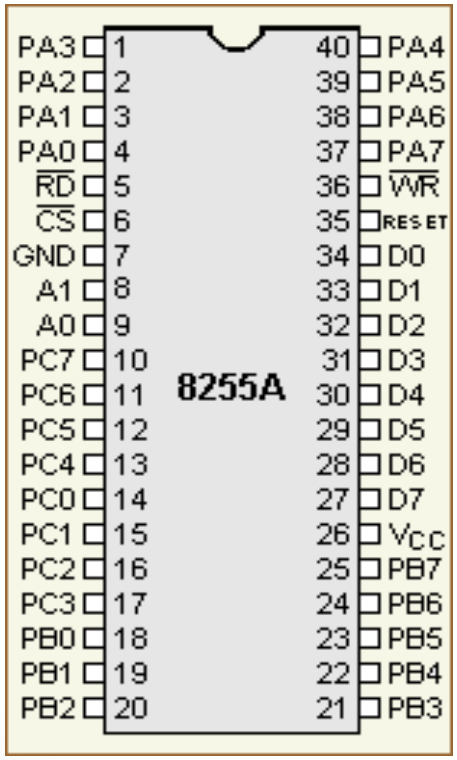
8255 PPI

PPI 8255

The Intel 8255A is a general purpose programmable I/O device which is designed for use with all Intel and most other microprocessors. It provides 24 I/O pins which may be individually programmed in 2 groups of 12 pin and used in 3 major modes of operation.

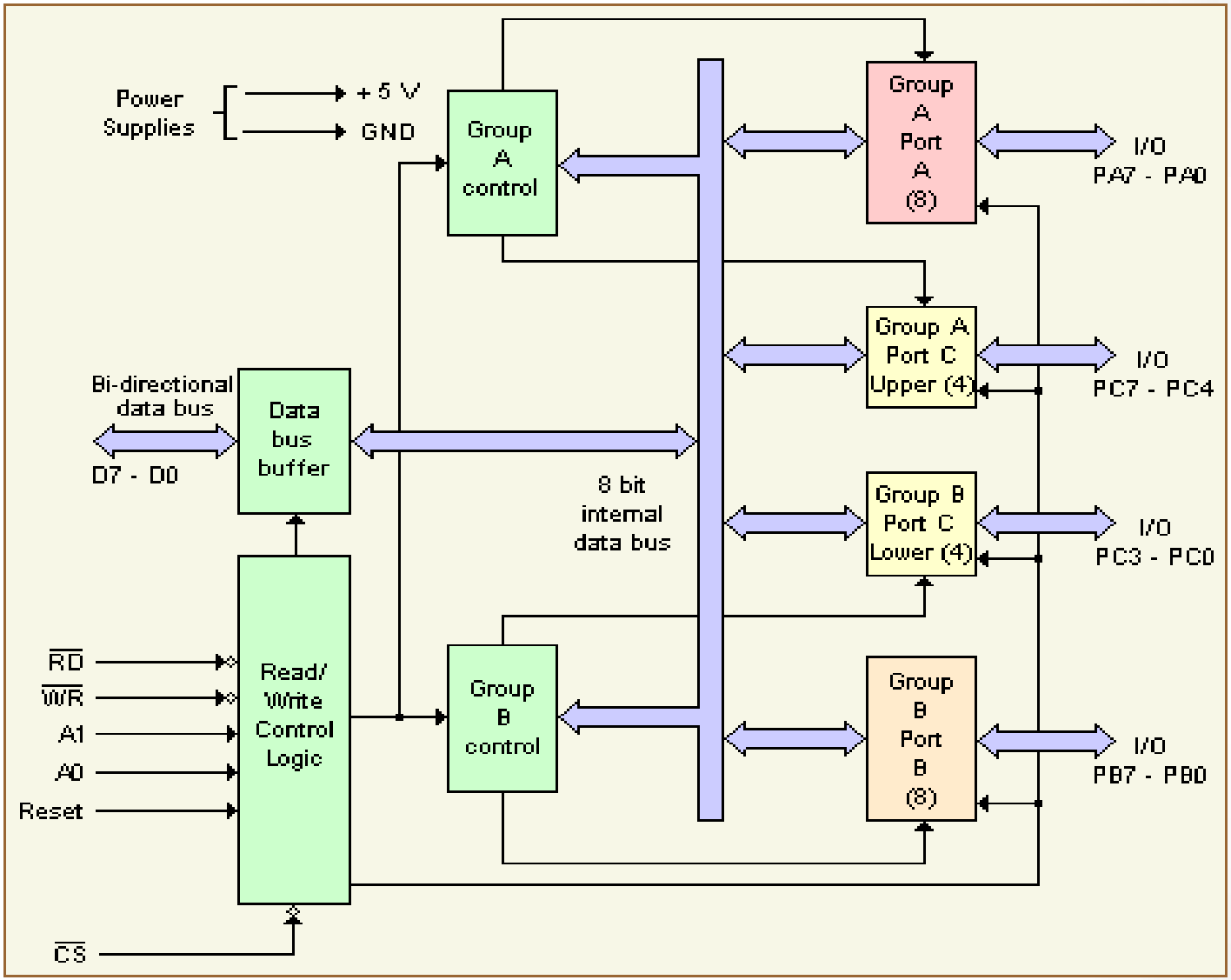


In MODE 0, each group of 12 I/O pins may be programmed in sets of 4 and 8 to be inputs or outputs. In MODE 1, each group may be programmed to have 8 lines of input or output. 3 of the remaining 4 pins are used for handshaking and interrupt control signals. MODE 2 is a strobed bi-directional bus configuration.

Features:

- 3 IO ports PA, PB, PC & TTL compatible
- It is programmable ;parallel I/O device
- Fully compatible with intel microprocessor
- PC has 2 4-bit parts: PC upper (PCU) and PC lower (PCL), each can be set independently for I or O. Each PC bit can be set/reset individually in BSR mode.
- PA and PCU are Group A (GA) and PB and PCL are Group B (GB)
- Direct bit set/reset capability is available for port C
- . Operates in 3 modes mode 0, mode 1, mode 2
- Improved dc driving capability

Block diagram of 8255



D0 D7 - These are the data input/output lines for the device. All information read from and written to the 8255 occurs via these 8 data lines.

CS (*Chip Select Input*). If this line is a logical 0, the microprocessor can read and write to the 8255.

RD (*Read Input*) Whenever this input line is a logical 0 and the RD input is a logical 0, the 8255 data outputs are enabled onto the system data bus.

WR (*Write Input*) Whenever this input line is a logical 0 and the CS input is a logical 0, data is written to the 8255 from the system data bus.

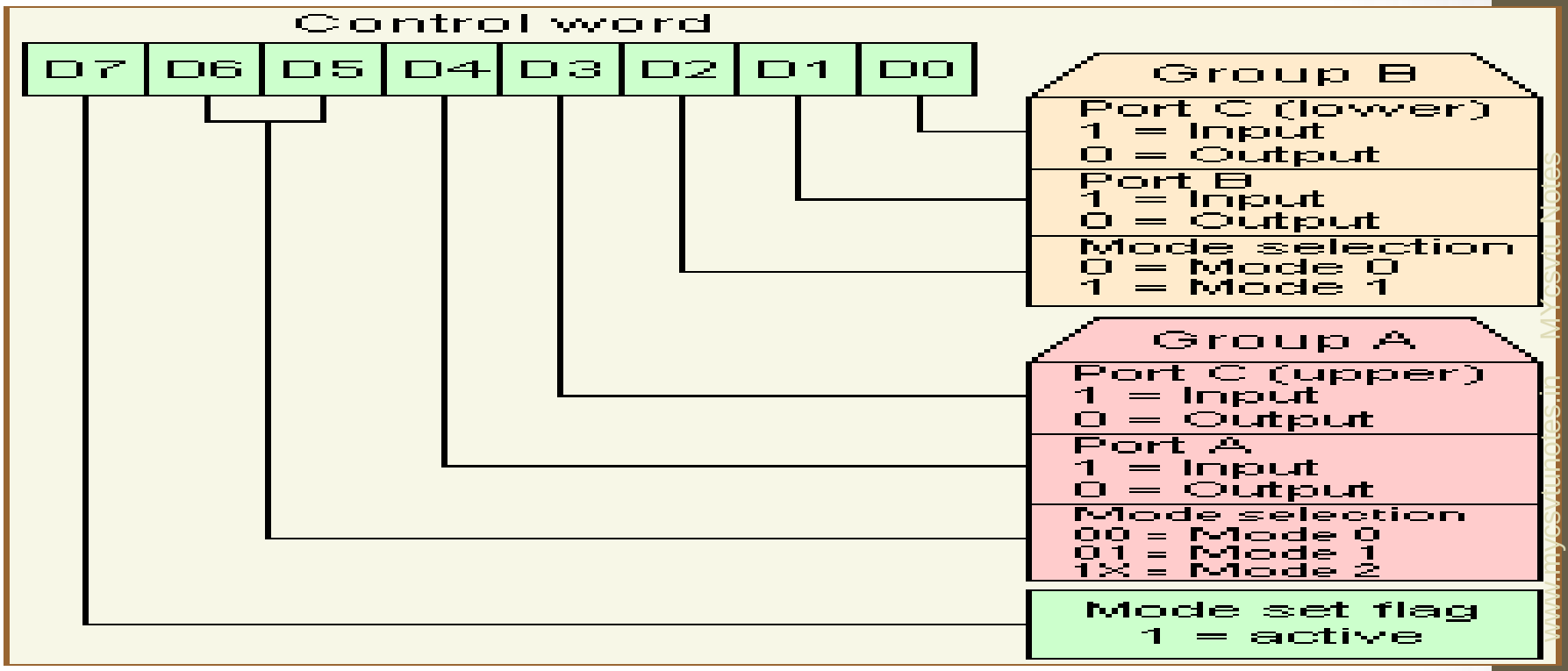
A0 - A1 (*Address Inputs*) The logical combination of these two input lines determines which internal register of the 8255 data is written to or read from.

RESET The 8255 is placed into its reset state if this input line is a logical 1. All peripheral ports are set to the input mode.

PA0 - PA7, **PB0 - PB7**, **PC0 - PC7** These signal lines are used as 8-bit I/O ports. They can be connected to peripheral devices. The 8255 has three 8 bit I/O ports and each one can be connected to the physical lines of an external device. These lines are labeled PA0-PA7, PB0-PB7, and **PC0-PC7.** The groups of the signals are divided into three different I/O ports labeled port A (PA), port B (PB), and port C (PC).

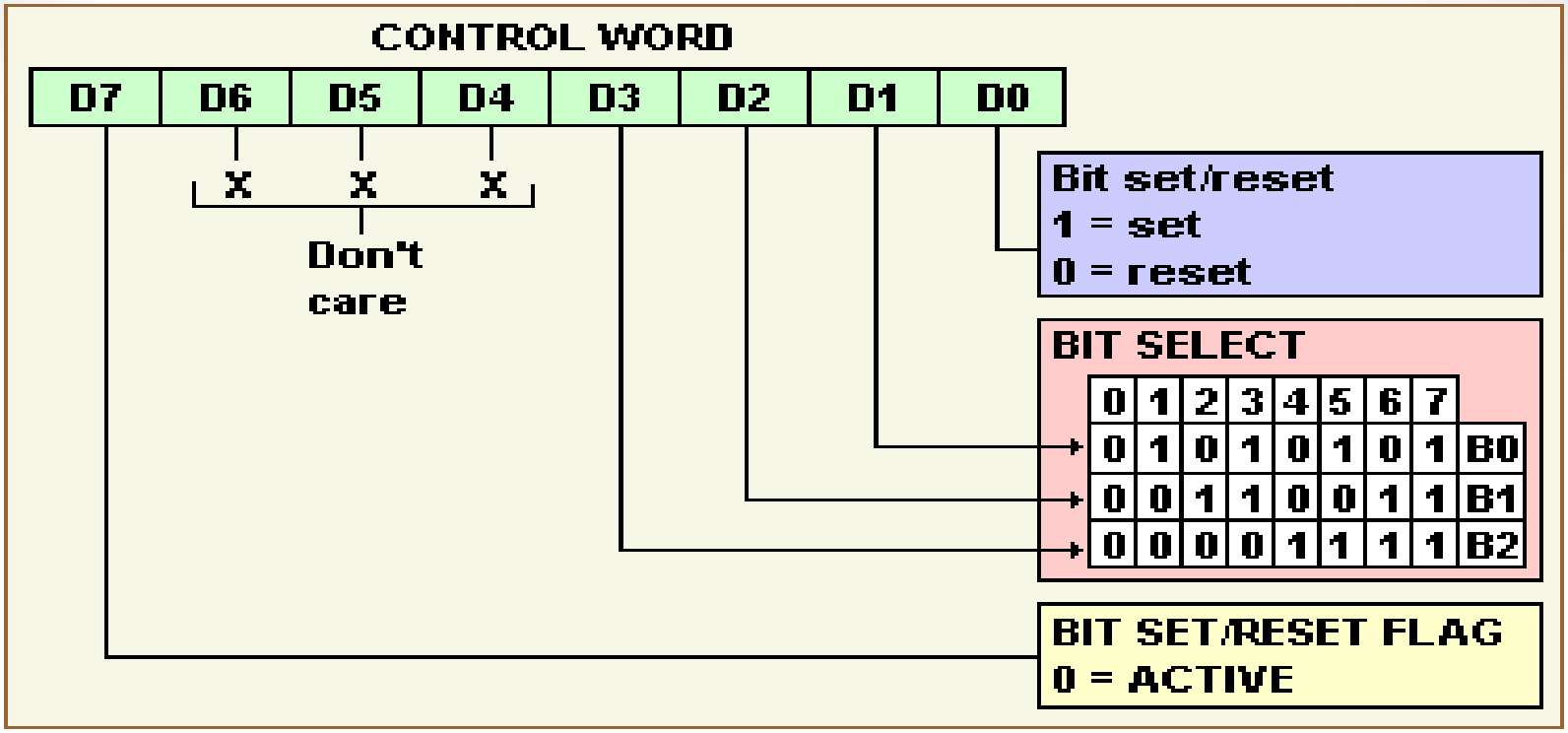
Control Word Register

In I/O mode if 7th bit is 1



The three modes are:

Mode 0: basic input / output
Mode 1: strobed input / output
Mode 2: bi-directional bus



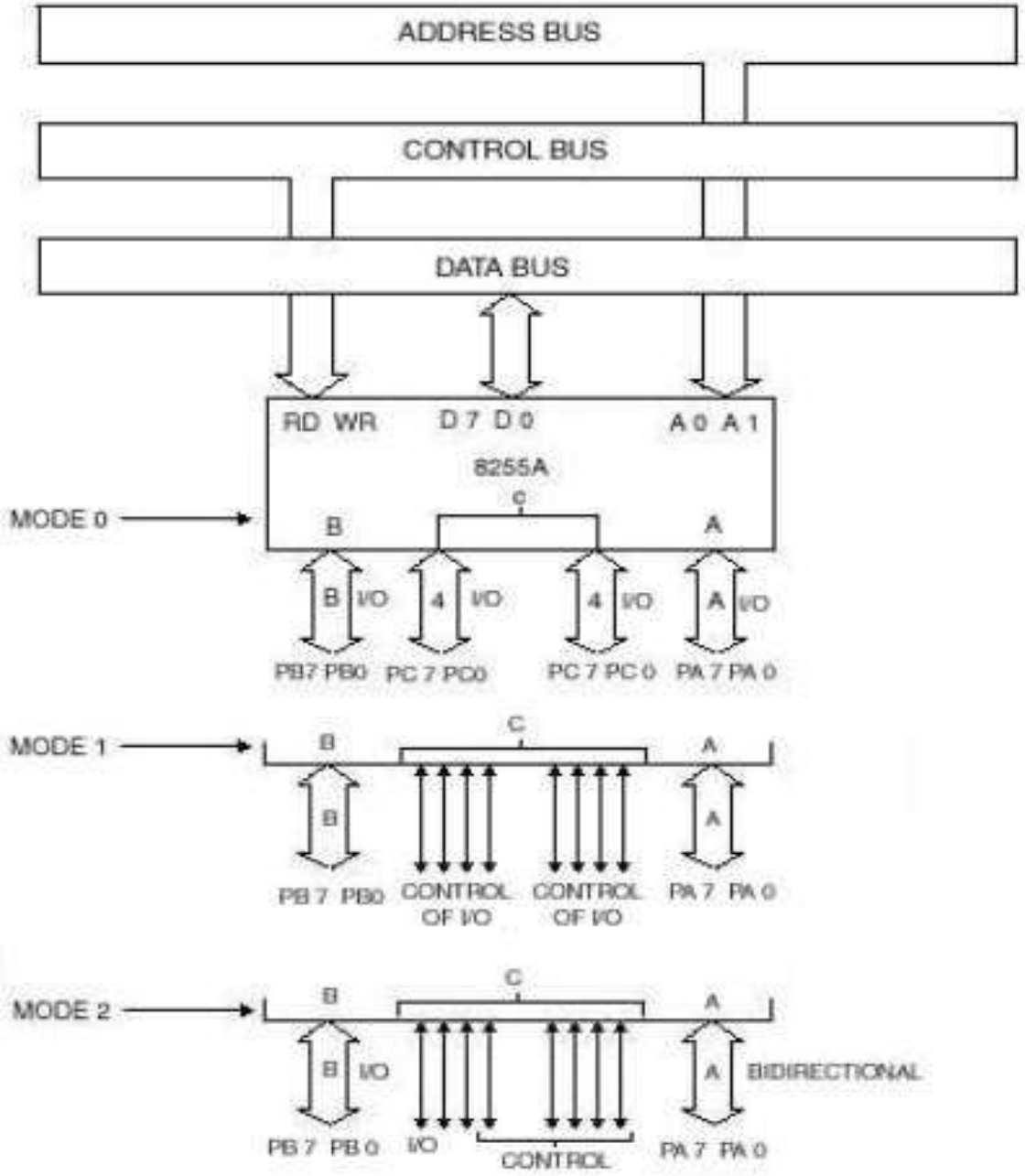
If the 7 th bit of CWR is 0 8255 initialize in BSR mode

Examples:

If you want to set/reset bit 0 of port C then set D3 to D1 to 000.

Bit 1 of port C will be set/reset if you code 001 to D3 to D1.

Bit 6 of port C is set/reset if D3 to D1 is 110.



Operating Modes Mode 0 (Basic Input/Output). This functional configuration provides simple input operations for each of the three ports. No “handshaking” is required data is simply written to or read from a specified port.

Mode O Basic Functional Definitions:

- Two 8-bit ports as port A and port B and two 4-bit port as PC low and PC up
- Any port can be input or output.
- If any port is programmed as output the out put are latched
- If any port is programmed as ouput the out put are latched
- 16 different Input/output configurations are not possible in this Mode.
- No facility for interrupt driven I/O

Operation

- CPU generates CS AO and A1 to select the the port for I/O operation according to the specified add ——
- Then CPU generates the RD/WR signal to read or write

MODE 1 (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or “handshaking” signals. In mode 1, port A and Port B use the lines on port C to generate or accept these “handshaking” signals.

Mode 1 Basic Functional Definitions:

- Two groups (Group A and Group B)

•Each group contains one 8-bit data port and one 4-bit control/data port

•The 8-bit data port can be either Inputs or output Both inputs and outputs are latched.

• The 4-bit port is used for control and status of the 8-bit data port.

Features of mode 1

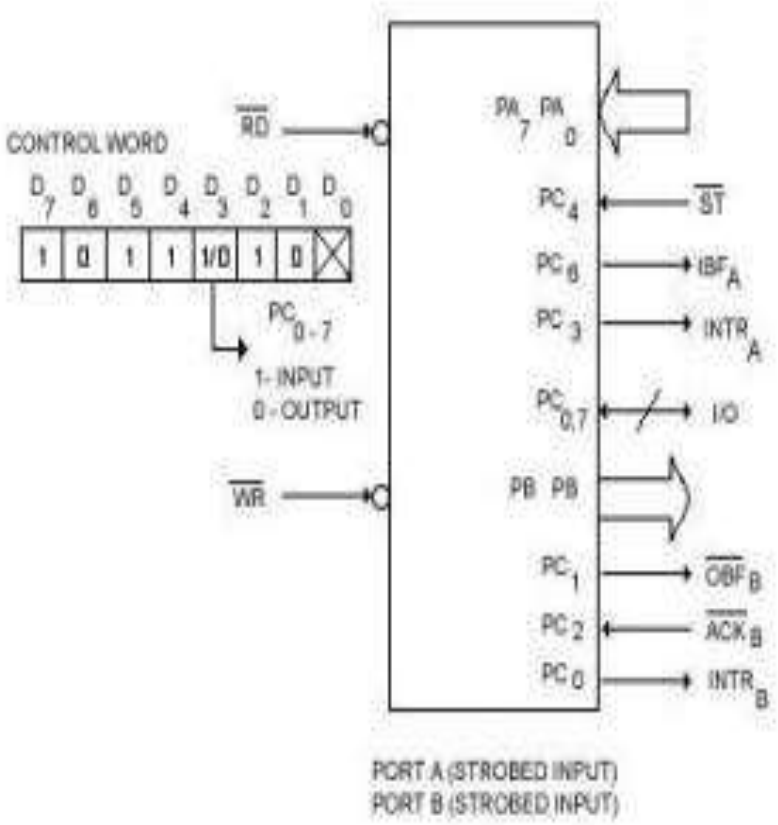
- Two 8 bit port A & B used as I/O port
- Each port use three lines from port c as hand shake signal
- Remaining line of port C can be used as I/o line
- The input and output data are latched
- Port C handshake signal can be used to interrupt the CPU it means interrupt logic is supported by 8255 to transfer data from 8255 to CPU

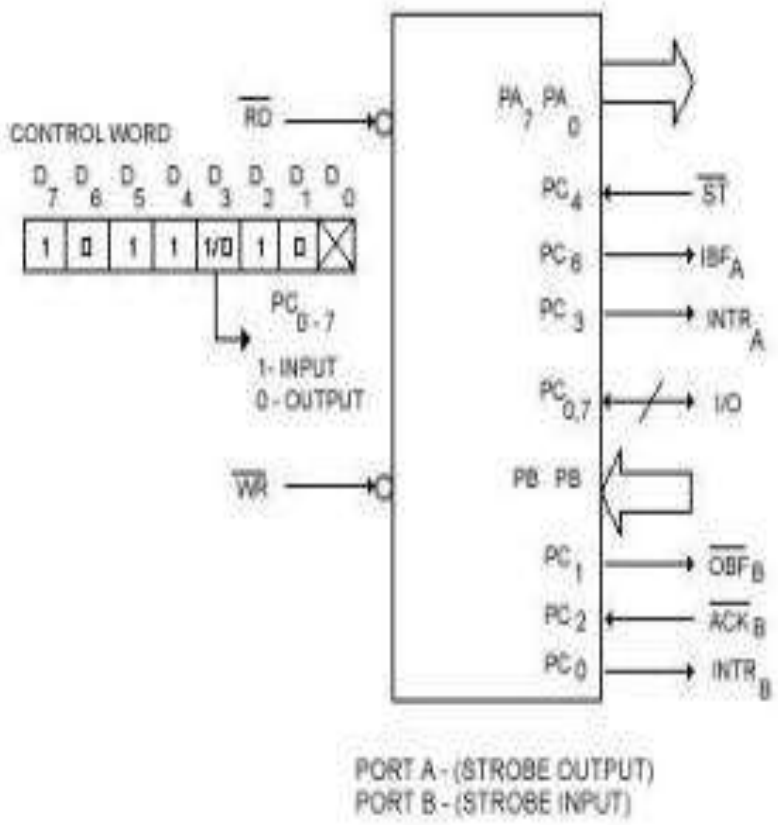
Operation

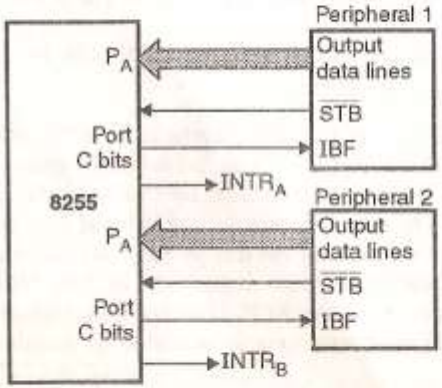
- STB generated by peripheral then 8255 generates IBF in response of STB
- INTR is generated by 8255 accordance with STB IBF and INTE

Combination of MODE 1

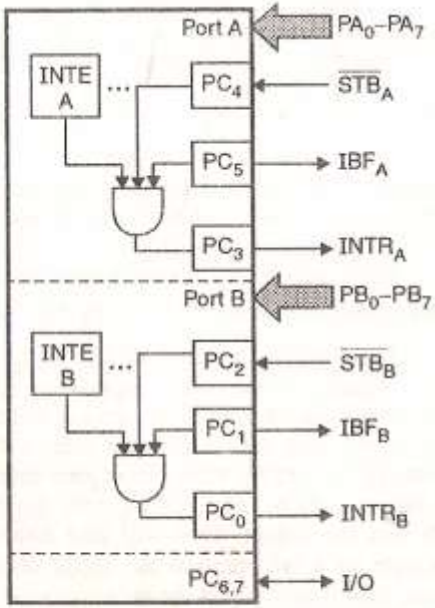
Port A and B can be Individually defined as Input or output in Mode 1 to support a wide varlety of strobed I/O application.







Mode 1 : Input mode interfacing
Fig. 9.25



P_A, P_B and P_C in mode 1 input mode
Fig. 9.26

Input output mode of mode 1

Timing diagram

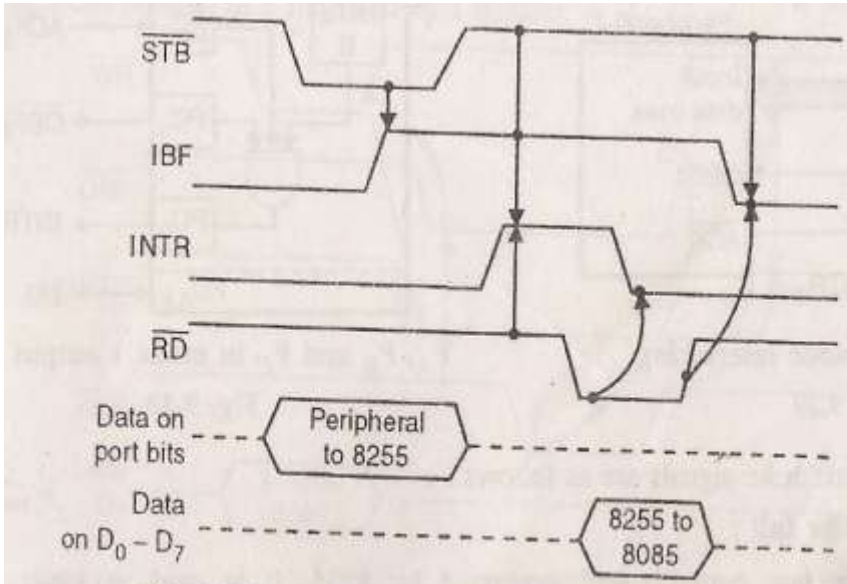


Fig. 9.27 : Timing diagram of mode 1 input mode

MODE 1 INPUT MODE OPRATION

STB : - it is generated by peripheral to indicate that it has transmitted data to input port

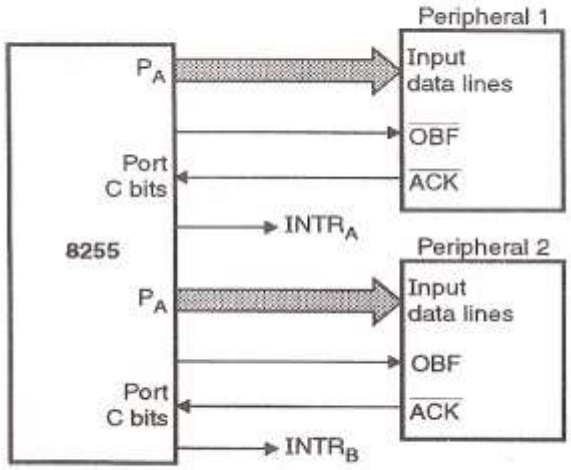
IBF : - this signal is generated by 8255 in response to STB

INTR : - this is active high signal given by 8255 to CPU

INTE : -if INTE is set interrupt will be generated depending upon

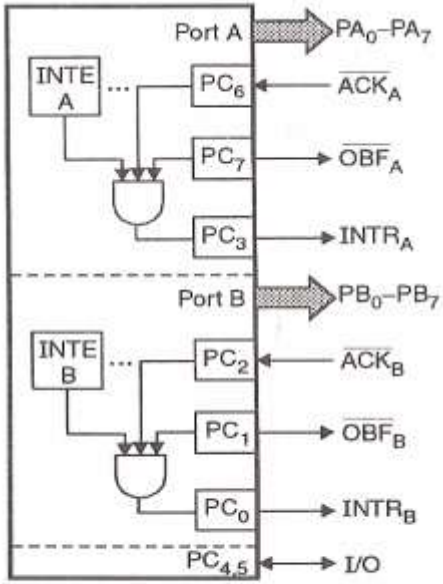
STB & IBF

INTR = INTE* STB*IBF



Mode 1 : Output mode interfacing

Fig. 9.29



P_A, P_B and P_C in mode 1 output mode

Fig. 9.30

MODE 1 OUT PUT MODE

•OBF: - it is active low signal generated by 8255 and used by peripheral as input when CPU wants to send the data on specified port

• ALK: -peripheral read the data and generate ALK in response of OBF

• INTR is generated by 8255 to interrupt CPU if INTE is set

•INTR = OBF*ALK*INTE = 1

Mode 2 (Strobed Bidirectional Bus I/O). This functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bi-directional bus I/O). "Handshaking" signals are provided to maintain proper bus flow discipline in a similar manner to MODE.

1. Interrupt generation and enable/disable functions are also available.

MODE 2 Basic Functional Definitions:

• Used in Group A only.

•One 8-bit, bi-directional bus Port (Port A) and a 5-bit control Port (Port C).

•Both Inputs and Outputs are latched.

•The 5-bit control port (Port C) is used for control and status for the 8-bit,bi-directional bus port (Port A).

MODE 2

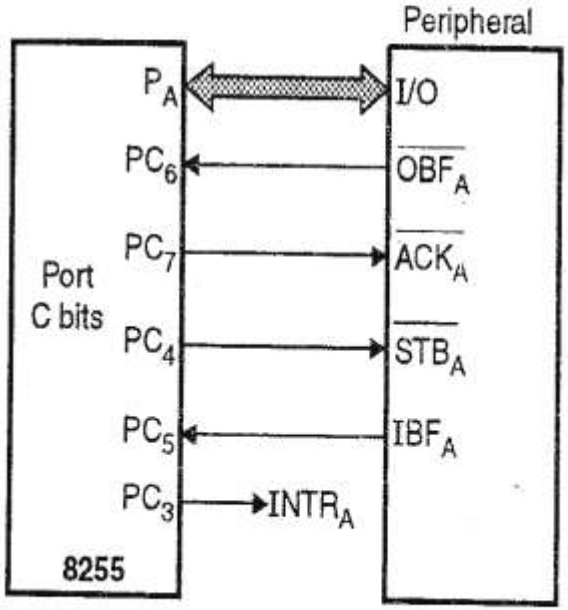


Fig. 9.33 : Mode 2 interfacing

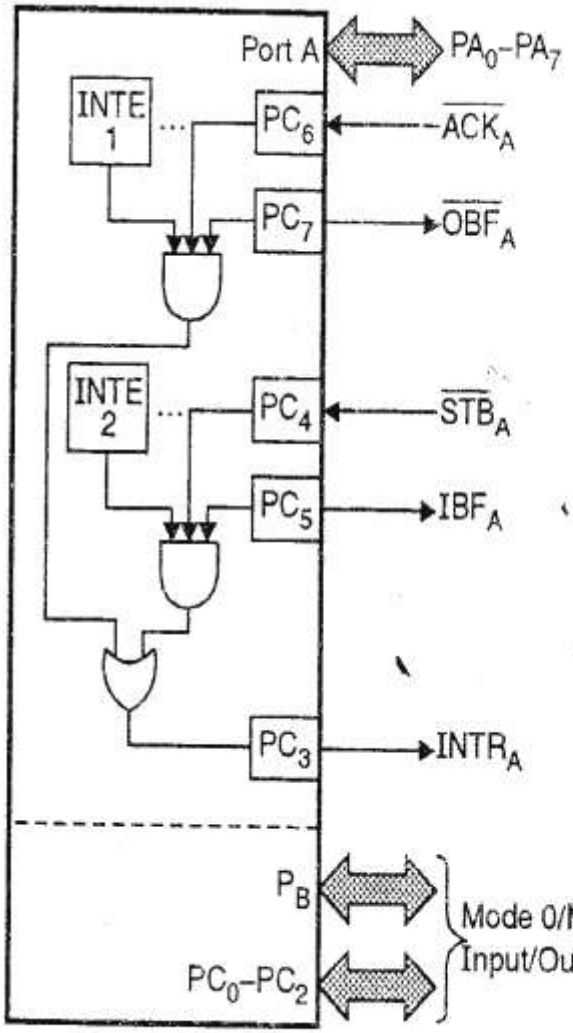


Fig. 9.34 : P_A, P_B & P_C in mode 2

8155 Multifunction Device

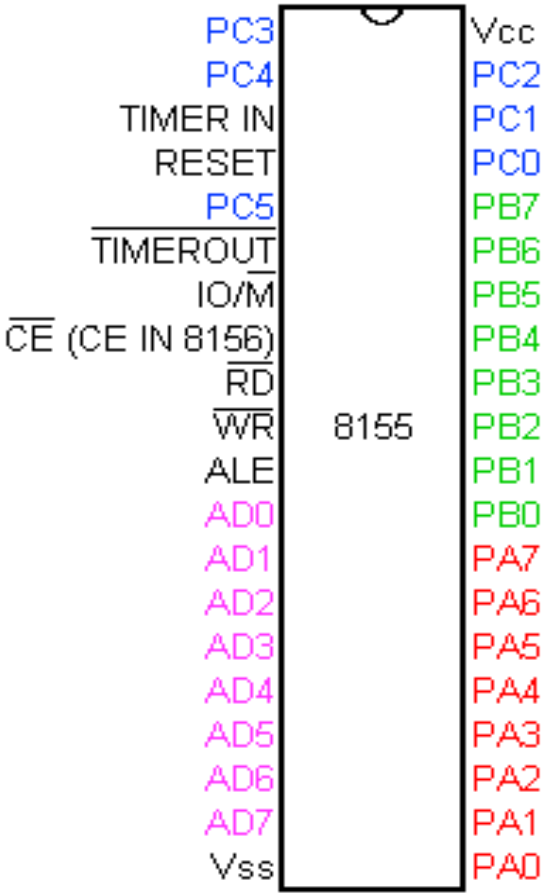
Features:

- 2kbits static RAM 256x8
- 2 programmable 8 bit I/O ports
- 1 programmable 6 bit I/O port
- 1 programmable 14 bit binary counter/timer
- Internal address latch to demux AD0-AD7, using ALE line

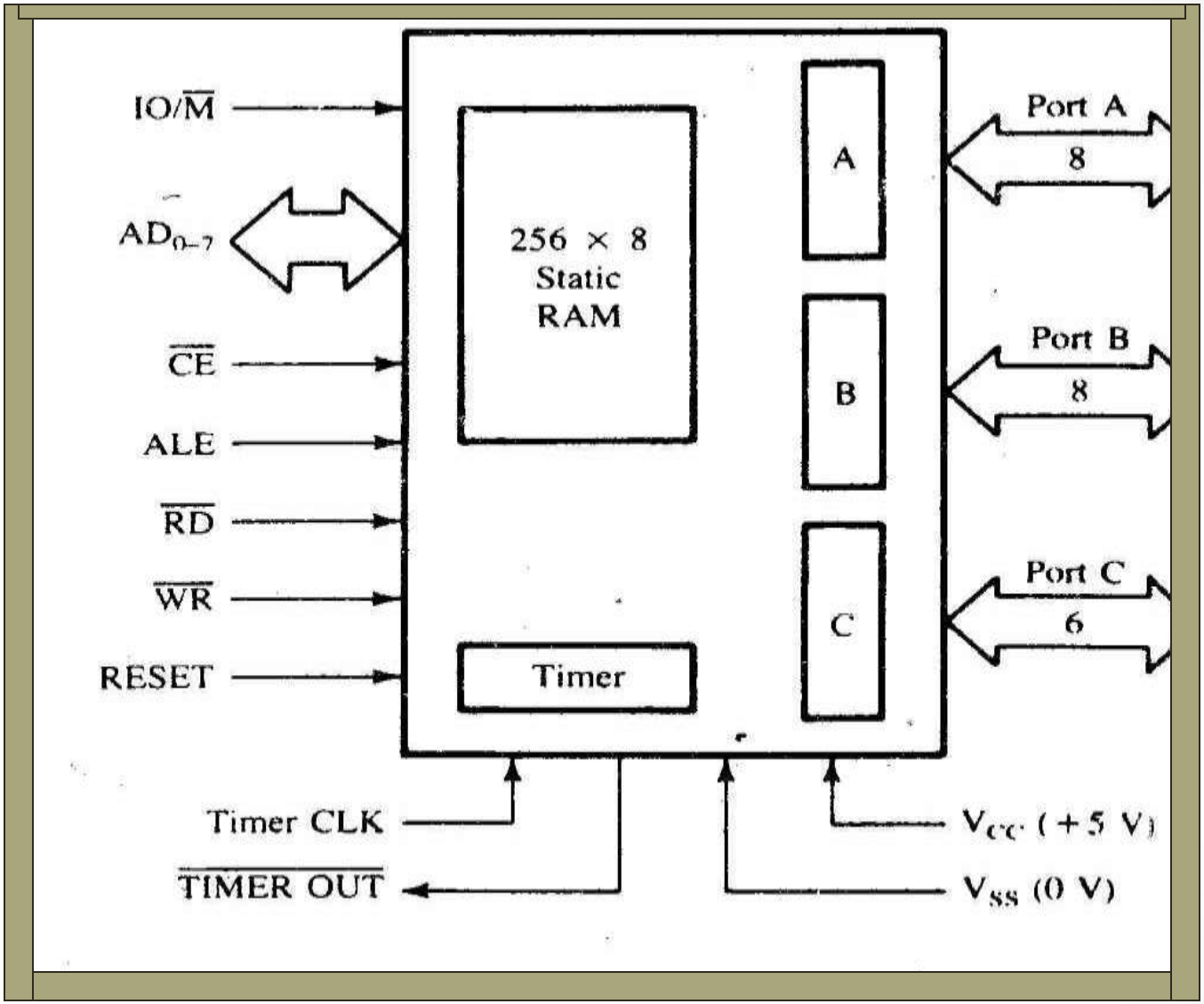
8155 Multipurpose Programmable Device

- 8155 is multi purpose programmable device ,consisting three I/O port and 256 kb memory and one timer
- Here it eliminate the need of external demultiplexing of AD0-AD7
- Multipurpose device 8355 include I/O port and one ROM where 8755 include a EPROM
- The control section of 8155 include 2 section
- Three I/O section of 8155 can be used as input or out put port where 2 I/O port are of 8 bit and one of the I/O port is of 6 bit
- 6 bit I/O port can be used as handshake signal for other two I/O port

Pin out : - 40 pins

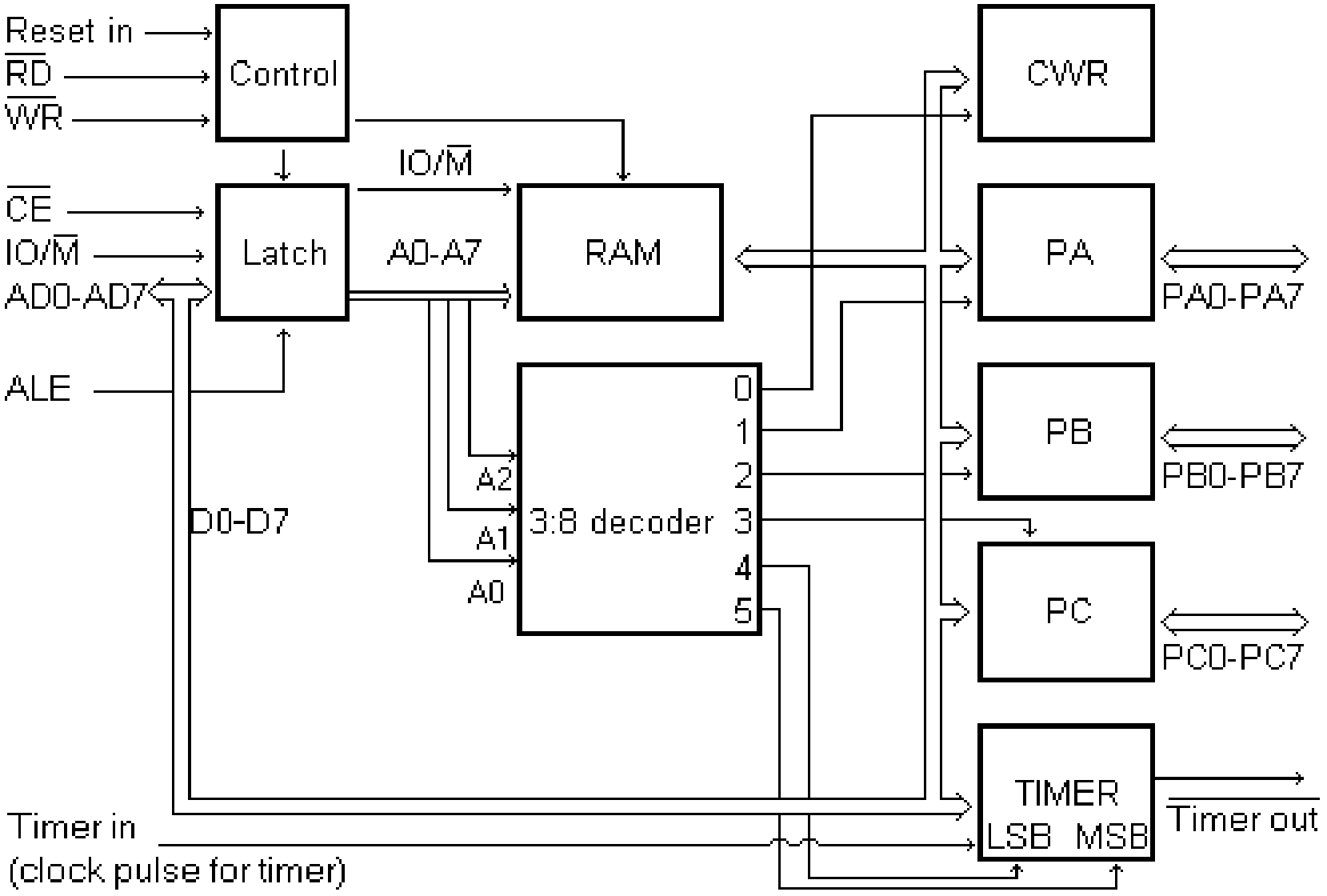


AD0-AD7	I/O	Addr/Data bus mux'd
RESET	I	Reset input
CE-bar or CE	I	Chip enable (55/56)
ALE	I	Address latch enable
RD-bar	I	Read input
WR-bar	I	Write input
IO/Mbar	I	I/O or memory section
PA0-7	I/O	Port A (8 bit)
PB0-7	I/O	Port B (8 bit)
PC0-5	I/O	Port C (6 bit)
TIMER-IN	I	Timer input
TIMER-OUT-bar	O	Timer output



Block diagram of 8155

Block diagram of 8155



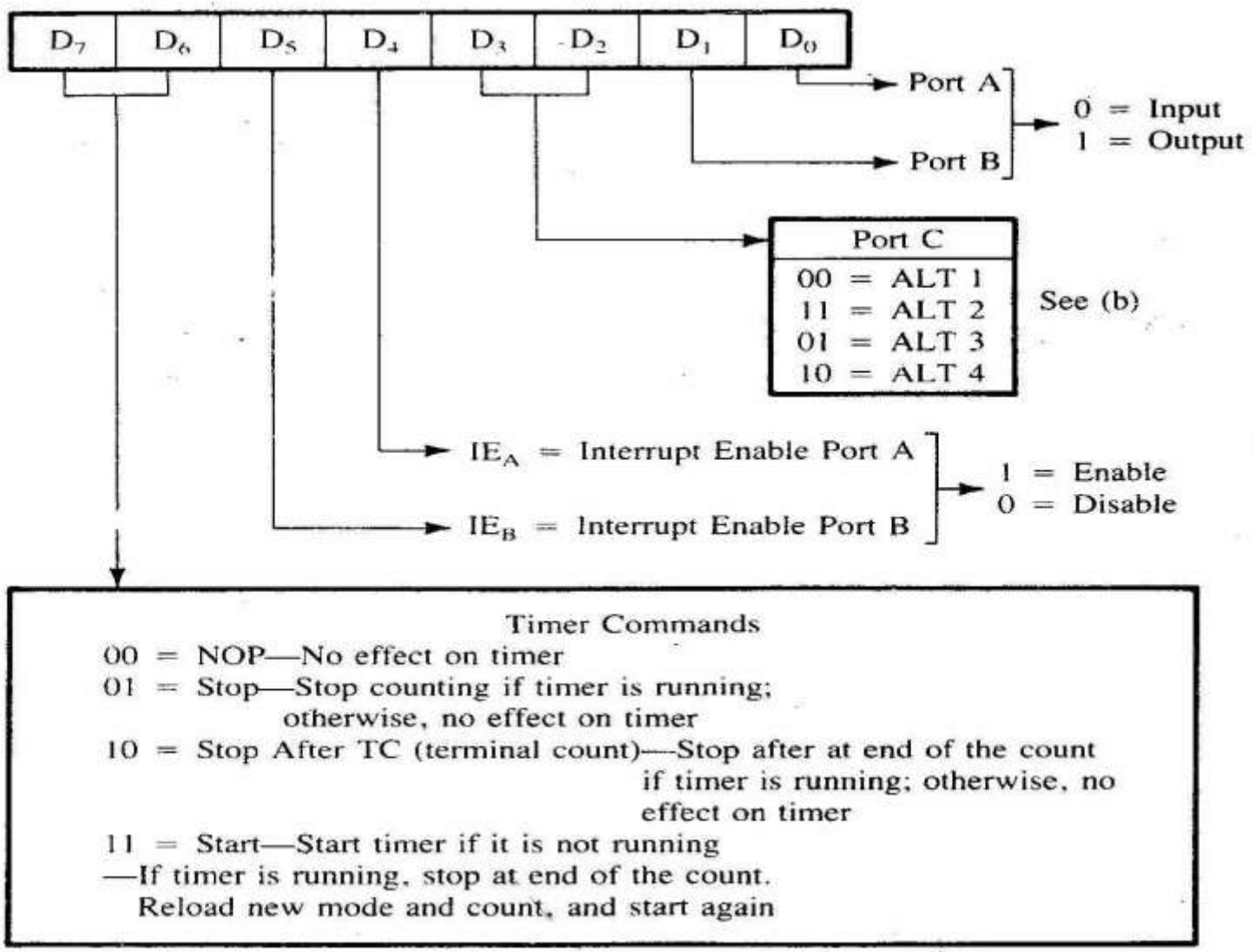
Registers:

A2	A1	A0	Port
0	0	0	Command/status reg.
0	0	1	PA
0	1	0	PB
0	1	1	PC
1	0	0	Timer LSB
1	0	1	Timer MSB

(ALE high, AD0=A0 etc)

(

CWR of 8155



Loading Format

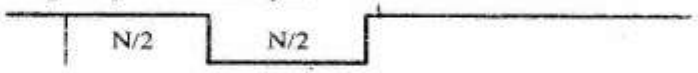
M_2	M_1	T_{13}	T_{12}	T_{11}	T_{10}	T_9	T_8

MSB

T_7	T_6	T_5	T_4	T_3	T_2	T_1	T_0

LSB

(a)

Modes	M_2 M_1	Timer Output
In this mode, the timer output remains high for half the count and goes low for the remaining count, thus providing a single square wave. The pulse width is determined by the count and the clock frequency.	0 0	Single Square Wave Cycle 
In this mode, the initial timer count is automatically reloaded at the end of each count, thus providing a continuous square wave.	0 1	Square Wave 
In this mode, a single clock pulse is provided at the end of the count.	1 0	Single Pulse Upon Terminal Count 
This is similar to Mode 2, except the initial count is reloaded to provide a continuous wave form.	1 1	Pulse Every Terminal Count 

N = Count in Timer

(b)

8253/8254 Programmable Interval Timer/Counter

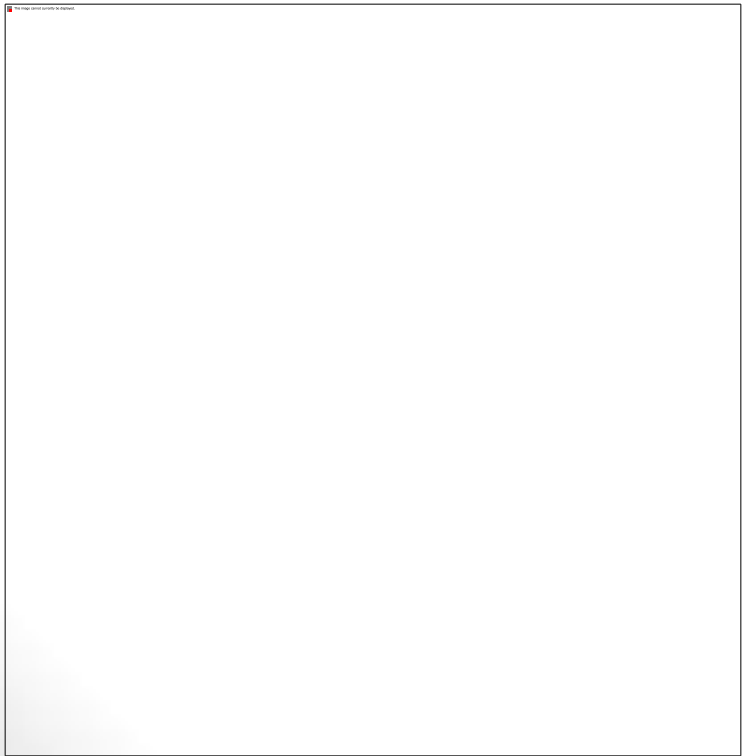
Features

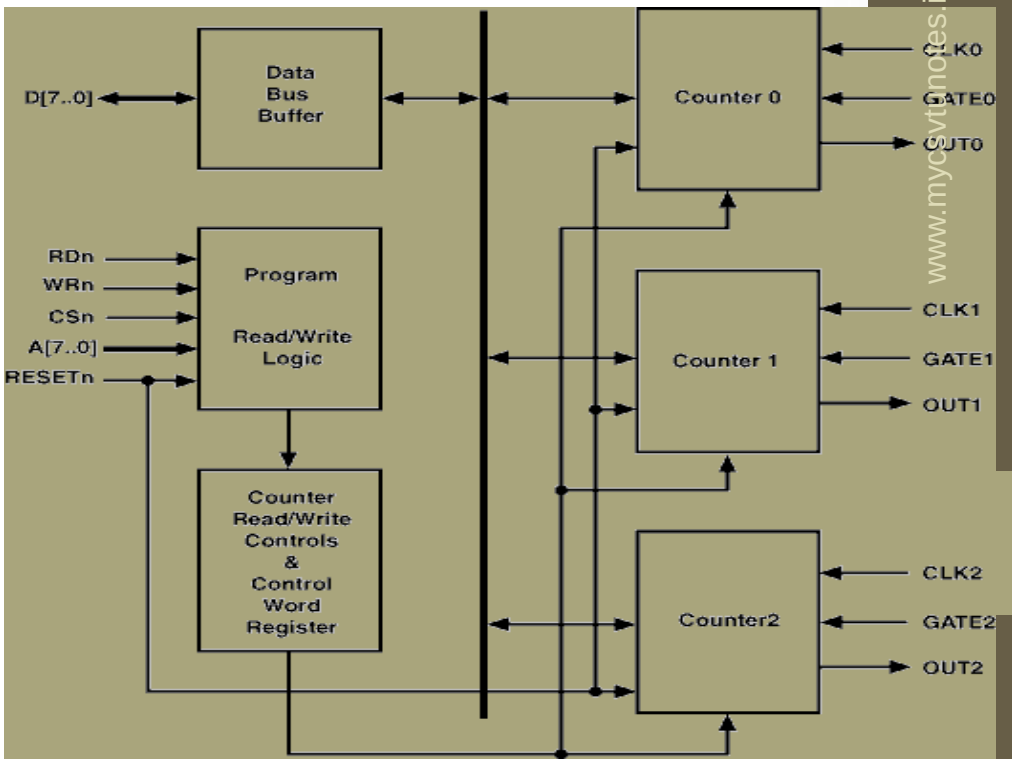
- Status read-back command
- Counter latch command
- Read/write least significant bit (LSB) only, most significant bit (MSB) only, or LSB first then MSB
- Six programmable counter modes
  - Interrupt on terminal count
  - Hardware retriggerable one-shot
  - Rate generator
  - Square wave mode
  - Software-triggered strobe
  - Hardware-triggered strobe (retriggerable)
- Binary or binary coded decimal strobe
- Developed in VHDL and synthesizes to approximately 5,000 gates

Functionally based on the Intel 82C54

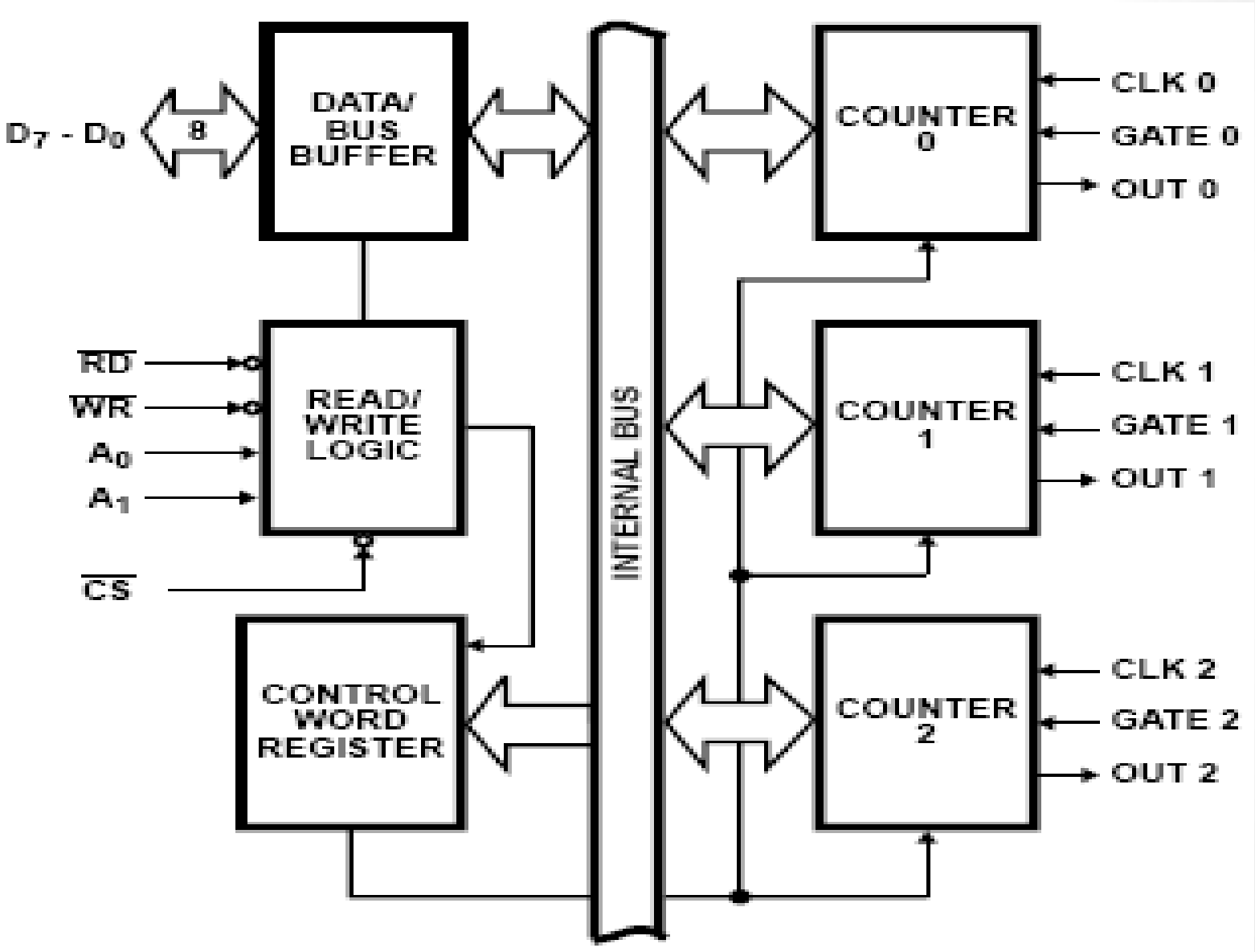
Description

The 8254 programmable interval time/counter megafunction is a high-performance function that is designed to solve the common timing control problems in microcomputer system design. It provides three independent 16-bit counters, and each counter may operate in a different mode. All modes are software programmable. The 8254 megafunction solves one of the most common problems in any microcomputer system: the generation of accurate time delays under software control. Instead of setting up timing loops in software, the 8254 megafunction can be programmed to match requirements by programming one of the counters for the desired delay





BLOCK DIAGRAM OF 8253/54



D0-D7: -these are * bit bi-directional data bus ,connect to system data bus

CS bar :- chip selection active low input to select 8253 chip

RD/WR:- bar input line to send or read the data though D0-D7

A0-A1:- address line to select the counter and CWR

CLK0,CLK1, CLK2:- thiese are clock input to 3 independent Counter,pulse applied at this pin is counted by respective counter

GATE0 to GATE2:- active high input pin to allow external hardware to control the respective counter the basic function is to start and end the counter

OUT0to OUT2:- active high output line used to give output of counter the output depends on selected mode

8253 contains following blocks

Data bus buffer : - it is used to interface 8253 with system data bus,it is bi-directional the direction is decided by RD/WR logic

Read/Write logic : -control signal used with A0 A1 and CS bar connected with IOR bar /IOW bar or MEMR/MEMW bar depending on mapping

Control Word Register :- it is selected when A0-1& A1-1

Counter : -3 independent 16 bit counter ,can be programmed to separately through control Word Register to decide mode of counter ,counter has thee input as CLK,GATE, and one output as OUT

Control Word Register

							D0
							BCD

SC1	SC0	Selection
0	0	Counter0
0	1	Counter1
1	0	Counter2
1	1	Read back

Mode of Counting	
0	Binary Counter
1	BCD Counter

RL1	RL0	Read/Load
0	0	Counter latching opration
0	1	Read/load LSB only
1	0	Read/load MSB only
1	1	Read/load LSB than MSB

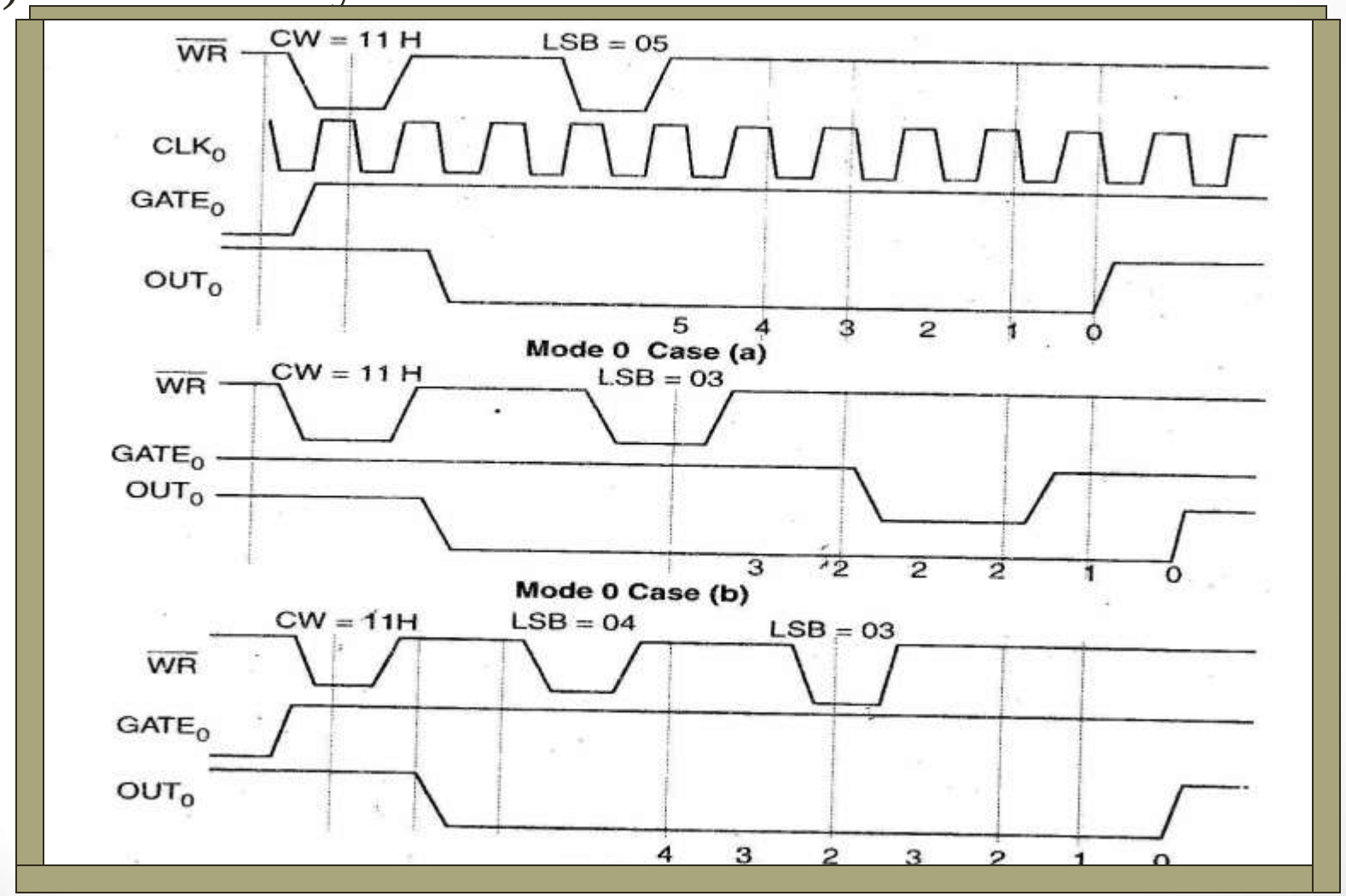
M2	M1	M0	Mode
0	0	0	MODE 0
0	0	1	MODE1
*	1	0	MODE2
*	1	1	MODE3
1	0	0	MODE4
1	0	1	MODE5

8253/8254 Modes of opration

- Interrupt on terminal count
- Programmable one-shot
- Rate generator
- Square wave mode
- Software-triggered strobe
- Hardware-triggered strobe

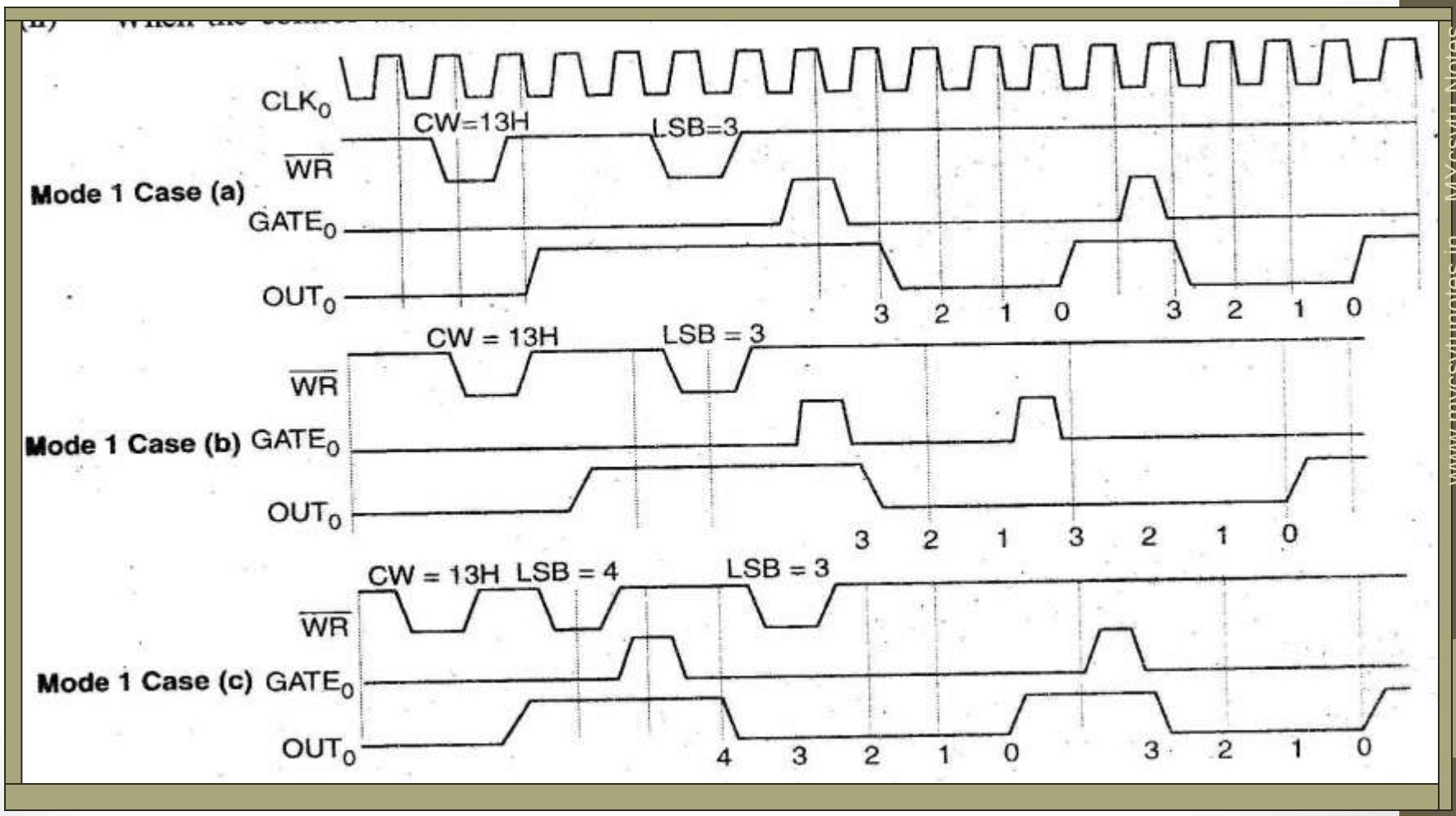
Mode 0 :- *Interrupt on terminal count*

- Here 2 WR bar signal is needed to load the CWR (CWR is loaded by 11H) and counter by count value



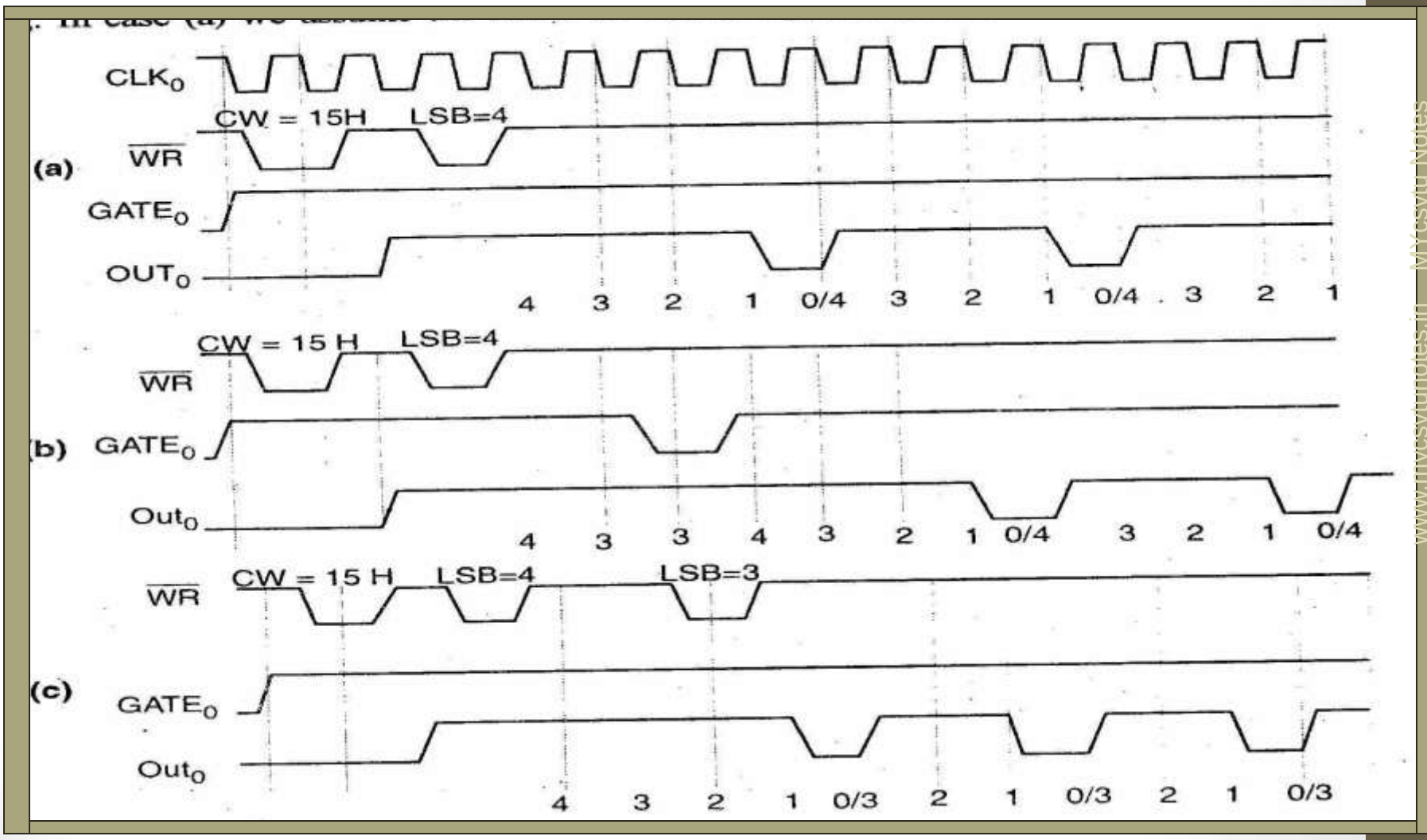
MODE 1 : -***Programmable one shot***

• CWR is loaded by 13 H to initialize 8253in mode 1 once CWR loaded OUT output goes low



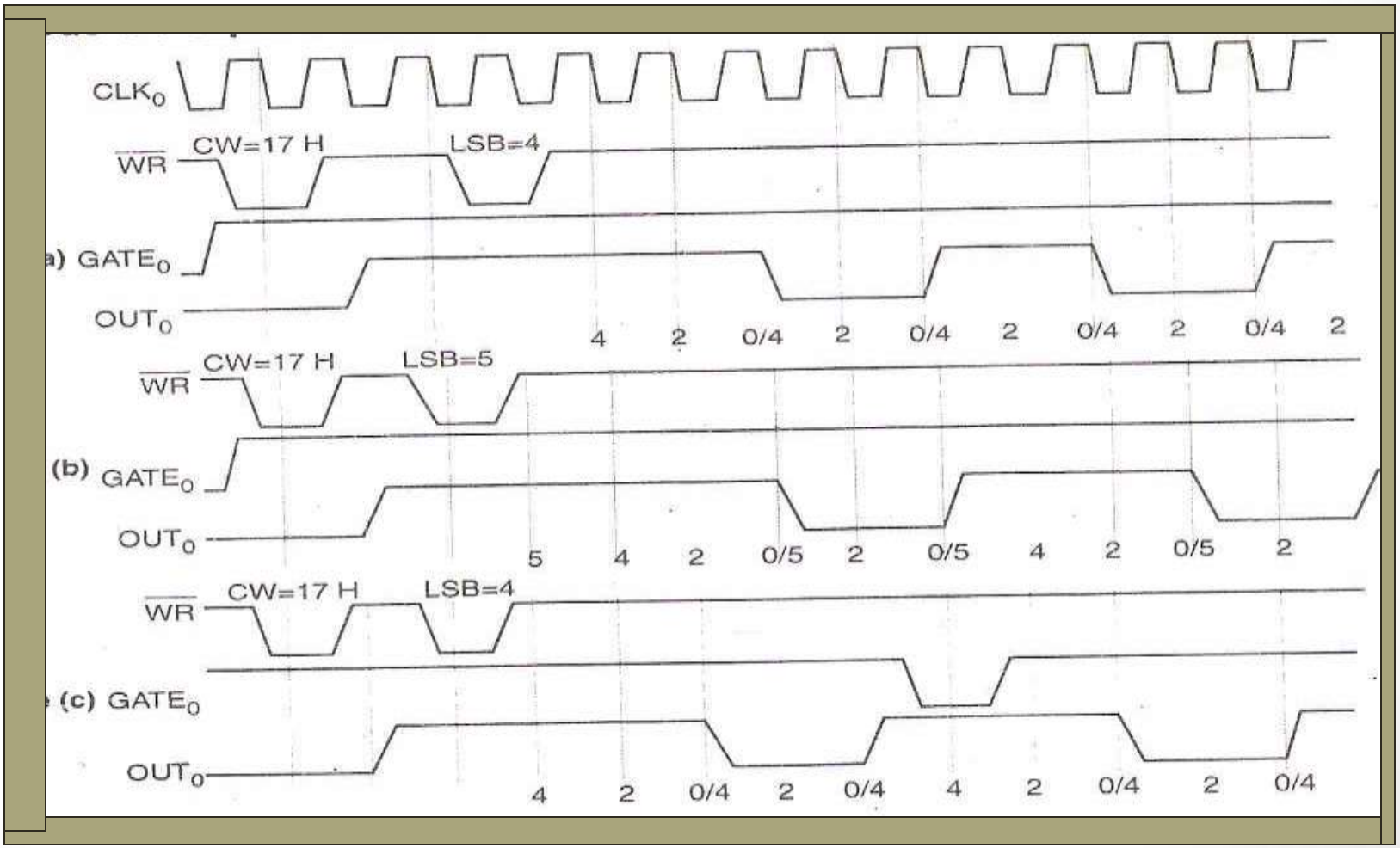
MODE 2 : - Rate Generator/Pulse Generator

- CWR is loaded by 15 H ,once the CWR is loaded OUT output goes high



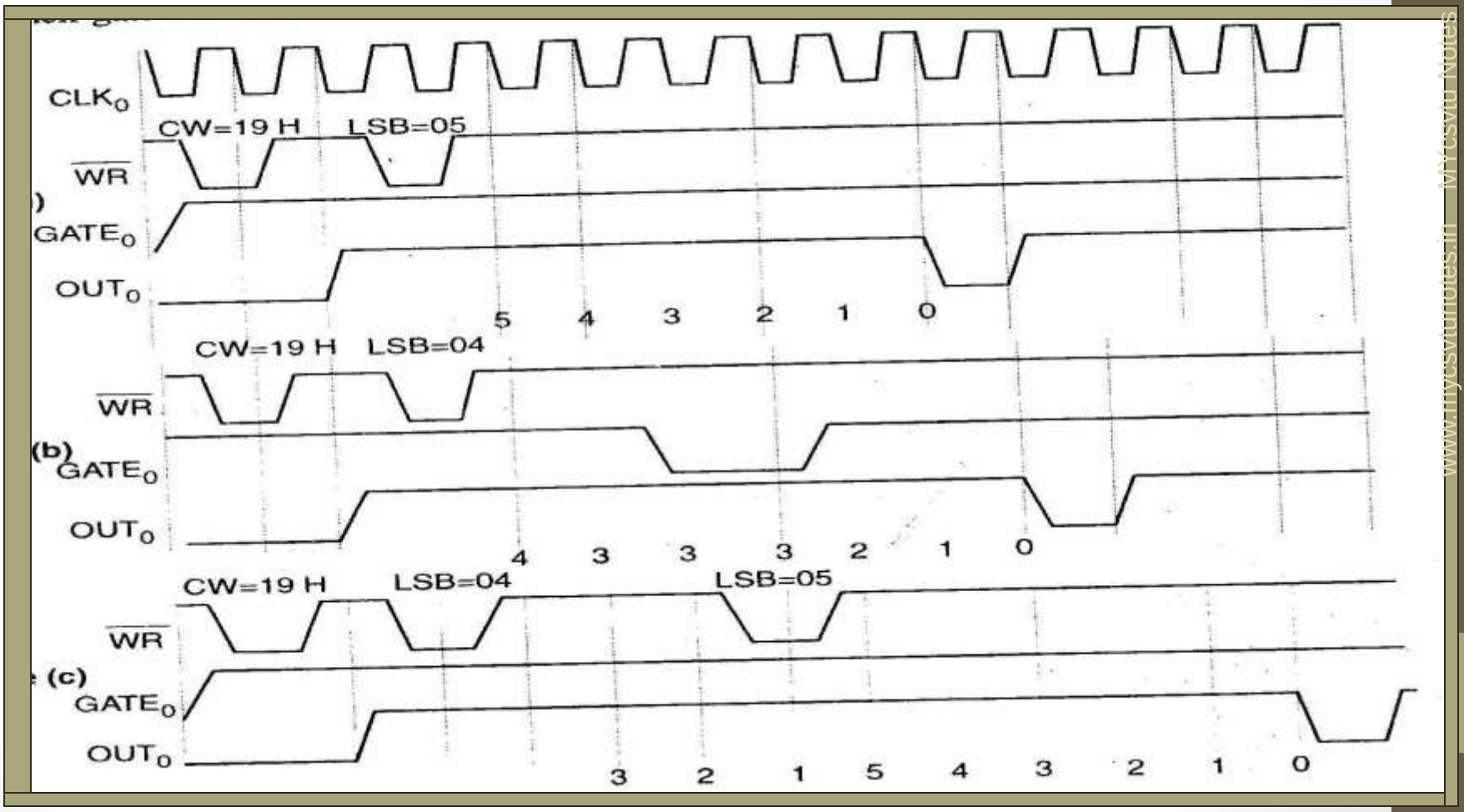
Mode 3 : -Square Wave Generator

- CWR is loaded by 17 H ,once the CWR is loaded OUT output goes high



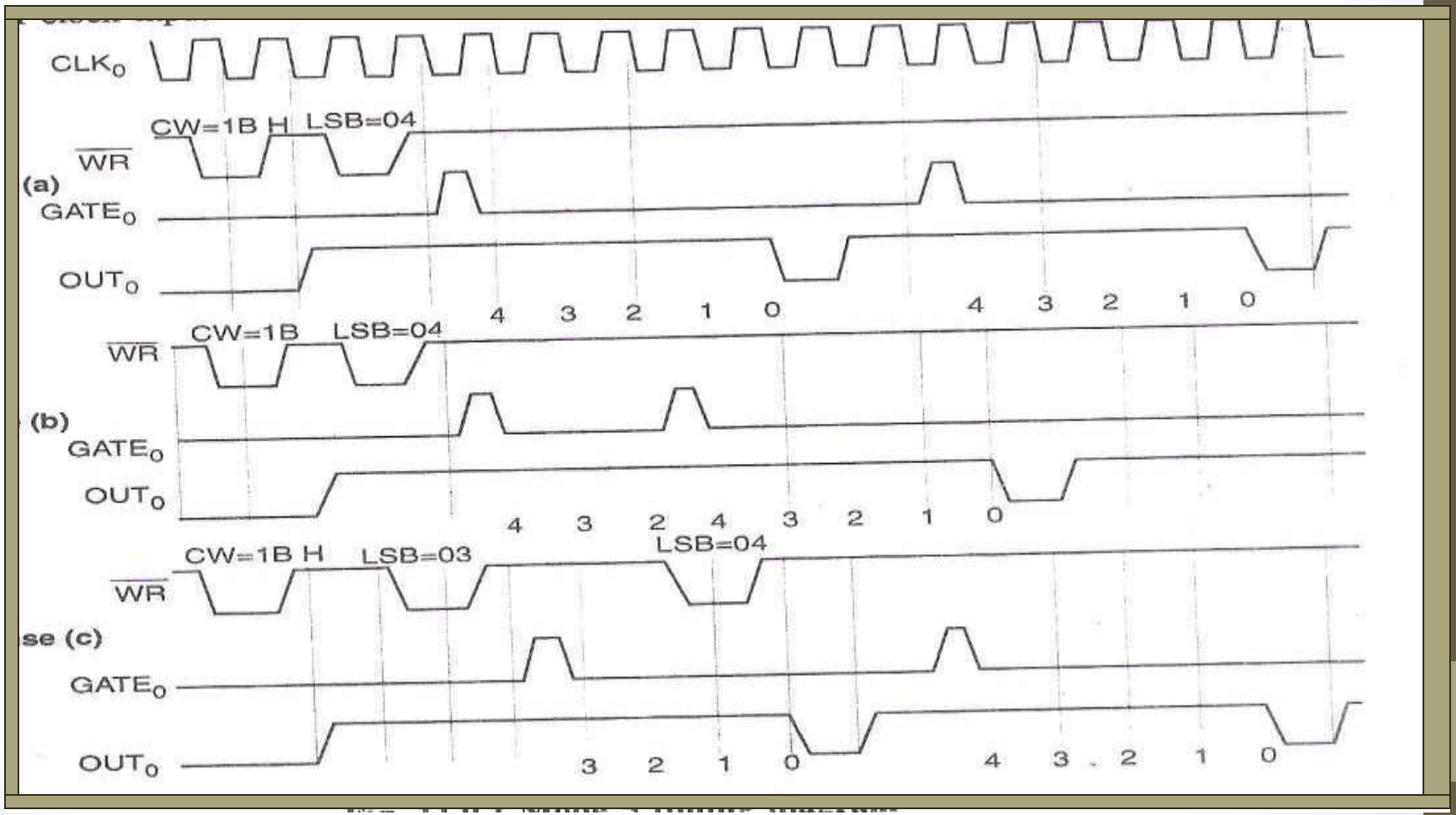
Mode 4 : -Software Triggered Strobe

- CWR is loaded by 19 H ,once the CWR is loaded OUT output goes high



Mode 5 : -hardware Triggered Strobe

- CWR is loaded by 1B H ,once the CWR is loaded OUT output goes high

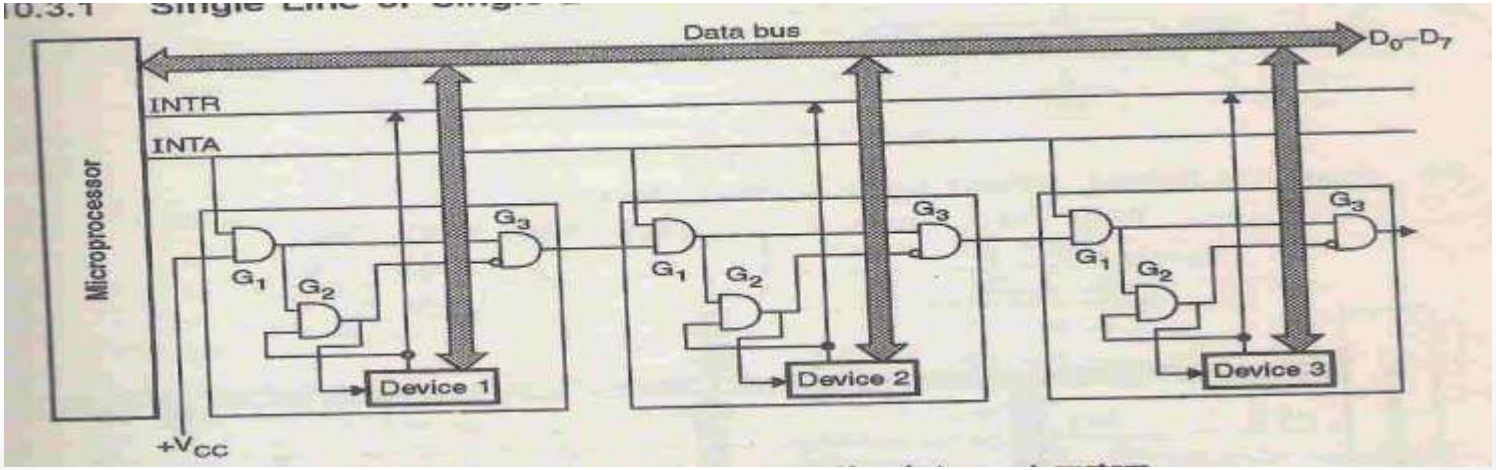


Interrupt system

- Single level interrupt system
- Multiple level interrupt System

Single level Interrupt System

1. CPU provide only one interrupt request line
2. MP ignore all other interrupt system during execution of any interrupt service routine
3. Here the problem to identify the and giving the priority to interrupting device can be solve using software and hardware polling



Multiple level Interrupt system

1. Here the interrupt control logic provide multiple interrupt line and pass one of them as per the priority of interrupting device
2. MP can accept other interrupt during execution of any interrupt service routine

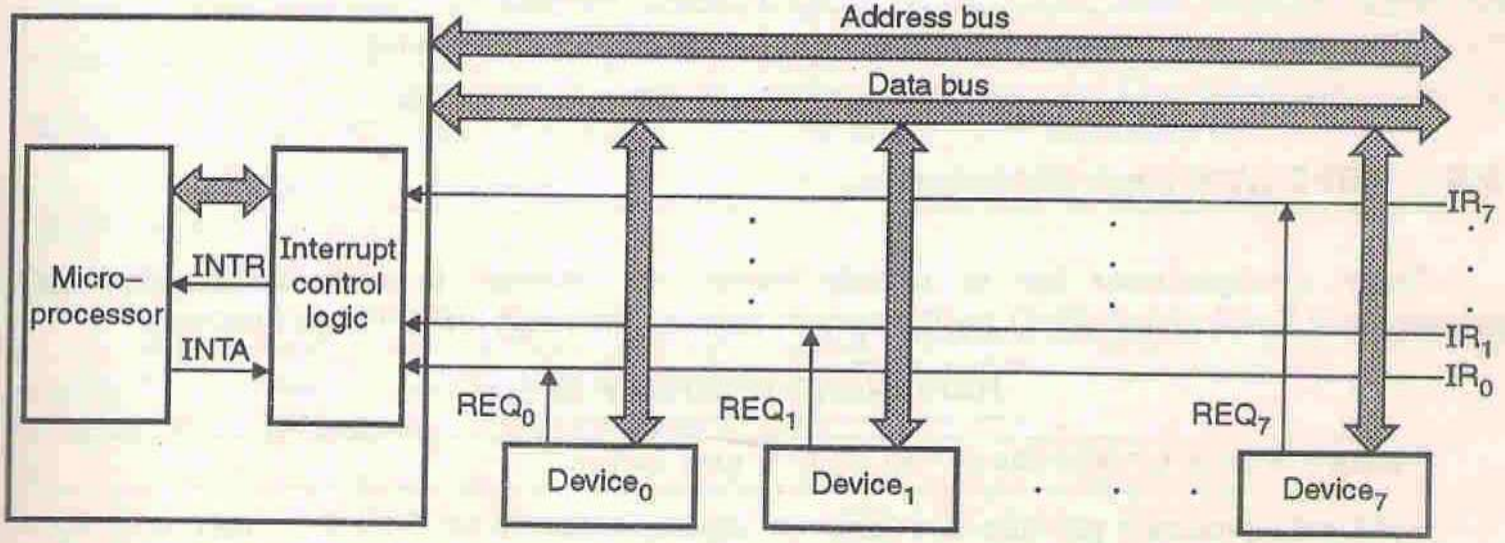


Fig. 10.6 : Multiline or multilevel interrupt system

INTERRUPTS

MP 8085 interrupt structure

Hardware interrupt

1.trap 2. RST7.5 3. RST6.5, RST5.5 INTR

Software interrupt

RST 0 to RST 7

TRAP : -

•It is nonmaskable edged and level triggered request

•It is used for emergency purpose

•MP doesn’t execute any INTA cycle to read the interrupt information

•MP execute idle machine cycle to acknowledge this interrupt during this cycle RST 4.5 is executed and bring the control on 0024 H

•It has highest priority among all interrupt

•**RST 7.5**

•It is maskable edged triggered interrupt requested line

•MP doesn’t execute any INTA cycle to read the interrupt information

•MP execute idle machine cycle to acknowledge this interrupt during this cycle RST 7.5 is executed and bring the control on 003c H

•It has highest priority among all maskable interrupt

•Disabled by SIM and DI

RST 6.5 & RST 5.5

•This are level triggered interrupt

•MP doesn’t execute any INTA cycle to read the interrupt information

•MP execute idle machine cycle to acknowledge this interrupt during this cycle RST 6.5/ RST 5.5 is executed and bring the control on 0034/002C H

•It is maskable interrupt

INTR

•It is level triggerd interrupt **request line**

•MP execute INTA cycle to read the interrupt information from interrupting device

•Starting address depends on interrupt information

•Not affected by SIM

•Enable and disable by EI & DI

Software interrupt

•It is software generated interrupt

•It can’t be maskable

•Has highest priority among all

•Here jump instruction must be stored in to corresponding location to transfer MP controle to user defined ISR addr

- Trap RST 7.5,6.5,5.5 take only one M/C to acknowledge request
- Response time is very low
- Nos of memory location for each ISR is very low hence JUMP inst must be use to transfer the MP's control from predetermined location to user defined location

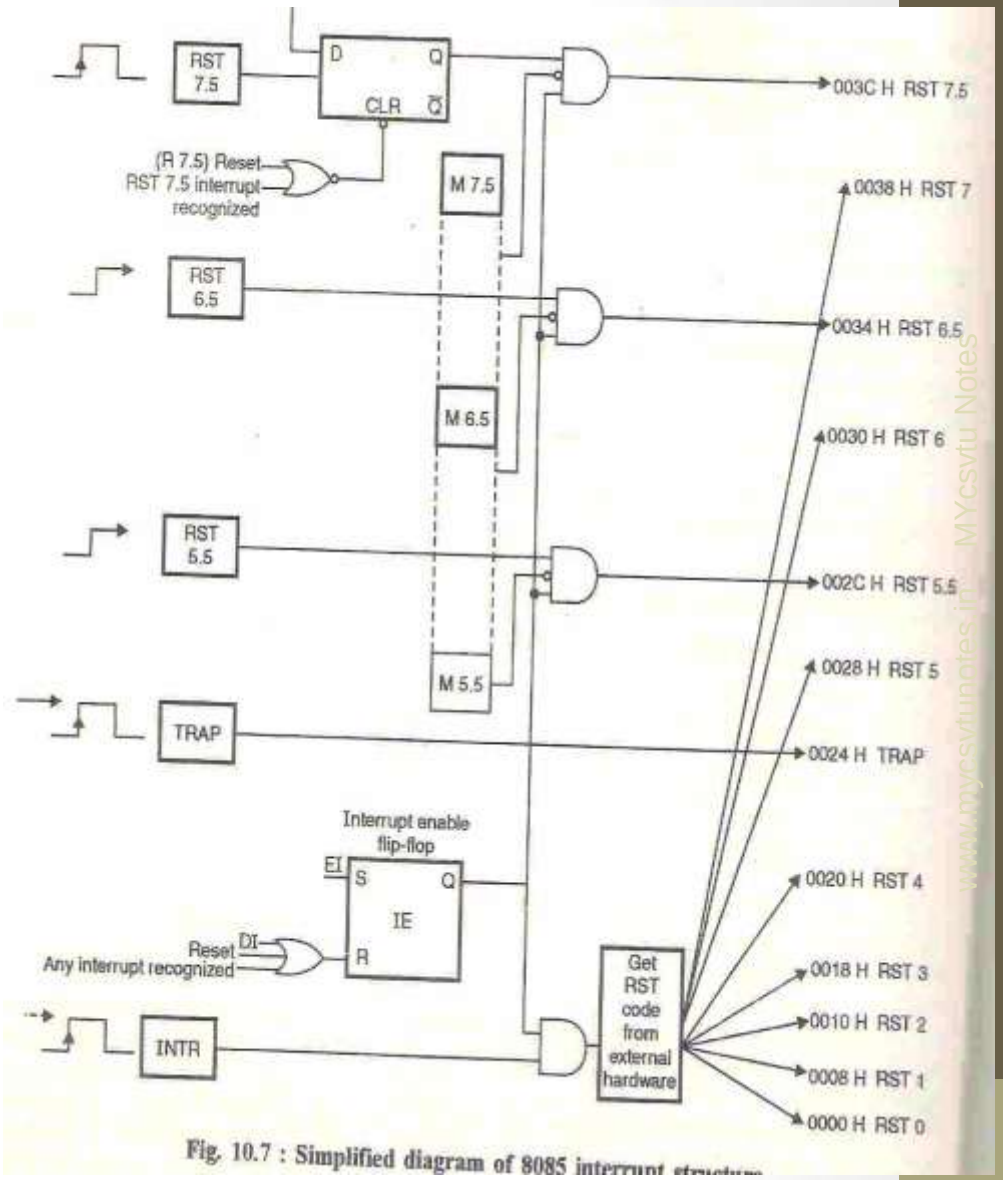


Fig. 10.7 : Simplified diagram of 8085 interrupt structure

Interrupt Structure

, for RST N, instruction is predefined. Following table depicts the same :

Instruction	Opcode	Address of ISR
RST 0	11000111 = C7	0000H (8 × 0) = 0000H
RST 1	11001111 = CF	0008H (8 × 1) = 0008H
RST 2	11010111 = D7	0010H (8 × 2) = 0010H
RST 3	11011111 = DF	0018H (8 × 3) = 0018H
RST 4	11100111 = E7	0020H (8 × 4) = 0020H
RST 5	11101111 = EF	0028H (8 × 5) = 0028H
RST 6	11110111 = F7	0030H (8 × 6) = 0030H
RST 7	11111111 = FF	0038H (8 × 7) = 0038H

e difference between two successive locations is only 8 bytes. Hence jump inst

Difference between two successive location is 8 byte hence if the ISR is greater then 8byte so JUMP instruction must be used to stored in corresponding location to transfer MP's control corresponding location

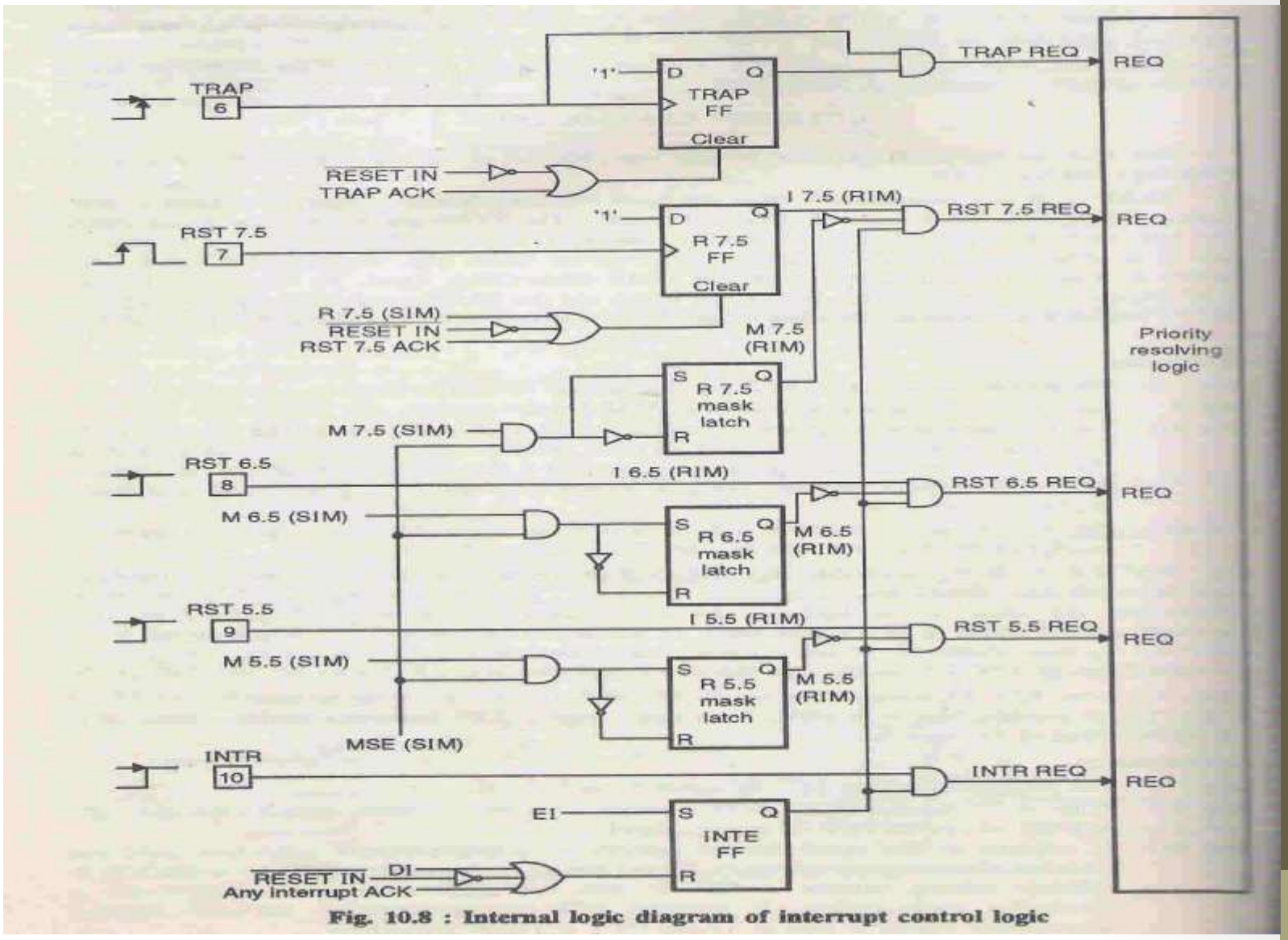


Fig. 10.8 : Internal logic diagram of interrupt control logic

Internal logic control of interrupt control logic

EI –ENABLE INTERRUPT

•It is used to enable all the maskable interrupt

•It sets an INTE f/f of interrupt control logic

•It is single byte instruction takes only one M/c

•It is also used to enable maskable interrupt during execution of ISR

•It does not effect on trap\

DI- DISABLE INTERRUPT

• It is used to disable all maskable interrupt

•It reset an INTE f/f of interrupt control logic

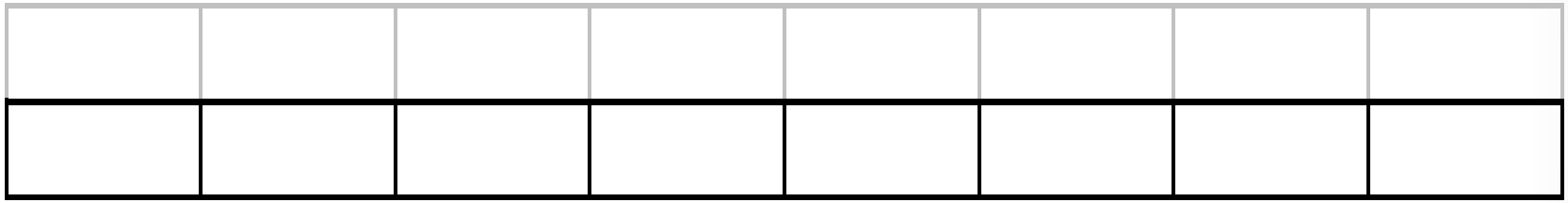
•It is single byte instruction takes only one M/c

•It is also used to prevent a critical part of program from maskable interrupt

•It does not effect on trap

SIM :SET INTERRUPT MASK

- It is one byte instruction used to enable all maskable interrupt
- It does not affect on TRAP and INTR
- It is also used in serial data transmission
- It transfer SIM format from accumulator to control logic
- It also transfer serial data bit D7 to the SOD pin



SOD: - serial output data

SDE : - serial data enable (1 enable , 0 disable)

R 7.5: - reset R 7.5 F/F (1 reset F/F , 0 no effect on F/F)

M7.5,M6.5,M5.5: - mask RST (1 mask or disable ,0 unmask)

MSE: - mask set enable (1 -d2,d1,d0 bit are effective 0 – bits are ignored)

RIM : READ INTERRUPT MASK

- It is single byte instruction used to check the status of all maskable interrupt
- It also transfer serial data bit from SID line to D7 bit of accumulator
- It does not provide status of trap & INTR
- This instruction transfer the content of interrupt control logic to accumulator
- It is used to check status of pending interrupt

							0
							5

SID:-serial data bit

I7.5,I6.5,I5.5:-RST bit is pending(1 pending ,0 not effected)

IE:-Interrupt enable (1 INTE F/F is SET , 0 F/F reset)

M7.5,M6.5,M5.5:-Mask RST (1 Masked or disable , 0 unmasked)

INTERRUPT CONTROLE LOGIC OPERATIONS

Trap

•PPI generate trap to set trap F/F

•In response to this MP completes current instruction and execute ideal cycle during this cycle MP calculate starting addr of ISR 0024H

•Than MP reset trap and INTE F/F and MP execute 2 M/C to store PC content to stack memory

RST 7.5

•PPI activate RST 7.5 to set RST 7.5 F/F

•If R 7.5 ids set and INTE F/F set and mask 7.5 F/F reset logic activates an RST 7.5 request

•In response to this MP completes current instruction and execute ideal cycle during this cycle MP calculate starting addr of ISR 003CH

•MP execute 2 M/C to store PC content to stack memory

RST 6.5 AND RST 5.5 Same as RST 7.5

INTR : -

• the PPI activates INTR signal

•In response to this INTE F/F is set

•MP completes current instruction cycle and execute one interrupt acknowledgement cycle (INTA)for call instruction from external hardware

8259 Programmable Interrupt controller

Features:

•8 levels of interrupts.

•Can be cascaded in master-slave configuration to handle 64 levels of interrupts.

•Internal priority resolve.

•Fixed priority mode and rotating priority mode.

•Individually maskable interrupts.

•Modes and masks can be changed dynamically.

•Accepts IRQ, determines priority, checks whether incoming priority > current level being serviced, issues interrupt signal.

•In 8085 mode, provides 3 byte CALL instruction. In 8086 mode, provides 8 bit vector number.

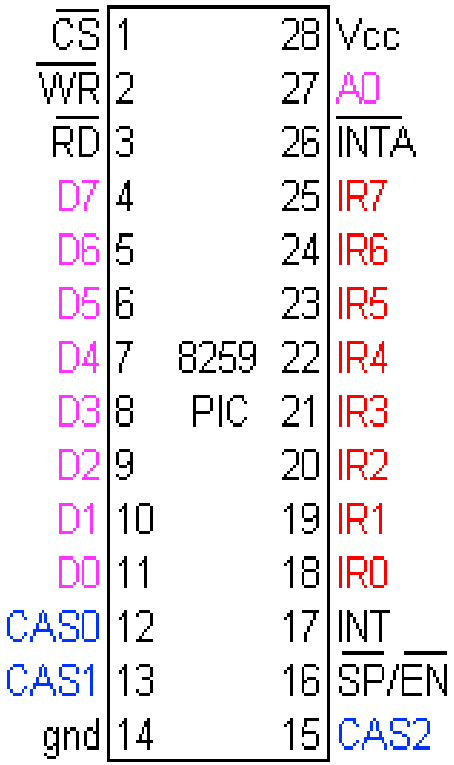
•Polled and vectored mode.

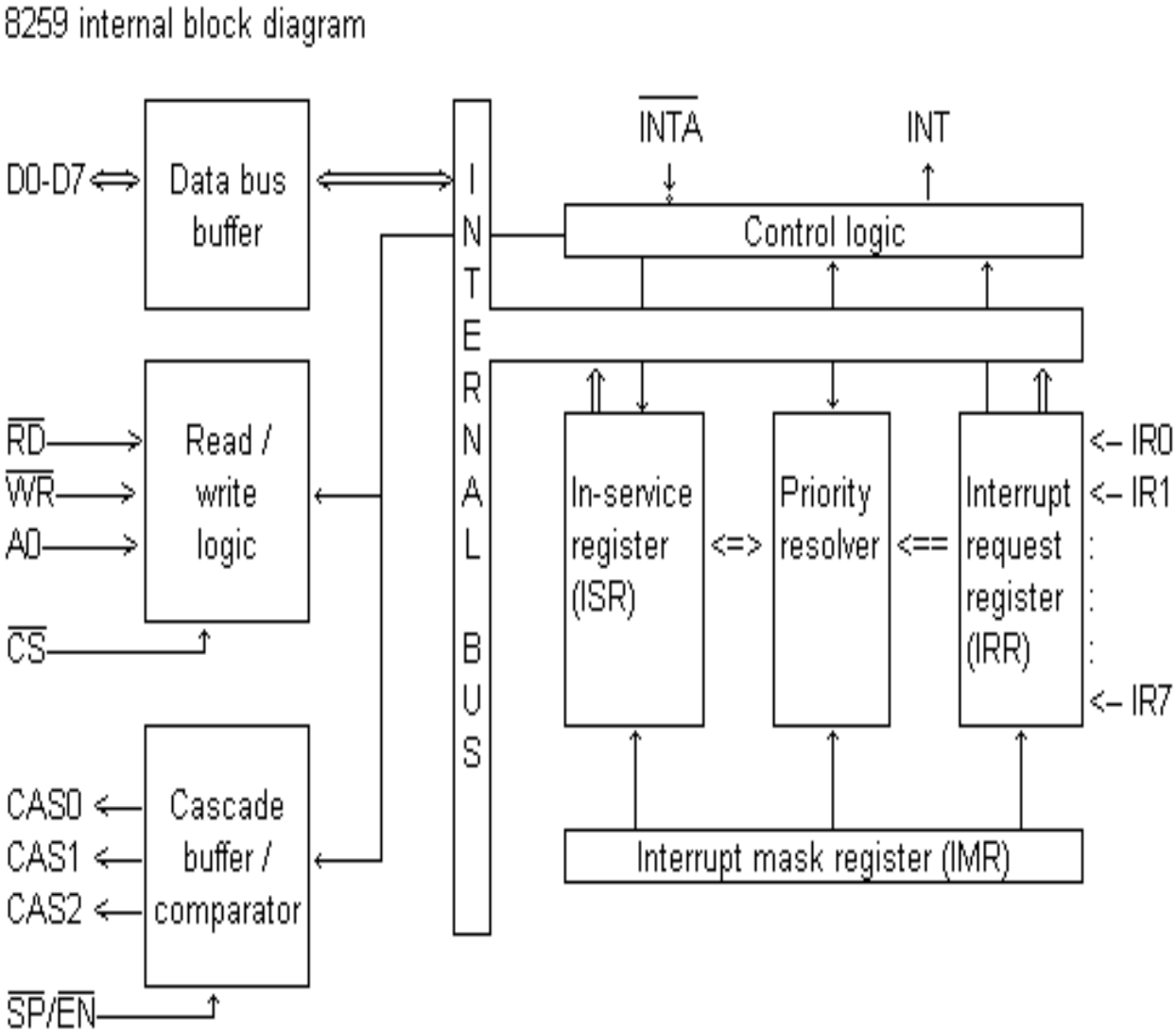
•Starting address of ISR or vector number is programmable.

•No clock required.

Block Diagram of 8259

Pin function





D0-D7	Bi-directional, tristated, buffered data lines. Connected to data bus directly or through buffers
RD-bar	Active low read control
WR-bar	Active low write control
A0	Address input line, used to select control register
CS-bar	Active low chip select
CAS0-2	Bi-directional, 3 bit cascade lines. In master mode, PIC places slave ID no. on these lines. In slave mode, the PIC reads slave ID no. from master on these lines. It may be regarded as slave-select.
SP-bar / EN-bar	Slave program / enable. In non-buffered mode, it is SP-bar input, used to distinguish master/slave PIC. In buffered mode, it is output line used to enable buffers
INT	Interrupt line, connected to INTR of microprocessor
INTA-bar	Interrupt ack, received active low from microprocessor
IR0-7	Asynchronous IRQ input lines, generated by peripherals.

Control logic :-it generates INT signal in response to this MP send INTA bar, it release 3 byte call instruction or one byte vectored addr

IIR :- Interrupt request –it is used to store all the pending request ,MP can read the content of this register by issuing appropriate command word

ISR : In service routine :- it is used to store all the interrupt levels currently being serviced each bit of this register is set by priority resolver and reset by end of interrupt command word ,MP can read the content of this register by issuing appropriate command word

Priority resolver : it determines the priority of IIR

IMR (interrupt mask register): -it is programmable register, it is used to mask out unwanted interrupt

Interrupt sequence (single PIC)

1. One or more of the IR lines goes high.
2. Corresponding IRR bit is set.
3. 8259 evaluates the request and sends INT to CPU.
4. CPU sends INTA-bar.
5. Highest priority ISR is set. IRR is reset.
6. 8259 releases CALL instruction on data bus.
7. CALL causes CPU to initiate two more INTA-bar's.
8. 8259 releases the subroutine address, first low byte then high byte.
9. ISR bit is reset depending on mode.

Single PIC System

085 microprocessor.

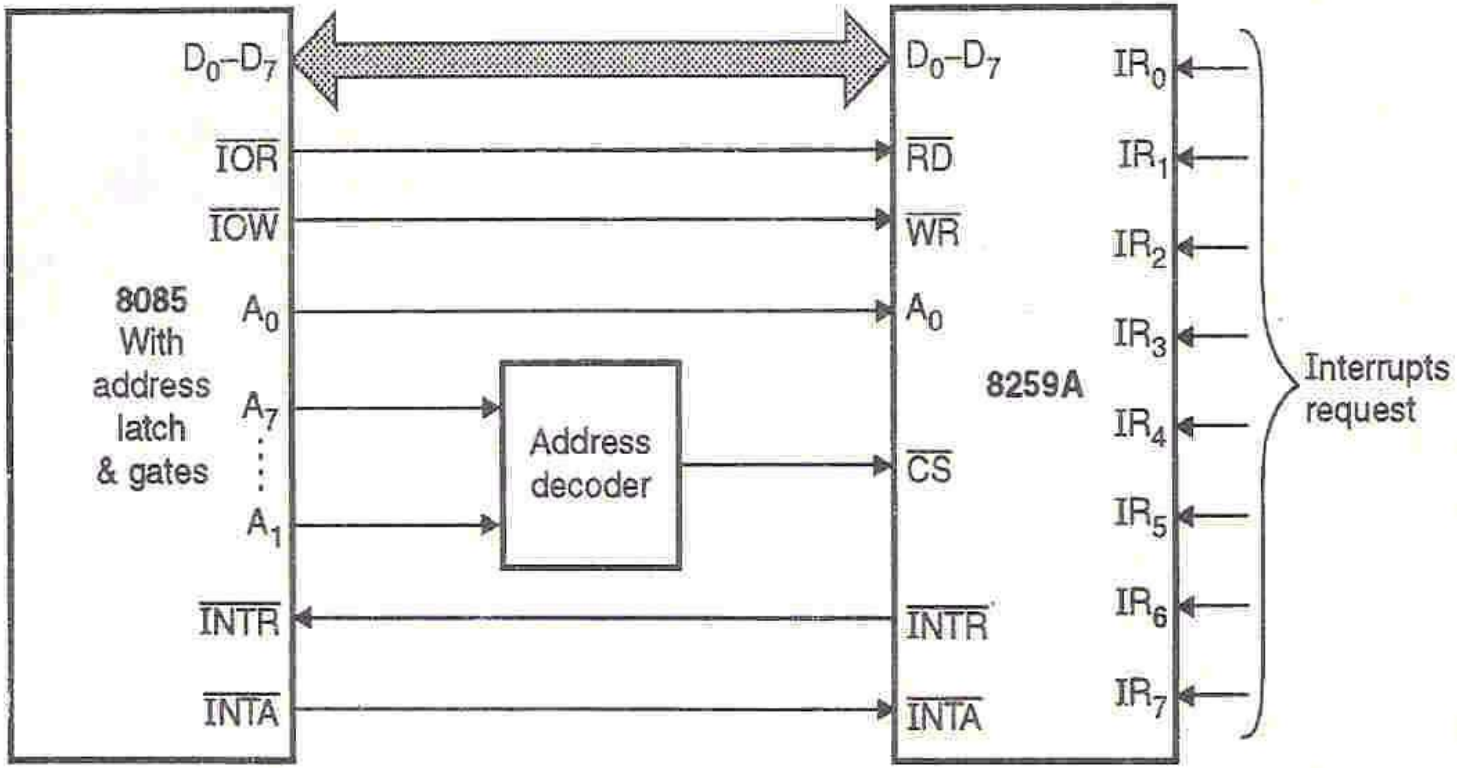


Fig. 10.29 : Single PIC system

NTERRUPT OPERATION ·

Cascaded PICs system (vectored mode)

- 8 pics may be cascaded to act as slave unit ot maser PIC thus total nos of 64 pics may be connected to MP

- In this system INT of salve PIC are connected to corresponding IR pin of master PIC

- Each PIC has its own addr know as slave identification nos

- 3 lines of CAS0 to CAS2 of master PIC are connected to slave pic

- SP of master is connected to Vcc and SP line of slave Pic is Grounded

- Each slave is identified by its slave identification nos through CAS 0 to CAS 2

Cascaded PIC

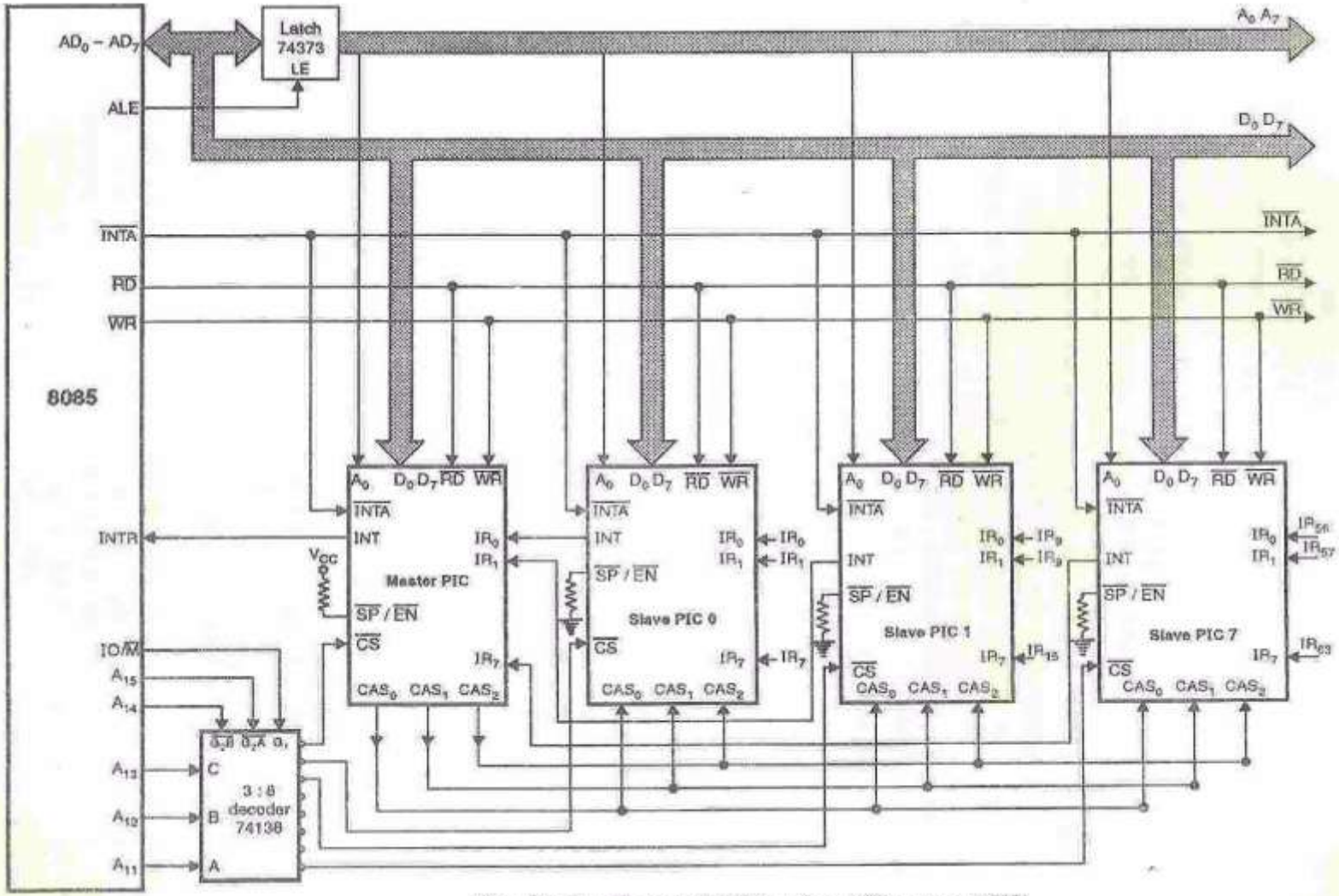


Fig. 10.30 : Cascaded PIC system I/O mapped I/O

Polled PIC system : -

- In this system Mp check status of each PIC
- Nos of PIC that can be connected with MP is very large
- The PPI activate one are more IR this sets the corresponding IIR
- MP issues a poll command to status of 8259
- The MP executes a I/O read cycle to read polled word from PIC
- MP decodes polled word and call subroutine
- If I = 0 the MP issue a poll command to another PIC in this way the MP checks status of all PIC

Polled PIC system

128 × 8 = 1024 peripherals). Fig. 10.31 shows 8259A interface in polled mode.

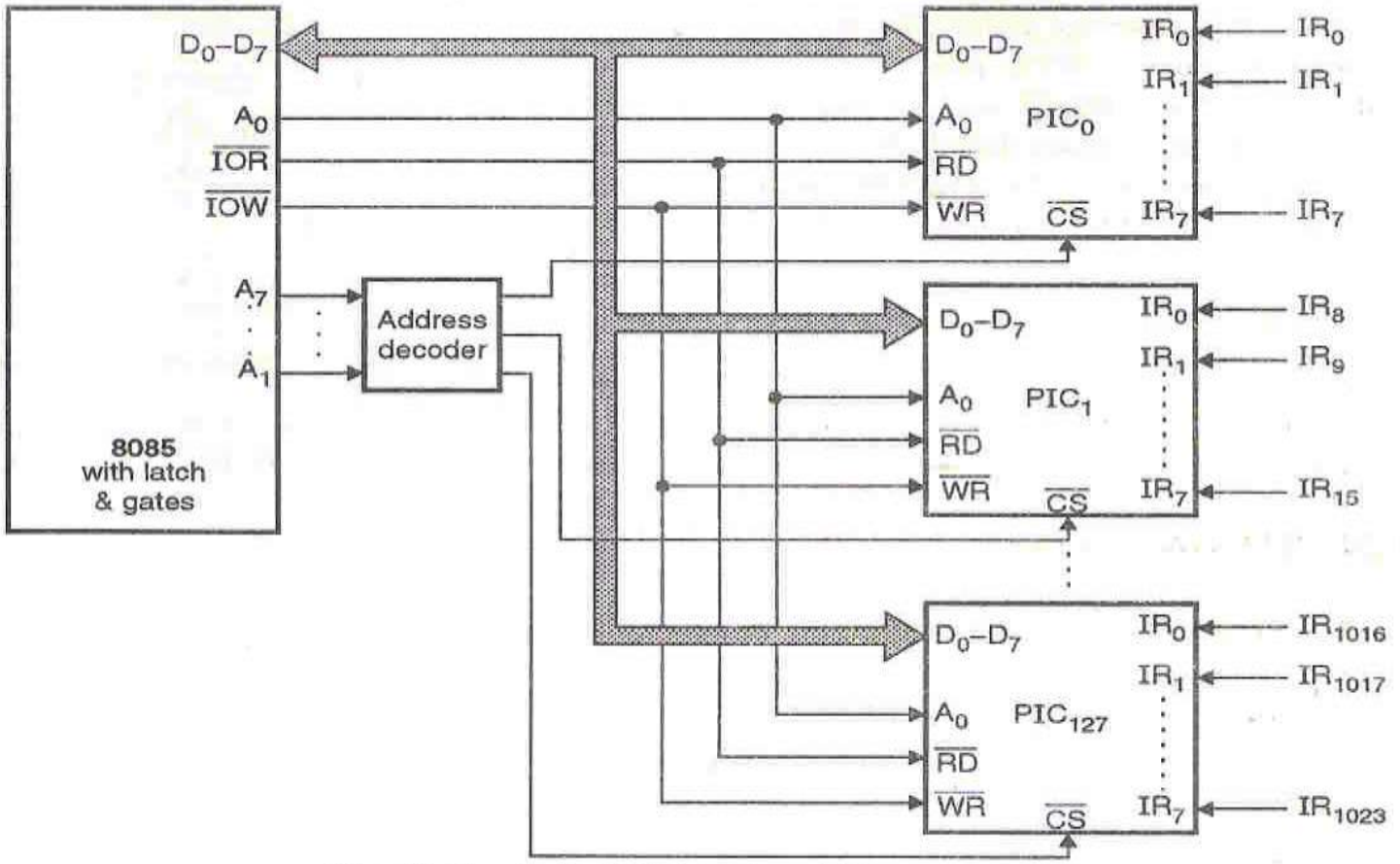
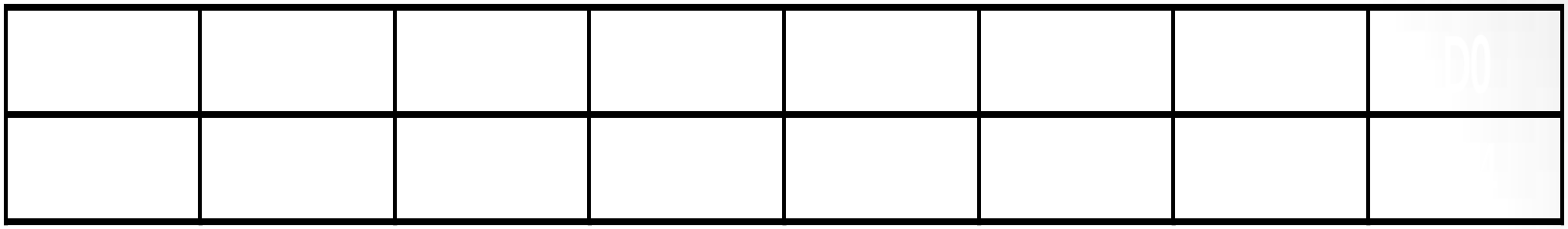


Fig. 10.31 : Polled PIC system (I/O mapped I/O)

Operation :

Initialization of Command Word [ICW 1]



IC4: - This indicate whether ICW4 is required or not(1 ICW 4 is issued , 0 ICW 4 is not issued)

SNGL: - indicate whether PIC is cascaded or not (1 single PIC , 0 cascaded PIC)

ADI : -it determine the spacing between successive interrupt routine (1 space between to ISR is 8 byte, 0 space between to ISR is 4 byte)

LTIM : - (If 1 all IR input level triggered,0 all IR input level triggered)

A5,A6,A7 : - this bits are used to provide 4 or 8 byte spacing between successive ISR (where the A0 to A$ is provided by 8259)

Initialization of Command Word [ICW 2]

							D0
							A8/0

- This bit is used to program higher byte of ISR addr in 8085
- Here A0 line should be 1
- Here D0 to D7 line used to program hihger addr bit of ISR addr
- Where D0 to D3 are provided by 8259 it self to choose IR0 to IR 7

Initialization of Command Word [ICW 3] master format

This is ICW 3 for master PIC here A0 is 1 means this ICW should be store in master PIC ,this tells the master PIC that which IR input is connected to slave

Initialization of Command Word [ICW 3] slave format

							D0
							ID0

- This CWR must be written in slave PIC
- Here ID2,ID1,ID0 bit are used to assign 3 bit slave identification number(000 to 111 for salve0 to slave7)

Initialization of Command Word [ICW 4]

- This CWR is used to initialize the 8259 in different modes
- [D0 = 1 8085 mode ,0 8086 mode][D1 =1normal EOI,0= AutoEOI]
- [D2 :- master/slave][D3 BUFFER mode][SFNM Special fully nested mode]

Operational command word [OCW 1]

•After initialization 8259 is ready to accept IR

•OCW can change the priority,modes of opration,controle and EOI command

•Here OCW 1 is used to mask out un wanted interrupt

Operational command word [OCW 2]

•Provide EOI which Clear appropriate ISR ,Rotate priority in normal and auto EOI mode

Operational command word [OCW 3] This command is used

•To operate 8259 in special mask mode ,polled mode

•To read IIR and ISR

- **8279 PKDC**

intel.

8279/8279-5
PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

- Simultaneous Keyboard Display Operations
- Scanned Keyboard Mode
- Scanned Sensor Mode
- Strobed Input Entry Mode
- 8-Character Keyboard FIFO
- 2-Key Lockout or N-Key Rollover with Contact Debounce
- Dual 8- or 16-Numerical Display
- Single 16-Character Display
- Right or Left Entry 16-Byte Display RAM
- Mode Programmable from CPU
- Programmable Scan Timing
- Interrupt Output on Key Entry
- Available in EXPRESS
  - Standard Temperature Range
  - Extended Temperature Range

The Intel® 8279 is a general purpose programmable keyboard and display I/O interface device designed for use with Intel® microprocessors. The keyboard portion can provide a scanned interface to a 64-contact key matrix. The keyboard portion will also interface to an array of sensors or a strobed interface keyboard, such as the hall effect and ferrite variety. Key depressions can be 2-key lockout or N-key rollover. Keyboard entries are debounced and strobed in an 8-character FIFO. If more than 8 characters are entered, overrun status is set. Key entries set the interrupt output line to the CPU.

The display portion provides a scanned display interface for LED, incandescent, and other popular display technologies. Both numeric and alphanumeric segment displays may be used as well as simple indicators. The 8279 has 16x8 display RAM which can be organized into dual 16x4. The RAM can be loaded or interrogated by the CPU. Both right entry, calculator and left entry typewriter display formats are possible. Both read and write of the display RAM can be done with auto-increment of the display RAM address.

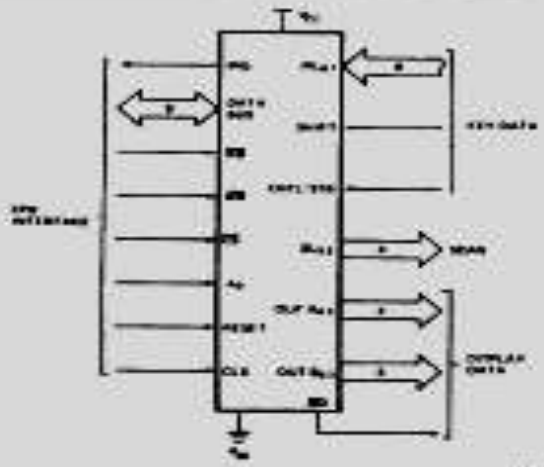


Figure 1. Logic Symbol

Figure 2. Pin Configuration

FUNCTIONAL DESCRIPTION

Since data input and display are an integral part of many microprocessor designs, the system designer needs an interface that can control these functions without placing a large load on the CPU. The 8279 provides this function for 8-bit microprocessors.

The 8279 has two sections: keyboard and display. The keyboard section can interface to regular typewriter style keyboards or random toggle or thumb switches. The display section drives alphanumeric displays or a bank of indicator lights. Thus the CPU is relieved from scanning the keyboard or refreshing the display.

The 8279 is designed to directly connect to the microprocessor bus. The CPU can program all operating modes for the 8279. These modes include:

Input Modes

- Scanned Keyboard—with encoded (8 x 8 key keyboard) or decoded (4 x 8 key keyboard) scan lines. A key depression generates a 6-bit encoding of key position. Position and shift and control status are stored in the FIFO. Keys are automatically debounced with 2-key lockout or N-key rollover.
- Scanned Sensor Matrix—with encoded (8 x 8 matrix switches) or decoded (4 x 8 matrix switches) scan lines. Key status (open or closed) stored in RAM addressable by CPU.
- Strobed Input—Data on return lines during control line strobe is transferred to FIFO.

Output Modes

- 8 or 16 character multiplexed displays that can be organized as dual 4-bit or single 8-bit (B_L = D_0, A_0 = D_1).
- Right entry or left entry display formats.

Other features of the 8279 include:

- Mode programming from the CPU.
- Clock Prescaler.
- Interrupt output to signal CPU when there is keyboard or sensor data available.
- An 8 byte FIFO to store keyboard information.
- 16 byte internal Display RAM for display refresh. This RAM can also be read by the CPU.

PRINCIPLES OF OPERATION

The following is a description of the major elements of the 8279 Programmable Keyboard/Display interface device. Refer to the block diagram in Figure 3.

I/O Control and Data Buffers

The I/O control section uses the $\overline{CS}$, A_0, $\overline{RD}$ and $\overline{WR}$ lines to control data flow to and from the various internal registers and buffers. All data flow to and from the 8279 is enabled by $\overline{CS}$. The character of the information, given or desired by the CPU, is identified by A_0. A logic one means the information is a command or status. A logic zero means the information is data. $\overline{RD}$ and $\overline{WR}$ determine the direction of data flow through the Data Buffers. The Data Buffers are bi-directional buffers that connect the internal bus to the external bus. When the chip is not selected ($\overline{CS}$ = 1), the devices are in a high impedance state. The drivers: input during $\overline{WR}$ • $\overline{CS}$ and output during $\overline{RD}$ • $\overline{CS}$.

Control and Timing Registers and Timing Control

These registers store the keyboard and display modes and other operating conditions programmed by the CPU. The modes are programmed by presenting the proper command on the data lines with A_0 = 1 and then sending a $\overline{WR}$. The command is latched on the rising edge of $\overline{WR}$. The command is then decoded and the appropriate function is set. The timing control contains the basic timing counter chain. The first counter is a ÷ N prescaler that can be programmed to yield an internal frequency of 100 kHz which gives a 5.1 ms keyboard scan time and a 10.3 ms debounce time. The other counters divide down the basic internal frequency to provide the proper key scan, row scan, keyboard matrix scan, and display scan times.

Scan Counter

The scan counter has two modes. In the encoded mode, the counter provides a binary count that must be externally decoded to provide the scan lines for the keyboard and display. In the decoded mode, the scan counter decodes the least significant 2 bits and provides a decoded 1 of 4 scan. Note that when the keyboard is in decoded scan, so is the display. This means that only the first 4 characters in the Display RAM are displayed.

In the encoded mode, the scan lines are active high outputs. In the decoded mode, the scan lines are active low outputs.