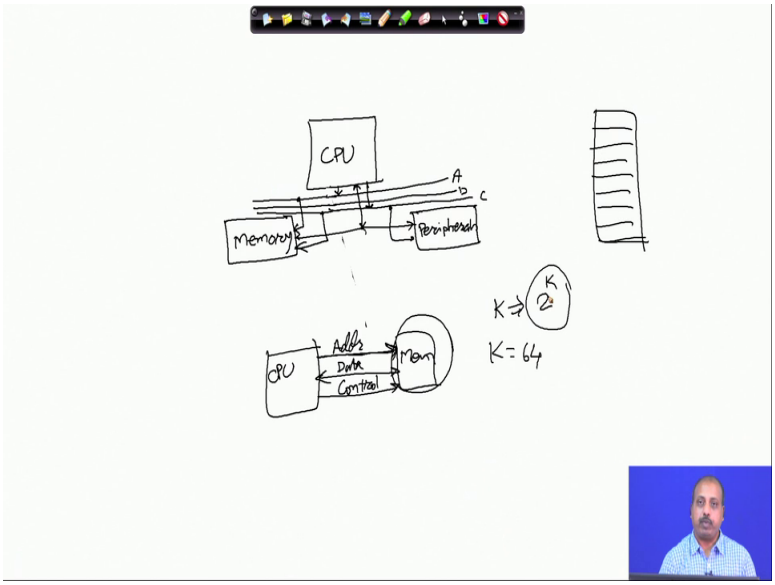
Microprocessors and Microcontrollers
Prof. Santanu Chattopadhyay
Department of E & EC Engineering
Indian Institute of Technology, Kharagpur

Lecture - 04
Basic Computer Organization

Next we will start with the basic computer organization part. So, all though the microprocessors and microcontrollers that will be discussing they will be part of this any computer system. So, we will start by looking into the computer system from the outside and the at the major component level and then slowly go inside and talk about how these processors are organized, and how are they designed how are they used etcetera.

(Refer Slide Time: 00:47)



So, in any computer system, so you can find some components like this the basic component that we have, one is the central processing unit or the CPU and we have got two major components which are apart from the CPU, one is the memory one component is the memory and the other component that we have are the peripherals the peripherals.

Now, so the memory so that is actually used for storing the program that the processor will execute and the peripherals. So, they are used for giving input and our giving input to the system or taking output from the system. So, when the user is interacting with the system. So, the user interacts via this peripherals and the program that does the processor

is executing is stored in the memory, and the processor is actually the central processing unit or CPU that we have talked about.

Now, the connection between the CPU peripheral and memory so this is often done by means of some by means of some bus. So, normally we have got a number of buses the first bus that we talk about. So, is known as the address bus we call it the A bus there is another bus which is the data bus or D bus and there is another bus which he called the control bus or the C bus.

So, this address bus is actually giving address to the memory that is memory is a collection of set of locations where the values are stored. So, whenever we are talking about memory. So, we have to tell which location we want to access. So, if I consider memory as a container like this. So, it has got several slots in it so I have to tell which slot am I referring to. So, that is actually given in the address bus. So, address bus does processor it will give the address on the address bus, the processor will give the address on the address bus.

Then we have got this data bus and this control bus. So, on the data bus the data will be available. So, either the processor as part of it is execution we will try to write something onto the memory or it will try to read something from the memory and same is true for the I O the processor may try to read something from the peripheral device or it may try to write on to the peripheral device. So, that way so this data bus, so it is it can contain the data coming from the processor or it can it can contain the data coming from the peripheral.

So, this address bus you can say. So, this is the direction is from the processor towards the memory or peripheral whereas, this data was that we have so this data bus is bi directional, because I can have this data transferred from the processor to the memory or from memory to the processor. And this control bus so control bus is again the collection of lines that sense control signals to the memory and control signals to the processor to the peripheral. So, these control lines are also coming to the coming from the CPU or the processor where it will give the control signals to these modules.

So, this way conceptually I can say that if this is my CPU and this is the memory, then there are three buses one is called the address bus, one is called the address bus one is called the data bus one is called this data bus and the other is the control bus . So, this

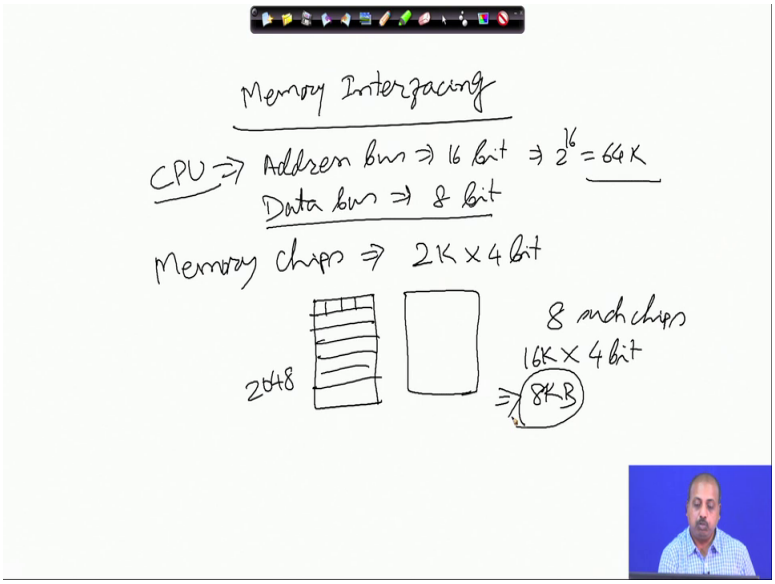
address data and control out of that this address is unidirectional from the processor towards the memory or you can have the data bus which is bi directional. So, it can carry data in both the ways where from processor to memory or memory to processor and we have got the control bus.

Now, so this is how big a memory the processor can handle. So, that is given by the size of the address bus. So, if you look into the initial bus systems or initial processors that were designed the number of bits allocated for this address bus was less, but as we are progressing so the size of this address bus is also increasing significantly and now a days it is common to have sixty 4 bit address bus or even higher than that.

Now, the question is if I have got a k bit address bus using this k bit address bus I can differentiate between 2 power k memory locations. So, just having the processor capability to handle k bit address is not sufficient to tell that the memory size is that much, because this memory that we are putting into the system. So, this should also have 2 power k locations in it and you can understand that if k equal to 64 then this 2 power 64 is a huge number.

So, it is very much it difficult to have that much of physical memory in the system. So, if you are thinking about a low end system which will not which may not have that much memory, but the processor has got. So, many address bits then how can I do this interfacing like how can I connect a memory that has got less number of locations to a processor that has got more number of address lines.

(Refer Slide Time: 06:52)



So, this is done by the by the set of topic which is known as the memory interfacing problem. So, we will start with an example suppose I have got a processor to the CPU where the address bus is 16 bit. So, it can go for 2 power 16 locations that is 64 k locations, and the data bus the data bus from the processor it happens to be 8 bit fine the data bus is 8 bit that is once the processor accesses a location. So, it is expecting 8 bit of data from there.

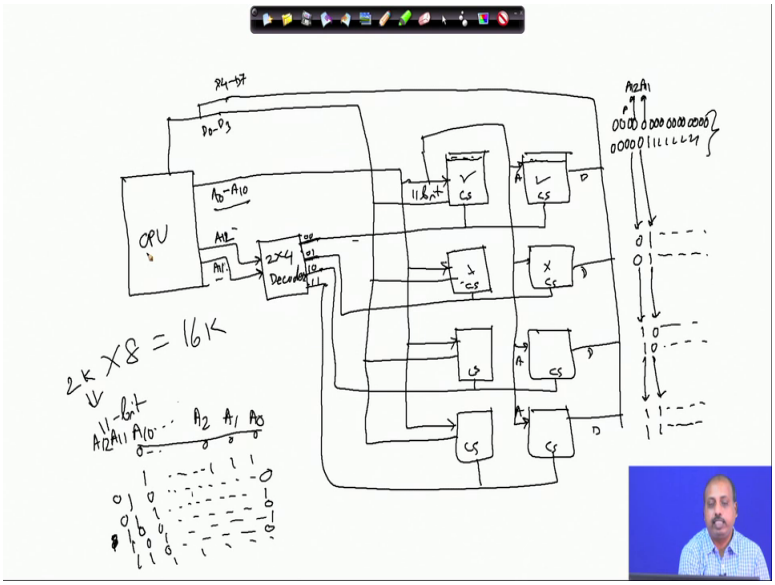
But the memory chips that we have memory chips that we have. So, they are of size 2 kilo into 4 bit, what I mean is that so this a if you look into a particular memory chip. So, it has got 2 k locations in it that is 2 into 1024 the 2048 locations are there and individual locations. So, they are consisting of 4 bits.

So, one thing is obvious that if I have to k 2 a data bus of 8 bit then I need to such parallel modules. So, one module will not be sufficient for giving 8 bit of data that is one side I have solved the problem from 4 bit chips I have got 8 bit data bus created 8 bit bus created, but what about the other side that is I have got this 64 k memory a 64 k address locations addressable by the CPU, but my memory may not be that much. Suppose I have got say less number of chips, suppose I have got only say 8 such chips only 8 such memory chips are available.

So, I have to I have to so what is the total size. So, I have got 16 k location and each of 4 bit ok, that is in terms of bytes we can say it is 8 kilo byte of memory that I am trying to

interface with the system, 8 kilobyte but the CPU has the capacity to address 64 kilobyte, but my system that I am developing. So, it will have only 8 kilo byte of memory. So, it does not may be my system does not require more memory. So, I will be sufficiently it a satisfied with 8 kilo byte of memory because putting more memory means the cost of the system will be high. So, I do not want that.

(Refer Slide Time: 09:50)



Now, how to do this interfacing, so what we have is if we take this as the CPU from the CPU we have got the address and data buses coming out, now I have got 4 bit memory chips. So, I take those chips here. So, these are those 4 bit, 2 k by 4 bit chips I take all the 8 chips in this fashion. Now this memory is 2 k so how what is the number of address lines that we need. So, for 2 k I need 11 bit of address because 2 power 11 is 2048. So, this individual memory chips so they have got the address lines which are 11 bit. So, from the CPU if I say that I have got the address lines, A 0 to A 10.

So, this A 0 to A 10 I connect to all these chips they are address lines. Fine so it is connected to similarly also goes to the next row and it is connected to all of them address lines, but I need to be for like any value of a 0 to a 10 coming. So, it will be selecting some location in each of these chips and for the data bus part I have I am I can do it very simply because I this data bus I have got two parts in it. So, 3 bit a 4 bit 4 bit. So, D 0 to D 3 and the other parties D 4 to D 7.

So, I can say that this d 0 to d 3 values so will be provided by this bank. So, I can connect this D 0 to D 3 here and this D 4 to D 7 will be provided by this bank. So, I connect them here. So, these are the data lines for the memory chips and these are the address lines for the memory chip these are the address lines.

Now, but that is not sufficient as I have said that if you put any address. So, all the so one location in each chip will get selected that I do not want I want that for a address range first 2 kilo address range that is from the address from the address 0000 to say up to 2 k the first chip should be selected. So, if I am writing the address values as say A 0 A 1 A 2 A 10 then A 11 A 12 etcetera then. So, from first for the first all these values being zeros till all these values coming to 1, for this range this first two blocks should be selected for the next range that is from this bit onwards from this address onwards that is A 11 bit is 1 and this other bits are 0 A 10 to A 0 as 0 to this position.

So, second block should be selected second pair should be selected, for the third A what I So, for since now I need to access this third one. So, I can say that as we for the A 12 bit for this 2 word 0, now when the A 12 bits also becomes 1, then when sorry when the A 12 bit becomes 1 then this being 0 now it will be 0000 on words to 1 0, 1 all 1 and finally for 1 1. So, this is going from 0 to all 1.

So, I can say that this range should work for the address value 0000 0000 0000 this 16 bit address from there. So, it should work up to say. So, I am interested about the bits A 11 and A 12. So, this is 15 14 13 12. So, this bit and this bit so these two bits I am interested in, this is A 12 this is A 11 so from 0000 up to this 1111 the first bank should be selected for the second bank this A 11 and this A 12. So, these two values should be 0 1 rest of the values remains unaltered, for the third bank what I want is that this value if this value is 1 and this value is 0 then the third bank should be selected, these values can be anything and similarly for the fourth bank I want that this value and this both of both the values they should be 1 these are do not cares whatever it is.

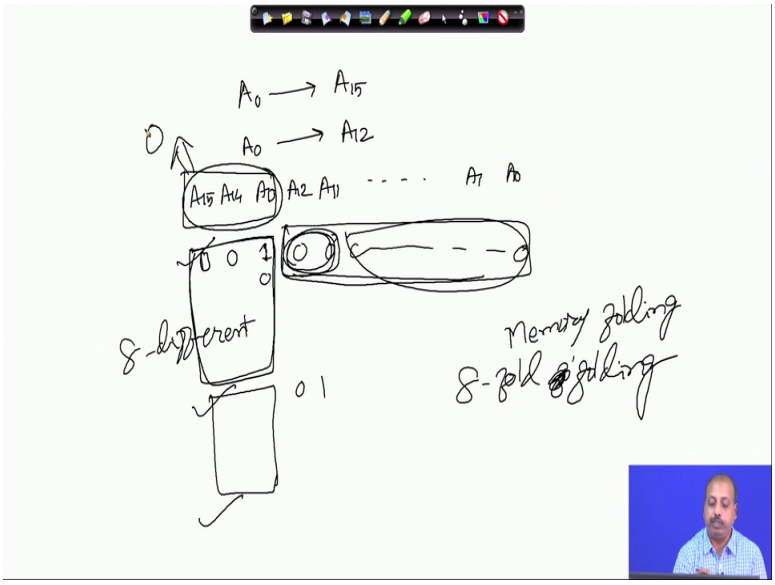
So, this is the range of address is that has to be selected for different values of A 11 and A 12. So, that way the model should be selected for that purpose what we do we put a decoder here. So, this is a 2 to 4 decoder, this decoder takes two inputs from the address line which is A 11 and A 12 now let us make now. So, if it is let us say this is A 12 and say this is A 11 this is A 11 and this is A 12.

Now, you see that it has. So, this is this is a 2 to 4 decoder. So, it will have 4 output lines. So, this output line be will be with this output line will be 1 when the both A 11 and A 12 are 0 0. So, this will be when these bits are 0 1, this will be selected into 1 0 and this will be selected when it is 1 1, now every memory chip has got one special control which is known as the chip select line. So, this chips it is so this line is connected to the chip select signal of this memory block similarly this is connected to the chips select block of the next memory block. So, this one is connected to the chip select of these two, this one is connected to the chip select of these two, and this one is connected to the chip select of these two.

Now, what will happen so for this range of addresses, you see that these 2 to 4 decoder it will output these value as 1 and this will be 0 as a result these chips will not be selected this lower chips. So, they will not be selected whereas, for this higher values for this A 11 A 12 will be 0 0. So, this chip select line will be 1 as a result these blocks will be selected and the whenever the, whatever be the address coming from the processor. So, the content of the corresponding locations from these two chips so they will be available on the on the data bus and those values will be coming to the CPU.

Now, when the address when the CPU generates an address beyond first 2 kilobyte the next it is in the range of the second 2 kilo byte region in that case this chip select line will get selected they it will be output will be 1 and the others will be 0. So, this block this block these two blocks will get selected. And accordingly content of those memory locations so they will come and they will be stored there. So, this way it will continue, so this way this if I have got less amount of memory so it can be we can have this start type of decoder incorporated into the system and this incorporation of this decoder. So this will be helping us in realizing the realizing the less number of chips less number less amount of memory.

(Refer Slide Time: 19:18)



Now, you see that I have said something like I have said that, so the processor has got the address lines A 0 to A 15 and out of that I am ultimately in the previous example that I have considered. So, I can utilize the address line A 0 to A 12. So, if you look into the previous slide previous page. So, you see that we could realize we could utilize A 0 to A 10 A 11 and A 12. So, you could realize A 11 up to A 0 to A 12.

Now, what about the remaining bits like if I am writing like say bits like say A 0 A 1 in this way up to A 11 A 12 then A 13 A 14 A 15. So, these three bits that I have what happens to these bits. If the processor generates some address so you can understand that in the in this case. So, we have not considered those bits A 13 A 14 A 15 with the implicit understanding that the processor will always generate 0 for those address lines, but this is not correct because the processor may generate some addresses in that range also in which this A 13 A 14 A 15 may be 0 for example, the processor may generate an address where say this bit is 0 this bit is 0, but this bit is 1 and these bits are all 0 these bits are all 0.

So, this address so what does this address get mapped in the previous diagram if you see this A 11 A 12 they are 0 0 and these bits are all 0. So, essentially in this diagram so it will get mapped to this first block of first two blocks and the very first location of that block because it is all 0. So, 0 0 A 11 it will being 0. So, this line this select line is will be high. So, these two chips will be selected and A 0 to A 10 being all 0's the very first location of these two chips so they will produce the output and that will be available on the data bus line ok.

So, that is what is happening. So, if this bit is 1 then it is actually the same memory is getting accessed compared to the case when this bit is 0. Similarly whenever this A 11 A 12 is 0 0. So, whatever will be the values of these three bits, so it will have a map to the same memory chips, similarly if this A 11 A 12 these two values becomes 0 1, whatever be the values of these three bits whatever the values of these three bits they will get map to the second memory block that we have got.

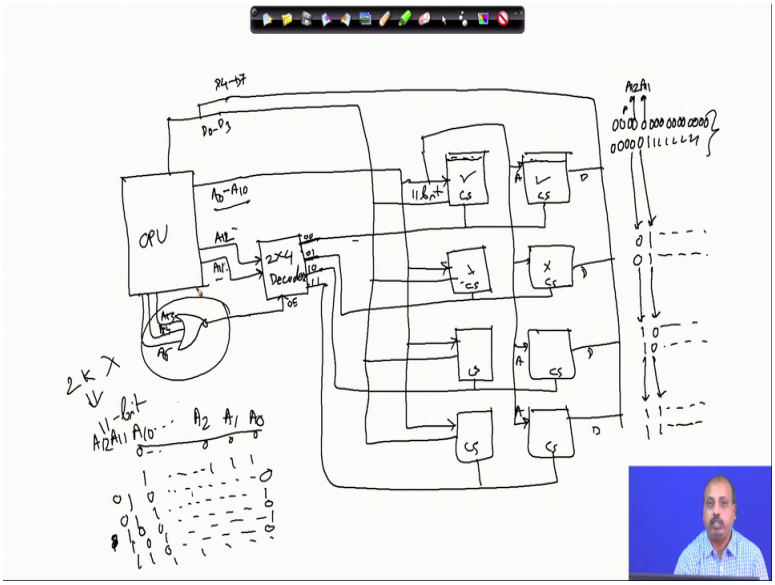
So, this particular phenomena is known as memory folding this is known as memory folding, that is the same memory location is being accessed because of by more than one address. So, I can say that this location this 0000 all 0. So, this location is accessed by several addresses how many addresses. So, it is accessed once here, once here, once here. So, every time it is accessed once and how many such possibilities are there. So, since there are three bits. So, with three bits I can have 8 different cases I can have 8 different cases. So, there is a 8 fold folding. So, I can say in this particular design there is one 8 fold folding ok.

So, this design that we have, so this gives us one 8 fold folding because it is as if for 8 times the same memory location will be accessed and that is quite natural you see that individual memory chips. So, they are 2 kilo and if you multiply it by 8 so you get 16 kilo and as I said that the CPU is generating 16 bit address bus. So, it is a 16 kilo address space the CPU generates 16 k address space and out of which these individual chips being 2 k in size. So, this covers the total of 16 k. So, this concept is known as the memory folding. So, it may be good maybe bad it is good because the programmer may not be concerned about whether using some larger memory space or not larger address or not which is beyond the I given physical memory size at the same time it is bad because of the same reason that two different memory addresses generated by the processor they are getting mapped onto the same address.

So, as a result it is the programmer may get confused the same memory location may get modified by two different statements in a different fashion. So, can we stop this folding definitely we can stop it. So, we can say that that this decoder if you want to stop it. So, what we can we have what we can say that we want this all the three bits that is it A 8 to A 13 14 15 all of them to be 0 then only we will be using this thing. So, if it is more than 0 if all these bits are not 0 then we can say that there is an address error or something like

that. So, if you want to incorporate that idea what you have to do is, you have to you have to just take some extra circuitry for solving this for this problem.

(Refer Slide Time: 25:04)



So, what we do, we take one nor gate and this A 13 A 14 and A 15 we put all of them here. So, this is A 13, this is A 14, this is A 15. So, put them into a nor gate and this nor gate output we. So, this nor gate output will be 1 when all the bits are 1. So, these we can give it as a output enable line of this decoder. So, normally these decoders they have got an output enable line so that means, if this output enable is 1 then only depending upon this A 11 A 12 this decoder will set one of these values to 1, if the output is if the output enable is 0 in that case it will be in a tri state condition.

So, all these output will be in a tri state condition. So, this decoder will not be able to generate any chip select lines. So, all these lines will be in a floating state so none of the chips will get selected. So, by introducing this additional circuitry we can solve memory folding. So, whether we like it or not so that depends on the system that we are considering.