UNIT II - PERIPHERAL INTERFACING

Unit-II

Peripheral Interfacing with Microprocessor: Static and Dynamic memories, Vector interrupt table, Interrupt service routine, Interfacing of microprocessor with Programmable Interrupt Controller 8259, DMA controller 8257, Programmable peripheral Interface-8255. **9 Hrs**

INTRODUCTION

Microprocessor based system design involves interfacing of the **processor with one or more peripheral devices** for the purpose of communication with **various input and output devices connected** to it. During the **early days** of the microprocessor revolution, these techniques required **complex hardware** consisting of medium scale integration devices making the design highly complex and **time consuming**. So, the manufacturers **(INTEL**) have **developed a large number of general and special purpose peripheral devices**, most of them being single chip circuits. They are also programmable devices. Hence these peripheral devices are found to be of tremendous use to a system designer.

Peripheral devices can broadly be classified into two categories.

(a) General purpose peripherals and

(b) Special purpose peripherals (Dedicated function peripherals)

General purpose peripheral devices that perform a task but may be used for interfacing a variety of I/O devices to microprocessors. The general-purpose devices are given below:

□ Simple I/O	--	(Non-programmable)
□ Programmable peripheral Interface (PPI)	–	(8255)
□ Programmable Interrupt Controller	–	(8259)
□ Programmable DMA Controller	–	(8237/8257)

□ Programmable Communication Interface	–	(8251)
□ Programmable Interval Timer	–	(8253/8254)

Special function peripherals are devices that may be used for interfacing a microprocessor to a specific type of I/O device. These peripherals are more complex and therefore, relatively more expensive than general purpose peripherals. The special function peripherals (Dedicated function peripherals) are

- □ Programmable CRT Controller
- □ Programmable Floppy Disc Controller
- □ Programmable Hard Disc Controller
- □ Programmable Keyboard and display interface.

The functioning of these devices varies depending on the type of I/O device they are controlling.

8255 - PROGRAMMABLE PERIPHERAL INTERFACING (8255 - PPI):

- **8255** is a widely used, programmable, parallel I/O device.
- It can be programmed to transfer data under various conditions from simple I/O to interrupt I/O.

Introduction

INTEL introduced this programmable peripheral interface (PPI) chip 8255A for interfacing peripheral devices to the 8085 system. This versatile chip **8255A is used as a general-purpose peripheral device for parallel data transfer between microprocessor and a peripheral device by interfacing the device to the system data bus.** The PPI has three programmable I/O ports viz., **Port A, Port B and Port C each of 8 bit width**. Port C can be treated as two ports – Port C upper (PC_{7}-4) and Port lower (PC_{3} – 0) and these two can be independently programmed as INPUT or OUTPUT ports also.

- **Salient Features**

i. It is a **general purpose programmable I/O** device which is compatible with all INTEL processors and also most other processors.

ii. It provides **24 I/O pins** which may be **individually programmed** in **two groups**.

iii. This chip is also completely TTL compatible.

iv. It is available in **40 pin DIP** and 44 pin plastic leaded chip carrier (PLCC) packages.

v. It has **three 8 bit ports. Port A, Port B and Port C**. Port C is treated as two 4 bitports also.

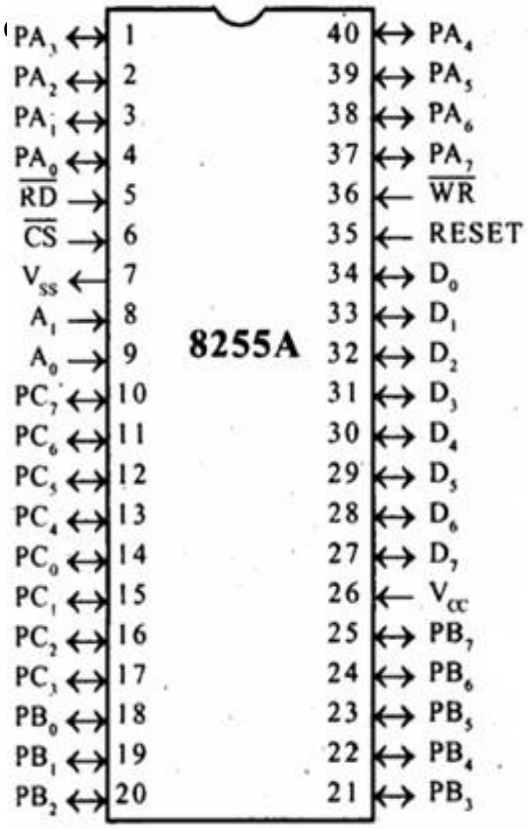
vi. This 8255 is mainly programmed in **two modes** (a) the **I/O mode** and (b)The **bit set/reset mode** (BSR) mode. The I/O mode is further divided into three modes:**Mode 0, Mode 1, and Mode 2**.

vii. An **8 bit control resister** is used to configure the modes of 8255.

There is also another 8 bit port called **control port**, which decides the configurationof 8255 ports. This port is written by the microprocessor only.

PIN CONFIGURATION OF 8255:

- **D_0-D_7 (Data Bus):** Bidirectional, tri-state, data bus lines connected to the system data bus. They are used to transfer data and control word from microprocessor to 8255 or receive data or status word from 8255 to 8085.
- **PA_0-PA_7(PortA):** These 8-bit bidirectional I/O pins are used to send or receive data from O/P or I/P device.
- **PB_0-PB_7(Port B):** These 8-bitbidirectionalI/O pins are used to send or receive



Pin	Description
D_0 - D_7	Data lines
RESET	Reset input
$\overline{CS}$	Chip select
$\overline{RD}$	Read control
$\overline{WR}$	Write control
A_0, A_1	Internal address
PA_7 - PA_0	Port-A pins
PB_7 - PB_0	Port-B pins
PC_7 - PC_0	Port-C pins
V_{CC}	+5V
V_{SS}	0V (GND)

Figure 1: Pin Diagram of 8255 **Table 1: Pin Description**

- **PC_0- PC_7(port C):** These 8-bit bidirectional I/O pins are divided into two groups PC_L (PC_0- PC_3)and PC_U (PC_4- PC_7). These groups can individually transfer data in or out when programmed I/O. When programmed in bidirectional or handshake modes these bits are used as handshake signals.
- **RD' (Read):** MPU or CPU reads data in the ports or the status word through data buffer.
- **WR' (Write):** MPU or CPU writes data in the ports or the control register through data Buffer.
- **CS' (Chip Select):** It is an active below input which can be used to enable 8255for data transfer operation between CPU (MPU) and 8255.
- **RESET:** It is an active high input used to reset 8255. When reset input is high, the control register is cleared and all the ports are set to the input mode. Usually RESETOUT signal from 8085 is used to reset 8255.
- **A_0&A_1:** These input signals along with RD', WR' inputs control the selection of control / status word registers or one of three ports.

$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	A_1	A_0	Input (Read) cycle
0	1	0	0	0	Port A to Data bus
0	1	0	0	1	Port B to Data bus
0	1	0	1	0	Port C to Data bus
0	1	0	1	1	CWR to Data bus

$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	A_1	A_0	Output (Write) cycle
1	0	0	0	0	Data bus to Port A
1	0	0	0	1	Data bus to Port B
1	0	0	1	0	Data bus to Port C
1	0	0	1	1	Data bus to CWR

$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	A_1	A_0	Function
X	X	1	X	X	Data bus tristated
1	1	0	X	X	Data bus tristated

Table 2: Control Word Register

BLOCK DIAGRAM OF 8255:

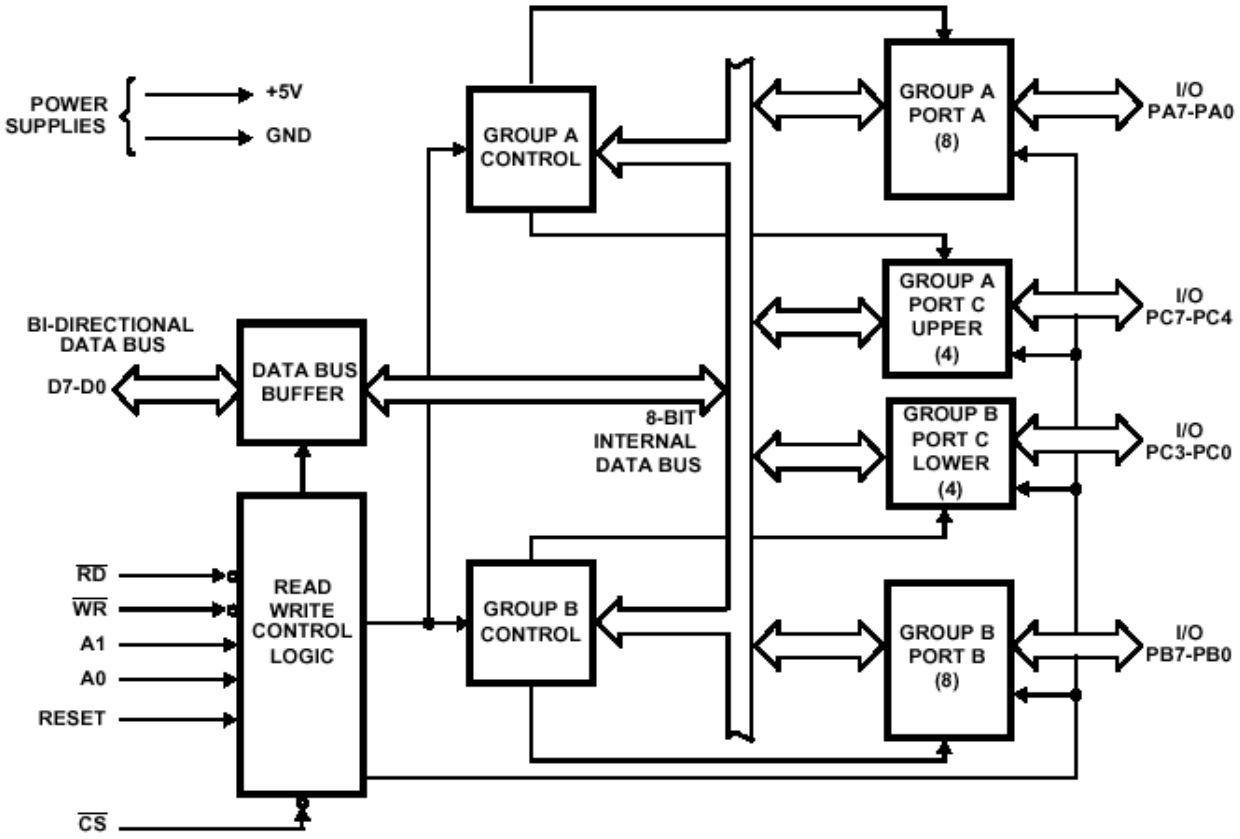


Figure 2: Block Diagram of 8255

- **Data Bus buffer :**

  Tri-state bidirectional buffer is used to interface the internal data bus of8255to the system data bus. Output data from the MPU to the ports or control register and the input data to the MPU from the ports or status register are all pushed through the buffer.

- **Control Logic**

  This block accepts control bus signals as well as inputs from the address busand issues commands to the individual group control blocks (Group A Control and Group B Control) as shown in Fig.2.

- **Group A Control and Group B Control**

  Group A control block controls PortA and PC7-PC4.Group B controls blockcontrols Port Band PC_3- PC_0.

- **PortA**

  This has 8-bit latched and buffered output and an 8-bit input latch. It can be Programmed in three modes:

  Mode 0: Simple I/O mode

  Mode 1: I/O with Handshaking mode

  Mode 2: Bidirectional data transfer mode.

- **Port B**

  This has 8-bit I/O latch/buffer and an 8-bit data input buffer. It can be programmed in mode 0 or mode 1.

- **Port C**

  This has 8-bit unlatched input buffer and an 8-bit output latch/buffer. Port C can be splitted into two parts and each bit can be used as control signals for Port Aand Port B in handshake mode. It can be programmed for BSR (Bit Set / Reset mode) operation.

Modes of Operation:

1. BSR mode
2. I/O mode
   - Mode 0: Simple I/O mode
   - Mode 1 : I/O with Handshaking mode
   - Mode 2: Bidirectional data transfer mode

1. BSR (Bit Set/Reset) Mode:

Individual bits of Port C can be set or reset by sending out a single OUT instruction to the control register. When Port C is used for control/status operation, this feature can be used to set or reset individual bits. For BSR mode control word is given below.

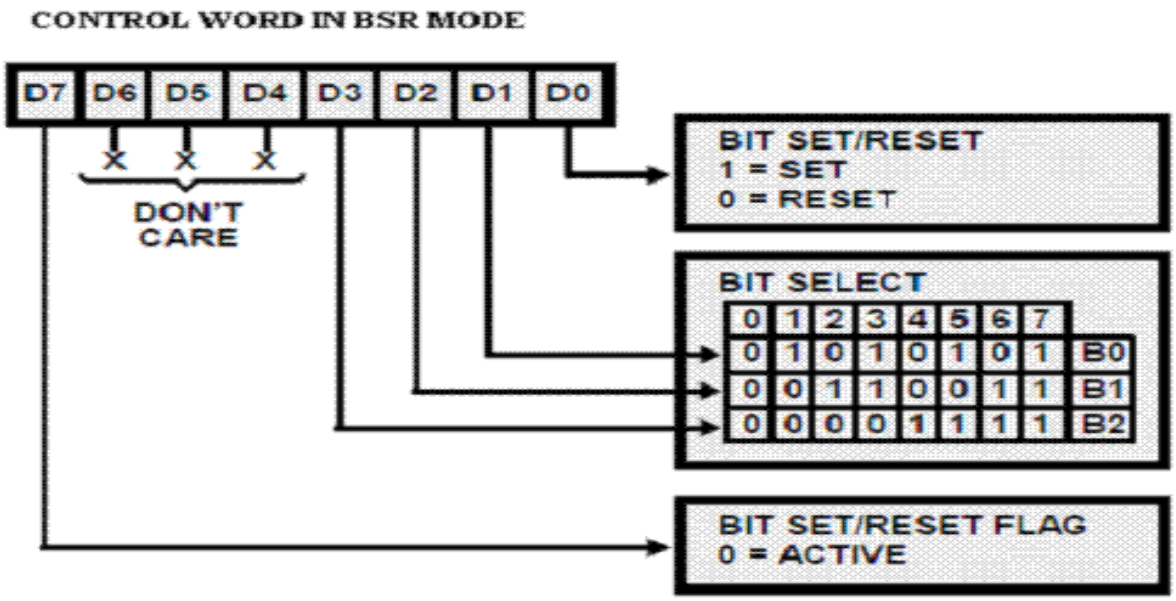


Figure 3:Control word for BSR mode

A BSR word is to be written for each bit that is to be set or reset. The BSR word can also be used for enabling or disabling the interrupt signals generated by Port c when 8255is programmed for mode 1or mode2 operation.

2. **I/O mode:**

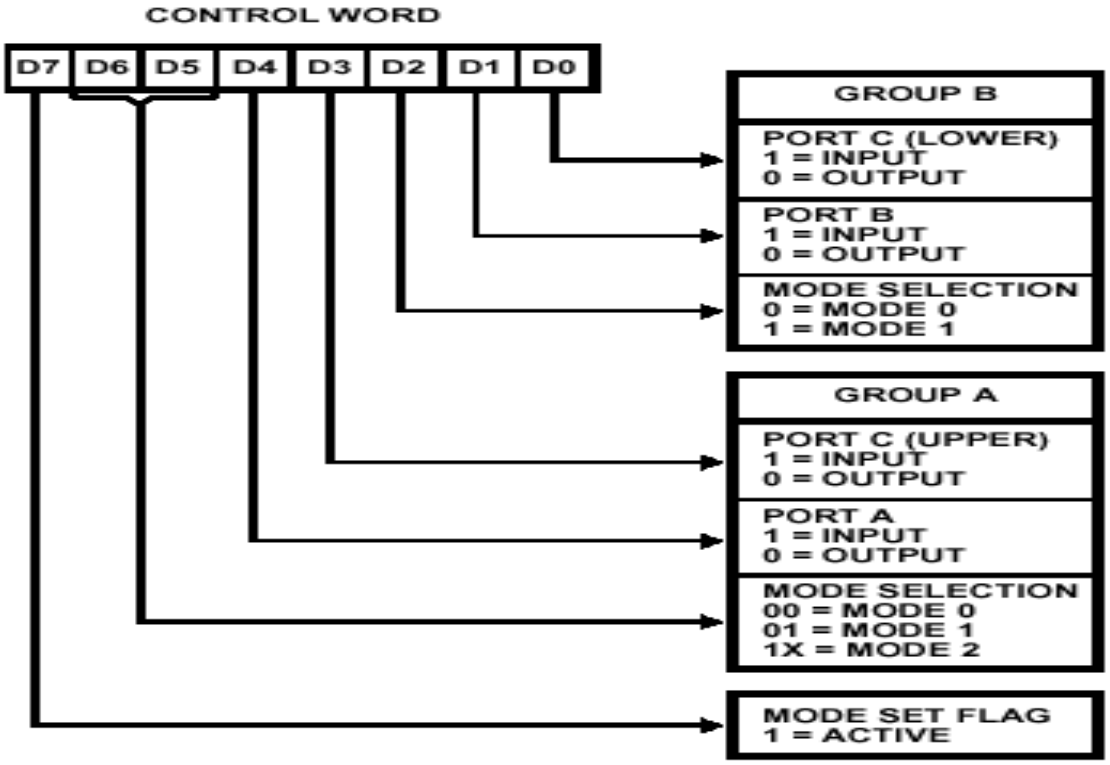


Figure 4: Control word format for I/O mode

i) Mode 0 : Basic Input/output

This mode provides simple input and output operations for each of the three ports. Data is simply written to or read from a specified port.

Basic functional definition

In mode 0:

- Therearetwo8-bitports (A and B) and two 4-bit ports [C (lower)] and [C (upper)].
- Any port can be an input port or an output port.
- Outputs are latched.
- Inputs are not latched.
- 16 different input/output configurations are possible in this mode.

ii) Mode1: Strobed Input/Output

It provides means for transferring I/O data to or from a specified port in conjunction with strobes or hand-shaking signals. Port A and port B use the lines on portC for handshaking signals.

Basic functional definition

In mode1:

- There are two groups (Group A and B). .
- Each group contains one 8-bit data port and one 4-bit control data port.
- The 8-bit data port can be either an input port or an output port. Both inputs and outputs are latched.
- The4-bit port is used for control as well as for status of the 8-bit data port.

iii) Mode2 : Strobed Bidirectional Bus

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. Handshaking signals are provided to maintain a proper bus flow discipline. Interrupt generation and enable/disable Functions are also available.

Basic functional definition

In mode 2 (used in Group A only):

- There is one 8-bit bidirectional 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 as well as for status of the 8-bit bidirectional bus port (port A).

PROGRAMMABLE INTERRUPT CONTROLLER (PIC) - 8259

Introduction

There is an absolute need of this Programmable Interrupt Controller for Interfacing I/O devices to the microprocessor. The 8085 processor has 5 interrupt lines namely, Trap, RST 7.5, RST 6.5, RST 5.5 and INTR. So, we can interface five I/O devices, which can perform the interrupt driven data transfer safely. But, suppose we wish to connect more than five I/O devices, to the microprocessor, then we may have to connect more than one I/O device to the interrupt lines. This will affect the interrupt driven data transfer and the microprocessor has to perform polling. i.e, it has to check each device, which is in need of interrupt service. This polling has the dis-advantage of long time and slow interrupt response. Hence to overcome all these problems, INTEL introduced the 28 pin DIP chip -8259. This device accepts interrupt requests from as many as 8 devices independently and as many as 64 I/O devices by cascading method.

Salient Features

INTEL 8259 is a single chip programmable interrupt controller which is compatible with 8085, 8086 and 8088 processors.

- □ It is a 28 pin DIP IC with N-Mos technology and requires a single +5 DC supply.
- □ It handles up to eight vectored priority interrupts for the CPU and cascadable for up to 64 vectored priority interrupts without the need of any additional circuitry.
- □ when two 8259s are cascaded through cascade lines the first 8259 will act as master and the second 8259 will act as a slave.

PIN CONFIGURATION OF 8529

The pin diagram of 8259 is shown below . The pin details are given below

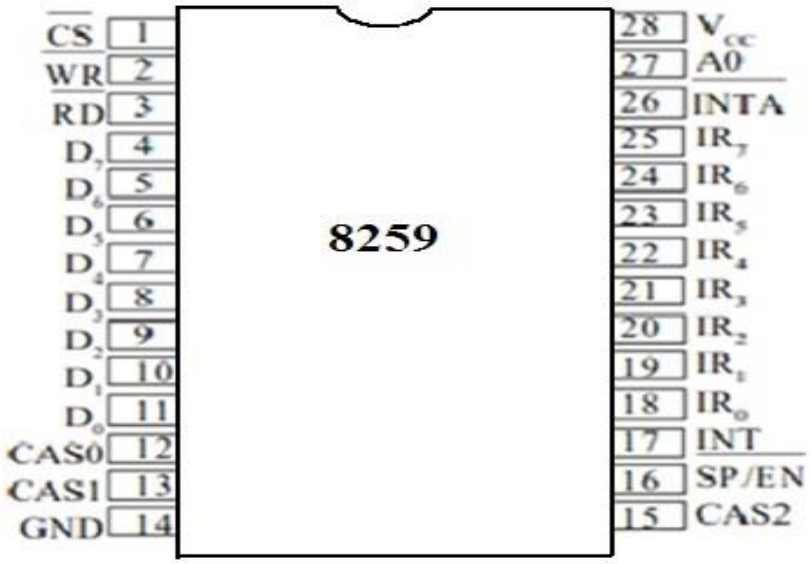


Figure 5:Pin Diagram of 8259

Symbol	Pin No.	Type	Name and Function
V_{CC}	28	I	**SUPPLY:** +5V Supply.
GND	14	I	**GROUND**
$\overline{CS}$	1	I	**CHIP SELECT:** A low on this pin enables $\overline{RD}$ and $\overline{WR}$ communication between the CPU and the 8259A. INTA functions are independent of CS.
$\overline{WR}$	2	I	**WRITE:** A low on this pin when CS is low enables the 8259A to accept command words from the CPU.
$\overline{RD}$	3	I	**READ:** A low on this pin when CS is low enables the 8259A to release status onto the data bus for the CPU.
D_7–D_0	4–11	I/O	**BIDIRECTIONAL DATA BUS:** Control, status and interrupt-vector information is transferred via this bus.
CAS_0–CAS_2	12, 13, 15	I/O	**CASCADE LINES:** The CAS lines form a private 8259A bus to control a multiple 8259A structure. These pins are outputs for a master 8259A and inputs for a slave 8259A.
$\overline{SP}/\overline{EN}$	16	I/O	**SLAVE PROGRAM/ENABLE BUFFER:** This is a dual function pin. When in the Buffered Mode it can be used as an output to control buffer transceivers (EN). When not in the buffered mode it is used as an input to designate a master (SP = 1) or slave (SP = 0).
INT	17	O	**INTERRUPT:** This pin goes high whenever a valid interrupt request is asserted. It is used to interrupt the CPU, thus it is connected to the CPU's interrupt pin.
IR_0–IR_7	18–25	I	**INTERRUPT REQUESTS:** Asynchronous inputs. An interrupt request is executed by raising an IR input (low to high), and holding it high until it is acknowledged (Edge Triggered Mode), or just by a high level on an IR input (Level Triggered Mode).
$\overline{INTA}$	26	I	**INTERRUPT ACKNOWLEDGE:** This pin is used to enable 8259A interrupt-vector data onto the data bus by a sequence of interrupt acknowledge pulses issued by the CPU.
A_0	27	I	**AO ADDRESS LINE:** This pin acts in conjunction with the $\overline{CS}$, $\overline{WR}$, and $\overline{RD}$ pins. It is used by the 8259A to decipher various Command Words the CPU writes and status the CPU wishes to read. It is typically connected to the CPU A0 address line (A1 for 8086, 8088).

Dr.

BLOCK DIAGRAM OF 8259

The block diagram of programmable interrupt controller is shown in Fig. below. The block diagram consists of eight sub units. They are Control logic, Read/write logic, Data bus buffer. Three register (IRR, ISR and IMR), 5 priority resolver and cascade buffer. The functions of each unit are explained below.

Interrupt Request Register (IRR) & Interrupt Service Register (ISR)

The interrupts at the IR input lines are handled by two registers in cascade, the Interrupt Request Register (IRR) and the In-Service (ISR). The IRR is used to store all the interrupt levels which are requesting service; and the ISR is used to store all the interrupt levels which are being serviced.

Priority Resolver

This logic unit determines the priorities of the bits set in the IRR. The highest priority is selected and strobed in to the corresponding bit of the ISR during pulse.

Interrupt Mask Register (IMR)

The IMR stores the bits which mask the interrupt lines. The IMR operates on the IRR. Masking of a higher priority input will not affect the interrupt request lines of lower priority.

Control Logic

This unit has two pins. INT (Interrupt) as an output pin and (interrupt acknowledge) as an input pin. The INT is connected to the interrupt pin of the microprocessor unit. Whenever an interrupt is noticed by the CPU, it generates signal

. Cascade Buffer

This function block stores the IDs of all 8259A are used in the system. The associated three I/O pins (CAS0-2) are outputs when the 8259A is used as a master and are inputs when the 8259A is used as a slave. As a master, the 8259A sends the ID of the interrupting slave device onto the CAS0 –2 lines. The slave thus selected will send its preprogrammed subroutine address onto the Data Bus during the next one or two consecutive INTA pulses. (See section ‘‘Cascading the 8259A’’.

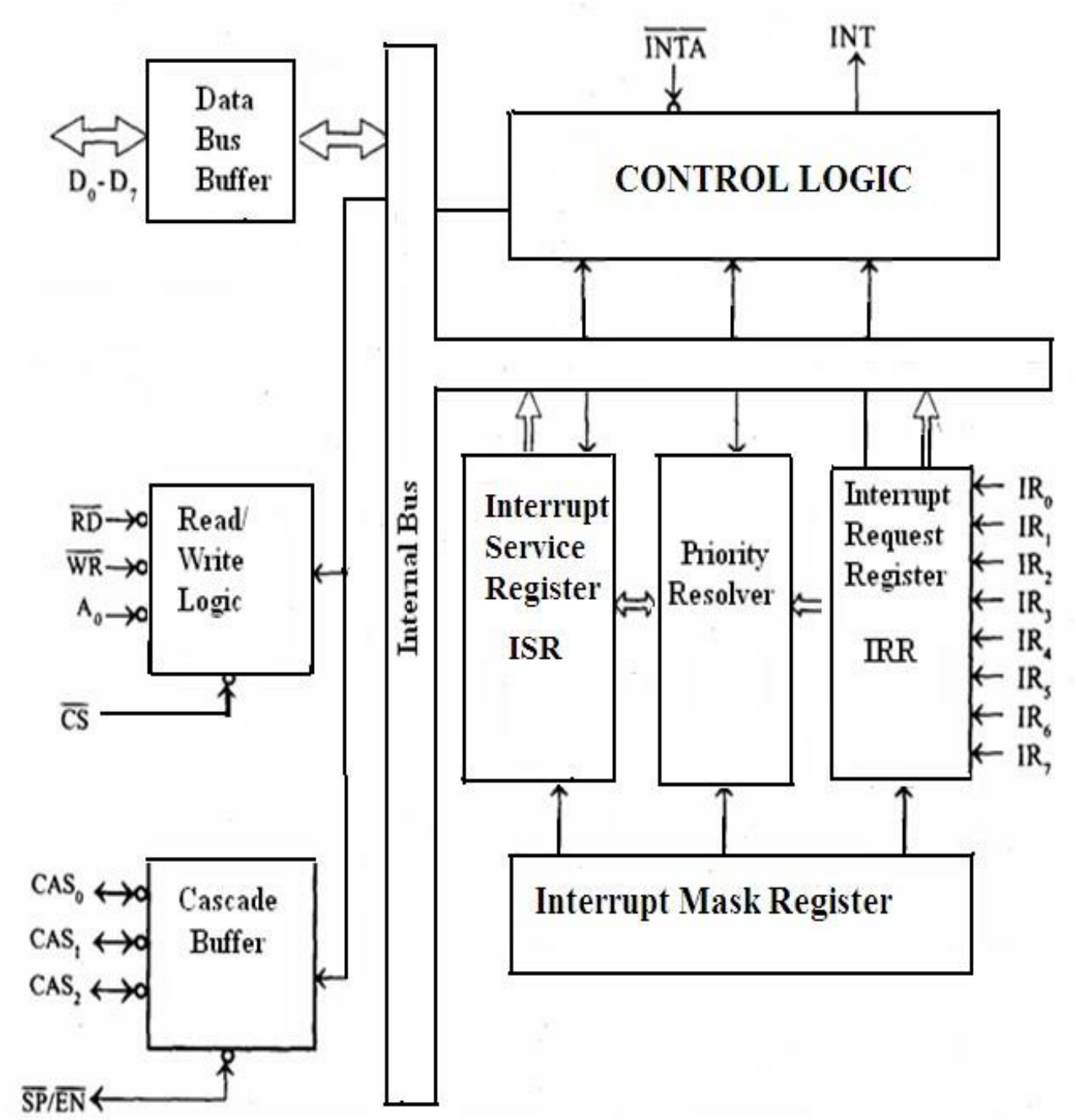


Figure 6: Block Diagram of 8259

WORKING OF 8259

The 8259 accepts interrupt requests from any one of the 8 I/O lines (IR_0 - IR_7). Then it ascertains the priority of the interrupt lines. Then it ascertains the priority of the interrupt lines. Suppose, the received interrupt has higher priority than currently serviced, it interrupts the microprocessor and after receiving the interrupt acknowledgement from microprocessor. It provides a 3 byte CALL instruction. The sequence of steps that occur when an interrupt request line of 8259 goes high is as follows.

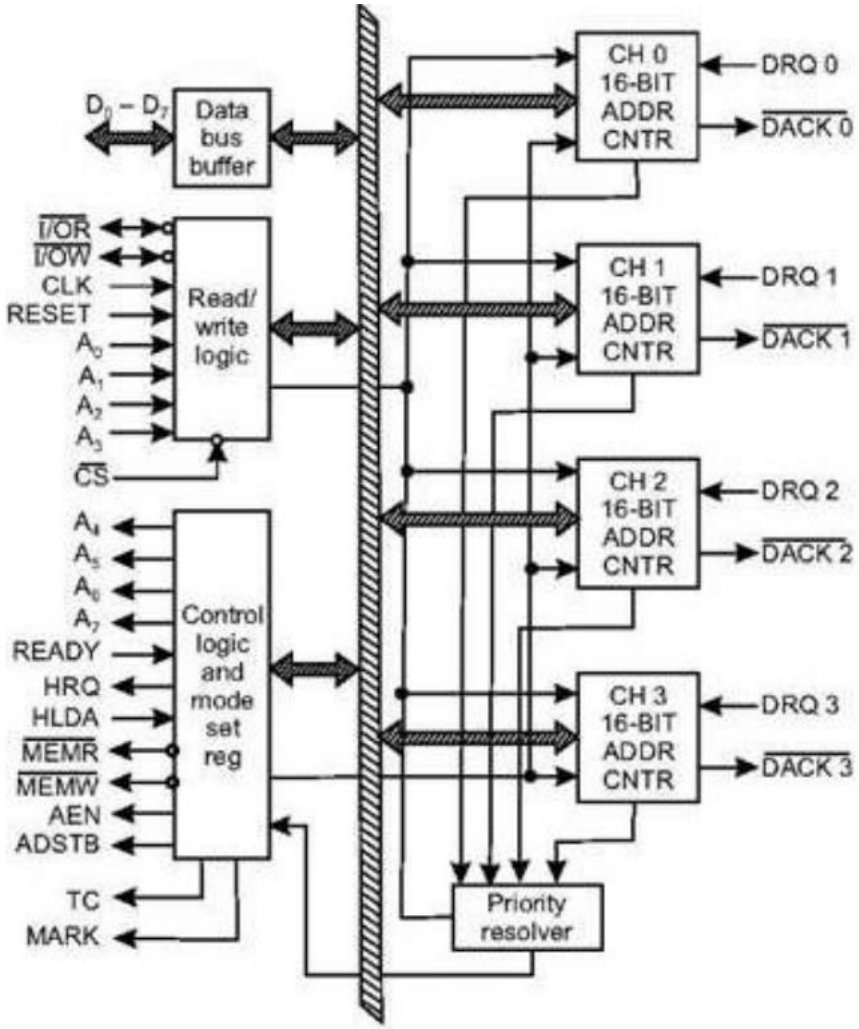
- The 8259 accepts the requests on IR_0 - IR_7 in IRR. Then it checks the contents of IMR whether that request is masked or not.
- The 8259, then checks ISR to know the interrupt levels that are being currently serviced. After this 8259 sends a high INT to 8085 processor. Normally, it is the job of the priority resolver to check the contents of IRR, IMR and ISR and decide whether to activate INT output of 8259 or not.
- Now 8085 processor responds by suspending the program flow at the end of the current instruction and makes low.
- On receiving, 8259 sends code for CALL to the microprocessor on D_{7-0} bus.
- This code for CALL in IR register of 8259 causes the 8085 to issue two more signals. When goes low the second time, 8259 places LSB of ISS address on the data bus. When goes low the third time, 8259 places the MSB of ISS address ont he data bus.
- Now, the microprocessor branches to the ISS after saving the contents of program counter on the stack top.
- After finishing the ISS, the control returns to the main program by popping the top of stack to PC.

Programming 8259

The 8259 requires two types of command words namely, Initialization Command Words (ICW) and Operational Command Words (OCW). The 8259 can be initialized with four ICWs, the first two are essential and the other two are optional based on the modes being used. These words must be issued in a sequence. Once the 8259 is initialized, the 8259 can operate in various modes by using three different OCWs.

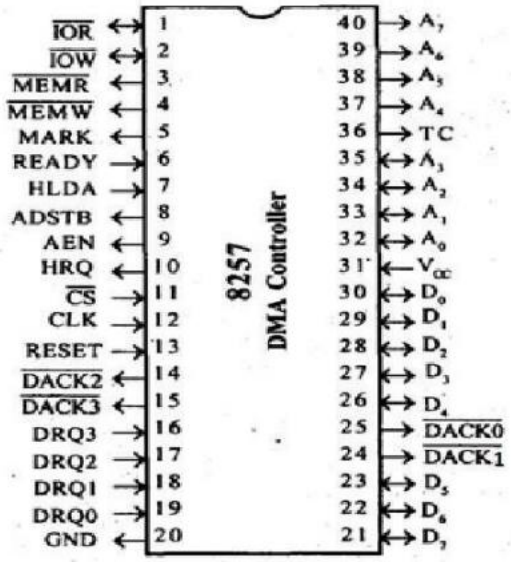
8257 Architecture

The following image shows the architecture of 8257 –



Pin diagram of 8257

8257 Pin Description The following image shows the pin diagram of a 8257 DMA controller



8257 pin description

DRQ0−DRQ3 These are the four individual channel DMA request inputs, which are used by the peripheral devices for using DMA services. When the fixed priority mode is selected, then DRQ0 has the highest priority and DRQ3 has the lowest priority among them.

DACKo − DACK3 These are the active-low DMA acknowledge lines, which updates the requesting peripheral about the status of their request by the CPU. These lines can also act as strobe lines for the requesting devices.

Do − D7 These are bidirectional, data lines which are used to interface the system bus with the internal data bus of DMA controller. In the Slave mode, it carries command words to 8257 and status word from 8257. In the master mode, these lines are used to send higher byte of the generated address to the latch. This address is further latched using ADSTB signal.

IOR It is an active-low bidirectional tri-state input line, which is used by the CPU to read internal registers of 8257 in the Slave mode. In the master mode, it is used to read data from the peripheral devices during a memory write cycle. IOW It is an active low bi-direction tri-state line, which is used to load the contents of the data bus to the 8-bit mode register or upper/lower byte of a 16-bit DMA address register or terminal count register. In the master mode, it is used to load the data to the peripheral devices during DMA memory read cycle. CLK It is a clock frequency signal which is required for the internal operation of 8257.

RESET This signal is used to RESET the DMA controller by disabling all the

DMA channels.

Ao - A3 These are the four least significant address lines. In the slave mode, they act as an input, which selects one of the registers to be read or written. In the master mode, they are the four least significant memory address output lines generated by 8257.

CS It is an active-low chip select line. In the Slave mode, it enables the read/write operations to/from 8257. In the master mode, it disables the read/write operations to/from 8257.

A4 - A7 These are the higher nibble of the lower byte address generated by DMA in the master mode.

READY It is an active-high asynchronous input signal, which makes DMA ready by inserting wait states.

HRQ This signal is used to receive the hold request signal from the output device. In the slave mode, it is connected with a DRQ input line 8257. In Master mode, it is connected with HOLD input of the CPU.

HLDA It is the hold acknowledgement signal which indicates the DMA controller that the bus has been granted to the requesting peripheral by the CPU when it is set to 1.

MEMR It is the low memory read signal, which is used to read the data from the addressed memory locations during DMA read cycles.

MEMW It is the active-low three state signal which is used to write the data to the addressed memory location during DMA write operation.

ADST This signal is used to convert the higher byte of the memory address generated by the DMA controller into the latches.

AEN This signal is used to disable the address bus/data bus.

TC It stands for 'Terminal Count', which indicates the present DMA cycle to the present peripheral devices.

MARK The mark will be activated after each 128 cycles or integral multiples of it from the beginning. It indicates the current DMA cycle is the 128th cycle since the previous MARK output to the selected peripheral device

V cc It is the power signal which is required for the operation of the circuit.