System Organization.

Scanned by CamScanner

Basic Concepts

The difficulty in transferring information among the units of a computer largely depends on the physical distances separating them. We distinguish two cases:

① Intrasystem communication.
② Intersystem Communication.

⇒ Intrasystem communications is primarily implemented by groups of electrical wires called buses, which support parallel, that is, word-by-word, data transmission.

Buses :

The various processor-level components, CPU, caches main memory, and IO (peripheral) devices within a computer system communicate via buses. The term bus in this context covers not only the physical links among the components, but also the mechanisms for controlling the exchange of signals over the bus.

The system bus consists of three main groups of lines:

① address ② data ③ Control.

The address lines, typically 8 to 32 in number, transmit the addresses of data items stored in the system's main memory or IO address spaces.

Scanned by CamScanner

The data lines, typically 16 to 128 in number, transmit data words over the bus. Finally, the control lines perform such functions as identifying the transaction type (memory read, memory write, IO interrupt, and so forth) and synchronizing communication between fast and slow units.

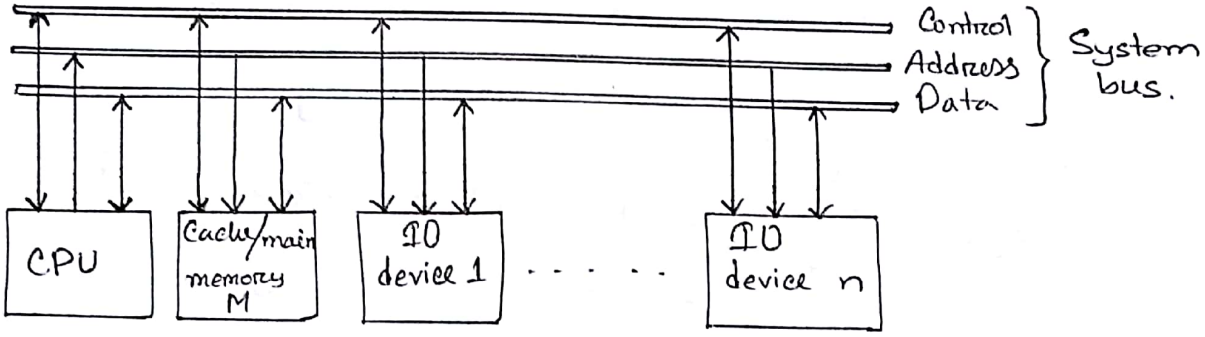


fig: Communication within a computer via a single shared bus.

Computer Networks:

Digital communication networks designed to link m independent computers are called computer networks. Their rationale is to permit sharing of computing ~~resourcing~~ resources (hardware, software, or data) that are widely dispersed.

LAN:

For communication over distances of few kilometers or so - within a single office building, for instance - local-area networks (LANs) are used

Scanned by CamScanner

A LAN is a computer network employing data-transmission links that are private to the network is question.

WAN:

Computer networks spread over large geographical areas, that is, wide-area networks (WANs), use data transmission facilities supplied by telecommunications companies.

Store and Forward:

One such technique is message switching, which uses intermediate switching centers (servers) on long communication path to store messages and subsequently forward them toward the final destination, this process is called store and forward.

ATM:

A type of packet switching called "asynchronous transfer mode" (ATM) combines voice and data communication using short packets that can be transmitted very fast. An ATM packet called a cell consists of a 5-byte header containing the destination address and containing control information, followed by a 48-byte data field.

Scanned by CamScanner

Bus Interfacing:

A significant contributor to the cost of a bus is the number and type of circuits required to transfer signals to and from the bus. A bus line represents a signal path with potentially very large fan-in and fan-out. Consequently, buffer circuits called "bus drivers and receivers" are needed to transfer signals to and from the bus, respectively.

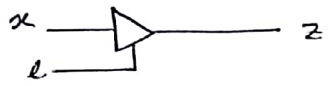
The special transistor circuit technology called tristate logic is often used in bus design. It is characterized by the presence of three signal values 0, 1 and Z, where the third value Z is the "high-impedance state". The binary values 0 and 1 have their usual interpretation and correspond to two specific electrical ~~stage~~ states of a line, such as 0 volts and 3.3 volts. The high-impedance state Z, on the other hand, denotes the state of a line that is electrically disconnected from all voltage sources, that is, an open-circuited or floating line. ① and ⑪ define a tristate buffer.

Tristate logic circuits have two big advantages in the design of shared buses:

⇒ They greatly increase the fan-in and fan-out limits of bus lines, permitting very large numbers of devices to be attached to the same line.

Scanned by CamScanner

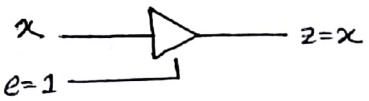
⇒ They support bidirectional transmission over the bus by allowing the same bus connection to serve as an input port and as an output port at different ~~time~~ times.



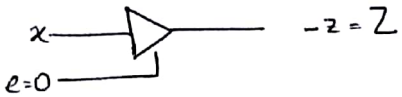
(i)

Inputs		Outputs
x	e	z
0	1	0
1	1	1
0	0	Z
1	0	Z

(ii)



(iii)

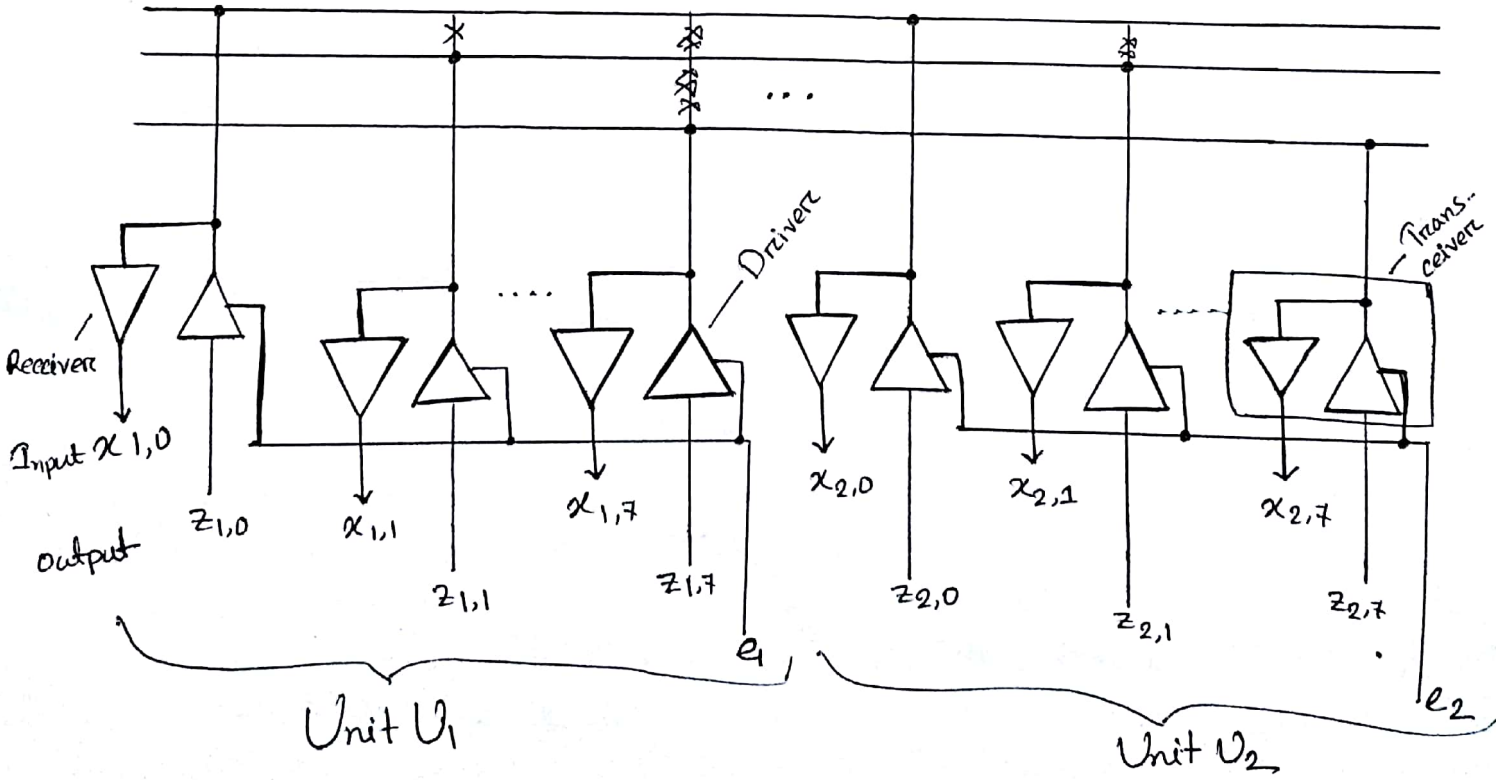


(iv)

fig: (i) logic symbol (ii) truth table (iii) equivalent circuit when enabled.

(iv) equivalent circuit when disabled.

Use of tristate logic for bus interfacing:



Scanned by CamScanner

figure shows how we use tristate logic to interface two units U_1 and U_2 to a set of bidectional bus lines. If $e_1 = 1$ and $e_2 = 0$, then U_1 controls or devices the bus lines in question; information is transferred over the bus from U_1 to U_2, in effect making $x_{2,i} = z_{1,i}$ for all i. Conversely, if $e_1 = 0$ and $e_2 = 1$, then U_2 drives the bus and information is transferred in the opposite direction from U_2 to U_1, making $x_{1,i} = z_{2,i}$ for all i. If $e_1 = e_2 = 0$, then the outputs of both U_1 and U_2 are logically disconnected from the bus and impose only a minuscule electrical load on it. The combination $e_1 = e_2 = 1$ is invalid, because it applies two different signals simultaneously to each bus line making the line's state indeterminate. Proper operation of the bus requires that at most one driver connected to each bus line be enabled at any time.

Scanned by CamScanner

Bus arbitration :

The possibility exists that several master or slave units connected to a shared bus will request access to the bus at the same time. A selection mechanism called "bus arbitration" is therefore required to enable the current master, which we will refer to as the bus controller, to decide among such competing requests.

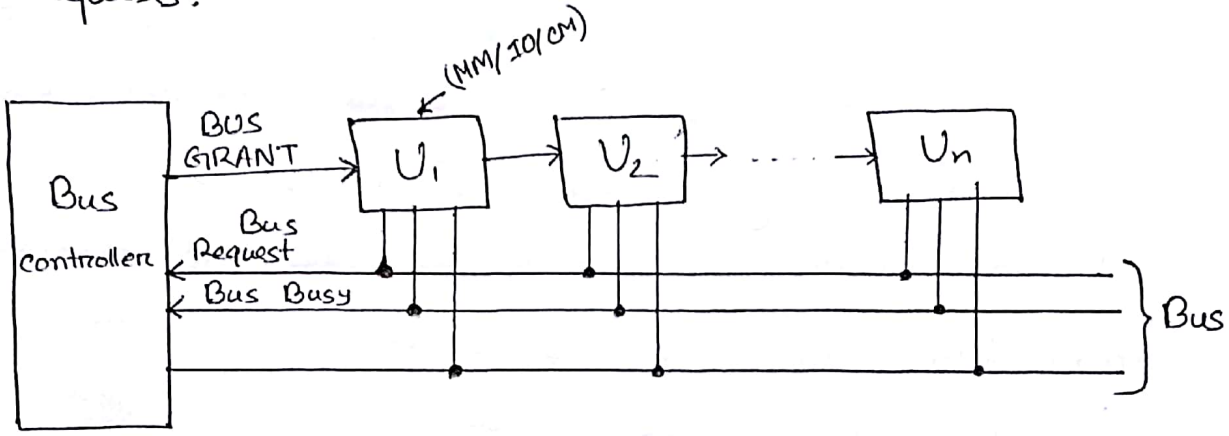


fig: Bus arbitration using daisy chaining.

figure illustrates daisy-chaining arbitration. This method involves three control signals to which we assign the generic names BUS REQUEST, BUS GRANT, and BUS BUSY All the bus units are connected to the BUS REQUEST line. When activated, it merely serves to indicate that one or more units are requesting use of the bus. The bus controller responds to a BUS REQUEST signal only if BUS BUSY is inactive. This response takes the form of a signal placed on the BUS GRANT line. On receiving the BUS GRANT signal, a requesting unit enables its

Scanned by CamScanner

physical bus connections and activates BUS BUSY & for the duration of its new bus activity.

The main distinguishing feature of daisy chaining is the BUS GRANT signal is distributed; it is connected serially from unit as shown in figure. When the first unit requesting access to the receives BUS GRANT, it blocks further propagation of that signal, actives BUS BUSY, and begins to use the BUS. When a nonrequesting unit receives the BUS GRANT signal, it forwards the signal to the next unit. Thus if two units simultaneously request bus access, the one closer to the bus controller, that is, the one that receives BUS GRANT first, gains access to the bus. Selection priority is therefore determined by the order in which the units are linked (chained) by the BUS GRANT lines.

IO control methods:

If such operations are completely controlled by the CPU, that is, the CPU executes programs that initiate, direct, and terminate the IO operations, the computer is said to be using "programmed IO". This type of IO control can be implemented with little or no special hardware, but causes the CPU to spend a lot of time performing relatively trivial IO-related functions. One such function is testing the status of IO devices to determine if they require servicing by the CPU.

Scanned by CamScanner

DMA:

The CPU and IO controller interact only when the CPU must yeild control of the memory bus to the IO controller in response to requests from the latter. This level of IO control is called "direct memory access" (DMA), and the IO device interface control circuits is called a DMA controller.

- The CPU must perform the following steps to determine the status of an IO device:

1. Read the IO device's status bit.
2. Test the status bit to determine if the device is ready to begin transferring data.
3. If not ready, return to step 1; otherwise, proceed with the data transfer.

Interrupts:

The word "interrupt" is used in a board sense for any infrequent or exceptional event that causes a CPU to temporarily transfer control from its current program to another program - an interrupt handler - that services the event in question.

Scanned by CamScanner