UNIT-2 - EMBEDDED NETWORKING

PART-A(2 marks)

1. Differentiate synchronous communication and Iso-synchronous communication.

Synchronous communication: When a byte or a frame of the data is received or transmitted at constant time intervals with uniform phase difference, the communication is called synchronous communication.

Iso-synchronous communication: Iso-synchronous communication is a special case when the maximum time interval can be varied.

2. What are the two characteristics of synchronous communication?

- Bytes maintain a constant phase difference
- The clock is not always implicit to the synchronous data receiver.

3. What are the three ways of communication for a device?

- Iso-synchronous communication
- Synchronous communication
- Asynchronous communication

4. Define half-duplex communication and full duplex communication.

Half-duplex communication: Transmission occurs in both the direction, but not simultaneously.

Full duplex communication: Transmission occurs in both the direction, simultaneously

6. Expand a) SPI b) SCI

SPI—**S**erial **P**eripheral **I**nterface, **SCI**—**S**erial **C**ommunication **I**nterface

7. What is I²C?

I²C is a serial bus for interconnecting ICs .It has a start bit and a stop bit like an UART. It has seven fields for start,7 bit address, defining a read or a write, defining byte as acknowledging byte, data byte, NACK and end.

8. What is a CAN bus? Where is it used?

CAN is a serial bus for interconnecting a central Control network. It is mostly used in automobiles. It has fields for bus arbitration bits, control bits for address and data length data bits, CRC check bits, acknowledgement bits and ending bits.

9. What is meant by status flag?

Status flag is the hardware signal to be set when the timer reaches zeros.

10. State the special features on I²C?

- Low cost

- Easy implementation
- Moderate speed (upto 100 kbps).

11. What are disadvantages of I^{2}C?

- ➢ Slave hardware does not provide much support
- ➢ Open collector drivers at the master leads to be confused

12. Draw the data frame format of CAN?

Start	Arbitration field	Control field	Data field	CRC field	Acknowledgement field	End of frame
1	12	6	0-64	16	2	7

13. Define device driver.

A device driver is software for controlling, receiving and sending byte or a stream of bytes from or to a device.

UNIT-2

PART B (16 marks)

1. Explain SPI protocol and describe its interface.

SYNCHRONOUS PERIPHERAL INTERFACE (SPI):

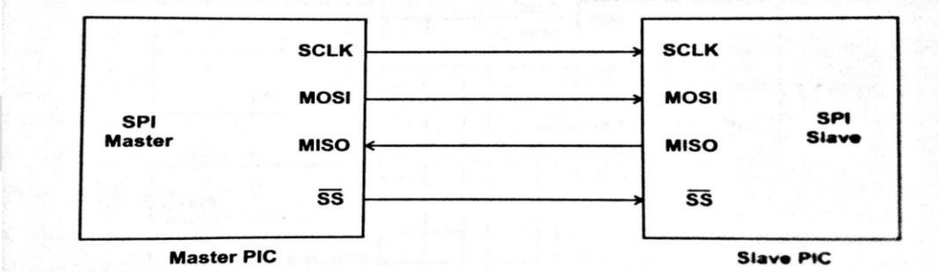
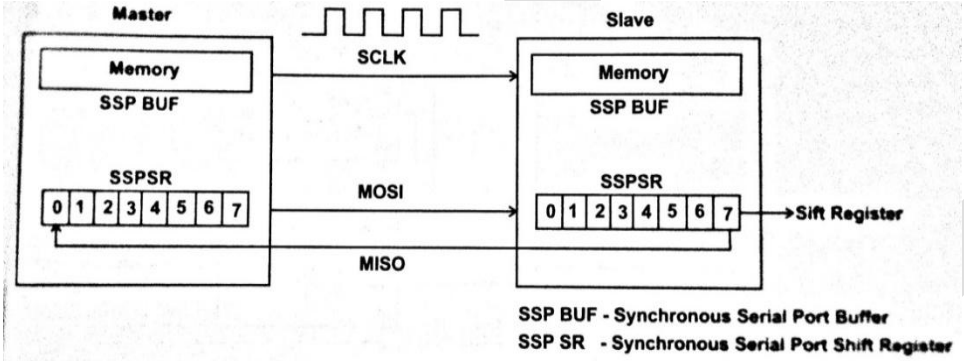


Fig. SPI Interface

SPI was defined by Motorola on the MC68HCXX line of microcontrollers.

- It is a full duplex communication.
- It supports 1. Full master mode, 2. Slave mode (with general address call)
- Addressing each device is not needed.
- SPI is a 3 wire communication such as MOSI, MISO and SCLK.
    - ➢ MOSI - Master output slave input
    - ➢ MISO - Master Input slave output
    - ➢ SCLK - Serial Clock

- If SS pin is set as 1, then the particular SPI device acts as a Master.
- If SS pin is set as 0, then the particular SPI device acts as a Slave.
- It supports 8 bit transmission and Reception
- Generally, SPI is faster than I^{2}C which is capable of several Mbps. Bus Arbitration logic is needed.
- A master sends a clock signal and upon each clock pulse it shifts one bit out to the slave and one bit in, coming from the slave.



SSP BUF - Synchronous Serial Port Buffer
SSP SR - Synchronous Serial Port Shift Register

Fig. Use of shift Registers

SSP BUF - Synchronous Serial Port Buffer

SSP SR - Synchronous Serial Port Shift Register

Bus Configuration:

1. *Parallel configuration (or) using separate chip selects:*

For the parallel connection, each device on the bus should have a separate CS line, while SCK, SDI and SDO lines are connected in parallel as shown in Figure.

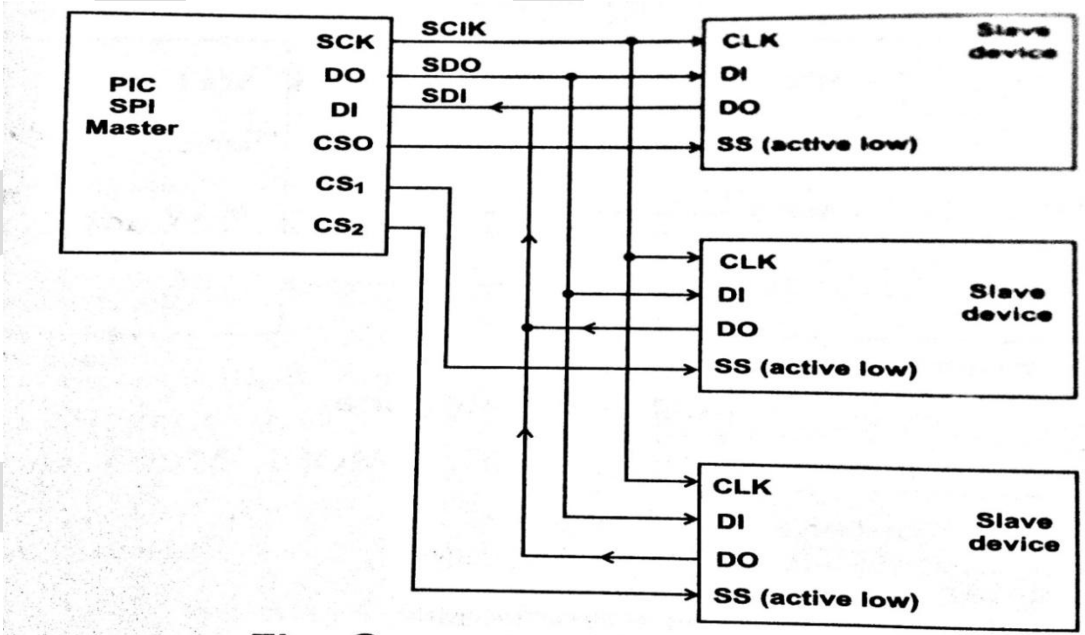


Fig. Separate chip selection

***2. Daisy chain configuration*:**

In this configuration CS and SCK lines connected in parallel and each SDO pin of previous chip is connected to SDI pin.

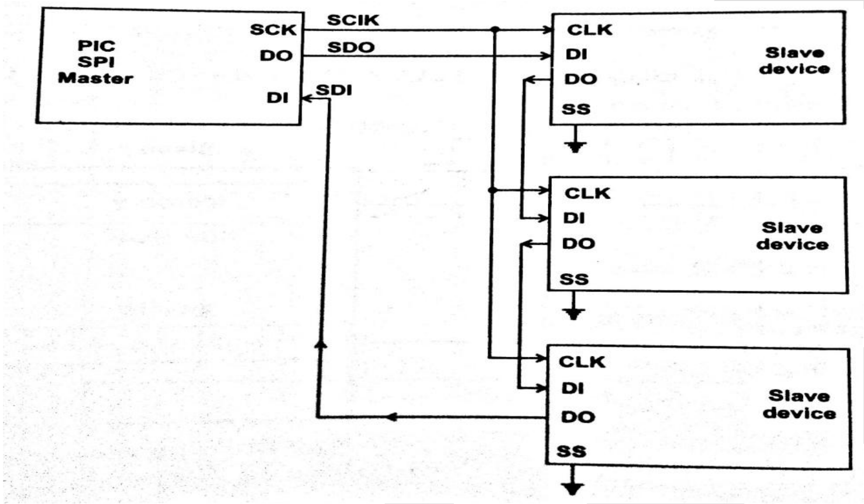


Figure: Daisy chain selection

Applications of SPI:

- SPI is used in EEPROM, Flash, and real time clocks.
- Better suited for data streams in ADC converters etc.
- Full duplex capability (i.e.) communication between a CODEC and digital signal processor.

2. Explain I^{2}C bus operation and describe its interface.

INTER INTEGRATED CIRCUIT (I^{2}C):

- I^{2}C Bus developed by Philips Semiconductor for TV sets in the 1980's
- I^{2}C is a 2 wire communication such as SDA (serial data) and SCL (serial clock).
- It performs half-duplex and synchronous communication.
- I^{2}C Supports:
  1. Master Mode
  2. Slave Mode
  3. Multi Master Mode
- Addressing is needed for each slave device. For example: In PIC microcontroller - MSSP Address Register is used for addressing.
- No chip select or Arbitration logic is required.

I²C Data Transfer:

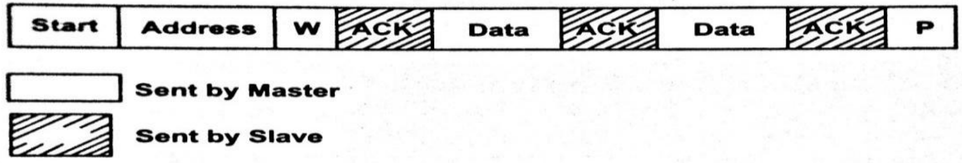


Fig. Data Transfer from Master to slave

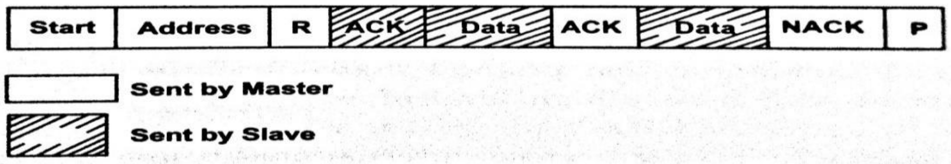


Fig. Data Transfer from Slave to Master

Steps:

- Master sends start condition (S) and controls the clock signal.
- Master sends a unique 7-bit slave device address.
- Master sends read / write bit (R/W) as 0 for slave receive and 1 for slave transmit.
- Wait for (or) send an acknowledge bit (A).
- Send (or) receive the data byte (8 bits) (DATA).
- Expect / send acknowledge bit (A)
- Send the stop bit (P).

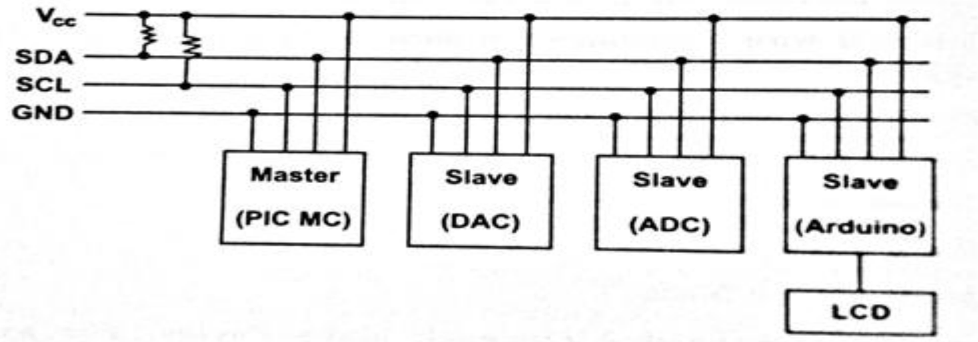


Figure: Application diagram

Application of I²C:

- Used as a control interface to signal processing devices that has separate data interfaces.
- Example RF tuners, video decoders and encoders and Audio processor.

3. Explain the CAN bus protocol with suitable diagrams.

CAN BUS:

- The Controller Area Network (CAN) is a serial bus communication protocol, which was originally developed for automotive applications by Bosch in 1980.
- A controller Area Network refers to a Network of Independent controllers. It is a Serial Communication protocol that efficiently supports distributed real time control with a very high level of security.
- CAN is a data link layer protocol internationally standardized as ISO-11898-1 and ISO-11519.
- The data on CAN bus is differential and can be in two states: dominant and recessive. The bus defines a logic bit 0 as a dominant bit and a logic bit 1 as a recessive bit.
- CAN bus uses 2 dedicated wires for communication such as CAN H and CAN L.
- When the CAN bus is in idle mode, both the lines carry 2.5V. When data bits are transmitted. CAN high line goes to 3.75V and CAN low drops to 1.25V. There by generating a 2.5V differential between the lines.
- CAN protocol is a message based protocol, not an address based protocol. CAN provides two communication services, the sending of a message (Data Frame Transmission) and the requesting of a Message (Remote Transmission Request) RTR.
- Each node is able to send and receive messages, but not simultaneously. A message consists primarily of an ID (Identifier), which represents the priority of the message and up to 8 data bytes. Signal pattern is NRZ (Non - return to zero)
- Baud rate is 1 Mbps and it is a Multi-master broadcast serial bus standard.
- Priority based bus arbitration mechanism is employed here.

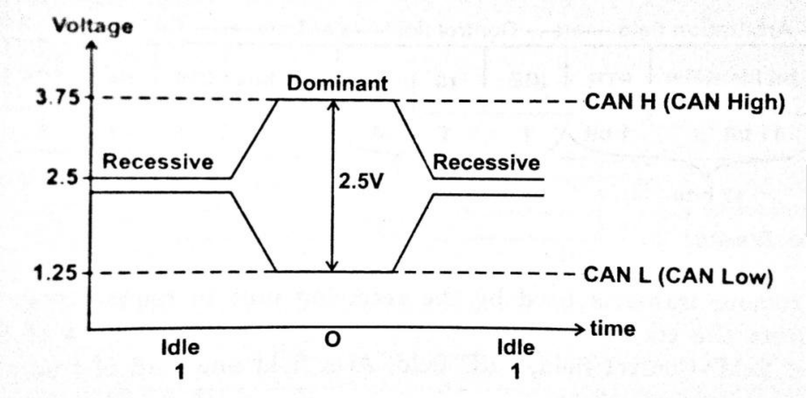


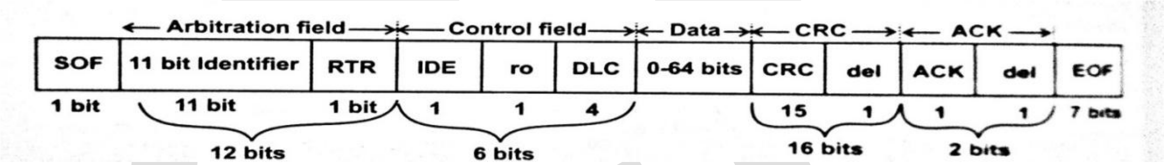
Fig. CAN Logic States

<u>There are basically four message frames in CAN:</u>

Data, Remote, Error and over head. The data and remote frames need to be set by the user. The other two are set by the CAN hardware.

<u>*1. Data Frame:*</u>

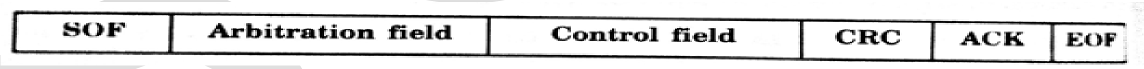
Data frame consists of fields such as start of frame (SOF), Arbitration field, Control field, Data field, CRC (Cyclic Redundancy Check) field, ACK (Acknowledge) field and EOF (End of Frame). If RTR is 1, then the packet is a data to the receiver, if RTR is 0, then the packet is a request for the data from the receiver.



	Arbitration field		Control field			Data	CRC		ACK		
SOF	11 bit Identifier	RTR	IDE	ro	DLC	0-64 bits	CRC	del	ACK	del	EOF
1 bit	11 bit	1 bit	1	1	4		15	1	1	1	7 bits
	12 bits		6 bits				16 bits		2 bits		

<u>*2. Remote Frame:*</u>

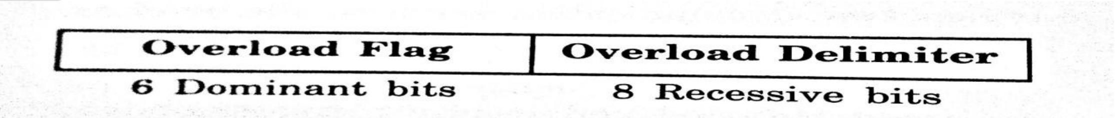
The remote frame is used by the receiving unit to request transmission of a Message from the transmitting unit. It consists of 6 fields: Start of Frame (SOF) Arbitration field, Control field, CRC field, ACK field and End of frame (EOF) field.



SOF	Arbitration field	Control field	CRC	ACK	EOF

<u>*3. Over Load frame:*</u>

The overload frame is used by the receiving unit to indicate that it is not get ready to receive frames. This frame consists of an overhead flag and an overload delimiter. The overload flag consists of 6 dominant bits and the overload delimiter consists of 8 recessive bits.



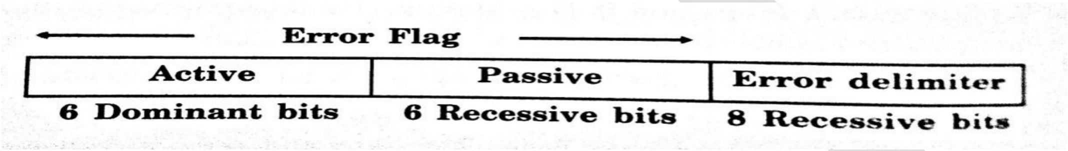
Overload Flag	Overload Delimiter
6 Dominant bits	8 Recessive bits

4. Error frame:

Error frames are generated and transmitted by the CAN hardware and are used to indicate when an error has occurred during transmission. This frame consists of an Error flag and an Error delimiter. Error Flag has 2 types.

1. Active with 6 dominant
2. Passive with 6 Recessive bits

The error delimiter consists of 8 Recessive bits.



Error Flag		
Active	Passive	Error delimiter
6 Dominant bits	6 Recessive bits	8 Recessive bits

CAN BUS Interface:

There are 2 Models of Interfaces depending upon the features present in Micro controller.

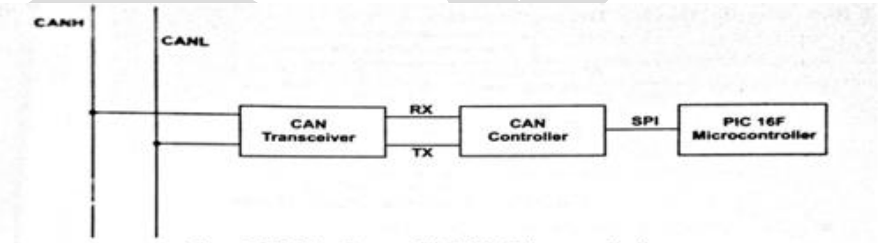


Fig. CAN Interface with PIC Microcontroller.

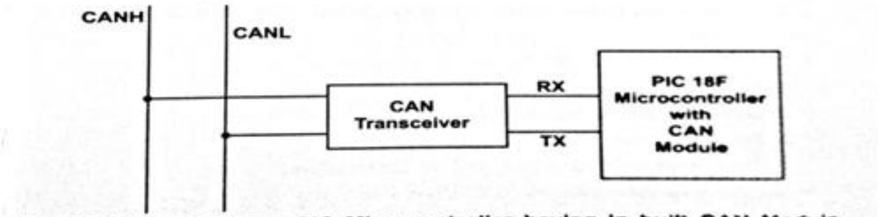
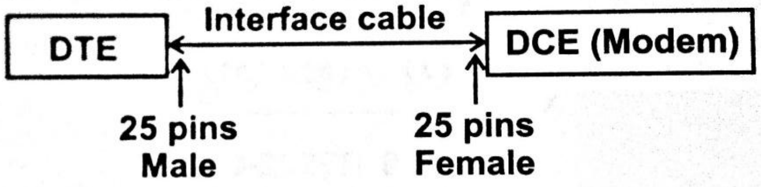


Fig. CAN Interface with PIC Microcontroller having in built CAN Module.

4. Compare RS232 & RS485 in detail.

RS 232 C STANDARD:

RS 232 C was defined by Electronic Industries Association (EIA) for data interchange. RS232 is an interfacing signal standard. It is a point to point interface.



DTE – Data Terminal Equipment

DCE – Data Communication Equipment

Fig. RS232 C Interface

- It provides communication between DTE and DCE. It follows the asynchronous UART protocol.
- Receiver end voltage level from -3V to -25V denotes logic 1 and voltage level from +3V to +25V denotes 0.
- Transmitter end voltage level from -5V to -15V denotes logic 1 and from +5 to +15V denotes logic 0.
- Commonly used Band rate for asynchronous serial communication with UART mode is 9600.
- The RS232 C standard allows data to be sent and received at the same time. (full duplex)
- RS232 C standard defines two types of lines such as DB9 and DB 25 for real world applications.
- The RS232 C standard defines,
    - The Mechanical aspects of interface.
    - The characteristics of electrical signals.
    - The functional aspects of the interchange.

RS232 C Connections:

Description	**Acronym**	**DB-25M Pin#**	**DTE DB-9M Pin#**	**Direction**	**DCE DB-9F Pin#**	**DCE DE-25F Pin#**
Transmit	TXD	2	3	→	2	3
Receive Data	RXD	3	2	←	3	2
Request to send	RTS	4	7	→	8	5
Clear to send	CTS	5	8	←	7	4
Data set ready	DSR	6	6	←	4	20

Data carrier detect	DCD	8	1	←	1	8
Data terminal ready	DTR	20	4	→	6	6
Ring indicator	RI	22	9	←	9	22
Signal ground	SG	7	5	←	5	7

RS232 C connections (DTE):

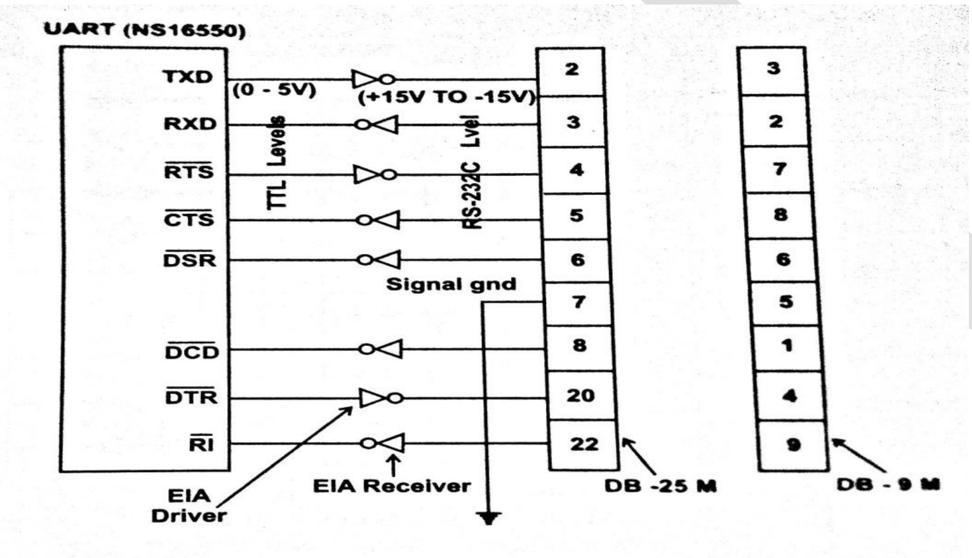


Fig. DTE Connections

RS232 C connections (DTE to DCE):

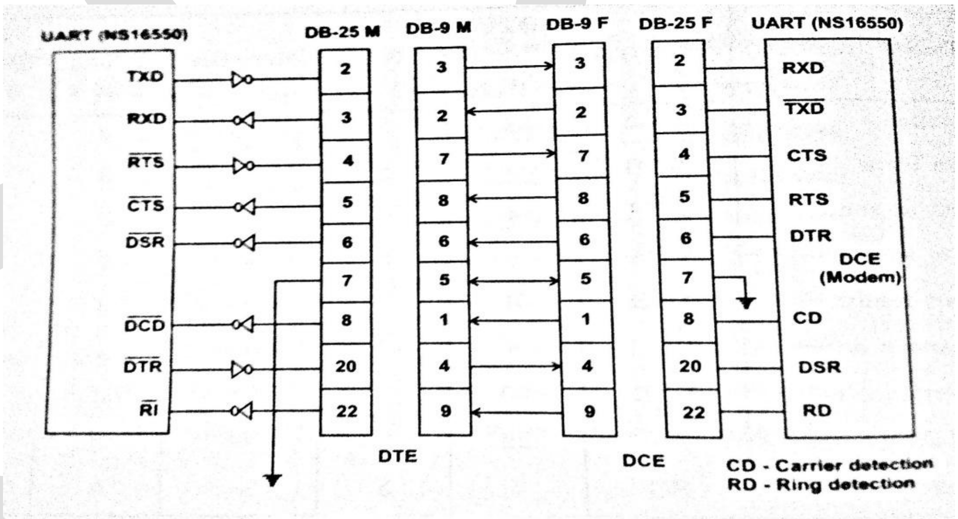
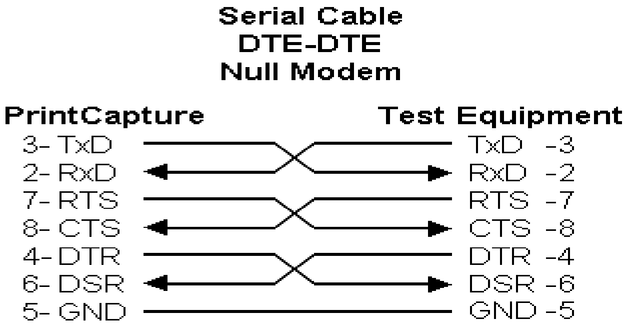
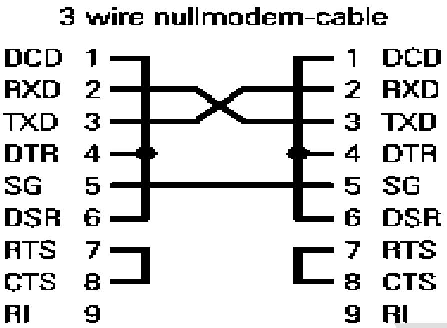


Fig. DTE to DCE Connections

RS232 C NULL MODEL (DTE TO DTE):

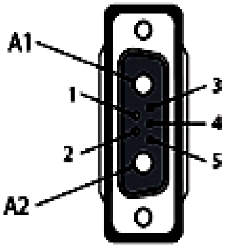


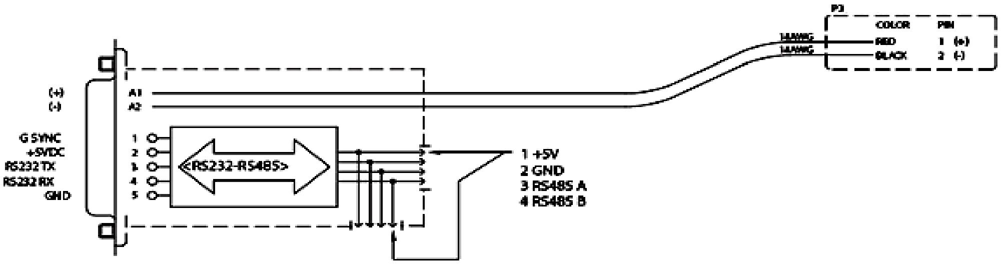
RS232 C 3 wire DTE to DTE:



RS485:

- RS-485 allows multiple devices (up to 32) to communicate at half-duplex and full duplex at distances up to 1200 meters .
- Both the length of the network and the number of nodes can easily be extended using a variety of repeater.
- Data is transmitted differentially on two wires twisted together, referred to as a twisted pair.
- A 485 network can be configured two ways, two-wire or four-wire
- In a two-wire network the transmitter and receiver of each device are connected to a twisted pair.
- Four-wire networks have one master port with the transmitter connected to each of the slave receivers on one twisted pair. The slave transmitters are all connected to the master receiver on a second twisted pair.
- In either configuration, devices are addressable, allowing each node to be communicated to independently.
- Only one device can drive the line at a time, so drivers must be put into a high-impedance mode (tri-state) when they are not in use.
- Two-wire 485 networks have the advantage of lower wiring costs and the ability for nodes to talk amongst themselves. But is limited to half-duplex.

- Four-wire networks allow full-duplex operation, but are limited to master-slave situations. Slave nodes cannot communicate with each other.
- RS485 communication is half duplex. Each communicating element on an RS485 interface is called a node. One of the nodes is called the master while all other nodes are called slaves. Each node has a unique ID number. Node #0 is generally assigned to the master.
- RS485 features are
  - Very noise immune
  - Maximum cable length of 4000 feet
  - Data rate up to 10 Mbps
  - Capable of supporting a multi-master configuration.
  - 



5. Write short Note on (i) Input and output ports (ii) UART

(i) **INPUT AND OUTPUT PORTS**:

Ports:

A port is a device to receive the bytes from external peripherals for reading them later using instructions executed on the processor (or) to send the bytes to external peripheral or device or processor using instructions executed on processor A part connects to the processor using address decoder and system buses.

The processor uses the addresses of the port registers for programming the port functions or modes, reading port status and for writing or reading bytes.

Example:

- Serial Peripheral Interface (SPI) in 68 HCII.

- Ports PO, P1, P2 and P3 in 8051.
- COM1 & COM2 ports in an IBM PC.

Types of Serial Ports:

- Synchronous Serial input
- Synchronous Serial output
- Asynchronous serial UART input
- Asynchronous serial UART output
- Both as Input and as output. Example: Modem

1. Synchronous Serial input Device:

- The sender along with the serial bits also sends the clock pulses SCLK to the receiver port pin. The port synchronized the serial data input bits with clock bits.
- Synchronization means separation by a constant interval (or) phase difference. If clock period = T, then each byte at the port is received at Input in period = 8T. The data transfer rate is (1/T) bps (bits per second).
- Serial data and clock pulses are sent either on a same input line or separate input line.
- The peripheral saves the byte at port register from where the Microprocessor reads the byte.
- MOSI (Master output slave Input), when the SCLK is sent from the sender to the receiver and slave is forced to synchronize sent inputs from the master as per the inputs from master clock.
- MISO (Master Input Slave Output) when the SCLK is sent to the sender from the receiver and slave is forced to synchronize for sending the inputs to master as per the master clock outputs.
- Synchronous Serial input is used for inter processor transfers, audio inputs and streaming data inputs.

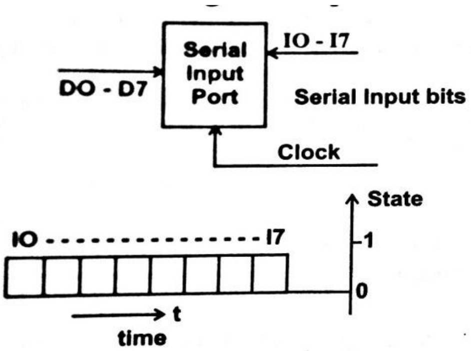


Fig. Synchronous Serial Input device

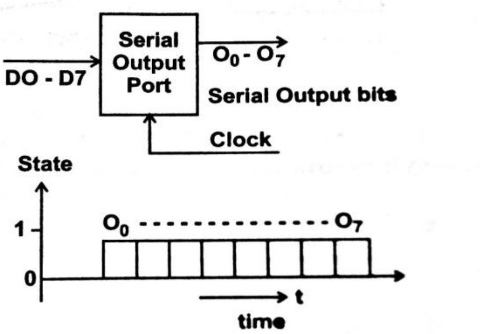


Fig. Synchronous Serial Output device

<u>2. Synchronous Serial output Device:</u>

- Each bit in each byte sent in synchronization with a clock.
- Bytes sent at constant rates. If clock period is T, then data transfer rate is (1/T) bps.
- Sender either sends the clock pulses at SCLK pin or sends the serial data output and clock pulse input through same output line with clock pulses either suitably modulate (or) encode the serial output bits.
- The peripheral at the port sends the byte through a shift register at the port to where the microprocessor writes the byte.
- Synchronous serial output is used for inter processer transfers, audio outputs and streaming data outputs.

Synchronous Serial Input / Output:

- Each bit in each byte is in Synchronization at input and each bit in each byte is in synchronization at output with the master clock output.
- The bytes are sent or received at constant rates.
- The processing element at the port sends and receives the byte at a port register to or from where the microprocessor writes or reads the byte.

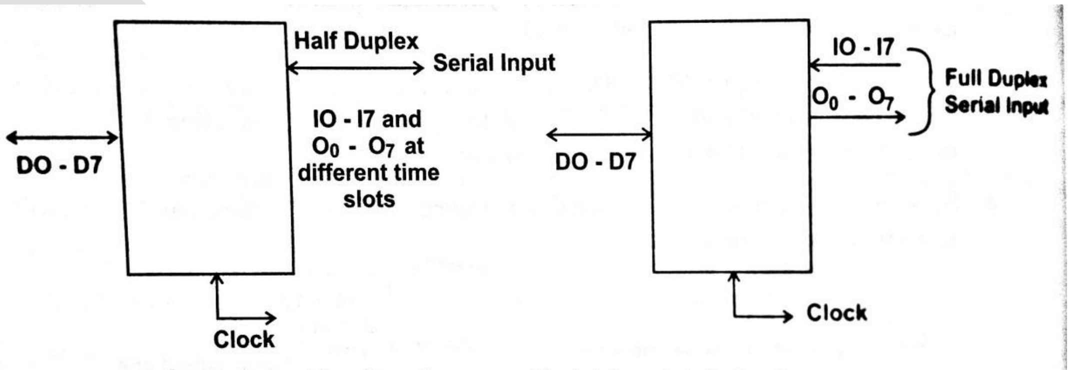


Fig. Synchronous Serial Input / Output

<u>3. Asynchronous Serial Input: (Serial Reception)</u>

- Each RXD (Receive data) bit is received in each byte at fixed intervals but each received byte is not in synchronisation.
- It does not receive clock Information along with the data bytes.
- Bytes are received at variable Intervals (or) phase differences. Asynchronous serial input is also called as UART Input.
- A 1 to 0 transition indicates the reception of a byte.
- Time period for 1 byte is 10 T, which includes the one start bit, 8 data bits and one step bit.
- The peripheral saves the byte at a port register from where the microprocessor reads the byte.
- Examples: Keyboard Input & Modem Inputs in Computer.

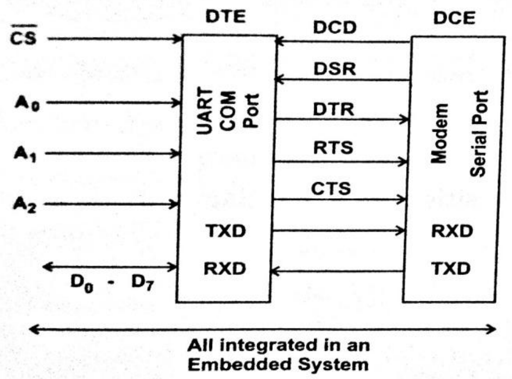


Fig. Handshaking Signals

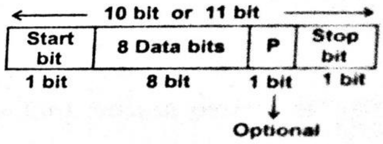


Fig. UART Frame format

4. Asynchronous Serial Output: (Serial Transmission)

- Each bit in each byte is sent at fixed intervals but each output byte is not in synchronization and its line is denoted by TXD (transmit data)
- Bytes are transmitted at variable Intervals (or) phase differences. Asynchronous serial output is also called as UART output.
- It does not send clock information along with the data bytes.
- The peripheral sends the byte at a port register to where the microprocessor is to write the byte.
- Examples: Modem & Printer Inputs from computer.

5. Half Duplex and Full Duplex:

Half Duplex: Both the nodes can be able to transmit and receive, but not at the same time.

Full duplex: Both the nodes can be able to transmit and receive, but not at the same time.

Types of Parallel ports:

1. Parallel port one bit Input:

Example: Filling of a liquid up to a fixed level

2. Parallel port one bit output:

Example:

- PWM output for a DAC.
- Pulses to an external circuit.
- Control signal to an external circuit.

3. Parallel port Multi bit input:

Example:

- ADC Input from Liquid level measuring sensor or temperature sensor (or) pressure sensor (or) Speed Sensor.
- Encoder inputs for bits for angular position of a rotating shift or a linear displacement of an object.

4. Parallel port Multi bit output:

Example:

- Print controller output
- Stepper Motor coil driving bits.

5. Parallel port input - output:

Example:

- PPI 8255
- Touch Screen in Mobile Phone.

(ii) UART (UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER):

Baud Rate:

It is assumed that the receiver knows how fast each bit is being transmitted. This transmission rate is known as the baud rate. The term “Baud” is spelled only in the asynchronous communication.

UART Mode is as Follows:

- Non return to zero (NRZ) state denotes logic state is 1 at serial line.
- Start of serial bits is signaled by 1 to 0 transitions on the line for a period equal to reciprocal of baud rate.
- Transmission of a byte consists of one start bit, eight data bits, one parity bit and one stop bit.

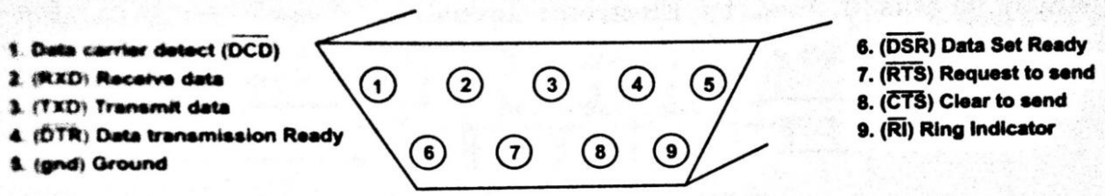


Fig. DB9 RS232-C

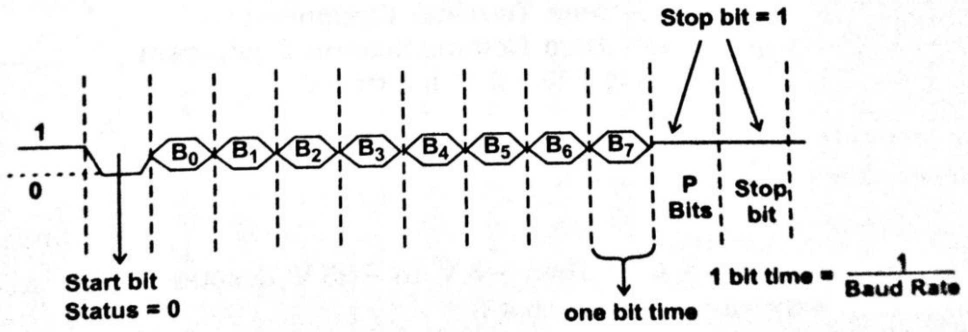


Fig. Transmission of bits

- Stop bit is denoted by the logic one.
- 1 bit time = δT

  Therefore 10 bit time = 10δT
- Programmable bit (P.bit) is used for parity detection (or) to specify the purpose of serial data bits.
- For transmission and reception, UART 16550 has a 16 byte FIFO buffer.