Unit → 6

Input/Output Interfaces

Date.
Page No.

⊛ Parallel Communication:

It is a method of conveying multiple binary digits (bits) simultaneously i.e all bits of word are transferred at a time. It is faster. Hardware requirement is complex. Eg. 8255A PPI.

Parallel communication is needed to send large amount of data. It is used when the data being sent is time-sensative and it is used when the data needs to be sent quickly. A scenario where parallel transmission is used to send data is vdo streaming. When a video is streamed to a viewer, bits need to be received quickly to prevent a video pausing or buffering.

Parallel communication is widely used within integrated circuits, in peripheral buses and in a memory devices such as RAM. It is used for short distance communication.

⊛ Serial Communication:

It is a method of conveying a single bit at a time i.e, only one bit of a word is transmitted at a time. It is slower. Hardware requirement is simple. Example: RS-232C.

Serial transmission is normally used for long distance data transfer. It is also used in cases where the amount of data being sent is relatively small. It ensures that data be integrity is maintained as it transmits the data bits in a specific order, one after another. In this way, data ~~are~~ bits are received en-sync with one another. In serial transmission data is sent bit by bit from one computer to another in two directions. Each bit has a clock pulse rate.

Date.
Page No.

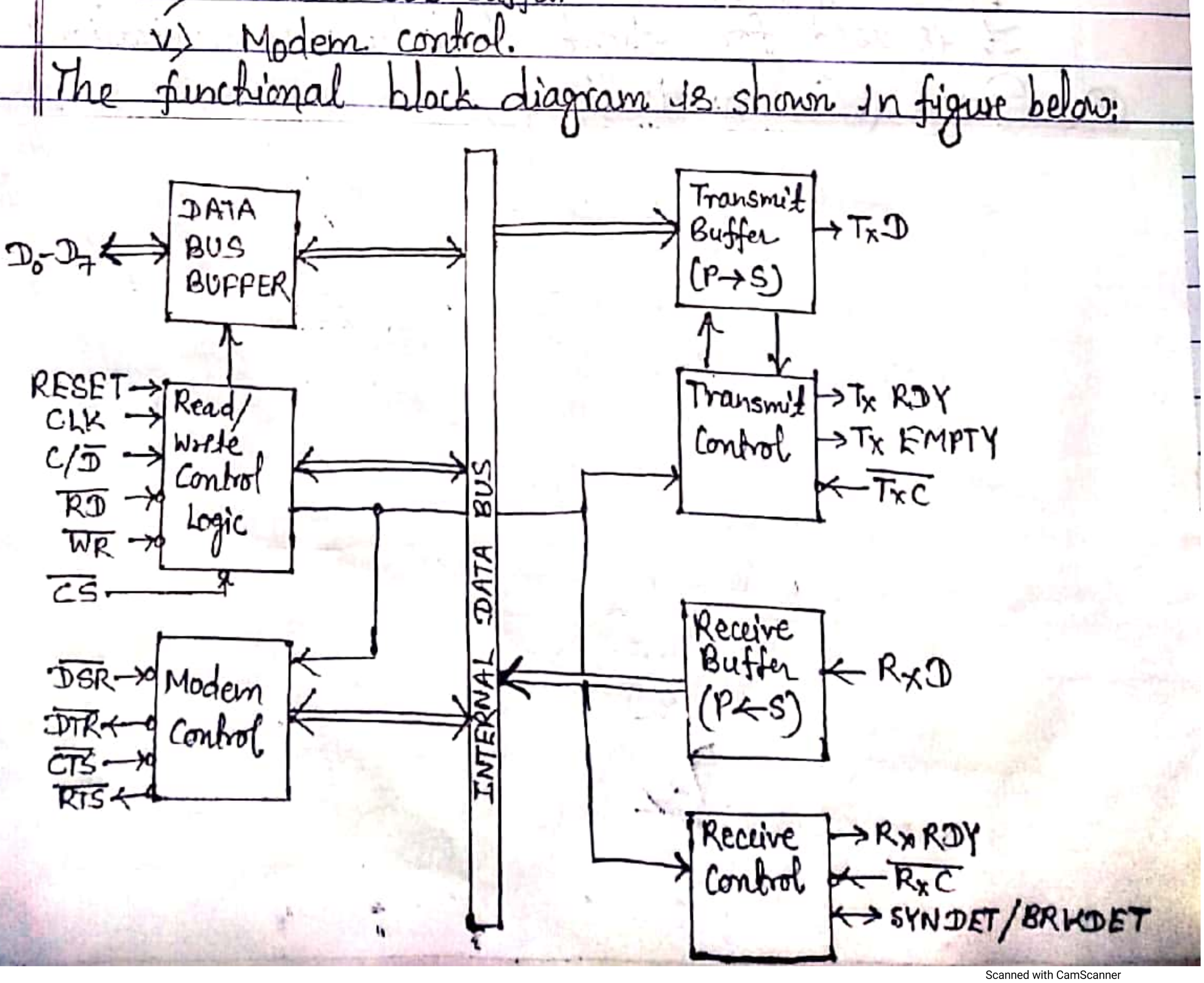
⊛ Introduction to Programmable Communication Interface 8251A

The 8251A is a programmable serial communication interface chip designed for synchronous and asynchronous serial data communication. It supports the serial transmission of data. It is packed in a 28 pin DIP. It is also called USART (Universal Synchronous Asynchronous Receiver Transmitter).

Block Diagram: The functional block diagram of 8251A consists of five sections. They are:-

i) Read/Write control logic
ii) Transmitter
iii) Receiver
iv) Data bus buffer.
v) Modem control.

The functional block diagram is shown in figure below:



Date.
Page No.

⊛ Basic concept of Synchronous and Asynchronous Modes:

Synchronous and Asynchronous modes are the types of serial data transfer which are described as follows:

i) Synchronous

→ Both transmitter and receiver are synchronized by same clock pulse.
→ It is also called clock-oriented data transmission.
→ Speed: >20 Kbps.
→ Always implemented with hardware.

ii) Asynchronous

→ Both transmitter and receiver are synchronized by seperate clock pulse.
→ It is also called character-oriented data transmission.
→ speed: <20 Kbps.
→ Always implemented with hardware and software.

⊛ Modes of serial data transfer.

i) Simplex mode
→ Data travel in only one direction. Eg. from computer to printer.

ii) Half duplex mode
→ Data travel in both directions but not at the same time.

iii) Full duplex mode
→ Data travel in both directions at the same time.

Date.
Page No.

Simple I/O, Strobe I/O, Single handshake I/O, Double handshake I/O

Simple I/O:

When we need to output data to simple display device, such as LED, all we have to do is connect the input of the LED buffer on an output port pin and ouput the logical level required to turn on the light.

Fig. Simple I/O

The timing waveform represents the situation. The crossed lines on the waveform represent the time at which a new data type becomes valid on the output lines of the port. The absence of other waveforms indicates that this output operation is not directly dependent on any other signal.

Strobe I/O

In many applications, valid data is present on an external device only at a certain time, so it must be read at that time. E.g. the ASCII-encoded keyboard. When a key is pressed, it sends ASCII code for the pressed key on eight parallel data lines and then sends out a strobe signal on another line to indicate that valid data is present on the eight data lines.

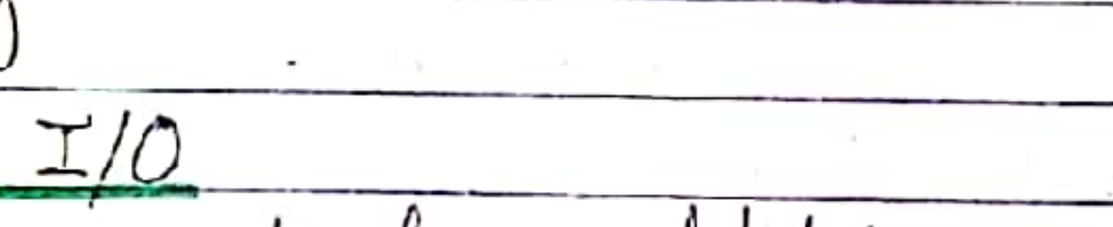


fig. Strobe I/O

This timing waveform represents strobe I/O. For low rates of data transfer, such as from a keyboard to a MP, a single strobe transfer works well.

Date.
Page No.

Single handshake I/O

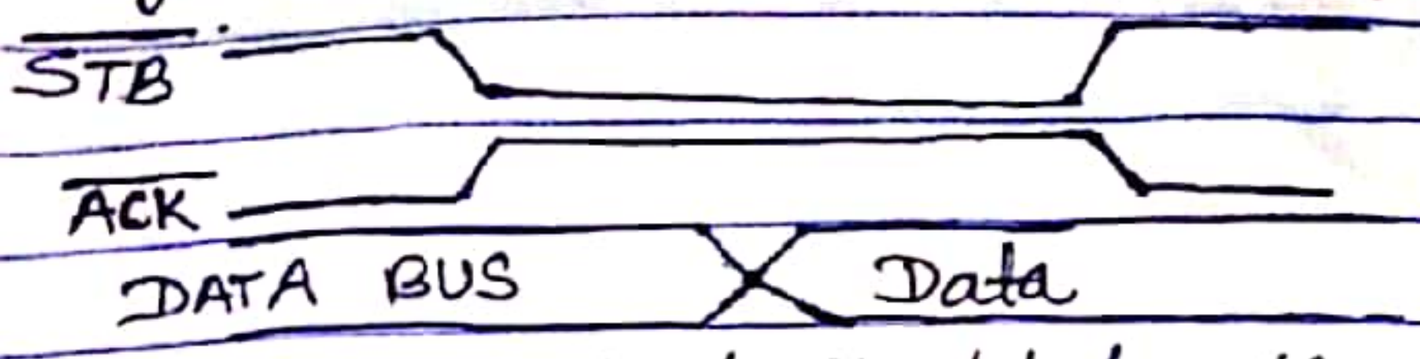


fig. Single Handshake I/O

It shows the timing waveform for a handshake data transfer from a peripheral device to a MP. The peripheral outputs some parallel data and sends an STB signal to the MP. The MP detects the asserted STB signal on a polled or interrupts basis and reads in the byte of data. Then the MP sends ACK (acknowledge) signal to peripheral to indicate that the data has been read and that the peripheral can send next byte of data.

Double handshake I/O

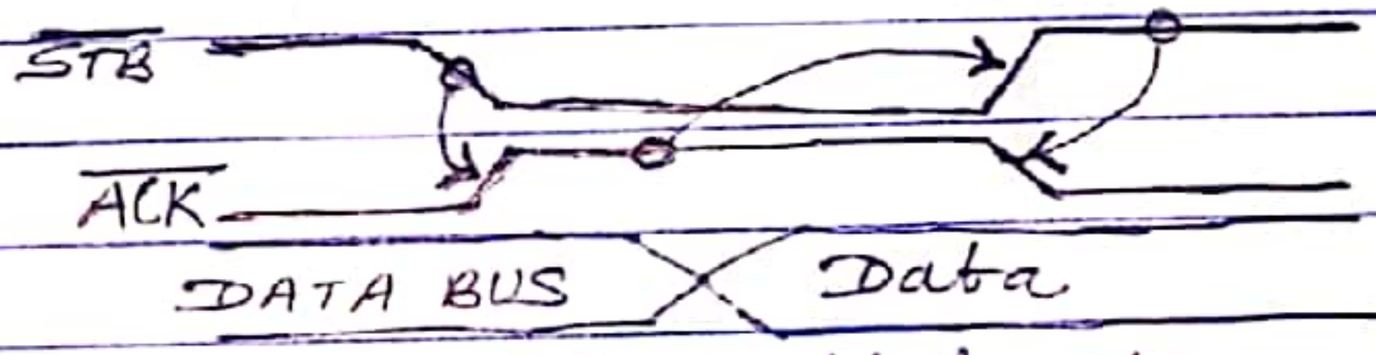


Fig. Double Handshake I/O

A double handshake is used for data transfer where more coordination is required between the sending system and receiving system. The sending (peripheral) device whether it is ready or not for data reception. The receiving system raises its ACK line high to indicate that it is ready. The peripheral device then sends the byte of data and raises its STB line high to assure that the valid data is available for the receiving device (MP). When MP reads the data, it drops its ACK line low to indicate that it has received the data and requests the sending system to send next byte of data.

Date.
Page No.

8255A PPI (Programmable Peripheral Interface).

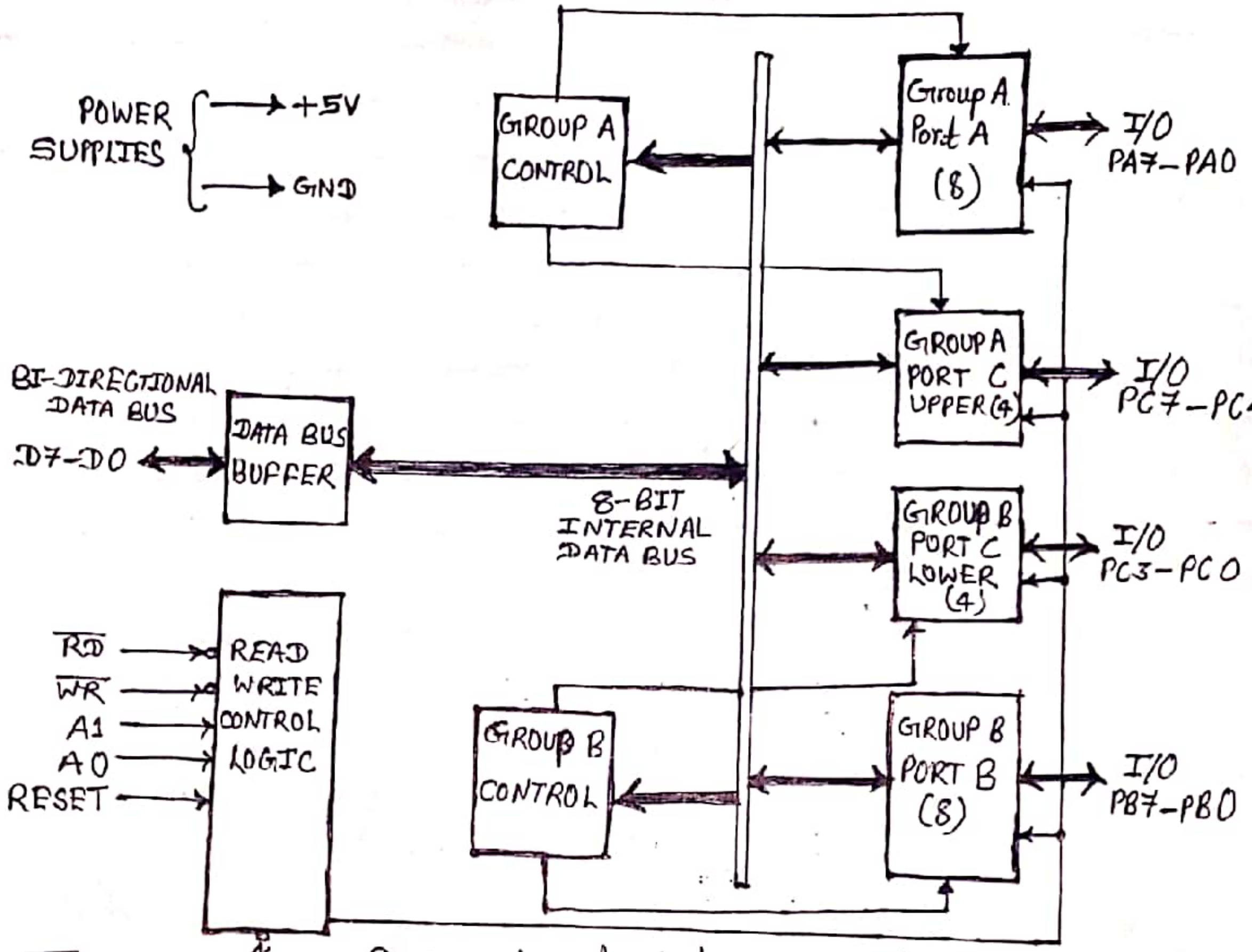


Fig. Internal Block Diagram of 8255A PPI

8225A is 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. It is flexible, versatile and economical (when multiple I/O ports are required), but somewhat complex. It is an important general purpose I/O device that can be used with almost any microprocessor.

Date.
Page No.

The 8255A has input output pins that can be grouped primarily in two 8 bits parallel ports: A and B, with the remaing 8 bits of port C can be used as individual bits or be grouped in two four bits ports: C upper (C_U) and C lower (C_L) as in figure (a). The functions of these ports are defined by writing a control word in the control registers.

Working:

Data Bus Buffer → This three-state bi-directional 8-bit buffer is used to interface the 8255 to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic → The function of this block is to manage all the internal and external transfers of both Data and Control or Status words.

CS (Select Chip) → Low input on this pin enables the communication between the 8255 and the CPU.

RD (Read) → It is also low input enabled. It allows CPU to read from the 8255.

WR (Write) → It is also low input enabled. It allows CPU to write data into the 8255.

A0 and A1 → This indicate port select 0 and port select 1. These help to select the selection of one of the three ports or the control word register.

RESET → A high on this input initializes the control register to 9Bh and all ports (A, B, C) are set to the input mode.

A1	A0	Selection
0	0	PORT A
0	1	PORT B
1	0	PORT C
1	1	CONTROL REGISTER

Group A and Group B Controls

The control word contains ~~three 8 bit~~ information such as "mode", "bit set", "bit reset", etc, that initializes ~~that in~~ the functional configuration of the 8255. Each of the control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Port A, B and C.

PORT A → One 8-bit data output latch/buffer and one 8-bit data input latch. Both "pull-up" and "pull-down" bus-hold devices are present on Port A.

PORT B → One 8-bit data input/output latch/buffer and one 8-bit data input buffer.

PORT C → One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under mode control.

Date.
Page No.

Modes of Operation:

The 8255A is primarily operated in two modes: I/O (input-output) mode and the BSR (Bit-Set-Reset) mode. The BSR mode is used to set or reset the bits in port C. The I/O mode is further divided into three modes: mode 0, mode 1 and mode 2. In mode 0, all ports function as simple I/O ports. Mode 1 is a handshake mode and in mode 2, port A can be set up for bidirectional data transfer using handshake signal from port C, and port B can be set up either in mode 0 or mode 1.

Control Word

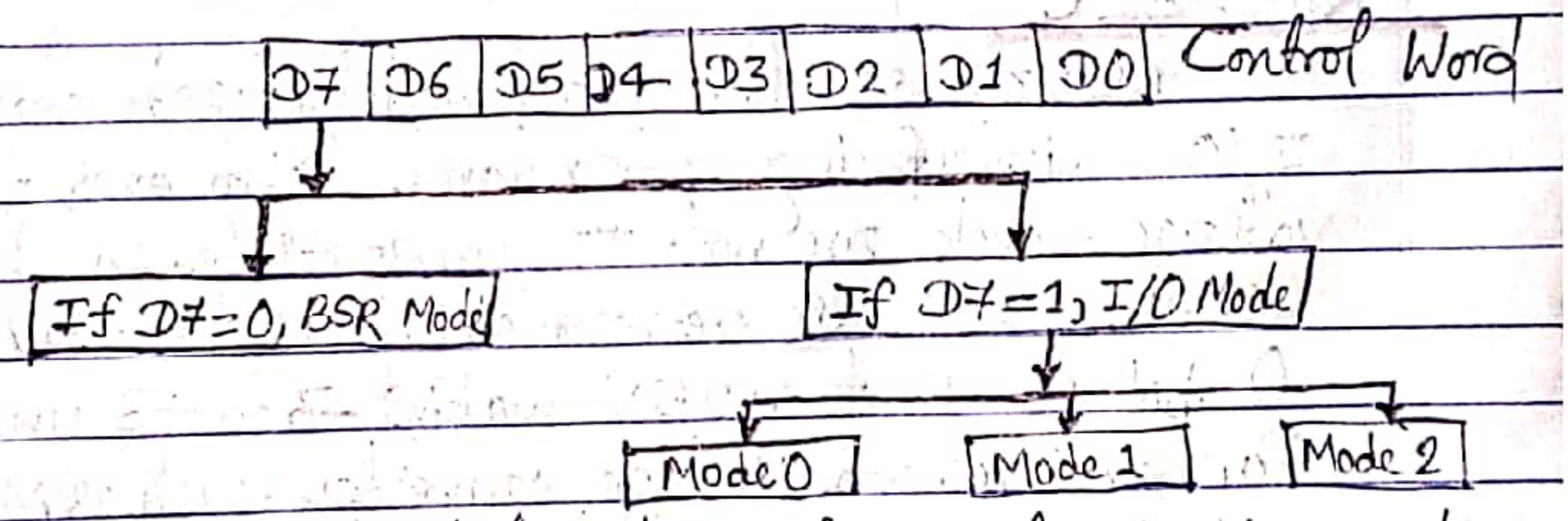


fig. Control word ~~specific~~ specifying various modes

When D7=0, BSR mode

→ For Port C.
→ No effect on I/O mode and functions of port A and B.
→ Individual bits of port C can be used for applications such as ON/OFF switch.

When D7=1, I/O mode

i) Mode 0 → Simple I/O interfacing for port A, B, C.
ii) Mode 1 → Interfacing with handshake signals for port A & B.
iii) Mode 2 → Bidirectional I/O interfacing for port A.
→ Port B either in mode 0 or mode 1.

Date.
Page No.

The content of control register is called control word specifying an input output functions for each port. The register can be accessed to write a control word when A_0 and A_1 are at logic 1. The register is not accessible for read operation. Bit D_7 of the control register specifies either I/O functions or Bit Set-Reset function. If bit $D_7=1$, bit D_6-D_0 determine I/O function in various modes. If bit $D_7=0$, port C operates in Bit Set-Reset mode. The BSR control word does not affect the function of port A and port B.

RS-232 C

RS 232 is the most widely used serial I/O interfacing standard. However the I/O voltage levels are not TTL compatible. In the RS 232, a 1 is represented by −3 to −25V while 0 bit is +3 to +25V making −3 to +3 undefined. For this reason voltage converter such as MC1488 and MC 1489 are used to convert the TTL logic levels to the RS232 voltage levels and vice versa.

RS 232 stands for Recommend Standard number 232. The serial ports on most computers use a subset of the RS-232C standard. C is the latest version of RS-232. The full RS-232C standard specifies a 25-pin "D" connector of which 22 pins are used.

Date.
Page No.

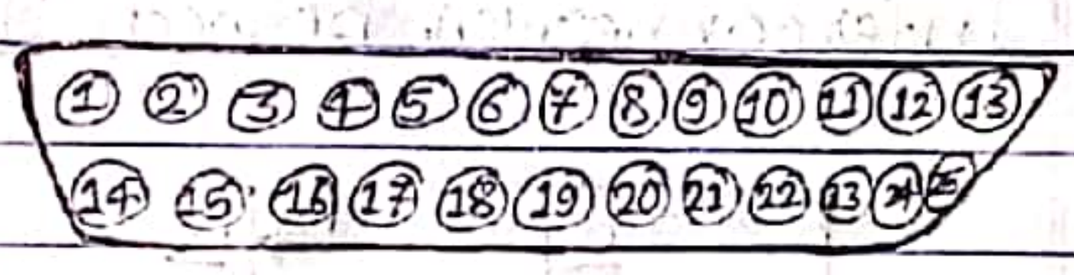
DCE and DTE Devices:

DTE stands for Data Terminal Equipment, and DCE stands for Data Communications Equipment. These terms are used to indicate the pin-out for the connectors on a device and the direction of the signals on the pins. Computer used in our daily life are DTE device, while most other devices are usually DCE devices.

The RS-232 standard states that DTE devices use a 25-pin male connector, and DCE devices use a 25-pin female connector. So we can connect a DTE device to a DCE using a straight pin-for-pin connection.

25 Pin Connector on a DTE device (PC connection).

Male RS232 DB25

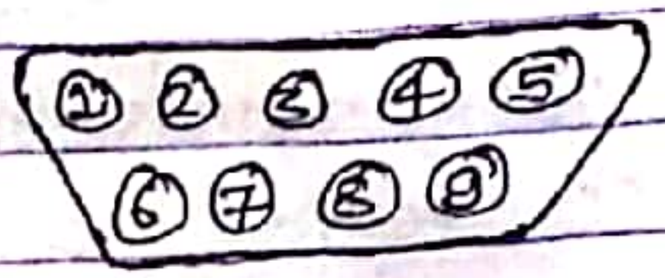


PIN Number	Direction of signal.
1	Protective Ground
2	Transmitted Data (TD) Outgoing Data (from DTE to a DCE)
3	Received Data (RD) Incoming Data (from DCE to a DTE)
4	Request to send (RTS) Outgoing flow control signal controlled by DTE
5	Clear to send (CTS) Incoming flow control signal controlled by DCE.
6	Data Set Ready (DSR) Incoming handshaking signal controlled by DCE.
7	Signal ground ~~connection~~ Common reference voltage.
8	Carrier Detect (CD) Incoming signal from a modem
20	Data Terminal Ready (DTR) Outgoing handshaking signal controlled by DTE.
22	Ring Indicator (RI) Incoming signal from a modem.

Date.
Page No.

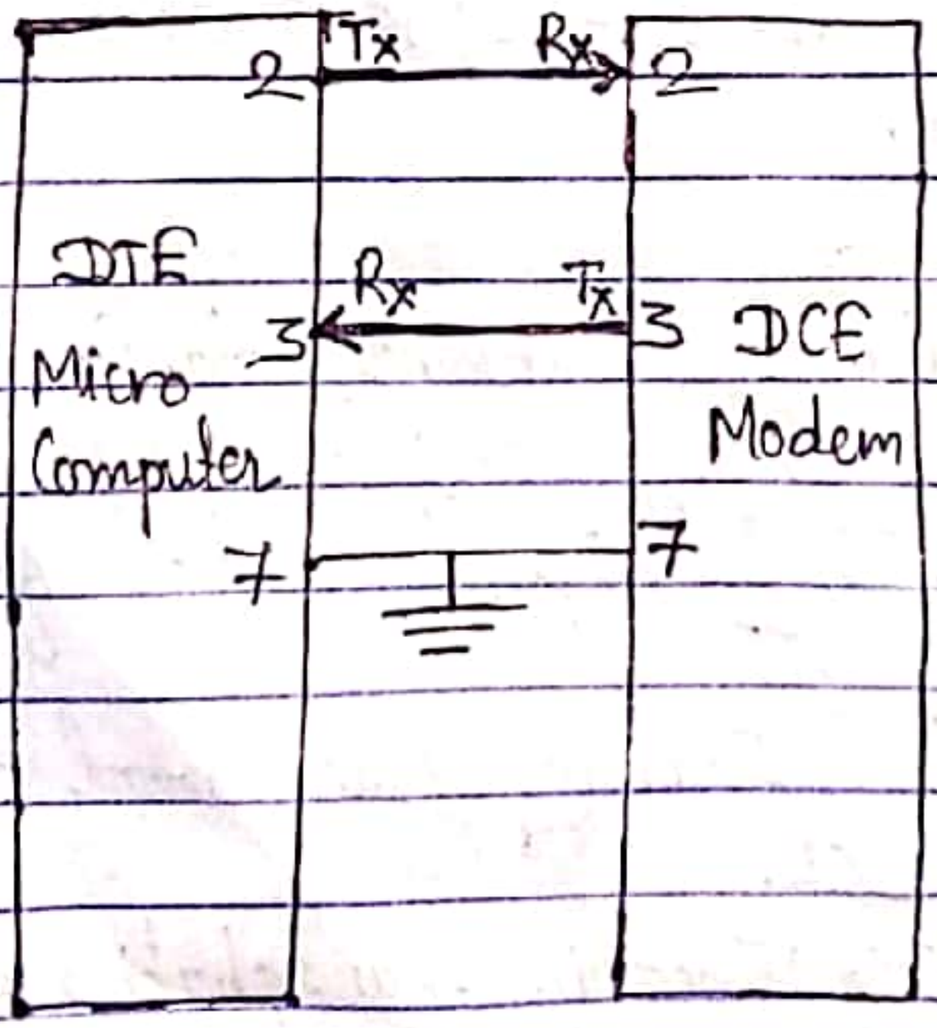
9 Pin Connector on a DTE device (PC connection)

Male RS232 DB9



Pin Number	Direction of signal:
1	Carrier Detect (CD) (from DCE) Incoming signal from a modem.
2	Received Data (RD) Incoming data from a DCE.
3	Transmitted Data (TD) Outgoing data to a DCE.
4	Data Terminal Ready (DTR) Outgoing handshake signal.
5	Signal Ground Common reference voltage.
6	Data Set Ready (DSR) Incoming handshaking signal.
7	Request to Send (RTS) Outgoing flow control signal.
8	Clear ~~to send~~ To Send (CTS) Incoming flow control signal.
9	Ring Indicator (RI) (from DCE) Incoming signal from a modem.

Interconnection between DTE-DTE and DTE-DCE



fig@. DTE to DCE

Date.
Page No.

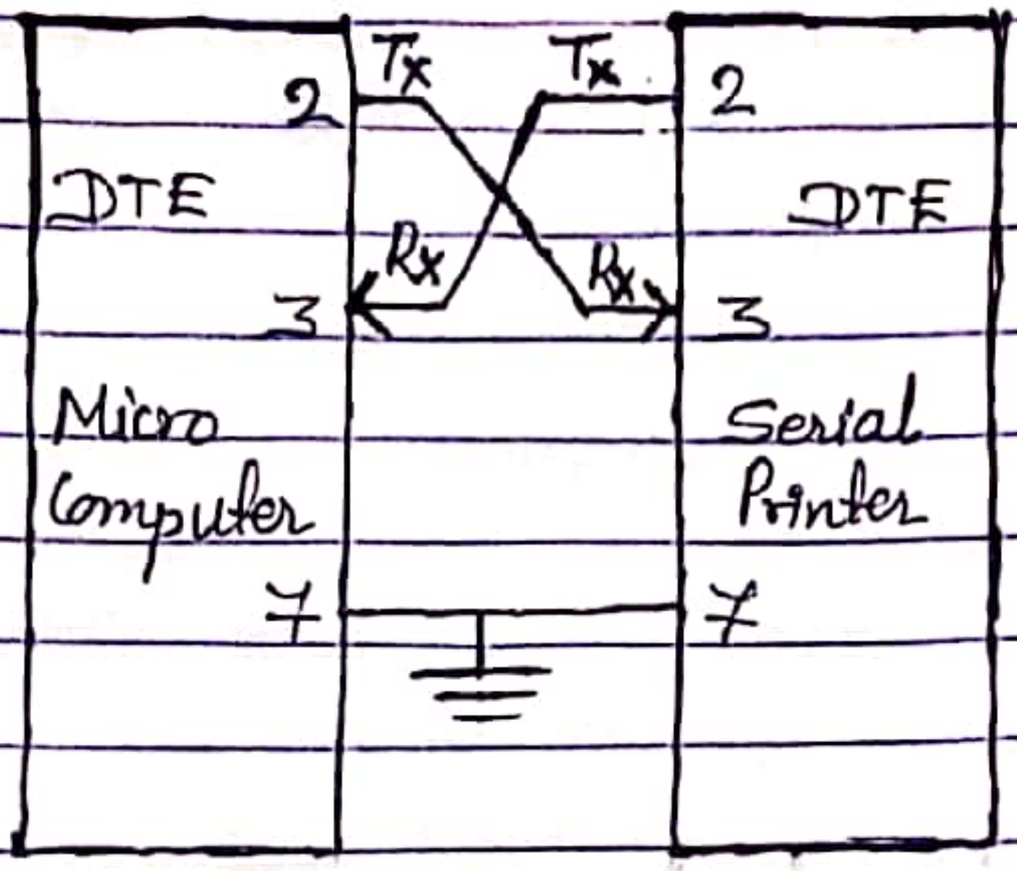


fig (b). DTE to DTE (null modem)

- The minimum interface between a computer and a peripheral requires 3 lines: pins 2, 3 and 7 as shown in figure.
- These lines are defined in relation to DTE; the terminal transmits on pin 2 and receives on pin 3. On the other hand, the DCE transmits on pin 3 and receives on pin 2.
- To remain compatible with the defined signals of RS-232C, the RS-232C cable must be reconfigured as shown in fig (b).
- In figure (a), the microprocessor is defined as a DTE, and it can be connected to the modem defined as a DCE, without any modification in RS-232C cable.
- When it is connected to printers, the transmitted and received lines must be crossed as shown in fig (b). This is known as Null modem connection.