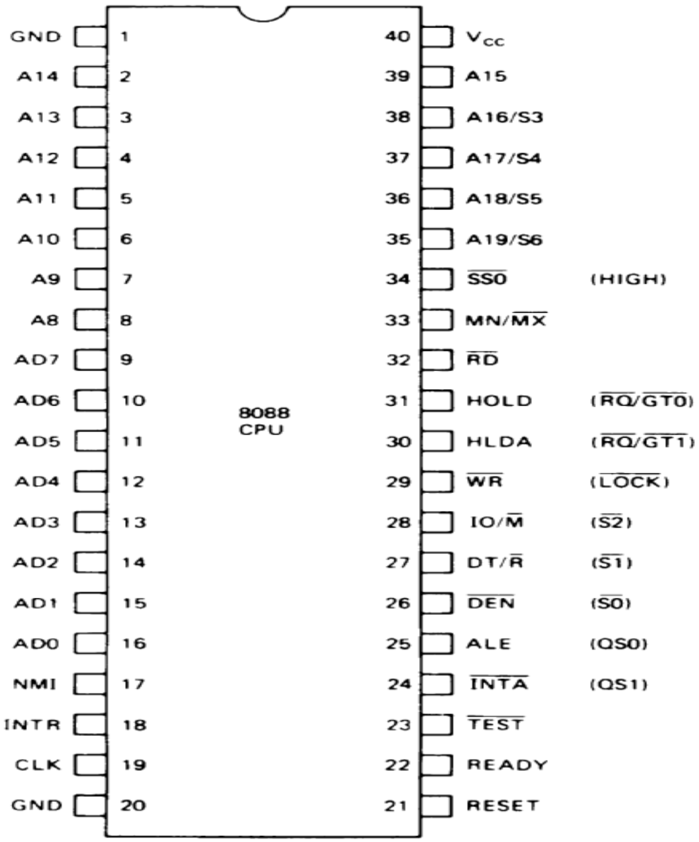
Week 6

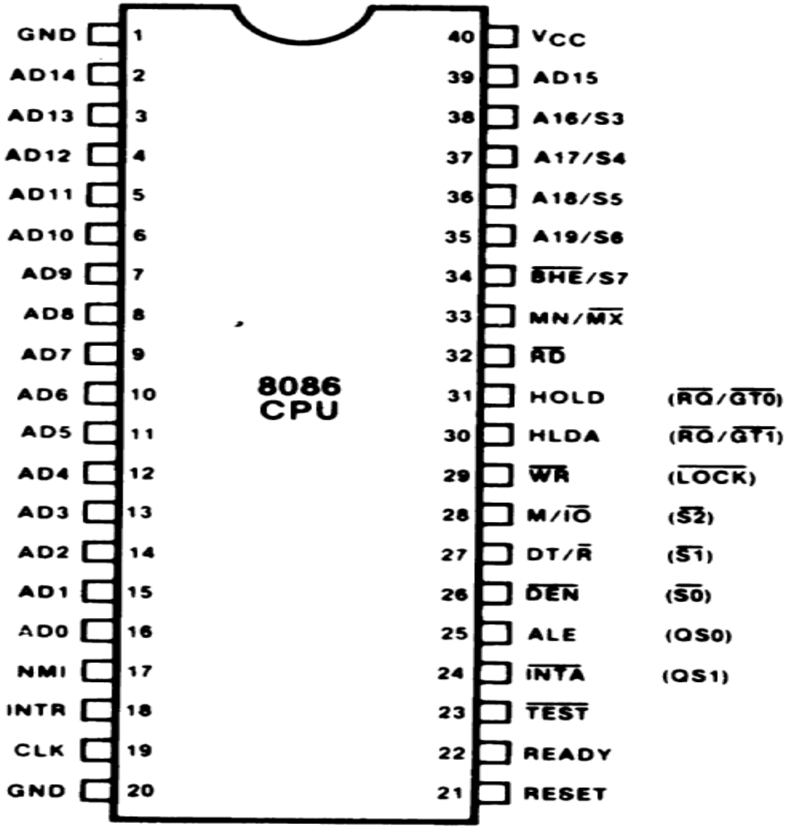
The 8088 and 8086 Microprocessors and Their Memory and Input/Output Interfaces

8086 and 8088 Microprocessors

- 8086 announced in 1978; 8086 is a 16 bit microprocessor with a 16 bit data bus
- 8088 announced in 1979; 8088 is a 16 bit microprocessor with an 8 bit data bus
- Both manufactured using High-performance Metal Oxide Semiconductor (HMOS) technology
- Both contain about 29000 transistors
- Both are packaged in 40 pin dual-in-line package (DIP)
- Address lines A0-A7 and Data lines D0-D7 are multiplexed in 8088
  - By multiplexed we mean that the same pysical pin carries an address bit at one time and the data bit another time
- Address lines A0-A15 and Data lines D0-D15 are multiplexed in 8086

8088 and 8086 Microprocessors



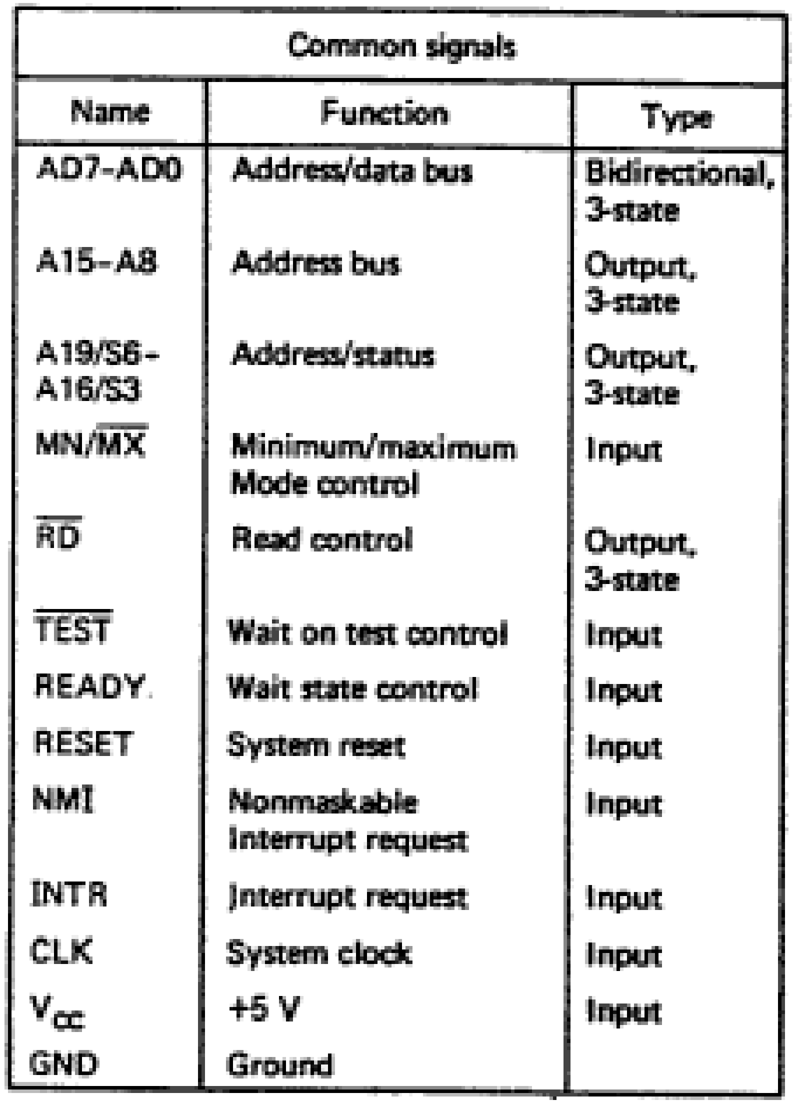


Minimum-mode and Maximum-mode Systems

- 8088 and 8086 microprocessors can be configured to work in either of the two modes: the minimum mode and the maximum mode
- Minimum mode:
  - Pull MN/$\overline{\mathbf{MX}}$ to logic 1
  - Typically smaller systems and contains a single microprocessor
- Maximum mode
  - Pull MN/$\overline{\mathbf{MX}}$ logic 0
  - Larger systems with more than one processor

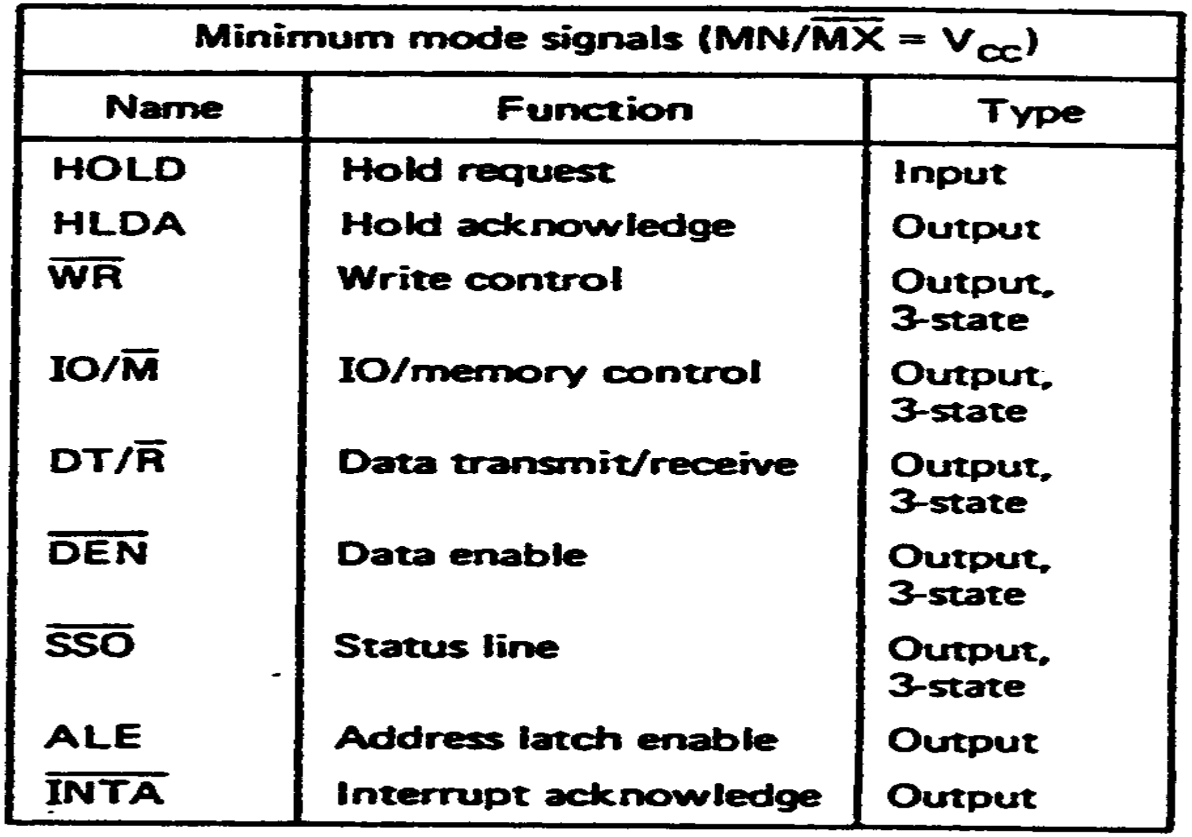
Minimum-mode and Maximum-mode Systems

Signals common to both minimum and maximum systems



Common signals		
Name	Function	Type
AD7–AD0	Address/data bus	Bidirectional, 3-state
A15–A8	Address bus	Output, 3-state
A19/S6–A16/S3	Address/status	Output, 3-state
MN/$\overline{\text{MX}}$	Minimum/maximum Mode control	Input
$\overline{\text{RD}}$	Read control	Output, 3-state
$\overline{\text{TEST}}$	Wait on test control	Input
READY	Wait state control	Input
RESET	System reset	Input
NMI	Nonmaskable Interrupt request	Input
INTR	Interrupt request	Input
CLK	System clock	Input
V_{CC}	+5 V	Input
GND	Ground	

Minimum-mode and Maximum-mode Systems

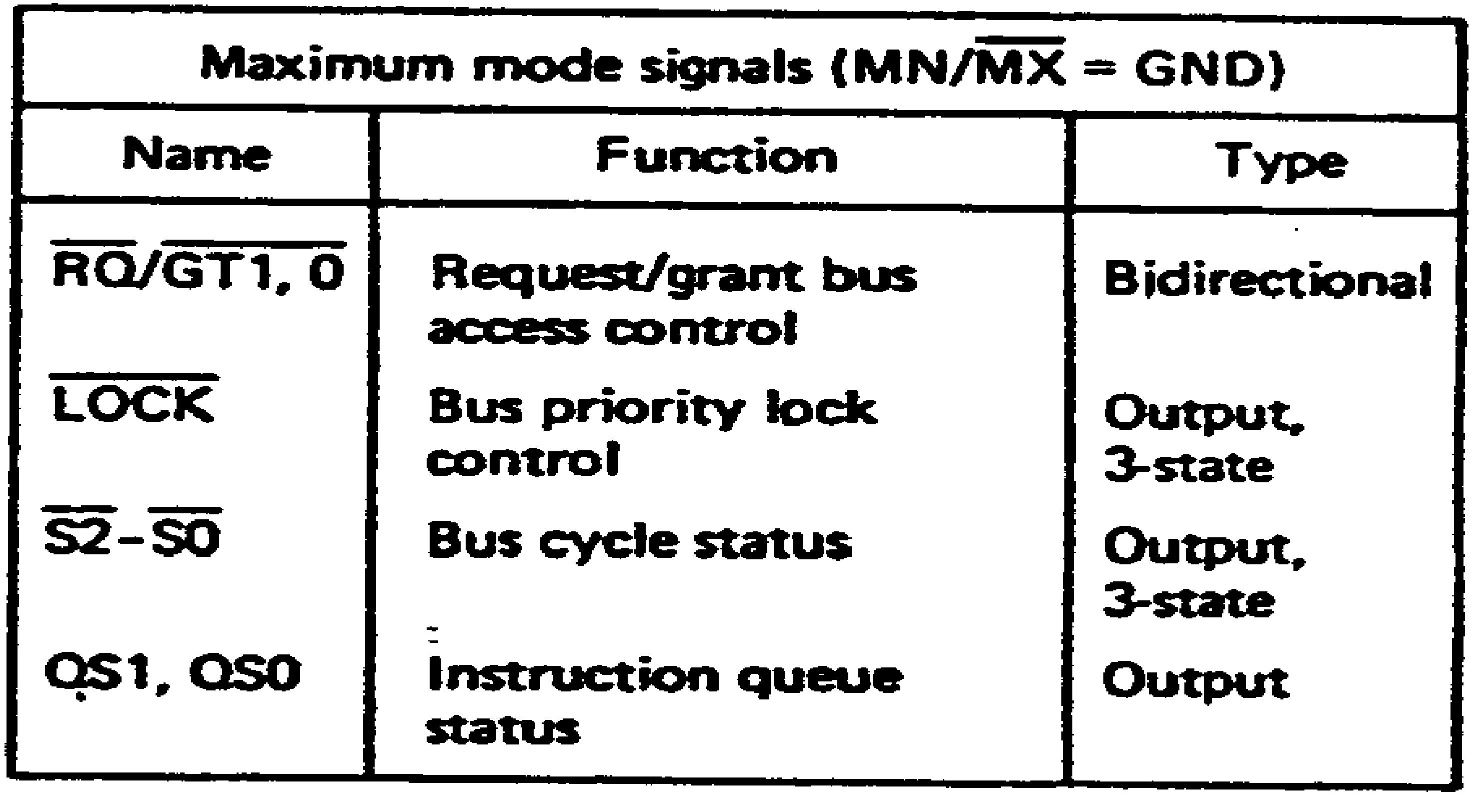


Minimum mode signals (MN/$\overline{MX}$ = V_{CC})

Name	Function	Type
HOLD	Hold request	Input
HLDA	Hold acknowledge	Output
$\overline{WR}$	Write control	Output, 3-state
IO/$\overline{M}$	IO/memory control	Output, 3-state
DT/$\overline{R}$	Data transmit/receive	Output, 3-state
$\overline{DEN}$	Data enable	Output, 3-state
$\overline{SSO}$	Status line	Output, 3-state
ALE	Address latch enable	Output
$\overline{INTA}$	Interrupt acknowledge	Output

Minimum mode unique signals

Minimum-mode and Maximum-mode Systems

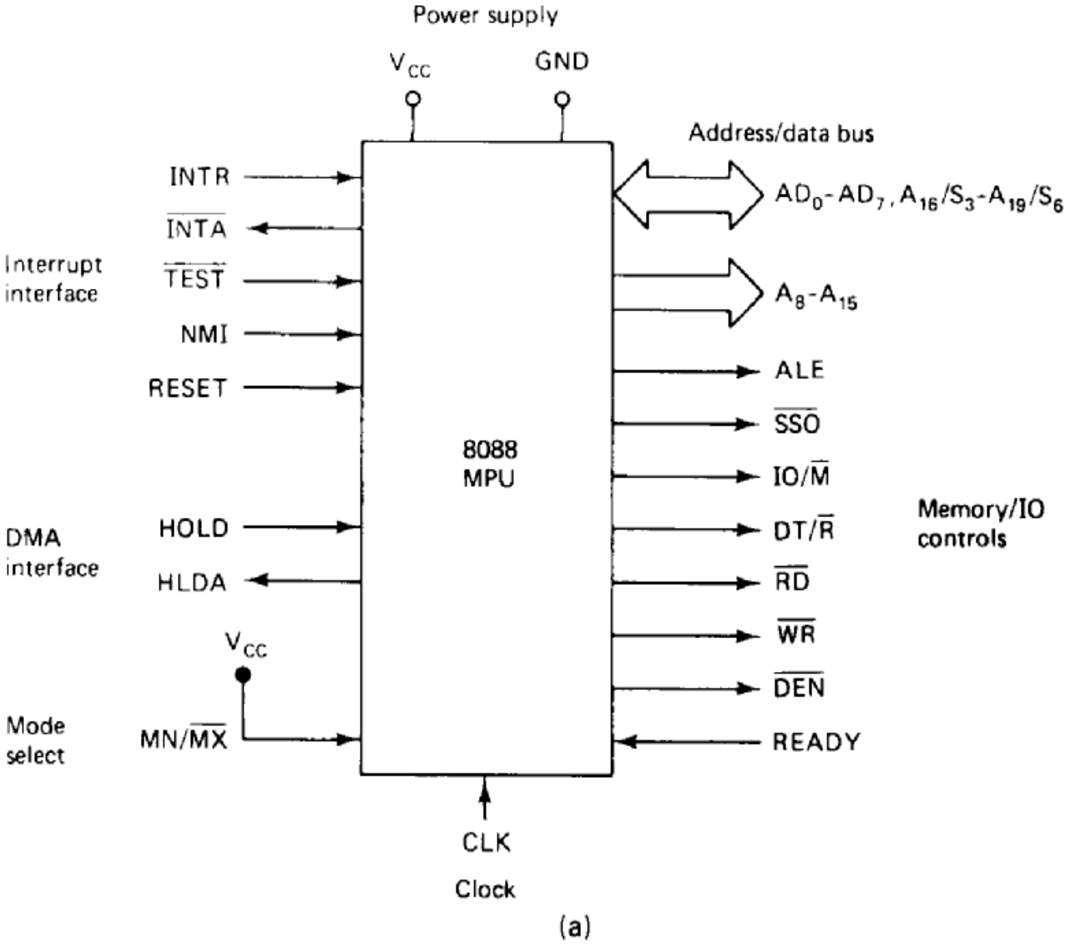


Maximum mode signals (MN/$\overline{\text{MX}}$ = GND)

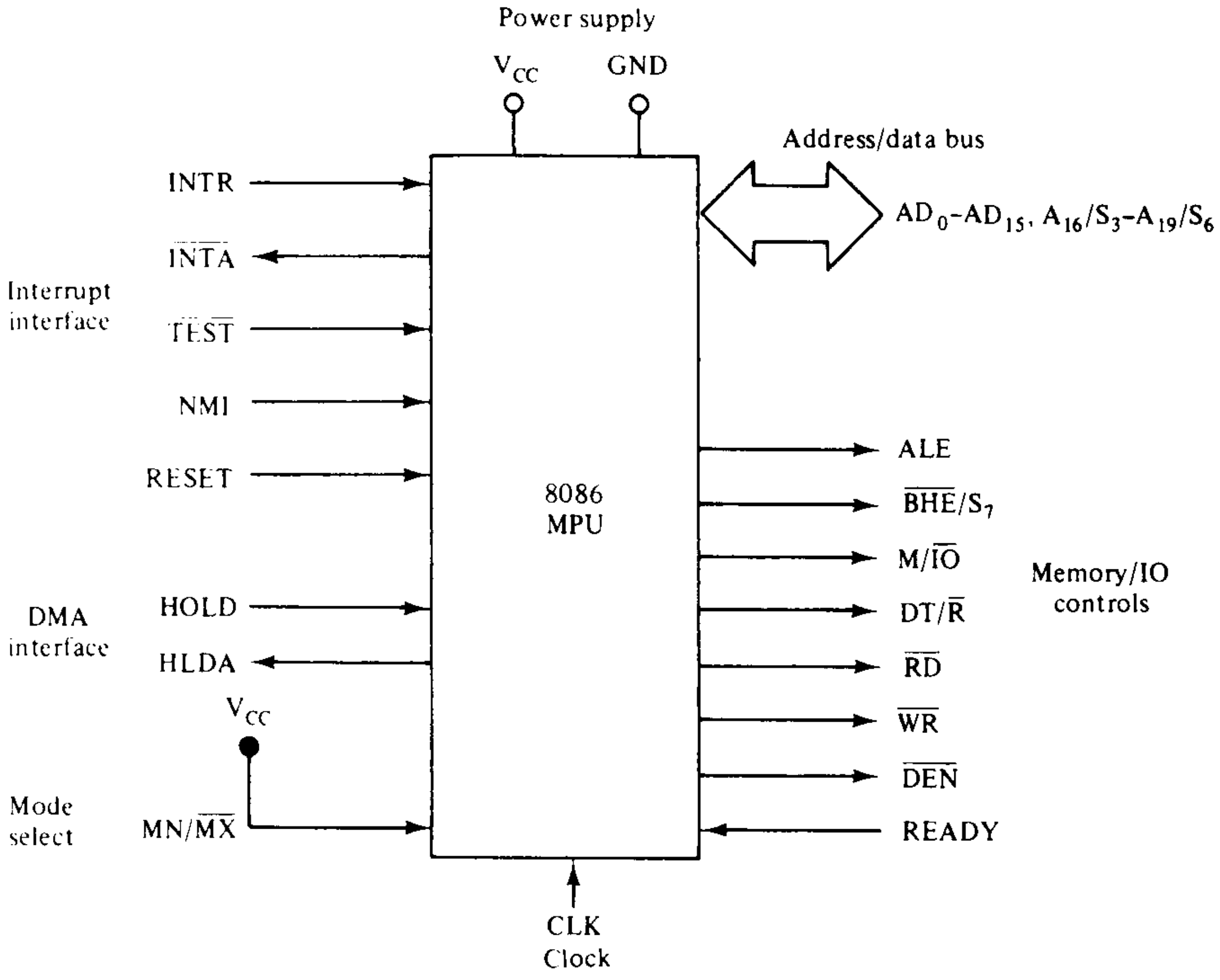
Name	Function	Type
$\overline{\text{RQ}}/\overline{\text{GT1, 0}}$	Request/grant bus access control	Bidirectional
$\overline{\text{LOCK}}$	Bus priority lock control	Output, 3-state
$\overline{\text{S2}}-\overline{\text{S0}}$	Bus cycle status	Output, 3-state
QS1, QS0	Instruction queue status	Output

Maximum mode unique signals

8088 Minimum-mode block diagram



8086 Minimum-mode block diagram



Minimum Mode Interface

- Address/Data bus: 20 bits vs 8 bits multiplexed
- Status signals: A_{16}-A_{19} multiplexed with status signals S_3-S_6 respectively
  - S3 and S4 together form a 2 bit binary code that identifies which of the internal segment registers was used to generate the physical address that was output on the address bus during the current bus cycle.
  - S5 is the logic level of the internal interrupt enable flag, s6 is always logic 0.

S4	S3	Addres status
0	0	Alternate(relative to ES segment)
0	1	Stack (relative to SS Segment)
1	0	Code/None (relative to CS segment or a default zero)
1	1	Data (relative to DS segment)

Minimum Mode Interface

- Control Signals: (8088)
  - **Address Latch Enable (ALE)** is a pulse to logic 1 that signals external circuitry when a valid address is on the bus. This address can be latched in external circuitry on the 1-to-0 edge of the pulse at ALE.
  - **IO/$\overline{\text{M}}$ line:** memory or I/O transfer is selected (complement for 8086)
  - **DT/$\overline{\text{R}}$ line**: direction of data is selected
  - **SSO (System Status Output) line**: =1 when data is read from memory and =0 when code is read from memory (only for 8088)
  - **$\overline{\text{BHE}}$ (Bank High Enable) line** : =0 for most significant byte of data for 8086 and also carries S_7
  - **$\overline{\text{RD}}$ line: =**0 when a read cycle is in progress
  - **$\overline{\text{WR}}$ line:** =0 when a write cyle is in progress
  - **$\overline{\text{DEN}}$ line: (Data enable)** Enables the external devices to supply data to the processor.
  - **Ready line:** can be used to insert wait states into the bus cycle so that it is extended by a number of clock periods.

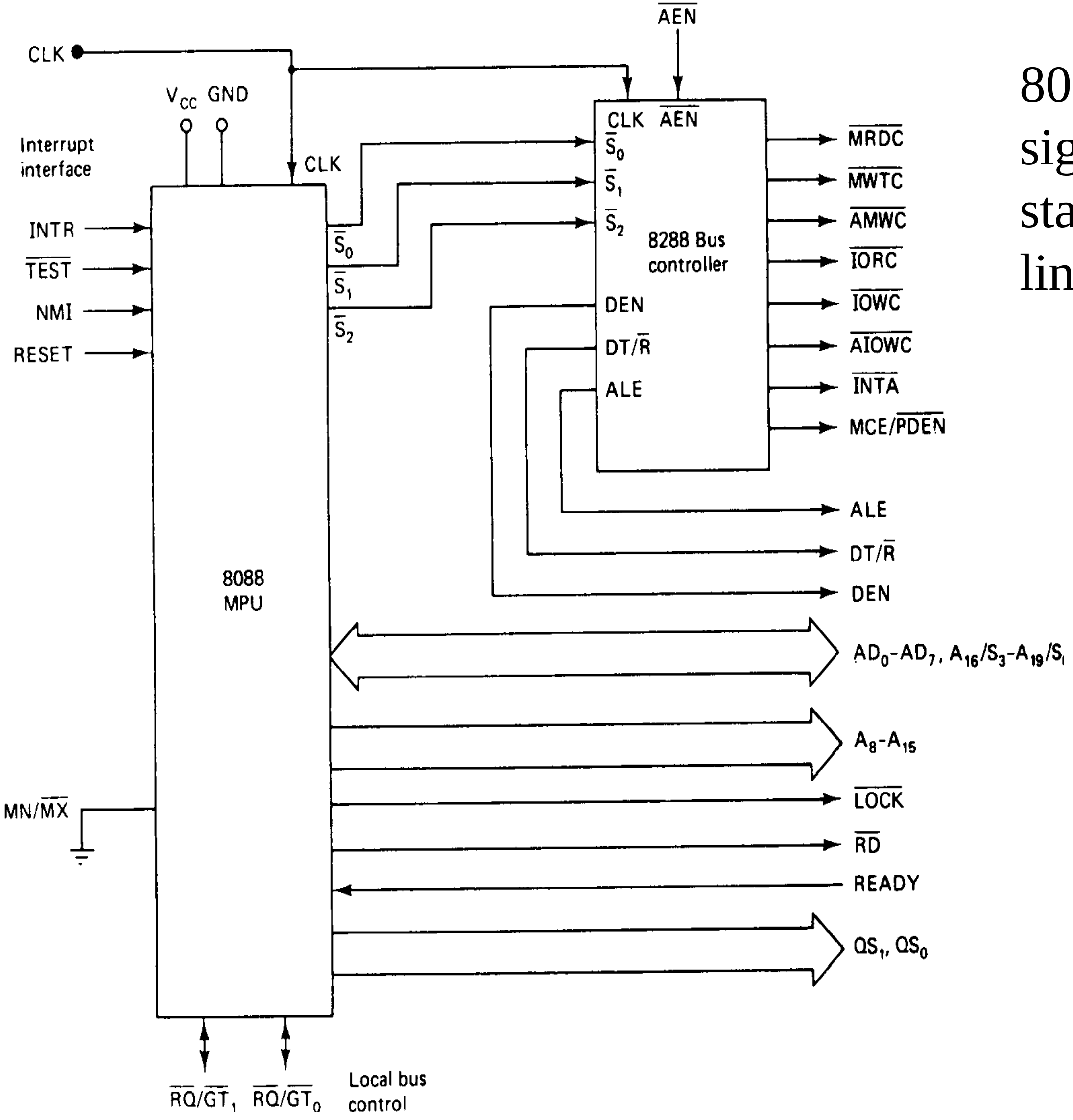
Minimum Mode Interface

- Interrupt signals:
  - INTR (Interrupt request) :=1 shows there is a service request, sampled at the final clock cycle of each instruction acquisition cycle.
  - $\overline{\textbf{INTA}}$: Processor responds with two pulses going to 0 when it services the interrupt and waits for the interrupt service number after the second pulse.
  - $\overline{\textbf{TEST}}$: Processor suspends operation when =1. Resumes operation when=0. Used to syncronize the processor to external events.
  - NMI (Nonmaskable interrupt) : A leading edge transition causes the processor go to the interrupt routine after the current instruction is executed.
  - RESET : =0 Starts the reset sequence.

Minimum Mode Interface

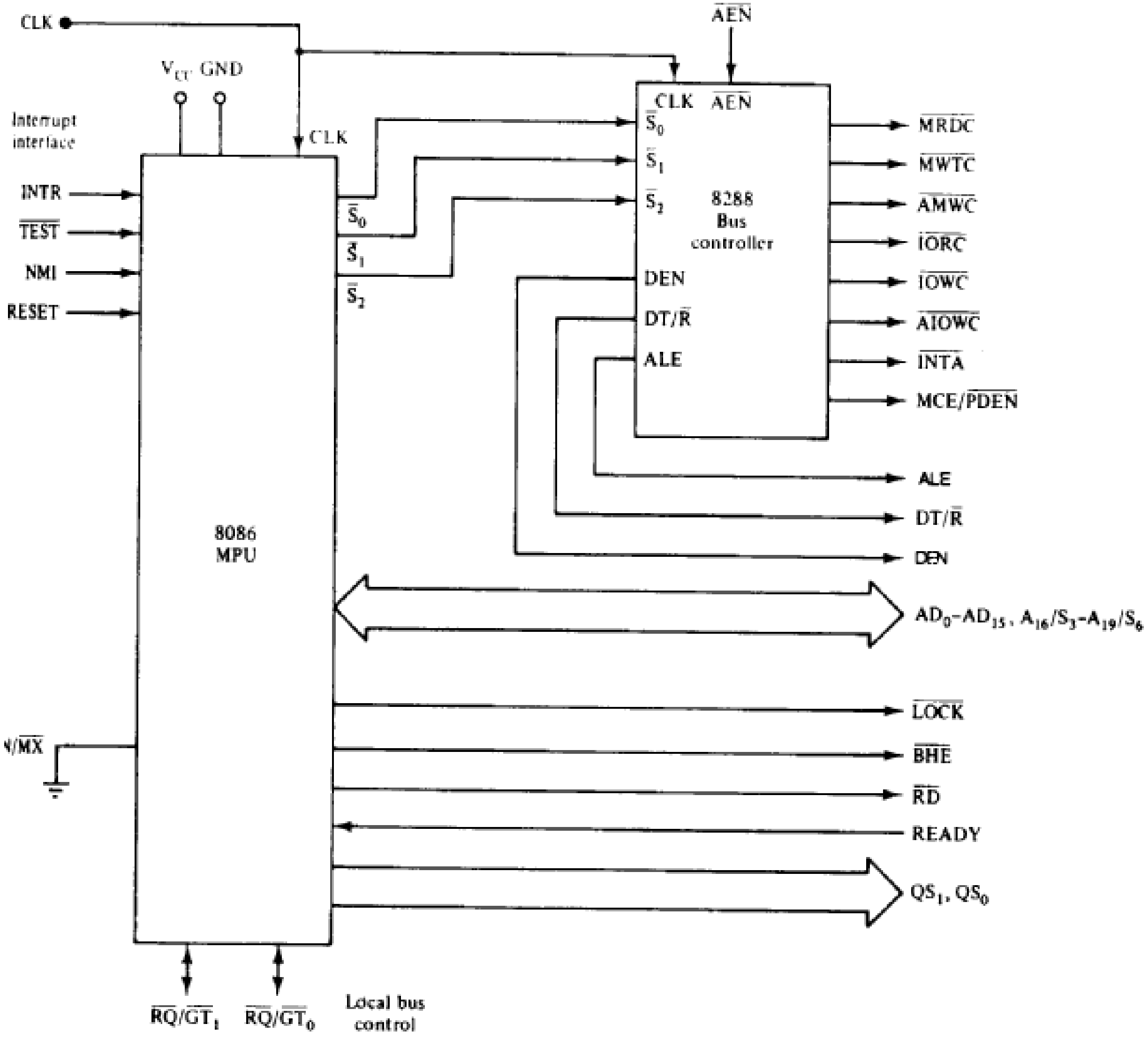
- DMA (Direct Memory Access) Interface Signals:
  - HOLD : External device puts logic level 1 to HOLD input to take control of the bus for DMA request.
  - HLDA (Hold acknowledge) : Processor responds by putting logic level 1 to HDLA.
  - In this state; Address and Data lines, $\overline{\text{SSO}}$, IO/$\overline{\text{M}}$, DT/$\overline{\text{R}}$, $\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{DEN}}$ signals are all put to high-Z state.

Maximum-mode interface circuit diagram (8088)



8088 does not provide all the signals, instead it outputs a status code on three signal lines S0, S1, and S2

Maximum-mode interface circuit diagram (8086)

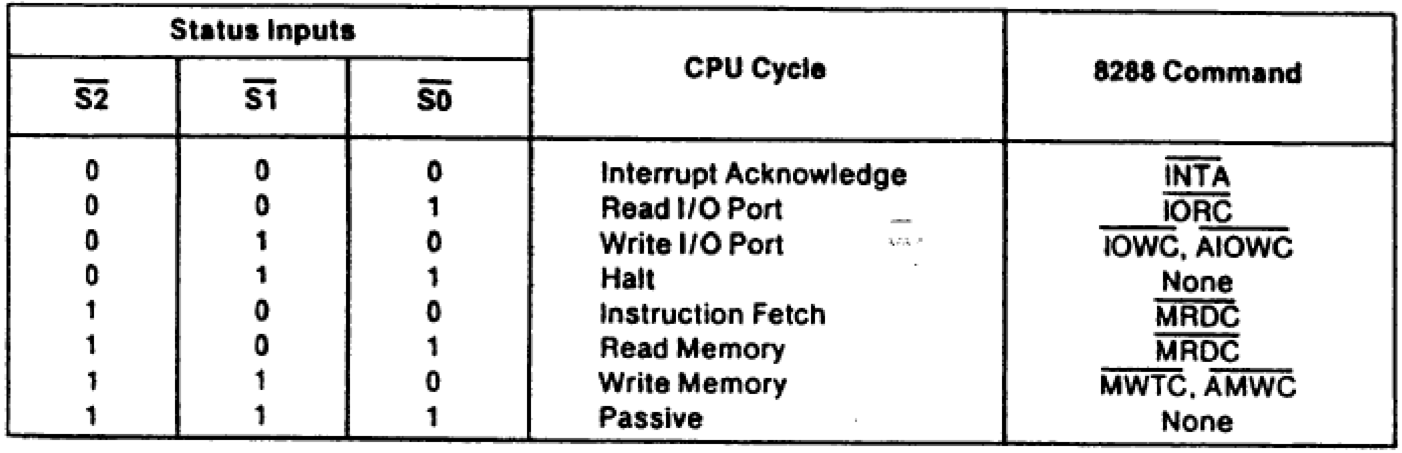


Maximum Mode Interface

- For multiprocessor environment
- 8288 Bus Controller is used for bus control
- WR¯,IO/M¯,DT/R¯,DEN¯,ALE, INTA¯ signals are not available
- Instead:
  - **MRDC¯** (memory read command)
  - **MWRT¯** (memory write command)
  - **AMWC¯** (advanced memory write command)
  - **IORC¯** (I/O read command)
  - **IOWC¯** (I/O write command)
  - **AIOWC¯** (Advanced I/O write command)
  - **INTA¯** (interrupt acknowledge)

Status Bits

They indicate the function of the current bus cycle. They are normally decoded by the 8288 bus controller



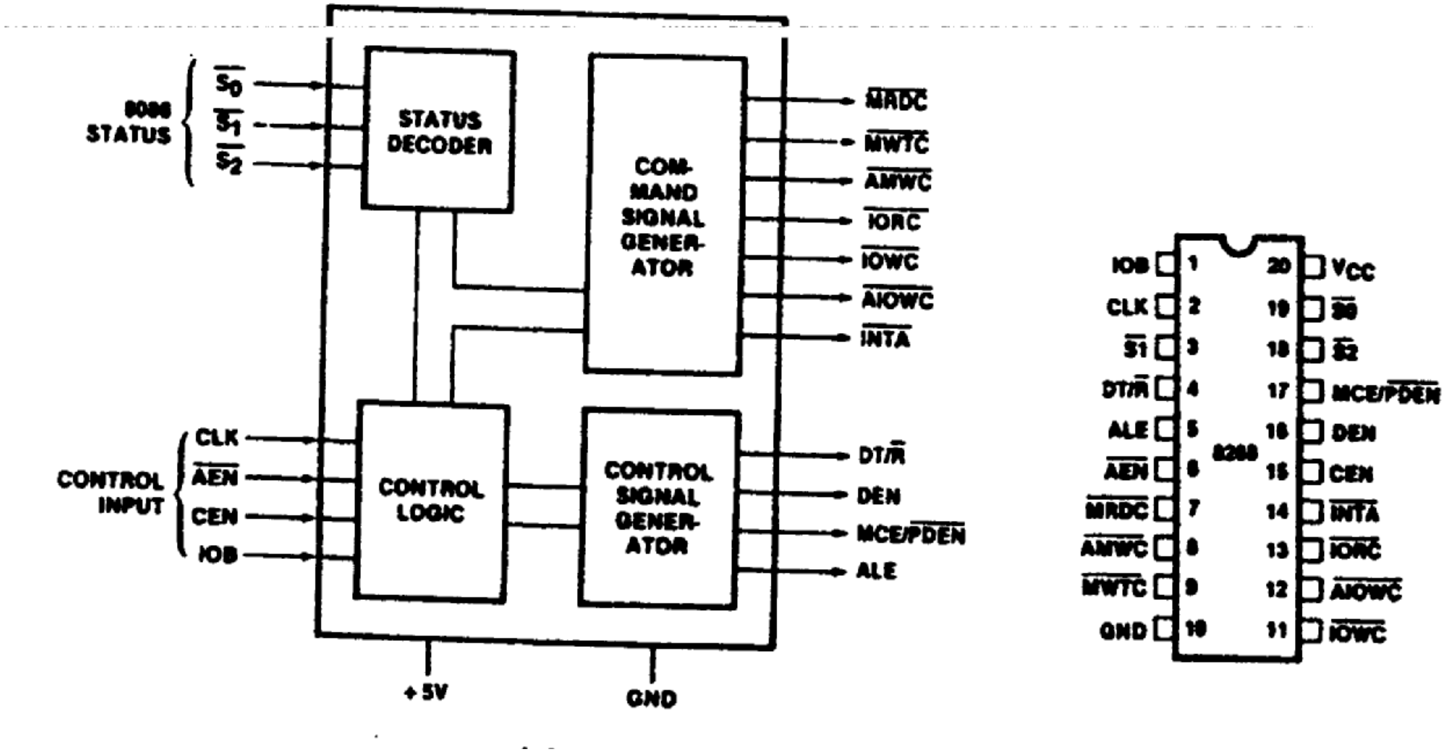
Status Inputs			CPU Cycle	8288 Command
$\overline{S2}$	$\overline{S1}$	$\overline{S0}$		
0	0	0	Interrupt Acknowledge	$\overline{INTA}$
0	0	1	Read I/O Port	$\overline{IORC}$
0	1	0	Write I/O Port	$\overline{IOWC}$, $\overline{AIOWC}$
0	1	1	Halt	None
1	0	0	Instruction Fetch	$\overline{MRDC}$
1	0	1	Read Memory	$\overline{MRDC}$
1	1	0	Write Memory	$\overline{MWTC}$, $\overline{AMWC}$
1	1	1	Passive	None

Maximum Mode Interface

  - The signals shown above are produced by 8288 depending on the state of S_0, S_1 and S_2.

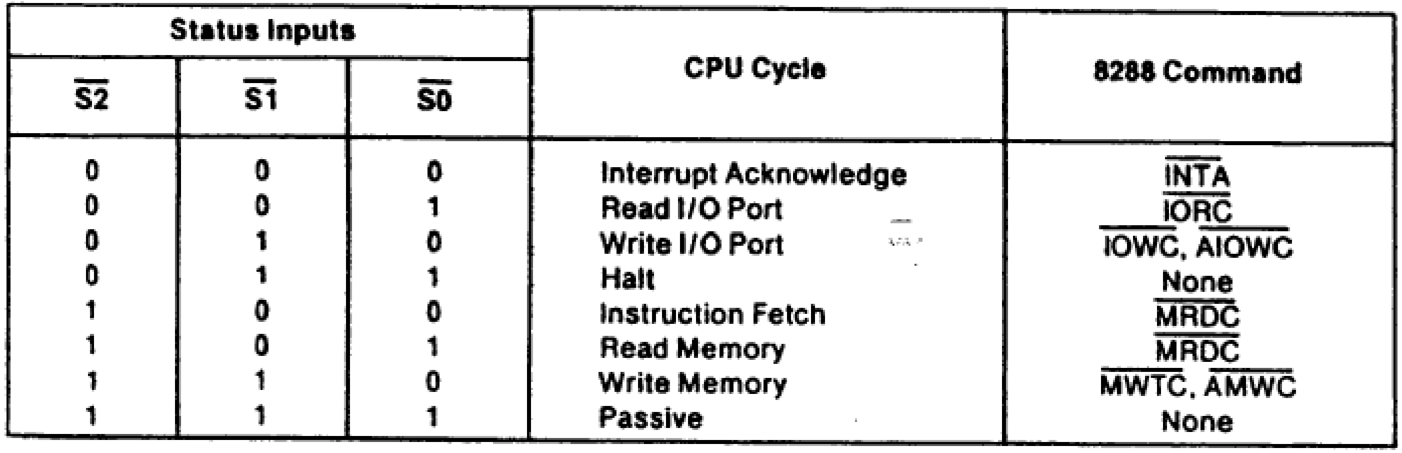
- **DEN, DT/R¯** and ALE signals are the same as minimum-mode systems
- **LOCK¯:** when =0, prevents other processors from using the bus
- QS_0 and QS_1 (queue status signals) : informs about the status of the queue
- **RQ¯/GT ¯$_{\mathbf{0}}$** and **RQ¯/GT ¯$_{\mathbf{1}}$** are used instead of HOLD and HLDA lines in a multiprocessor environment as request/grant lines.

Maximum Mode Interface



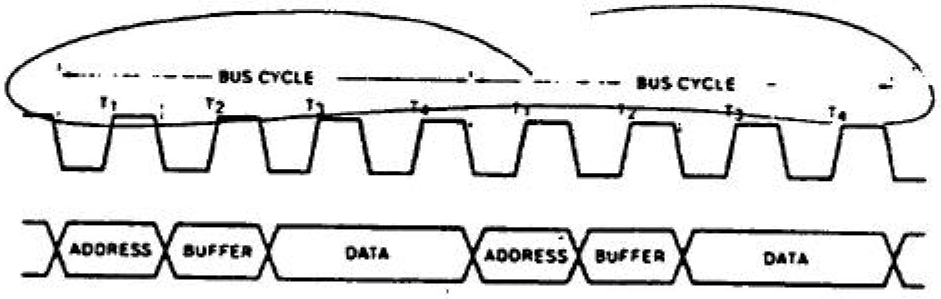
Block Diagram of the 8288

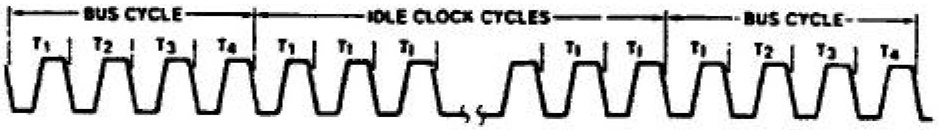
Maximum Mode Interface

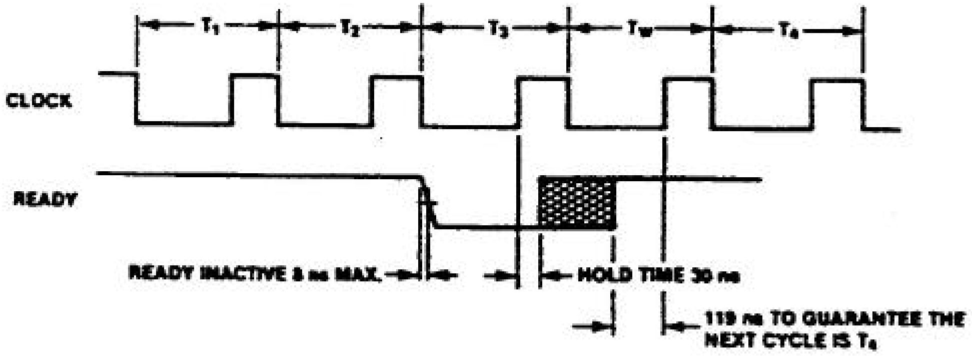


Status Inputs			CPU Cycle	8288 Command
$\overline{S2}$	$\overline{S1}$	$\overline{S0}$		
0	0	0	Interrupt Acknowledge	$\overline{\text{INTA}}$
0	0	1	Read I/O Port	$\overline{\text{IORC}}$
0	1	0	Write I/O Port	$\overline{\text{IOWC}}$, $\overline{\text{AIOWC}}$
0	1	1	Halt	None
1	0	0	Instruction Fetch	$\overline{\text{MRDC}}$
1	0	1	Read Memory	$\overline{\text{MRDC}}$
1	1	0	Write Memory	$\overline{\text{MWTC}}$, $\overline{\text{AMWC}}$
1	1	1	Passive	None

Bus Cycle and Time States







Bus Cycle and Time States

- A bus cycle defines the basic operation that a microprocessor performs to communicate with external devices
- Examples of bus cycles are memory read, memory write, input/output read and input/output write.
- A bus cycle corresponds to a sequence of events that starts with an address being output on the system bus followed by a read or write data transfer
- During these operations, a series of control signals are also produced by the MPU to control the direction and timing of the bus.
- Bus cycle consists of at least four clock periods, T1, T2, T3, and T4.
  - During T1 the MPU puts the address on the address bus
  - For a write memory cycle, data are put on the bus during state T2 and maintained thru T3 and T4.
  - When a read cycle is performed, the bus is first put in the high-Z state during T2 and data to be read must be available on the bus during T3 and T4.
  - Bus cycle duration of 125 ns x 4 = 500 ns in an 8 mhz 8088 system

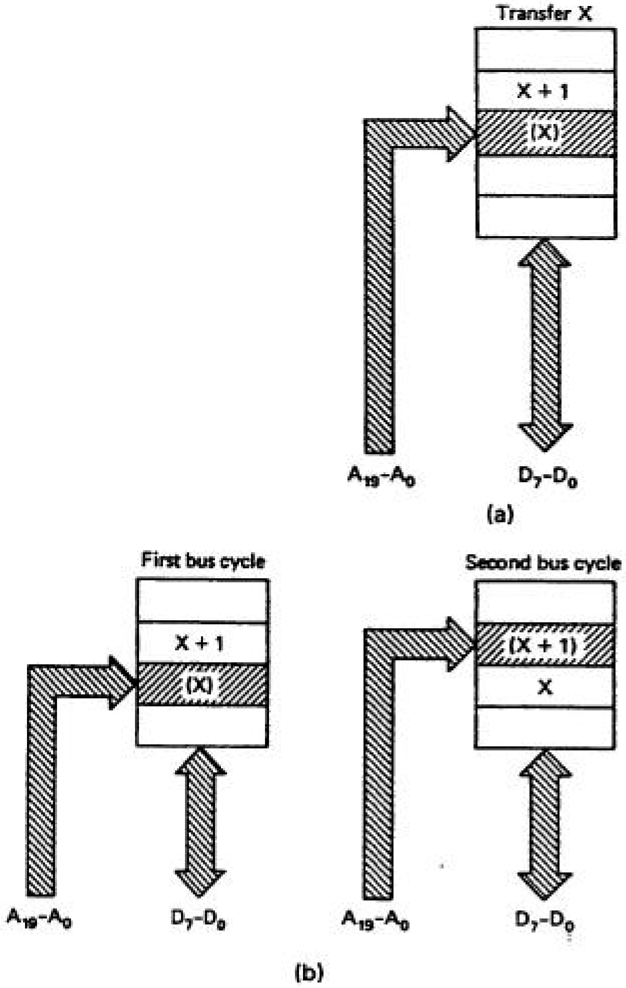
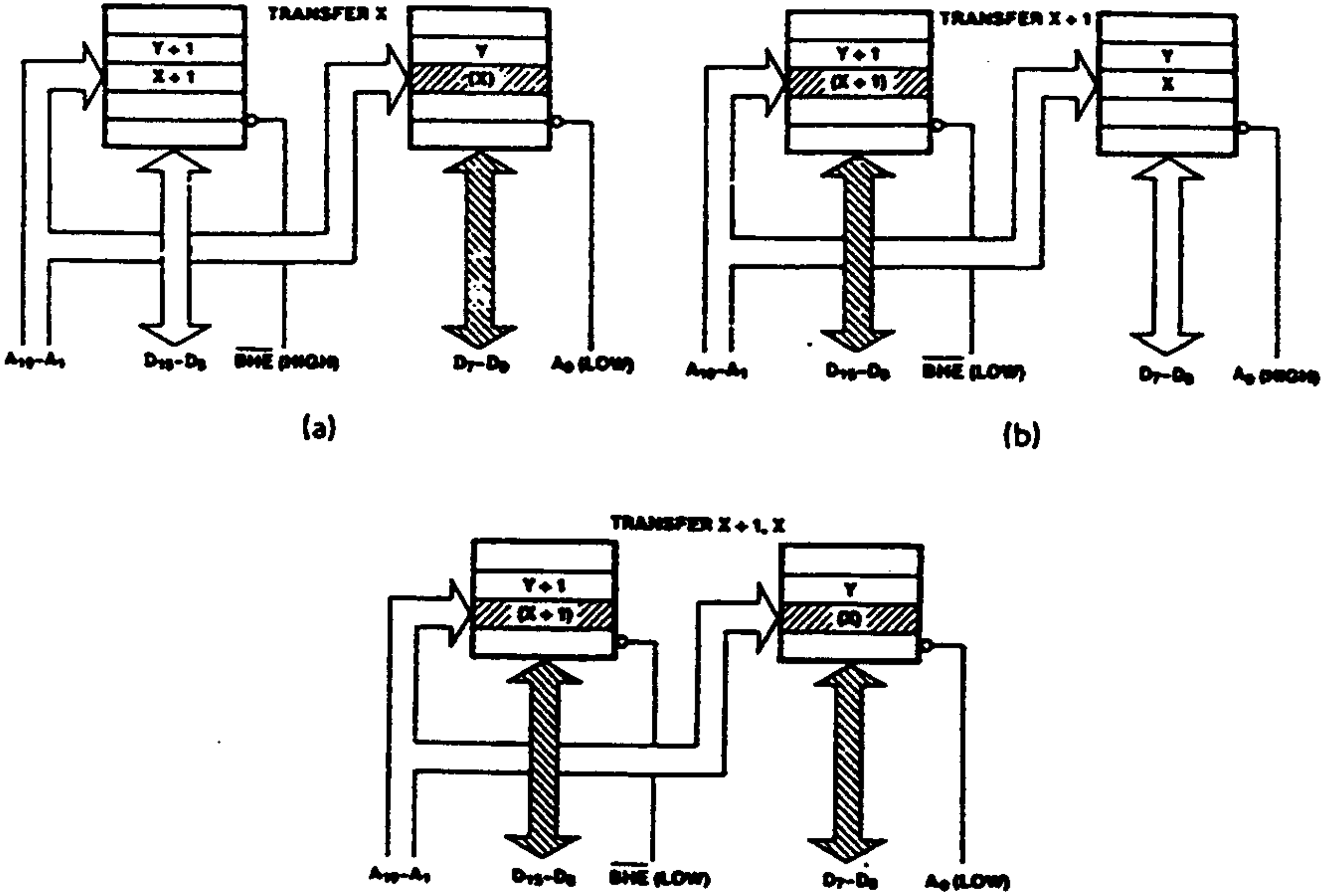


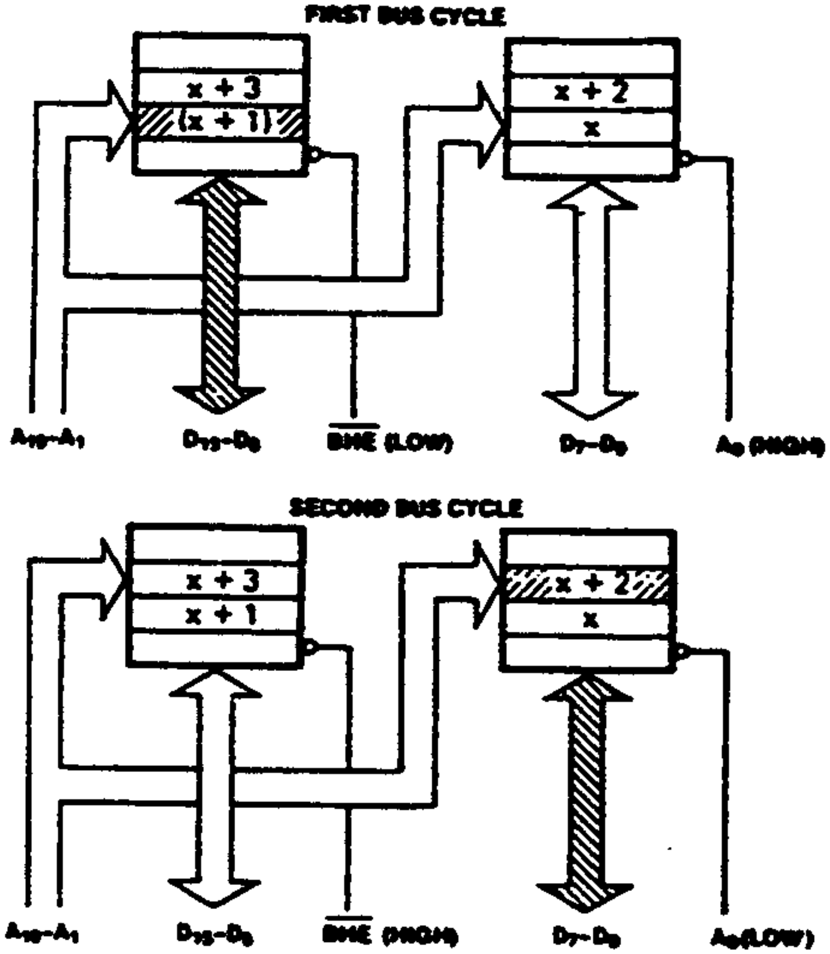
Figure 8–16 (a) Byte trnasfer by the 8088. (b) Word transfer by the 8088.


TRANSFER X
Y + 1
X + 1
Y
(X)
A19–A1
D15–D8
BHE (HIGH)
D7–D0
A0 (LOW)
TRANSFER X + 1
Y + 1
(X + 1)
Y
X
A19–A1
D15–D8
BHE (LOW)
D7–D0
A0 (HIGH)
TRANSFER X + 1, X
Y + 1
(X + 1)
Y
(X)
A19–A1
D15–D8
BHE (LOW)
D7–D0
A0 (LOW)

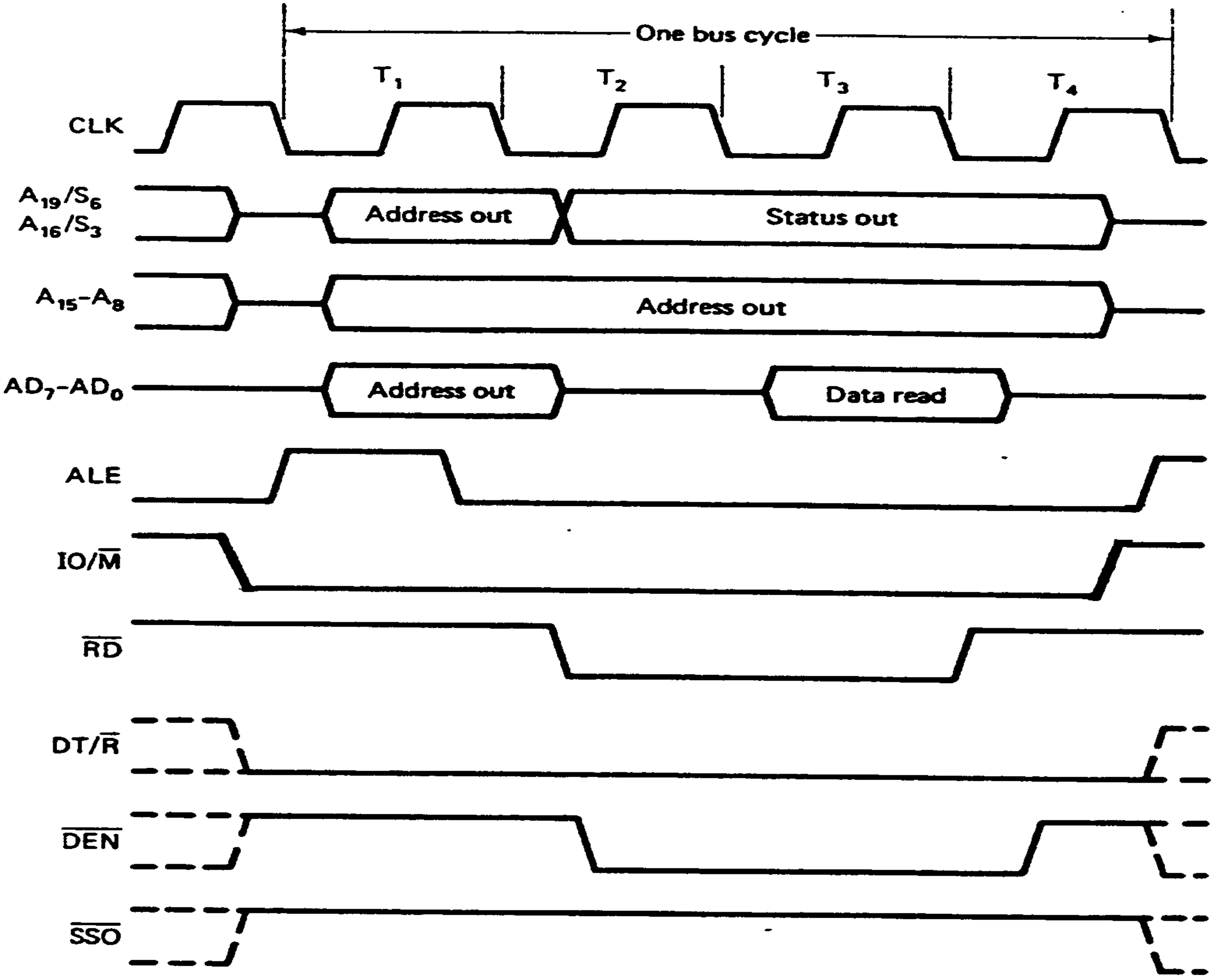
(a)

(b)

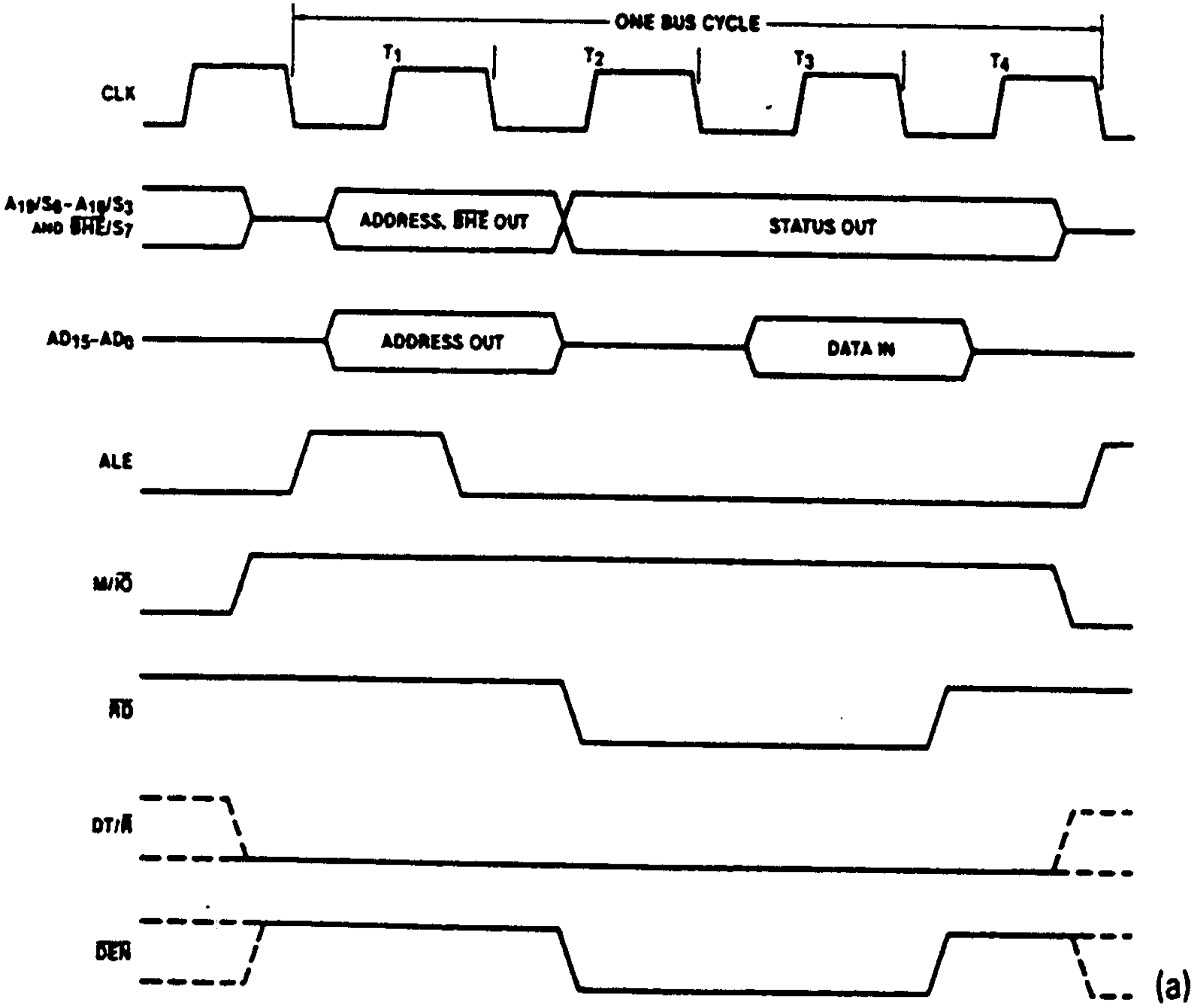
(c)


FIRST BUS CYCLE
x + 3
(x + 1)
x + 2
x
A_{19}-A_1
D_{15}-D_8
$\overline{BHE}$ (LOW)
D_7-D_0
A_0 (HIGH)
SECOND BUS CYCLE
x + 3
x + 1
x + 2
x
A_{19}-A_1
D_{15}-D_8
$\overline{BHE}$ (HIGH)
D_7-D_0
A_0(LOW)

Read Cycle of the 8088

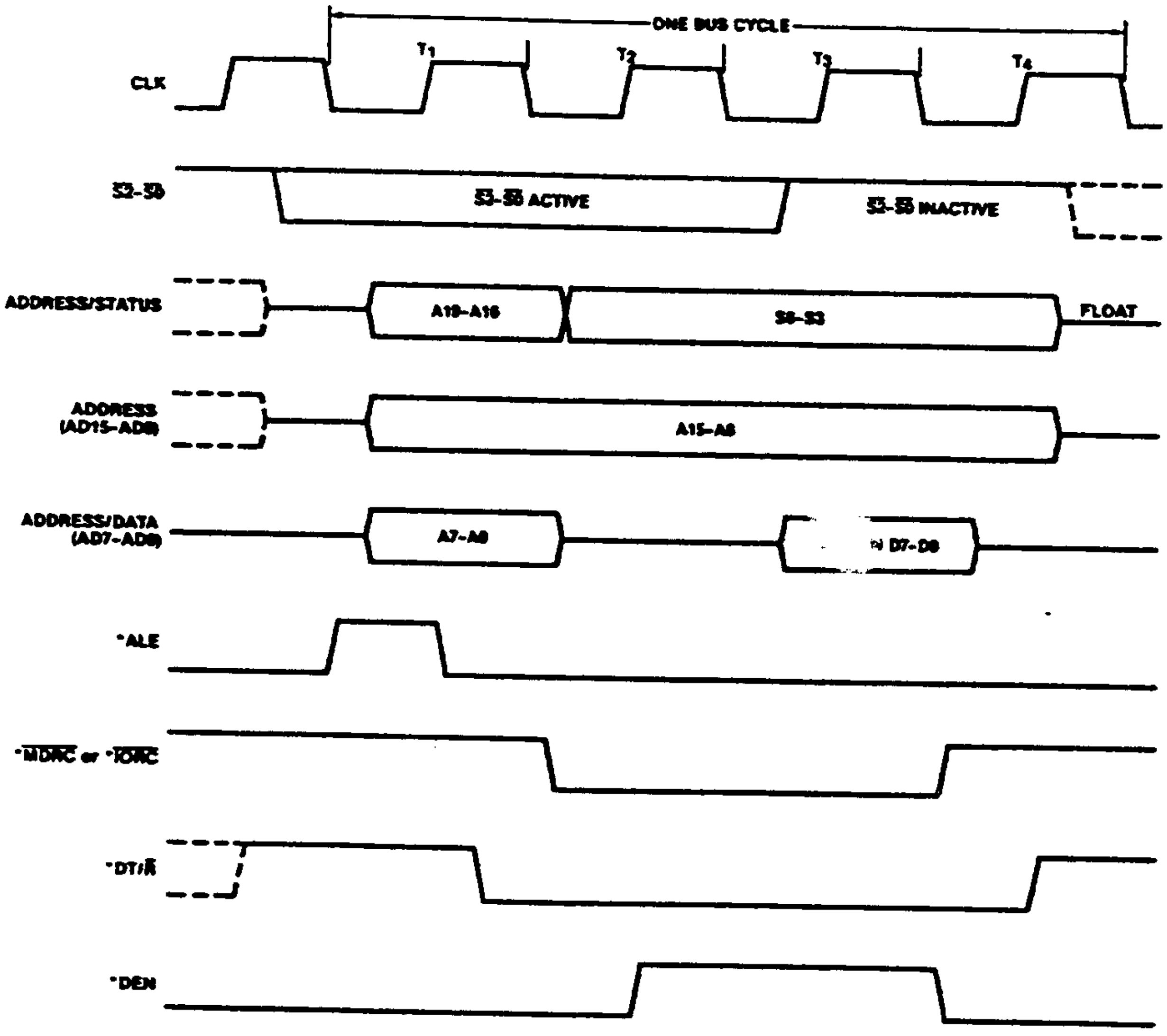


Read Cycle of the 8086 - minimum mode



- $\overline{\text{BHE}}$ is output along with the address during T1
- Data can be read during T3 over all 16 data bus lines
- M/$\overline{\text{IO}}$ replaces IO/$\overline{\text{M}}$
- $\overline{\text{SSO}}$ status signal is not produced

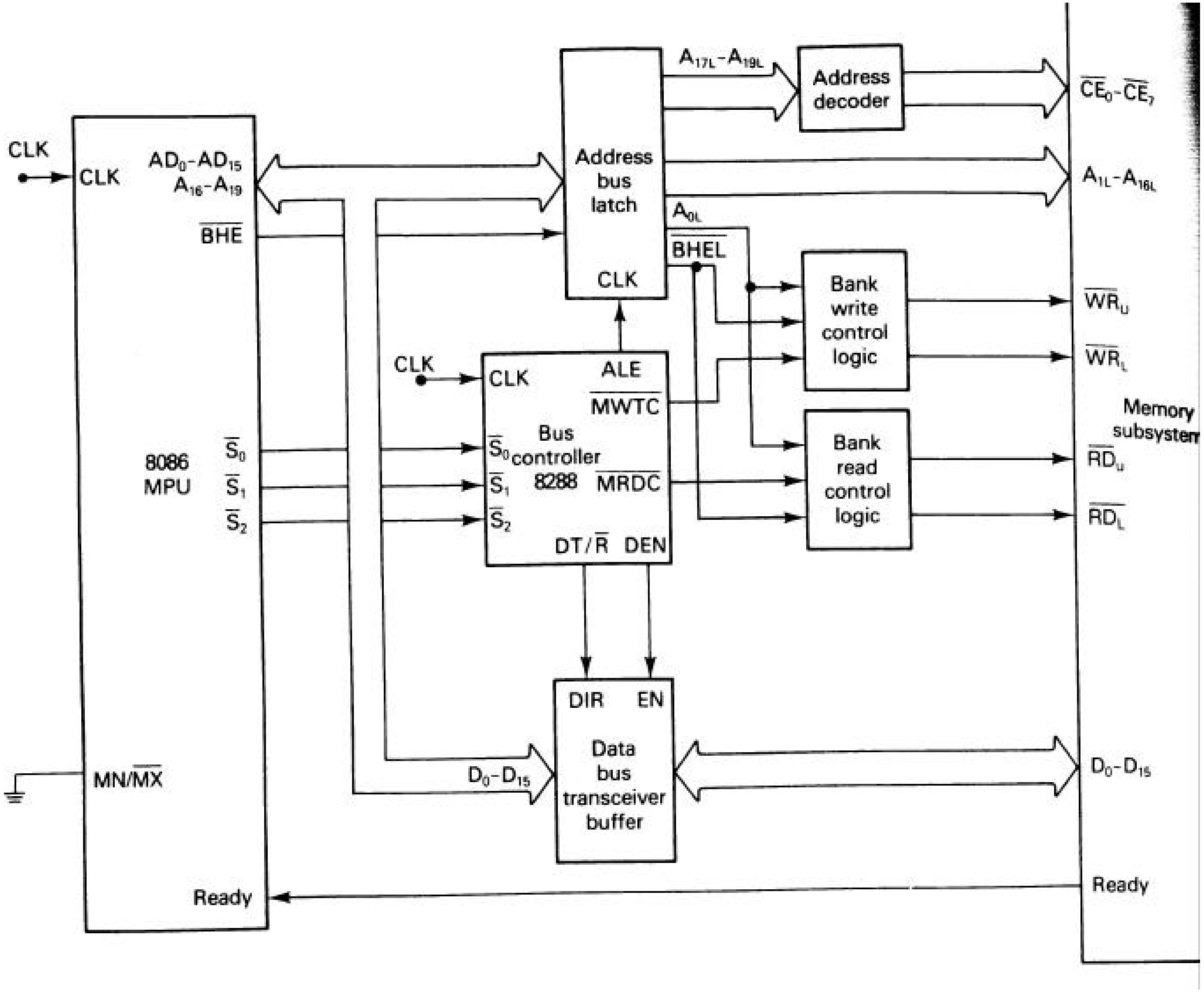
Read Cycle of the 8086 - maximum mode



*8288 BUS CONTROLLER OUTPUTS

(b)

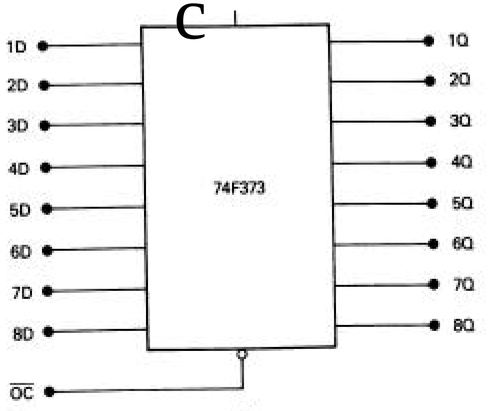
Memory Interface



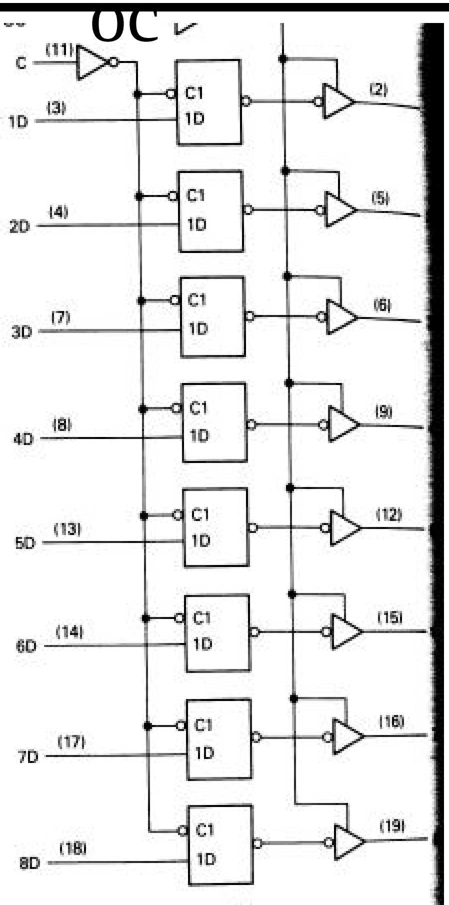
Address Bus Latches and Buffers

C

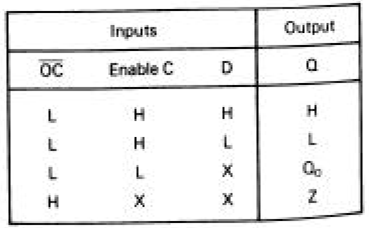
$\overline{\text{OC}}$



(a)



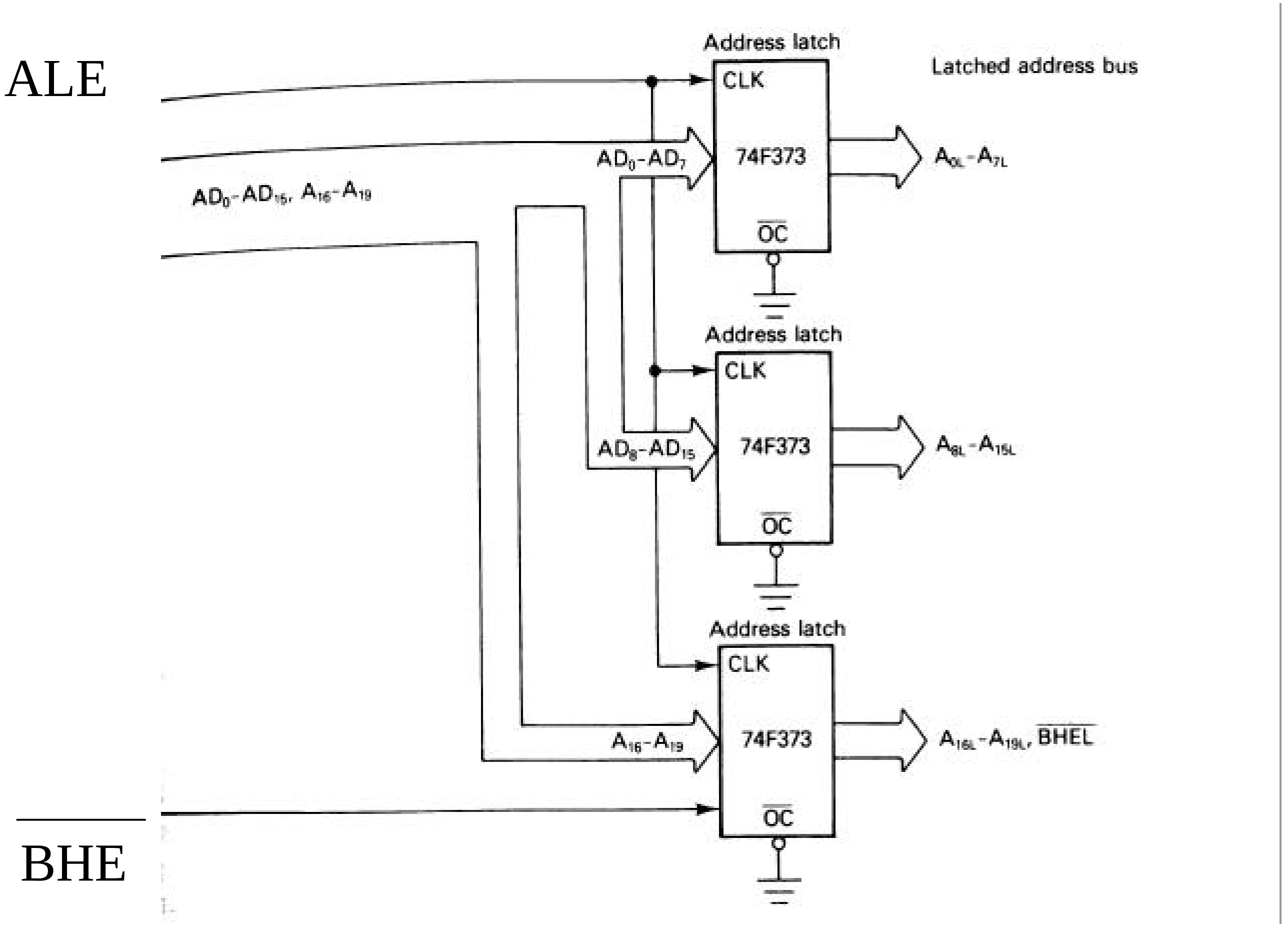
(b)



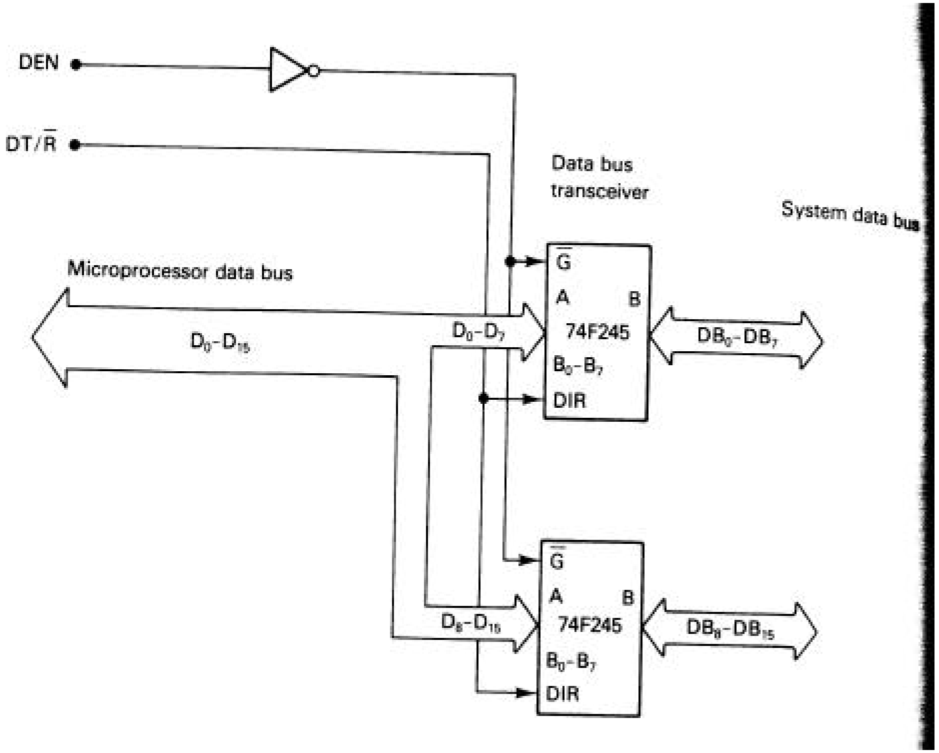
Inputs			Output
$\overline{\text{OC}}$	Enable C	D	Q
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z

(c)

Address Latch Circuit



Data Bus Transceiver Circuit



GBAR ENABLE	DIR	OPERATION
L	L	B data to A
L	H	A data to B
H	x	Isolation

Buffered Systems

- Buffering (boosting) of the control, data, and address busses to provide sufficiently strong signals to drive various IC chips
  - When a pulse leaves an IC chip it can lose some of its strength depending on how far away the receiving IC is located
  - Plus the more pins a signal is connected to (I.e., fanout) the stronger the signal must be to drive them all which requires bus buffering
  - bus buffering = boosting the signals travelling on the busses
  - unidirectional bus 74LS244
  - bidirectional bus 74LS245

8088 system

9–3 BUS BUFFERING AND LATCHING 315

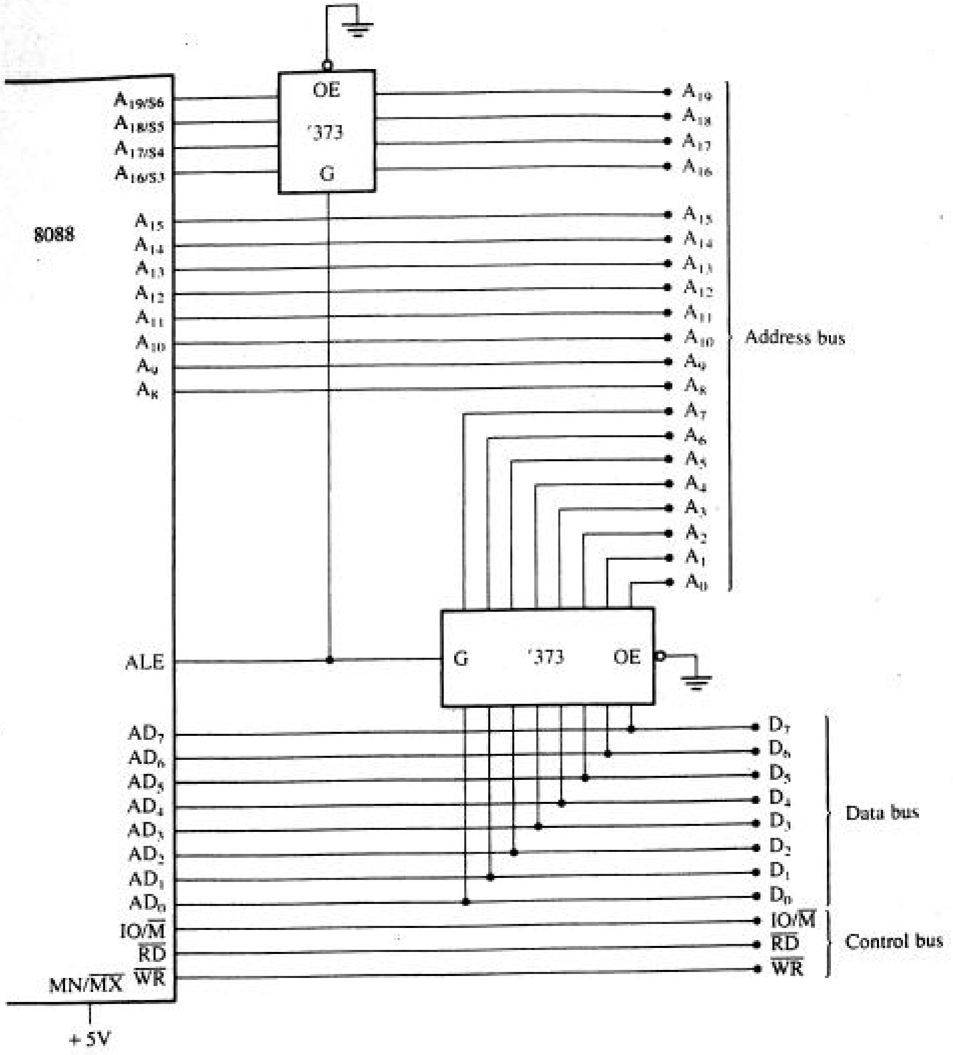
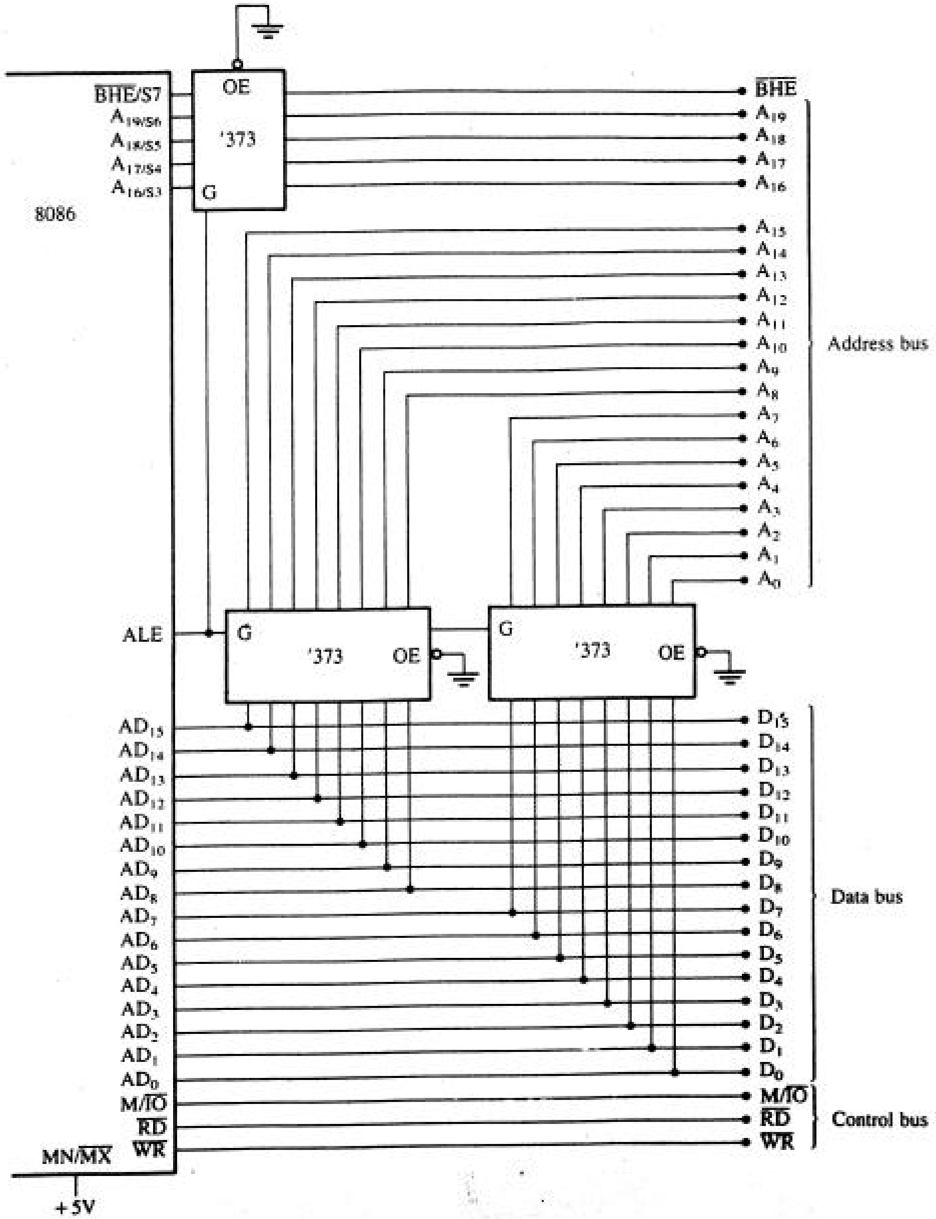


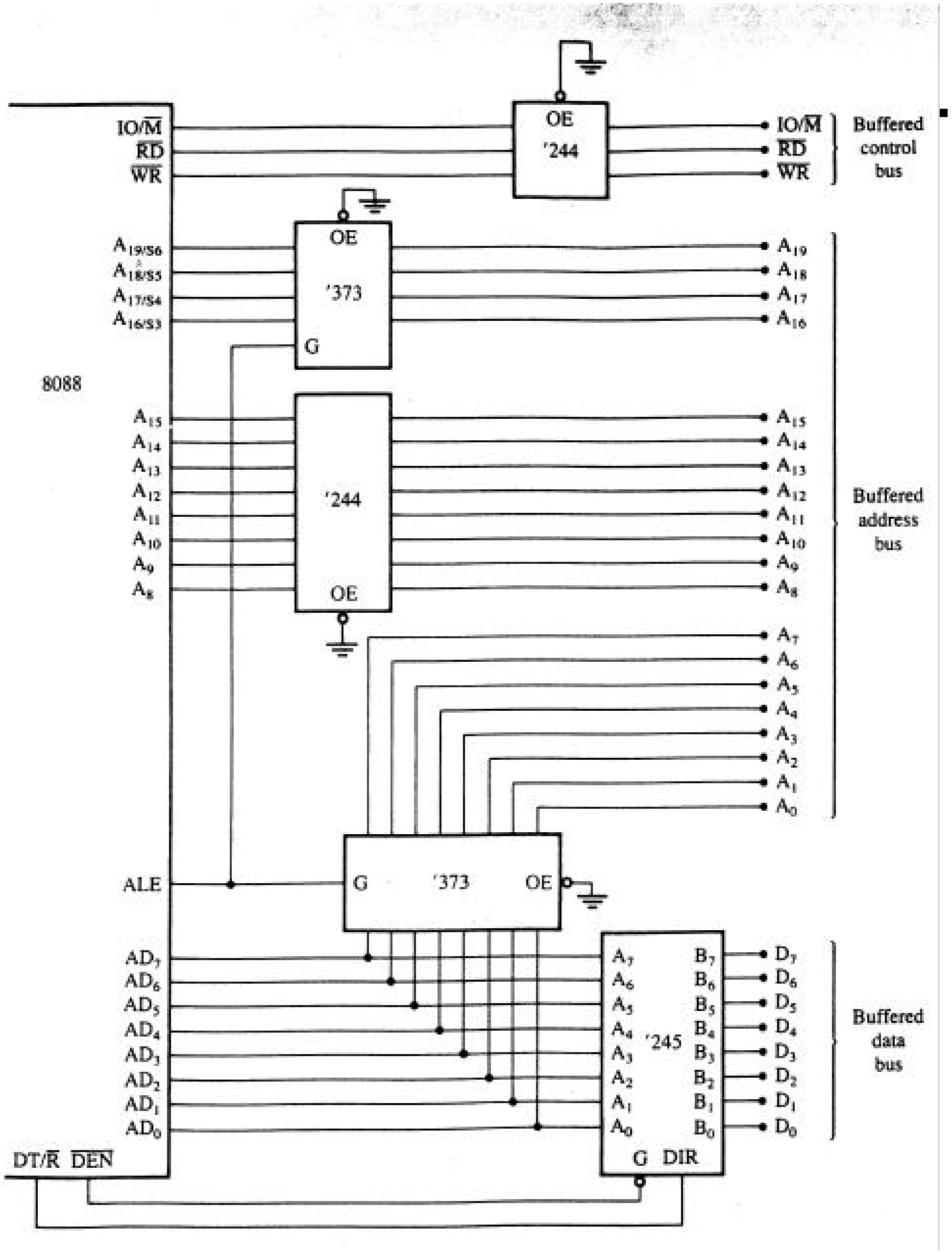
FIGURE 9–5 The 8088 microprocessor shown with a demultiplexed address bus. This is the model used to build many 8088-based systems.

8086 System

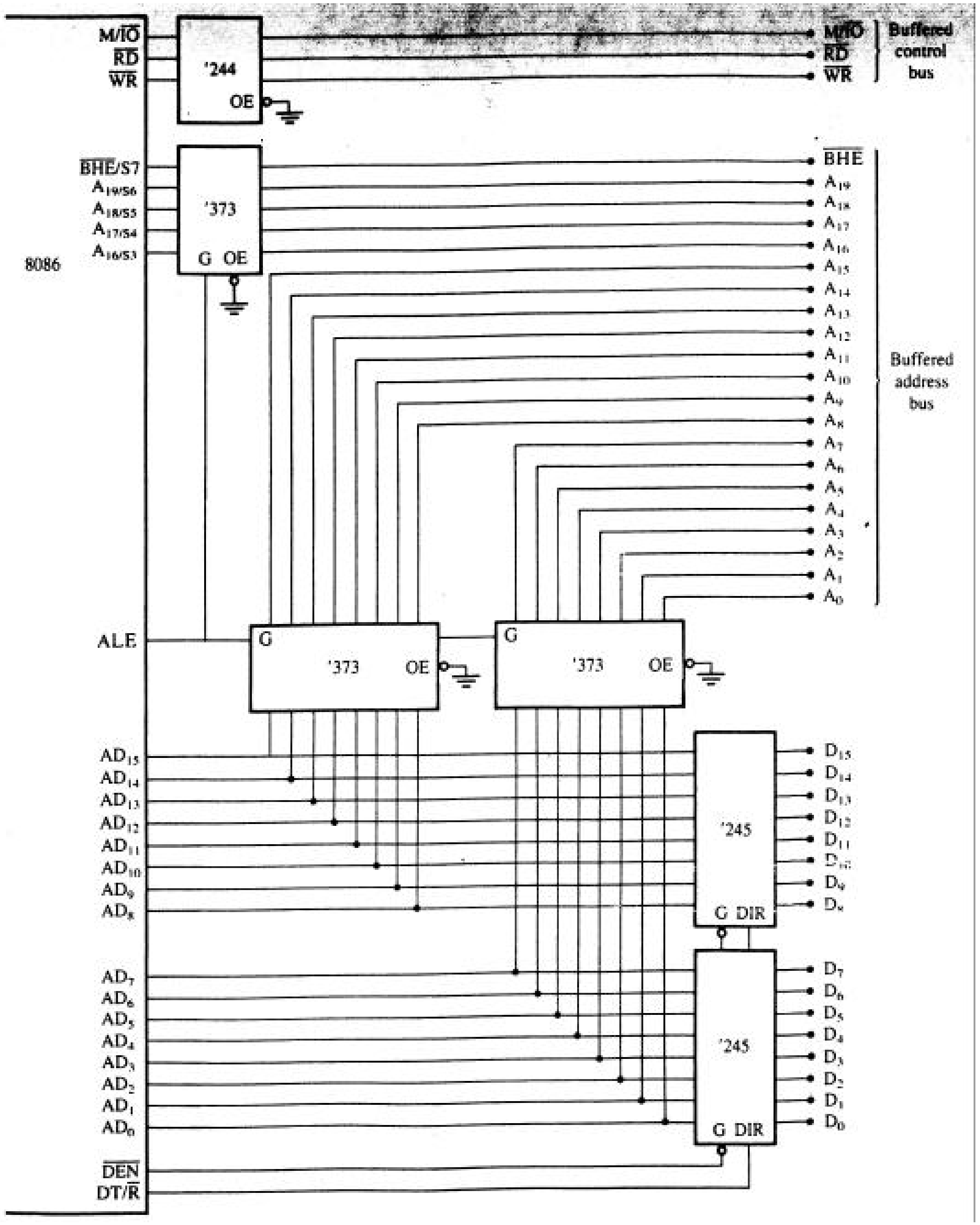


OE
'373
G
$\overline{BHE}$/S7
$A_{19/S6}$
$A_{18/S5}$
$A_{17/S4}$
$A_{16/S3}$
8086
$\overline{BHE}$
A_{19}
A_{18}
A_{17}
A_{16}
A_{15}
A_{14}
A_{13}
A_{12}
A_{11}
A_{10}
A_9
A_8
A_7
A_6
A_5
A_4
A_3
A_2
A_1
A_0
Address bus
ALE
G
'373
OE
G
'373
OE
AD_{15}
AD_{14}
AD_{13}
AD_{12}
AD_{11}
AD_{10}
AD_9
AD_8
AD_7
AD_6
AD_5
AD_4
AD_3
AD_2
AD_1
AD_0
M/$\overline{IO}$
$\overline{RD}$
$\overline{WR}$
MN/$\overline{MX}$
+5V
D_{15}
D_{14}
D_{13}
D_{12}
D_{11}
D_{10}
D_9
D_8
D_7
D_6
D_5
D_4
D_3
D_2
D_1
D_0
Data bus
M/$\overline{IO}$
$\overline{RD}$
$\overline{WR}$
Control bus

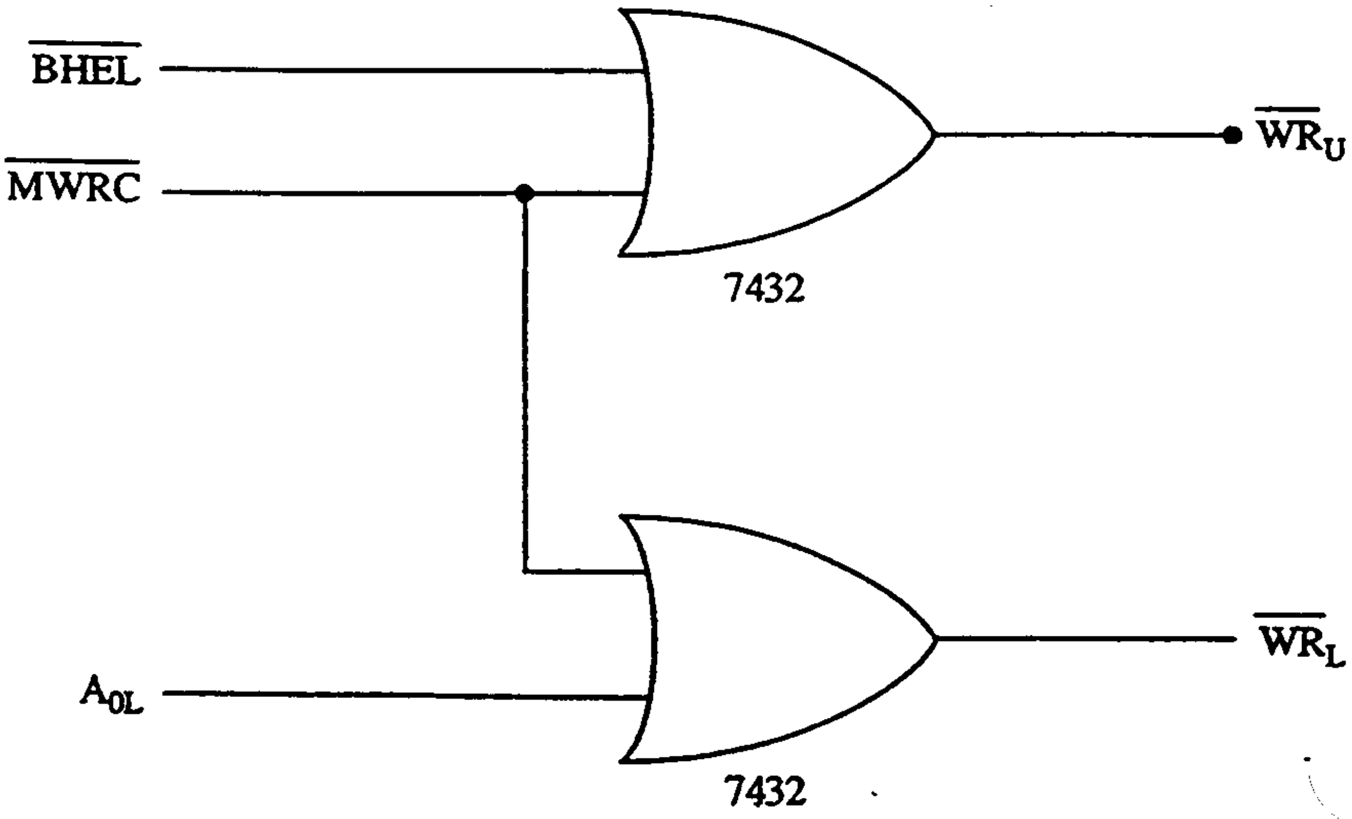
Fully buffered 8088



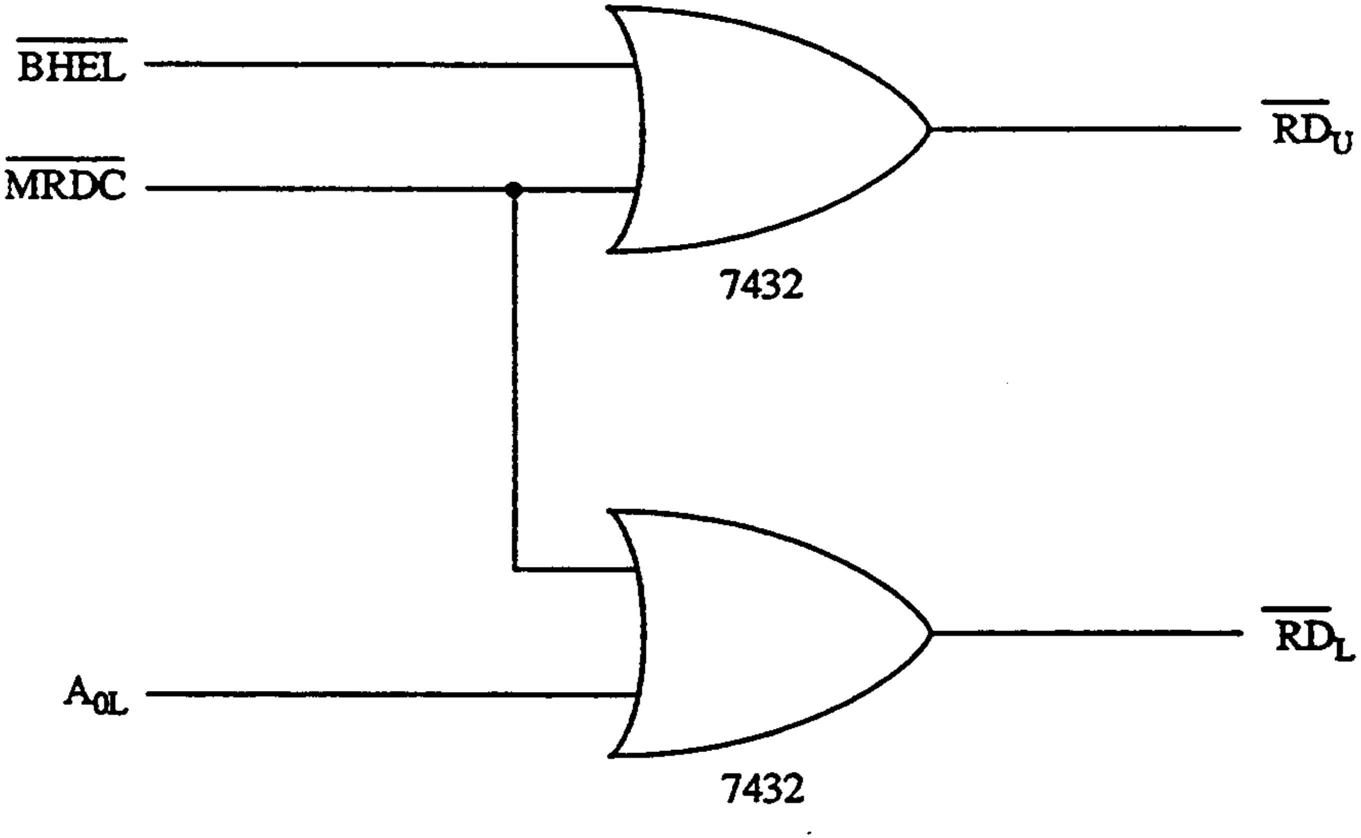
Fully Buffered 8086



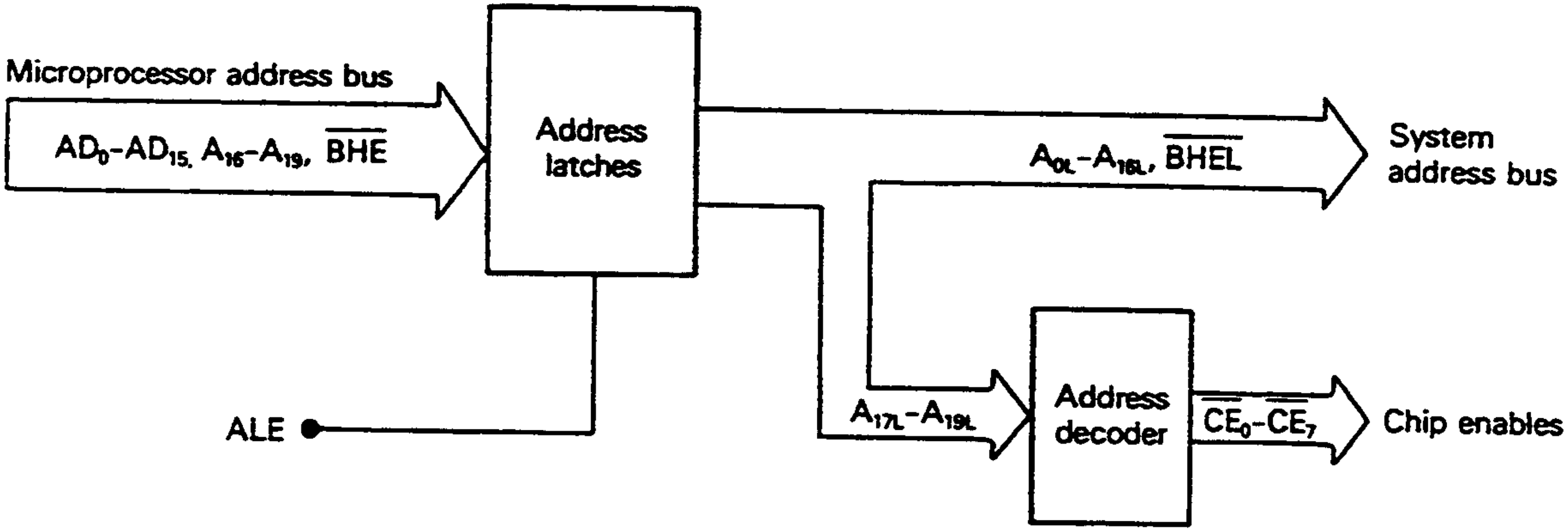
Bank Write Control Logic



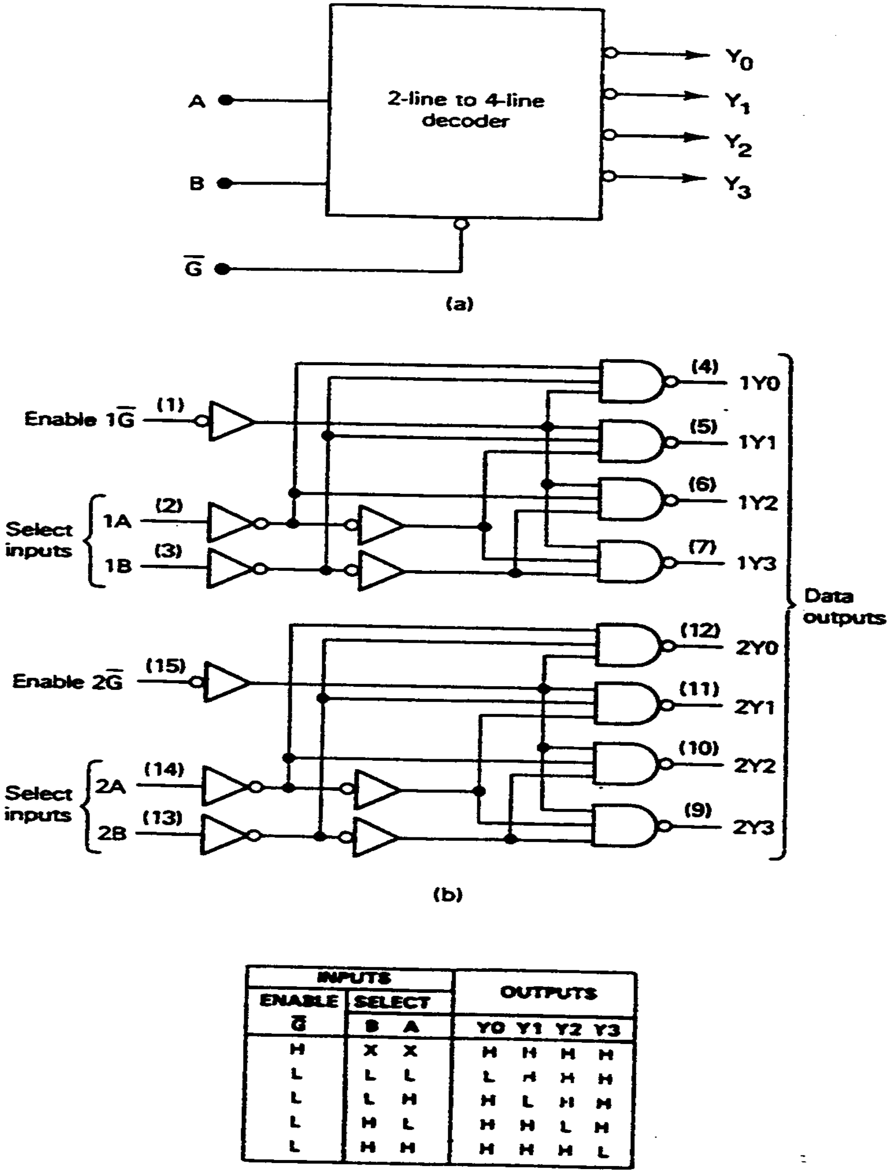
Bank Read Control Logic



Address Bus Configuration with Address Decoding



74F139 2-line to 4-line decoder



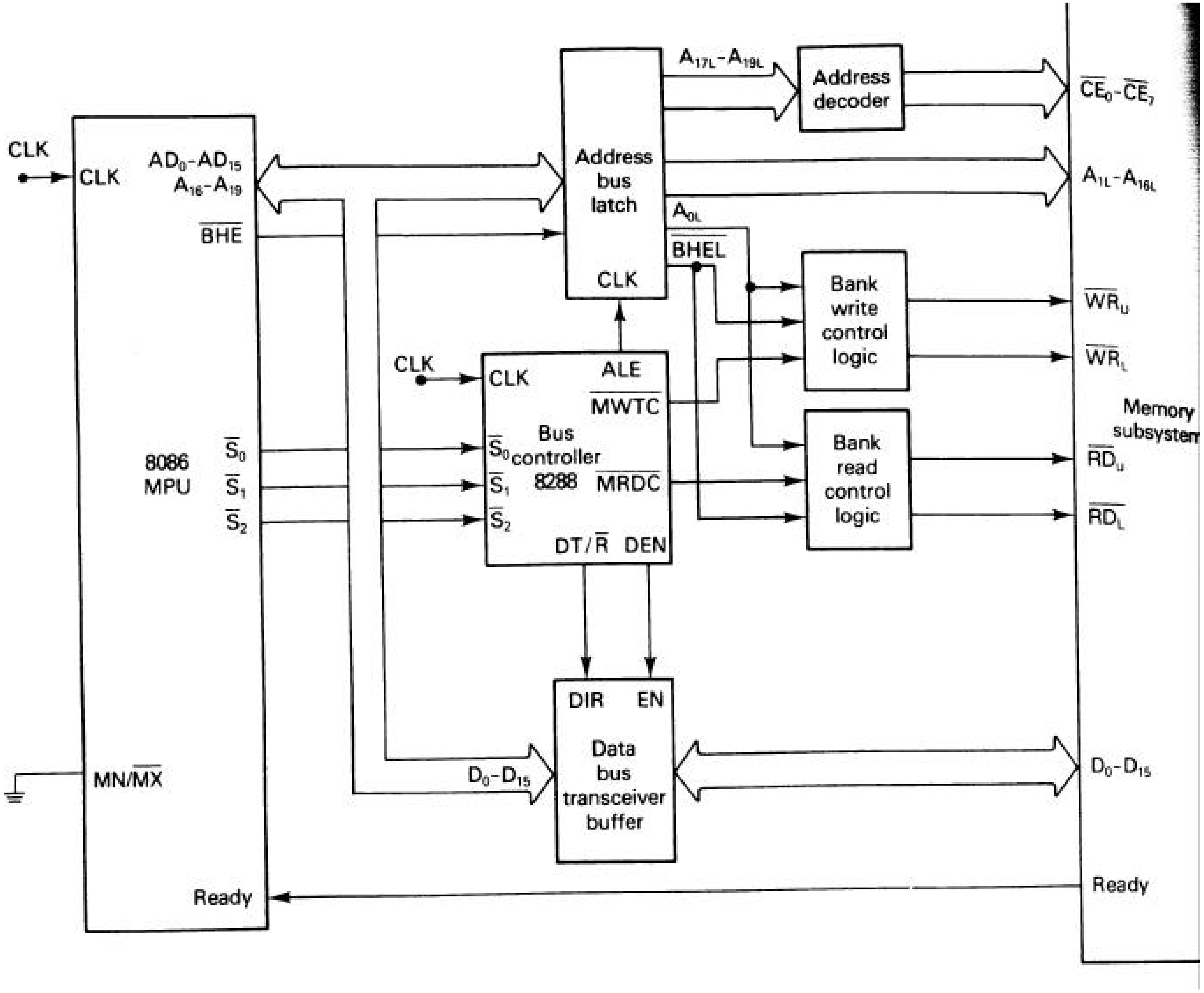
(a)

(b)

INPUTS			OUTPUTS			
ENABLE	SELECT					
$\overline{G}$	B	A	Y0	Y1	Y2	Y3
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	L	H	H	L	H	H
L	H	L	H	H	L	H
L	H	H	H	H	H	L

(c)

Memory Interface - Again



Address Decoder Circuit

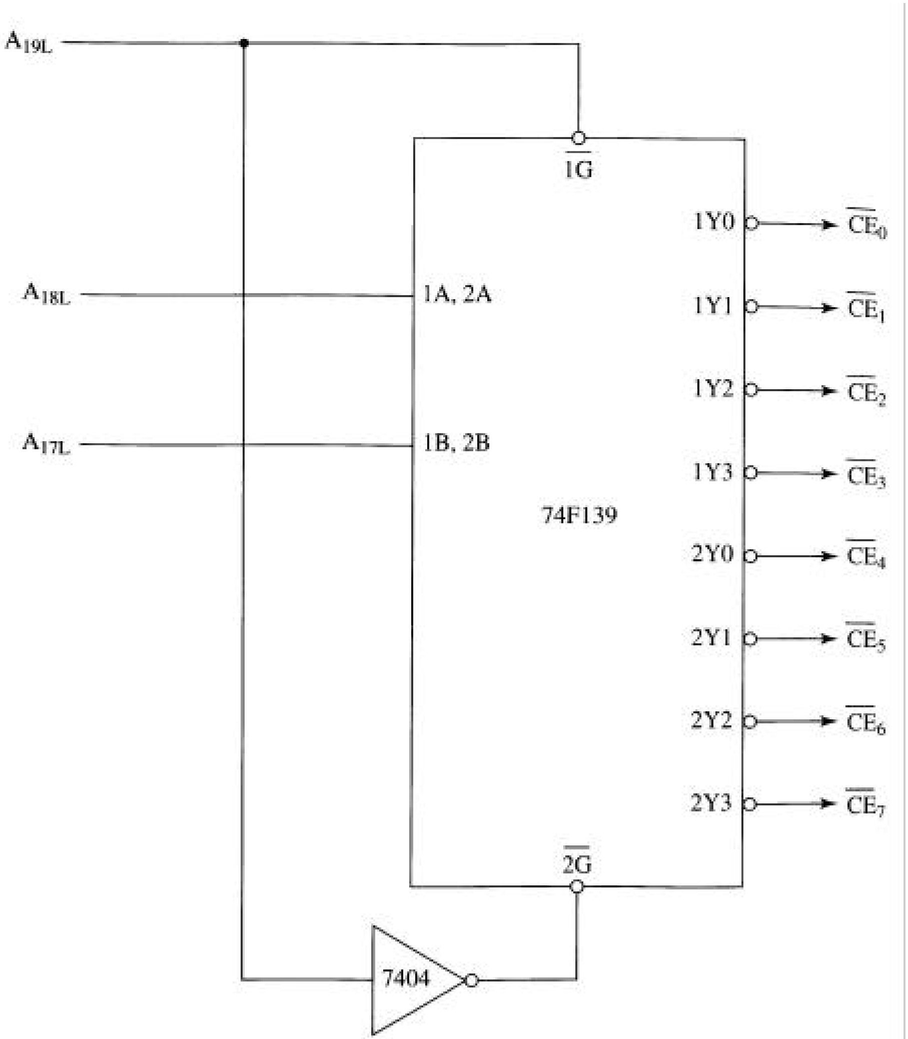


Figure 8–35 Address decoder circuit

8088 memory and address spaces

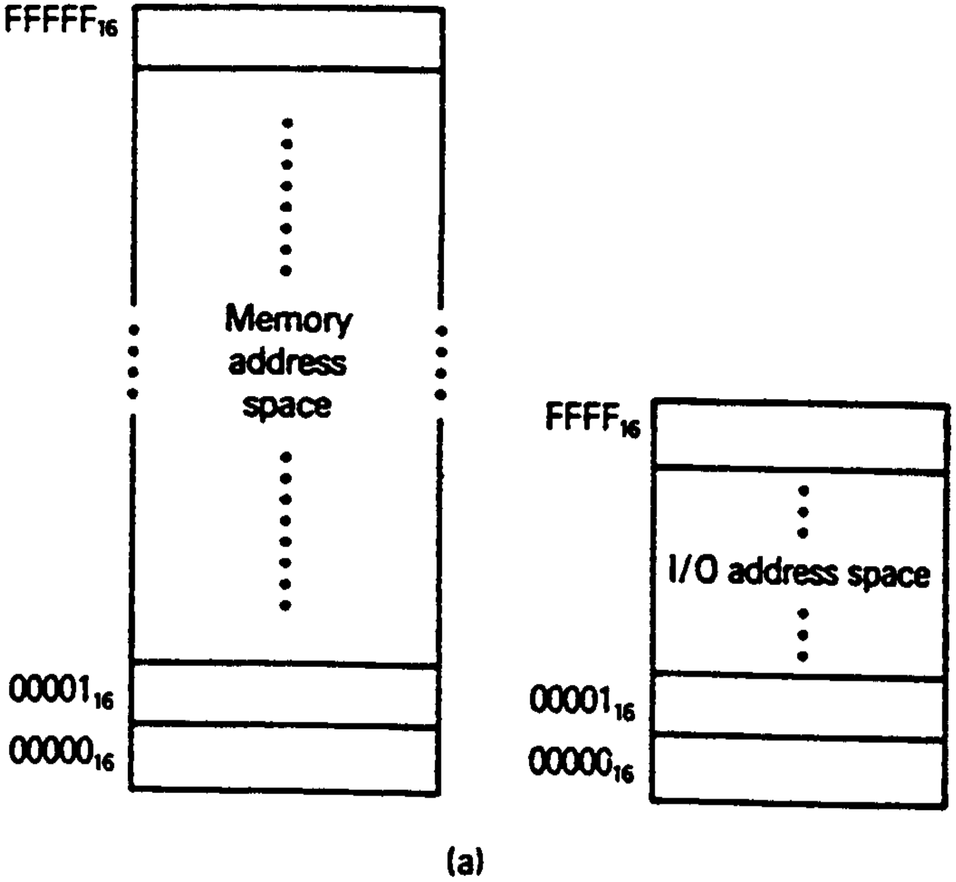
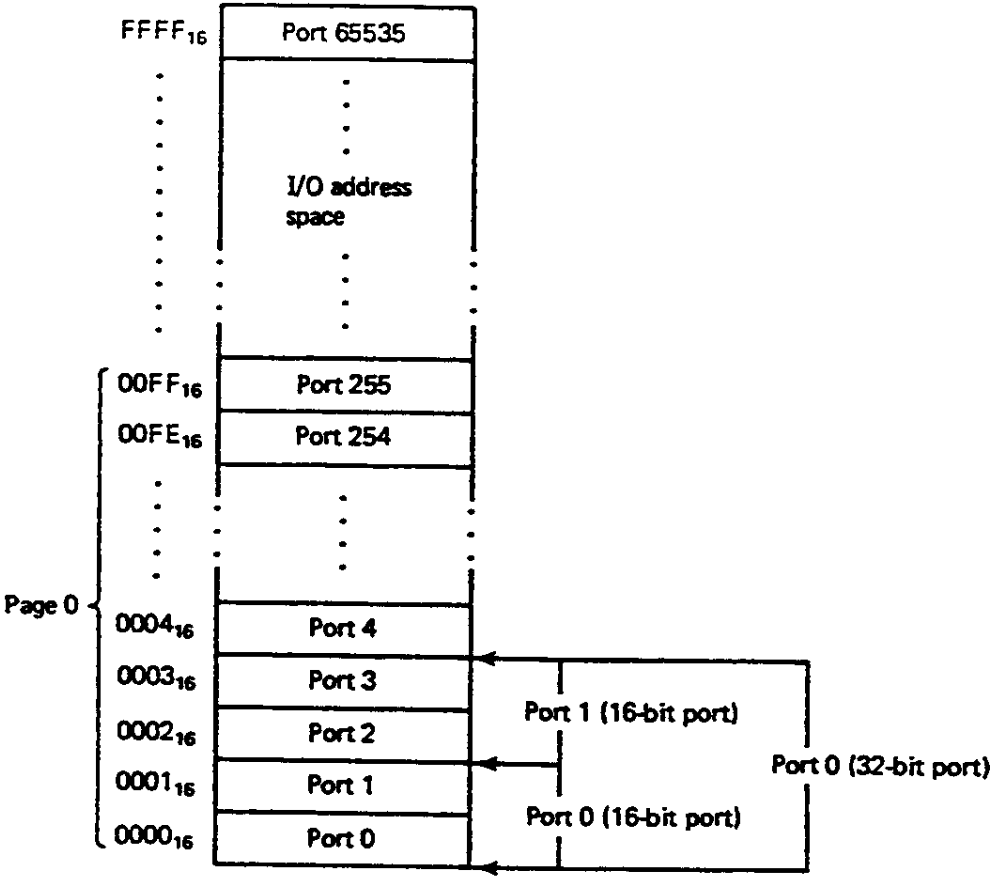


Figure 8–46 8088/8086 memory and I/O address spaces.

Isolated I/O



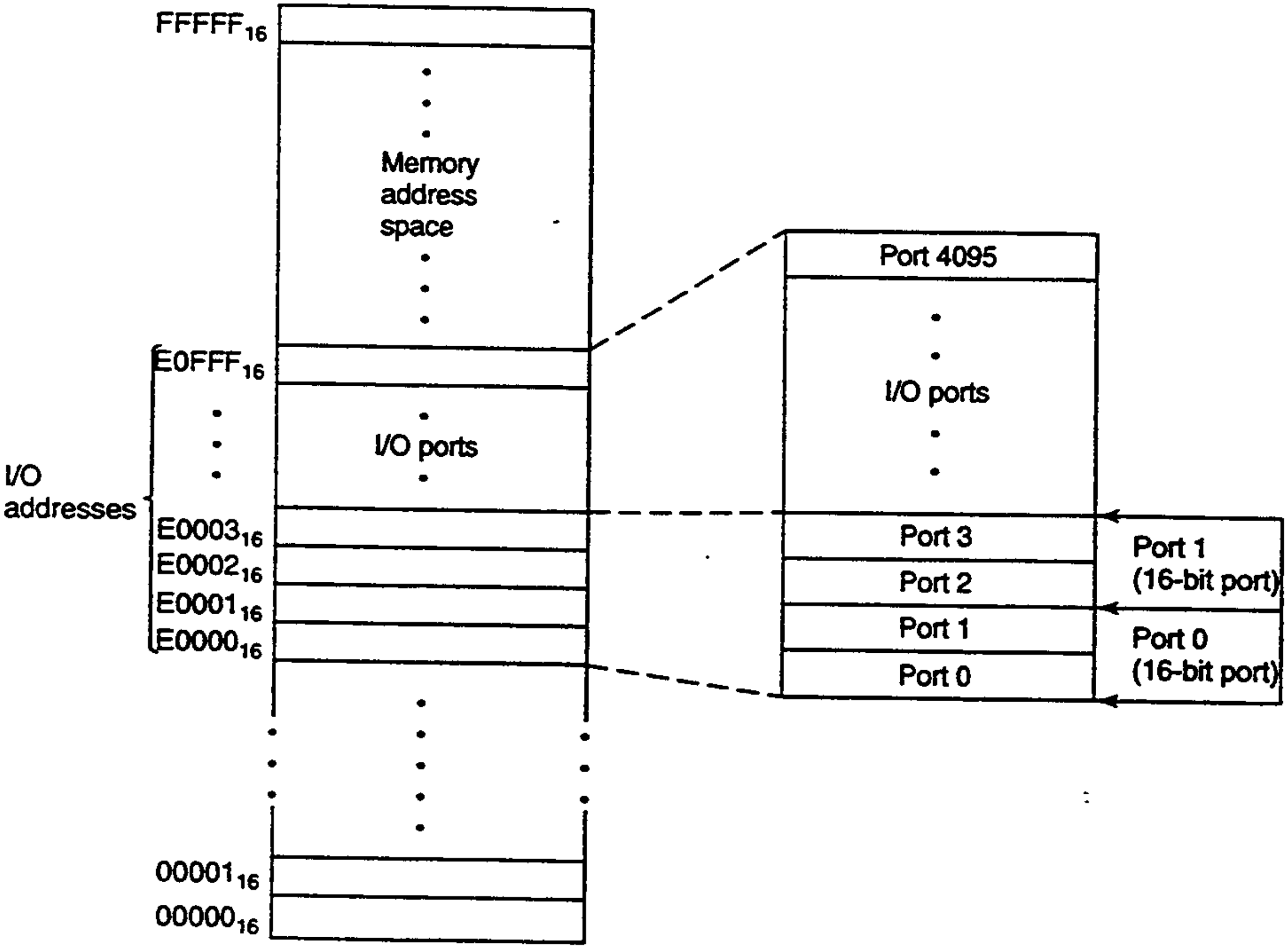
I/O devices are treated separately from memory

Address 0000 to 00FF is referred to page 0. Special instructions exist for this address range

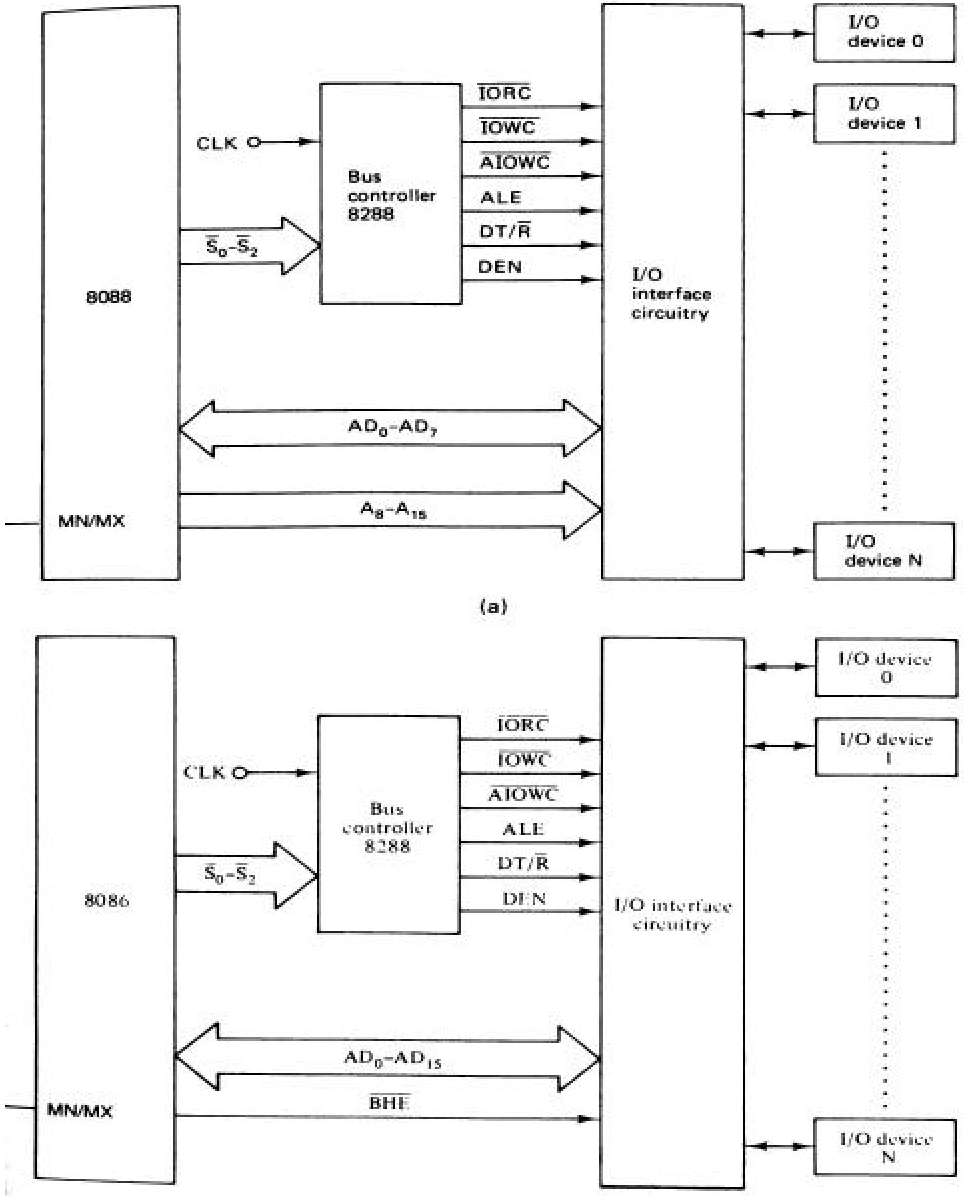
Byte wide ports

Word wide ports

Memory Mapped I/O



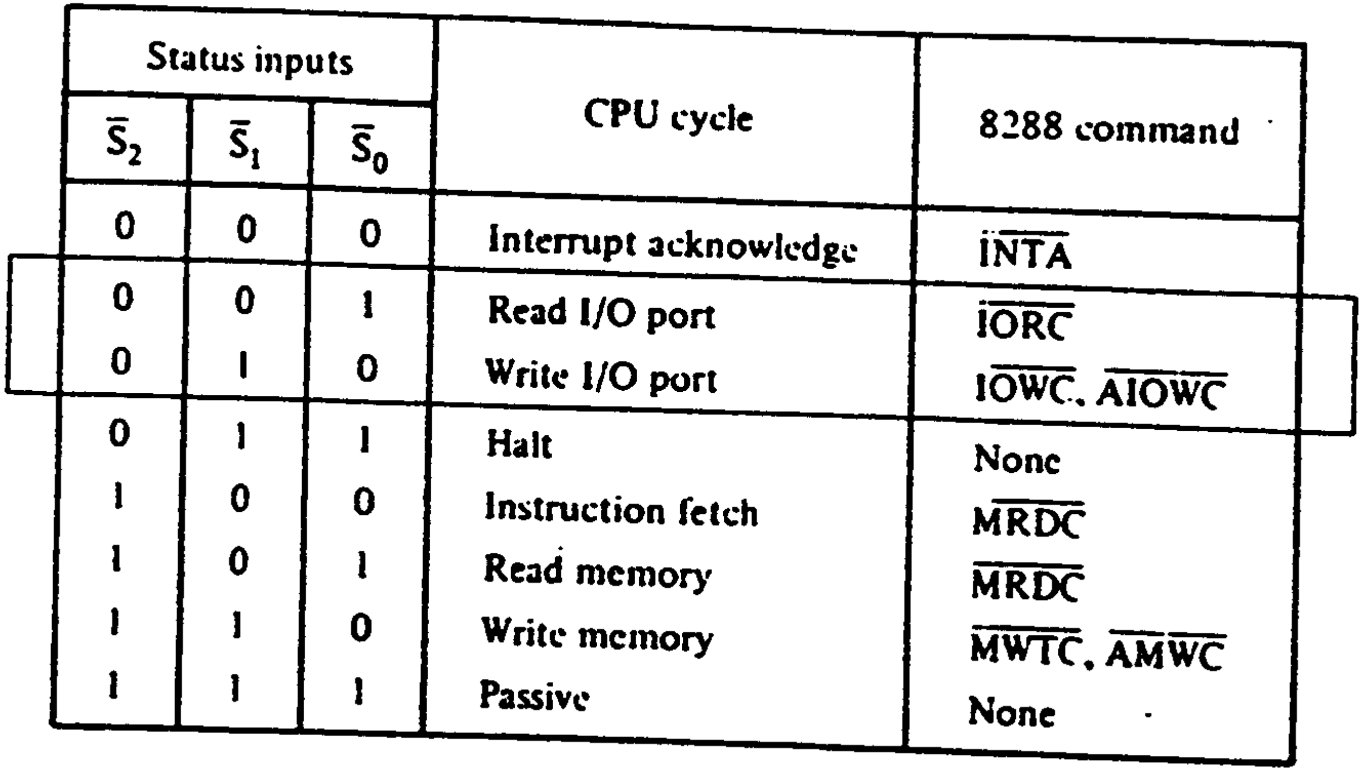
Maximum Mode I/O interface - 8088/8086



(a)

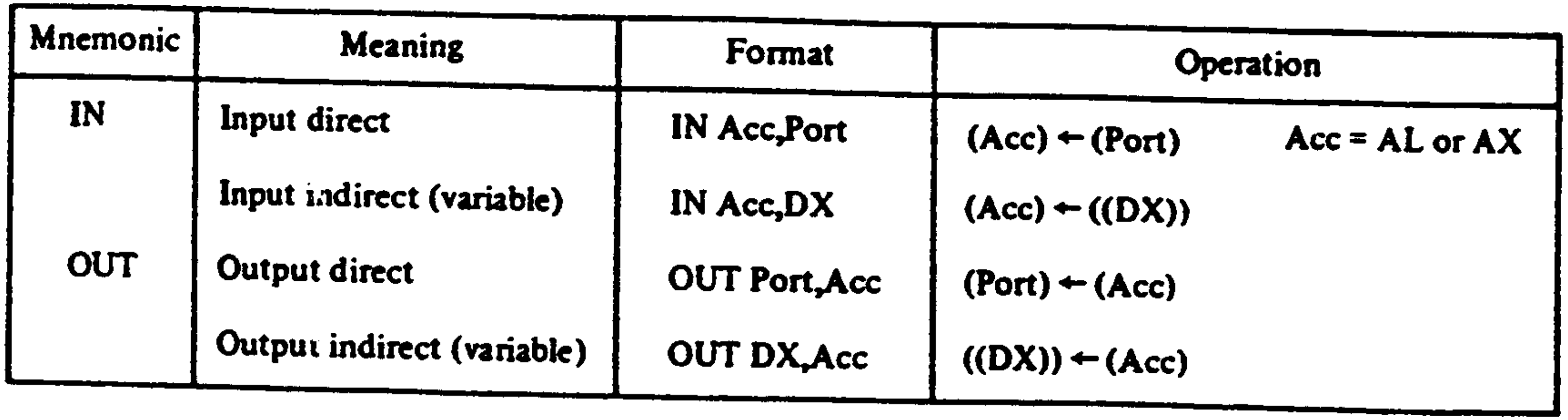
(b)

I/O Bus Cycle Status Codes



Status inputs			CPU cycle	8288 command
$\overline{S}_2$	$\overline{S}_1$	$\overline{S}_0$		
0	0	0	Interrupt acknowledge	$\overline{\text{INTA}}$
0	0	1	Read I/O port	$\overline{\text{IORC}}$
0	1	0	Write I/O port	$\overline{\text{IOWC}}$, $\overline{\text{AIOWC}}$
0	1	1	Halt	None
1	0	0	Instruction fetch	$\overline{\text{MRDC}}$
1	0	1	Read memory	$\overline{\text{MRDC}}$
1	1	0	Write memory	$\overline{\text{MWTC}}$, $\overline{\text{AMWC}}$
1	1	1	Passive	None

I/O Instructions



Mnemonic	Meaning	Format	Operation
IN	Input direct	IN Acc,Port	(Acc) ← (Port) Acc = AL or AX
	Input indirect (variable)	IN Acc,DX	(Acc) ← ((DX))
OUT	Output direct	OUT Port,Acc	(Port) ← (Acc)
	Output indirect (variable)	OUT DX,Acc	((DX)) ← (Acc)

Example. Write a sequence of instructions that will output the data FFh to a byte wide output at address ABh of the I/O address space

```
MOV AL,0FFh
OUT 0ABh, AL
```

Example. Data is to be read from two byte wide input ports at addresses AAh and A9h and then this data will then be output to a word wide output port at address B000h

```
IN AL, 0AAh
MOV AH,AL
IN AL, 0A9h
MOV DX,0B00h
OUT DX,AX
```

Input Bus Cycle of the 8088

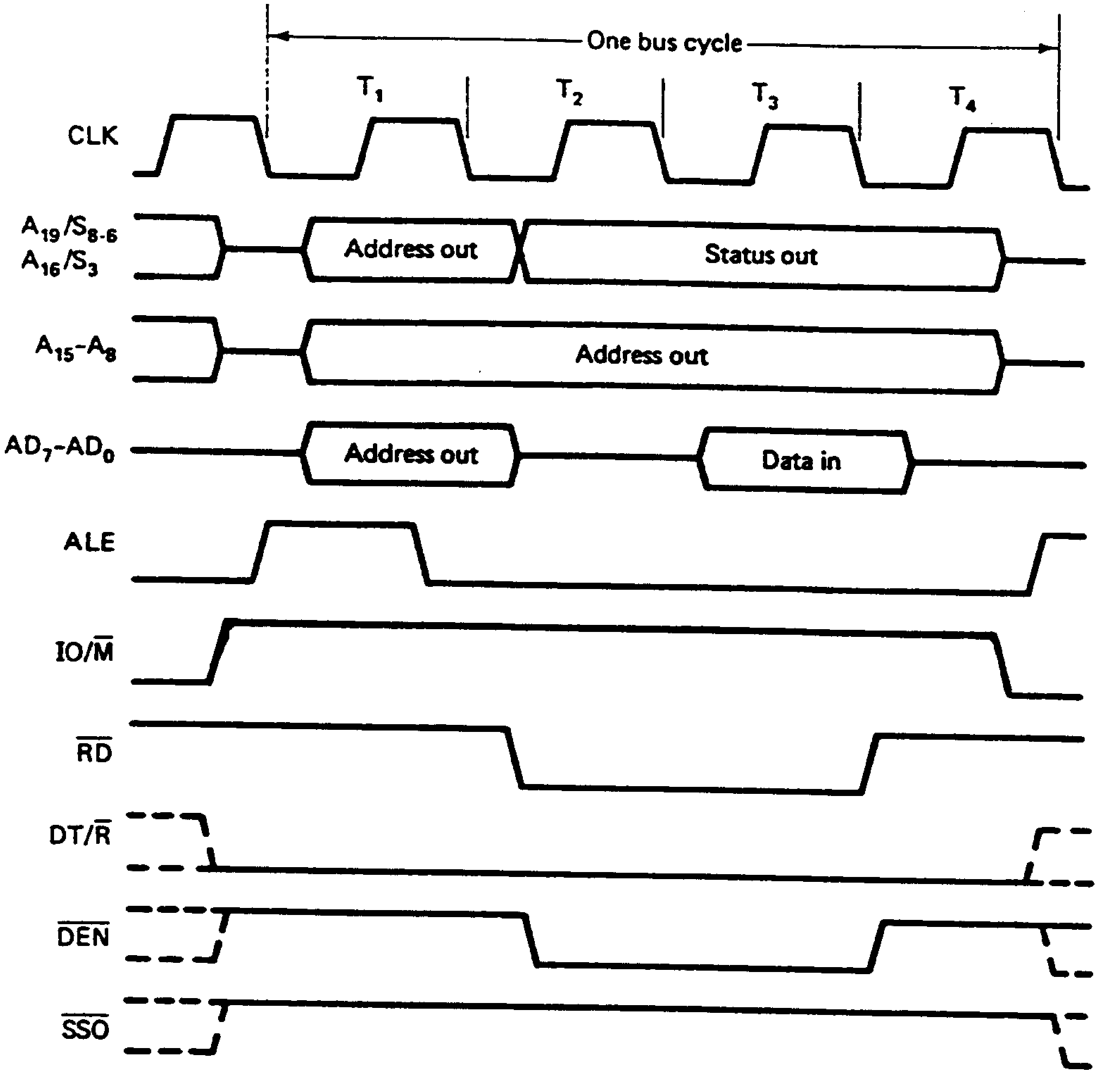


Figure 8–52 Input bus cycle of the 8088.

Output Bus Cycle of the 8088

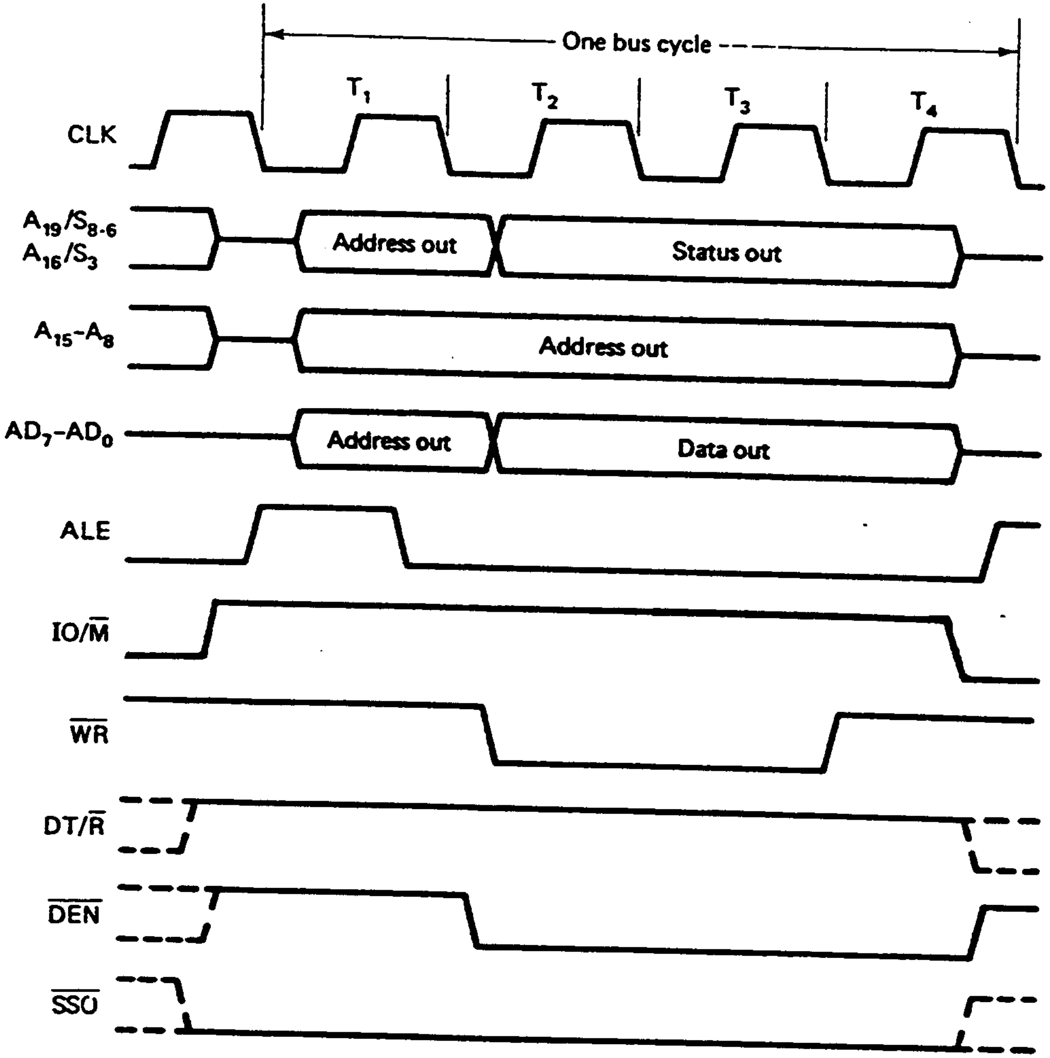


Figure 8–53 Output bus cycle of the 8088.