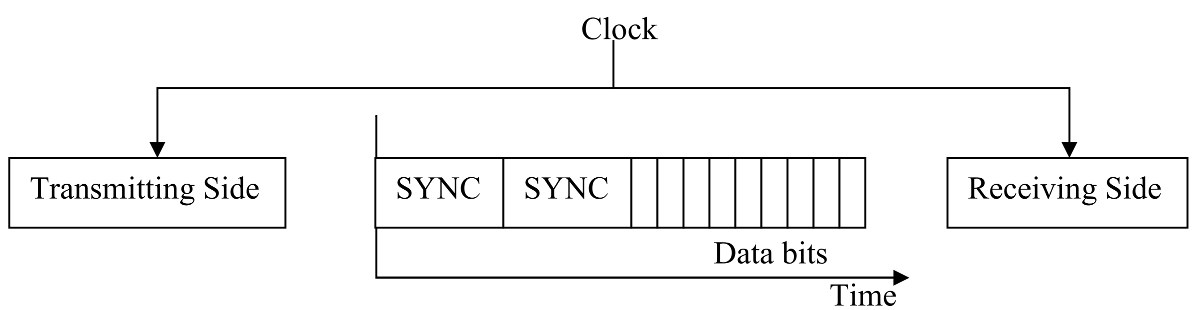
Serial Data Transmission

- In serial I/O, 8 bit parallel data is converted into a stream of 8 serial bits, known as parallel to serial conversion. After conversion, one bit at a time is transmitted over a single line. In general, serial I/O is the most commonly used than parallel I/O.
- Types:
  1. Synchronous serial data transmission
  2. Asynchronous serial data transmission

1. Synchronous Serial Data Transmission

- Data transmission is based on clock pulse.
- Transmitting and receiving sides are synchronized by means of one or two SYNC characters.
- After SYNC characters are sent, data are sent bitwise one after another continuously and the receiving unit interprets the data.
- Data unit generally consists of 7 data bits and 1 parity bit.
- The synchronous receiver waits in “hunt mode” while looking for the data. As soon as it matches one or two SYNC characters, the receiver starts interpreting the data.

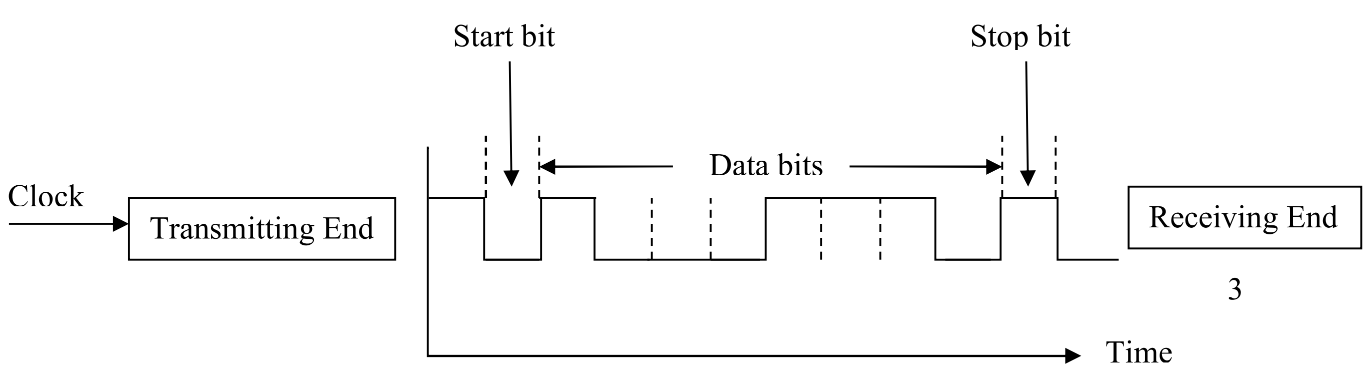
SYNC	Data bits	Parity
10110101	XXXXXXX	X



2. Asynchronous Serial Data Transmission

- Synchronization between transmitting and receiving end is not required.
- Each data to be sent must be in a particular format, i.e. each data unit contains a START bit that indicates the beginning of the data and one or more STOP bits that indicate the end of the data.

Start bit	Data bits	Stop bit
0	10001110	1



- General format for asynchronous serial data contains the following information:
    - A START bit (Low) that indicates the beginning of the data to be transferred.
    - 8 bit Data
    - An optional PARITY bit
    - One or two STOP bit

Comparison of Synchronous and Asynchronous Serial Data Transmission

Parameter	Synchronous	Asynchronous
1. Data format	Group of characters	One character at a time
2. Speed	High (≥20kbps)	Low (≤20kbps)
3. Framing information	Synchronous characters (SYNC) are sent with each data group	START and STOP bits with each data character
4. Implementation	Hardware	Both hardware and software
5. Data direction	Simplex, half duplex and full duplex	Simplex, half duplex and full duplex

Methods of Parallel Data Transfer

1. **Simple Input and Output**
    - Data is always present and ready to be transferred.
    - When we need to send/receive data we connect data lines and send/receive data.
    - Example: LED
2. **Strobe Input and Output**
    - When an external device desires to send a data to a µp, it places the data in data bus and asserts its $\overline{STB}$ signal low to indicate presence of data byte.
    - Suitable for low data rates such as keyboard to PC.
    - Problem: lacks an acknowledge signal to indicate receiving of data.
3. **Single handshake Input and Output**
    - Peripheral outputs some data, asserts its $\overline{STB}$ line low to indicate presence of data.
    - µp reads the data and acknowledges through ACK signal.
    - Sender does not send next character until first is received.
    - Parallel printer
4. **Double handshake Input and Output**
    - Used for better coordination
    - Sending device asserts its $\overline{STB}$ line low to indicate a valid data and says "Are you ready".
    - Receiver raises its ACK line to indicate "I am ready".
    - Sender places the data on the data bus and raises its $\overline{STB}$ indicating "Here is the data"

- Receiver receives the data and lowers its ACK to indicate “Data received, waiting for the next data”

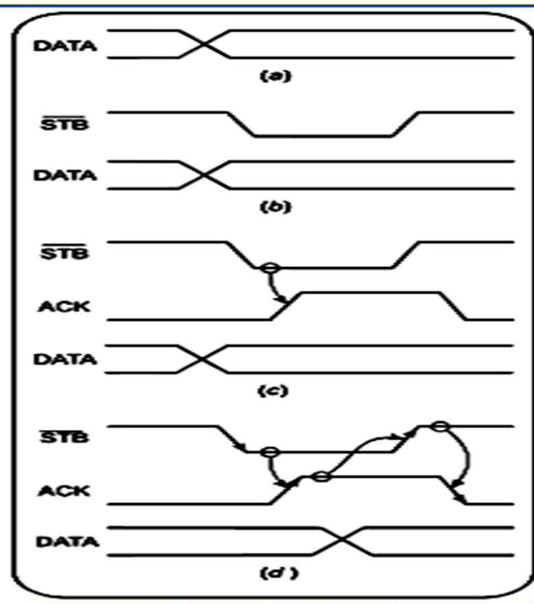


Fig. 10.1 Parallel data transfer. (a) Simple output. (b) Simple strobe I/O. (c) Single handshake I/O. (d) Double handshake I/O.

RS232C Standard

- Provides signal and handshaking standards between DTE (Data Terminal Equipment) such as computer terminal and DCE (Data Communication Equipment) such as modem.
- RS232C specifies 25 signal pins. These signal pins are divided into four groups: Data, Control, Ground and Timing signal
- Voltage level: +3V to +15V is logic 0 (+25V on no load) and -3V to -15V is logic 1 (-25V on no load)
- The RS232C standard specifies a 25 pin "D" connector (DB25) of which 22 pins are used. Most of these pins are not needed for normal PC communications, and indeed, most new PCs are equipped with male D type connectors having only 9 pins (DB9).
- Voltage level of RS232C and TTL standards are different, so conversion must be done before connecting DTE and DCE.
    - MC1488 converts TTL to RS232.
    - MC 1489 converts RS232 to TTL.
    - MAX232 converts both ways.

- Signal pins:

25 Pins	9 Pins	Signal	Direction	Description
1	-	-	-	Protective Ground
2	3	TXD	From DTE to DCE	Transmitted Data
3	2	RXD	From DCE to DTE	Received Data

4	7	RTS	From DTE to DCE	Request To Send
5	8	CTS	From DCE to DTE	Clear To Send
6	6	DSR	From DCE to DTE	Data Set Ready
7	5	GND	-	Ground (Common reference voltage)
8	1	DCD	From DCE to DTE	Data Carrier Detect
20	4	DTR	From DTE to DCE	Data Terminal Ready
22	9	RI	From DCE to DTE	Ring Indicator

- TXD: Data is transmitted from DTE through this line. DCE receives data through TXD.
- RXD: Data is transmitted from DCE through this line. DTE receives data through RXD.
- RTS: When DTE has data to transmit, it informs DCE by transmitting RTS signal.
- CTS: In response to RTS, if DCE is ready, then it signals DTE that DTE can begin data transfer.
- DTR: After DTE is turned on, it sends out DTR signal to indicate that it is ready for communication.
- DSR: After DCE is turned on, it sends out DSR signal to indicate that it is ready to establish the communication with DTE.
- DCD: Used by modem to signal that it has a made a connection with another modem, or has detected a carrier tone.
- RI: A modem toggles the state of this line when an incoming call rings your phone.
- The DCD and RI lines are only available in connections to a modem. Because most modems transmit status information to a PC when either a carrier signal is detected (i.e. when a connection is made to another modem) or when the line is ringing, these two lines are rarely used.

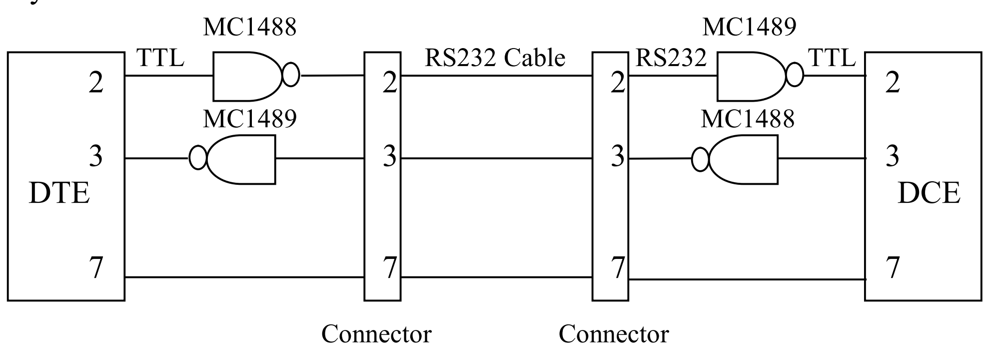


Figure: Connection of DTE and DCE with RS232 interface

Assignment: Differentiate RS423A and RS422A

Assignment: BISYNC Protocol