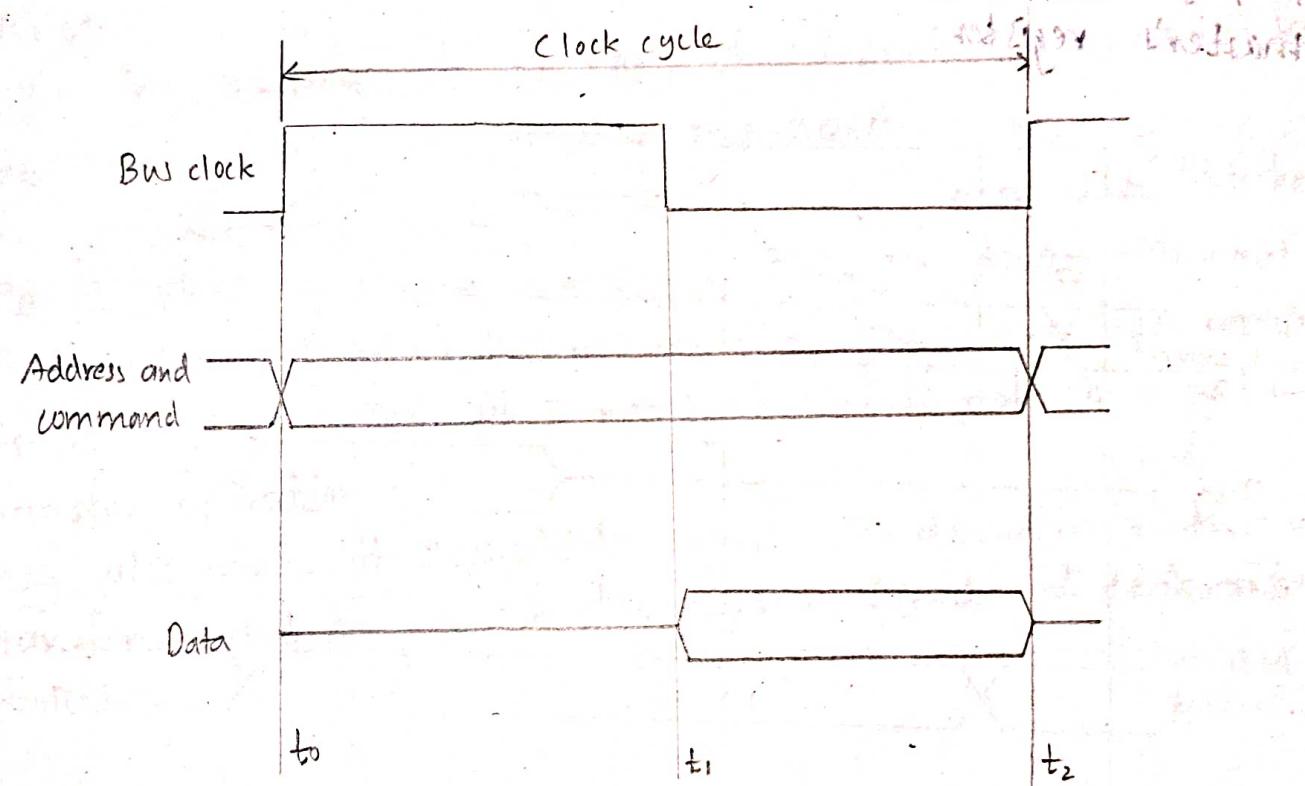


Assignment

1. With a neat (diagram) timing diagram, Explain what input transfer on a synchronous bus.

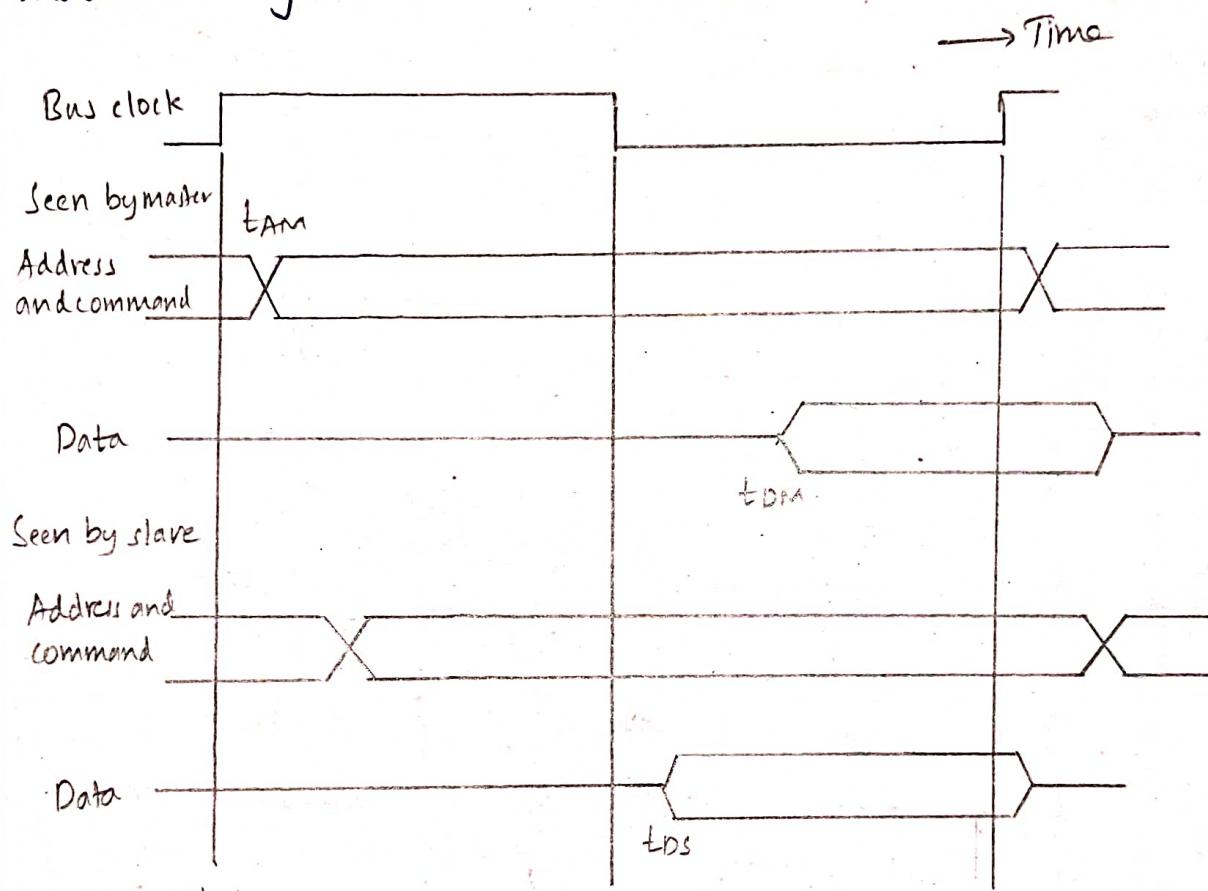


Timing of an input transfer on a synchronous bus

- * On a synchronous bus, all devices derive timing information from a control line called the bus clock.
- * The signal on this line has two phases: a high level followed by a low level.
- * The two phases constitute a clock cycle.
- * The first half of the cycle, between the low-to-high and high-to-low transitions, is often referred to as a clock pulse.
- * clock pulse width, t_1-t_0 , must be longer than the maximum propagation delay over the bus.

02

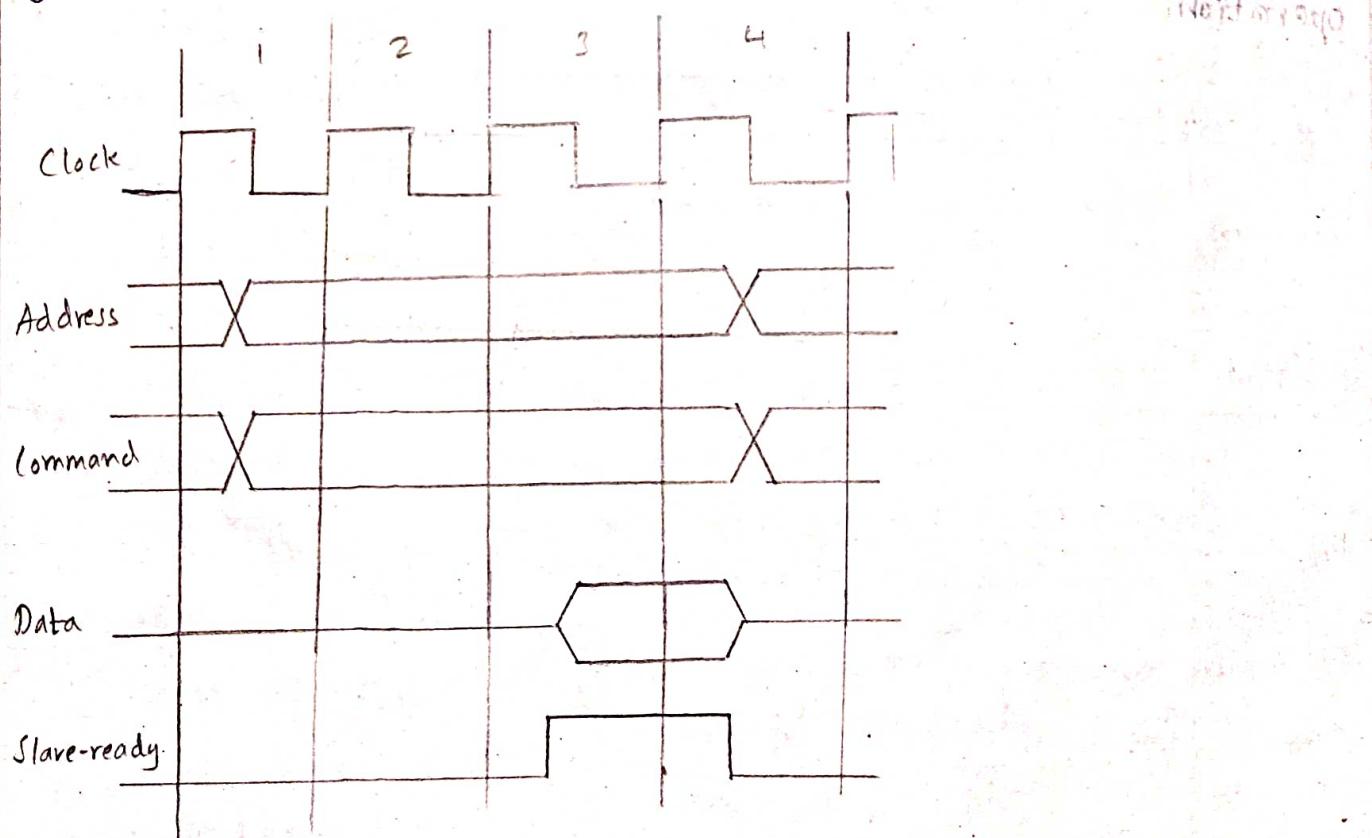
- * It must be long enough to allow all devices to decode the address and control signal
- * The period $t_2 - t_1$ must be greater than the maximum propagation time on the bus plus the setup time of the master's register



A detailed timing diagram for input transfer.

- * The master sends the address and command signals on the rising edge of the clock at the beginning of the clock cycle (at t_0).
- * However, these signals do not actually appear on the bus until t_{AM} .
- * A short while later, at t_{AS} , the signals reach the slave.
- * Slave decodes address and sends requested data at t_1 .
- * Data signal appears on bus at t_{DS} and arrives at master about t_{DPA} .
- * Master loads the data into register at t_2 .

- Period $t_2 - t_{pd}$ must be greater than the setup time of that register.
- Transfer has to be completed within one clock cycle **03**
- the clock period, $t_2 - t_0$, must be chosen to accommodate the longest delays on the bus and the slowest device interface.
- This forces all devices to operate at the speed of the slowest device.
- Also the processor has no way of determining whether the addressed device has actually responded.
- To overcome these limitations, most buses incorporate control signals that represent a response from the device.
- These signals inform the master that the slave has recognized its address and that it is ready to participate in a data transfer operation.
- They also make it possible to adjust the duration of the data transfer period to match the response speeds of different devices.

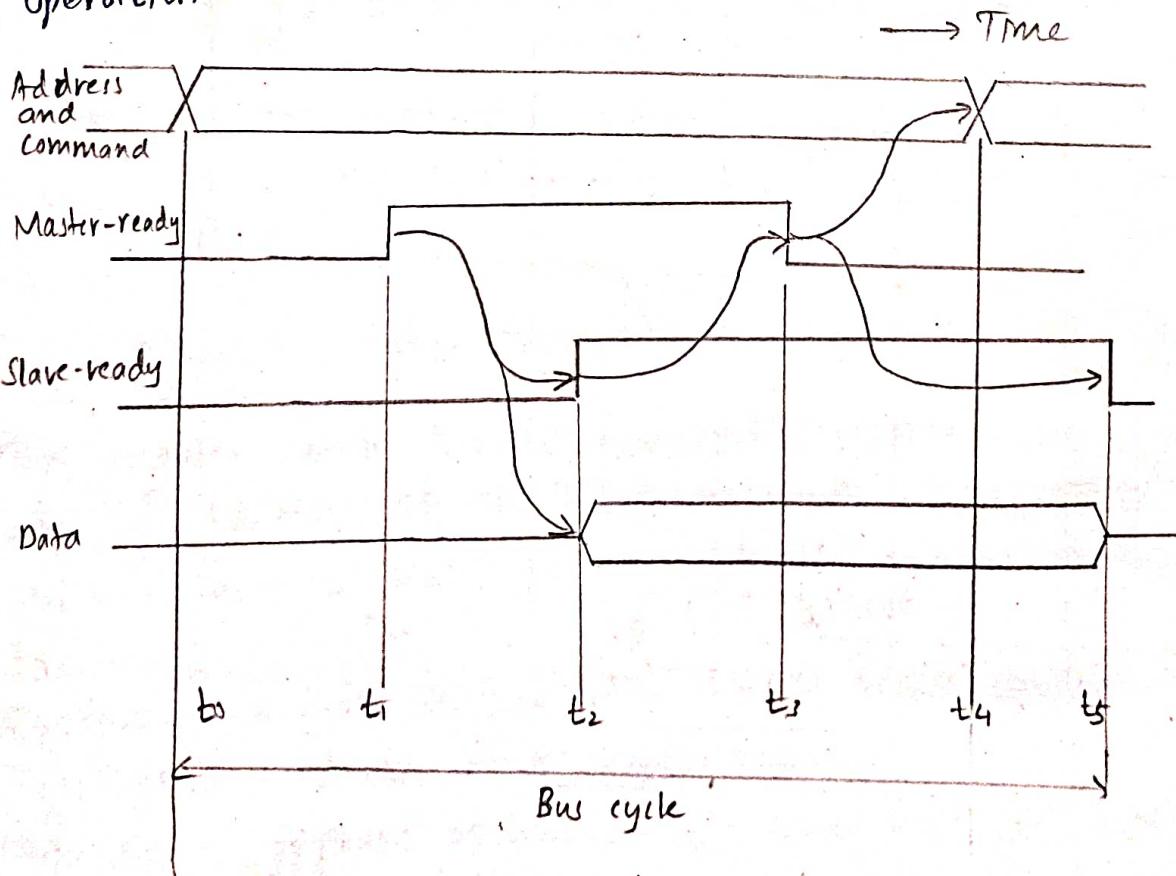


An input transfer using multiple clocks

04

- * This is often accomplished by allowing a complete data transfer operation to span several clock cycles.
- * During clock cycle 1, the master sends address and command information.
- * At the beginning of clock cycle 2 the slave begins to access the requested data.
- * The data become ready and placed on the bus during clock cycle 3.
- * The slave removes its data signals from the bus and returns its Slave-ready signal to the low level at the end of cycle 3.
- * A new transfer can start in clock cycle 4.

2. Explain Handshake control of data transfer during an input operation.



Handshake control of data transfer during an input operation

- + An alternative scheme for controlling data transfers on a bus is based on the use of a handshake protocol between the master and the slave.
- * A handshake is an exchange of command and response signals between the master and the slave.
- * A control line called master-ready is asserted by the master to indicate that it is ready to start a data transfer. The slave responds by asserting slave-ready.
- * The master places the address and command information on the bus, and all devices on the bus decode this information.
- * At t_1 , the master sets the master-ready line to 1 to inform the devices that the address and command information is ready. The delay t_1-t_0 is intended to allow for any skew that may occur on the bus. Skew occurs when two signals transmitted simultaneously from one source arrive at the destination at different times. Thus, to guarantee that the master-ready signal does not arrive at any device ahead of the address and command information, the delay t_1-t_0 should be longer than the maximum possible bus skew. The delay required can be included in the period t_1-t_0 .
- * At t_2 , the selected slave, having decoded the address and command information, performs the required input operation by placing its data on the data lines. At the same time, it sets the slave-ready signal to 1. The period t_2-t_1 depends on the distance between the master and the slave and on the delays introduced by the slave's circuitry.
- * At t_3 , the slave-ready signal arrives at the master, indicating

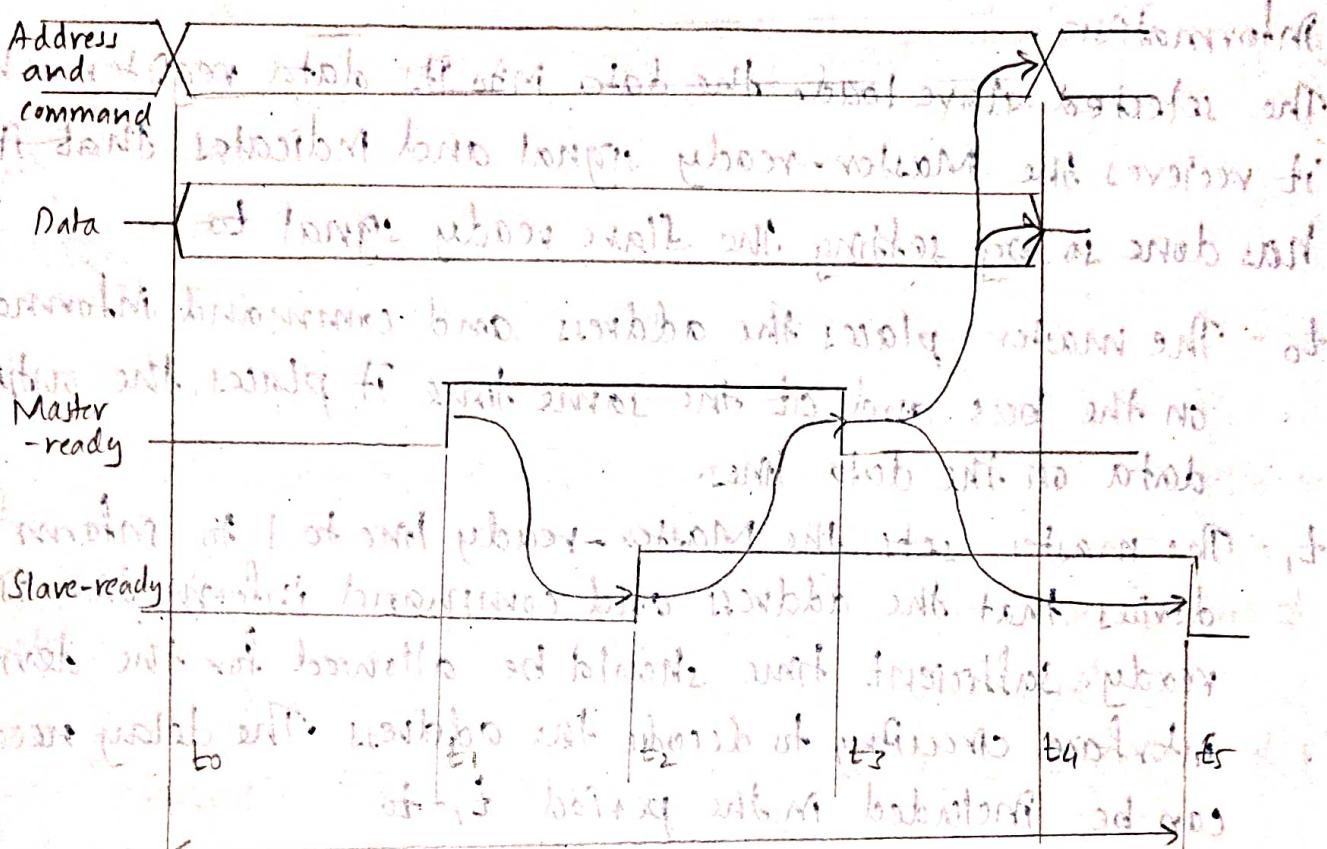
06

- that the input data are available on the bus. The master must allow for bus skew. It must also allow, for the setup time needed by its register. After a delay equivalent to the maximum bus skew and the minimum setup time the master loads the data into its register. Then it drops the Master-ready signal, indicating that it has received the data. Both t_3 and t_4 are intended to allow for bus skew.
- * t_4 - The master removes the address and command information from the bus. The delay between t_3 and t_4 is again intended to allow for bus skew. Erroneous addressing may take place if the address, as seen by some device on the bus, starts to change while the master-ready signal is still equal to 1.
 - * t_5 - When the device interface receives the 1-to-0 transition of the Master-ready signal, it removes the data bus and the Slave-ready signal from the bus. This completes the input transfer.

3. Explain handshake control of data transfer during an output operation.

07

Interfacing busses works with the help of a Timechart.



Handshake control of data transfer during output operation

- * An alternative scheme for controlling data transfers on a bus is based on the use of a handshake protocol between the master and the slave.
- * A handshake is an exchange of command and response signals between the master, and the slave.
- * A control line called Master-ready is asserted by the master to indicate that it is ready to start a data transfer. The slave responds by asserting Slave-ready.
- * The master places the output data on the data lines at the

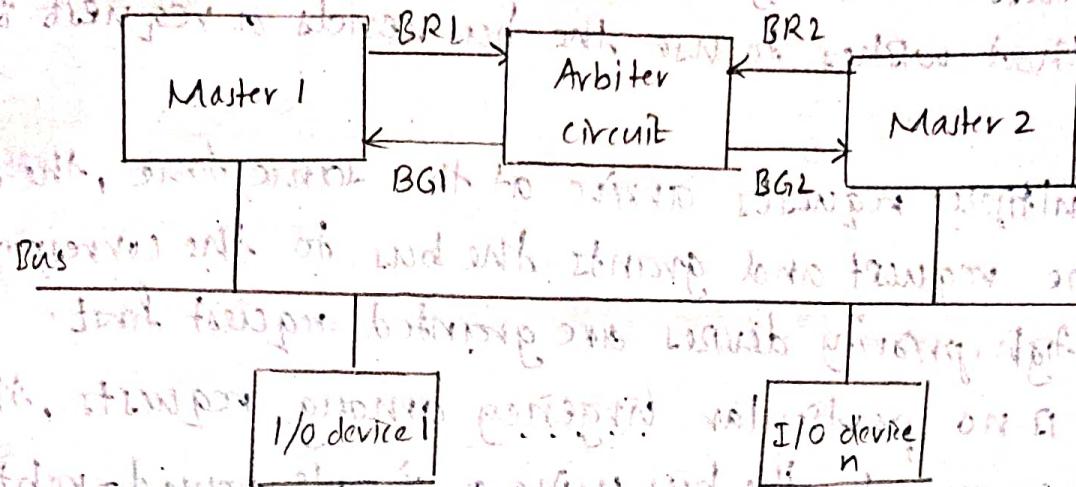
08.

same time that it transmits the address and command information.

- * The selected slave loads the data into its data register when it receives the Master-ready signal and indicates that it has done so by setting the Slave ready signal to 1.
- * t_1 - The master places the address and command information on the bus and at the same time it places the output data on the data lines.
- * t_2 - The master sets the Master-ready line to 1 to inform the device that the address and command information is ready. Sufficient time should be allowed for the device interface circuitry to decode the address. The delay needed can be included in the period t_1 to t_2 .
- * t_3 - The selected slave, ~~has~~ loads the data into its data registers ~~and~~ and indicates that it has done so by setting the Slave ready signal to 1.
- * t_4 - The master removes the address and command information from the bus.
- * t_5 - When the device interface receives the 1-to-0 transition of the Master-ready signal, it removes the data and the Slave-ready signal from the bus. This completes the output transfer.

4. Define the process of Bus Arbitration. Explain with neat diagram.

09



Bus arbitration

Arbitration

- * There are occasions when two or more entities contend for the use of a single resource in a computer system.
- * For example, two devices may need to access a given slave at the same time.
- * The decision is usually made in an arbitration process performed by an arbiter circuit.
- * If it receives two requests at the same time, it grants the use of the slave to the device having the higher priority first.

Bus Arbitration

- * To illustrate the arbitration process, we consider the case where a single bus is shared resource.
- * The device that initiates data transfer requests on the bus is the bus master.
- * It is possible that several devices in a computer system need to be a bus master to transfer data.
- * Since the bus is a single shared facility, it is essential to provide

10

orderly access to it by the bus masters.

- + A device that wishes to use the bus sends a request to the arbiter.
- + When multiple requests arrive at the same time, the arbiter selects one request and grants the bus to the corresponding device - High priority devices are granted request first.
- + If there is no particular urgency among requests, the arbiter may grant the bus using a simple round-robin scheme.
- + In ~~the~~ previous fig. BR1 and BR2 are two bus request lines and BG1 & BG2 are two bus grant lines connecting the arbiter to the masters.
- + When master requests use of bus by activating bus request line.
- + If single request is activated the arbiter activates the corresponding bus grant.
- + When transfer is completed, master deactivates bus request and arbiter deactivates bus grant.

→ Time

11

BR1

BG1 (bus grant 1) asserted after device 1 has completed its operation.

BR2

BG2

BR3

BG3

Granting use of the bus based on priorities

- * Assume that master 1 has highest priority followed by others in increasing numerical order.
- * Master 2 sends request first. Since there are no other requests, arbiter grants the bus by asserting BG2.
- * When master 2 completes operation it deactivates BR2.
- * By the time both 1 and 3 have activated their request lines.
- * Device 1 has higher priority, arbiter activates BG1 after deactivating BG2.
- * When master 1 deactivates BR1, arbiter deactivates BG1 and activates BG3 and grants bus to master 3.

12

5. Define Bus. Explain I/O Interface for an input device with suitable diagram.

Bus is a simple structure that implements the interconnection network

- * Only one source/destination pair units can use this bus to transfer data at any one time.
- * The bus consists of three sets of lines used to carry address, data and control signals.
- * I/O device interfaces are connected to these lines for an input device.

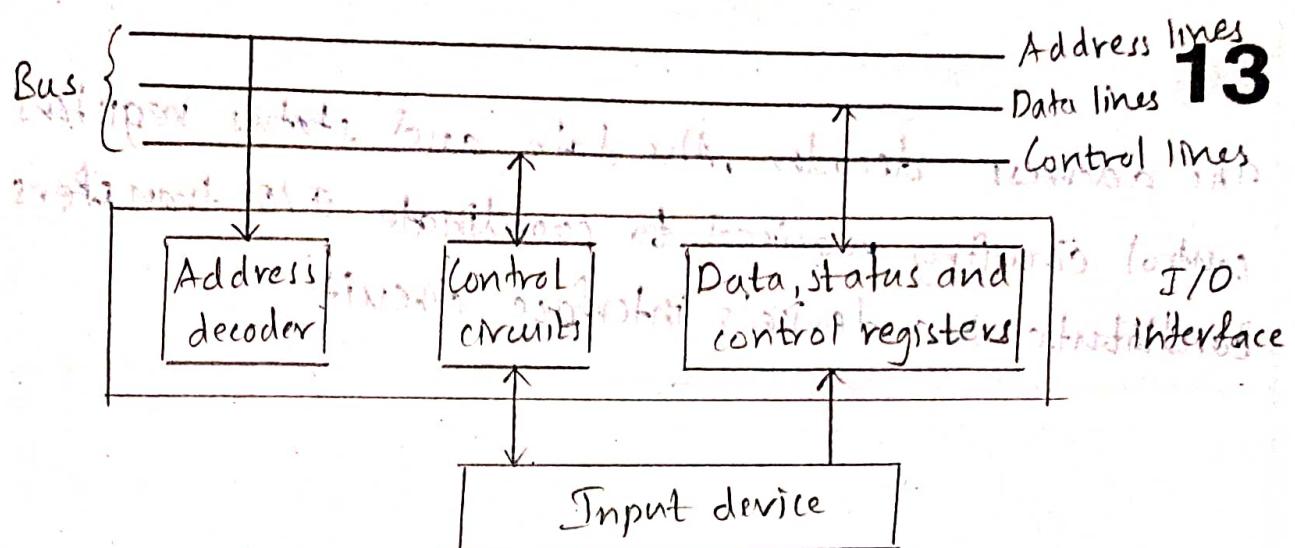


Bus

I/O device 1

I/O interface for an input device

- * Each I/O device is assigned a unique set of addresses for the registers in its interface.
- * When the processor places a particular address on the address lines, it is examined by the address decoders of all devices on the bus.



I/O interface for an input device

- + The device that recognizes this address responds to the commands issued on the control lines.
- + The processor uses the control lines to request either a Read or a Write operation , and the requested data are transferred over the data lines.
- + When the device and the memory share the same address space , the arrangement is called memory-mapped I/O.
- + Any instruction that can access memory can be used to transfer data to or from an I/O device.
- + In a keyboard the instruction
 Load R2, DATAIN
 reads the data from DATAIN and stores them into processor register R2
- + The instruction
 store R2, DATAOUT
 send contents of register R2 to location DATAOUT , which may be a data register of a display device interface.
- + The status and control registers contain information relevant to the operation of the I/O device.

14

- * The address decoder, the data and status registers, and the control circuitry required to coordinate I/O transfers constitute the device's interface circuit.