CHAPTER 08: INPUT – OUTPUT ORGANIZATION

Chapter 8 — Input-output organisation

* **Asynchronous Data transfer:** The data transfer takes place when peripherals & computer does not share the common clock. Asynchronous transfer use control signals & their associated hardware to coordinate movement of data.

* **Synchronous Data transfer**

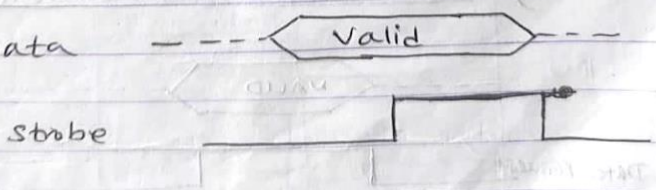
when peripherals & computer share common clock. when peripheral are located within the same computers as the CPU

* **Modes of asynchronous data transfer**

1) **Source-initiated data transfer :-**

- It is data transfer without handshaking.
- The source outputs its data, strobes a control signal for a set of time.
- The destination device reads the data during this time. Then source device next deasserts the strobe & stop outputting data.

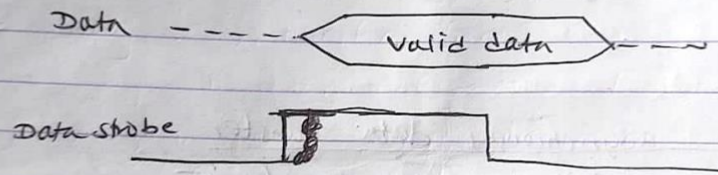
(Timing diagram)



In this mode, source doesn't know whether the destination device receive data or not because destination device doesn't send any information back.

2) Destination - initiated data transfer

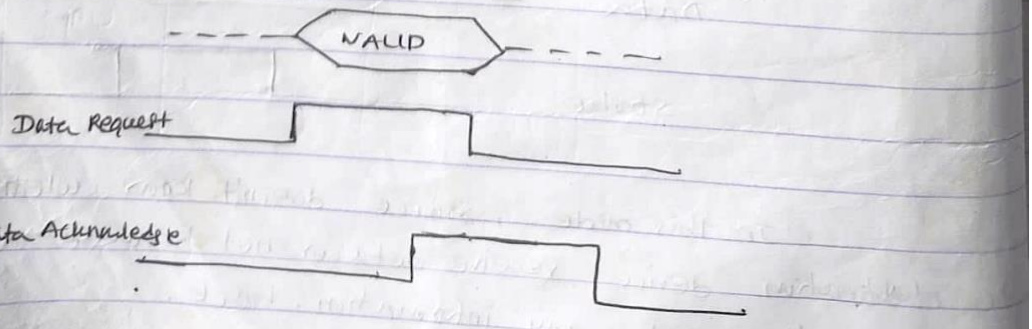
- Destination device initiates the data transfer.
- data transfer without handshaking.
- The destination device transmits a strobe signal to source device
- After a brief delay, data is available.
- when valid data is ready, the destination device reads the data & deasserts data strobe.
- This in turn causes sources to stop transmitting valid data.



3) Handshaking :-

Handshaking is the process of coordinating the data transfer. if the same amount of time is not required for data's transfer then handshaking is used.

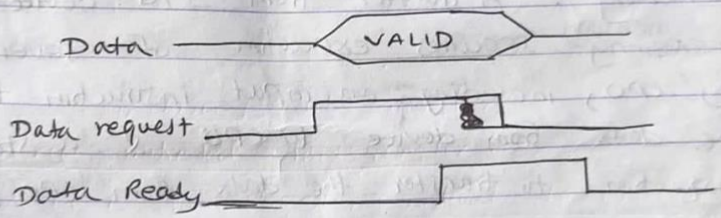
a) Source initiated data transfer with handshaking



Handshaking scheme provides a high degree of flexibility & reliability because the successful completion of data transfer relies on active participation by both units.

- First of all, the source sets the data request signal high & then makes valid data available to destination device. The destination device reads the data after data is stabilize. (delay) once the destination device has read data, it sends a data acknowledge signal to the source. This tell the source that destination has read in & no longer needs this data. Then source sets its data request line low & stops sending data. The destinatim then resets its data acknowledge signal.

3 b) Destination initiated data transfer with handshaking



This is similar to source initiated data transfer using handshaking except that data acknowledge signal is replaced by data-ready signal. (source)

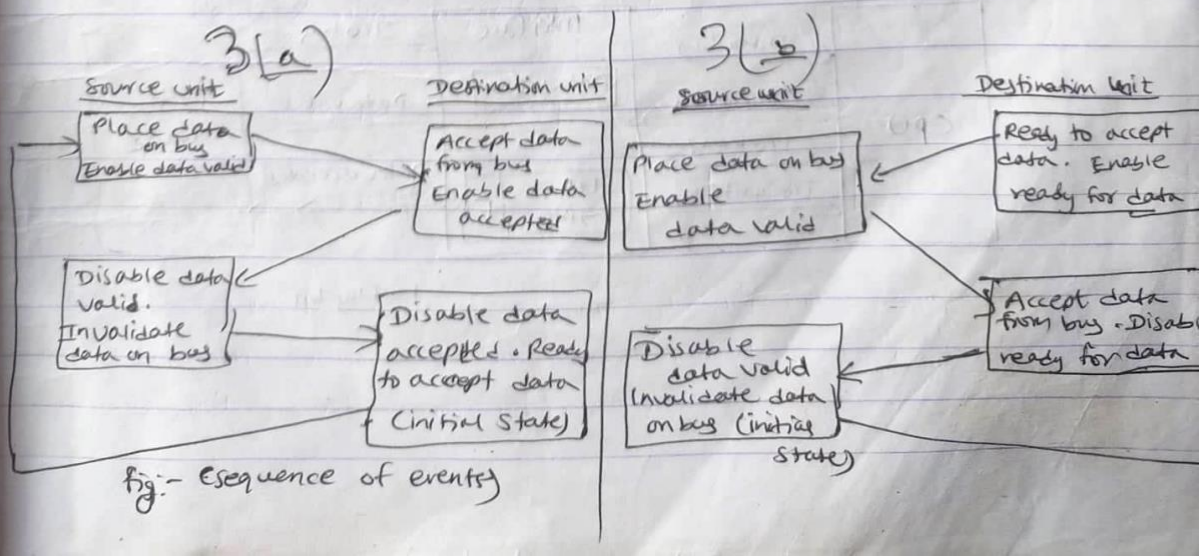


fig:- Esequence of events

* Input/output Techniques
1) Programmed I/O
2) DMA
3) Interrupt I/O

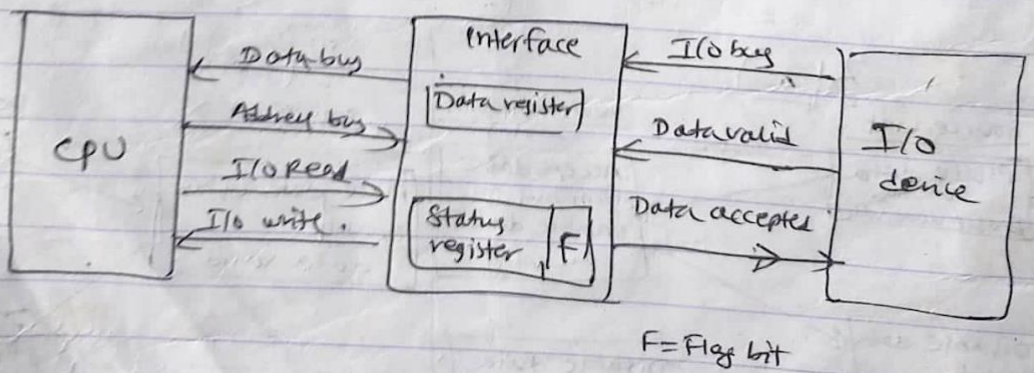
1) **Programmed I/O** :- A mode for handling data transfer to & from peripherals.

- (Programmed I/O operations are the result of I/O instructions written in computer program)

Each data item transfer is initiated by an instruction in the program. -(Usually, the transfer is to & from a CPU register and peripheral.)

(In this mode, (CPU stays in a program loop until I/O unit indicates that it is ready for data transfer). This is time consuming process since it keeps the processor busy needlessly.)

(The I/O device does not have direct access to memory). (A transfer from I/O device to memory requires execution of several instructions by CPU, including an input instruction to transfer the data from device to CPU & a store instruction to transfer the data from CPU to memory) other instructions may be needed to verify that data are available from device & to count numbers of words transferred.



F = Flag bit

3) ... explain interrupt processing steps
4) Define vectored ... type of

Interrupts :- Interrupt is external event or signal generated by I/O device to processor to request for service.

Types of interrupts :-

i) External interrupts
ii) Internal interrupts
iii) Software interrupts

~~Process of~~ Interrupts processing steps:

vectored interrupt & polled interrupts
interrupt priority

Direct memory access :-

The transfer of data between a ~~fast~~ storage device (magnetic disk) & memory is often limited by speed of CPU. By Removing the CPU from the path & letting the peripheral device manage the memory buses directly would improve speed of transfer. This transfer technique is called DMA.

During DMA transfer, the CPU is idle & has no control of memory buses. A DMA controller take control over the buses to manage transfer directly betn I/O device & memory.

DMA transfer modes:

a) Burst transfer mode:

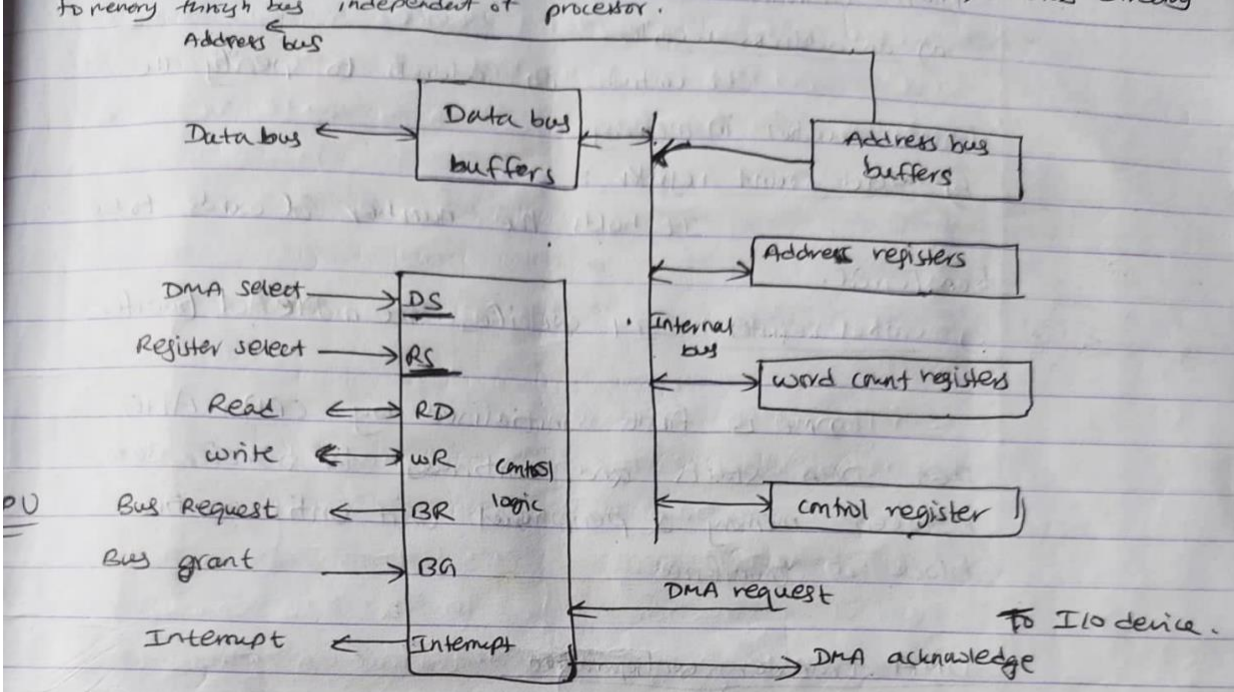
(In this mode, a block sequence consisting of a number of memory words is transferred in a continous burst while DMA controller is master of memory buses.) This transfer mode is needed for fast devices such as magnetic disks, where data transmission cannot be stopped or slowed down until an entire block is transferred.

b) Cycle stealing:

(In this mode, DMA controller transfer one data word at a time, after which the control of buses return to CPU). The CPU merely delay its operation for one memory cycle to allow direct memory I/O transfer to 'steal' one memory cycle.

c) Transparent mode: (In this mode, DMA controller only transfer data when CPU is performing operations that do not use system buses.)

DMA controller: DMA controller keeps track of memory locations, transfers directly to memory through bus independent of processor.



The figure shows a block diagram of DMA controller. The unit communicates with CPU through data bus & control lines. The registers in DMA are selected by CPU through address bus by enabling DS & RS inputs. The RD & WR signals / inputs are bidirectional.

i) when BG input is 0, CPU communicate with DMA registers through data bus to read from or write to DMA registers

ii) when BG = 1, CPU has relinquished (releases) the buses & DMA can communicate directly with memory by specifying an address in address bus & activating RD or WR control.

DMA communicates with external peripheral through request & acknowledge lines by using a handshaking procedure.

DMA controller has 3 registers

a) an address register

It contain an address to specify the desired location in memory.

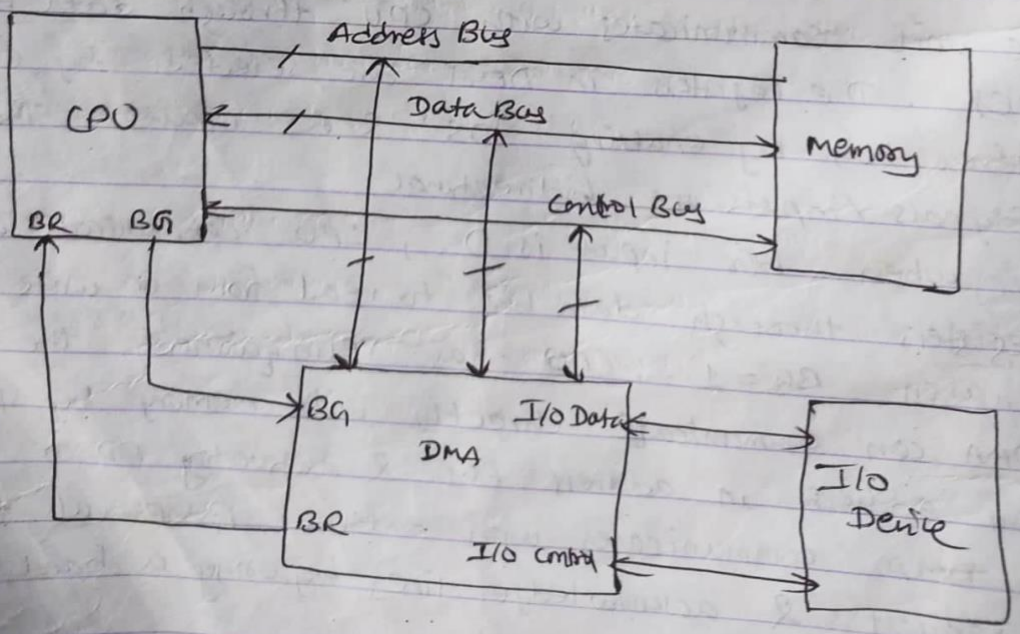
b) word count register:

It holds the number of words to be transferred.

c) control register: It specifies the mode of transfer

DMA is first initialised by CPU. After that DMA starts and continues to transfer data between memory & peripheral unit until entire block is transferred.

DMA transfer configuration



Transferring data from I/O device to memory.

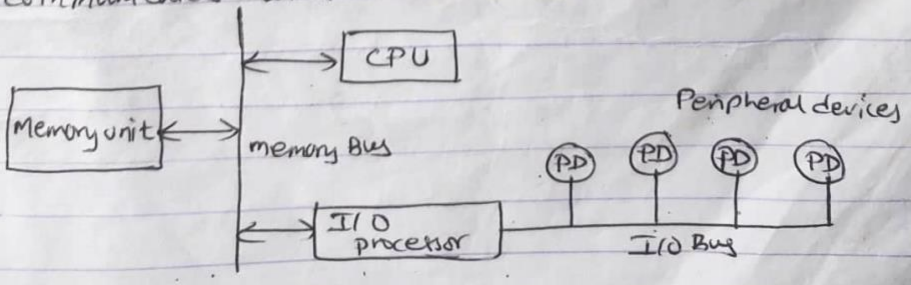
At first DMA controller sends a BR (Bus request) signal to CPU by setting BR to 1. Then CPU sets it Bus grant (BG) signal to 1 to grant the request.

CPU also grant control of system buses to DMA controller (Address bus, Data bus, control bus). Since DMA controller has control of system buses, it can perform data transfers.

For loading data from i/o device to memory, it asserts appropriate i/o control signals & loads data from I/O device to its internal DMA data registers, Then, DMA writes this data to memory. For doing this, it o/p mem. address on system address bus, data onto data bus & appropriate control signals on control bus. DMA continue until entire block of data is transferred.

when DMA doesn't need system bus (after the work is finished) it sets BR=0. And then CPU sets BG=0.

I/O processors :- The processor having the capability of direct memory access to communicates with I/O devices.



The IOP provides a path for transfer of data between various peripheral devices & memory unit. IOP operates independent of CPU & continues to transfer data from external device & memory.

An Intel's processor with Built in DMA

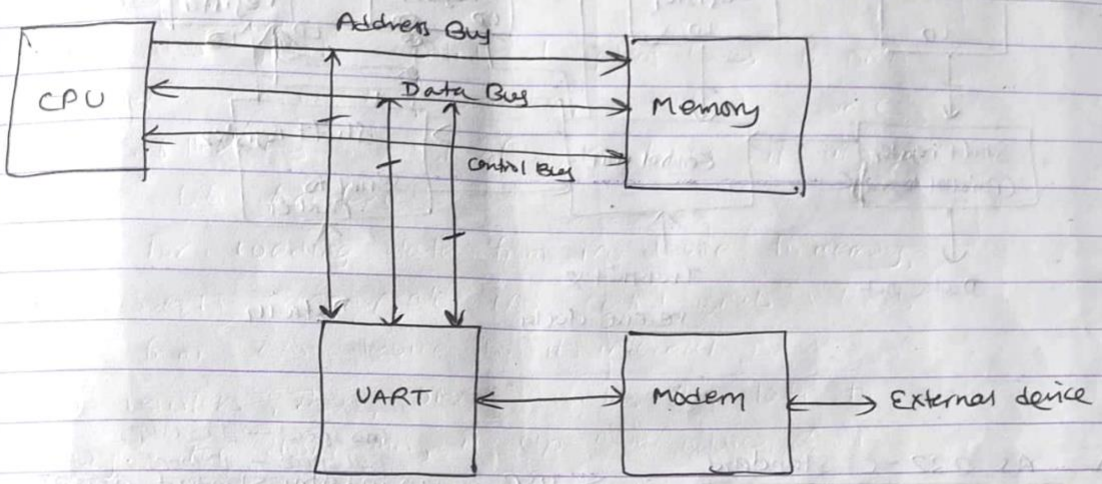
- i960 I/O processor
- It includes functions associated with IOP & other functional units.
- i960 IOP runs at 100 MHz & has a 16k internal instruction cache for its I/O command.
- It contains a PCI bus interface to interact with system bus & a local bus connection for comm^n with I/O devices.
- It includes DMA controller with two channels. They can transfer 132 MB per second between PCI bus & local bus.

Serial communication :— one bit at a time

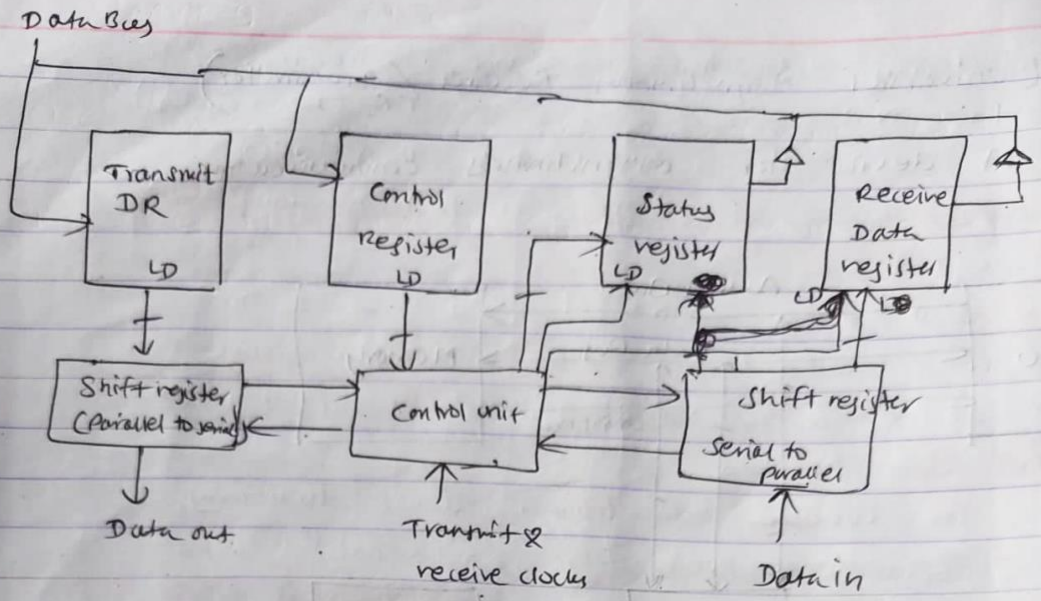
- synchronous serial comm^n
- asynchronous serial comm^n

UART (Universal Asynchronous Receiver / Transmitter)

- A device for asynchronous communication.



- UART is parallel I/O device as seen by CPU
- But it inputs & o/p data serially. It can interact with any device that can access serial data.
- In given figure, UART exchange data with a modem.
- For transmitting data, UART o/p sequential data to modem. which modulates the data, combining with carrier frequency & transmitting it.
- For receiving data, modem accepts a signal at different carrier frequency & demodulates it, extracts data & transmit it serially to UART.



- RS 232-C Standard
- RS 422 Standard
- Universal serial Bus Standard

low speed — 1.5 Mbps
Full speed — 12 Mbps

USB is an industry standard developed to provide two speed of operation called low speed and full speed. They provide simple, low cost & easy to use interconnection system.

USB is a specification to establish communication devices & computer. It replace serial & parallel ports

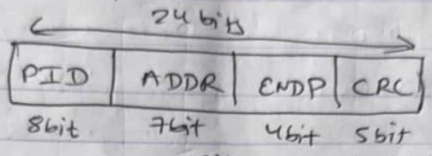
- USB transmits data in packets. This packet can specify an address, allowing several devices to be connected to USB port.
- USB port is faster then RS-232 C port.
- The USB standard specifies 4 types of packets which are used to communicate between a computer & its USB peripherals.

Token packets :- It is used to initiate data transfers. It specifies address & end point for a transfer.

Data packets : It contains data transferred to or from a device.

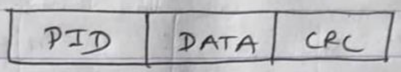
Handshake packet: It transfer information used to coordinate data transfer.

special packets: It includes diff. other functions.

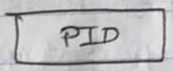


a) Token packet.

PID = Packet identifier
ENDP = end of packet
CRC – cyclic Redundancy check.



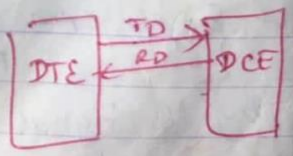
b) Data packet



8 bits. c) Handshake packet

RS 232-C Standard: RS 232 is an interface convention developed to standardise the interface between data terminal equipment (DTE) & data comm^n equipment (DCE), employing serial binary data exchange.

9-pin connection or 25 pin connection.



9 pin

3	TD	transmitted data	DTE → DCE
2	RD	received data	DCE → DTE
7	RTS	Request to send	DTE → DCE
8	CTS	Clear to send	DCE → DTE
6	DSR	Data set ready	DCE → DTE
5		Signal ground	

Chapter 9 RISC

CISC computers uses CISC processors. (Complex instruction set computer)

RISC uses RISC processors (Reduced instruction set computer)

1	DCD	DCE → DTE	Data carrier detect
4	DTR	DTE → DCE	Data terminal ready
9	RI	DCE → DTE	Ring indicator

* RS 422A Standard : The main problem with RS 232C is that it can only transfer data reliably about 50 f at a max^m rate of 20,000 Bd. This limitation is due to open signal lines with a common ground that are used for RS-23

RS 422A specifies that each signal will sent differentially over two adjacent wires in a ribbon or a twisted pair cable. Data rates for this standard are 10 MB for a distance of 50ft or 100,000 Bd for a distance of 4000ft.

It is because the differential lines are terminated by resistors so that they act as simple tx- lines inst of simply open wire.

For RS 422A, a logic high or 1 is indicat by B signal line being more positive then A signal line. & logic low or 0 by vice versa. The voltage difference betⁿ two lines must be greater then 0.4 V but not greater then 12V.