CS & IT ENGINEERING

COMPUTER ORGANIZATION AND ARCHITECTURE

IO Organization

Lecture No.- 03

By- Vishvadeep Gothi sir

Recap of Previous Lecture

- **Topic** Peripheral Device
- **Topic** IO vs Memory Buses
- **Topic** Memory Mapped IO vs IO Mapped IO
- **Topic** Asynchronous Data Transfer
- **Topic** Modes of Transfer

Topics to be Covered

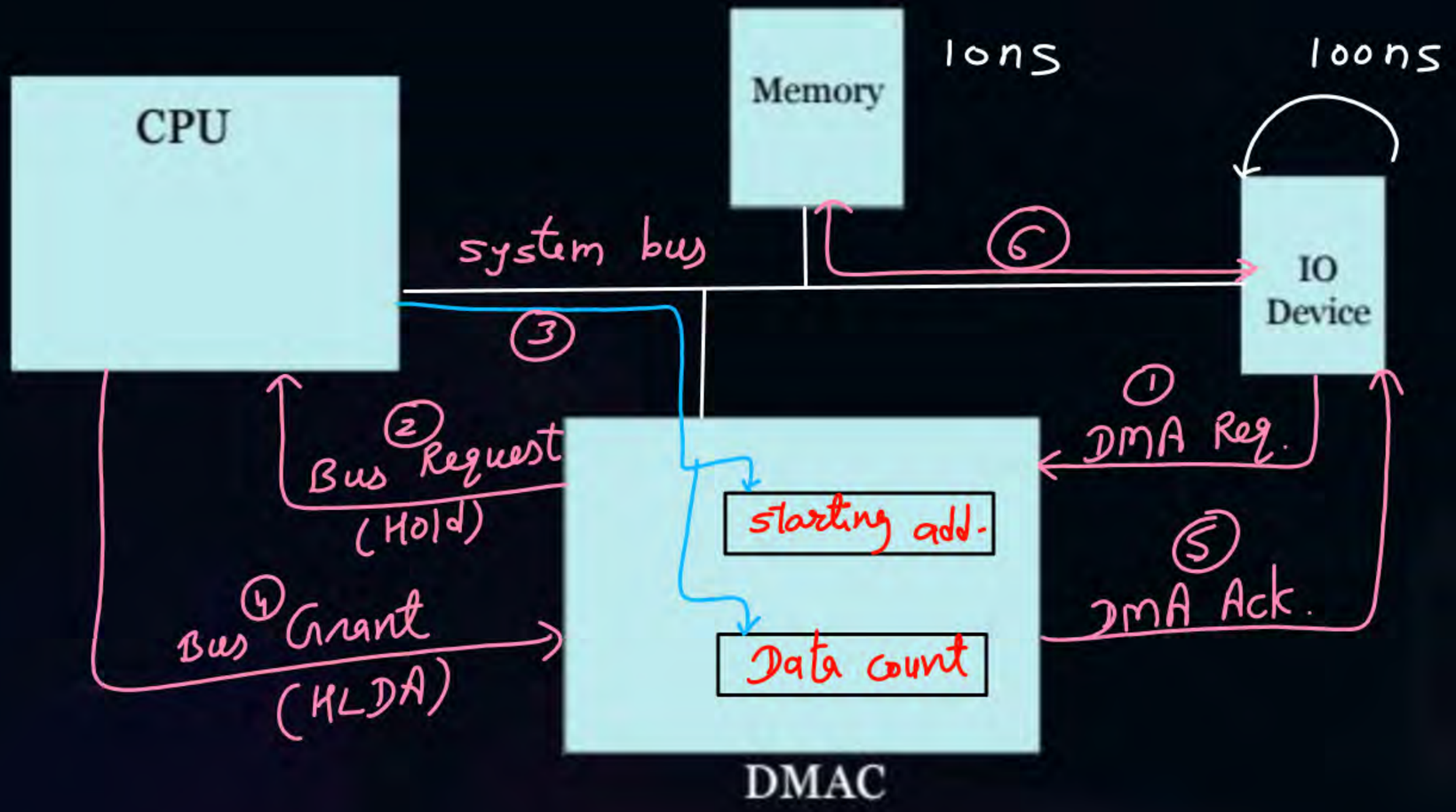
Topic DMA

Topic Cycle Stealing

Topic : DMA

- Enables data transfer between I/O and memory without CPU intervention
- Need a hardware: DMAController

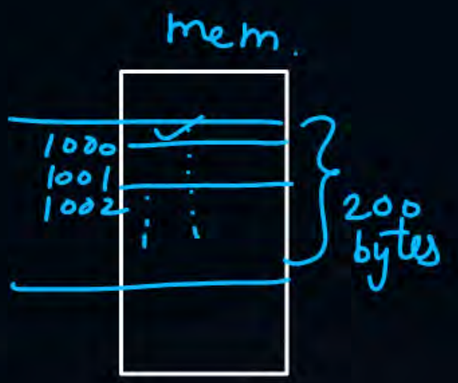
Topic : DMA


Memory
10ns
100ns
CPU
system bus
IO Device
6
3
1
DMA Req.
2
Bus Request
(Hold)
starting add.
5
DMA Ack.
4
Bus Grant
(HLDA)
Data count
DMAC

Topic : DMA

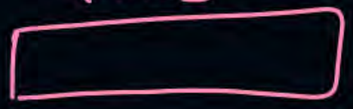
1. Starting Address — mem address from where data transfer should start

2. Data Count — No. of bytes or words to be transferred
   - for byte addressable mem.
   - for word addressable mem.



	Initially	After 1 Byte	After 2 Byte	After 3 Byte	After 4 Byte	
Start add.	1000	1001	1002	1003	1004	...
Data count	200	199	198	197	196	... 0

when data count becomes 0 then transfer stops.

Data count = 

max value of data count = $(1111)_2 = (15)_{10}$

max no. of bytes transferred with one initializat^n of DMA = 15

⇒ DMAC is a special purpose processor, because it generates address and control signals for data transfer between I/O & mem.

⇒ During DMA transfer CPU's status (Reg. values) will be as it is.

when DMA uses buses for data transfer then CPU can perform only those operations which do not require system buses.
which means CPU will be blocked mostly.

Topic : Modes of DMA Transfer

when CPU gives control of the buses to DMA and it takes back control.

Topic : Modes of DMA Transfer

1. Burst Mode
2. Cycle Stealing
3. Interleaving DMA

Topic : Modes of DMA Transfer

Burst Mode :

when CPU gives control of the buses to DMAC, then a block of data is transferred.

1 block ⇒ 512B to 4KB

Topic : Modes of DMA Transfer

Cycle Stealing :

Slow I/O devices takes time to prepare data internally. Till then CPU keeps control of buses.

When data is ready, then DMAC steals CPU cycles and takes control of buses to transfer prepared data to mem.

Topic : Modes of DMA Transfer

Interleaving DMA:

CPU gives control of the buses to DMA when CPU does not need it.

exc:- during ALU operation, inst^{n} decode.

Time for which CPU is blocked due to DMA is almost zero.

Time needed to prepare data in I/O = t_x ← depends on I/O speed

— " — transfer data to mem = t_y ← depends to mem speed

% of time CPU is blocked due to DMA (burst mode) = $\frac{t_y}{t_x + t_y} * 100\%$

— " — (cycle stealing) = $\frac{t_y}{t_x} * 100\%$

[NAT]

Ans = 12.5

#Q. Consider a device operating on 1MBPS speed and transferring the data to memory using cycle stealing mode of DMA. If it takes 2 microseconds to transfer 16 bytes data to memory when it is ready/prepared. Then percentage of time CPU is blocked due to DMA is?

$t_y = 2\ \mu sec$

$t_{xc} = 16\ \mu sec$

for 1MB, preparat^n time = 1sec

for 16B —"— = $\frac{1\ sec}{1\ MB} * 16\ B$

$= 16\ \mu sec$

% of time CPU blocked due to DMA = $\frac{2\ \mu s}{16\ \mu s} * 100\%$

$= 12.5\%$

Ques) % of time CPU blocked = 20%

2MBPS
cycle stealing
Data to prepare at once = 8B
ty = _____ μsec ?

$$20\% = \frac{ty}{4\ \mu sec} * 100\%$$

$$ty = \frac{20}{25}$$

$$= 0.8\ \mu sec$$

for 2MB, time = 1sec

for 8B, —11— = $\frac{1\ sec}{2\ MB} * 8B$ (4)

= 4 μsec

Ques) $t_y = 10$ μsec

% of time CPU blocked = 5%

$t_x =$ ____ μs ?

cycle stealing

Soln

$$5\% = \frac{10\ \mu sec}{t_x} * 100\%$$

$$t_x = \frac{1000\ \mu sec}{5}$$

$$= 200\ \mu sec$$

Ques)

In this quest^n if data prepared at once is 4 bytes then speed of I/O is ?

Soln

In 200 μsec, data prepare = 4B

in 1 sec, —11— = $\frac{4B}{200 * 10^{-6} sec}$

$= 2 * 10^4$ B/sec

$= 20$ KB/sec

[NAT]

Ans = 80000

GATE – 2021

#Q. Consider a computer system with DMA support. The DMA module is transferring one 8-bit character in one CPU cycle from a device to memory through cycle stealing at regular intervals. Consider a 2 MHz processor. If 0.5% processor cycles are used for DMA, the data transfer rate of the device is bits per second?

$t_y = 1 \text{ cycle} = \frac{1}{2\text{MHz}} = 0.5 \text{ usec}$

$0.5\% = \frac{0.5 \text{ usec}}{t_x} * 100\%$

$t_x = 100 \text{ usec}$

in 100 usec, data = 8 bits

in 1 sec, —11— = $\frac{8 \text{ bits}}{100 * 10^{-6} \text{ sec}} = 80000 \text{ bits/sec}$

[NAT]

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

```
        Initialize the address register   — 1 ]
        Initialize the count to 500       — 1 ]   — 2
LOOP:   Load a byte from device           — 2 ]
        Store in memory at address given by address register — 2 ]   7 * 500
        Increment the address register    — 1 ]
        Decrement the count               — 1 ]
        If count !=0 go to LOOP           — 1 ]
```

500 times ←

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 DMA controller requires 20 clock cycles for initialization and other overheads. Each DMA transfer cycle takes two clock cycles to transfer one byte of data from the device to the memory.

What is the approximate speed up when the DMA controller-based design is used in a place of the interrupt driven program-based input-output?

Interrupt time $= 2 + (500 * 7) = 3502$

DMA time $= 20 + (2 * 500) = 1020$

$$\text{speed up} = \frac{\text{Interrupt I/O time}}{\text{DMA time}} = \frac{3502}{1020}$$

$\cong 3.4$

[NAT]

#Q. The DMA controller has data count register of size 8-bits. The memory is byte addressable. The maximum number of bytes the DMA can transfer to memory at a time without giving the control of the buses back to CPU?

max value of 8 bits = $(11111111)_8 = (255)_{10}$

Ans = 255

[NAT]

#Q. The DMA controller has data count register of size 8-bits. The memory is byte addressable.

1. Minimum how many times DMA needs to take control from CPU to transfer a file of 500 bytes?
2. Minimum how many times DMA needs to take control from CPU to transfer a file of 15K bytes?

1. $\left\lceil \frac{500\ B}{(2^8-1)\ B} \right\rceil = 2$

2. $= \left\lceil \frac{15 * 1024\ B}{255} \right\rceil = 61$

[NAT]

GATE-PYQ

#Q. The size of the data count register of a DMA controller is 16bits.

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 456 .

$$= \left\lceil \frac{29154 * 1024\,B}{(2^{16}-1)\,B} \right\rceil = 456$$

[MCQ]

$t_x = t_y$

#Q. If a word preparation time in IO device and word transfer time to memory from IO device are same. Then?

cycle stealing, % of time CPU blocked $= \frac{t_y}{t_x} * 100\%$
$= 100\%$

A ✓ 100% time CPU is blocked due to DMA in cycle stealing mode

B 50% time CPU is blocked due to DMA in cycle stealing mode

C 100% time CPU is blocked due to DMA in burst mode

D ✓ 50% time CPU is blocked due to DMA in burst mode

for burst mode. $= \frac{t_y}{t_x + t_y} * 100\% = 50\%$

[MCQ] [2024 : 1M]

#Q. Which one of the following statements is FALSE?

A In the cycle stealing mode of DMA, one word of data is transferred between an I/O device and main memory in a stolen cycle

B The CPU can start executing an interrupt service routine faster with vectored interrupts than with non-vectored interrupts

C For bulk data transfer, the burst mode of DMA has a higher throughput than the cycle stealing mode

D Programmed I/O mechanism has a better CPU utilization than the interrupt driven I/O mechanism

[MCQ] [2020 : 1M]

#Q. Consider the following statements:

I. Daisy chaining is used to assign priorities in attending interrupts.

II. When a device raises a vectored interrupt, the CPU does polling to identify the source of interrupt.

III. In polling, the CPU periodically checks the status bits to know if any device needs its attention.

IV. During DMA, both the CPU and DMA controller can be bus masters at the same time.

Which of the above statements is/are TRUE?

A I, II only

B I and IV only

C I and III only

D III only

#Q. A keyboard connected to a computer is used at a rate of 1 keystroke per second. The computer system polls the keyboard every 10 ms (milli seconds) to check for a keystroke and consumes 100 μs (micro seconds) for each poll. If it is determined after polling that a key has been pressed, the system consumes an additional 200 μs to process the keystroke. Let T_1 denote the fraction of a second spent in polling and processing a keystroke.

In an alternative implementation, the system uses interrupts instead of polling. An interrupt is raised for every keystroke. It takes a total of 1 ms for servicing an interrupt and processing a keystroke. Let T_2 denote the fraction of a second spent in servicing the interrupt and processing a keystroke.

The ratio $\frac{T_1}{T_2}$ is 10.2. (Rounded off to one decimal place)

$$\frac{T_1}{T_2} = \frac{10.2\ ms}{1\ ms} = 10.2$$

Soln

no. of polls in 1 sec = $\frac{1\,sec}{10\,msec} = 100$

Total time in polling per sec = $100 * 100\ \mu sec$

= 10 msec

keystroke processing time = 200 μsec = 0.2 ms

$T_1 = 10 + 0.2$

$= 10.2$ ms

$T_2 = 1$ msec

2 mins Summary

Topic DMA

Topic Cycle Stealing

Happy Learning

THANK - YOU