UNIT III

External Memory Interfacing of Microcontroller system with other peripheral devices

I/O Interface

- Any application of a microprocessor system requires the transfer of data between microprocessor and external environment and also with in the microprocessor. This is known as Input/Output. There are three different ways that the data transfer can take place. They are

(1) Program controlled I/O

(2) Interrupt Program Controlled I/O

(3) Hardware controlled I/O

Interfacing and interface

- In program controlled I/O data transfer scheme the transfer of data is completely under the control of the microprocessor program. In this case an I/O operation takes place only when an I/O transfer instruction is executed.
- In an interrupt program controlled I/O an external device indicates directly to the microprocessor its readiness to transfer data by a signal at an interrupt input of the microprocessor. When microprocessor receives this signal the control is transferred to ISS (Interrupt service subroutine) which performs the data transfer.

MEMORY

- are two main types of memory.

(i) Read only memory (ROM): As the name indicates this memory is available only for reading purpose. The various types available under this category are PROM, EPROM, EEPROM which contain system software and permanent system data.

(ii) Random Access memory (RAM): This is also known as Read Write Memory. It is a volatile memory. RAM contains temporary data and software programs generally for different applications.

Interface and interfacing

- Hardware controlled I/O is also known as direct memory access DMA. In this case the data transfer takes place directly between an I/O device and memory but not through microprocessors. Microprocessor only initializes the process of data transfer by indicating the starting address and the number of words to be transferred.

Interface and interfacing

- 3 Serial I/O signals

There are 2 serial signals, i.e. SID and SOD and these signals are used for serial communication.

- SOD (Serial output data line) – the output SOD is set/reset as specified by the SIM instruction.
- SID (Serial input data line) – the data on this line is loaded into accumulator whenever a RIM instruction is executed.

External Memory Interfacing in 8051 Microcontroller

- External memory interfacing in 8051 microcontroller involves connecting external memory devices such as RAM and ROM to the microcontroller to provide additional memory space. This allows the microcontroller to execute larger and more complex programs, store more data, and perform more complex operations.
- External memory interfacing typically involves connecting the memory devices to the microcontroller through a data bus and an address bus. The data bus is used to transfer data between the microcontroller and the memory device, while the address bus is used to select a specific memory location in the memory device.

External Memory Interfacing in 8051 Microcontroller

- To interface with external memory, the 8051 microcontroller uses dedicated pins such as ALE (Address Latch Enable), PSEN (Program Store Enable), and RD (Read) and WR (Write) signals. These signals are used to control the flow of data between the microcontroller and the external memory device.
- For minimal memory applications, the 8051 has internal data and code memory. In such a position. For certain applications, this memory capacity will not be adequate. To expand the memory space of the 8051 micro-controller, we must bind external ROM/EPROM and RAM. We also understand that ROM serves as program memory and RAM serves as data memory. Let’s take a look at how the 8051 accesses these memories.

Why need External Memory Interfacing in 8051 Microcontroller ?

- External memory interfacing is necessary in the 8051 microcontroller for several reasons:
- Limited internal memory: The 8051 microcontroller has a limited amount of internal memory, including 128 bytes of RAM and 4KB of on-chip ROM. This memory may not be sufficient for some applications that require larger program memory or more data storage.
- Larger programs: For applications that require larger programs, such as complex algorithms or multiple functions, external memory interfacing can provide the necessary program memory space to store these programs.
- Data storage: Applications that require the storage of large amounts of data, such as data logging or data analysis, may require external memory interfacing to store the data.

Why need External Memory Interfacing in 8051 Microcontroller ?

- Flexibility: External memory interfacing provides greater flexibility in the design of embedded systems, allowing for customization and adaptability to meet specific application requirements.
- Cost-effective: External memory devices such as RAM and ROM are relatively inexpensive, making it cost-effective to interface them with the microcontroller instead of using more expensive on-chip memory.

Important Points to Remember in Accessing External Memory :

- In the Case of accessing external memory, All external data moves with external RAM or ROM involve the A register.
- While accessing external memory, R can address 256 bytes and DPTR can address 64 k-bytes
- MOV X instruction is used to access external RAM or 1/O addresses.

8255A PROGRAMMABLE PERIPHERAL INTERFACE (PPI)

Introduction

- In parallel communication, the data bits are transferred simultaneously on separate lines.
- Due to simultaneous transmission high data rate can be attained.
- However, results in the cost of extra lines, and as the distance increases, parallel communication is used over long distances only if high data rates are required.

 Parallel data transfer with the help of Handshake signals (e.g., Data-in-Ready, Data-in-Ack, Data-out-Ready, Data-out-Ack)

8255A PROGRAMMABLE PERIPHERAL INTERFACE (PPI)

Data transfer from I/O to Interface (Inputting Data)

1. I/O initiates the operation by putting data on the data bus and by activating Data-in-Ready.

2. Upon receiving the Data-in-Ready signal, the Interface accepts data in its Data-in-Buffer register and sends the acknowledge signal, Data-in-Ack, to the device (I/O) to indicate that the data is read by the interface.

3. In response to the acknowledge signal, the input device may drop the data and ready signal

Parallel data transfer with the help of Handshake signals (e.g., Data-in-Ready, Data-in-Ack, Data-out-Ready, Data-out-Ack)

8255A PROGRAMMABLE PERIPHERAL INTERFACE (PPI)

Data transfer from Interface to I/O(Outputting Data)

1. The interface puts data on the data bus and sends the Data-out-Ready signal to the output device.
2. Upon receiving the Data-out-Ready signal, the Output device accepts data on the data lines and sends the acknowledge signal, Data-out-Ack, to the Interface to indicate that the data is read by the output device.
3. In response to the acknowledge signal, the output device may drop the data and ready signals

Typical Parallel Communication Interface

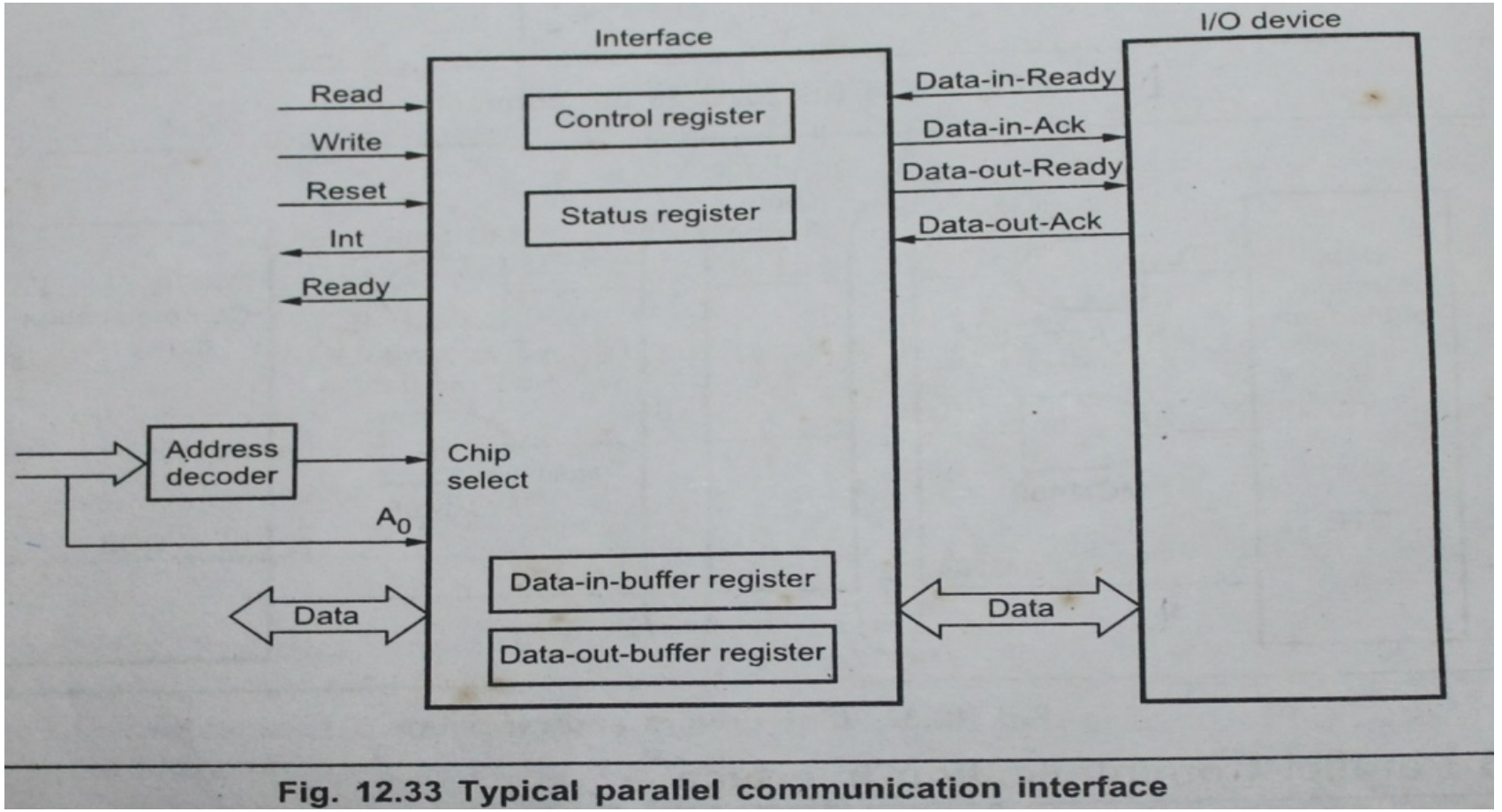


Fig. 12.33 Typical parallel communication interface

Typical Parallel Communication Interface

- The 8255A is a general-purpose programmable I/O device used for parallel data transfer. It is a Programmable Peripheral Interface (PI)
- The INTEL 8255 is a 40-pin IC having a total of 24 I/O pins. consisting of 3 numbers of 8 –bit parallel I/O ports (i.e., PORT A, PORT B, PORT C).
- The eight bits of port C can be used as individual bits or grouped in two 4-bit ports: Cupper (Cu) and CLower (CL)

Modes of 8255

The 8255 can primarily be programmed in two modes:

❖Bit Set/Reset (BSR)

❖ I/O Mode.

1. The BSR mode is used to set or reset the bits in port C.

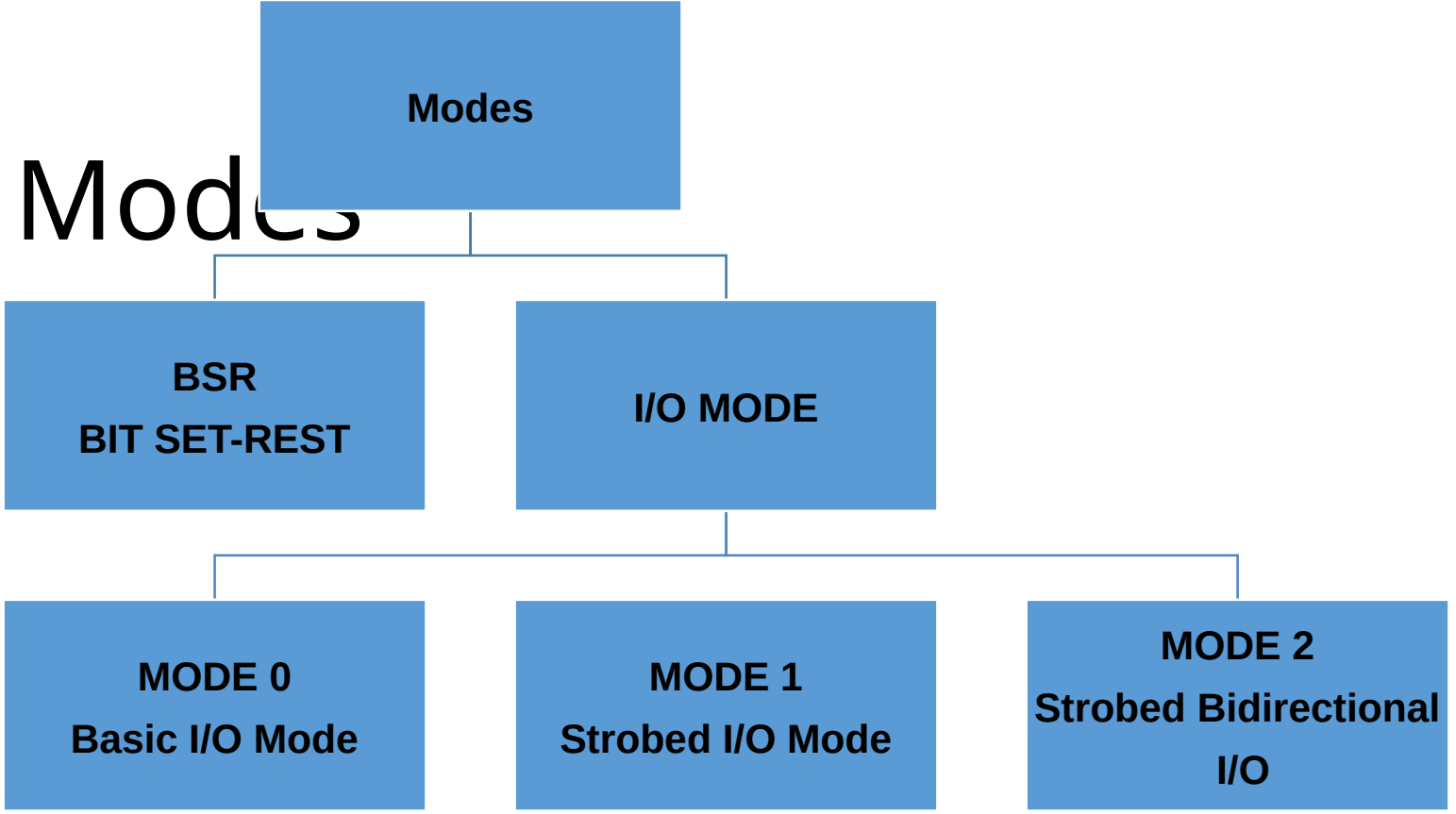
2. The I/O mode is further divided into three modes

- Mode 0: Simple Input/Output
- Mode 1: Input/Output with handshake
- Mode2: Bi-Directional I/O data transfer

Modes of 8255

- The function of I/O pins (input or output) and modes of operation of I/O ports can be programmed by writing proper control word in the control word register.
- Each bit in the control word has a specific meaning and the status of these bits decides the function and operating mode of the I/O ports.
- The 8255 primarily can primarily be programmed in two modes: Bit Set/Reset (BSR) & I/O Mode.

Modes


Modes
BSR
BIT SET-REST
I/O MODE
MODE 0
Basic I/O Mode
MODE 1
Strobed I/O Mode
MODE 2
Strobed Bidirectional
I/O

Block Diagram of 8255A

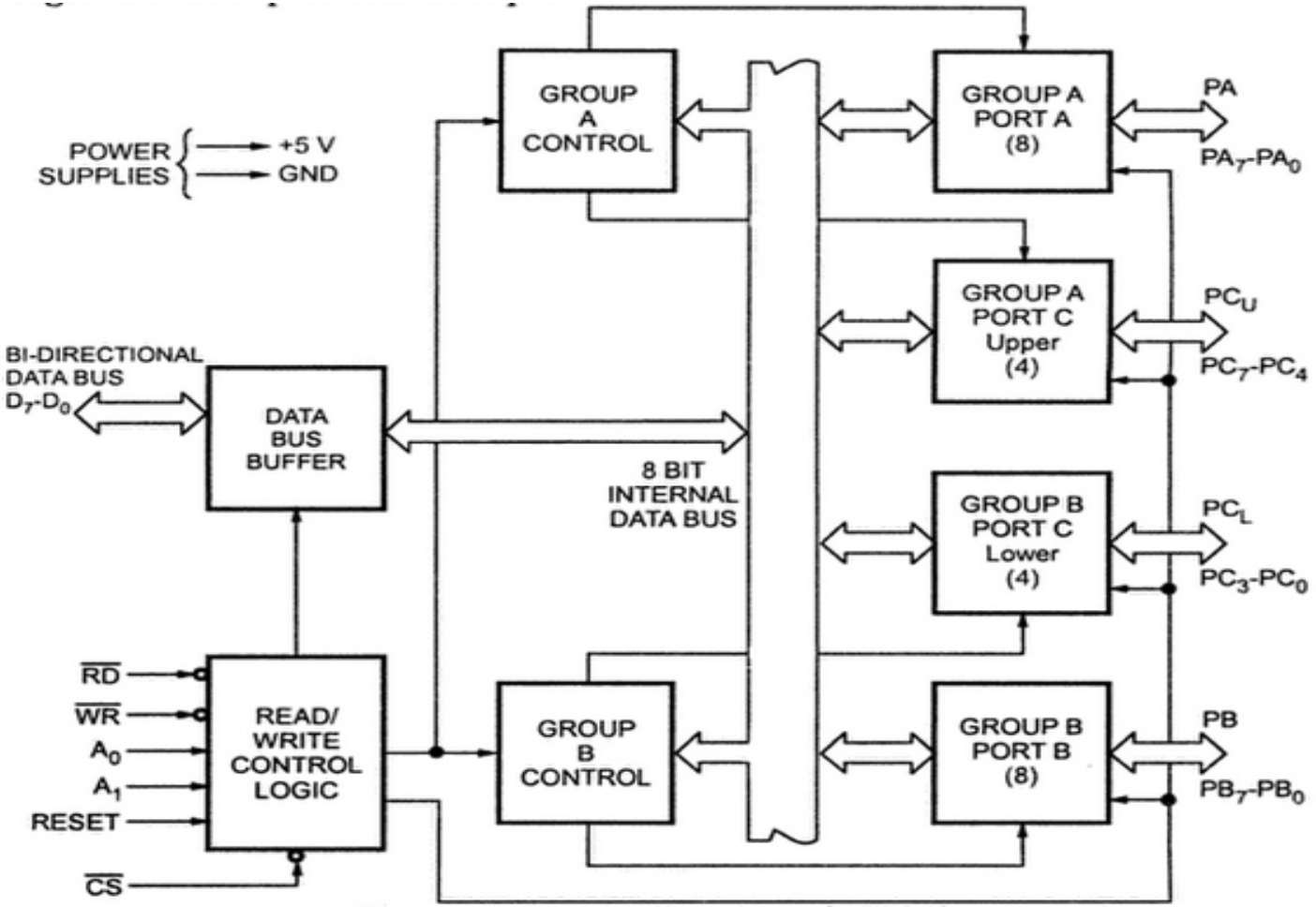
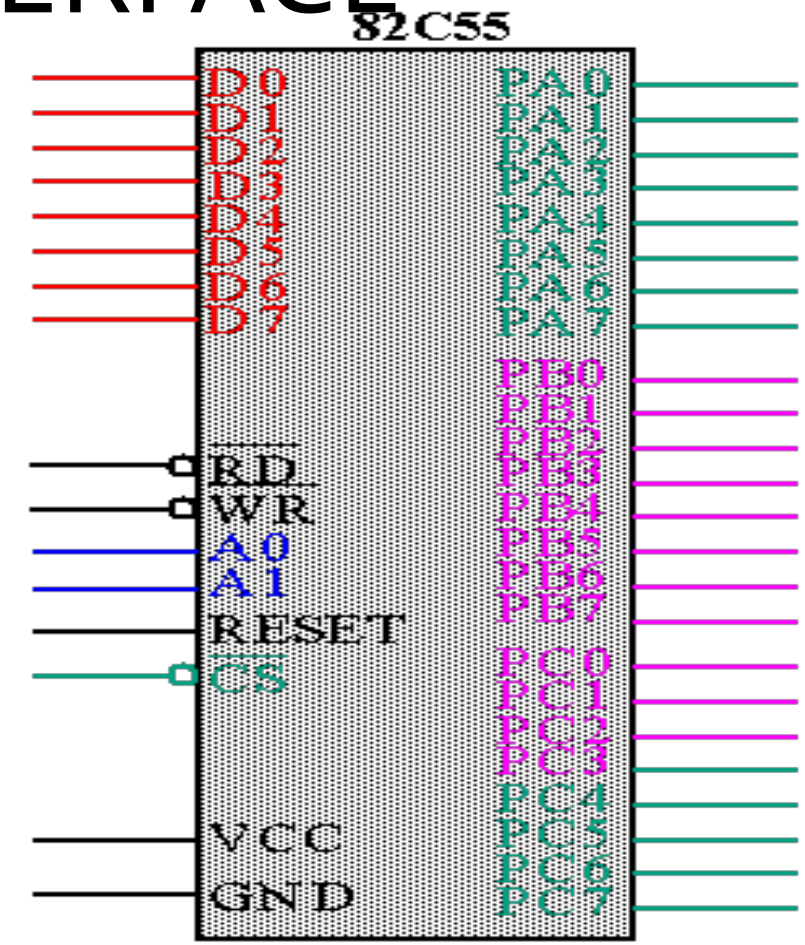


Fig. 10.2 Block diagram of 8255A

FUNCTION OF BLOCKS

BLOCK	FUNCTION OF BLOCK
Data Bus Buffer	It is used to interface the internal data bus of 8255 to the system data bus by reading and writing operations.
Read/write Control logic	It accepts the input from the address bus and issues commands to the individual group blocks. also issues appropriate enabling signals to access the required data/control words/status words.
Port A	It can be programmed in three modes Mode0, Mode1 and Mode2.
Port B	It can be programmed in three modes Mode0 and Mode1.
Port C	It can be programmed for Bit Set/reset operation.

8255 PROGRAMMABLE PERIPHERAL INTERFACE



Group A

Port A (PA7-PA0) and upper half of port C (PC7 - PC4)

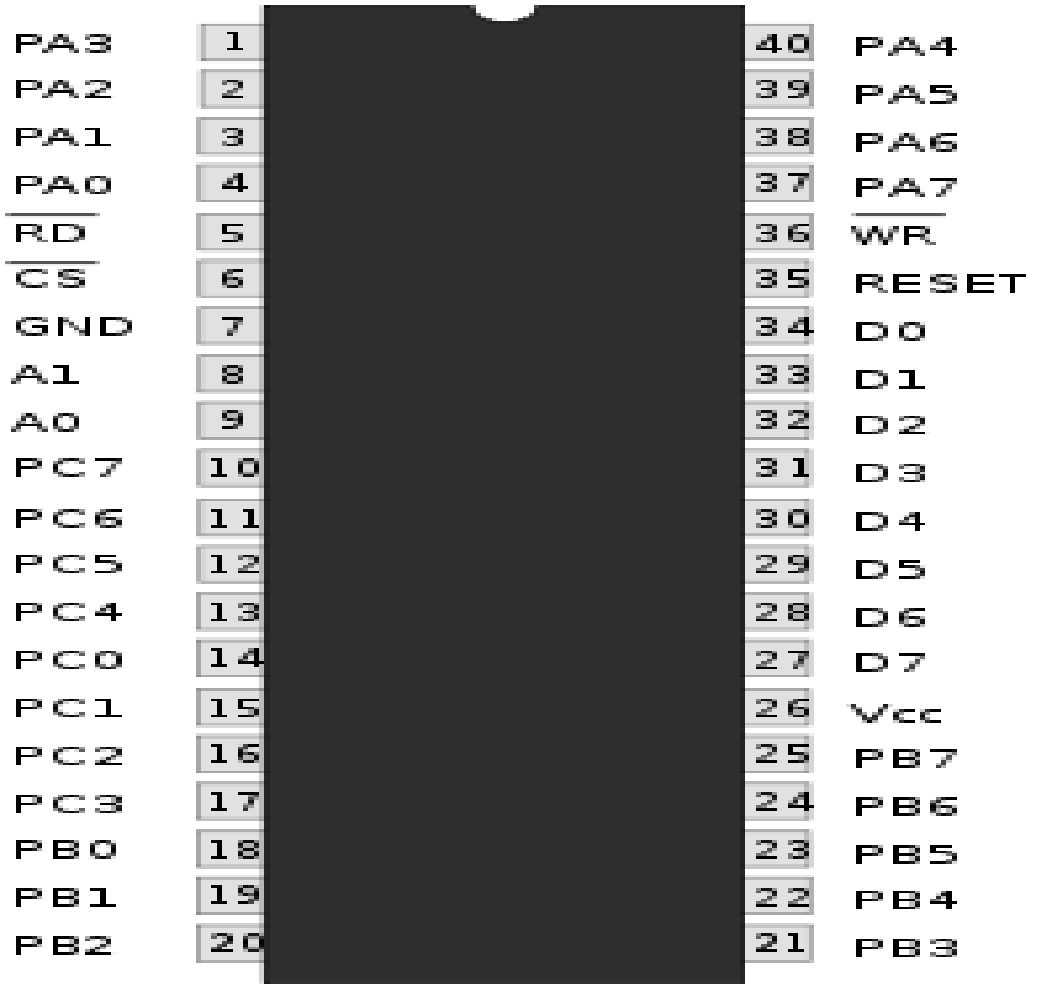
Group B

Port B (PB7-PB0) and lower half of port C (PC3 - PC0)

I/O Port Assignments

A_1	A_0	Function
0	0	Port A
0	1	Port B
1	0	Port C
1	1	Command Register

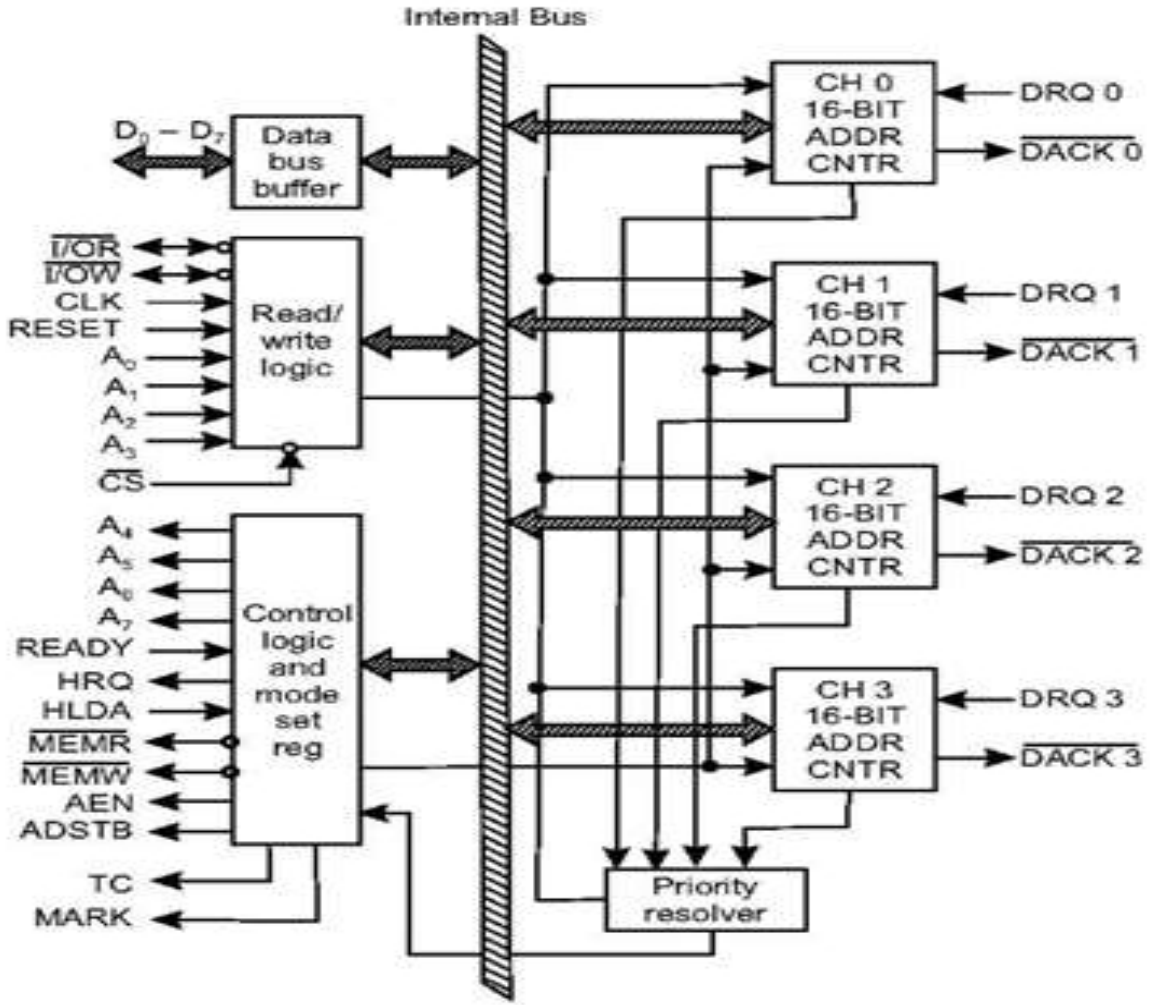
8255 PROGRAMMABLE PERIPHERAL INTERFACE



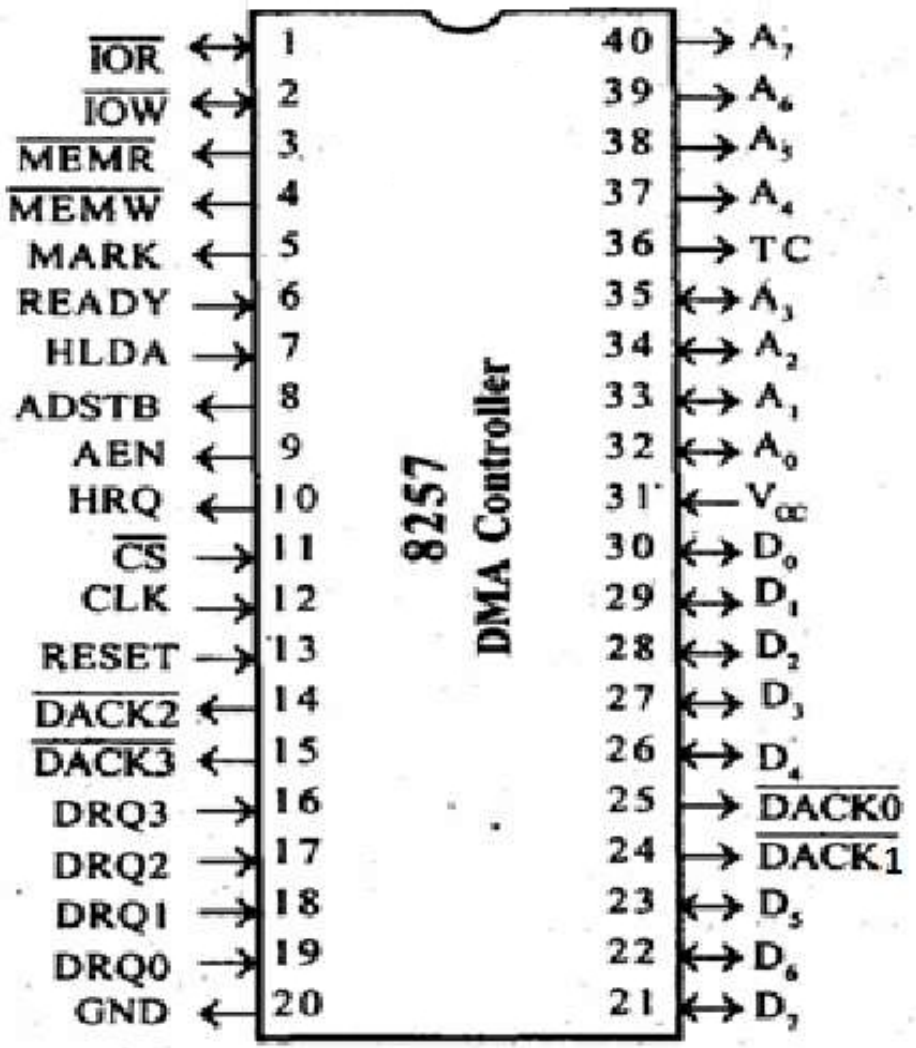
8255 PROGRAMMABLE PERIPHERAL INTERFACE

FUNCTION OF PIN
These are bidirectional, tri-state data bus lines are connected to the system data bus. They are used to transfer data and control word from microprocessor (8085) to 8255 or receive data or status word from 8255 to the 8085.
These are 8 Bit bidirectional I/O pins used to send data to output device and to receive data from input device. It functions as an 8 Bit data output latch/buffer when used in output mode and as an 8 Bit data input latch/buffer when used in input mode.
These are 8 Bit bidirectional I/O pins used to send data to output device and to receive data from input device. It functions as an 8 Bit data output latch/buffer when used in output mode and as an 8 Bit data input latch/buffer when used in input mode.

8257 Architecture



8257 Pin Description



8257 Pin Description

- DRQ0DRQ3

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.

8257 Pin Description

- 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.

8257 Pin Description

- 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.

8257 Pin Description

- 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.

- AEN

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

8257 Pin Description

- 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.

8257 Pin Description

- 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.

8257 Pin Description

- 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.

8257 Pin Description

- 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.

- Vcc

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