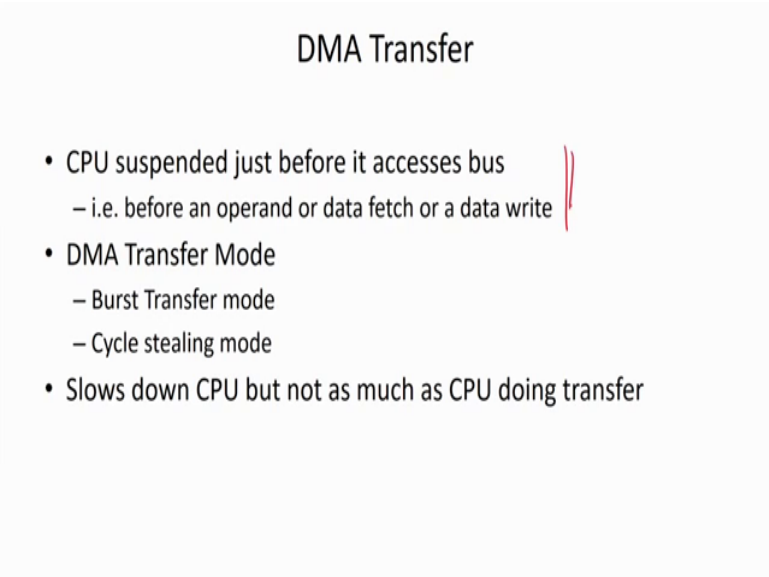
So, this is the difference you must remember it. So, in case of interrupt driven I/O, here is a change of context. In case of DMA transfer there is no context change, the context of the processor remains same whatever program it is executing, it is still going to execute that particular program.

(Refer Slide Time: 26:03)



So, how we are going to do? CPU suspended just before it accesses the bus ok. We will see before an operation operand or data fetch or a data write, we will see this thing. I will explain these things, what says that I am saying that there is no context change ok. So, now, I can I can draw these things and this is the processor, this is the main memory and this is the DMA controller. Now the system bus is given to the DMA controller. Now processor cannot access the main memory ok. This is the situation, there is no change of context; that means, processor can carry out its own work.

Now what it says then. Now CPU suspended just before it accessed the bus. Now we are having some information inside the processor, now processor is going to carry out this particular information, carry out those particular tasks. So, it's going to execute some instruction, but at some point of time, now processor needs some information from memory, but bus is now with this particular DMA controller processor cannot access this.

So, processor cannot fetch this particular data. Since processor cannot fetch this particular data, so processor is going to suspend that particular work. So, processor is going to wait and till it is going to get the control of this particular bus. So, CPU will be suspended at some point of

time, it is not like that processor is going to carry out some work while DMA transfer is going on. Of course, if we are having some relevant information.

Now in most of the processor we are having a buffer space. In the buffer space again two type of buffers we are having. We are having an instruction buffer and we are having a data buffer. So, we have fetched some of the instruction and it is available in the instruction buffer and we have fetch some data and it is available in the data buffer, then processor can carry out those particular instruction while DMA transfer is going on.

And once this particular data got existed that now processor need to get information from the main memory, at that time processor is going to get suspended. It will wait till this data transfer is over. So, what are the data transfer mode? there is two way of transferring the information; one is called burst transfer mode and second one is your cycle stealing mode. So, now, in case of burst transfer mode we are going to transfer the entire information in one go.

So, in that particular case what will happen? I am saying that I want to transfer 1000 byte and I am going to transfer it to the memory location 7000. So, in case of burst mode what will happen? when we are going to transfer information that DMA controller is going to get that access of the bus and it is going to transfer the entire information, all the 1000 bytes and when its complete then it is going to give a interrupt signal to the processor. Now processor can get back the particular system bus. Now it can going to access the information.

So, in that particular case what will happen. In one go we are going to transfer the entire information, this is known as burst mode. So, what is the problem that we are having burst mode. So, once processor is going to complete the tasks that is available inside the processor. Now processor is going to wait till this data transfer is complete, because now processor want to fetch some more information from the memory. So, processor will be suspended for a longer period time.

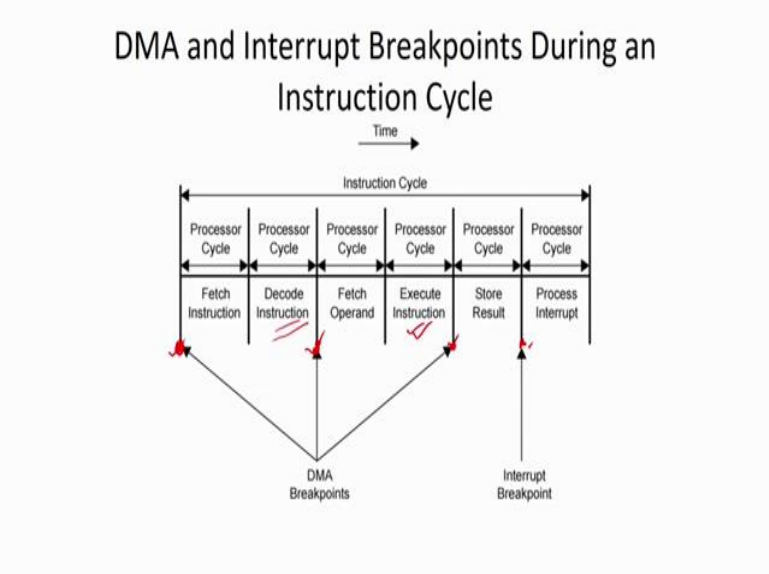
So, for that second one is talking about a cycle stealing mode; that means, it is going to steal a cycle from the processor, it is saying something like that. So, in the particular case what will happen? When processor is going to or say DMA controller is going to get the access of the bus control of the bus, it is going to transfer the information after transferring each byte of information; say it is a byte transfer.

So, its unit of transfer, it is temporarily, the bus will be given to the processor to do some transfer if some way processor is suspended or waiting to get some information from the memory. So, in the particular way in cycle stealing mode intermittently the bus access will be given to your processor. So, as to transfer some information and after that again control will come to the DMA controller. So that means, the access of the bus will be played between your processor and DMA controller during the entire period of transfer.

So, in burst mode in one go we are going to transfer everything and control of the bus will be given to the processor, but in case of cycle stealing, after every transfer intermediately that control of the bus may be given to the processor to transfer some of the information. If nothing is pending then again bus will come to the DMA controller.

So, basically it slows down the CPU, but not as long as CPU doing the transfer, because if CPU is doing the transfer, it is totally slowed down, it cannot do any other work, but here it will be slowed down in case of DMA transfer, because in some point of time processor is going to wait for the bus.

(Refer Slide Time: 31:21)



So, this is the DMA and interrupt breakpoints you can say. So, what is the interrupt breakpoint, basically when these breakpoint, basically what we are saying that when the processor is going to suspend its current program execution. So, if interrupt is coming. Say, currently it is executing one particular instruction then what will happen? It is going to complete this instruction and at that time after completion of the instruction it will check whether any

interrupt is pending or not, then if interrupt is pending and it is going to process this particular interrupt. So, this is the interrupt process cycle.

So, this is the only breakpoint, it will execute a complete instruction and then it will check for the interrupt, but in case of; that means, this breakpoint is in a what I am saying, this is a suspension or suspending the existence of the current program, but in case of DMA, it may suspend at different points say, already I talked about that instruction buffer, we have some instruction in the instruction buffer. Once it completes all those particular instruction then processor need to fetch a new instruction.

So, if case of DMA transfers in system bus is given to the DMA controller. So, processor cannot now going to get this bus so processor will wait over here, but once we are having an instruction at least processor can do the fetch the instruction; that means, it is there, it can decode the instruction after decoding the instruction if we have to fetch some more operand again, now processor will be suspended, processor will wait at that particular point, because now bus is with DMA controller.

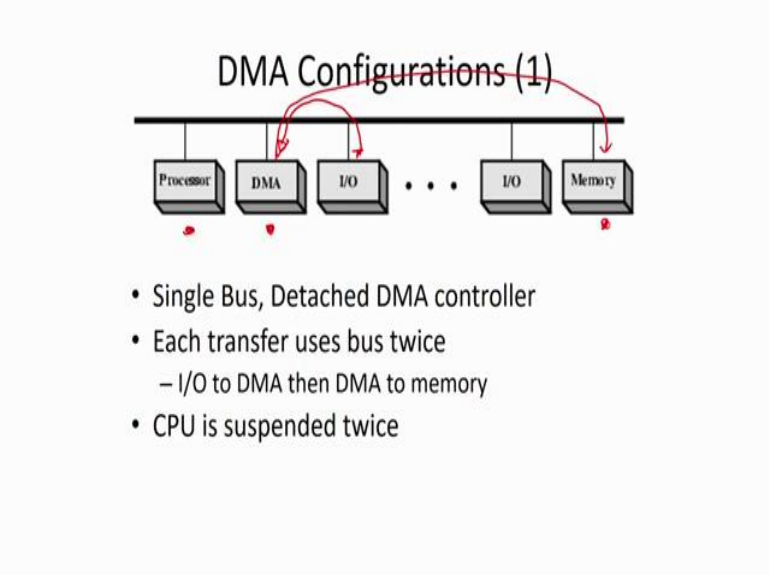
So, again say data is with me, operation with me then it can execute the instruction. After execution of the instruction; say if we want to store the result then again processor will suspended over here, wait over here. So, in that particular case, see processor may suspend at several points. So, these are basically that we say about DMA breakpoint, but in case of interrupt, there is only one interrupt breakpoint where the execution of the current program will be suspended.

Now, in this particular case. Now just see if we are using the bus mode of transfer then what will happen? The processor will wait either of this 4 points till the completion of the data transfer, but if we are using that cycling stealing mode then what will happen? At some point of time processor is going to get a control of the bus. At least it can fetch some of the information again it can carry out. So, this DMA breakpoint may be reduced if you go for cycle steal mode, but it will take more time to completion of the data transfer, because in between the bus has given to the processor.

So, these are the two ways you can transfer it and these are issues related to DMA transfer. So, in case of DMA transfer we are directly transferring information from devices to the memory or memory to the devices and processor is free from transferring the information. So, processor can carry out some other operation if all the information related to that operation is available

inside the processor. If processor need to access something from the memory then processor is going to wait at that particular point and we say these are the DMA breakpoint.

(Refer Slide Time: 34:45)



Now how you are going to connect those particular DMA controller. So, these are simple. So, this is a single bus, we detached DMA controller. So, in the particular user, see this is the processor. Processor is directly connected to the memory through this particular system bus. So, now processor can transfer information from memory.

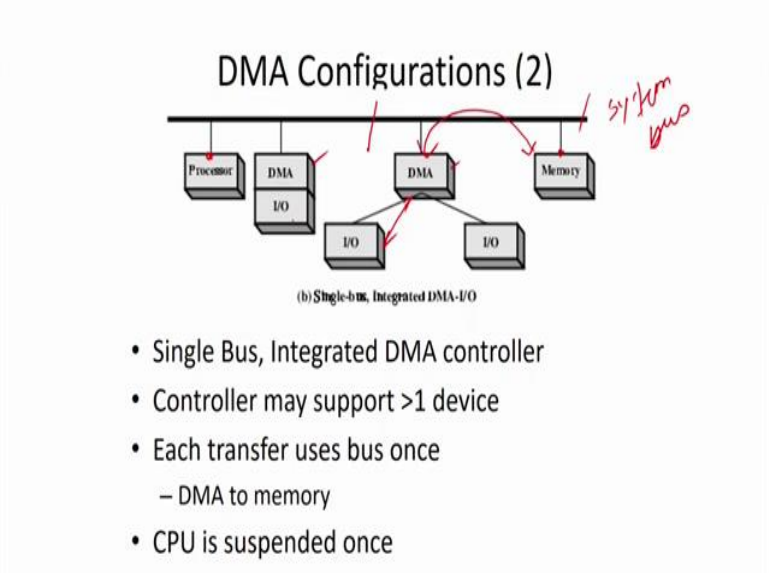
So, when I am going to transfer I/O devices, that I/O devices will also maybe directly connected to the bus, along with that we are having a DMA controller which is also connected to the bus. So, we are having a single bus and everything is connected to these things when we are going to perform I/O transfer, maybe I/O to DMA then DMA to memory. So, we are having two type of transfer.

First we are going to have I/O to DMA ok. At the time the processor is stealing from the memory and once we are coming to the DMA then what will happen? From DMA to memory this is the second. So, each transfer you just bus twice one for one transfer from I/O devices to the DMA, we are using a bus during that time, processor cannot access information from memory.

So, once you collect the information in your DMA controller then again DMA is going to take the control of the bus and going to transfer the information from DMA to the memory. So, it is

going to get the bus twice; that means, CPU suspended twice for one single transfer. So, this is one way of connecting DMA and I/O devices to the processor.

(Refer Slide Time: 36:31)

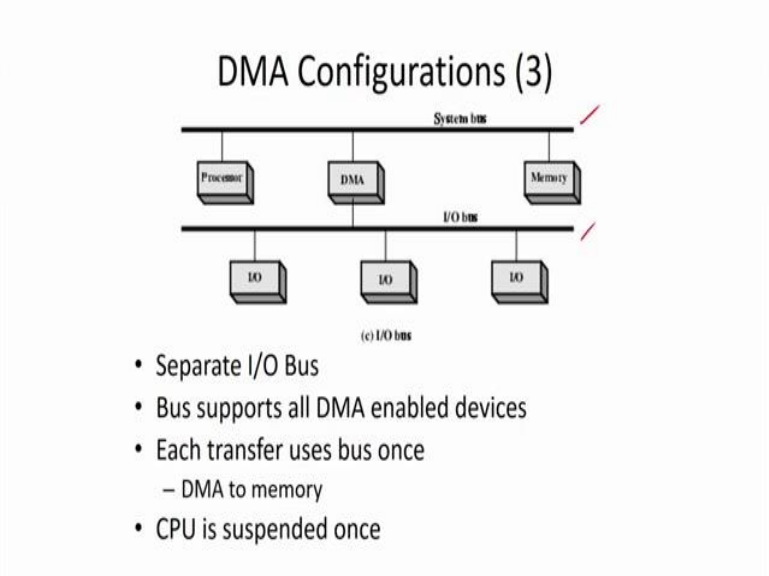


So, another configuration is your here I/O devices is not directly connected to the bus, I/O devices are connected through DMA module. So, this is the same thing. So, this is, processor is connected to the main memory through this particular system bus and that DMA controllers are connected to the system bus. Now all the I/O devices are connected through that DMA controller. So, in one DMA controller we may use more I/O devices also. So, this is in this particular DMA controller, we are using only one I/O devices in this particular DMA controller, we are connecting two I/O devices.

So, in that particular case you just see that during the transfer, bus will be suspended only once, because when your processor is initiating at DMA transfer, it will give the information to DMA. Now DMA controller is going to collect the information from devices ok. If it is an input and once everything is ready then DMA is going to give a request to the processor, and at that time processor is going to release the bus to the DMA controller and DMA controller is going to transfer the information to the memory.

So, in this particular case your bus will be suspended only once or CPU is suspended only once, because bus, the control of the bus is given to DMA controller.

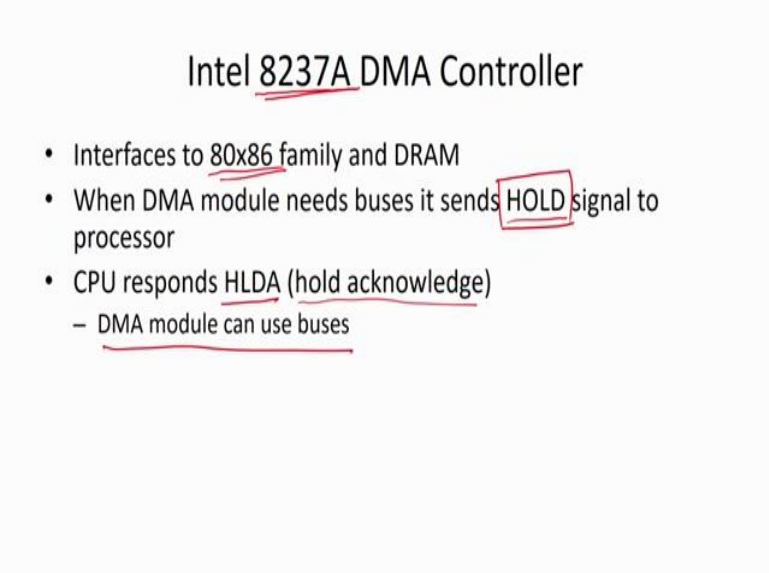
(Refer Slide Time: 37:51)



So, this is another configuration. So, again it is a two bus system; one is your I/O bus and second one is system bus; that means, two system bus we are connecting to the memory to the processor, we are using a separate bus called I/O bus and that I/O bus that all the I/O devices are connected to the I/O bus and that I/O bus is connected to the DMA controller. So, you just see while again transfer of the information, the system bus will be used only once. So, CPU will be suspended only one. So, this is another configuration that we are having.

So, we can use these three different configuration, while going to connect DMA controller one is well system bus and all devices are also connected to the system bus. In that particular case if we will be suspended twice, second case we are having one system bus, but I/O devices are connected to the DMA module and DMA is connected to the system bus. The system bus will be, or say processor will be suspended only once and here we are using two bus; one is your system bus and one is your I/O bus and said DMA controller is the interface between your system bus and I/O bus. So, during the transfer CPU will be suspended only once.

(Refer Slide Time: 39:09)

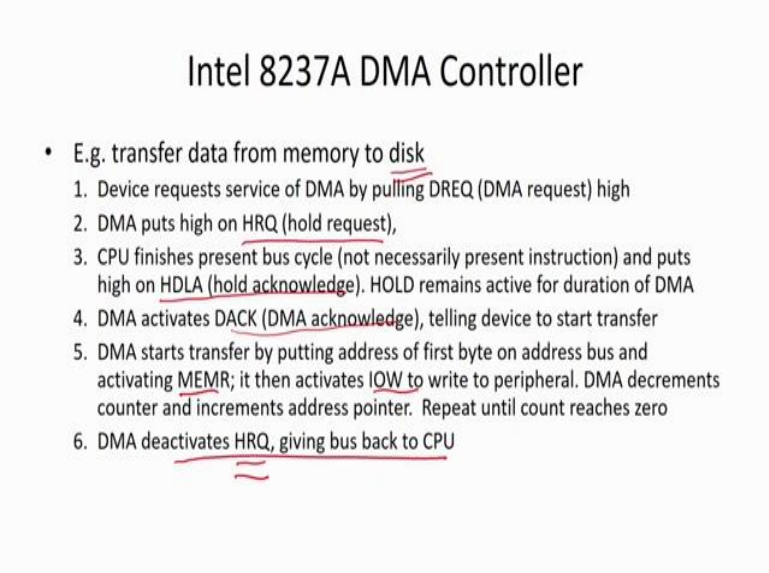


So, like that when we talked about, discussed about the interrupt driven I/O. For every processor we are having an interrupt controller. So, like that for every processor or every familiar processor we are having a DMA controller, when we talk about 8086 families. So that means, 8086 processor or 80186 processor or 286 processor for that particular families, Intel families we are having a DMA controller, the number of DMA controller is 8237A, this is a DMA controller. So, when DMA module needs buses, it sends Hold signal to the processor.

So, these are several signals that we are having and CPU responds by hold acknowledgment signal. So, basically say one is, we talk about DMA request and DMA acknowledgement, when we talk about this basic structure of the DMA then DMA request and DMA acknowledgement ok. So, now for different industries or different companies use their own proprietary signals and they give some name to it, but they are similar to that DMA request and DMA acknowledgement.

So, in this particular DMA controller, we are having one signal called hold which is nothing, but similar to your DMA request and one is your HLDA, hold acknowledgement. This is basically nothing, but DMA acknowledgement ok. So, once that hold acknowledgement is received by this controller, then DMA module can use the buses; that means, bus control of the bus will come to the DMA controller and that memory will be directly, will be directly connected to the DMA controller.

(Refer Slide Time: 40:51)

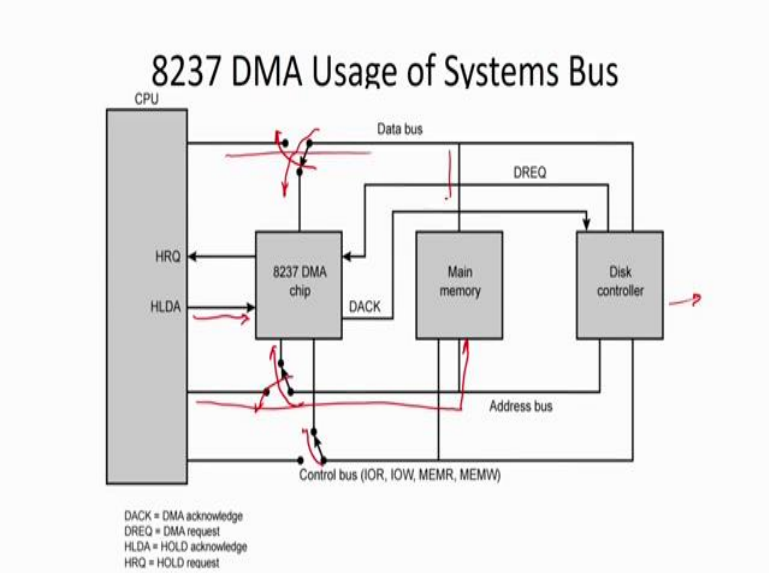


So, how it works. So, for example, data transfer from memory to disk. Disk means here we are talking about a hard disk, because you know that, in your machines you are having a hard disk of capacity say 500 Gb or like that. So, now, device requests service of the DMA by pulling DMA request ok. Now DMA request is between device and DMA controller, DMA puts high on HRQ hold requests. So, it is getting a request now DMA puts hold it says that ok it is reserved.

Now, CPU finishes the present bus cycle, not necessarily present instruction and puts high on hold acknowledgment ok. So, you just see it is getting a hold signal, hold request. Now processor is going to suspend operation and it is going to that hold request. Now after getting the hold acknowledgment, now DMA activates the DMA acknowledgement, because DMA is coming from the processor telling the devices to start the transfer.

DMA start transfer by pulling the address of the first byte of address bus and activate the memory read, it then activate low IOW to write the peripherals ok. So, basically now this is basically, if it is a memory transfer or I/O write. So, basically it is going to work in that particular point and DMA deactivates the HRQ giving bus back to the CPU. So, here we are getting hold request; that means, DMA puts high on it; that means, now DMA going to work with the bus, once it pulled down hold request then now, bus can be used by the processor. So, this is the way we are going to configure the system that bus will be directly connected to the devices.

(Refer Slide Time: 42:43)



So, this is the scenario, you can see something like that; say this is the DMA controller, initially what will happen just you think that processor is directly connected to the main memory with this particular bus ok. This is the address bus, this is the, this is the data bus. So, processor is working with this particular memory.

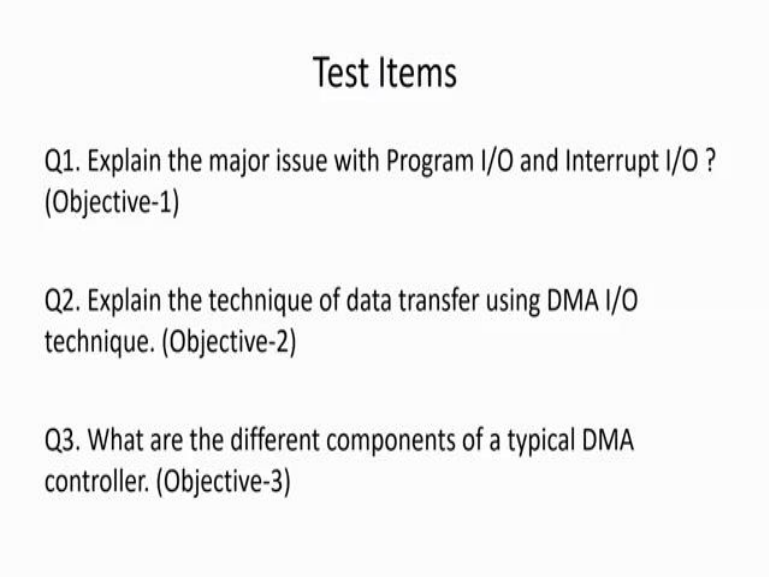
But when this situation is coming, then when DMA gives this hold request, basically it is coming from this DMA request coming from the controller then the disk controller, then it will give DMA acknowledgement, means it can work with that before giving it, it will give the hold request. So, when it is giving the hold request, now processor is going to release the bus, it will give the hold acknowledgement, when it is coming the hold acknowledgment now this bus is connected to this particular controller and this is the control bus.

Now, you just see when we are connecting in this particular way. Now processor is independent, it is not connected to any of the devices through system bus. Now through this particular DMA controller. Now it is going to access the address bus and data bus, now where from is going to get the address, because already in DMA controller we having a address register, we were having a data register. Now disk controller is going to get the information from disk and this is going to transfer it to the main memory. Once transfer is completed then what will happen, bus will be given back to the processor.

This is the way we are going to reconfigure a whole system that DMA controller is going to get access of the bus and going to carry out the transfer, without interfering this particular

processor and if some information is already available inside the processor, then processor again carry out those particular instruction or process those particular information ok. So, this is the way we are going to transfer information with the help of DMA controller.

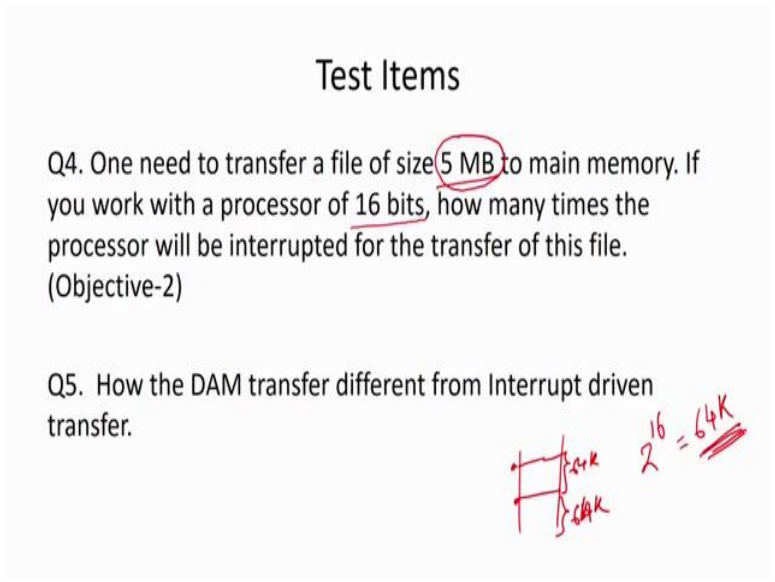
(Refer Slide Time: 44:44)



Now, see some test item. Question one; here I am saying that explain the major issues with program I/O and interrupt I/O. This is meeting the objective one already I said that in one case program I/O we are having the busy waiting. In case of interrupt I/O what we have? We have the intervention or processor is involved during the transfer. So, both have been eliminated in case of DMA. Explain the technique of data transfer using DMA I/O techniques.

Already I have discussed about the basic structure of the DMA modules. So, with respect to that I think you can say how we are transferring the information using the DMA technique. What are the different components of a typical DMA controller. So, this is the design issue. So, when we are going to. So, basically meeting the objective 3. Now we have to see what are the components that we have to put in a DMA controller, at least you see that data count is required, data register is required to transfer the data and we need the address register also to specify the address of the memory location that we are going to use ok. So, we have discussed those things.

(Refer Slide Time: 45:57)



Now, test item four, one need to transfer a file size of 5 MB to main memory. If you work with a processor of 16 bits how many times the processor will be interrupted for the transfer of this particular files. So, I am talking about the 5 MB and we are talking about the processor of 16 bit ok; that means, you just consider that everything is of 16 bit; that means, processor registers are 16 bit, my data bus is 16 bit maybe, address bus maybe 16 bit. So, in that particular case what will happen.

The data count will be your maximum 16 bit; that means, this will be your 2^{16} which is equal to your 65,000 something or you can say this is your 64 k memory location; that means, you can set the data count to 64 k only, but we need to transfer 5 MB of information. That means, in one go you can transfer 64 k only, because my data count can go up to 64 k, it is your 2^{16}. So, first you set it to 64 k.

Similarly in the memory also you start the starting address. So, once you transfer the 64 k then next time we have to again transfer 64 k. So, in one go I can transport 64 k only, and after every transfer we have to set the data count, because it will be reset to 0. You have to set the new address like that.

So, basically this question is like that, if I want to transfer of 5 MB from hard disk to your main memory. So, how many times the processor will be interrupted, because in one go you can go for 64 k only. Now you can calculate it. I think is now very simple once I give you the hints.

Now you can find it out and it will get that how many times that processor will be interrupted. Just think that we are using the burst mode of transfer.

How the DMA transfer. So, in question 5 how the DMA transfer different from interrupt driven transfer ok. Already I have mentioned, the basic difference is your context switching in interrupt driven it is there is a change of context. So, it is context switching, but in DMA transfer it is not context switching, second one in case of interrupt driven during data transfer processor is involved, but in case of DMA processor is not involved; like that you can see that what are the differences that we may have ok.

With this I come to end of this particular unit. I wish that we have got an idea or issues related to DMA transfer and I think you are in a position to design a DMA controller to transfer information from input output devices to main memory.

Thank you all.