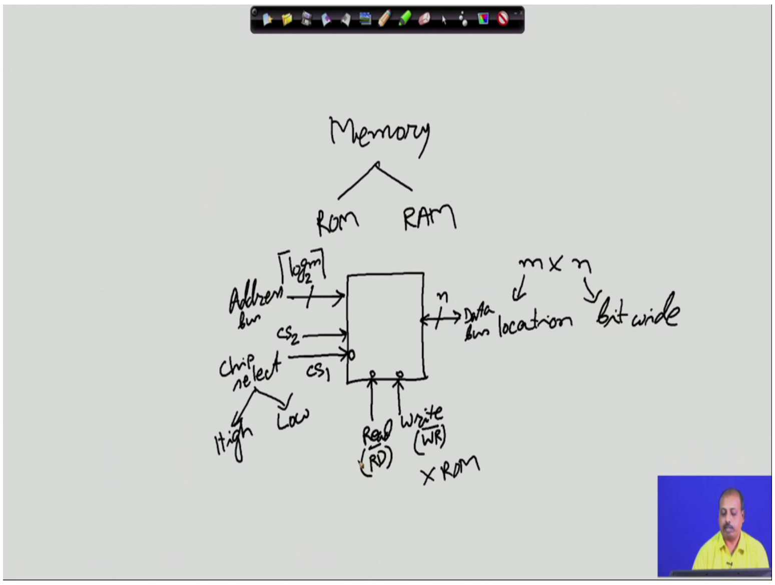
Microprocessors and Microcontrollers
Prof. Santanu Chattopadhyay
Department of E & EC Engineering
Indian Institute of Technology, Kharagpur

Lecture - 05
Basic Computer Organization (Contd.)

As far as memory chips are concerned so they can be classified into two major classes: one is the ROM that is read only memory, another is RAM which is random access memory.

(Refer Slide Time: 00:26)



So, the memory chips that we are talking about, so there are two types one is ROM another is RAM. So, these we know from our digital logic classes. Now as far as the controlling this ROM and RAM chips are concerned from the microprocessor or any other of CPU. So, these chips they normally have a number of standard lines so that they are used for interfacing. So, first one is the address lines so these are the address pins.

Now, So there will be a number of lines like if this a ROM a or this memory is having say m cross n memory, that is there are m locations in the memory and each location is of n bit wide and there are m such memory locations then this address line. So, there I will need log m to the base 2, so many numbers of ceilings of that so many number of lines which will serve as the address line to select individual locations of the chip. So, this is an input to the memory chip.

So, similarly I will have the output which is given by the data. So, this is often known as address bus and this data part is known as the data bus, so this is having n number of lines. So, each since each location has got n bits, so I need n locations to identify to get the data of those n bits values. Now apart from that there are some other controls which are also there, that can help us in our connecting memory chips to the processor. One of the important lines that we have is known as the chip select line.

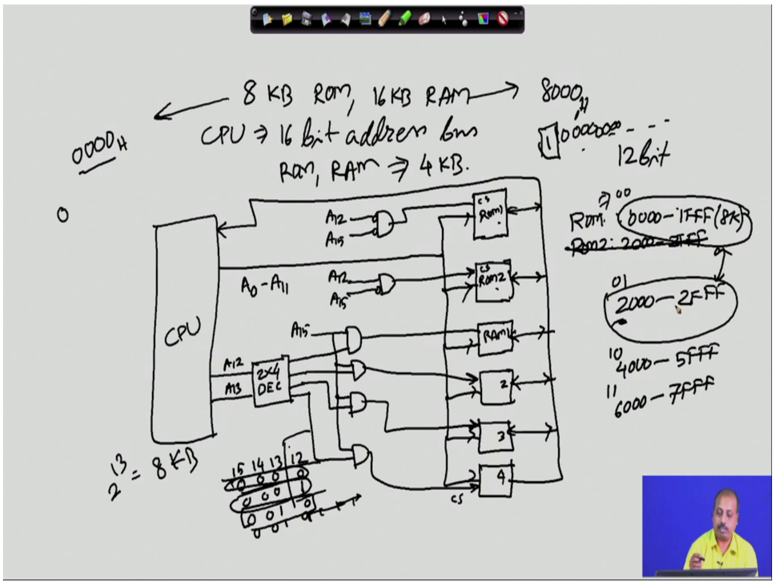
So, this is the chip select line, so if this chip select line is disabled then this memory does not put any output on to the data bus, the memory content will be available on the data bus only when the chip select line is enabled and this may be active high or active low. So, this may be active high or active low. So, the connection that we have shown here is for active high connection, if you have a bubble here so that will mean that it is active low connection.

So, when the chip select line is 0, then the chip is enabled otherwise it is disabled and also depending upon the type of operation that we want to do on this chip, on this memory. So, we will have a 2 more control lines 1 is called the read and the other is called write. So, they are originally written as read is represented as RD and write is represented as WR and most remain most of the cases this read and write control they are active low. So, they are there are bubbles here, so we get the lines as read bar and write bar so these are the lines.

Now, we can find some chips that have got more than one chip select line. So, in this whole what can happen is that so this may be chip select one or CS 1, there may be some another chip select line which is called CS 2 and the logic is that for the chip 2 operate both CS 1 and CS 2 be enabled. So, to get the output and if this helps in the decoding process, when you are connecting this chip to processor 2 to ensure that we select the proper chip, so the decoder has to be there as we have seen in the last class. So, the decoder design will be simple if we have got multiple chip select lines.

Now, depending upon the type of memory though so this read bar and write bar lines will come, for example for a ROM there is no option to write the content on to the ROM. So, this write bar line will not be therefore ROM. So for RAM I will have both read bar and write bar lines, for a ROM will have only read bar line. So, let us take an example and try to see how this can be utilized in represented in the connecting some memory chips to a processor.

(Refer Slide Time: 05:09)



We take an example where I have to I need to connect eight kilobyte of ROM and 16 kilobyte of RAM via system, with a CPU that has got 16 bit address bus.

Now, 16 bit address bus means the address is can go from 0 to 64 k. So, I can connect 64k space memory to the processor, now this ROM and ram so it is required that this ROM will be starting at location 0000x; whereas this RAM it should start at location 8000x. So, this will help us in distinguishing the RAM space and ROM space. So, that is useful in many cases. So, we do not want that may be for the future extension we would like to put more ROMs into the system. So, that if I keep some gap between ROM and RAM connected to the system their addresses, then I will be able to extend the ROM space later.

So, this may be the requirement, so accordingly we can design a system where this connection will be there like that and it is given that the memory chip that is the that are both this ROM and RAM chips, so they are of size say 4 kilobyte. So, each of them is 4 kilobytes. So, took a make an 8 kilobyte ROM, so you need to too such ROM chips for making a 16 kilobyte RAM we need 4 such RAM chips.

So, how this system will look like? So, if we want to get it representation so suppose first to 2 chips that I put so they are the ROM chips. So, these are the 2 ROM chips that I have put, so we call it say ROM 1 and ROM 2. Now after that this is ROM 1 this is ROM 2 like that and then I will have so that will. So, each ROM is 4 kilobyte. So, that makes 8 kilobyte then this 16 kilobyte of RAM. So, that can be made by putting for such RAM chips so these are 4 such RAM chips, which makes it 16 kilobyte of RAM space; so they are all rams RAM 1 2 3 and 4.

Now, this side I have the processor, this side I have that CPU from where I have got that address bus and data bus lines going out, so since each chip is 4 kilobytes. So, to address 4 kilobyte I will need 12 bits of address, 12 bit of address lines will be sufficient to differentiate between the locations of individual chips. So, from the address bus what I do I take the lines A 0 to A11. So, that is the 12 bit lines and that goes to the address line of individual chips. So, they go to the address line of individual chips this part is done and similarly that data bus that we have. So, data bus will connect to all the chips. So, the data bus is 8 bit, so data bus connects to all the chips ok.

So, this is bidirectional this data bus is bi directional and the address bus is unidirectional, so that is done here address bus is like this. Now, so this is the basic connection now I have to select the chips. So, for the chip selection what we do that see you look into the difference in the address that the ROM starts at 0000, so, it the most significant bit. So, this is a 16 bit number the most significant bit there is 0 and here this most significant bit, first digit is a most significant digit here will be 1 because the 8will be represent 1000 followed by 4 zeros. So, this way that 16 bit number will go and the most significant digit there is 1.

So, we use that to differentiate between the ROM space and the RAM space. So, for differentiating between these 2 ROMs, what we do we put this type of gate. So, where this line A 15 and A12, so they are connected here. So, A12 and A15 the lines being 0 will say that this will connect to the chip select off this ROM chip. So, this is the chip select of this ROM chip similarly, the second ROM chip will be selected provided this A15 line is 0, but A12 is 1 in that case this chip select will be generated and it will be connected to the second ROM s. So, this is the chip select of the second ROM.

Now, for the remaining RAM chips, so what we can do we can put a decoder a 2 to 4 decoder and in that 2 to 4 decoder we take the lines A12 and A13. So, this is my line A12 and this is the line A13. So, they are coming here and accordingly this will generate 4 outputs and these 4 outputs I can connect to ram, but I need that the line the bit A15 that should also be 1, for making the line A15 to be equal to 1. What I need to do is I need to connect it in this fashion, so this should go to the chip select of RAM 1. Similarly the second one can be connected like this, A15 line this is ended with this output of the decoder and it is going to the chip select of the second RAM chip; similarly this line a fifteen ended with the third output of the decoder, that goes to the chip select line of the third chip and this A15 line ended with the fourth output of the decoder. So, this will go to the chip select of the fourth RAM chip.

So, this way now you see that. So, this is a 2 to 4 decoder so that decoder, so now you see that we can this RAM has been put into the address space 8000 onwards and the ROM has been put into the address space 0 onwards, now I said that there. So, what is the address range for the first ROM? So, for ROM 1 the address range is basically 0000 to 1FFF. So, this is the FFF. So, this is the address range for the second ROM. And of course, since you see there will be folding because I have not used the line A13 and A14, while lab decoding this ROM chips. So, they will get folded as a result, I will also get this. So, these actually these ROM chips, so this is a 1FFF. So, this is your 12 plus 13 bit, so 13 bit means to power 13.

So, it is basically 8 kilobyte so this is 1FFF. So, this is the complete ROM space that we have and out of that so this is rom. So 8 k here out of that first 4 key locations will belong to ROM on and the second 4 k location they will belong to ROM 2.

Now, there will be folding as I was telling so for if you again come to the next address space like what happens is the lines A13 and A14 they are taken as 0, they are taking there they have not taken in the in this diagram. So, in this particular case what is happening is that A13 A14 is remaining as 0. Now if a A13 A14 b creates a value 01, then what you are getting is the address range 2000 to 2FFF.

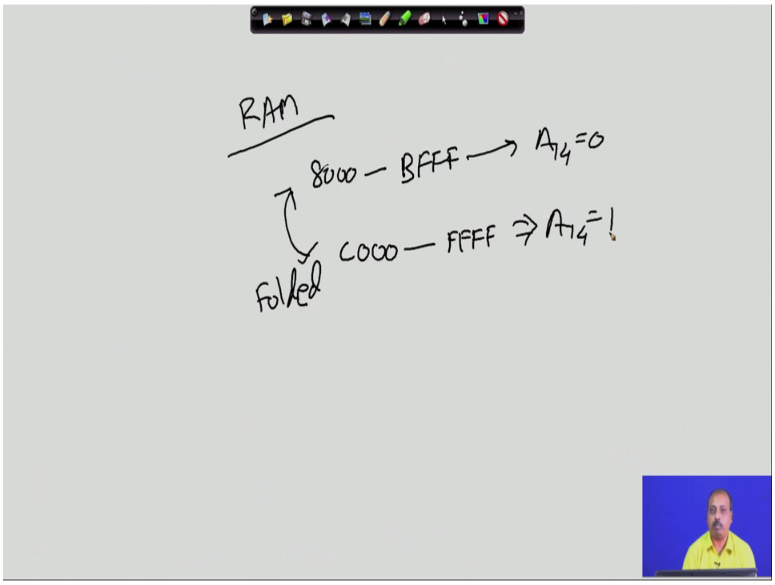
So, this is the next address space that we are getting, so this is basically a folded address space of the first the first address space that we have got. So, these 2 address spaces are same they map to the same set of memory locations, but because of folding they are coming and similarly the 2 bits a thirteen a fourteen. So, they may be 1 0 and as a result you will again get the same address space repeated from 4000 to 5FFF and when the bit 2 bits become 1, so you get the space repeated from 6000 to 7FFF. How are you getting this 2 4 6 these numbers? So, if I just write the bit numbers 15 14 13 12 the last 4 bits of this address space.

Now, for the RAM selection this bit is always 0. So, out of that so this bit number 2 up to bit number 12, so these are they have been utilized in the ROM selection, now if these 2 bits are 00; now this can go to this can and also now this bit can remaining bits a 0 to a 12, it can vary from all 0 to all 1. So, that way we get the first address space which is 0000, it can go up to the situation where these bits remain 0, but this is 1 ok.

So, that way I when this bit is when this a 12 bit is 0 it is scattered by the first ROM, when this a12 bit is 1 it is scattered by the second ROM. So, that way it is generating this whole address space. So, this first 0 that we have here is corresponding to this 4bit 0000 here and this first 1 that we have here. So, that corresponds to this 1 here and now if we are going for the next setting of A13 A14. So, this is 0 1 so this bit is 0. So, this value is basically 2, so you see that this next address space starts at 2000 and go up to and goes up to 2FFF, where these bits are 001 and then it can go up to this, all these values it can go up to all ones values. So, that way it can go up to this 2FFF configuration.

So, this way this ROM chip and RAM chip they can be connected, now if you are looking into the RAM space.

(Refer Slide Time: 17:57)

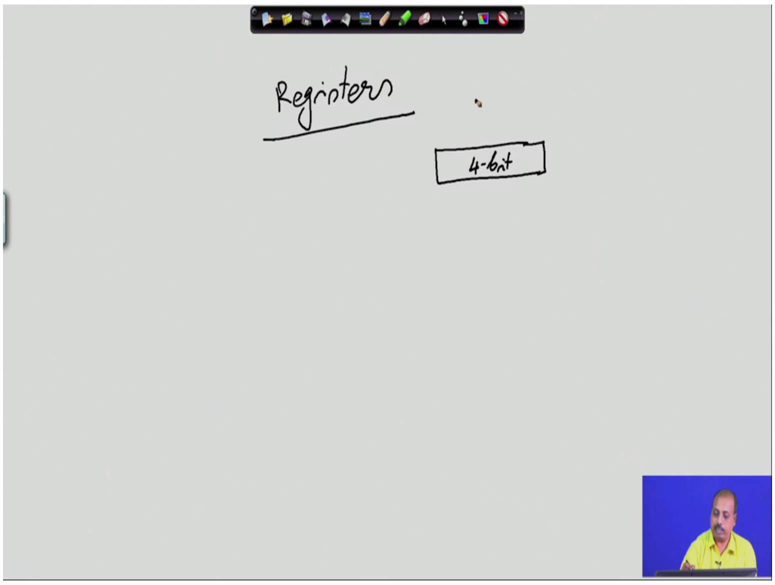


If you are looking into the RAM space that we have connected here, then this RAM it will have the address range 8000 to BFFF, as the first address space and there will be another folded address space which is C000 to FFF, so this will be the folded address space. So, the there is a folding here and this is again guided by that it is again guided by this selection like A15 has been used A12 A13 has been, but only the A14 line is not being used. So, if you just put values of a fourteen to be 0 or 1. So, when A14 value is 0, you get the first address range and when A14 value is 1 get the second address range.

So, this value will come when A14 is equal to 0 and this address range will come when A14is equal to 1. So, this way you can you can interface ROM and RAM chips to a system for getting the whole design memory getting, the memory map as you desire for your system.

Next will look into another important topic that we have that will meet for this microprocessor a microcontroller force, which is the concept of the resistance.

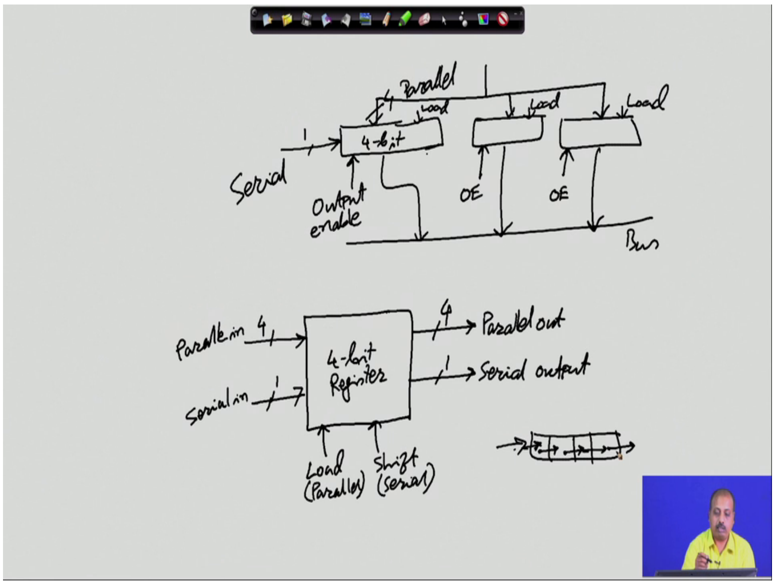
(Refer Slide Time: 19:37)



So, in the last class I have introduced the concept of registers and I said that a register it is a sequential element and it stores the values of some bits and these values can be stored in different ways it you know. So, I can have this if this is a register and I say that this is say a 4 bit registers so it stored a 4 bit value.

Now, the requirement is that these 4 bits that we are talking about. So, these 4 bits may be loaded parallel or so if this is a 4 bit register.

(Refer Slide Time: 20:20)



Then these 4bits may be a loaded parallel or these 4 bits may be loaded serially one bit at a time. So, one is the parallel load and other combinations that I can have is a serial load, both are possible; similarly I can happen that I have got a number of such register instead of having a single register, I have got a number of such register. Now I need their output to come on to a common line which we call red bus, so they are outputs are connected to the bus.

Now at any point of time I may want that output of only one register be available and others are not. So, for that purpose what I need is these registers they should have another control which is output enable and when this output enable signal is given then only the output of the register be available on to the bus. So, this is the bus similarly this has got separate output enable line, this also has got a separate output enable line and this output enable lines. So, they whenever the control is given then only the output should be available on the bus otherwise not.

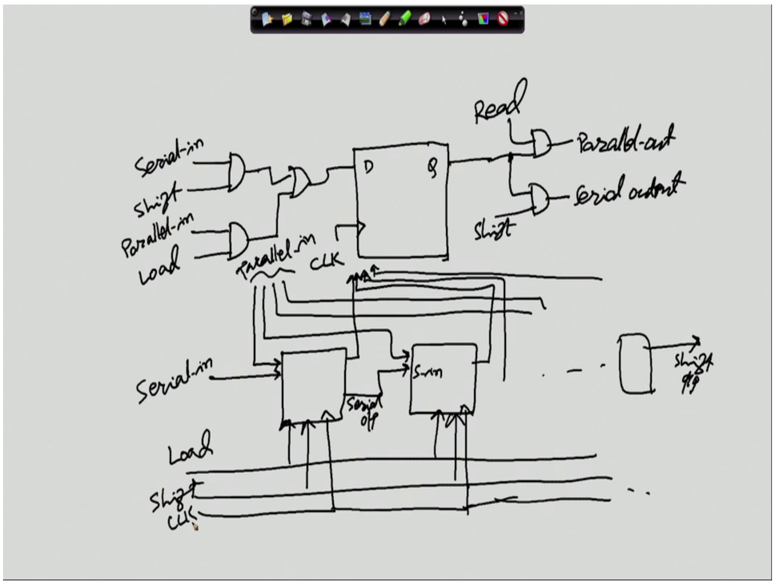
Similarly, when I am feeding input to this register, so instead of feeding them differently I can feed them from a single 4 bit bus and there can be a separate signal which we call load. So, there can be a load signal and using this load signal the values should be loaded into the register. So, if I put a value on to this input bus and give load signal to the first register, then the value should be loaded into the first register. If I give load signal to the second register then the value should be loaded in the second register, so that sort of facility is needed.

So, we can very easily design such a circuit where these individual stages individual registers they can have loaded and output enable control lines. So, ultimately I have a register will be something like this, if this is a say 4 bit register. If it is a 4bit register then I should have the facility to parallel load the line. So, this is 4 bit parallel data, parallel input or I can have 1 bit serial input, I can have a 4 bit parallel output. So, this is 4bit or I can have a single bit serial output.

So, this is a serial output can have single bit serial output and apart from that I should have control like whether I want a parallel load, so this load signal. So this normally stands for parallel or I have for the serial load. So, the control signal is generally called shift this is the serial load it is called shift because, when you are if this is a register and it has got 4stages, then when you are loading the next value serially, here the previous value that was here get shifted in the next position, this goes for next and this value the final value gets lost or it is available at the serial output.

So, this is known as the shift output, a shift control or serial control. So, these are the control lines that I should have in the register; now how to design such a thing such a system so that will see.

(Refer Slide Time: 24:53)



So, this Individual stage of the register; so it consists of a flee d type flip flop. So, we have got d and q lines there, now what I want is if the serial is the shear shifting is there then I should give the serial value and that should be shifted in. So, 1 input for the D can come from this serial input serial input and the shift control.

The other possibility is that I am loading it parallel. So, now loading it parallel then I should have this parallel in and the load control and depending upon whichever control is given, so I can have an or gate they can connect to this d flip flop and for the output part I can have 2 output parallel output and serial output. So, for that matter so I can have 1 and gate here where this value is coming and the other value that is there you say read. If I give the read signal so this is the parallel output and if I give a shift signal, then this will be the serial output.

So, this is what we are looking for plus the clock signal will be there, everything is synchronized with a clock. So, that clock signal will be there. Now if you are connecting

2 stages say I am just showing 2 stage of that register, though there can be 4 such stages. So, I am just showing 2 stages now this parallel input line. So, though it is so there are 4 lines. So, out of that the first 2 lines I am showing there are 2 more lines, which will be connected to the next stages that are not shown here. So, this is the 4 bit parallel input, out of that the first bit is connected to the first flip flop, second bit is connected to the second flip flop like that, but serial in. So, the first bit is connected directly. So, this is the serial in and this serial output of this flip flop, should be connected to the serial in of the next flip flop.

So, this is the serial output of the first flip flop connected to the serial in of the next flip flop, then the I will need the parallel output line as well. So, this parallel output lines will be taken from these individual flip flops and they will be taken into they will form this 4 bit output coming from the successive stages, they will make the 4 bit output parallel output and this serial output for the last flip flop that we have whatever be the shift out. So, that will be this serial output will be the shift output.

Now, other controls like this the load control and the shift control they should be given to all the stages, for doing the operation. Clock signal should be given to all the stages is the clock signal, the clock should go to all the stages. Then this structure will behave as a as a registered, where you can parallel load the value or you can serially load the value, you can take a parallel output you can take a serial output. So, all those operations are needed when we are incorporating these registers inside microprocessors.