5V.
- **M/#IO or IO/#M output:** indicates whether the address on the address bus is a memory address (IO/#M = 0) or an I/O address (IO/#M = 1)
- **WR/** output**:** indicates a write operation.
- **INTA/** output: interrupt acknowledgement. Goes low in response to a hardware interrupt request applied to the INTR input. Interrupting device uses it to put the interrupt vector number on the data bus. The microprocessor read the number and identifies the Interrupt Service Routine (ISR)
- **ALE** (address latch enable) output: Indicates that the muxed AD bus now carries address (memory or I/O). Use to latch that address to an external circuit before the processor removes it!.

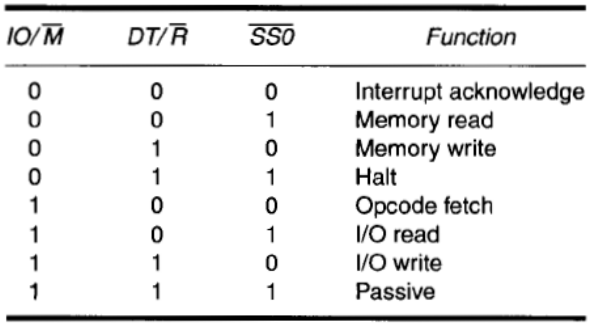



8088
CPU
GND 1
A14 2
A13 3
A12 4
A11 5
A10 6
A9 7
A8 8
AD7 9
AD6 10
AD5 11
AD4 12
AD3 13
AD2 14
AD1 15
AD0 16
NMI 17
INTR 18
CLK 19
GND 20
40 VCC
39 A15
38 A16/S3
37 A17/S4
36 A18/S5
35 A19/S6
MAX MODE
34 $\overline{SS0}$ (HIGH)
33 MN/$\overline{MX}$
32 $\overline{RD}$
31 HOLD ($\overline{RQ}$/$\overline{GT0}$)
30 HLDA ($\overline{RQ}$/$\overline{GT1}$)
29 $\overline{WR}$ ($\overline{LOCK}$)
28 IO/$\overline{M}$ ($\overline{S2}$)
27 DT/$\overline{R}$ ($\overline{S1}$)
26 $\overline{DEN}$ ($\overline{S0}$)
25 ALE (QS0)
24 $\overline{INTA}$ (QS1)
23 $\overline{TEST}$
22 READY
21 RESET

Minimum Mode Control Signals (Cont'd)

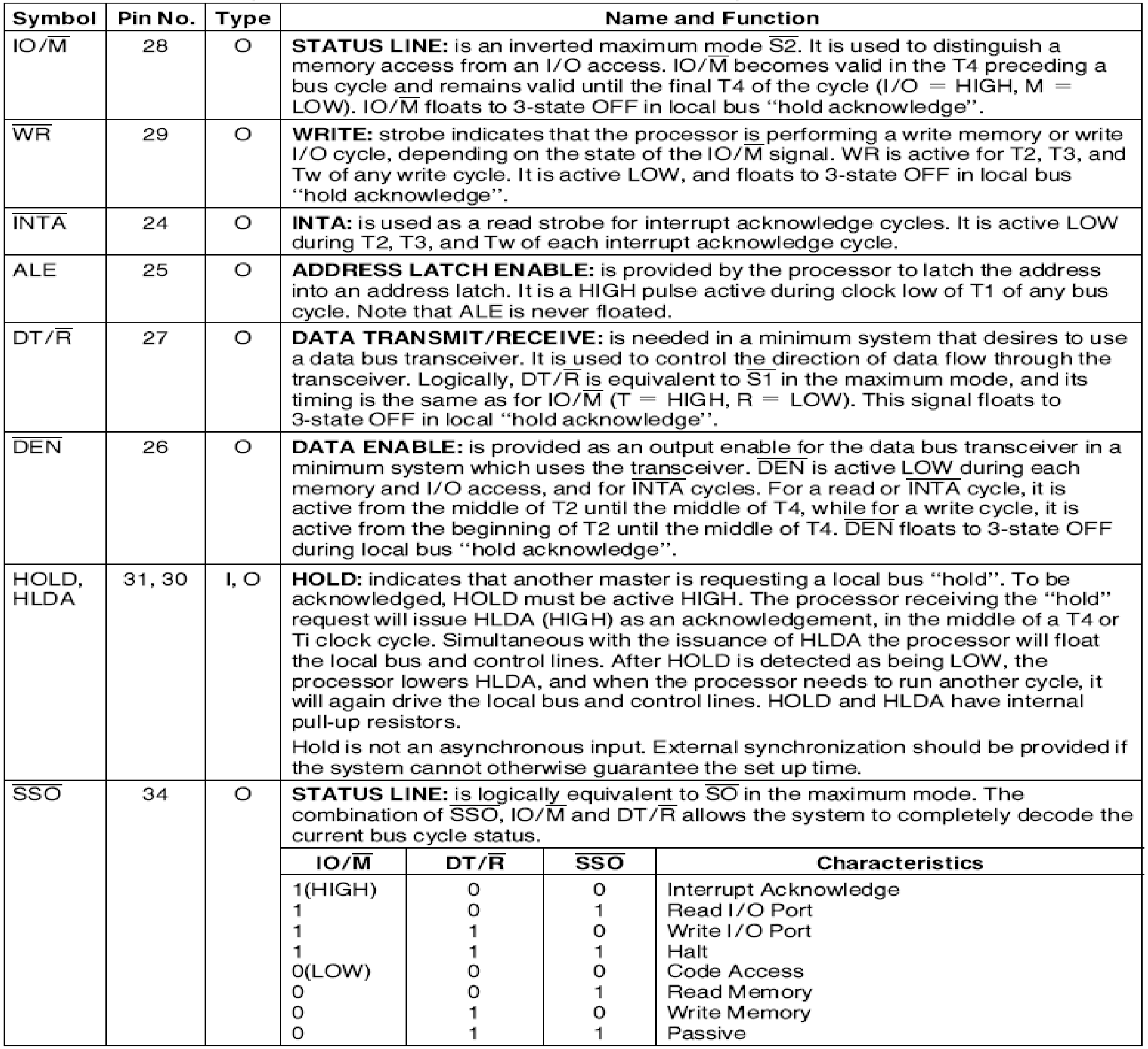
- **DT/#R** output: indicates if the data bus is transmitting (outputing) data (=1) or receiving (inputting) data (=0). Use to control external bidirectional buffers connected to the data bus.
- **DEN/** output: (data bus enable). Active when AD bus carries data not address Use to activate external data buffers.
- **HOLD** input: Requests a direct memory access (DMA) from the microprocessor. In response, the microprocessor stops execution and places the data, address, and control buses at **High Z state (floats them).** Signals such as RD/ and WR/ are also floated.
- **HLDA** output: Acknowledges that the processor has entered a hold state in response to HOLD.
- **SS0/** output: Equivalent to the S0 status output of the maximum mode. Use with IO/#M and DT/#R to decode the current bus cycle.





$IO/\overline{M}$	$DT/\overline{R}$	$\overline{SS0}$	Function
0	0	0	Interrupt acknowledge
0	0	1	Memory read
0	1	0	Memory write
0	1	1	Halt
1	0	0	Opcode fetch
1	0	1	I/O read
1	1	0	I/O write
1	1	1	Passive

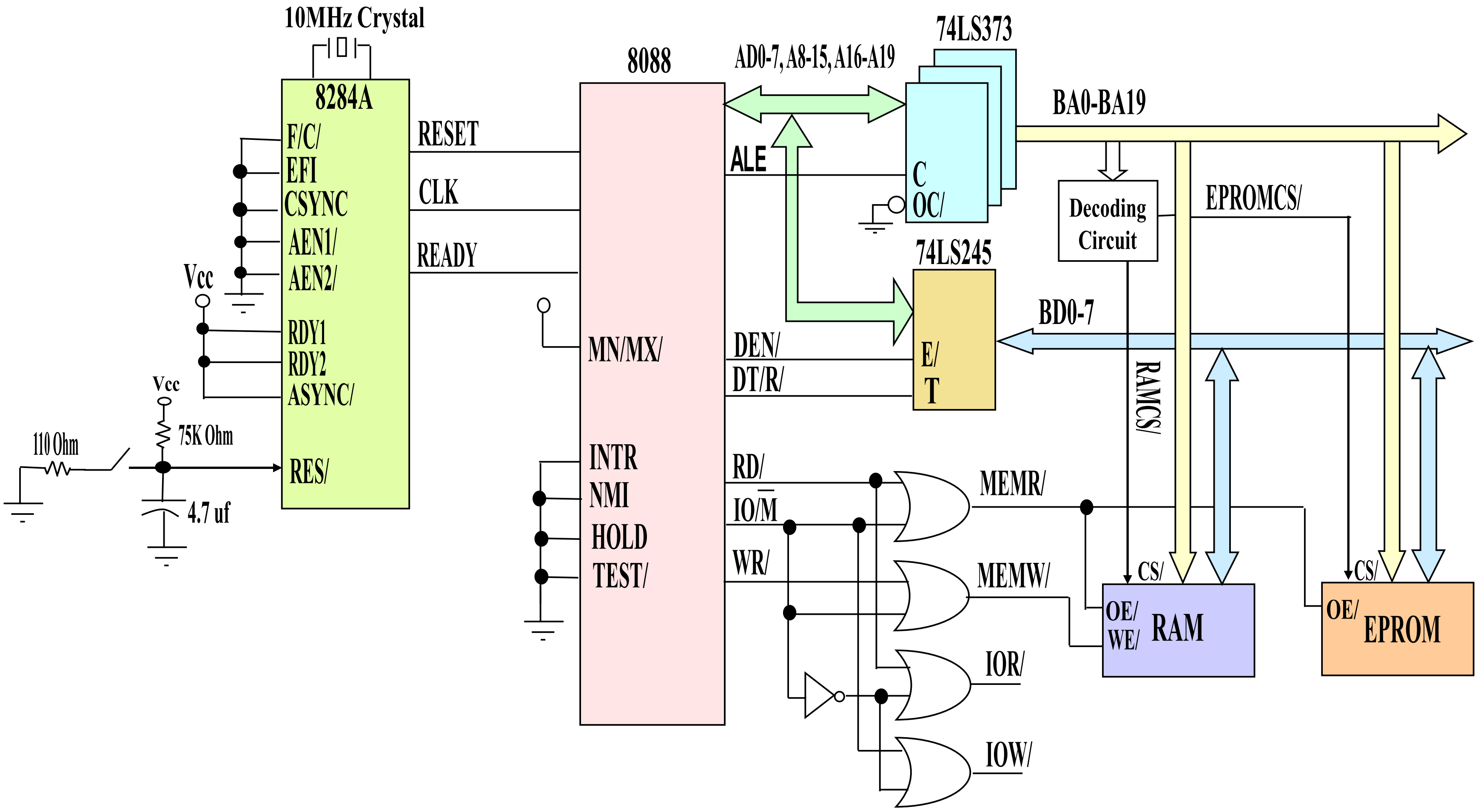
Minimum Mode Control Signals (Cont'd)



Symbol	Pin No.	Type	Name and Function
IO/$\overline{M}$	28	O	**STATUS LINE:** is an inverted maximum mode $\overline{S2}$. It is used to distinguish a memory access from an I/O access. IO/$\overline{M}$ becomes valid in the T4 preceding a bus cycle and remains valid until the final T4 of the cycle (I/O = HIGH, M = LOW). IO/$\overline{M}$ floats to 3-state OFF in local bus "hold acknowledge".
$\overline{WR}$	29	O	**WRITE:** strobe indicates that the processor is performing a write memory or write I/O cycle, depending on the state of the IO/$\overline{M}$ signal. WR is active for T2, T3, and Tw of any write cycle. It is active LOW, and floats to 3-state OFF in local bus "hold acknowledge".
$\overline{INTA}$	24	O	**INTA:** is used as a read strobe for interrupt acknowledge cycles. It is active LOW during T2, T3, and Tw of each interrupt acknowledge cycle.
ALE	25	O	**ADDRESS LATCH ENABLE:** is provided by the processor to latch the address into an address latch. It is a HIGH pulse active during clock low of T1 of any bus cycle. Note that ALE is never floated.
DT/$\overline{R}$	27	O	**DATA TRANSMIT/RECEIVE:** is needed in a minimum system that desires to use a data bus transceiver. It is used to control the direction of data flow through the transceiver. Logically, DT/$\overline{R}$ is equivalent to $\overline{S1}$ in the maximum mode, and its timing is the same as for IO/$\overline{M}$ (T = HIGH, R = LOW). This signal floats to 3-state OFF in local "hold acknowledge".
$\overline{DEN}$	26	O	**DATA ENABLE:** is provided as an output enable for the data bus transceiver in a minimum system which uses the transceiver. $\overline{DEN}$ is active LOW during each memory and I/O access, and for $\overline{INTA}$ cycles. For a read or $\overline{INTA}$ cycle, it is active from the middle of T2 until the middle of T4, while for a write cycle, it is active from the beginning of T2 until the middle of T4. $\overline{DEN}$ floats to 3-state OFF during local bus "hold acknowledge".
HOLD, HLDA	31, 30	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 T4 or Ti 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 lowers HLDA, and when the processor needs to run another cycle, it will again drive the local bus and control lines. HOLD and HLDA have internal pull-up resistors. Hold is not an asynchronous input. External synchronization should be provided if the system cannot otherwise guarantee the set up time.
$\overline{SSO}$	34	O	**STATUS LINE:** is logically equivalent to $\overline{SO}$ in the maximum mode. The combination of $\overline{SSO}$, IO/$\overline{M}$ and DT/$\overline{R}$ allows the system to completely decode the current bus cycle status.

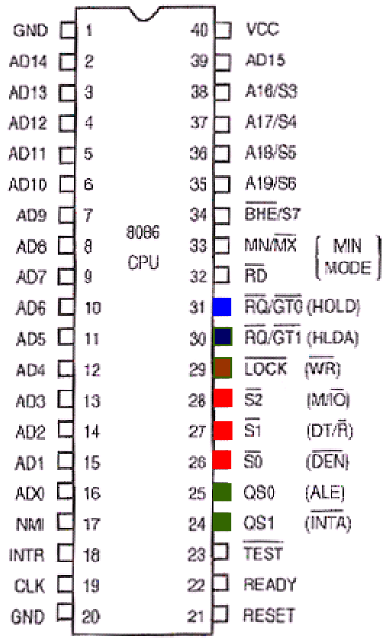
IO/$\overline{M}$	DT/$\overline{R}$	$\overline{SSO}$	Characteristics
1(HIGH)	0	0	Interrupt Acknowledge
1	0	1	Read I/O Port
1	1	0	Write I/O Port
1	1	1	Halt
0(LOW)	0	0	Code Access
0	0	1	Read Memory
0	1	0	Write Memory
0	1	1	Passive

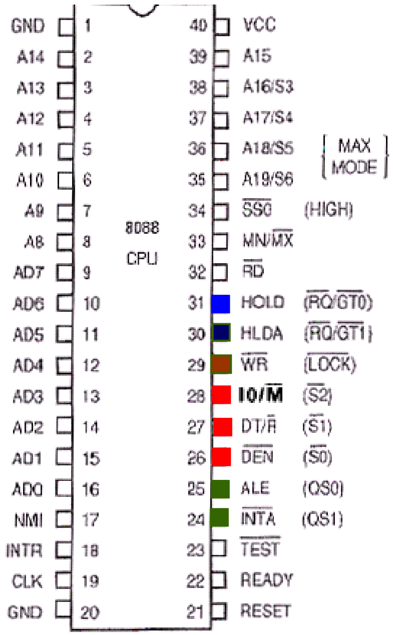
Minimum Mode 8088 System

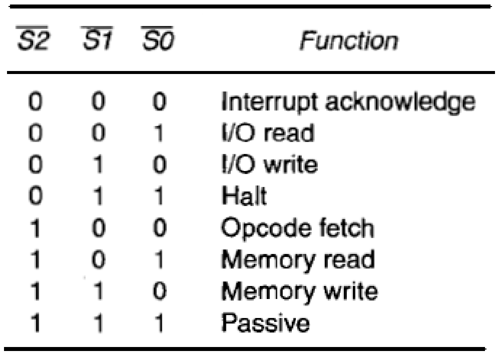


Maximum Mode Control Signals (Cont’d)

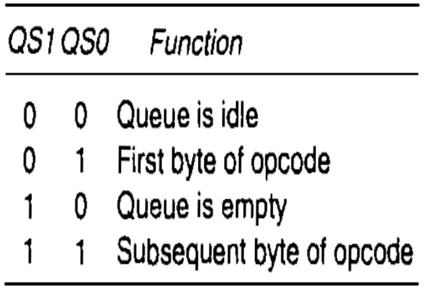
- For the processor to operate in the maximum mode, connect **MN/#MX** input to ground.
- **S0/, S1/, S2/** outputs: Status bits that encode the type of the current bus cycle, Used by the 8288 bus controller and the 8087 coprocessor.
- **#RQ/GT0, #RQ/GT1**: Bidirectional lines for requesting and granting DMA access (Request/Get). For use in multiprocessor systems. The RG/GT0 line has higher priority.
- **LOCK/** output: Activated for the duration of multiprocessor instructions having the LOCK prefix. Can be used to prevent other microprocessors from using the system buses and accessing shared memory or I/O for the duration of such instructions, e.g. **LOCK:MOV AL,[SI]**
- **QS0, QS1** (Queue Status) outputs: indicate the status of the internal instruction queue. For use by the 8087 coprocessor to keep in step





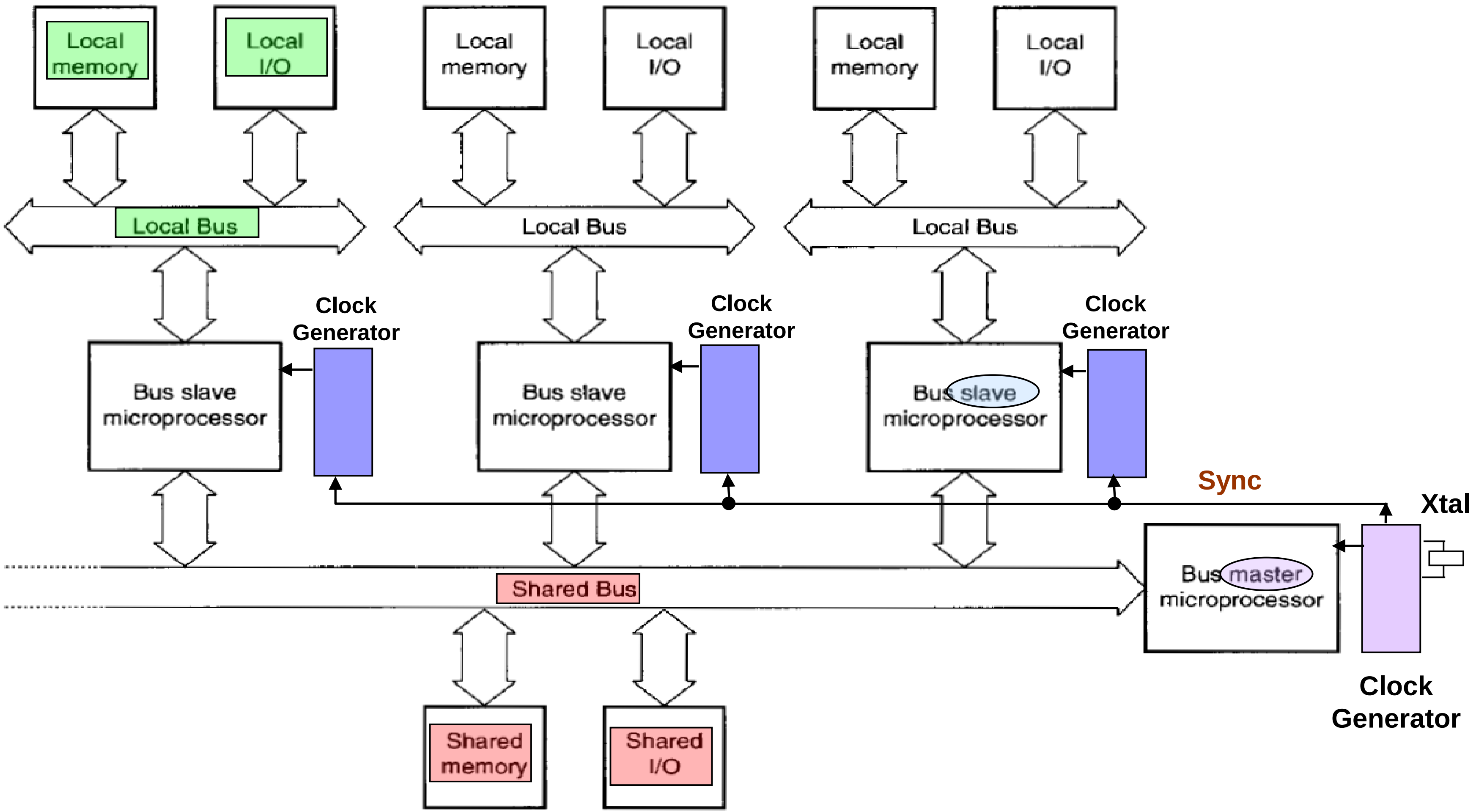


$\overline{S2}$	$\overline{S1}$	$\overline{S0}$	Function
0	0	0	Interrupt acknowledge
0	0	1	I/O read
0	1	0	I/O write
0	1	1	Halt
1	0	0	Opcode fetch
1	0	1	Memory read
1	1	0	Memory write
1	1	1	Passive

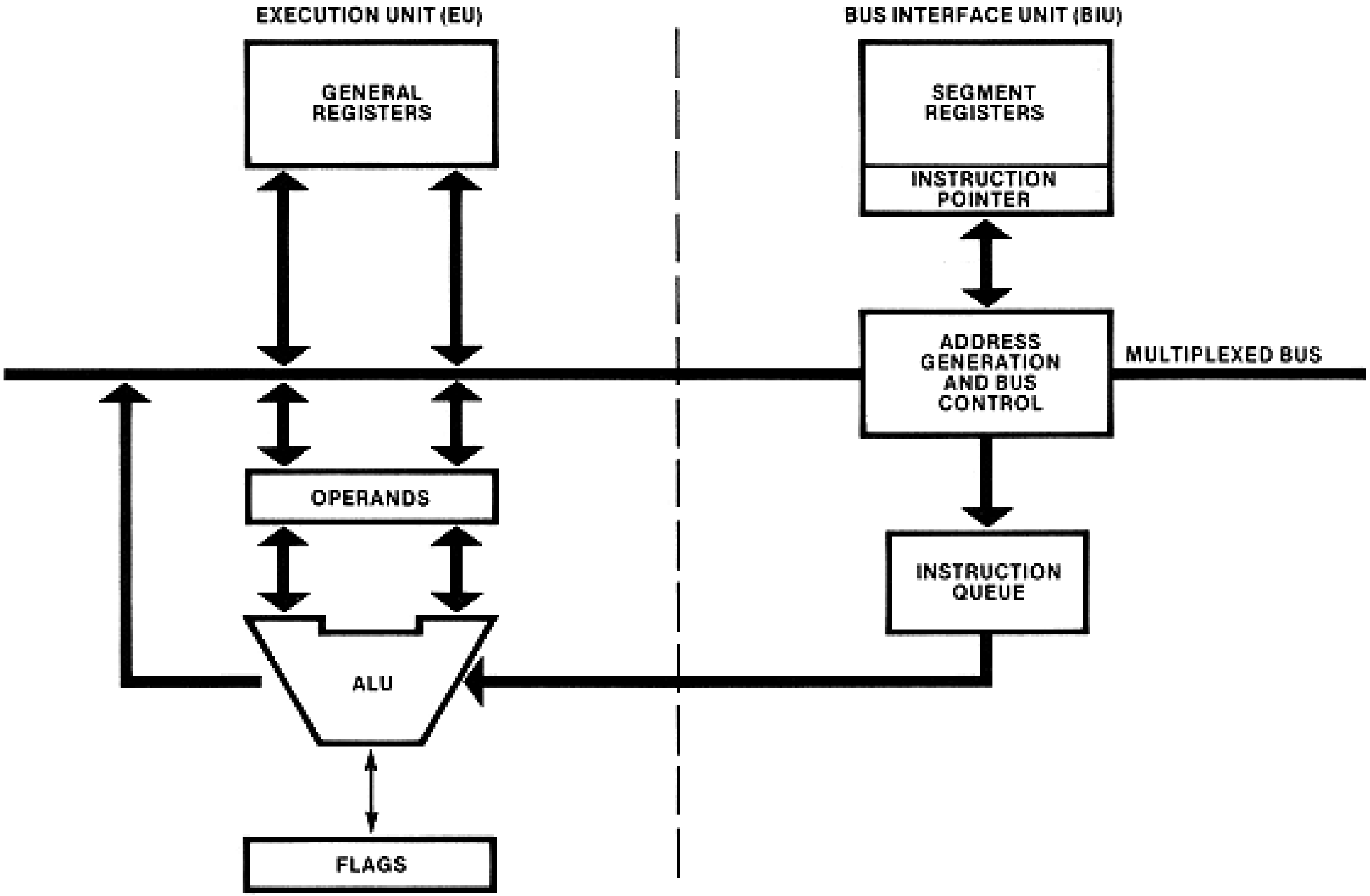


QS1	QS0	Function
0	0	Queue is idle
0	1	First byte of opcode
1	0	Queue is empty
1	1	Subsequent byte of opcode

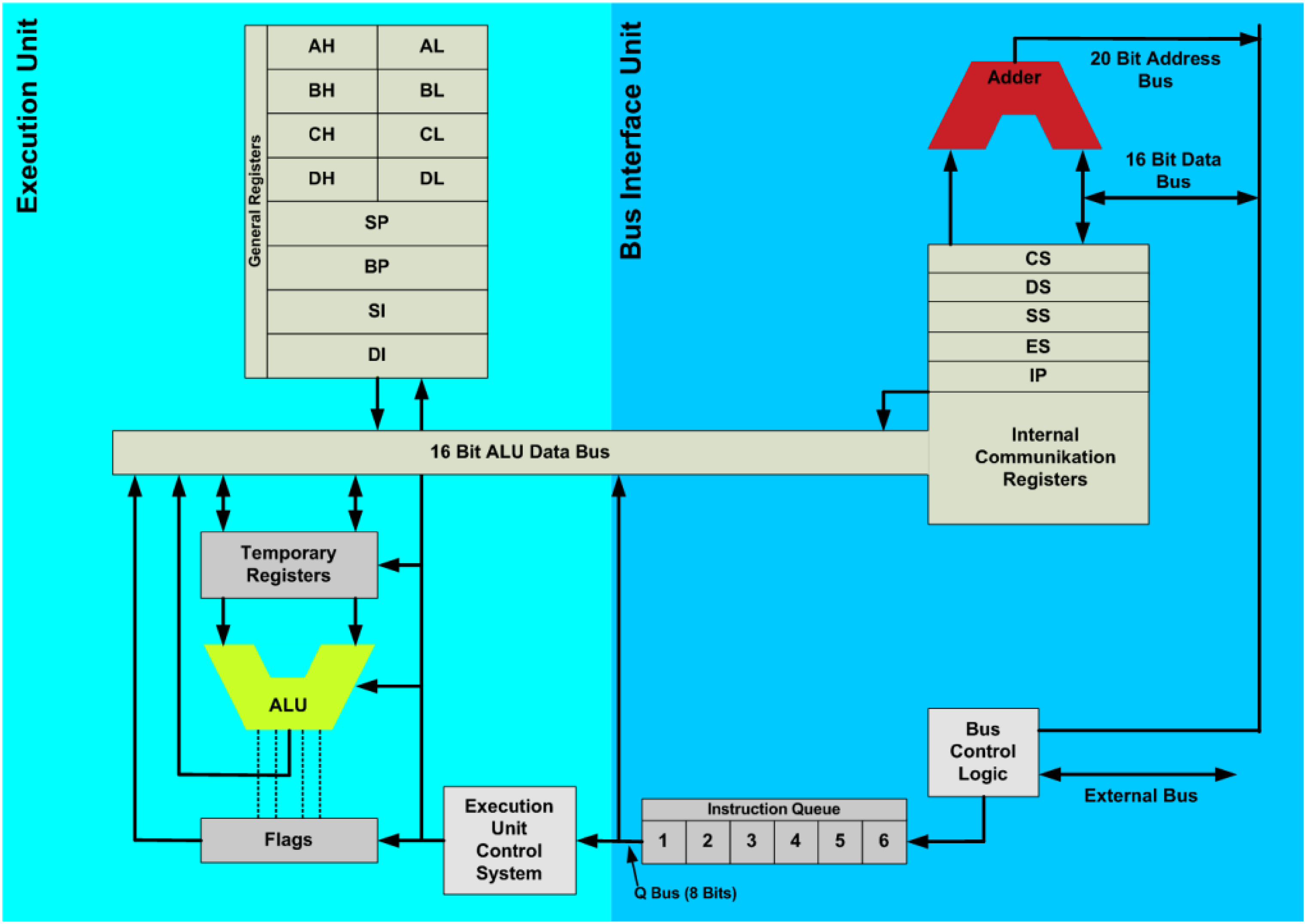
Maximum Mode Control Signals (Cont'd)



8088/8086 CPU Functional Block Diagram



8088/8086 CPU Functional Block Diagram



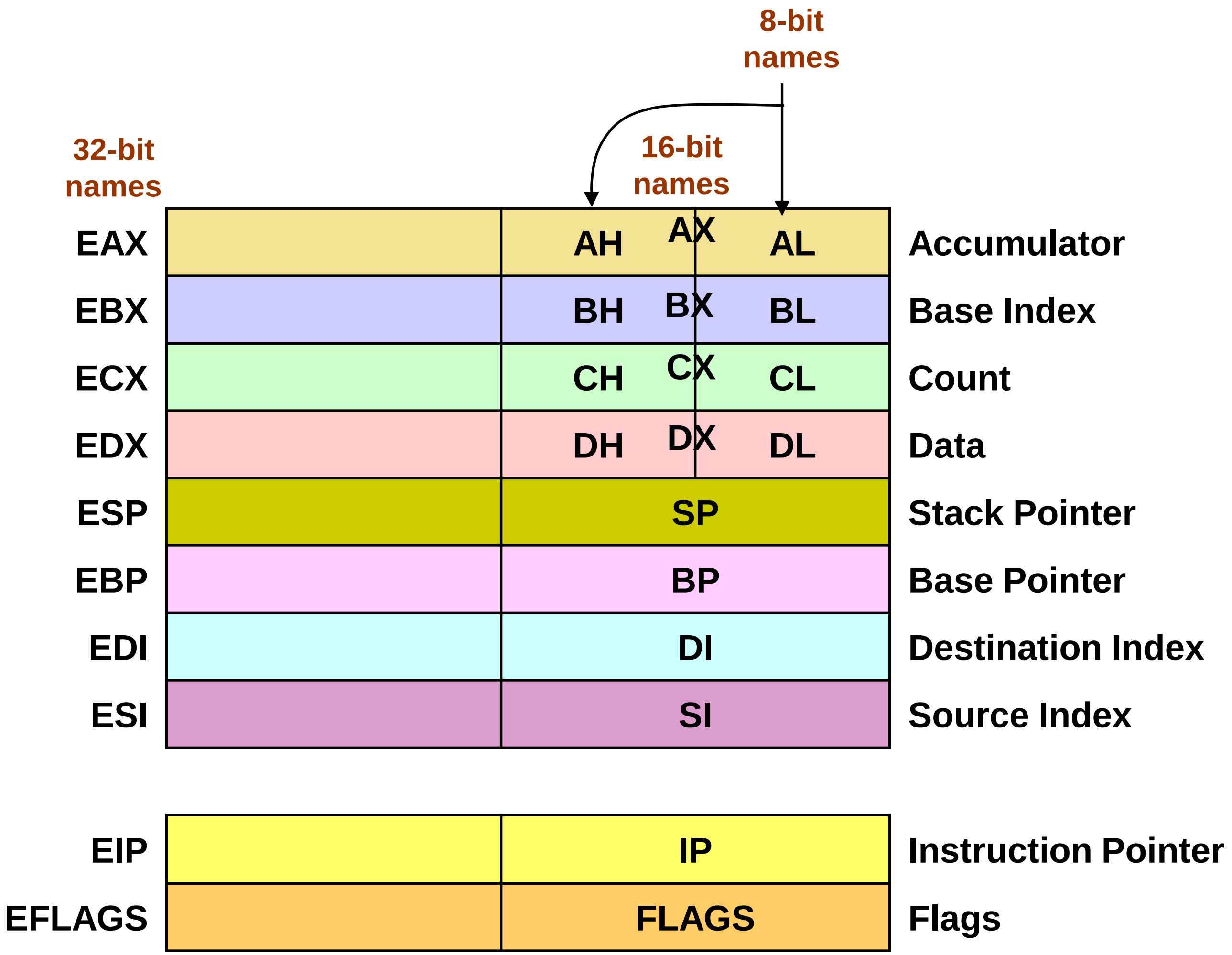
Internal Architecture of 8088/8086

- The 8088/8086 microprocessor has two units: the **Bus Interface Unit (BIU)** and the **Execution Unit (EU)**.
- Both units operate simultaneously. This parallelism makes the fetch and execution of instructions independent.
- The **Bus Interface Unit (BIU)** performs all external bus operations, such as instruction fetching, reading/writing operands from/to memory, and inputting/outputting data from/to Input/output peripherals.
- The **BIU** also performs address generation. To perform this function, the **BIU** contains segment registers, instruction pointers, address generation adder, bus control logic, and an instruction queue.

Internal Architecture of 8088/8086

- The **Execution Unit (EU)** performs decoding and execution of instructions.
- The **EU** consists of the arithmetic logic unit (**ALU**), the **FLAG** register, multipurpose registers, and temporary operand register.
- The **EU** accesses instructions from the instruction queue, decodes them, generates operand addresses if necessary, passes the operand addresses to the **BIU**, requests from the **BIU** to perform read/write bus cycle from/to memory or I/O peripherals, and performs the operation specified by the instruction on the operands.

Multipurpose and Special-Purpose Registers



32-bit names	16-bit names	8-bit names	
EAX	AX	AH AL	Accumulator
EBX	BX	BH BL	Base Index
ECX	CX	CH CL	Count
EDX	DX	DH DL	Data
ESP	SP		Stack Pointer
EBP	BP		Base Pointer
EDI	DI		Destination Index
ESI	SI		Source Index
EIP	IP		Instruction Pointer
EFLAGS	FLAGS		Flags

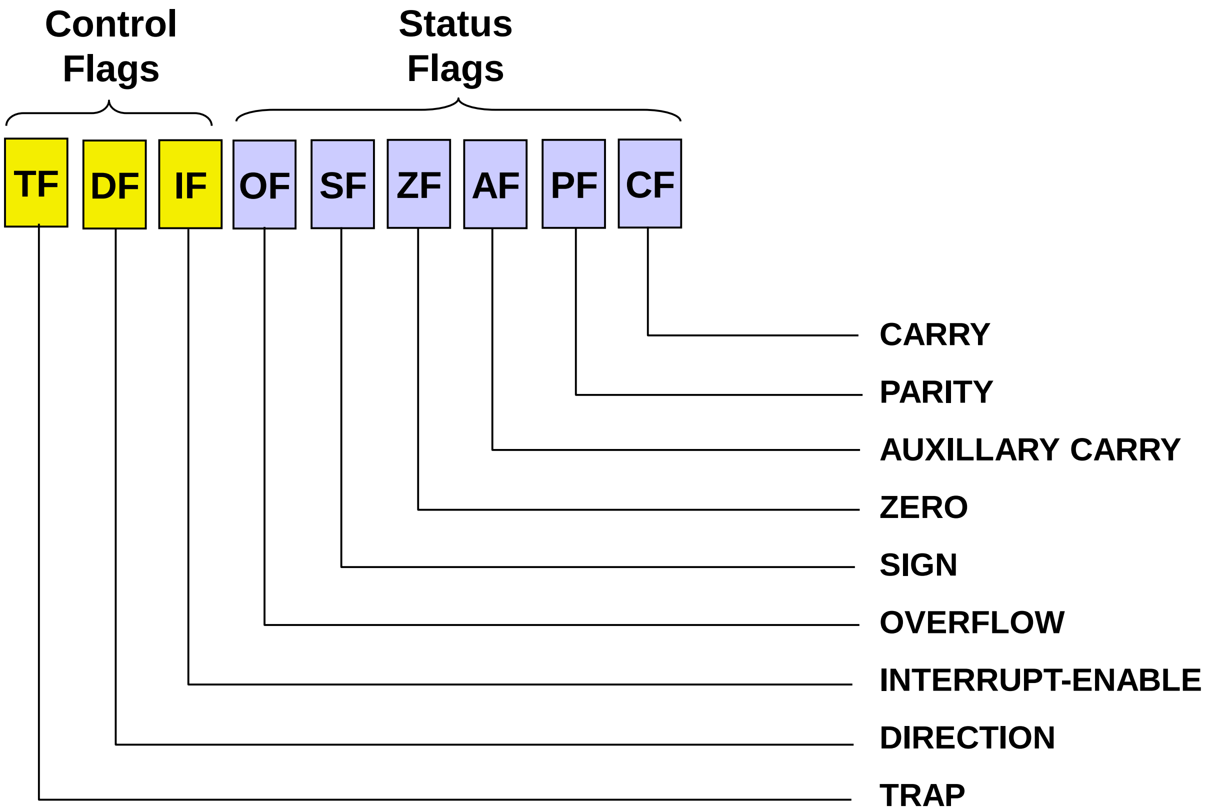
Multipurpose and Special-Purpose Registers of the Intel 8086 through the Pentium

Multipurpose and Special-Purpose Registers

- **All general registers of the 8088 microprocessor can be used for arithmetic and logic operations. The general registers are:**
  - **Accumulator register** consists of 2 8-bit registers AL and AH, which can be combined together and used as a 16-bit register AX. Accumulator can be used for I/O operations and string manipulation.
  - **Base register** consists of 2 8-bit registers BL and BH, which can be combined together and used as a 16-bit register BX. BX register usually contains a data pointer used for based, based indexed or register indirect addressing.
  - **Count register** consists of 2 8-bit registers CL and CH, which can be combined together and used as a 16-bit register CX. Count register can be used as a counter in string manipulation and shift/rotate instructions.
  - **Data register** consists of 2 8-bit registers DL and DH, which can be combined together and used as a 16-bit register DX. Data register can be used as a port number in I/O operations. In integer 32-bit multiply and divide instruction the DX register contains high-order word of the initial or resulting number.
- **The following registers are both general and index registers:**
  - **Stack Pointer (SP**) is a 16-bit register pointing to program stack.
  - **Base Pointer (BP)** is a 16-bit register pointing to data in stack segment. BP register is usually used for based, based indexed or register indirect addressing.
  - **Source Index (SI)** is a 16-bit register. SI is used for indexed, based indexed and register indirect addressing, as well as a source data address in string manipulation instructions.
  - **Destination Index (DI)** is a 16-bit register. DI is used for indexed, based indexed and register indirect addressing, as well as a destination data address in string manipulation instructions.
- **Other registers:**
  - **Instruction Pointer (IP)** is a 16-bit register.
  - **Flag Register**

The Flag Register

O	D	I	T	S	Z		A		P		C



Control Flags

Status Flags

TF DF IF OF SF ZF AF PF CF

CARRY

PARITY

AUXILLARY CARRY

ZERO

SIGN

OVERFLOW

INTERRUPT-ENABLE

DIRECTION

TRAP

The Flag Register

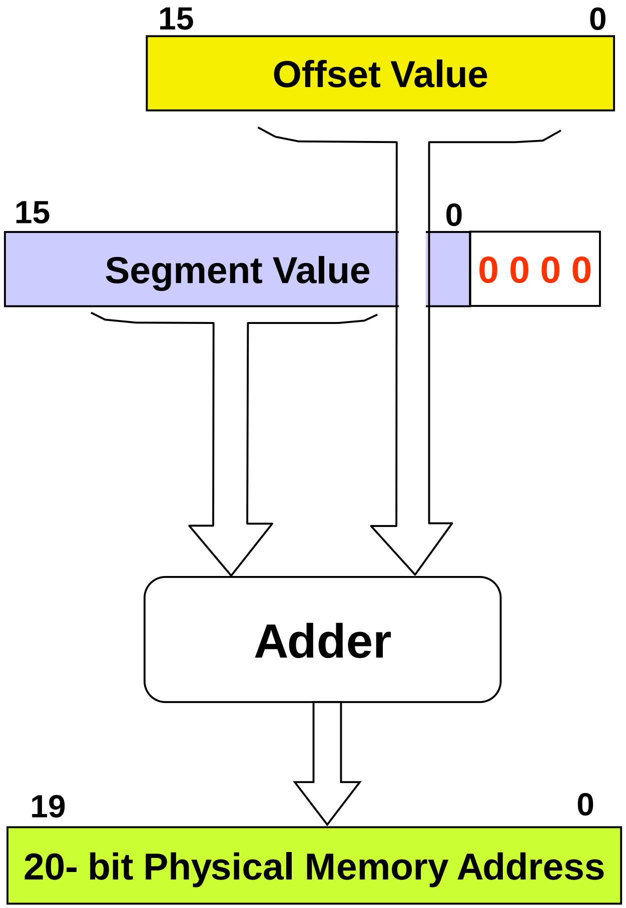
- **Overflow Flag (OF)** - set if the result is too large positive number, or is too small negative number to fit into destination operand.
- **Direction Flag (DF)** - if set (**STD**) then string manipulation instructions will auto-decrement index registers. If cleared (**CLD**) then the index registers will be auto-incremented.
- **Interrupt-enable Flag (IF)** - setting this bit (**STI**) enables maskable interrupts. Clearing this bit (**CLI**) disables maskable interrupts.
- **Single-step Flag (TF)** - if set then single-step interrupt will occur after the next instruction.
- **Sign Flag (SF)** - set if the most significant bit of the result is set.
- **Zero Flag (ZF)** - set if the result is zero.
- **Auxiliary carry Flag (AF)** - set if there was a carry from or borrow to bits 0-3 in the AL register.
- **Parity Flag (PF)** - set if parity (the number of "1" bits) in the low-order byte of the result is even.
- **Carry Flag (CF)** - set if there was a carry from or borrow to the most significant bit during last result calculation.

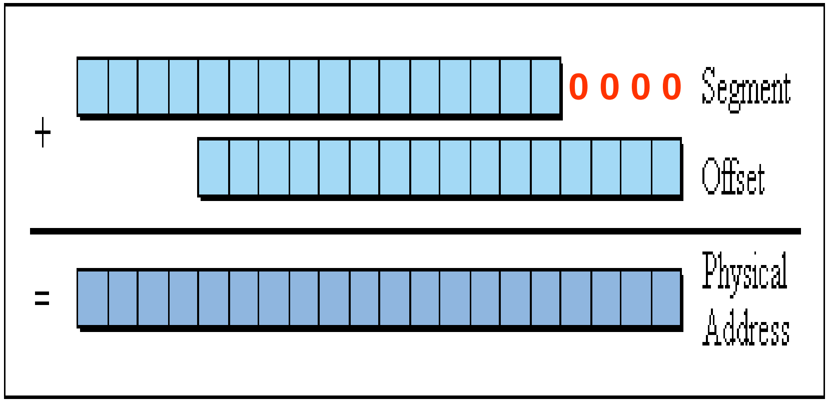
Segment Registers

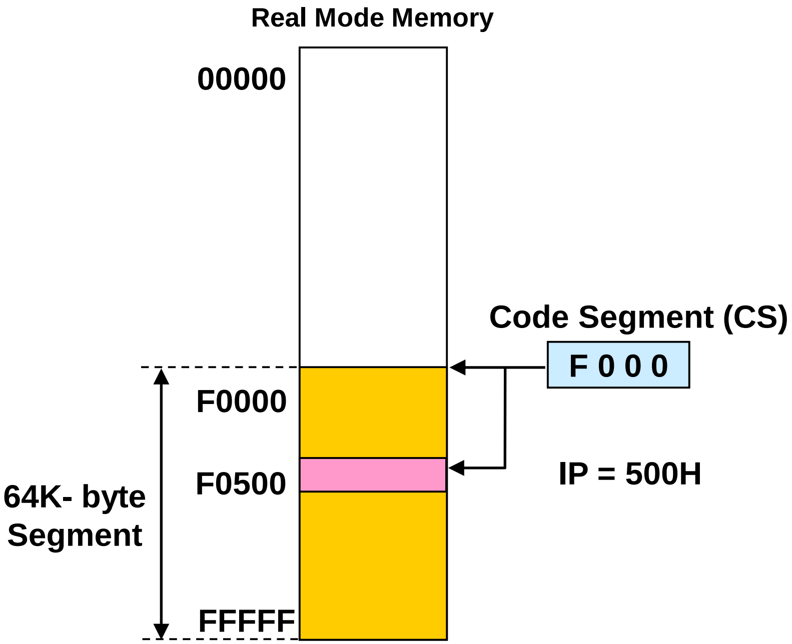
CS	Code
SS	Stack
DS	Data
ES	Extra

- There are four different 64 KB segments for instructions, stack, data and extra data. To specify where in 1 MB of processor memory these 4 segments are located the processor uses four segment registers:
- **Code segment (CS)** is a 16-bit register containing address of 64 KB segment with processor instructions. The processor uses CS segment for all accesses to instructions referenced by instruction pointer (IP) register.
- **Stack segment (SS)** is a 16-bit register containing address of 64KB segment with program stack. By default, the processor assumes that all data referenced by the stack pointer (SP) and base pointer (BP) registers is located in the stack segment.
- **Data segment (DS)** is a 16-bit register containing address of 64KB segment with program data. By default, the processor assumes that all data referenced by general registers (AX, BX, CX, DX) and index register (SI, DI) is located in the data segment.
- **Extra segment (ES)** is a 16-bit register containing address of 64KB segment, usually with program data. By default, the processor assumes that the DI register references the ES segment in string manipulation instructions.

Real Mode Memory Addressing



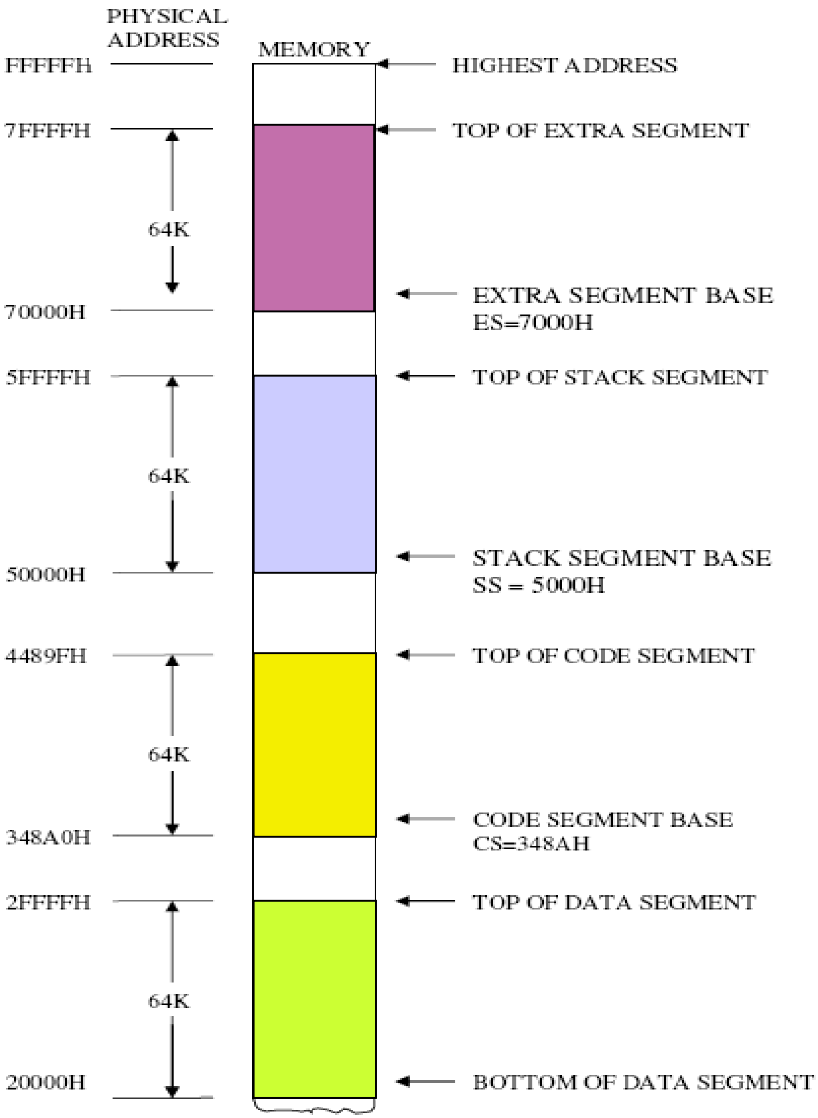




Real Mode Memory Addressing

CS = 348A	**Code**
SS = 5000	**Stack**
DS = 2000	**Data**
ES = 7000	**Extra**

Segment	Offset	Special Purpose
CS	IP	Instruction Address
SS	SP or BP	Stack Address
DS	BX, DI, SI, an 8-bit number, a 16-bit number	Data Address
ES	DI for string Instructions	String Destination Address



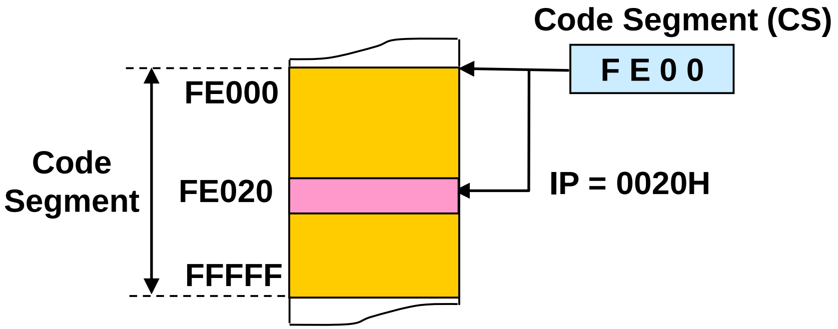
Real Mode Memory Addressing

Example 1:

CS = FE00H

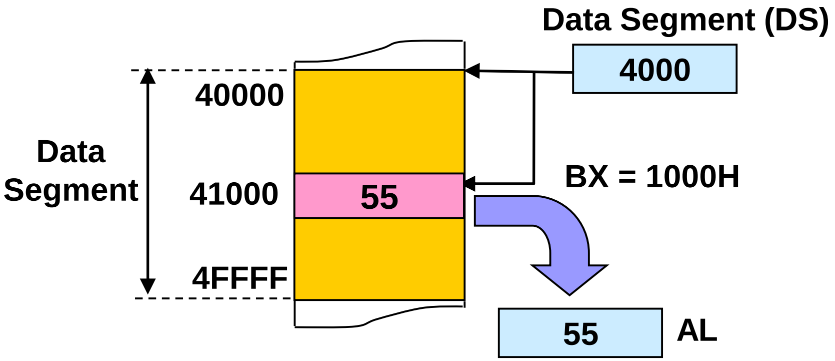
IP = 0020H

→ Physical Address = FE020H



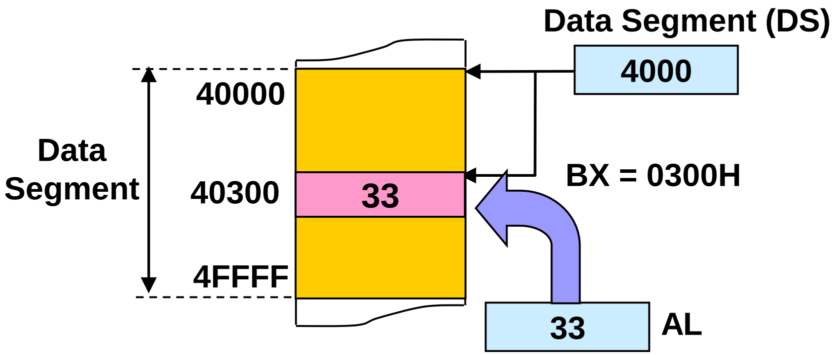
Example 2:

```
MOV     AX, 4000H
MOV     DS, AX
MOV     BX, 1000H
MOV     AL, [BX]
```



Example 3:

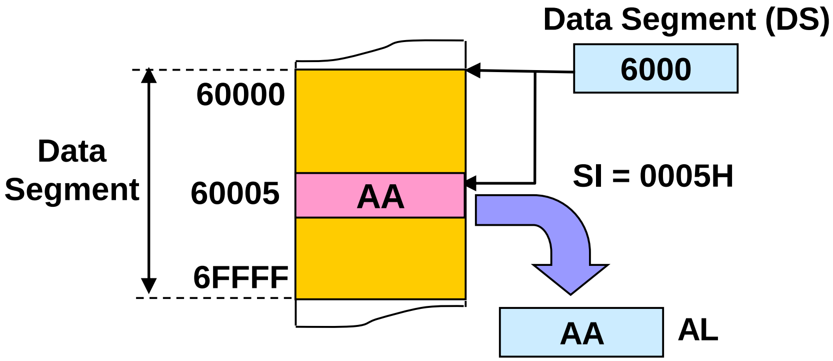
```
MOV     AX, 4000H
MOV     DS, AX
MOV     AL, 33H
MOV     BX, 0300H
MOV     [BX], AL
```



Real Mode Memory Addressing

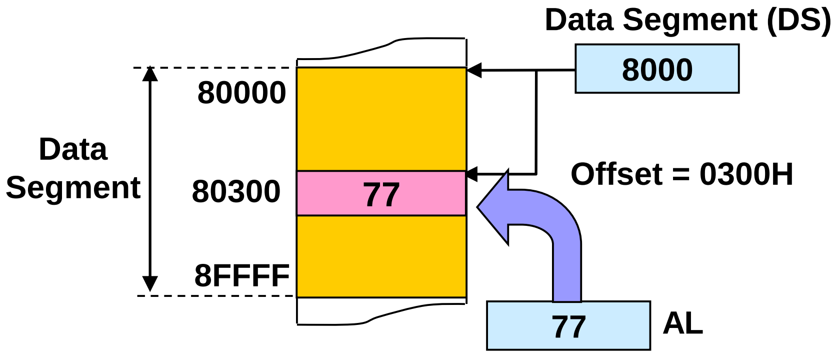
Example 4:

```
MOV     AX, 6000H
MOV     DS, AX
MOV     SI, 5H
MOV     AL, [SI]
```



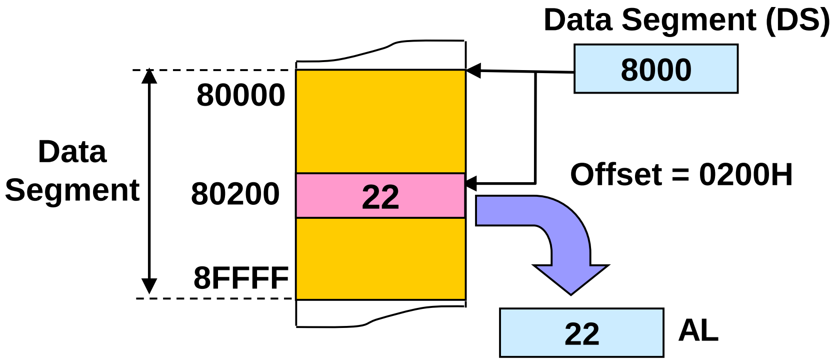
Example 5:

```
MOV     AX, 8000H
MOV     DS, AX
MOV     AL, 77H
MOV     [300], AL
```



Example 6:

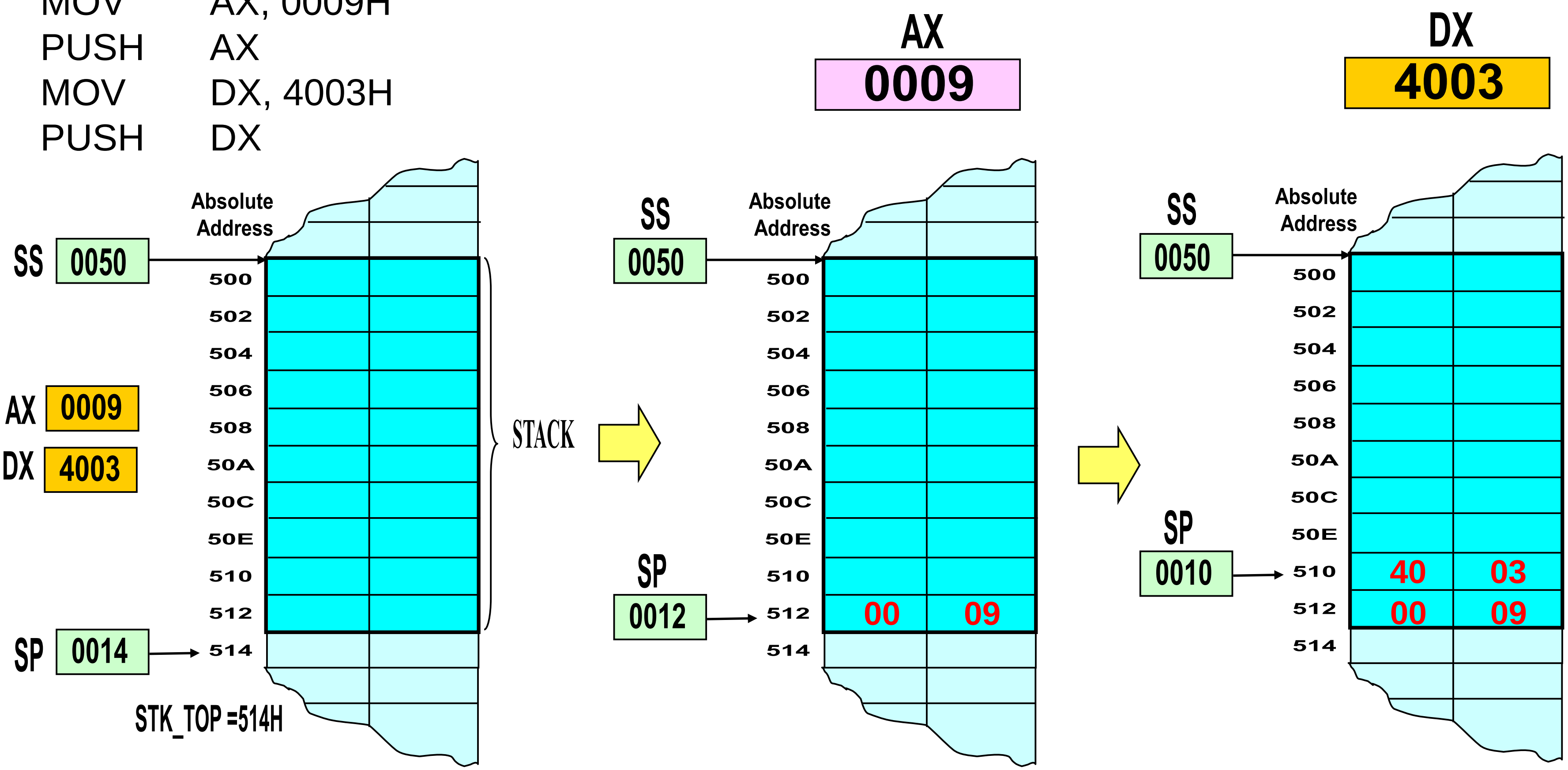
```
MOV     AX, 8000H
MOV     DS, AX
MOV     AL, [200]
```



Real Mode Memory Addressing

Example 7:

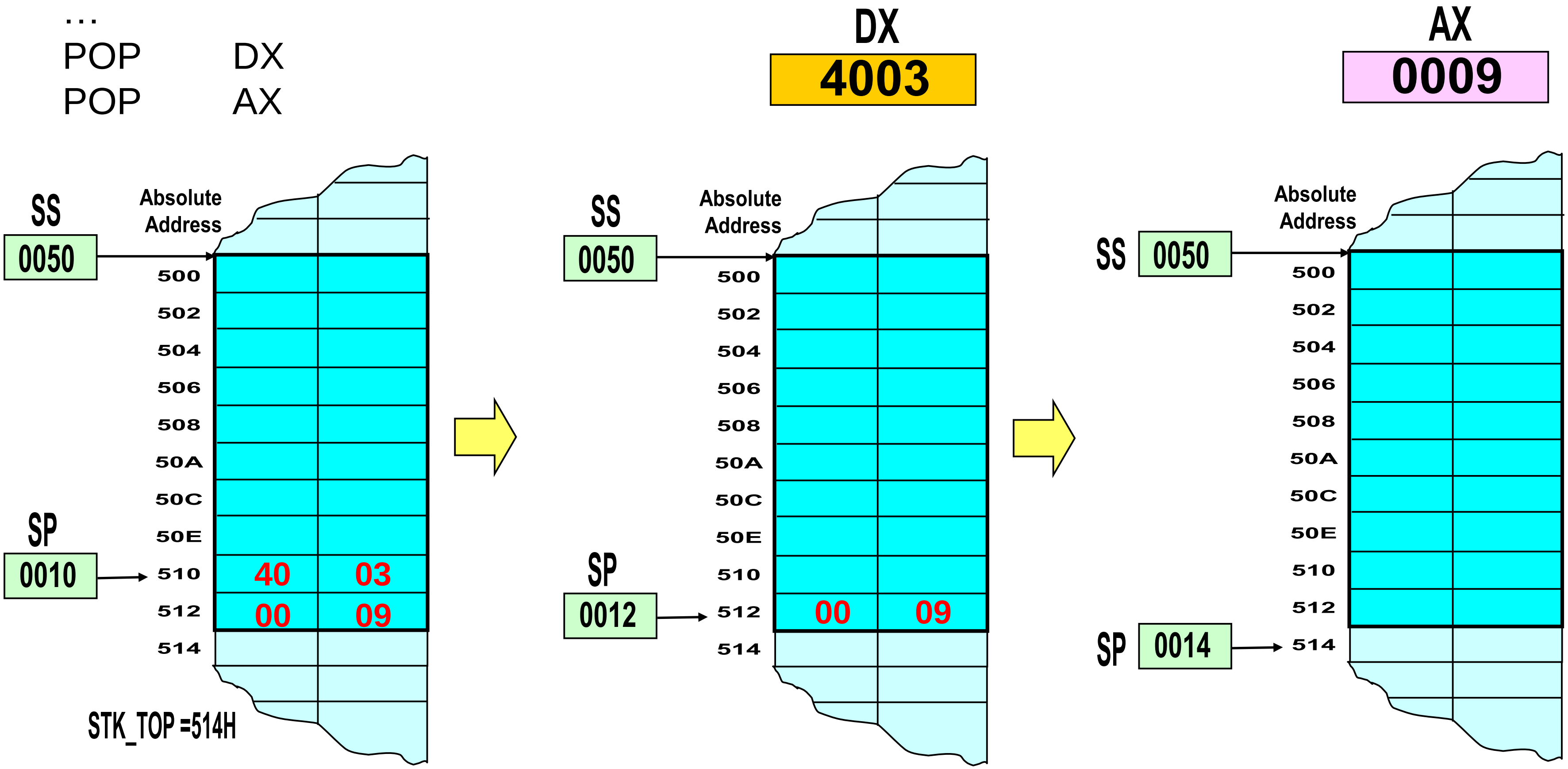
```
MOV     AX, 50H
MOV     SS, AX
MOV     SP, OFFSET STK_TOP
…
MOV     AX, 0009H
PUSH    AX
MOV     DX, 4003H
PUSH    DX
```



Real Mode Memory Addressing

Example 8:

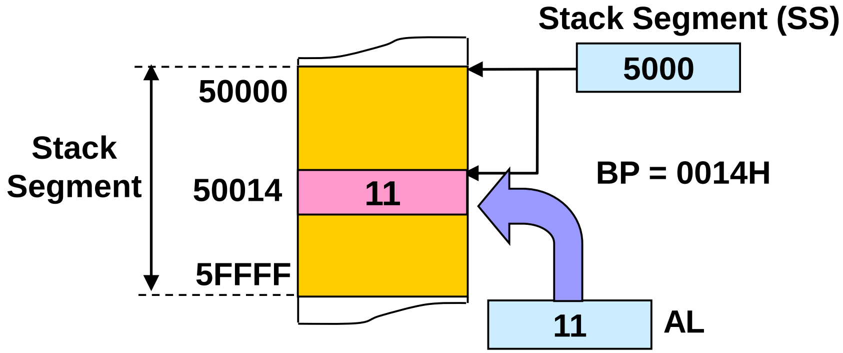
```
MOV     AX, 50H
MOV     SS, AX
MOV     SP, OFFSET STK_TOP
…
POP     DX
POP     AX
```



Real Mode Memory Addressing

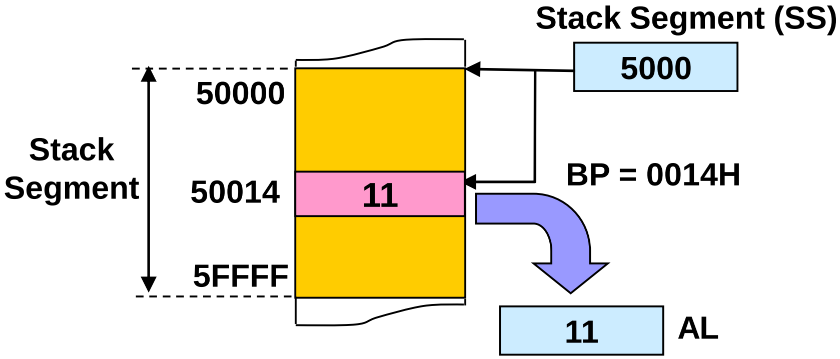
Example 9:

```
MOV     AX, 5000H
MOV     SS, AX
MOV     BP, 14H
MOV     [BP], AL
```



Example 10:

```
MOV     AX, 5000H
MOV     SS, AX
MOV     BP, 14H
MOV     AL, [BP]
```



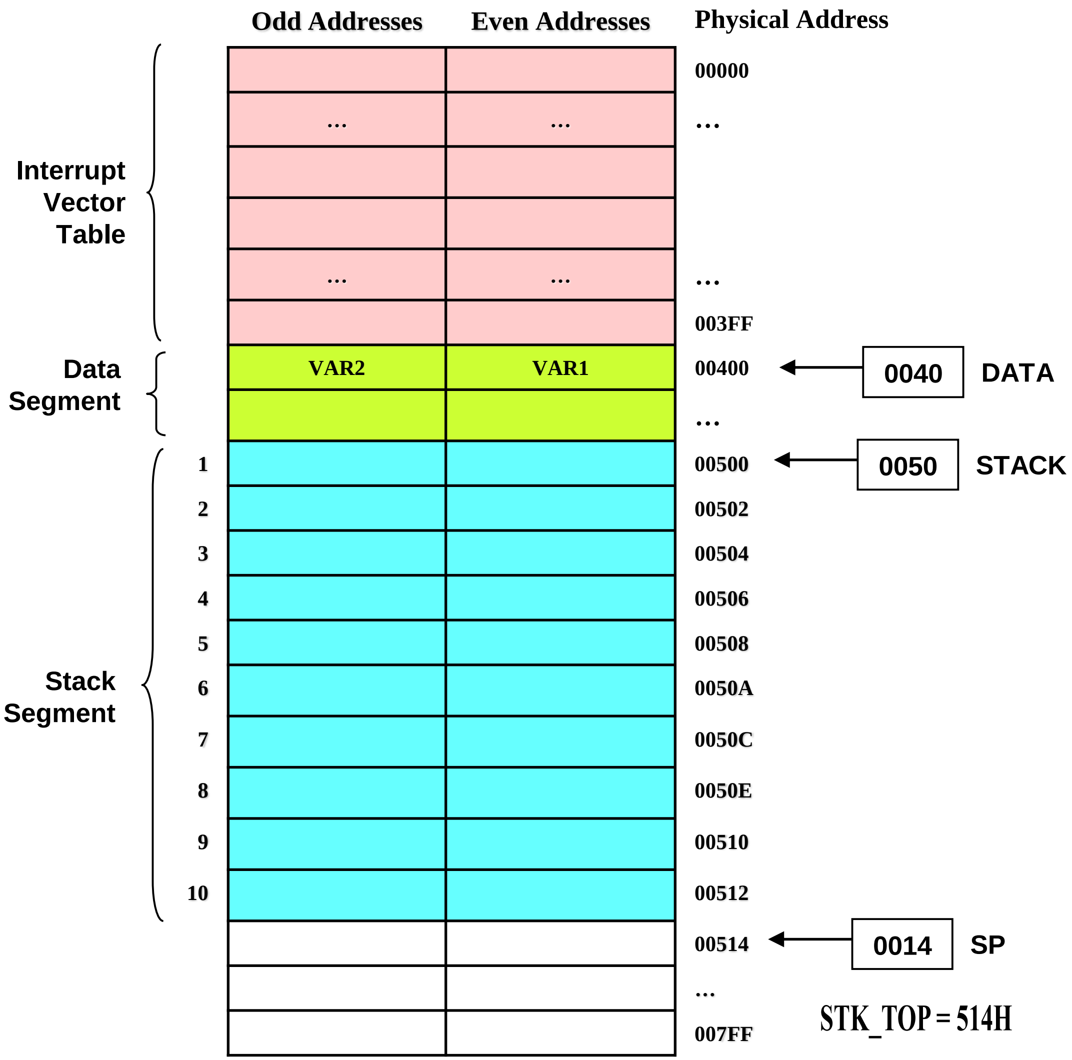
Structure of Assembly Programs

```
           NAME       PROJECT
DATA       SEGMENT  AT    40H
           ORG        0H
VAR1       DB         ?
VAR2       DB         ?
DATA       ENDS
;------------------------------------------------
STACK      SEGMENT  AT   50H
           DW         10 DUP(?)
STK_TOP    LABEL      WORD
STACK      ENDS
;------------------------------------------------
EPROM      SEGMENT  AT   0FE00H
           ASSUME   CS:EPROM, DS:DATA, SS:STACK
           ORG        0H
BEGIN      LABEL      FAR
           MOV        AX, DATA
           MOV        DS, AX



EPROM      ENDS
;-------------------------------------------------
CODE       SEGMENT  AT    0FFFFH
           ASSUME   CS:CODE, SS:STACK
           ORG        0H
START:     CLI
           MOV        AX,STACK
           MOV        SS,AX
           MOV        SP,OFFSET STK_TOP
           JMP        BEGIN
CODE       ENDS
END        START
```

RAM Mapping



	Odd Addresses	Even Addresses	Physical Address
			00000
	…	…	…
	…	…	…
			003FF
	VAR2	VAR1	00400
			…
1			00500
2			00502
3			00504
4			00506
5			00508
6			0050A
7			0050C
8			0050E
9			00510
10			00512
			00514
			…
			007FF

Interrupt Vector Table

Data Segment

Stack Segment

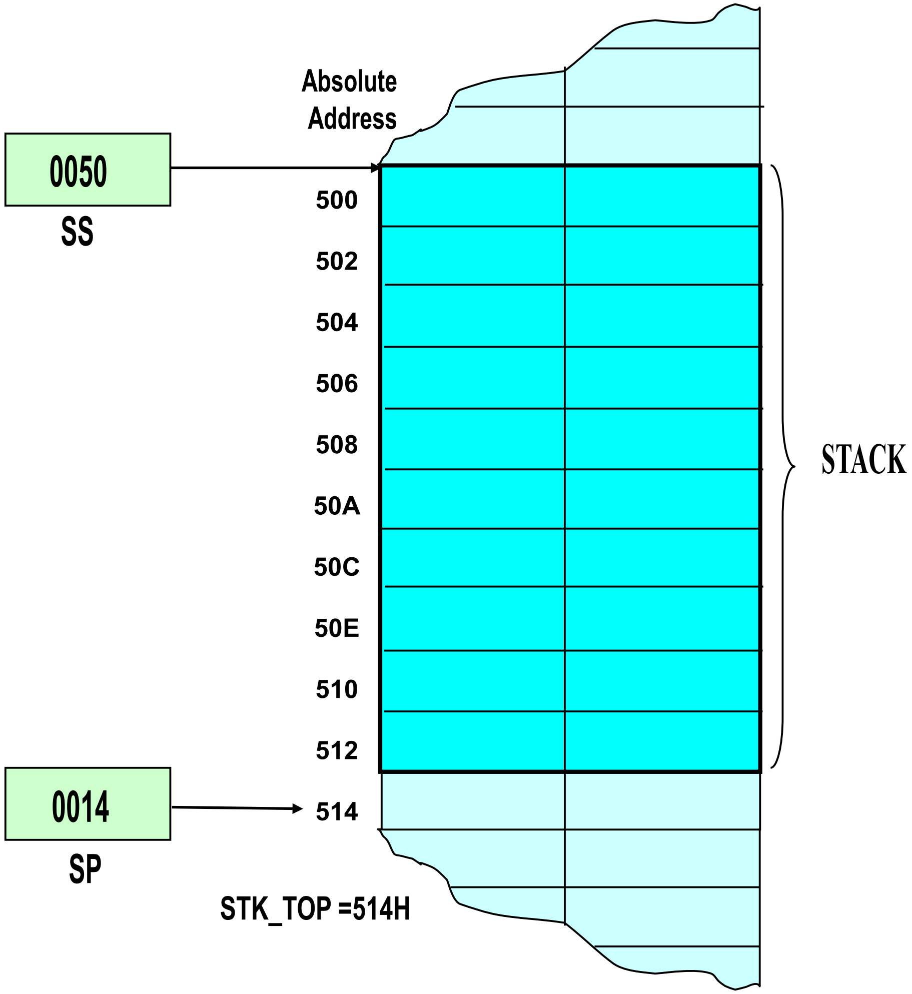
0040 DATA

0050 STACK

0014 SP

STK_TOP = 514H

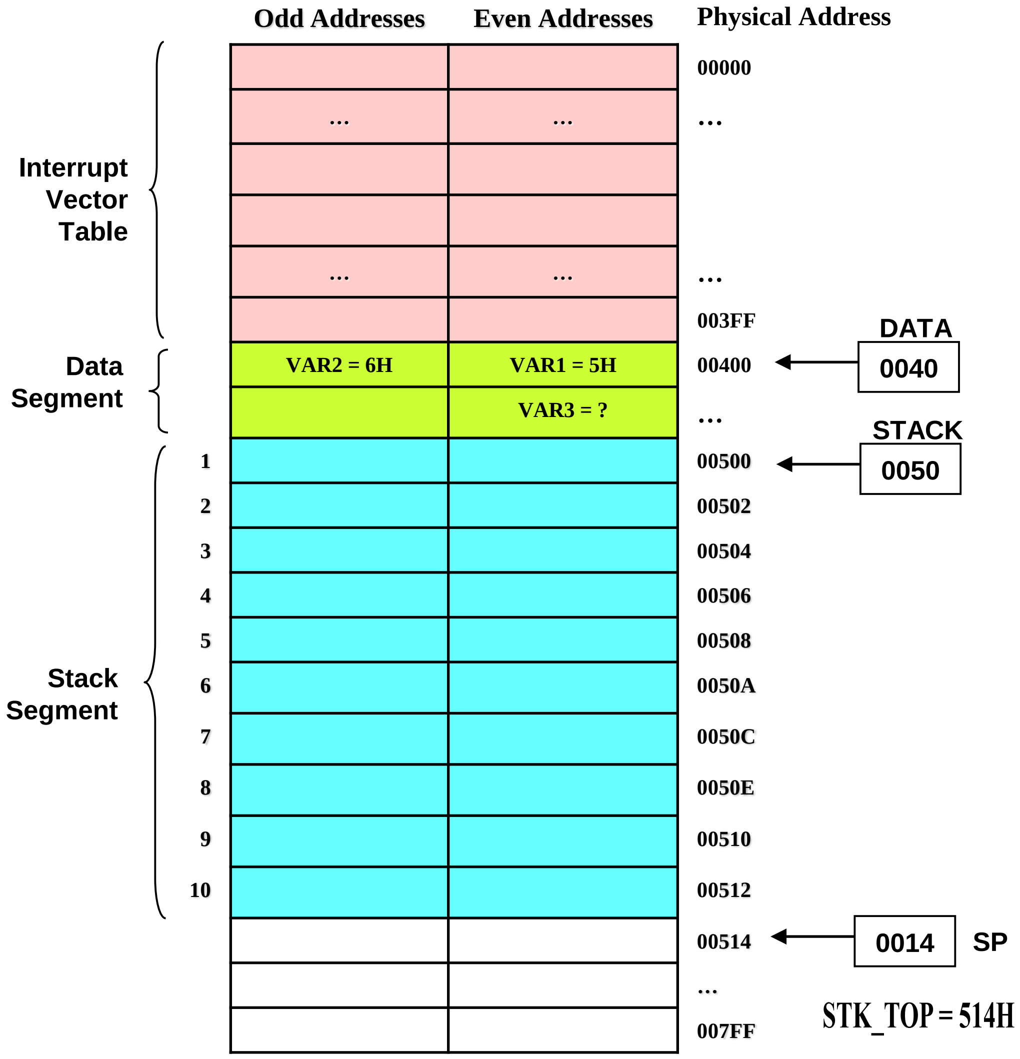
Stack Segment



Example

```
            NAME        PROJECT
DATA        SEGMENT  AT    40H
            ORG         0H
VAR1        DB          5H
VAR2        DB          6H
VAR3        DB          ?
DATA        ENDS
;------------------------------------------------
STACK       SEGMENT  AT   50H
            DW          10 DUP(?)
STK_TOP     LABEL       WORD
STACK       ENDS
;------------------------------------------------
EPROM       SEGMENT  AT   0FE00H
            ASSUME  CS:EPROM, DS:DATA, SS:STACK
            ORG         0H
BEGIN       LABEL       FAR
            MOV         AX, DATA
            MOV         DS, AX
            MOV         AL, VAR1
            ADD         AL, VAR2
            MOV         VAR3, AL
AGAIN:      JMP         AGAIN
EPROM       ENDS
;-------------------------------------------------
CODE        SEGMENT  AT    0FFFFH
            ASSUME  CS:CODE, SS:STACK
            ORG         0H
START:      CLI
            MOV         AX,STACK
            MOV         SS,AX
            MOV         SP,OFFSET STK_TOP
            JMP         BEGIN
CODE        ENDS
END         START
```

Example (Cont'd)



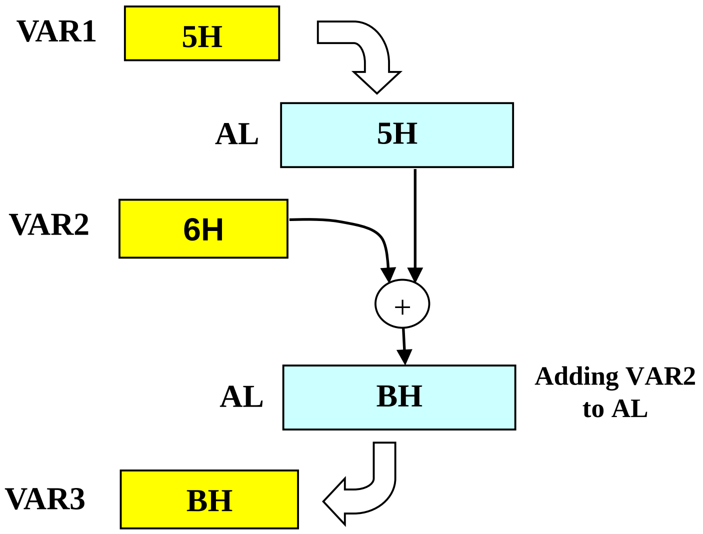
		Odd Addresses	Even Addresses	Physical Address
Interrupt Vector Table				00000
		…	…	…
		…	…	…
				003FF
Data Segment		VAR2 = 6H	VAR1 = 5H	00400
			VAR3 = ?	…
Stack Segment	1			00500
	2			00502
	3			00504
	4			00506
	5			00508
	6			0050A
	7			0050C
	8			0050E
	9			00510
	10			00512
				00514
				…
				007FF

DATA 0040

STACK 0050

0014 SP

STK_TOP = 514H



VAR1 5H

AL 5H

VAR2 6H

+

AL BH

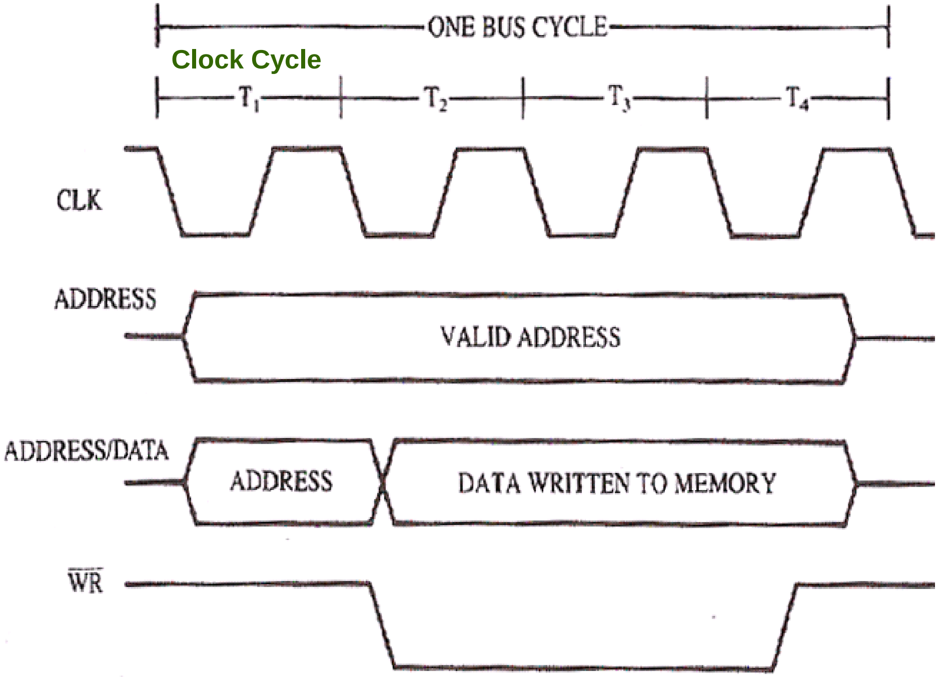
Adding VAR2 to AL

VAR3 BH

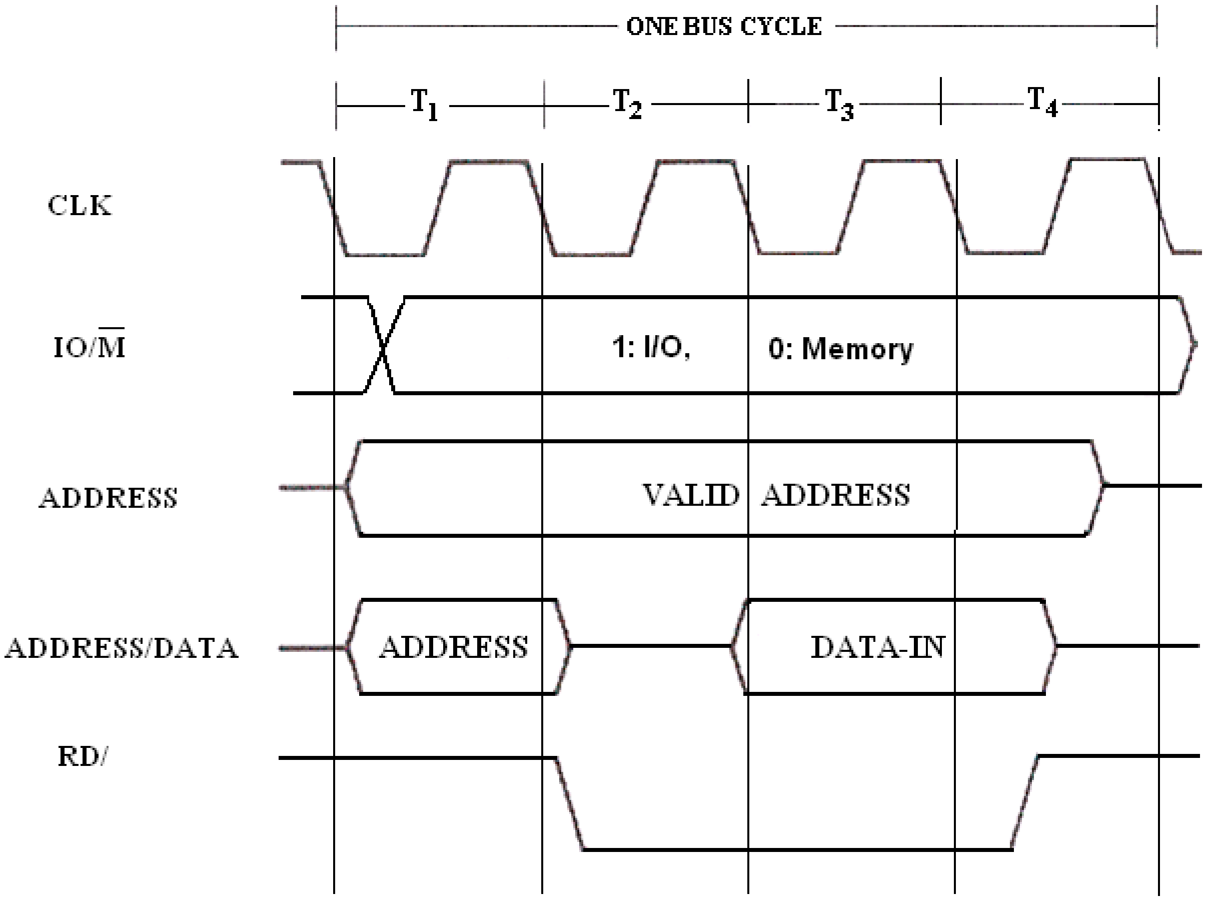
Description	Value
DS	40H
SS	50H
STK_TOP	514H
SP	14H
Absolute Address of VAR1	400H
Absolute Address of VAR3	402H
Value stored in VAR3 after executing the program	BH
EPROM Size	8KB

Bus Timing

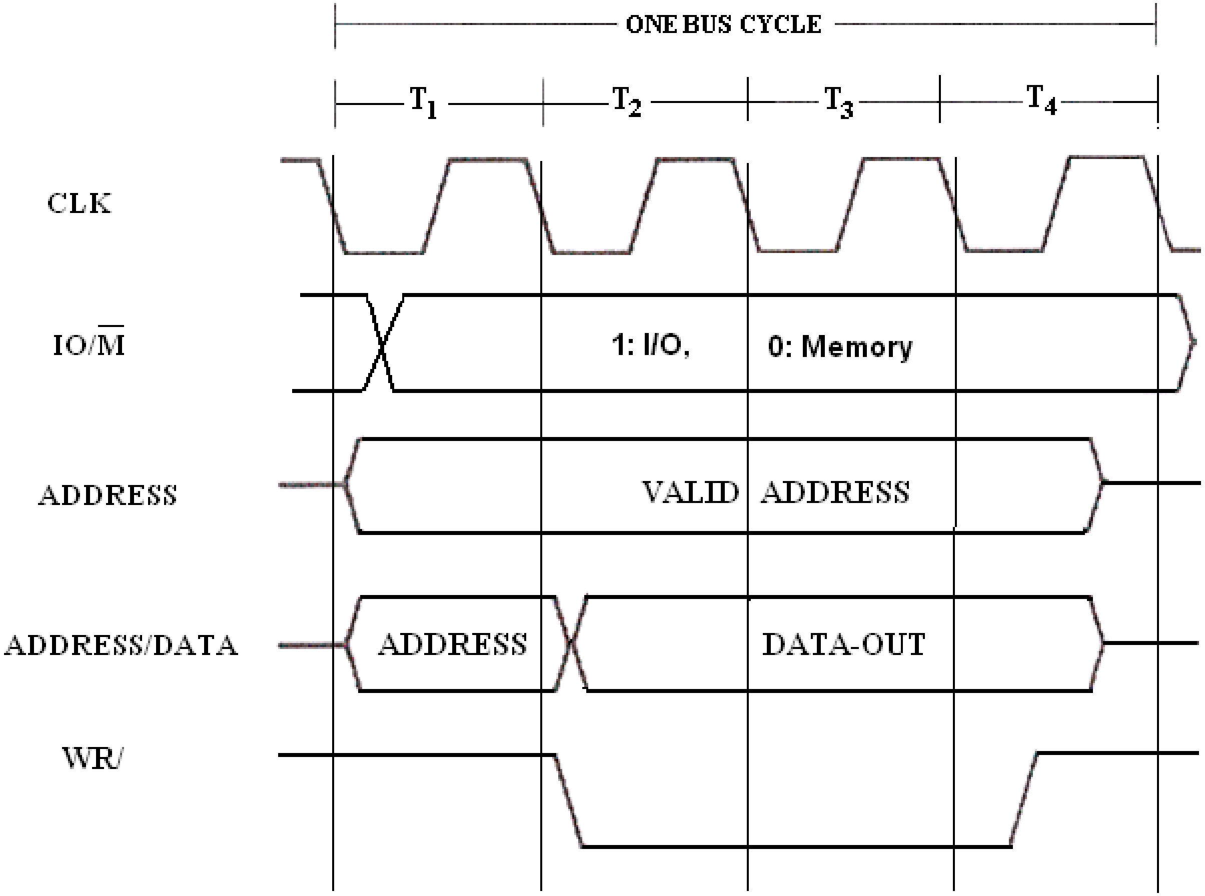
- A data transfer operation to/from the microprocessor occupies at least one bus cycle
- Each bus cycle consists of 4 clock cycles, T1, T2, T3, T4, each of period T
- With 5 MHz processor clock:
- T = 1/5 MHz = 0.2 ms
- Bus cycle = 4 T = 0.8 ms
- Max rate for memory and I/O transactions = 1/0.8 = 1.25 M operations per sec (Fetch speed).
- Processor executes 2.5 Million Instructions per sec (MIPS) (Execute speed)



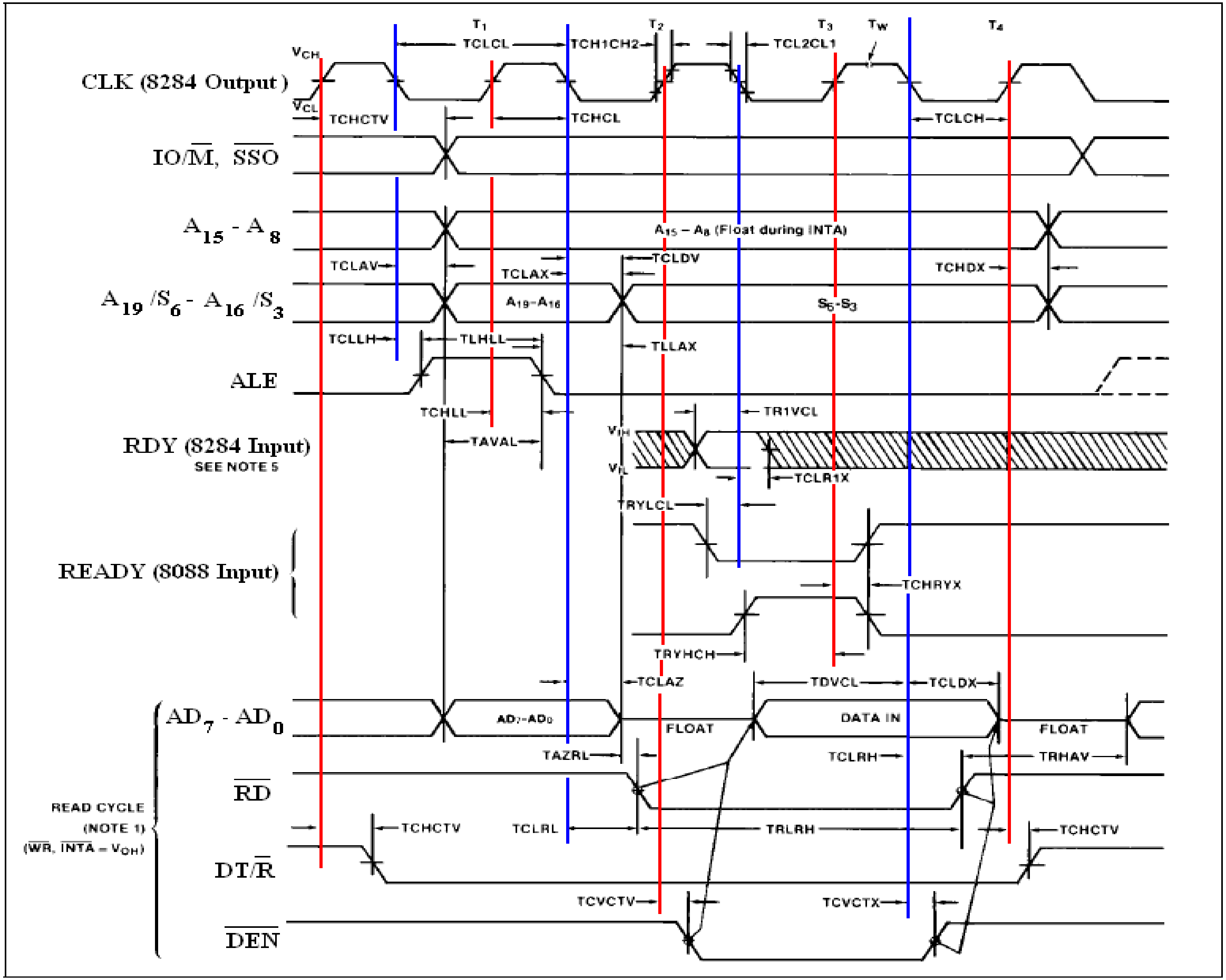
Timing Waveforms (Read Bus Cycle)



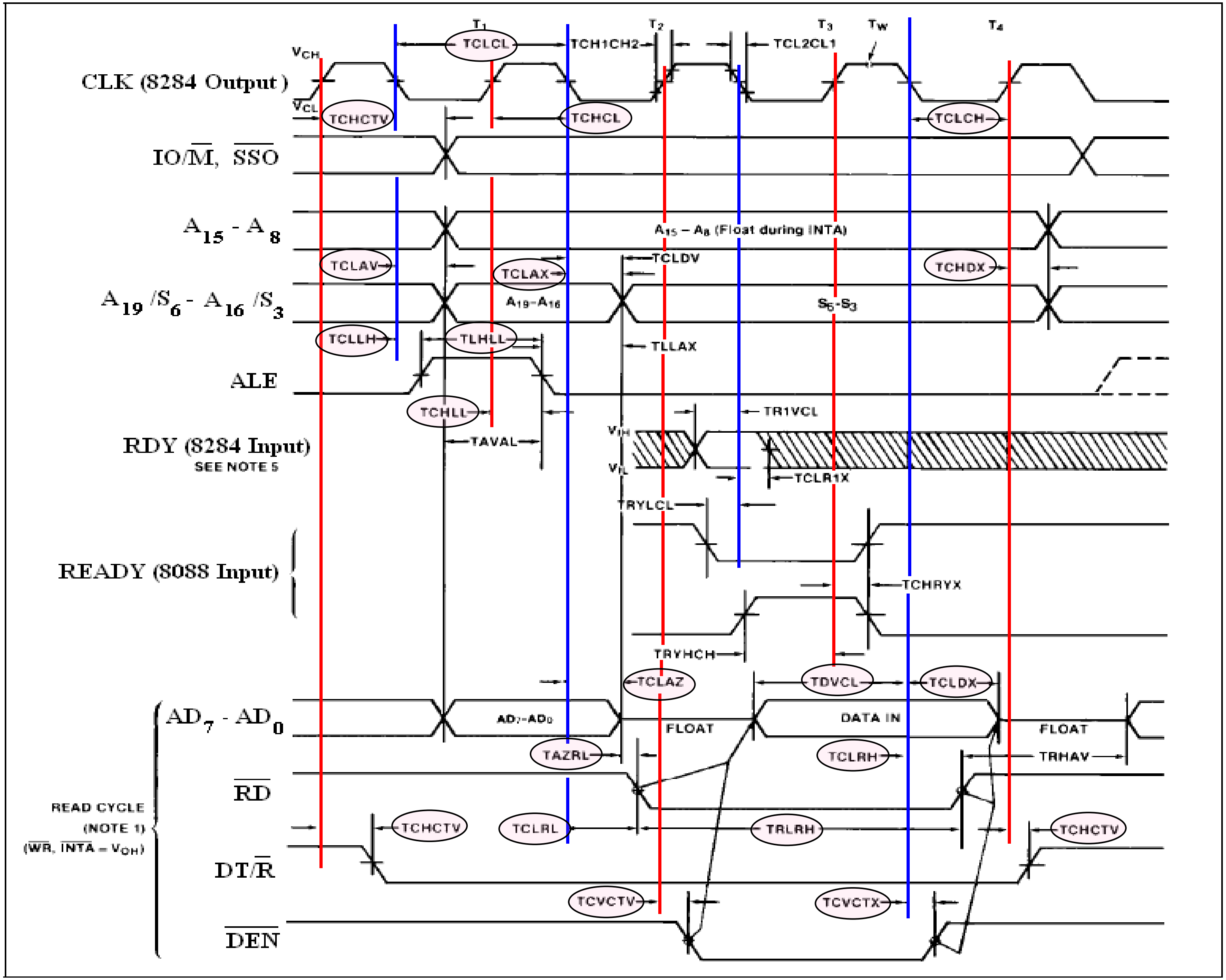
Timing Waveforms (Write Bus Cycle)



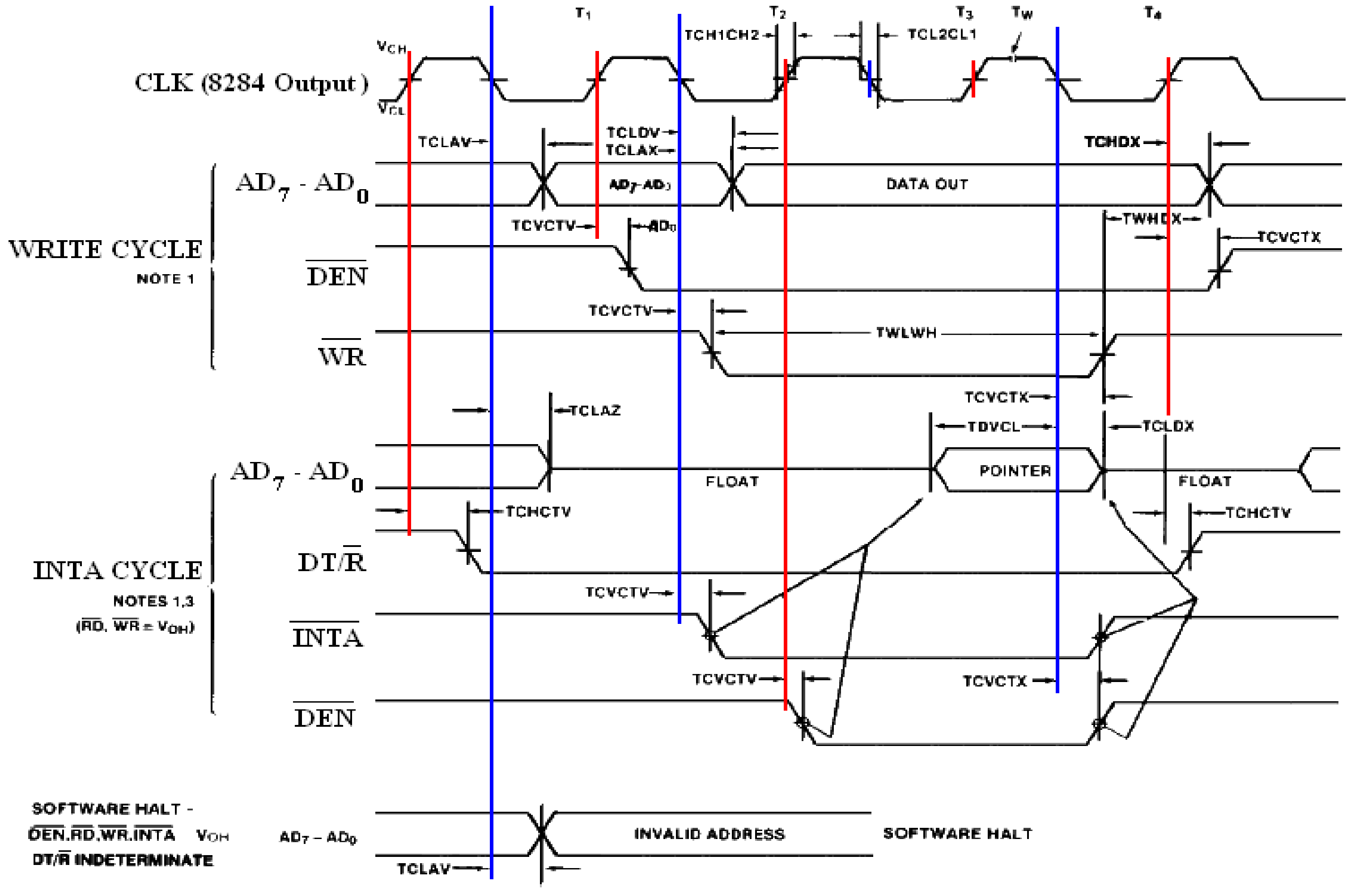
BUS TIMING—MINIMUM MODE SYSTEM



BUS TIMING—MINIMUM MODE SYSTEM



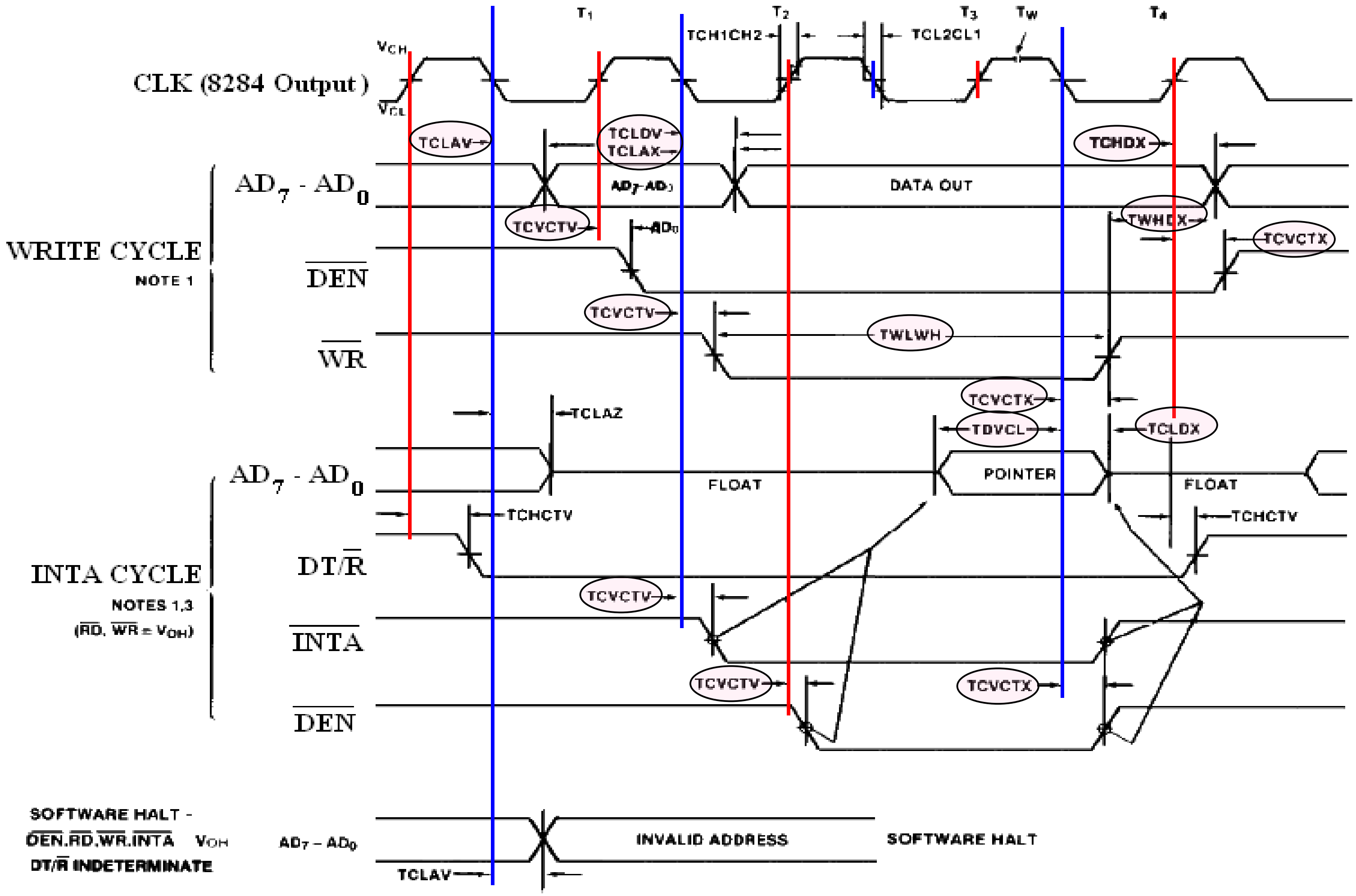
BUS TIMING—MINIMUM MODE SYSTEM (Continued)



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 8088 local ADDR/DATA bus is floating during both INTA cycles. Control signals are shown for the second INTA cycle.
4. Signals at 8284 are shown for reference only.
5. All timing measurements are made at 1.5V unless otherwise noted.

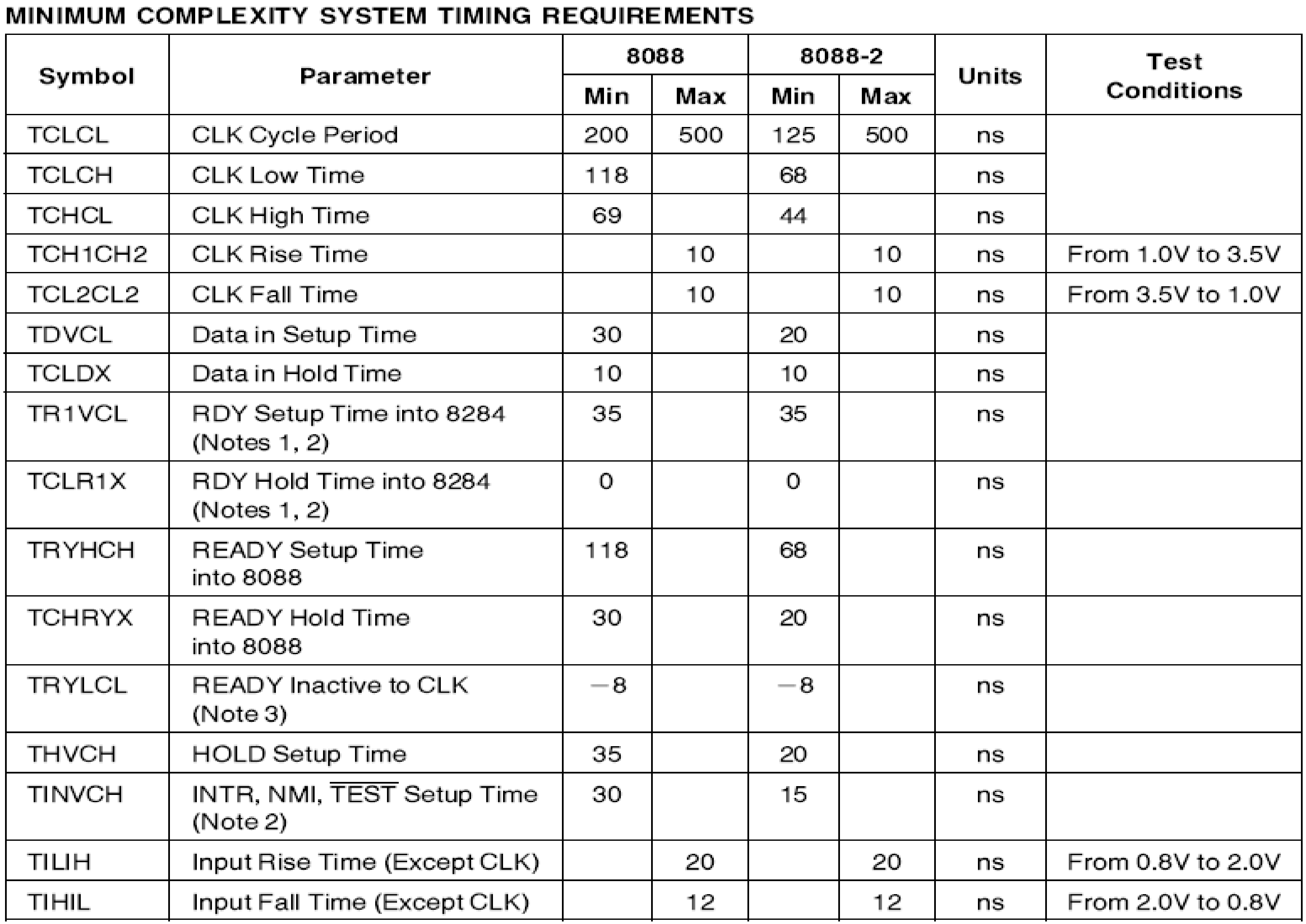
BUS TIMING—MINIMUM MODE SYSTEM (Continued)



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 8088 local ADDR/DATA bus is floating during both INTA cycles. Control signals are shown for the second INTA cycle.
4. Signals at 8284 are shown for reference only.
5. All timing measurements are made at 1.5V unless otherwise noted.

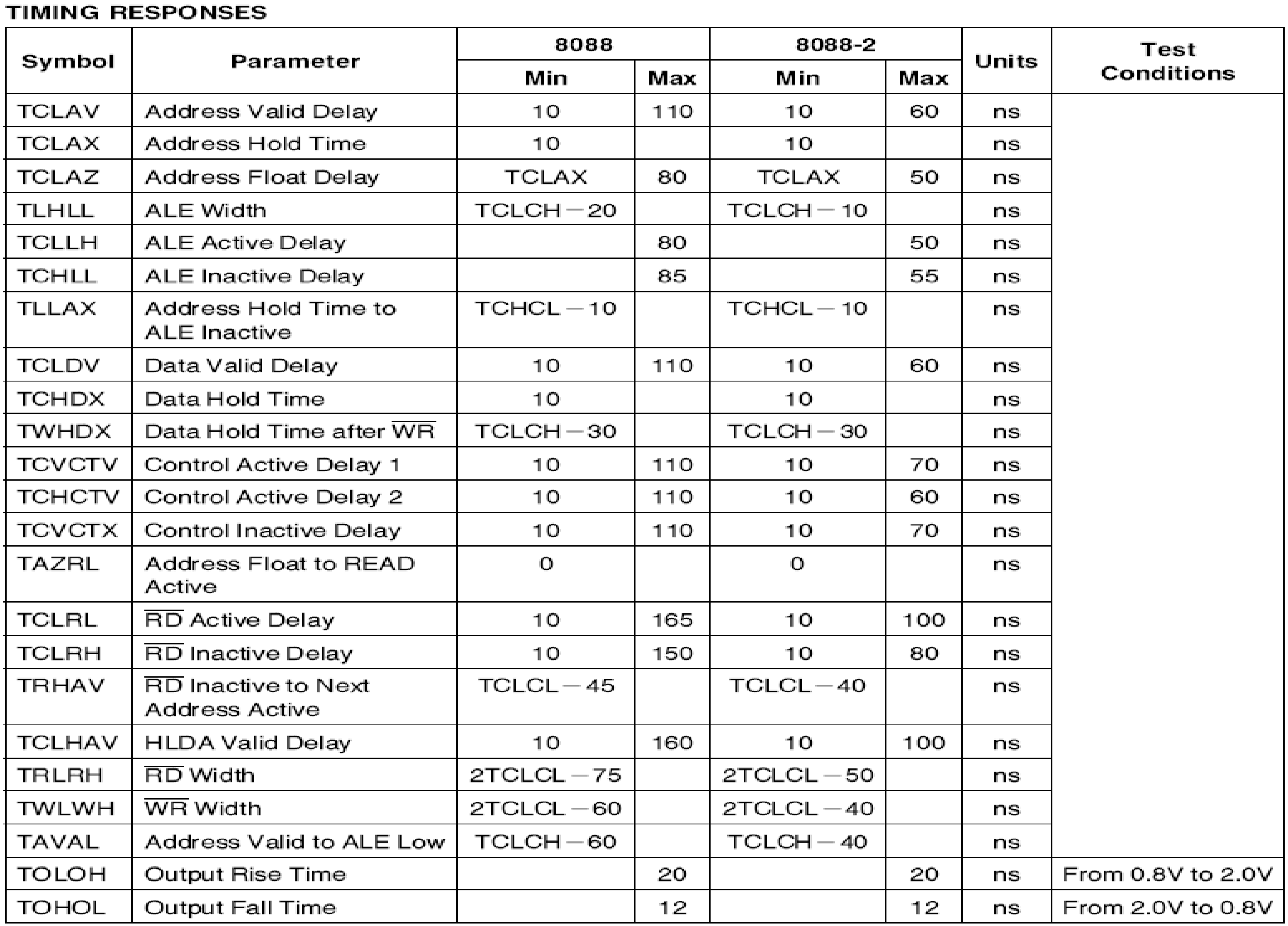
The 8088 Timing Parameters



MINIMUM COMPLEXITY SYSTEM TIMING REQUIREMENTS

Symbol	Parameter	8088		8088-2		Units	Test Conditions
		Min	Max	Min	Max		
TCLCL	CLK Cycle Period	200	500	125	500	ns	
TCLCH	CLK Low Time	118		68		ns	
TCHCL	CLK High Time	69		44		ns	
TCH1CH2	CLK Rise Time		10		10	ns	From 1.0V to 3.5V
TCL2CL2	CLK Fall Time		10		10	ns	From 3.5V to 1.0V
TDVCL	Data in Setup Time	30		20		ns	
TCLDX	Data in Hold Time	10		10		ns	
TR1VCL	RDY Setup Time into 8284 (Notes 1, 2)	35		35		ns	
TCLR1X	RDY Hold Time into 8284 (Notes 1, 2)	0		0		ns	
TRYHCH	READY Setup Time into 8088	118		68		ns	
TCHRYX	READY Hold Time into 8088	30		20		ns	
TRYLCL	READY Inactive to CLK (Note 3)	−8		−8		ns	
THVCH	HOLD Setup Time	35		20		ns	
TINVCH	INTR, NMI, $\overline{\text{TEST}}$ Setup Time (Note 2)	30		15		ns	
TILIH	Input Rise Time (Except CLK)		20		20	ns	From 0.8V to 2.0V
TIHIL	Input Fall Time (Except CLK)		12		12	ns	From 2.0V to 0.8V

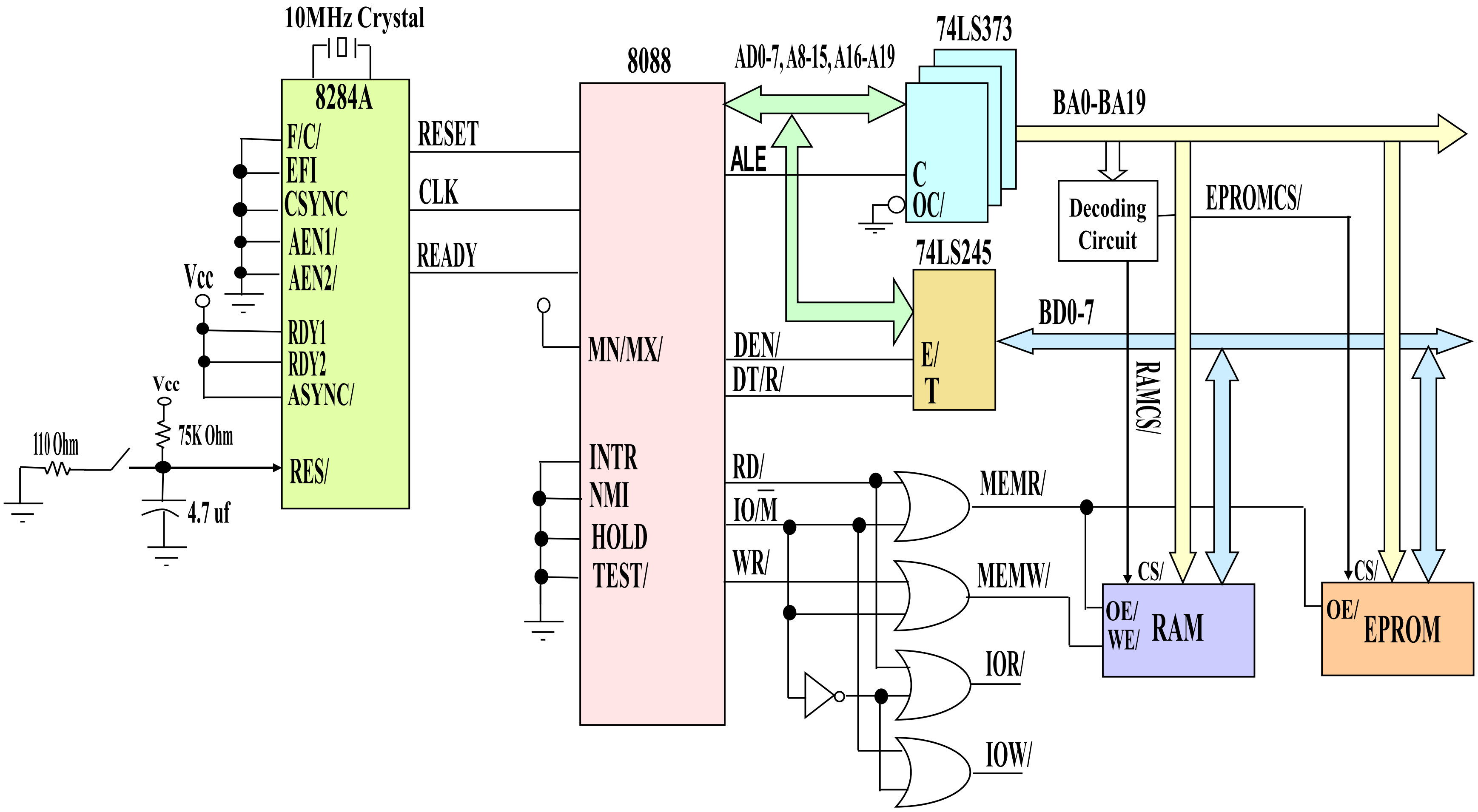
The 8088 Timing Parameters



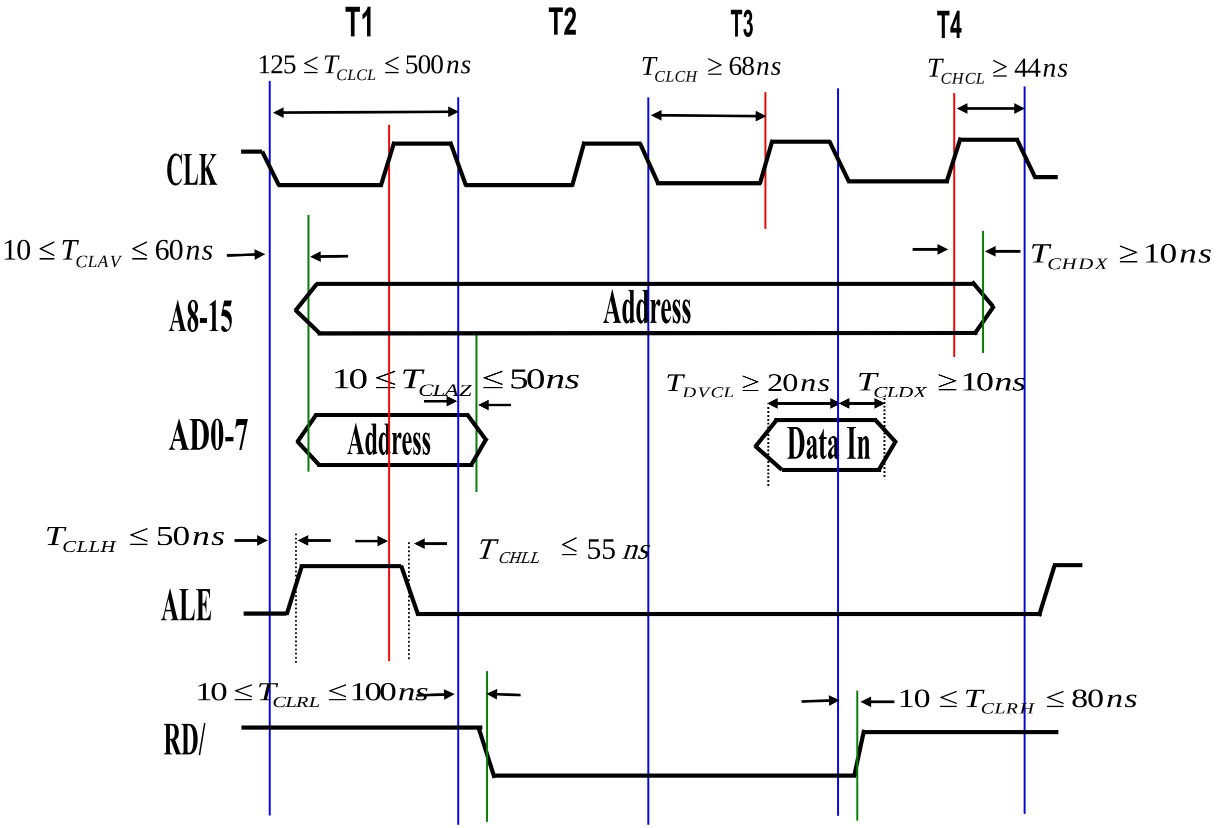
TIMING RESPONSES

Symbol	Parameter	8088		8088-2		Units	Test Conditions
		Min	Max	Min	Max		
TCLAV	Address Valid Delay	10	110	10	60	ns	
TCLAX	Address Hold Time	10		10		ns	
TCLAZ	Address Float Delay	TCLAX	80	TCLAX	50	ns	
TLHLL	ALE Width	TCLCH – 20		TCLCH – 10		ns	
TCLLH	ALE Active Delay		80		50	ns	
TCHLL	ALE Inactive Delay		85		55	ns	
TLLAX	Address Hold Time to ALE Inactive	TCHCL – 10		TCHCL – 10		ns	
TCLDV	Data Valid Delay	10	110	10	60	ns	
TCHDX	Data Hold Time	10		10		ns	
TWHDX	Data Hold Time after $\overline{\text{WR}}$	TCLCH – 30		TCLCH – 30		ns	
TCVCTV	Control Active Delay 1	10	110	10	70	ns	
TCHCTV	Control Active Delay 2	10	110	10	60	ns	
TCVCTX	Control Inactive Delay	10	110	10	70	ns	
TAZRL	Address Float to READ Active	0		0		ns	
TCLRL	$\overline{\text{RD}}$ Active Delay	10	165	10	100	ns	
TCLRH	$\overline{\text{RD}}$ Inactive Delay	10	150	10	80	ns	
TRHAV	$\overline{\text{RD}}$ Inactive to Next Address Active	TCLCL – 45		TCLCL – 40		ns	
TCLHAV	HLDA Valid Delay	10	160	10	100	ns	
TRLRH	$\overline{\text{RD}}$ Width	2TCLCL – 75		2TCLCL – 50		ns	
TWLWH	$\overline{\text{WR}}$ Width	2TCLCL – 60		2TCLCL – 40		ns	
TAVAL	Address Valid to ALE Low	TCLCH – 60		TCLCH – 40		ns	
TOLOH	Output Rise Time		20		20	ns	From 0.8V to 2.0V
TOHOL	Output Fall Time		12		12	ns	From 2.0V to 0.8V

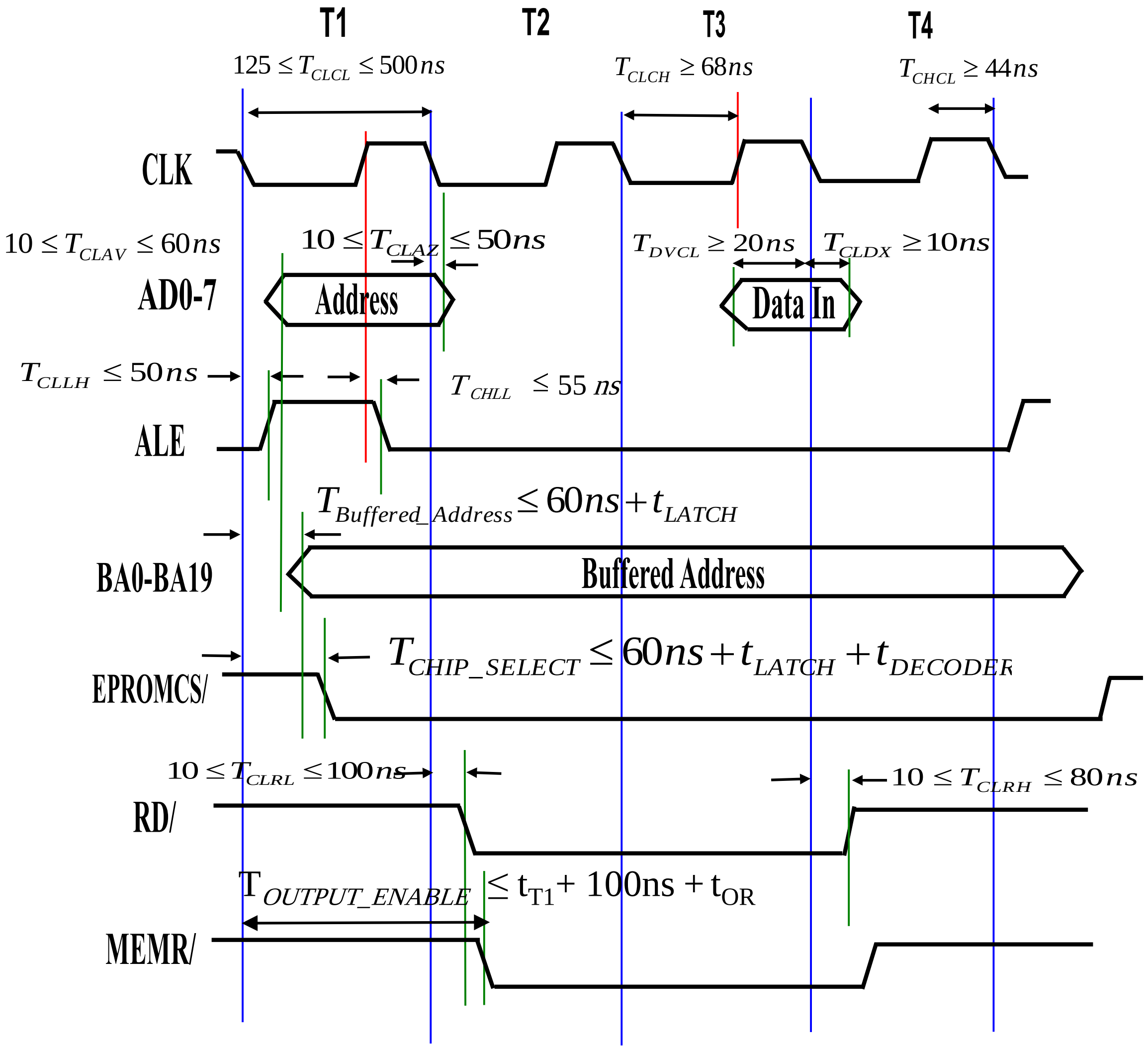
Minimum Mode 8088 System



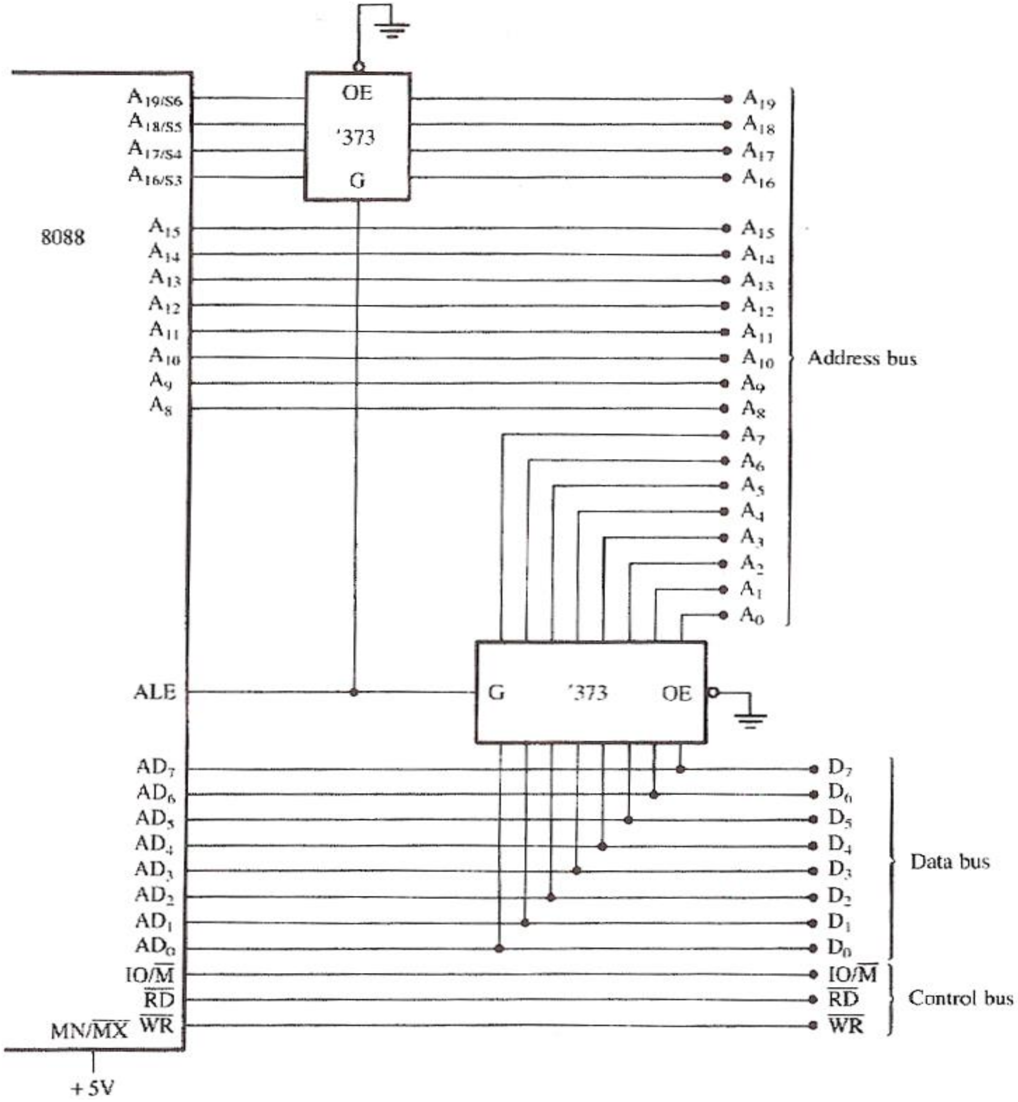
The 8088-2 Timing Waveforms (Minimum Mode)



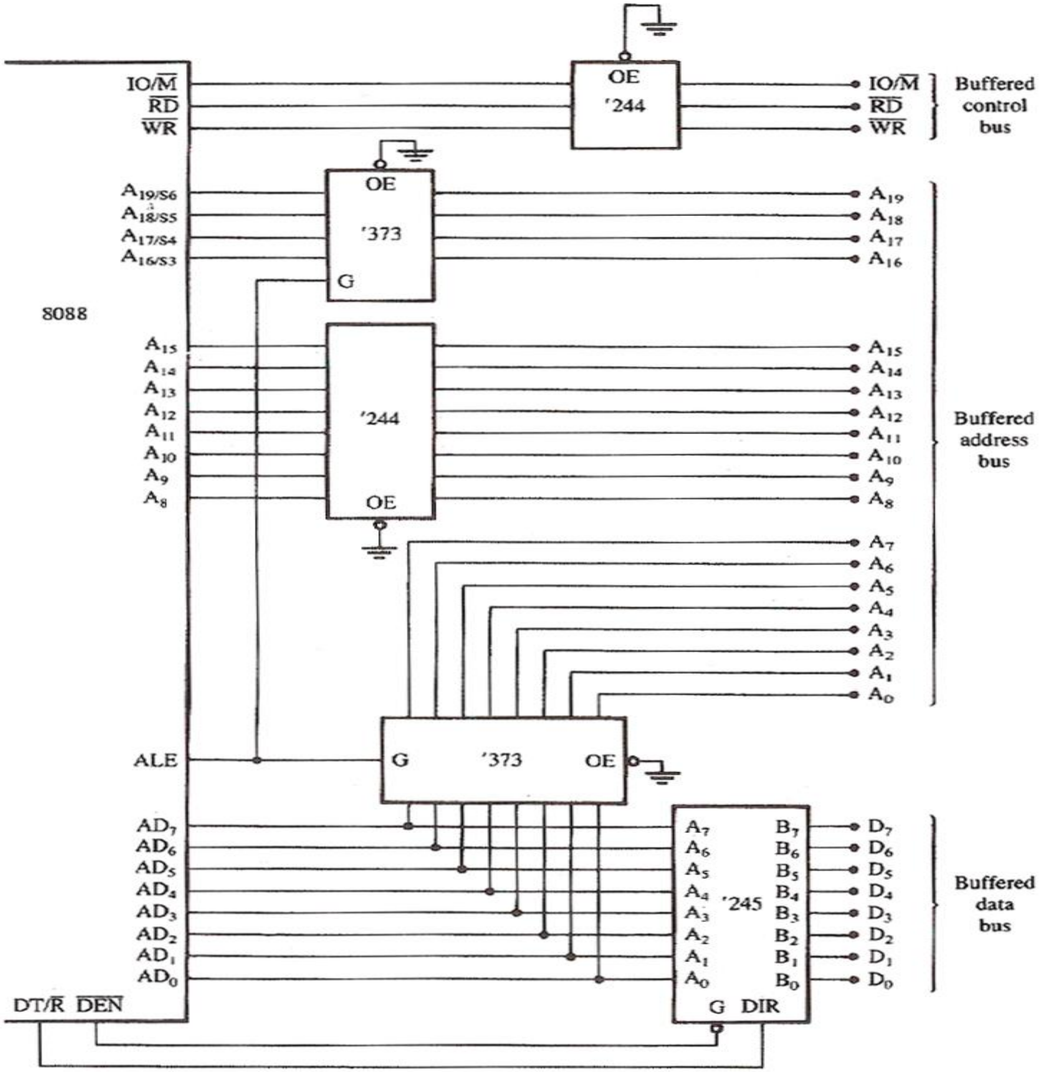
Interfacing EPROM to 8088-2


T1
T2
T3
T4
$125 \le T_{CLCL} \le 500ns$
$T_{CLCH} \ge 68ns$
$T_{CHCL} \ge 44ns$
CLK
$10 \le T_{CLAV} \le 60ns$
$10 \le T_{CLAZ} \le 50ns$
$T_{DVCL} \ge 20ns$
$T_{CLDX} \ge 10ns$
AD0-7
Address
Data In
$T_{CLLH} \le 50ns$
$T_{CHLL} \le 55\ ns$
ALE
$T_{Buffered\_Address} \le 60ns + t_{LATCH}$
BA0-BA19
Buffered Address
$T_{CHIP\_SELECT} \le 60ns + t_{LATCH} + t_{DECODER}$
EPROMCS/
$10 \le T_{CLRL} \le 100ns$
$10 \le T_{CLRH} \le 80ns$
RD/
$T_{OUTPUT\_ENABLE} \le t_{T1} + 100ns + t_{OR}$
MEMR/

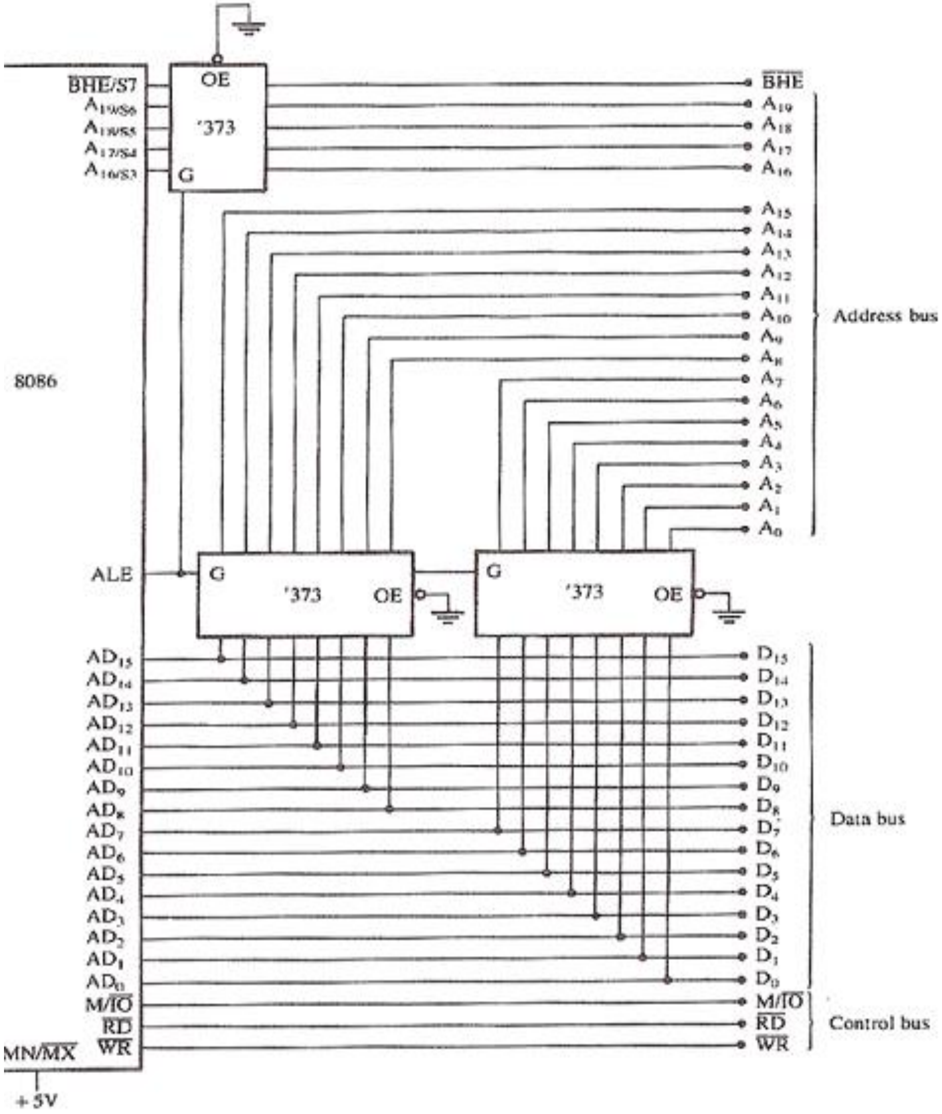
Demultiplexing the 8088 Microprocessor



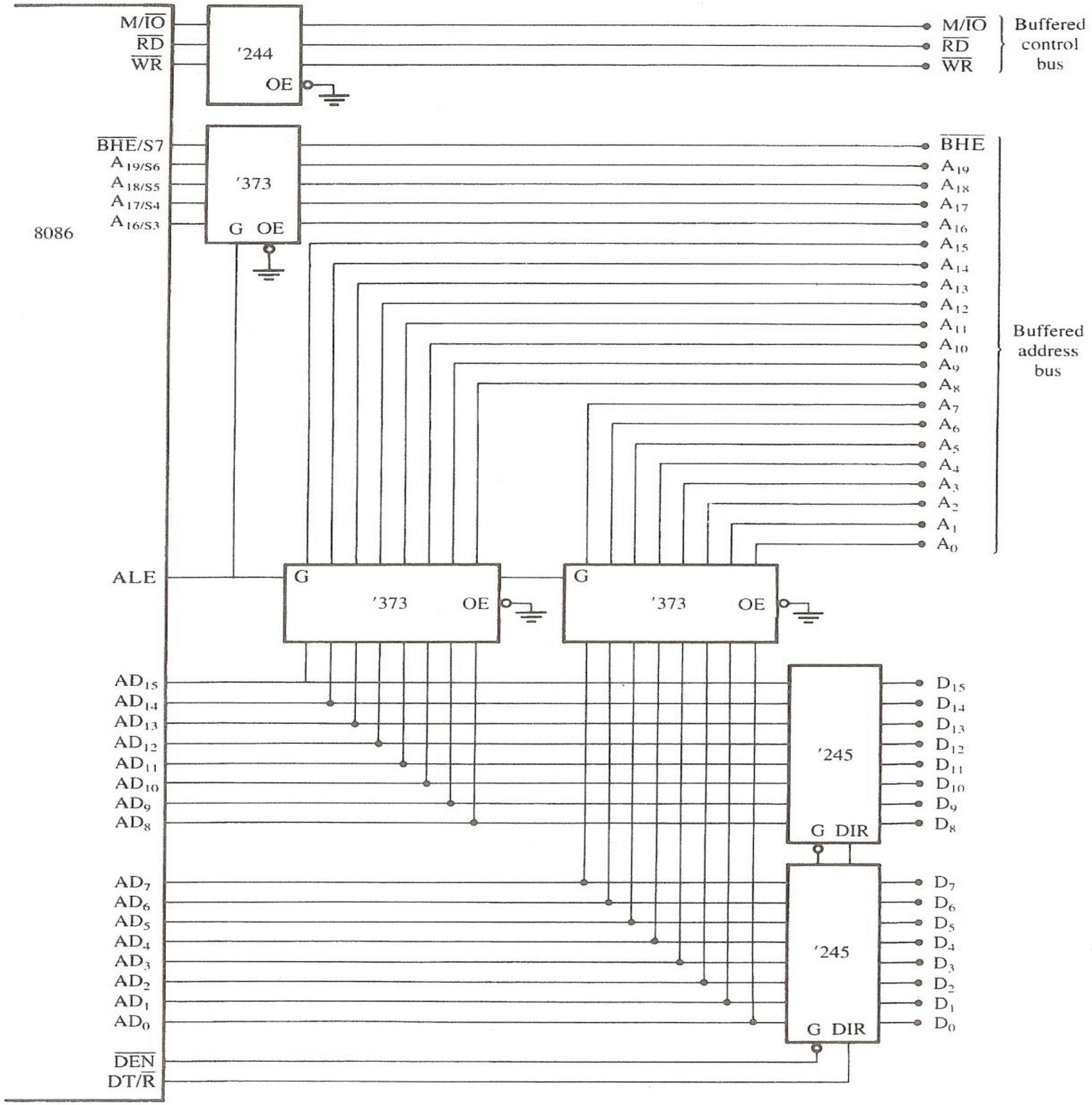
Fully Buffered 8088 Microprocessor



Demultiplexing the 8086 Microprocessor



Fully Buffered 8086 Microprocessor



Buffering

- Since the microprocessor output pins provide minimum drive current at the 0 logic level, buffering is often needed if more TTL loads are connected to any bus signal: Consider 3 types of signals.

- For demuxed signals: Latches used for demuxing, e.g. ‘373, can also provide the buffering for the demuxed lines:
  - 0-level output can sink up to 32 mA (20 x 1.6 mA loads)
  - 1-Level output can source up to 5.2 mA (1 load = 40 mA)

- For non-demuxed unidirectional (always output) address and control signals (e.g. A8-15 on the 8088), buffering is required- often using the **74ALS244** (unidirectional) buffer.

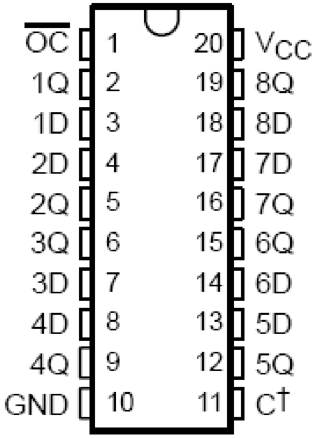
- For non-demuxed bidirectional data signals (pin used for both in and out), buffering is often accomplished with the **74ALS245** bidirectional bus buffer.

- Caution: Buffering introduces a **small delay** in the buffered signals. This is acceptable unless memory or I/O devices operate close to the maximum bus speed

SN54LS373, SN54LS374, SN54S373, SN54S374, SN74LS373, SN74LS374, SN74S373, SN74S374
OCTAL D-TYPE TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS

- Choice of Eight Latches or Eight D-Type Flip-Flops in a Single Package
- 3-State Bus-Driving Outputs
- Full Parallel Access for Loading
- Buffered Control Inputs
- Clock-Enable Input Has Hysteresis to Improve Noise Rejection ('S373 and 'S374)
- P-N-P Inputs Reduce DC Loading on Data Lines ('S373 and 'S374)

SN74S373 . . . SN74LS374



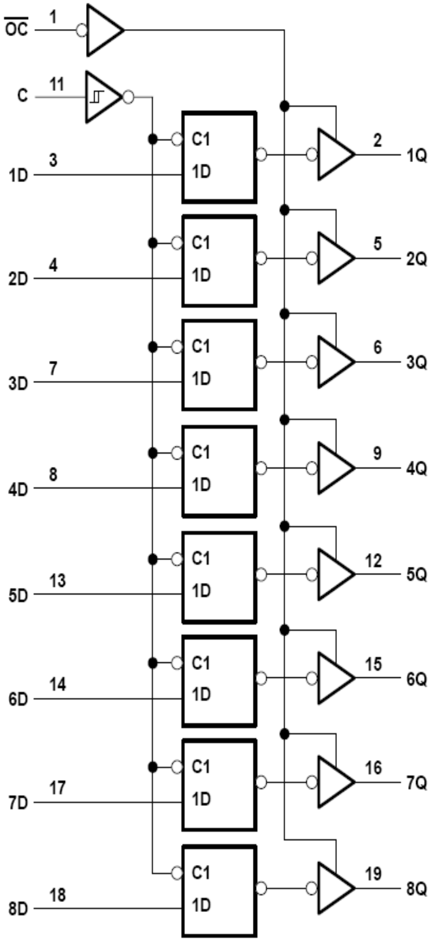
† C for 'LS373 and 'S373; CLK for 'LS374 and 'S374

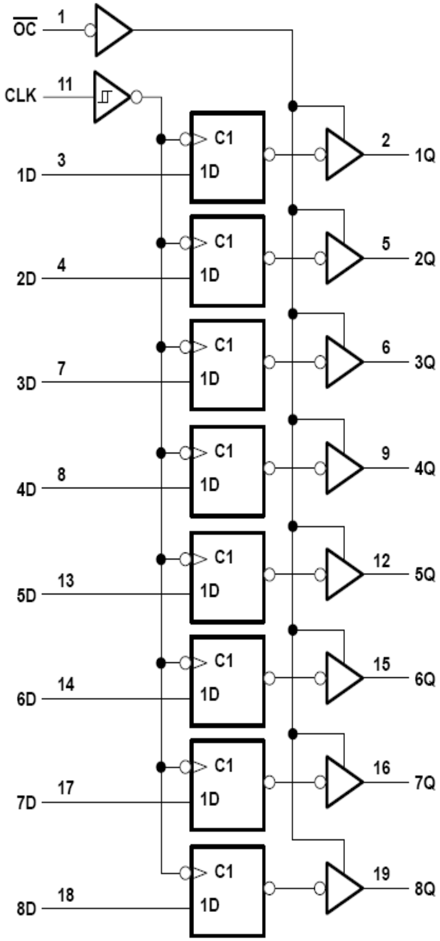
'LS373, 'S373

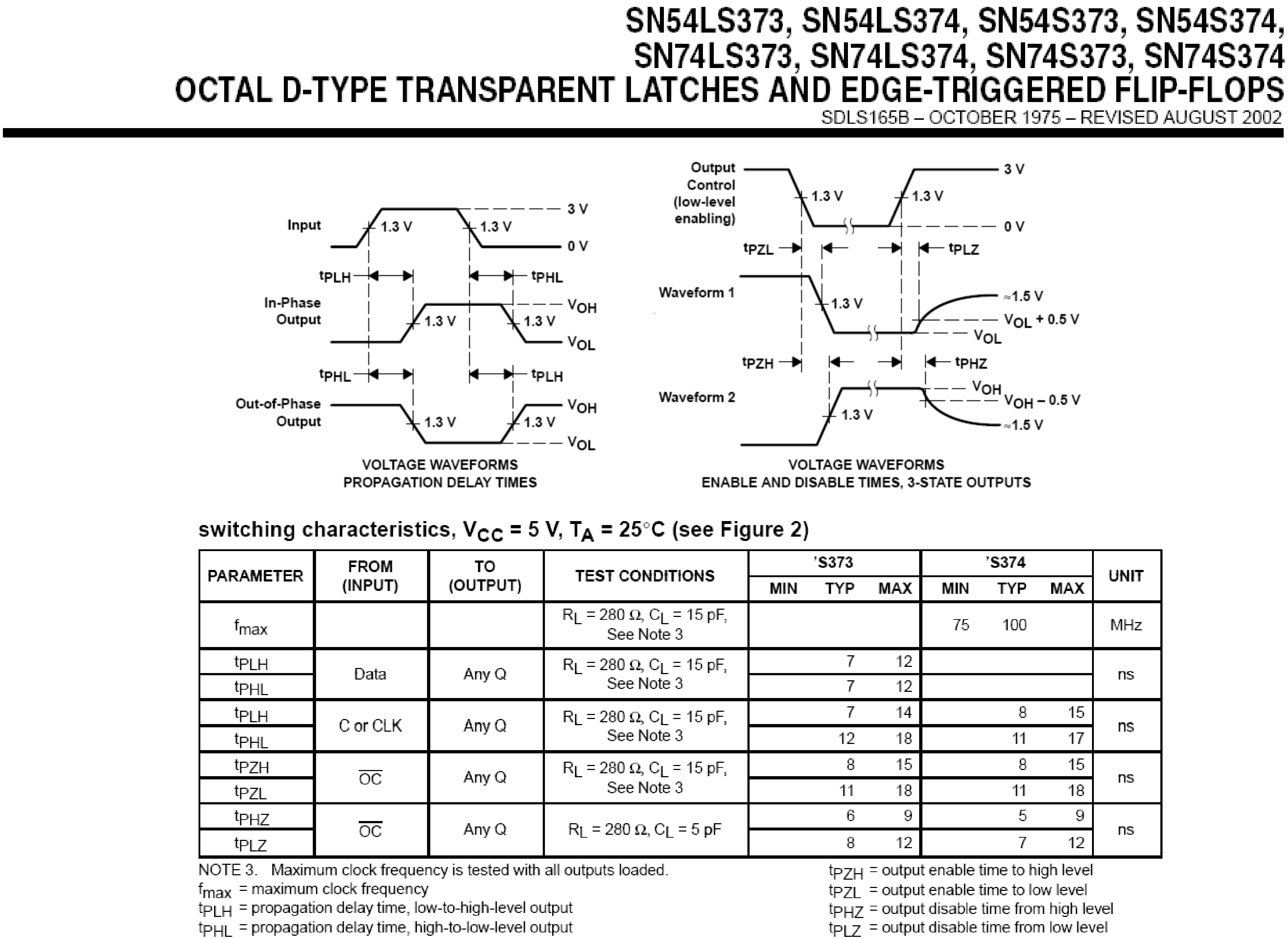
INPUTS			OUTPUT
$\overline{OC}$	C	D	Q
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z

'LS374, 'S374

INPUTS			OUTPUT
$\overline{OC}$	CLK	D	Q
L	↑	H	H
L	↑	L	L
L	L	X	Q_0
H	X	X	Z







SN54LS373, SN54LS374, SN54S373, SN54S374, SN74LS373, SN74LS374, SN74S373, SN74S374 OCTAL D-TYPE TRANSPARENT LATCHES AND EDGE-TRIGGERED FLIP-FLOPS

SDLS165B – OCTOBER 1975 – REVISED AUGUST 2002

Input
1.3 V
1.3 V
3 V
0 V
tPLH
tPHL
In-Phase Output
1.3 V
1.3 V
VOH
VOL
tPHL
tPLH
Out-of-Phase Output
1.3 V
1.3 V
VOH
VOL
Output Control (low-level enabling)
1.3 V
1.3 V
3 V
0 V
tPZL
tPLZ
Waveform 1
1.3 V
≈1.5 V
VOL + 0.5 V
VOL
tPZH
tPHZ
Waveform 2
1.3 V
VOH
VOH – 0.5 V
≈1.5 V

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES, 3-STATE OUTPUTS

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Figure 2)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	'S373			'S374			UNIT
				MIN	TYP	MAX	MIN	TYP	MAX	
f_{max}			R_L = 280 Ω, C_L = 15 pF, See Note 3				75	100		MHz
t_{PLH}	Data	Any Q	R_L = 280 Ω, C_L = 15 pF, See Note 3		7	12				ns
t_{PHL}					7	12				
t_{PLH}	C or CLK	Any Q	R_L = 280 Ω, C_L = 15 pF, See Note 3		7	14		8	15	ns
t_{PHL}					12	18		11	17	
t_{PZH}	$\overline{OC}$	Any Q	R_L = 280 Ω, C_L = 15 pF, See Note 3		8	15		8	15	ns
t_{PZL}					11	18		11	18	
t_{PHZ}	$\overline{OC}$	Any Q	R_L = 280 Ω, C_L = 5 pF		6	9		5	9	ns
t_{PLZ}					8	12		7	12	

NOTE 3. Maximum clock frequency is tested with all outputs loaded.

f_{max} = maximum clock frequency
t_{PLH} = propagation delay time, low-to-high-level output
t_{PHL} = propagation delay time, high-to-low-level output

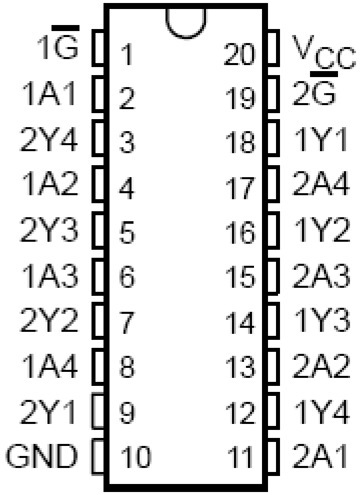
t_{PZH} = output enable time to high level
t_{PZL} = output enable time to low level
t_{PHZ} = output disable time from high level
t_{PLZ} = output disable time from low level

SN54LS240, SN54LS241, SN54LS244, SN54S240, SN54S241, SN54S244 SN74LS240, SN74LS241, SN74LS244, SN74S240, SN74S241, SN74S244 OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

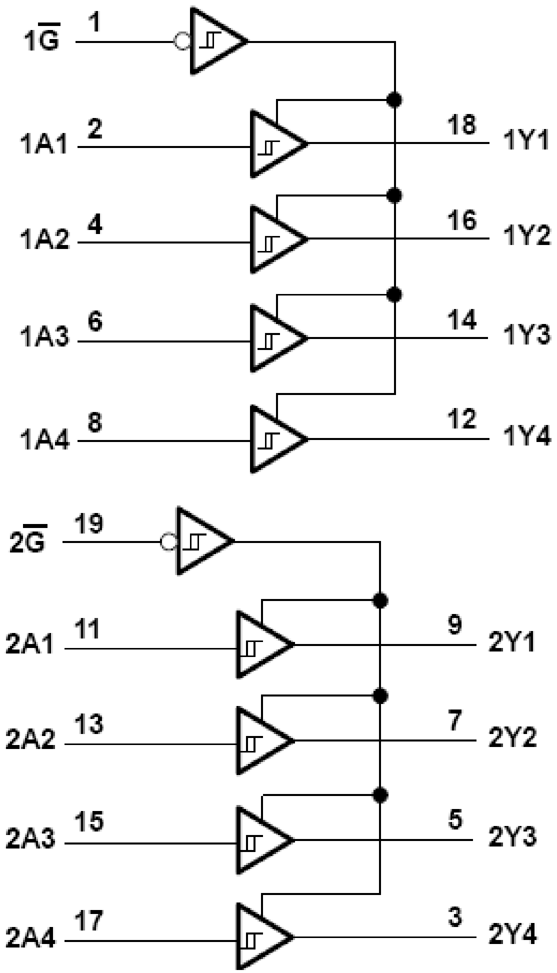
SDLS144B – APRIL 1985 – REVISED FEBRUARY 2002

- 3-State Outputs Drive Bus Lines or Buffer Memory Address Registers
- PNP Inputs Reduce DC Loading
- Hysteresis at Inputs Improves Noise Margins

SN74LS244

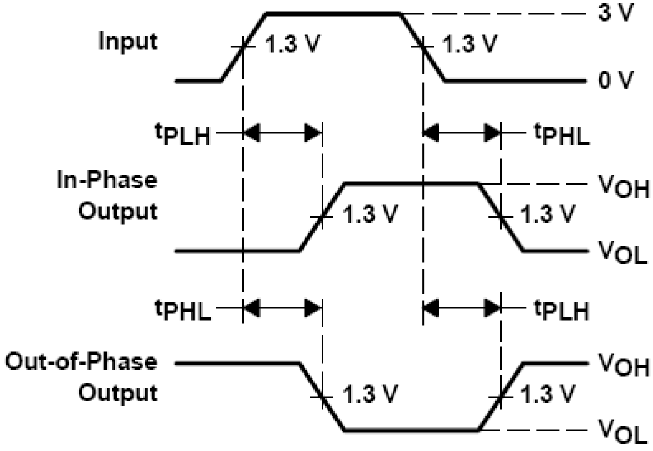


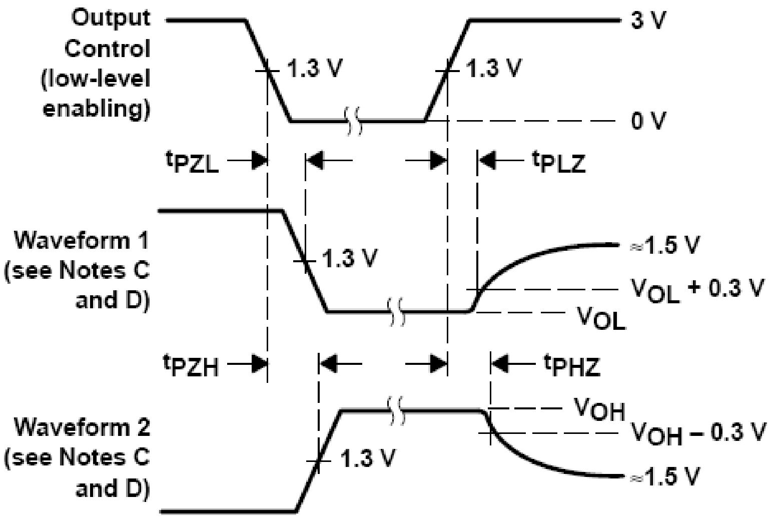
'LS244, 'S244



SN54LS240, SN54LS241, SN54LS244, SN54S240, SN54S241, SN54S244
SN74LS240, SN74LS241, SN74LS244, SN74S240, SN74S241, SN74S244
OCTAL BUFFERS AND LINE DRIVERS WITH 3-STATE OUTPUTS

SDLS144B – APRIL 1985 – REVISED FEBRUARY 2002




Output Control (low-level enabling)
1.3 V
1.3 V
3 V
0 V
tPZL
tPLZ
Waveform 1 (see Notes C and D)
1.3 V
≈1.5 V
VOL + 0.3 V
VOL
tPZH
tPHZ
Waveform 2 (see Notes C and D)
1.3 V
VOH
VOH – 0.3 V
≈1.5 V

switching characteristics, V_{CC} = 5 V, T_A = 25°C

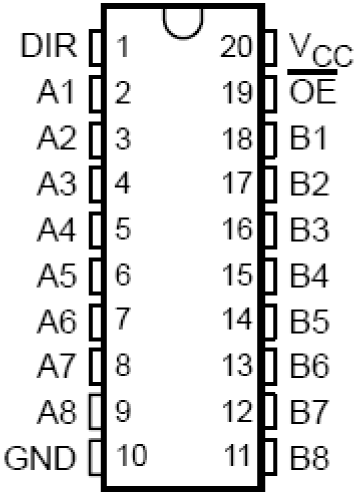
PARAMETER	TEST CONDITIONS		'LS240			'LS241, 'LS244			UNIT
			MIN	TYP	MAX	MIN	TYP	MAX	
t_{PLH}	R_L = 667 Ω,	C_L = 45 pF		9	14		12	18	ns
t_{PHL}				12	18		12	18	
t_{PZL}	R_L = 667 Ω,	C_L = 45 pF		20	30		20	30	ns
t_{PZH}				15	23		15	23	
t_{PLZ}	R_L = 667 Ω,	C_L = 5 pF		10	20		10	20	ns
t_{PHZ}				15	25		15	25	

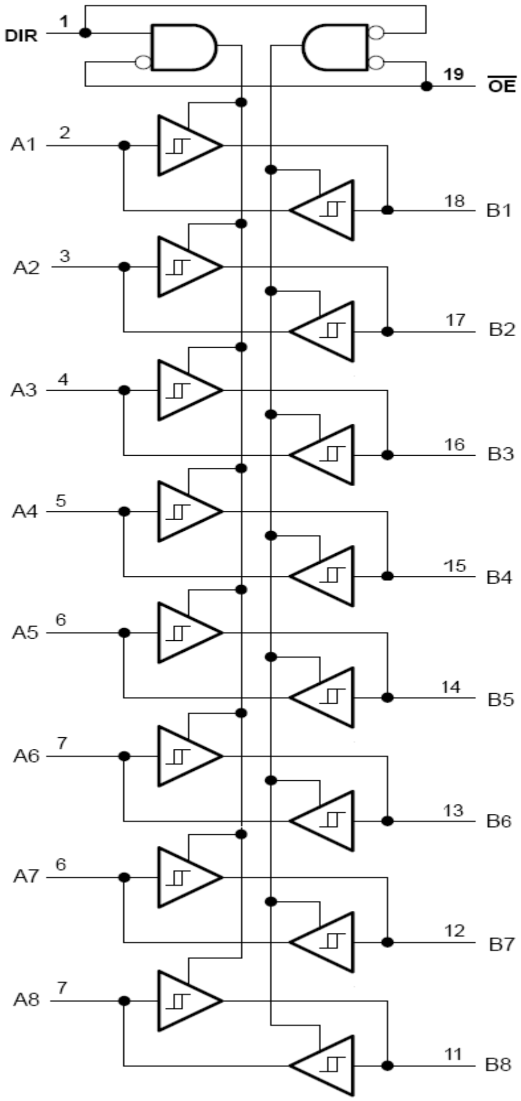
SN54LS245, SN74LS245 OCTAL BUS TRANSCEIVERS WITH 3-STATE OUTPUTS

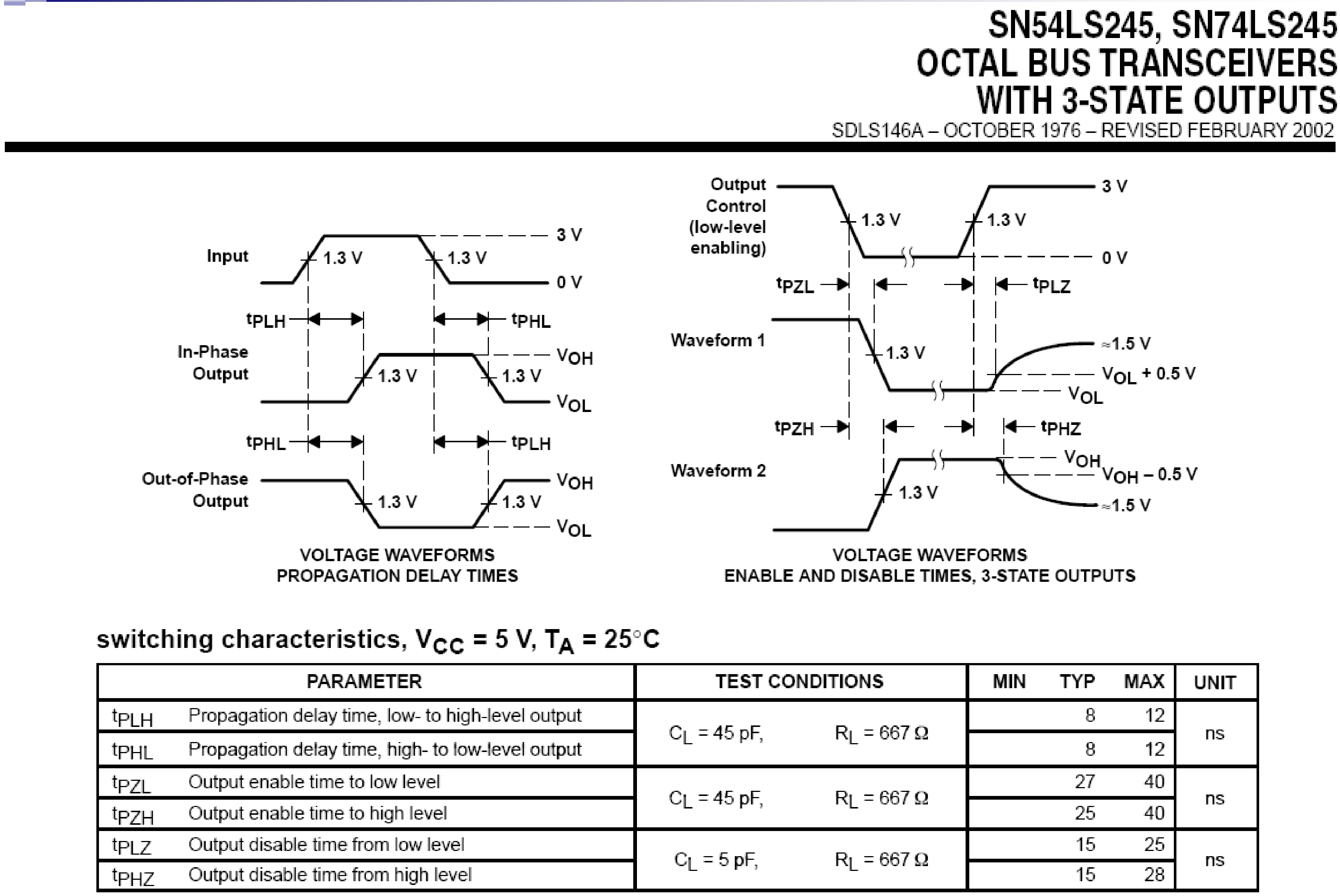
SDLS146A – OCTOBER 1976 – REVISED FEBRUARY 2002

- 3-State Outputs Drive Bus Lines Directly
- PNP Inputs Reduce dc Loading on Bus Lines
- Hysteresis at Bus Inputs Improves Noise Margins
- Typical Propagation Delay Times Port to Port, 8 ns

SN74LS245







SN54LS245, SN74LS245
OCTAL BUS TRANSCEIVERS
WITH 3-STATE OUTPUTS

SDLS146A – OCTOBER 1976 – REVISED FEBRUARY 2002

Input
1.3 V
1.3 V
3 V
0 V
tPLH
tPHL
In-Phase
Output
1.3 V
1.3 V
VOH
VOL
tPHL
tPLH
Out-of-Phase
Output
1.3 V
1.3 V
VOH
VOL

VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES

Output
Control
(low-level
enabling)
1.3 V
1.3 V
3 V
0 V
tPZL
tPLZ
Waveform 1
1.3 V
≈1.5 V
VOL + 0.5 V
VOL
tPZH
tPHZ
Waveform 2
1.3 V
VOH
VOH – 0.5 V
≈1.5 V

VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES, 3-STATE OUTPUTS

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
t_{PLH}	Propagation delay time, low- to high-level output	C_L = 45 pF,	R_L = 667 Ω		8	12	ns
t_{PHL}	Propagation delay time, high- to low-level output				8	12	
t_{PZL}	Output enable time to low level	C_L = 45 pF,	R_L = 667 Ω		27	40	ns
t_{PZH}	Output enable time to high level				25	40	
t_{PLZ}	Output disable time from low level	C_L = 5 pF,	R_L = 667 Ω		15	25	ns
t_{PHZ}	Output disable time from high level				15	28	