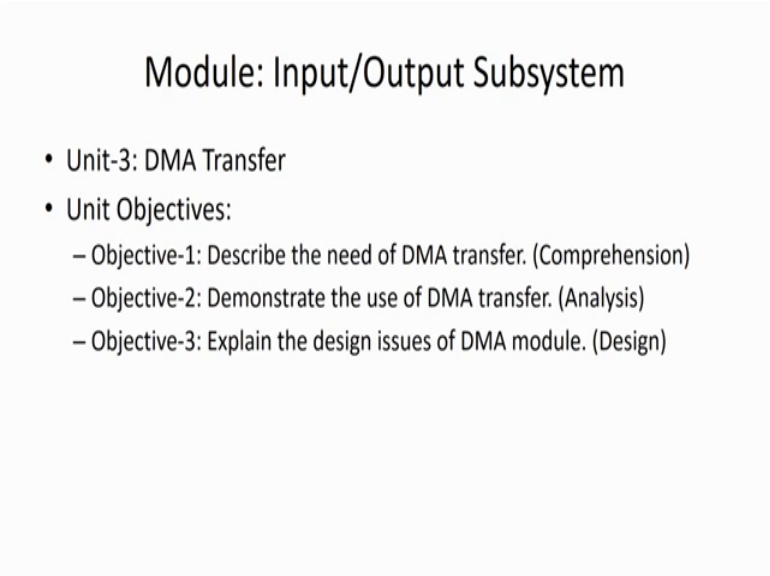
Computer Organization and Architecture: A Pedagogical Aspect

Prof. Jatindra Kr. Deka
Dr. Santosh Biswas
Dr. Arnab Sarkar
Department of Computer Science & Engineering
Indian Institute of Technology, Guwahati

Lecture – 35
DMA Transfer

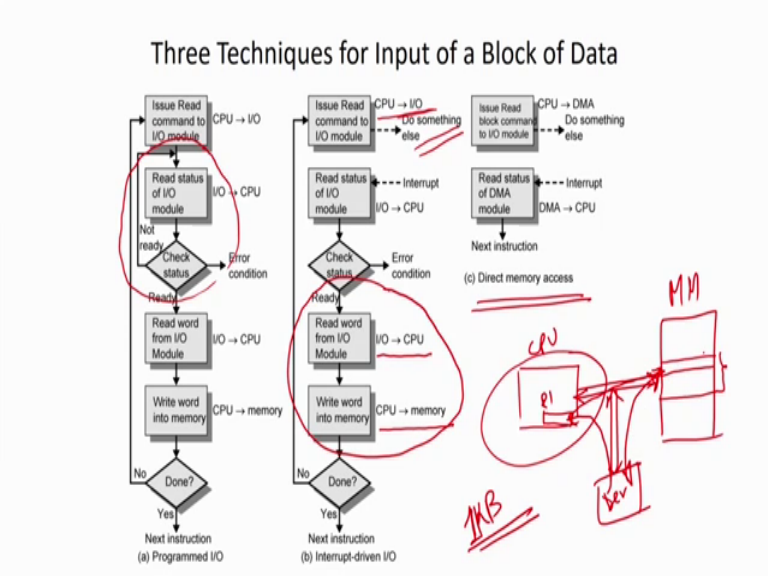
Hello everybody, welcome back to the online course on computer organization and architecture. We are in the input output subsystem. Now unit three is related to DMA transfer, already we have said that there are three ways to transfer information programmed IO interrupt drive IO and DMA transfer. Already we have discussed about the other two issues. Now we are coming to the third mode of transport this is your DMA transfer. What is the objective of this particular module DMA transfer.

(Refer Slide Time: 01:00)



So, objective one describe the need of DMA transfer, it will be in the comprehensive level objective two; demonstrate the use of DMA transfer. This will be discussed in the analysis level, explain the design issues of DMA module. So, it will be in the design happen. So, we were going to say what are issues that are, that need to be discussed when we are going to design a DMA module direct memory access, DMA is direct memory access. So, already you just see that what we did in the case of your programmed I O.

(Refer Slide Time: 01:36)



We have problem over here. Well processor is having busy waiting. So, to overcome these things we have come with the interrupt driven IO. So, we have eliminated this particular busy waiting and after initiating the transport. Now processor can do something else, but here also if you look into the complete transfer process what you will find that, in this particular during transfer if you look into it then what will happen? If it is coming from IO two processor.

Basically processors are having some registers, temporal storage and if we are going to keep more information, transfer more information. Finally, we are going to keep it into the main memory. We just see that when device is ready to transfer information and what will happen? First we are transferring it from devices to the CPU; that means, you are bring it to some processor register and from processor register we are transferring it to memory.

So, basically what will happened? You just see that if this is my processor CPU and this is my main memory, main memory and it is connected to system bus and this two system bus; say we are connecting to a device. So, say that I am not eliminating the i, almost the device is direct connected then in this particular mode of operation what will happen.

If we want to transfer some information from some device, may be, say even if you are, if you are trying to transfer in file from your hard disk to your main memory, then we are storing this particular information inside this and we are going to bring it to the main

memory, just adapt the file size of say 1 kilobyte, then what will happen, and we are going to bring this particular 1 kilobyte to the memory, because we do not have a storage space inside the processor that one kilobyte storage space. We are having highly pure resistors, may be 8 to 16.

. So, in this particular way what it is doing from device positive come to some register in, say the processor may be say register R 1. So, it is coming to the register and from register we are storing it to the memorial operation. So, during this data transfer operation, processor is involved, processor is actively involved in the data transfer. So, when we are going to transfer 1 kilobyte of information then what will happen.

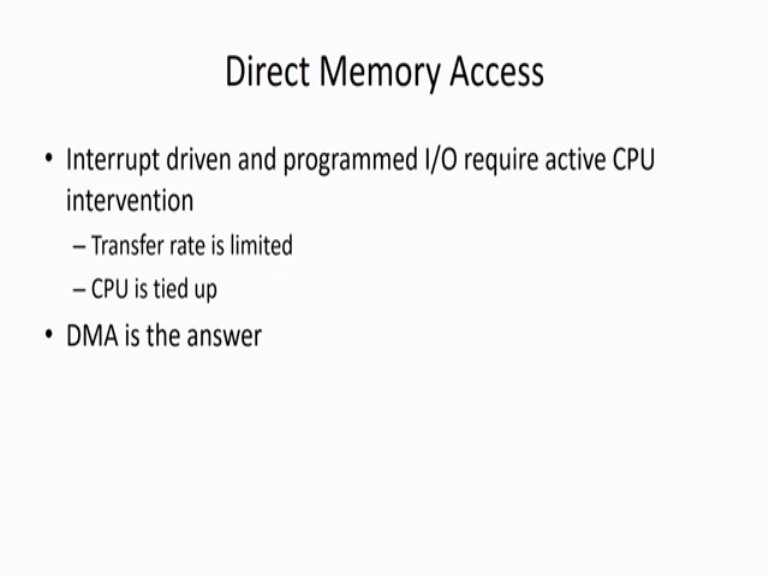
We are going to transfer it byte by byte and not maybe say if it is a 16 bit configuration maybe 2 byte at a time like that we are going to transfer it and for transferring enter 1 kilobyte, what will happen? the processor is involved, processor cannot do any other work, because it is taking the information from device, bring it to the processor is the, from processor register we are transferring into the memory. So, processor is grossly involved in transferring the information.

. So, we should think about that you know. So, where the processor can be freed while doing the data transfer. So, that processor can carry out some other work. So, for that the solution is your DMA; direct memory access. So, in case of direct memory access what will happen. The data transfer will take place between device and memory involvement of processor will be eliminated.

So, this is the basic crux about a DMA, and why you are coming to DMA. This is the main reason that we want to remove the load of the processor during the data transfer. So, if we can directly transfer information from device to memory, during that time process on, may carry out some other work, but what are the things processors and carryout, we will see that thing also ok.

So, this is the basic concept, why you are coming for the DMA direct memory access. So, the, a message direct memory access and data transfer takes place between memory and devices intervention of processor is eliminated.

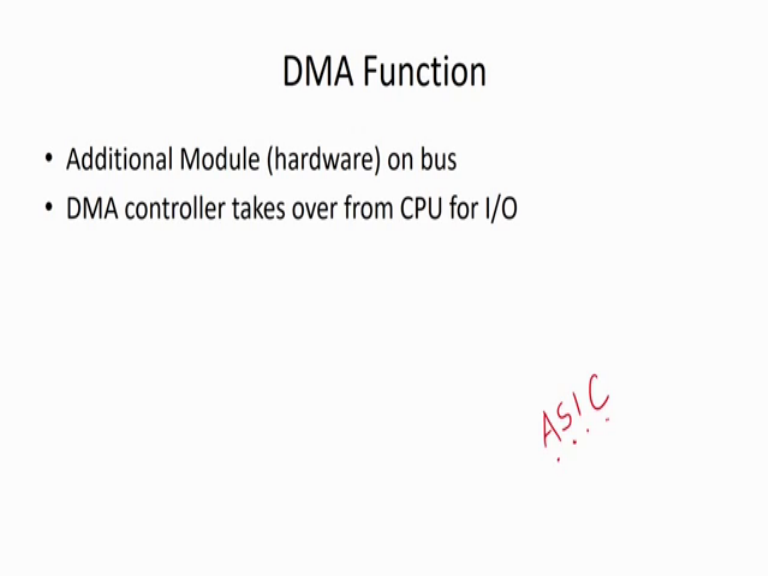
(Refer Slide Time: 05:47)



So, this is basically already expand interrupt driven and program IO requires the active CPU intervention; that means, CPU is always engage well transferring the information, transfer rate is limited and CPU is tied up. So, it cannot do any other work, just to eliminate all those things, what is the insert? DMA is the insert. So, direct memory access we can look for it. So, that the overhead; that is getting by the processor can be eliminated if we are going to use DMA.

Now we are going to see what are the issues while you are going to design that DMA, DMA controller direct memory access. And I think after the completion of this course you will be having an idea how DMA works, even you will be in a position to design a new DMA controller.

(Refer Slide Time: 06:38)



So, for that what will happen? We are having an additional module in the hardware or we are connecting to the bus, it is known as your DMA controller and when we are going to have transfer, then this DMA controller. So, you just see that we are having IO module that I have module can transfer information. So, DMA controller can also treated as an IO module only, it is an input output model, but it is having a specific functionality.

So, in that particular case what will happen. Now DMA controller going to text about the data transfer from CPU and it will carry out this; that means, it is having some processing tasks event; that means, you can think about that we are designing another processor dedicated processor which can carry out a specific tasks only. What specific task is can do. It can do the data transfer between device and processor.

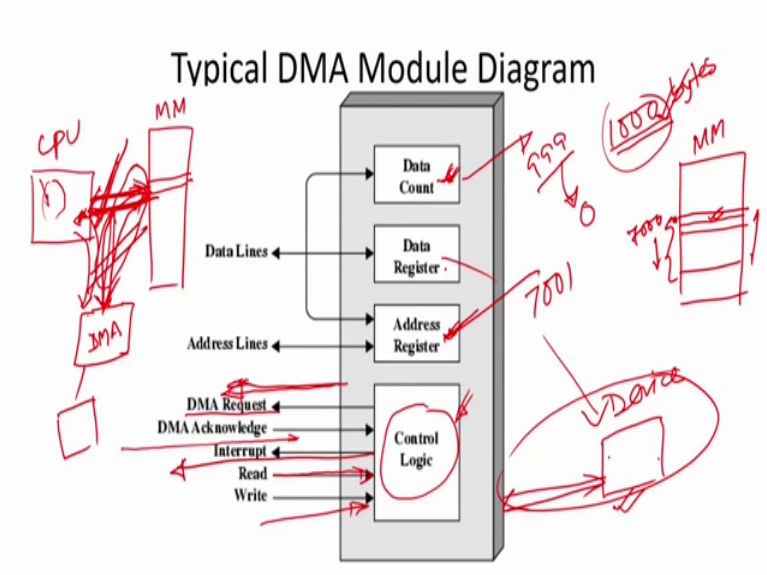
So, I think I a at some point up time, I have mentioned about ASIC application specific integrated circuit. So, DMA may fall in this particular category. In this category devices ASIC application specific integrated circuit and what is that application the basic application that DMA is going to handle, is your data transfer between IO devices and memory and how why you require these things, because first of all you have to bring the information to the memory and how processor works? Processor works on human stored program principle and processors going to access the information from main memory which is the storage in that particular case.

. So, see processor works on one human stored program principle. So, when processor is going to execute some, a program are going to carry out some tasks, it is going to get the information from main memory which is basically storage unit. Now how we are going to get the information to the storage unit from input output devices. So, for that, know to transfer information from input devices to the main memory.

We can say that we are having a device controller called DMA controller which is an ASIC application specific ASIC and the tasks perform by DMA controller is transferring information from input devices to the memory. On the other hand along with that you can say that the transferring information from memory to the output devices, because the result we have to give it to the users.

. So, DMA controller is coming in between and it is going to transfer the information from devices to the memory all right. So, this is a function of a DMA controller and we can view it as ASIC application specific integrated circuit.

(Refer Slide Time: 09:26)



Now, the typical DMA model diagram, you just see what are the things that we are having. So, these are that issues that whatever is, even just I am keeping it here, but along with that this addressing techniques and other things are similar to our IO module, because we have to have a status register, we have to give addresses, so that we can identify the devices. All those things are there in DMA controller also, but along with that we are having some specific components.

So, what are the specific component one is your control logic ok. So, this is the control logic after getting the signals from the output environment; that means, what is the output environment in that particular case. Now output environment is processor, it is going to get some indication, some signal from the processor along with that, it is going to get some information or signals from the devices connected to the DMA controller ok.

So, in the DMA model, those are the signals that it is going to get and depending on those particular signal this control logic, we need to design this particular control logic to carry out the appropriate task and to carry out the complete operation of transferring, the information from devices to the memory or from memory to the devices. So, this is the control logic we need to design. So, when you are going to design it, it will be a simple control logic after knowing the all the input signals, all the output signal and that way we know what as the things that it is going to perform, depending on that we can design this particular control logic.

So, it is a simple control logic. I think by this time you know how you are going to design a control unit. So, control unit is a part of my processor CPU, because CPU is having some registers ALU and control unit and in this particular subject, in this particular course, we have seen the design issues of this particular control unit to synchronized operation of the processor. So, with respect to that control unit, this control logic is a very simple one. If you put your time, if you put slight mine what need to be done, you can design this particular control logical also.

. So, what are the inputs and output to this control logic; one signal is your read or write, you are saying two signals. So, if it is a read; that means, you are going to take information from devices to the memory than it is an read information. If we are going to put information from memory to the output devices, it is right. So, this signal will come from the processor. It will give in the process, we will say whether it is a read or it is a write.

Along with that we are going to get other information, and so from which devices we are going to get it, we are not showing it over here. So, that information, also it will be given and accordingly this DMA module is going to walk with that particular device also that I am just writing it as a some devices there. So, this device is also connected to the IO

model. This portion we have not shown in the wire. So, basically it is giving the addresses and the status line of these particular devices.

So, one, it DMA module is getting the information that it is a read information and it is going to read from a particular device that after getting this information, it will pass these things to the device and going to collect the appropriate thing and going to set it up. So, once it will set it up that this device is ready for the data transfer and things then what will happen? That DMA controller or DMA module will give DMA request to the processor. So, processor is connected over here.

Now once it is giving that DMA request then it will give DMA acknowledgement, but when it will give DMA acknowledgment, when the processor is ready to perform the DMA transfer, now what it is going to do. So, one it is coming that DMA device is ready and everything is said, it will give the DMA request in between. Now processor, what processor is going to do. It is at least we must know what is the volume of the data we are going to transfer.

So, it will be in the, you are having a data count registers, just say that for a simple example I am saying that we are going to transfer say 1000 byte of information ok. So, this count will be said to 1000 ok. So, say that we are going to transfer 1000 byte of information. This is the requirement along with that. Now processor is going to say that it is going to read it.

Now after that bringing this information where we are going to store it. So, I am having a memory, main memory; say I am going to store this particular information from starting at this a 7000 onward. So, what will happen? that processor is going to make it ready that address register will be set to this particular 7000. Now just see what does it means?

We are going to transfer 1000 bytes of information; say it is byte organized and I am in memory also in, we are going to store 1 byte of information and where we are going to store it up, bringing it from the devices say from memory location 7000. So, this addresses that will be set to the 7000.

So, this is the address line which is connected to the processor. This is the data line connected to the processor. So, through this data bus we are setting this that account to

that 1000. Through this data line we are setting it to the address is start to 7000. Now everything is said. Now device is ready, it will be going to give this DMA request once processor is get is than processor will give the DMA acknowledgement.

Now what will happen in that particular case when the DMA acknowledgement is coming over here at the particular point, that DMA controller is going to take the control of the bus. Now you just see what will happen? Said this is my processor CPU, this is the memory ok. This is connected through the system bus and say that is I am showing DMA controller is connected to it. Now say processor is initiating, it is saying that after setting is that data count and your starting at this.

Now processor is initiating that it wants to perform a read operation. So, in that particular time that control logic, we have design in appropriate way. Now it is going to look for the appropriate devices, because address will also come from these particular devices, from which devices we are going to read it. So, once everything is said. Now device is ready, we are at the point of transferring the information then processor is giving the DMA request ok. It is having that getting the DMA request. Now when DMA request is coming, now processor is going to say that. Now we can perform the time operation give the DMA acknowledgement.

Then when DMA acknowledgment is coming for the DMA at the particular time, this control of the bus will be given to the DMA; that means, now processor is slightly dealing through control signal we are setting it and at the particular point. Now we are having this particular connection. Now processor is slightly dealing. Now processor is not going to use the system bus, system bus will be used by the DMA controller. Now what it will do? Now it will take the information from device to the data register.

So, this is the DMA module from data register, it will going to store into the memory location ok. Once 1 byte of information is transfer; that means, you are the getting the information from device through data register, it is going to the memory location then what will happen? Then data count will be decremented; that means, it will become say 999; that means, already have transfer one, we need to transfer 999 bytes and this address register will be implemented and it will say that 7001; that means, first byte we have stored in that register memory location 7000. Next byte we do, we need to store in 7001.

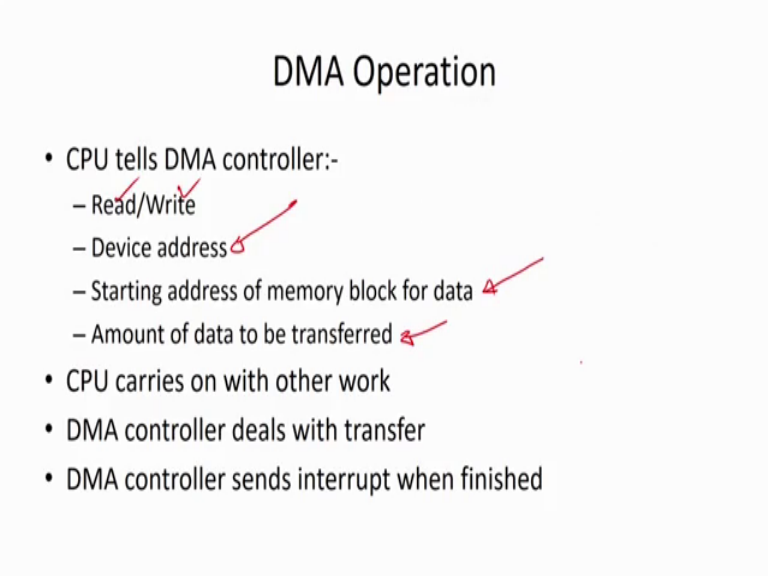
So, like that data count will be decremented after a every transfer and address register will be incremented just to point to the next memory location. So, when we transferred that 1000 memory byte; that means, when this value will come to 0; that means, we have transfer all the 1000 byte of information. So, after completion of the transport, now this particular DMA controller will issue this particular interrupt signal to the processor.

So, this interrupt signal will say that now transfer is over, it is completed. Now what it is going to do? once it get this particular signal then processor is going to take the control of the system, but; that means, now system bus will be connected to this particular processor and now this is the link. Now DMA is not directly connected to the system bus. So, this is the way that DMA is going to transfer information from devices to the memory.

So, what is the basic principle over here. Basically DMA is going to take control of the system bus and in system bus, basically it is going to look for address bus and data bus and their means carry out the transfer one, its completes the operation, it will give an indication to the processor, then processor is going to take back the system bus; that means, now DMA controller is no longer connected to the DMA bus, that now processor can worked with the main memory through this particular system bus.

So, this is the way we are transferring information from devices to the memory through DMA controller. And similarly from memory to the devices. Also you can transfer by following the same principle, except that here we are having this particular right signal and that will be transfer from memory to the devices through this particular data register. So, these are the basic components that we have in a DMA module or DMA controller and it works with the help of those control signals ok. This is the way that DMA controller works.

(Refer Slide Time: 20:16)



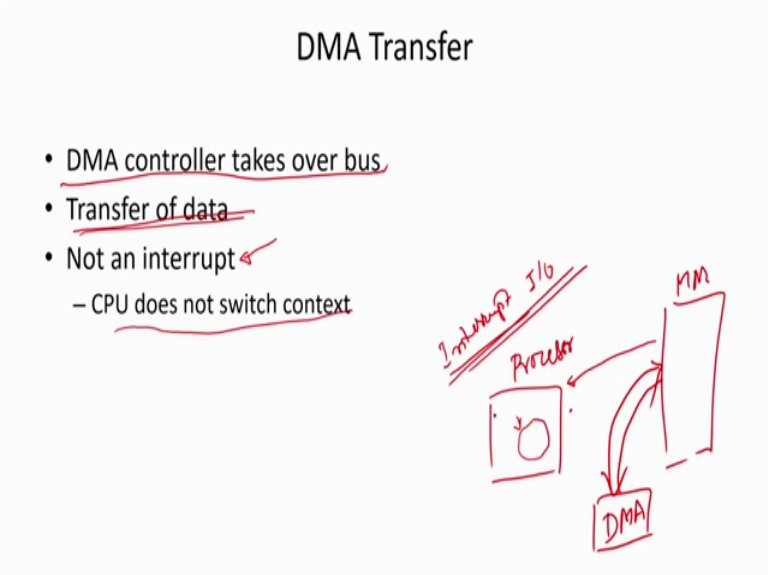
So, basically now we know how DMA controller is going to work, how it transferred the information without intervening the processor operation that this processor is not involved at a, not involved in this particular data transfer. So, basically now if what are the DMA operations, just see in that particular case, CPU tells the DMA controller whether it is read or write. This is the information that processor will give. It will give that device addresses also from which device it is going to take the information starting address of the memory block of data.

So, it will give the starting address of the memory location also, memory block also where we are going to store the information or from which memory block. We are going to transfer information to the output device along with that it will give that amount of data to be transfer ok. So, these are the things that I have saying it is going to give data count. It is going to give the starting address where which memory blocks that we are going to use. So, these are the information that CPU is going to tell to the DMA controller.

Now, CPU carries out with other works. Now this now it needs to transfer some information, but now processor is giving that information; that means, it is delegate the job to the DMA controller. Now processor can carry out its own work. DMA controller will deal with the transfer, DMA controller sense interrupt when finish now deals with the. Now after getting all those info required information.

Now DMA controllers will deal with the transfer operation from your device to the memory or memory to the device once everything over then everything sends an interrupt to the processor saying that this finish, basically why it is required. Now at that particular point we are going to transfer the bus from the DMA controller to the processor now. So, that procedure can fetch information from main memory. So, these are the operation, generally we perform when you are going to do a DMA transfer.

(Refer Slide Time: 22:25)



So, what basically, thus now basic info difference of what here is that DMA controller takes over a bus. So, now, system bus, basically data bus and address bus is used to connect the memory and along with that IO devices also. So, when we are using DMA controller then this control of this particular bus will be taken over by DMA controller; that means, processor is no longer going to use this particular bus, while DMA is transferring the information.

And so since DMA is taking order the control of the bus, now DMA is directly going to transfer a information from device to the memory or from memory to the devices. So, it is going to take over the bus then it is going to transfer the information. So, in that particular case it says that it is not an interrupt and we do not have or your CPU does not have switched context. Now this is the difference between your interrupt. In case of interrupt driven IO what will happen.

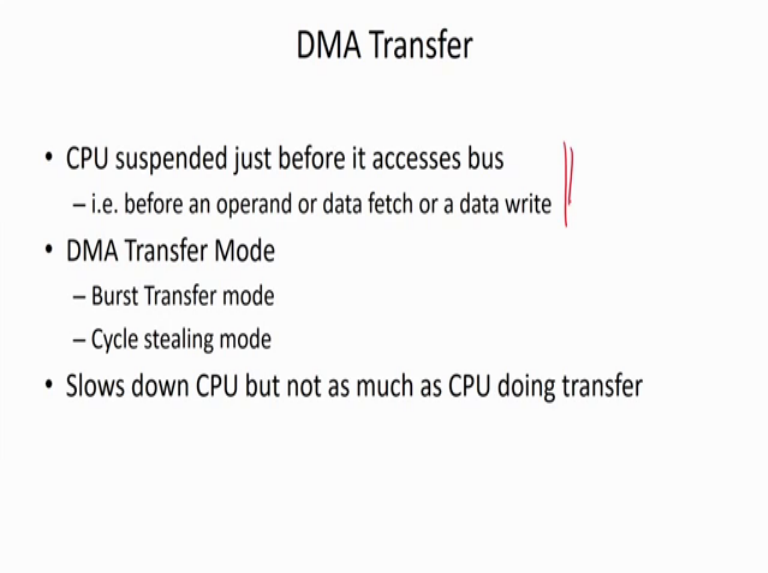
In case of interrupt driven IO what we are doing? We are running an interrupt service routine in the processor itself; that means, there is a context switch. So, basically processor is running a program it is in one particular context. This context is related to the program that was executed in the processor, but when interrupt comes, when we are going to give service to the interrupted devices; that means, we have to run the appropriate interrupt service routines. So, there is a sense of context in the processor, it is running one program, but it now processor is going to run another program in case of interrupt driven IO.

So, for that we have to retain the processor status and how we are doing it? we are just transferring the status of the processor to the systems state and processor is now going to execute the interrupts service routine. So, there is a sense of context on the program being executed in. The processor is now suspended, we are storing the context or the relevant information in the systems state and now processor is going to execute that another program which is in that service routine once the transfer is over, then processor is going to restore its initial state; that means, again this coming to the same context.

So, this is a context sense ok. So, it is going to sense its context from one program to the other program. But in case of DMA transfer there is no context. Sense processor is executing one particular program it till is going to execute the particular program or it will return state, because the data transfer operation has been delegated to that DMA controller. Now DMA controller is going to take care of transfer this information. So, processor is the same state, same context it can execute the same program provided the relevant information is available inside the processor.

So, this is the difference you must remember it. So, in case of interrupt driven IO, here is a sense of context. In case of DMA transfer there is no context sense, 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? It is a CPU suspended just before it access the bus ok. We will see before an operation of oper an 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 sense 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 sense of context; that means, processor can carry out its own work.

now what it says then. Now CPU suspended just before it access 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 ah. So, it is going to execute some instruction, but at some point of time, now processor need 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 suspended that particular work. So, processor is going to wait and still 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 that 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 fetch some of the instruction and 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 exist 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 bus transfer mode and second one is your cycle stealing mode. So, now, in case of bus 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 going to transfer it to the memory location 7000. So, in case of bus 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 byte and 1, 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 bus mode. So, what is the problem that we are having bus mode. So, once processor is going to complete that 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. 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 transpiring each byte of information; say it is a byte transfer.

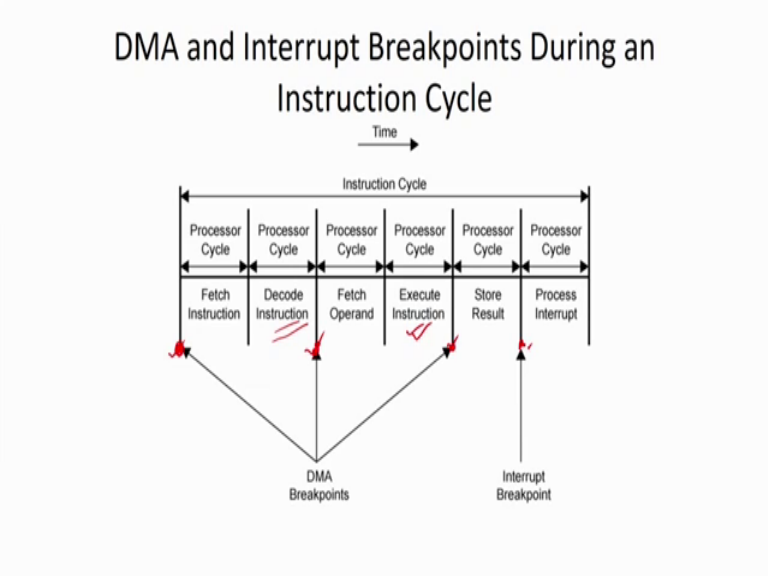
So, its unit of transfer, it is temporally, the bus will be given to the processor to do some transfer. If some way processor is suspended or waiting to get some information from the memory. So, in the particular way in cycle stealing mode intermittently the bus access will be given to your processor. 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 bus 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 spending then again bus will come to the DMA controller.

So, basically it slows down the CPU, but not as much as CPU doing the transfer, because 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; an interrupt breakpoints you can say. So, what is the interrupt breakpoint, basically when these breakpoint, basically what we have saying that when the processor is going to suspend its current program execution. So, if interrupt is coming. So, 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 spending or not, then if interrupt is spending 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 ah. What I am saying, this is a suspension or suspending the existence of the current program, but in case of DMA, it may suspended a defined point said 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 or 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 they are, 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.

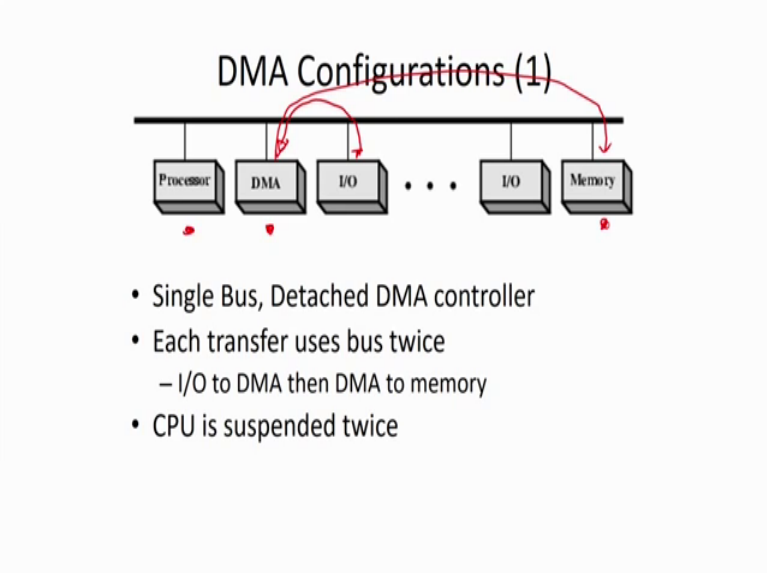
So, again said data is with me, operation with me then it can execute the instruction. After execution of the instruction; say if we want to store the result then again processor will suspended over here, wait over here. So, in that particular case, see processor may suspend at several points. So, these are basically talk 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 point 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 a information from devices to the memory or memory to the devices and processor a street from transferring the information. So, processor can carry out some other operation if all the information related to that operation is available inside a processor. If processor need to access

something from the memory then processor is going to wait at that particular point and we said 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.

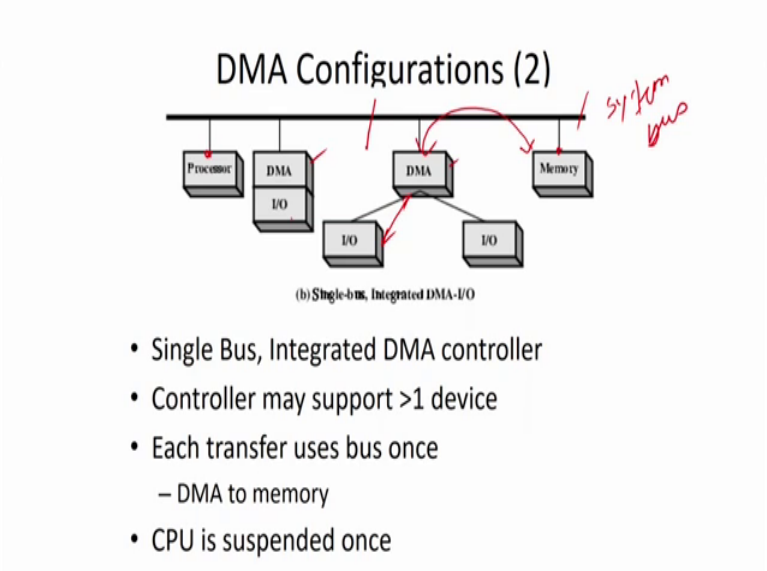
So, when I am going to transfer IO devices, that IO devices will also maybe directly connected to the bus, along with that we are having a DMA controller which is also connect to the bus. So, we are having a single bus and everything is connected to these things when we are going to perform IO transfer,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

First we are going to have IO to DMA ok. At the time the processor is dealing 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 IO 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 a to the

memory. So, it is going to get the bus twice; that means, if its suspended twice for one single transfer. So, this is one way of connecting DMA an IO devices to the processor.

(Refer Slide Time: 36:31)

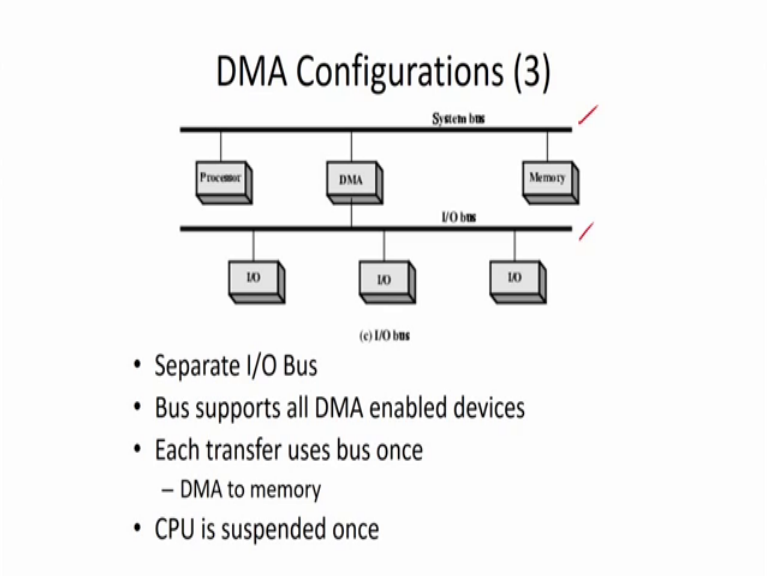


So, another configuration is your here IO devices is not directly connected to the bus, IO devices are connected through DMA module. So, this is the something. So, this is, processor is connected to the main memory through this particular system bus and that DMA controllers are connected to the system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

So, in that particular case you just see that during the transfer, bus will be suspended only once, because when we are 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)

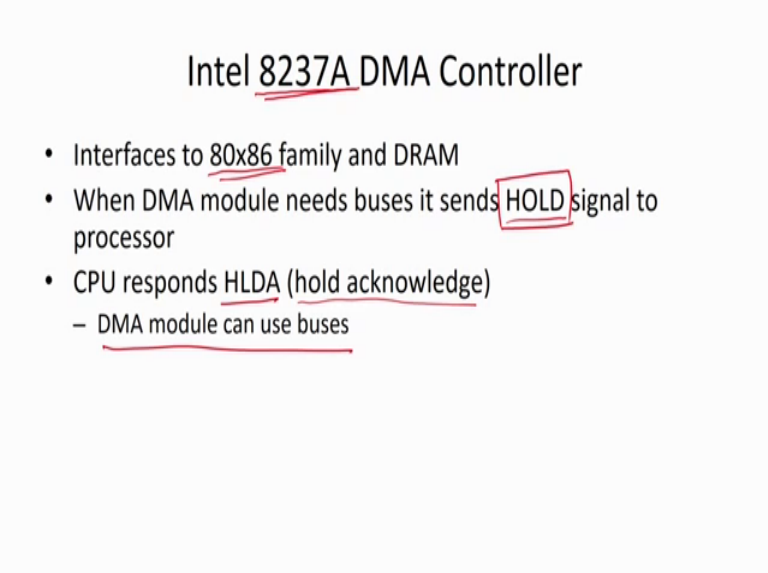


So, this is another configuration. So, again it is a two bus system; one is your IO bus and second one is system bus; that means, two system bus we are connecting to the memory to the processor, we are using a separate bus call IO bus and that IO bus that all the IO devices are connected to the IO bus and that IO bus is connected to the DMA controller. So, you will 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, well 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 IO devices are connected to the DMA module and DMA is connected to the system bus.

The system bus will be, what is a processor will be suspended only once and here we are using two bus; one is your system bus and one is your IO bus and said DMA controller is the interface between your system bus and IO bus. So, during the transfer CPU will be suspended only once.

(Refer Slide Time: 39:09)

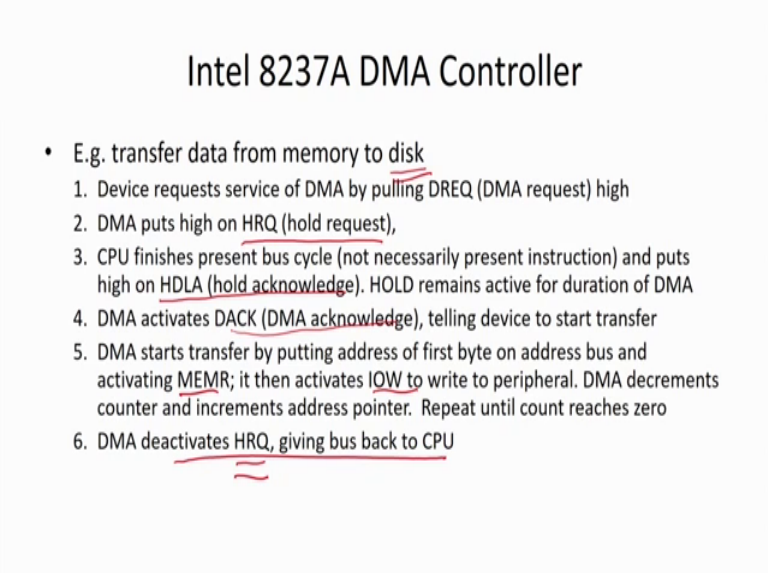


So, like that when we talked about, discuss about the entire driven IO. For every processor we are having interrupt controller. So, like that for every processor or every familiar processor we are having a DMA controller, when we talk about 8086 families. So; that means, 8086 processor or 80186 processor or 286 processor for that particular families, Intel families we are having a DMA controller, the number of DMA controller is 8237 a, this is a DMA controller. So, when DMA module buses, it sends Hols signal to the processor.

So, these are several signals that we are having and CPU responded by hold acknowledgment signal. So, basically say so one is, we talked about DMA request and DMA acknowledgement, when we talk about this basic structure of the DMAs and 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 hl da, hold acknowledgement. This is basically nothing, but DMA acknowledgement ok. So, once that hold acknowledgement is received by this a 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)

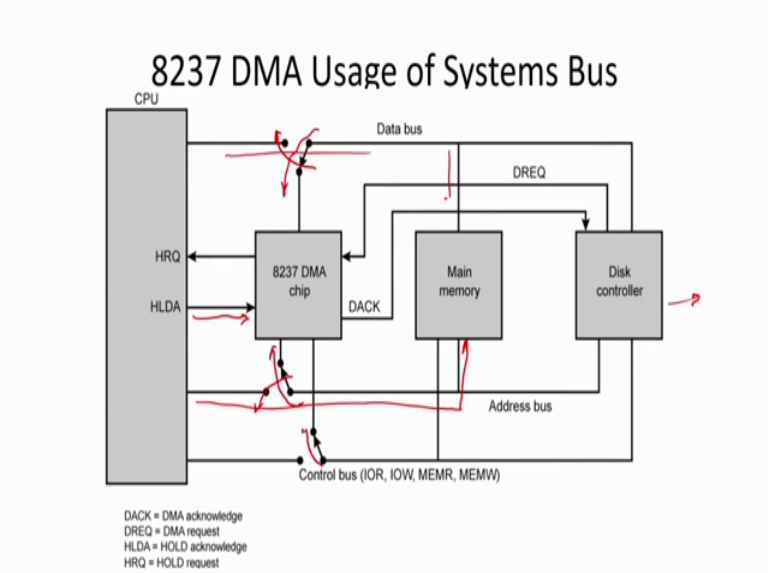


So, how it works. So, for example, data transfer for memory to disk. Disk means here we are talking about a hard disk, because you know that, you know may. Since you are having a hard disk of capacity say 500 Gb or like that. So, now, device requests service of the DMA by pulling DMA request ok. Now DMA request is between device and DMA controller, DMA puts high on HRQ hold requests. So, it is getting a request not DMA puts is all it says that does deserve.

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

DMA start transfer by pulling the address of the first byte of address bus and activate the memory with it, then activate low IOW to write the peripherals ok. So, basically now this is basically, if it is a memory transfer or IO right. So, basically it is going to work in that particular point and DMA deactivates a HRQ giving bus back to the CPU. So, here we are getting whole request; that means, DMA puts IO in; that means, now DMA going to work with the bus, once it pull it down hold request then now, but 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)



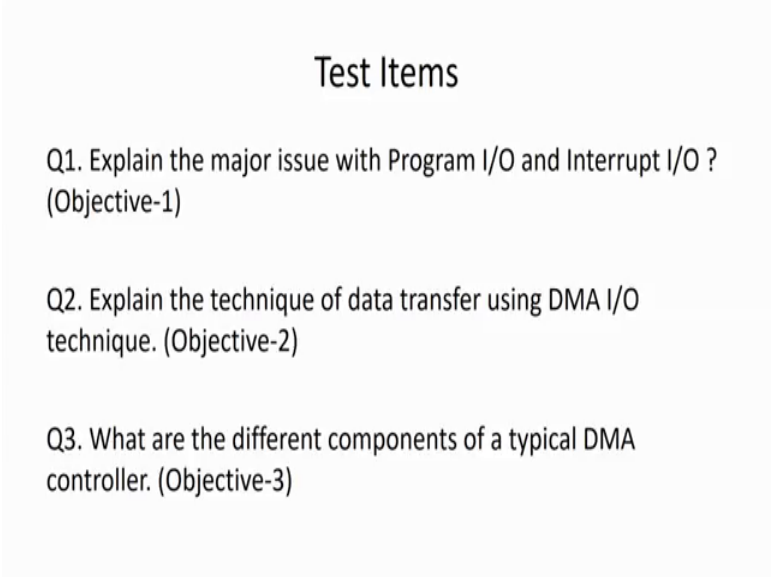
So, this is the scenario, you can see something like that; say this is the DMA controller, initially what will happen just you think that processor is directly connected to the main memory with this particular bus. This is the address bus, this is the, this is the data bus. So, processor is working with 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 acknowledge one, it is coming the hold acknowledgment. Now this bus is connected to this particular controller and this is the control bus.

Now, we just see when we are connecting in this particular way. Now processor is independent, it is not connected to any of the devices through this 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 get that just, because already in DMA controller we having a address says that we were having a data register. Now disk controller is going to get that 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 re reconfigure a hold 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)

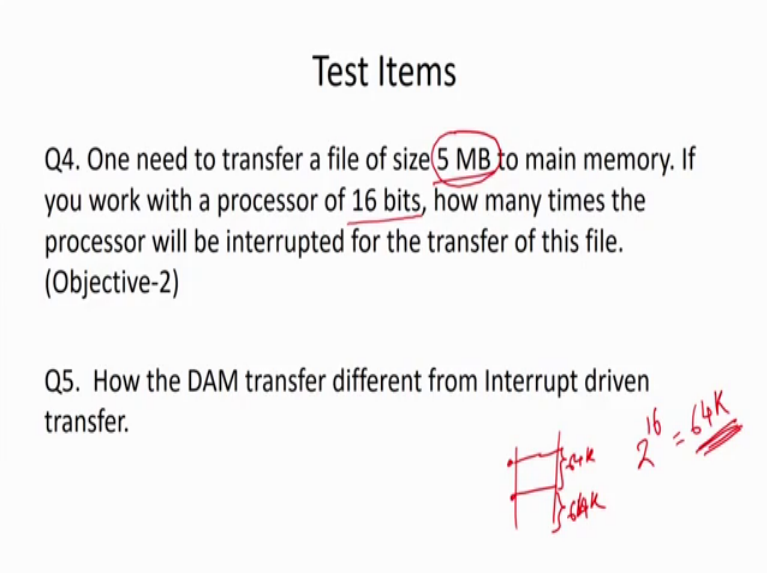


Now, see some test item. Question one; here I am saying that explain the major issues with program IO and interrupt IO. This is meeting the objective one on the web, said that in one case program IO we are having the busy waiting. In case of interrupt IO what we have? We have the intervention or processor is involved during the transfer. So, what has been eliminated in case of DMA external technique of data transfer using DMA IO 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 meaning the objective fees. 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 that address register also to specify the address of the memory location that we are going to use. 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 transport 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 or maximum 16 bit; that means, this is your 2 to the power 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 3 or 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 hope 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 set 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 interrupt. Just think that we are using the bus mode of transfer.

How the DMA transfer. So, in question 5 how the DMA transfer defined for interrupt driven transfer ok. Already I have mentioned, it 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.

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.