COMPUTER SCIENCE

Computer Organization and Architecture

Secondary Memory

Lecture_03

Vijay Agarwal sir

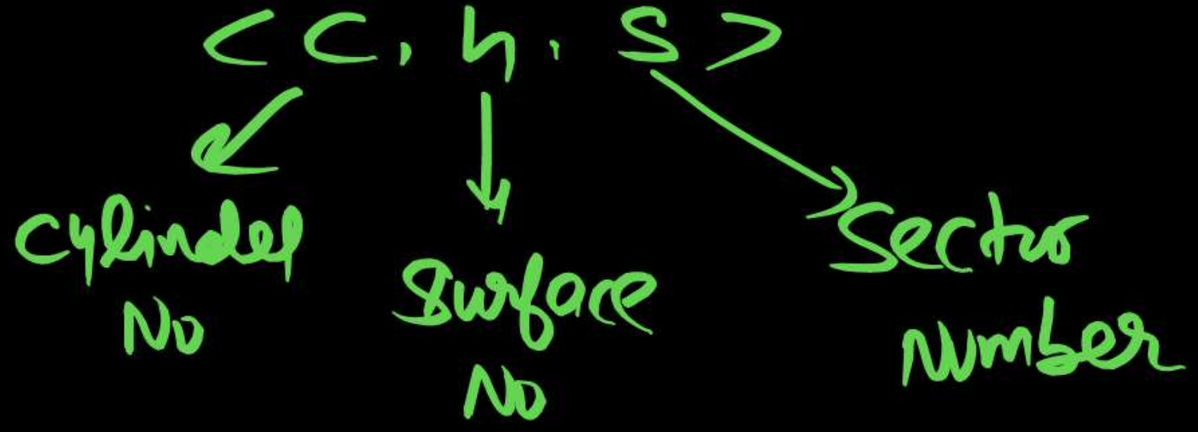
01 IO Organization

Secondary Memory

1. Disk Concept
2. Disk Structure
   - Platter
     - Surface
       - track
         - Sector
           - DATA
3. Capacity of Disk
4. Disk Access time
   - Seek Time
   - Rotational Latency
   - D.T.T
   - & Data transfer Rate

Disk Addressing

Concept of Cylinder, Surface



How to Calculate Sector Number when

<C, h, S>

How to calculate Cylinder Number. Surface No

When Sector is given.

(C, h, S)

$$\text{Sector Number} = S + ST * h + (ST * TC) * C$$

SC

ST: # sector / track

TC: # Track / Cylinder

SC ⇒ # Sector per Cylinder.

Eg 4 Surface
8 Sector
sector per Cylinder = 8 × 4
= 32 Sector Per Cylinder

In GATE Question 63 sector
20 surface (# Track/Cylinder)

400 × 63 × 20 (1260 sector per Cylinder) ⇒ Cylinder 400 × 1260

Now Start

Some More Concept Regarding Disk.

Disk Access Time = S.T + R.L + D.T.T + overHead.

(i) When we have sequential Access then x Byte

No Need of additional S.T & R.L

x Byte * D.T.T

1 sector = 512B
4 × 512 = 2kB
4 Consective sector

2kB → D.T.T

(ii) When there is a Randomly Access then

D.A.T = x * (S.T + R.L + D.T.T)

x → Library

MCQ

An application loads 100 libraries at start-up. Loading each library requires exactly one disk access. The seek time of the disk to a random location is given as 10 ms. Rotational speed of disk is 6000 rpm. If all 100 libraries are loaded from random locations on the disk, how long does it take to load all libraries? (The time to transfer data from the disk block once the head has been positioned at the start of the block may be neglected.)

100 X 1 Library D.A.T (S.T+R.L+D.T)

[GATE-2011-CS: 2M]

A 0.50s

B 1.50s

C 1.25s

D 1.00s

NAT

Consider a disk pack with a seek time of 4 milliseconds and rotational speed of 10000 rotations per minute (RPM). It has 600 sectors per track and each sector can store 512 bytes of data. Consider a file stored in the disk. The file contains 2000 sectors. Assume that every sector access necessitates a seek, And the average rotational latency for accessing each sector is half of the time for one complete rotation. The total time (in milliseconds) needed to read the entire file is ___.

[GATE-2015(Set-1)-CS: 2M]

$2000 \times (x)$

NAT

Consider a typical disk that rotates at 15000 rotations per minute (RPM) and has a transfer rate of 50×10^6 bytes/sec. If the average seek time of the disk is twice the average rotational delay and the controller's transfer time is 10 times the disk transfer time, the average time (in milliseconds) to read or write a 512-byte sector of the disk is _____.

[GATE-2015(Set-2)-CS: 2M]

Q. If the disk is rotation at 3600 rpm, determine the effective data transfer rate which is defined as the number of bytes transferred per second between disk and memory. (Given size of track = 512 bytes)

[GATE-: 2 Marks]

3600 Rotation Per minute (60 sec)

1 Rotation = $\frac{60}{3600} = \frac{1}{60}$ sec.

In One Rotation One Track Transversed (Data transferred)

$\frac{1}{60}$ sec ——— 512 Byte

1 sec ——— 512 X 60 BPS ⇒ 512 X 2 X 30 ⇒ 30 KBPS

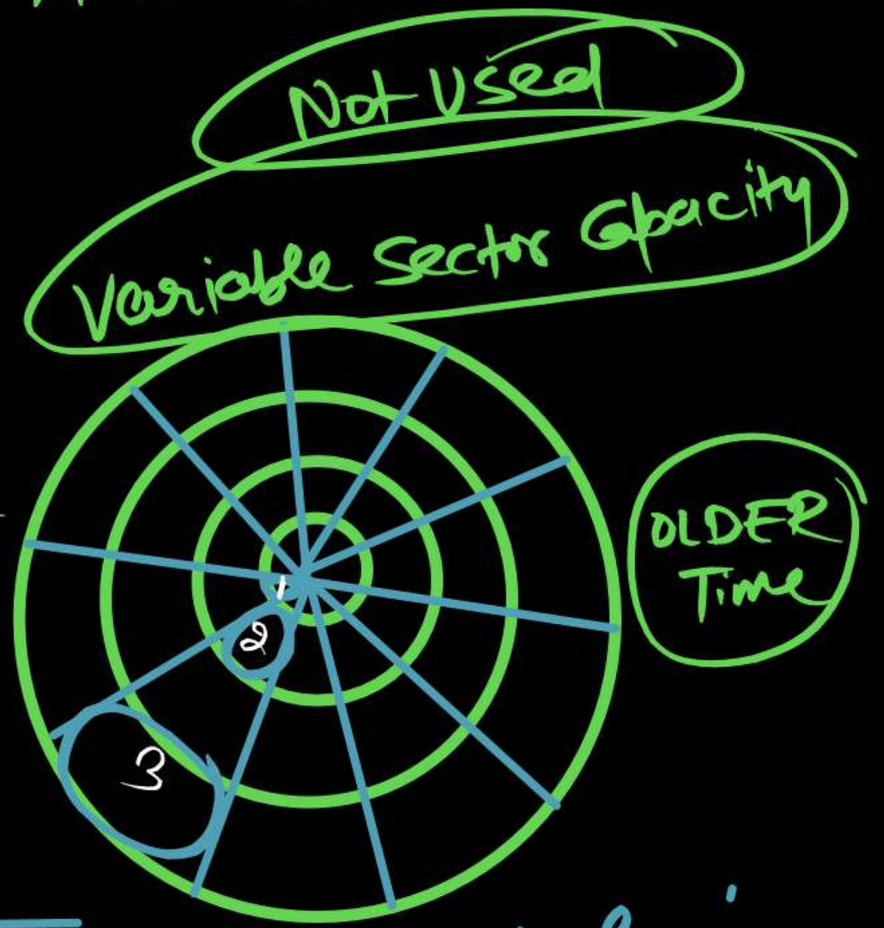
In a Disk, Data is Stored in a Sector.

In Digram

Sector 1 : Capacity is small

sector 2: Capacity is large

Sector 3 : Capacity is very-very large



Inner Most Sector have the Smallest Capacity

Out Most Sector have very large Capacity

$\times$ Linear Velocity / Angular Velocity

Time $\propto$ track size (it Varies for Inner most & outer most Track)

Generally we (in Todays time)

① Fixed sector Capacity : Storage Density is Variable

② Variable Sector Capacity : Storage Density is Fixed.
(Inner most sector small size
& outermost large size)

Fixed Sector Capacity : Outermost & Inner Most Sector have the same Size (So Density variable)

(eg) Mumbai Dharavi & Village Farm House (1 Acre)
Mumbai Dadar

Input-Output Interface

WHY Need of IO Interface ?

CPU is Electronic Device (Fastest) & I/O Devices are Electro magnetic (very slowest)

So there is a Difference in term of

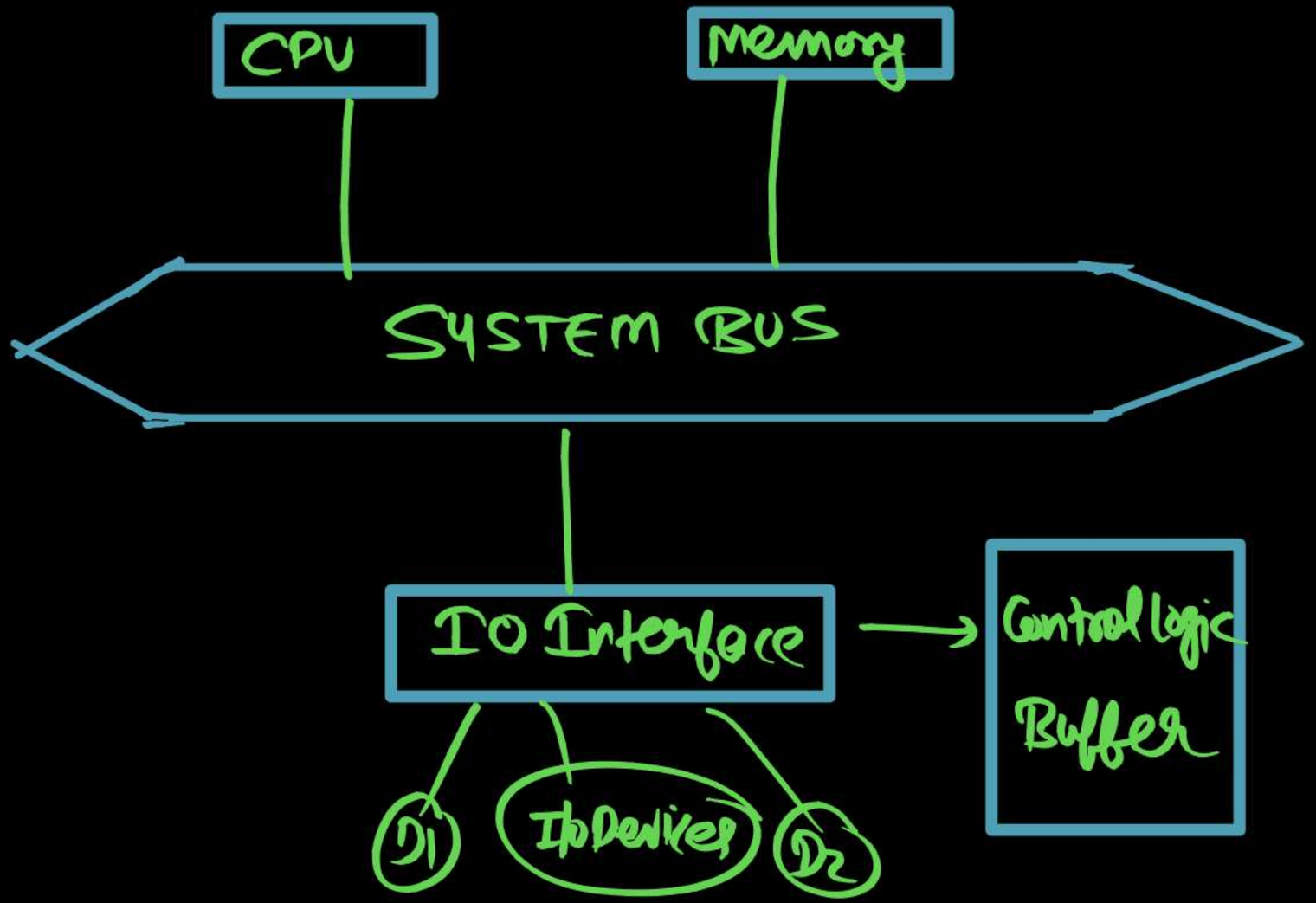
- Speed
- Data format
- Data transfer Rate
- Operating Modes

Input-Output Interface

So To Synchronize the I/O Device Speed with Processor (CPU) Speed High Speed Inface chip called I/O Interface or I/O Module is used.

IO Interface is Responsible for All IO Operation. So in the System Design, IO Device are Connected with the Help of I/O Interface to CPU.

Input-Output Interface



CPU taking the IO Data from the IO Interface Chip.

Different IO Interface chips are used.

PIN's
- 8251 USART (Universal Synchronous Asynchronous Receiver Transmitter)
- 8255 PPI (Programmable Peripheral Interface)
- 8237/8257 DMA
- 8259 Interrupt Controller.

Input-Output Interface

Input-output interface provides a method for transferring information between internal storage (main memory) and external I/O devices. Peripherals connected to a computer need special communication links for interfacing them with the central processing unit. The purpose of the communication link is to resolve the differences that exist between the central computer and each peripheral. The major differences are:

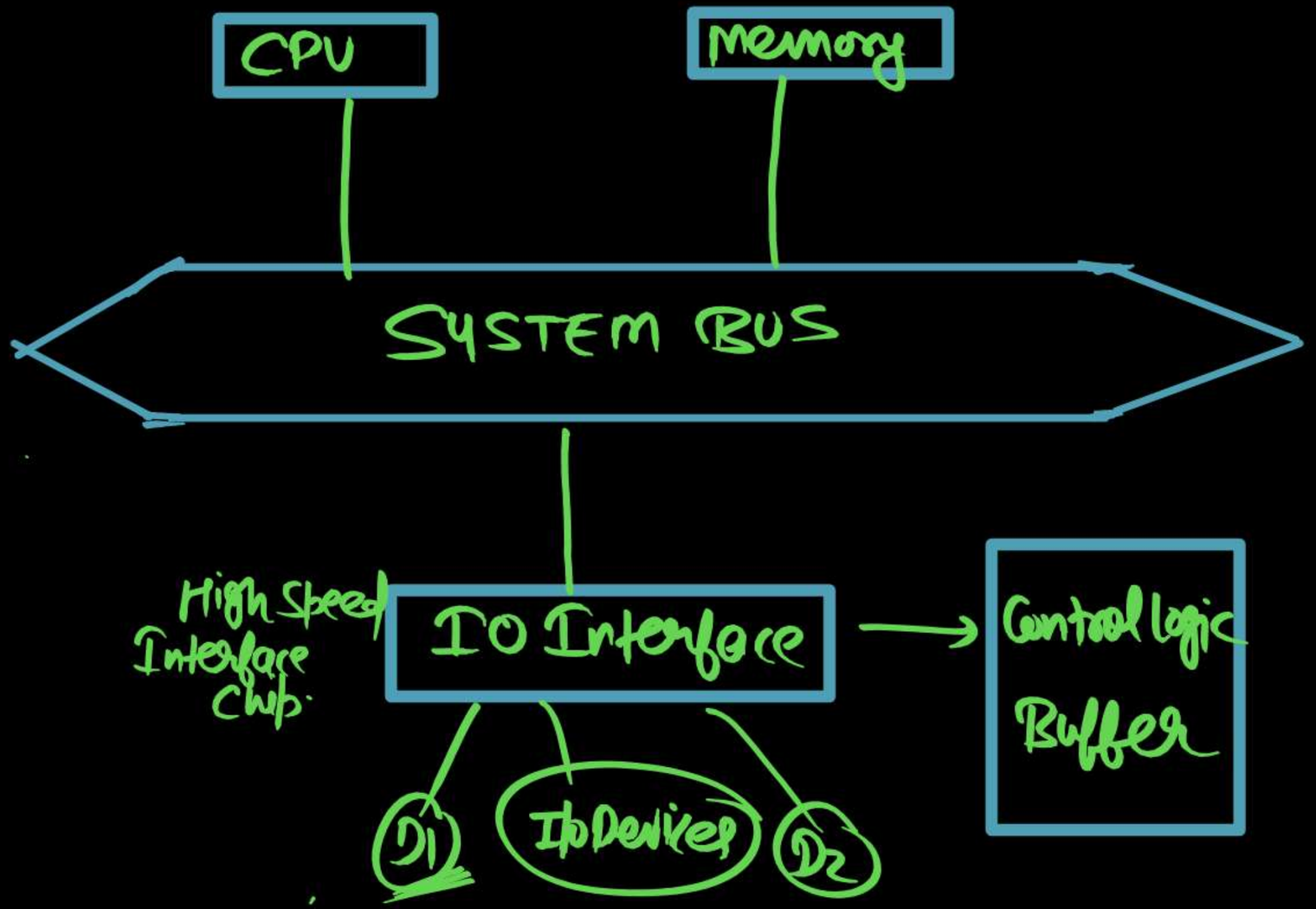
- Peripherals are electromechanical and electromagnetic devices and their manner of operation is different from the operation of the CPU and memory, which are electronic devices. Therefore, a conversion of signal values may be required.
- The data transfer rate of peripherals is usually slower than the transfer rate of the CPU, and consequently, a synchronization mechanism may be needed.

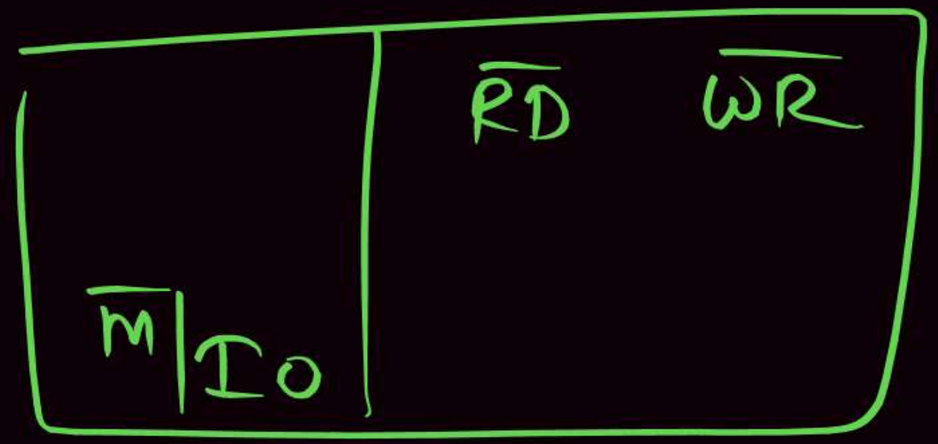
- Data codes and formats in peripheral differ from the word format in the CPU and memory.
- The operating modes of peripherals are different from each other and each must be controlled so as not to disturb the operation of other peripherals connected to the CPU.

Modes of Transfer

1. Programmed I/O
2. Interrupt-initiated I/O
3. Direct memory access (DMA)

Input-Output Interface





0 : Memory Address

1 : Io Port Address

Working Process

① CPU initialize the IO Interface chip Along with the IO Command & then CPU go & Performing the other work.

② IO Interface chip Control Logic Interpret the IO Command & Accordingly IO Port will be Enabled for the operation.

③ Based On the Speed of IO Device & Amount of Data to be transfer Consume some time [Preparation time], then Data is transfer from I/O Device to IO Interface Buffer.

④ When the Data is available in the Buffer then IO Interface generate the Interrupt Signal & send to the CPU. & waiting for Acknowledgement.

⑤ After Receiving the Acknowledgment from the CPU, IO Interface Chip transfer the Data to the CPU.

⑥ So CPU will be Accessing the Data from the IO Interface chip So time will save & (Best Performance).

(∵ IO Interface chip is Fastest)

IO Transfer Mode

1. Programmed Io (Program Driven Io)
2. Interrupt Driven IO
3. DMA [Direct Memory Access]

↓↓

Highest Priority

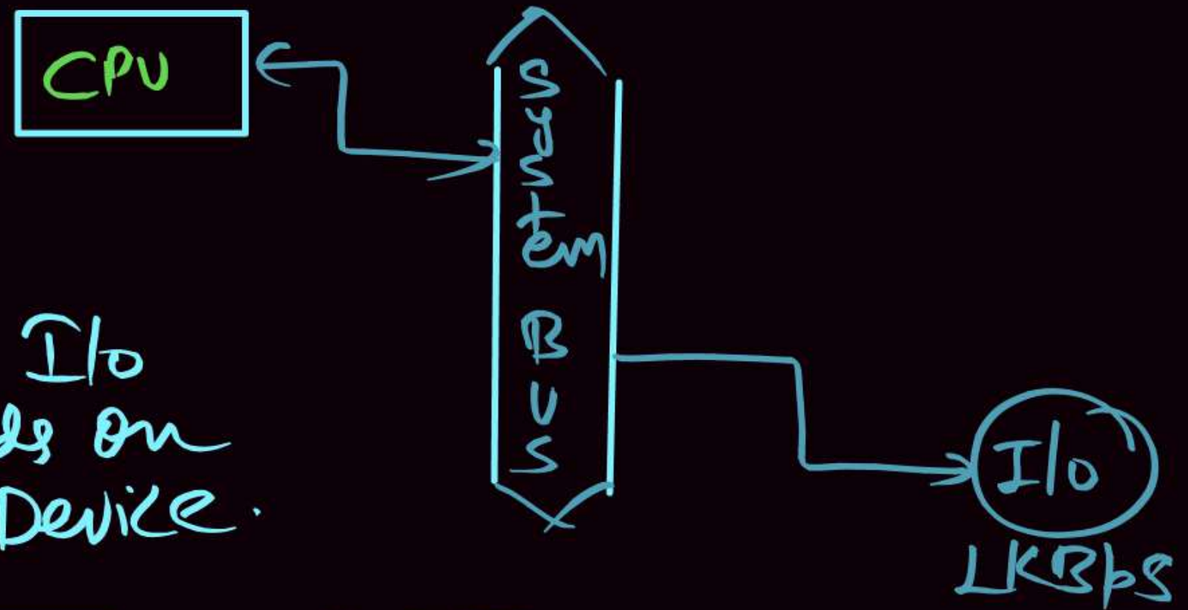
① Program Driven I/O

- In the Program Driven IO there is No High Speed Interface Chip is Using between the IO Devices & other Component of Computer [CPU, Memory]
- In this CPU is Responsible for the IO operation.

Drawback

- CPU Utilization is very - very In Efficient [very - very Poor]
- CPU Time Depends on Speed of IO Device.
- CPU will Wait Untill the IO operation Complete.

(eg) 1KBPS I/o Device is Connected to CPU Using Programmed I/O. & Data is transfer Byte Wise then CPU Time for 1 Byte ?



Soln

In Programmed I/o CPU time Depends on Speed on IO Device.

CPU Time (1 Byte transfer time)	= 1 msec.

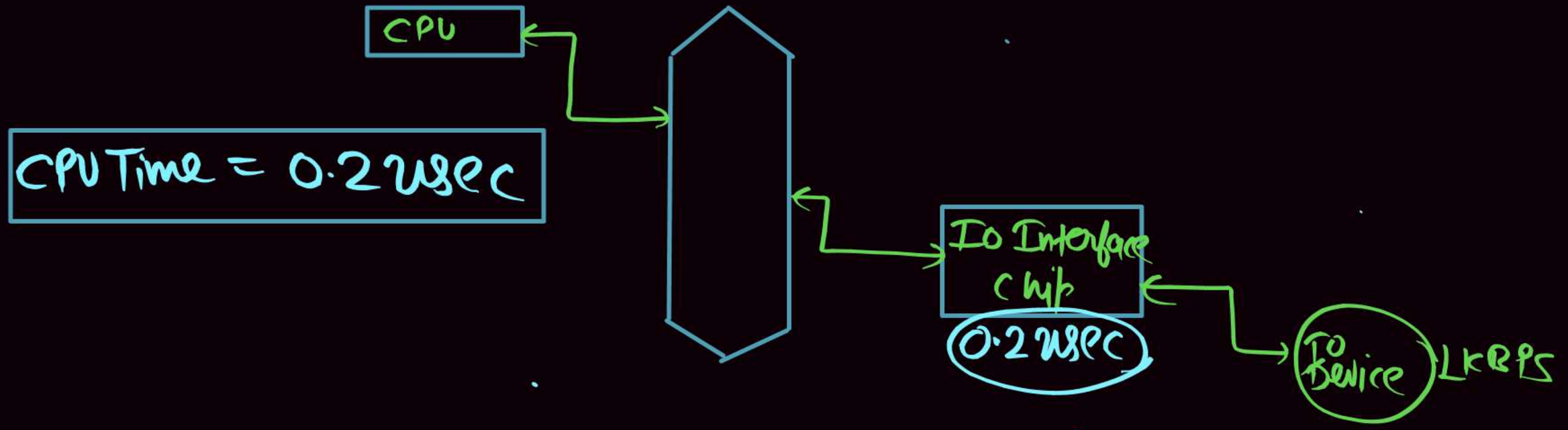
1KBPS

1 kilo Byte ——— 1 Sec

1 Byte ——— $\frac{1}{K} = \frac{1}{10^3} = 10^{-3}$ = 1 millisec

② Interrupt Driven IO

In Interrupt Driven IO High Speed Interface chip is used between the IO Device & other component of (CPU) the Computer.



In the Interrupt Driven IO CPU Time is Depends on Latency of IO Interface chip Rather than IO Device

Advantage :

. CPU Utilization is effiicent

. CPU Not Waiting. Bcz Here CPU time Dcpands on High speed Interface Logic.

A device with data transfer rate 10 KB/sec is connected to a CPU. Data is transferred byte-wise. Let the interrupt overhead be 4 usec. The byte transfer time between the device interface register and CPU or memory is negligible. What is the minimum performance gain of operating the device under interrupt mode over operating it under program controlled mode?

(a) 15 (b) 25 (c) 35 (d) 45

[GATE-2005 : 2 Marks]

$P \propto \frac{1}{ET}$

$10\,KB \longrightarrow 1\,sec$

$1\,Byte \longrightarrow \frac{1}{10k}\,sec = \frac{10^4 \times 100}{100} = 100 \times 10^{-6}$

$ET_{ProgIO} = 100\,usec$

$ET_{InterruptIO} = 4\,usec.$

$S = \frac{Performance\ of\ InterIO}{Perf\ of\ ProgIO}$

$\Rightarrow \frac{ET_{ProgIO}}{ET_{InterruptIO}}$

$= \frac{100}{4} = 25$

NAT

The size of the data count register of a DMA controller is 16 bits. The processor needs to transfer a file of 29, 154 kilobytes from disk to main memory. The memory is byte addressable. The minimum number of times the DMA Controller needs to get the control of the system bus from the processor to transfer the file from the disk to main memory is____.

[GATE-2016(Set-1)-CS: 2M]

MCQ

On a non-pipelined sequential processor, a program segment, which is a part of the interrupt service routine, is given to transfer 500 bytes from an I/O device to memory.

Initialize the address register
Initialize the count to 500
LOOP: Load a byte from device
Store in memory at address given by address register
Increment the address register
Decrement the court
If count! = 0 go to LOOP

Assume that each statement in this program is equivalent to a machine instruction which takes one clock cycle to execute if it is a non-load/store instruction. The load-store instructions take two clock cycles to execute.

The designer of the system also has an alternate approach of using the DMA controller to implement the same transfer. The DM controller requires 20 clock cycles for initialization and other overhead. Each DMA transfer cycle takes two clock cycles to transfer one byte of data from interrupt driven program-based input-output?

[GATE-2011-CS: 2M]

A 3.4

B 4.4

C 5.1

D 6.7

COA

1. Introduction of COA
2. Machine Instⁿ & AM.
3. Floating Point Representation.
4. ALU, Data Path & Control Unit
5. Pipelining & its Hazard
6. Cache Memory
7. Secondary Memory & IO Interface

THANK YOU!