Lecture - 17
Example II

In the last class we are discussing about the interfacing of the memory to the 8086.

(Refer Slide Time: 00:37)
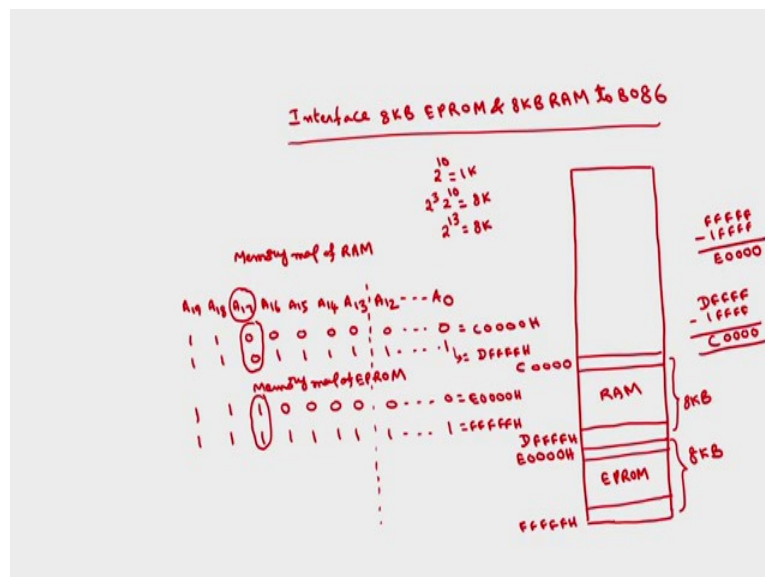
So, we have discussed about example interface 8 kilo bytes of the EPROM and 8 kilo bytes of random-access memory to 8086. So, in the last class we have discussed about the address map of this 8 K EPROM and 8 K RAM. We assumed that, so out of this I mean 1 megabyte. So, I want to interface this EPROM in the last 8 kilo byte of the locations.

So, the ending address will be FFFFH. So, for 8 K you have to subtract 1 FFF ok. So, FFFFF minus 1 FFF will be E0000H. So, so starting address is E0000H. Here I want to interface the EPROM this is total 8 kilo bytes. And I want to interface RAM just above this EPROM. So, they address which is one less than and this one will be DFFFF and to get the starting address you have to subtract 1 FFF.

So, DFFFFH minus 1 FFFF, so this will be 0000 C0000 this will be C0000 to DFFFF ok. Now, in order to interface this, we have to write the memory map of this particular

addresses. So, what is the memory map of RAM? So, totally we have twenty address lines A 19, A 18, A 17, A 16, A 15, A 14, A 13, A 12 so on up to A naught, okay. So, in case of 8 kilo bytes 8 kilo byte of the memory how many address lines will be present for 8 kilo byte?.

So, we know that 2 raise to the power of 10 is equal to 1 kilo byte 1 K. And if you want 8 K you multiply on both sides with 8 which is 2 cube 2 cube into 2 to the power of 10 will be 2 cube is 8 8 K. So, total we have 13 address lines for 8 K memory. So, A 0 to A 12 these are the address lines of the microprocessor has to be connected to 8 kilo bytes of the memory. Memory will be having some address lines depends upon the capacity, for 8 K memory will have 13 address lines.

So, I am going to connect this A 0 to A 12 to the corresponding address lines of the memory ok. So, what is the starting address? Starting address is for RAM C0000H. So, what will be this bits A 19 into A naught, C is 12 which is 1100 and 0 is 000 and then all 0s is C0000H. So, this is E0000 D00 minus 1 is C00 this is C00H then ending address will be DFFFFH. So, this will be D is 13 1 1 0 1, F is 1 1 1 1 and all 1s this is the address corresponding to DFFFFH.

Similarly, what is memory map of EPROM? The starting address is E000H; E is nothing but 14 which is 1 1 1 0 then all zeros is E000H and ending address is all ones FFFFFH. This is the memory map by varying this A naught to A 12 from 0 0 0 0 to all ones we can choose any of the location from 8 kilo bytes of the memory ok. Now, what are the address in that are left A 13 to A 19 are left.

So, now we have to use this A 13 to A 19 to select the chip selects of this particular memories. And also we know that in 8086 the memory is available in 2 banks EVEN and ODD banks ok. So, total we have 4 memories 2 EVEN memories 2 ODD memories and how this has to be connected?
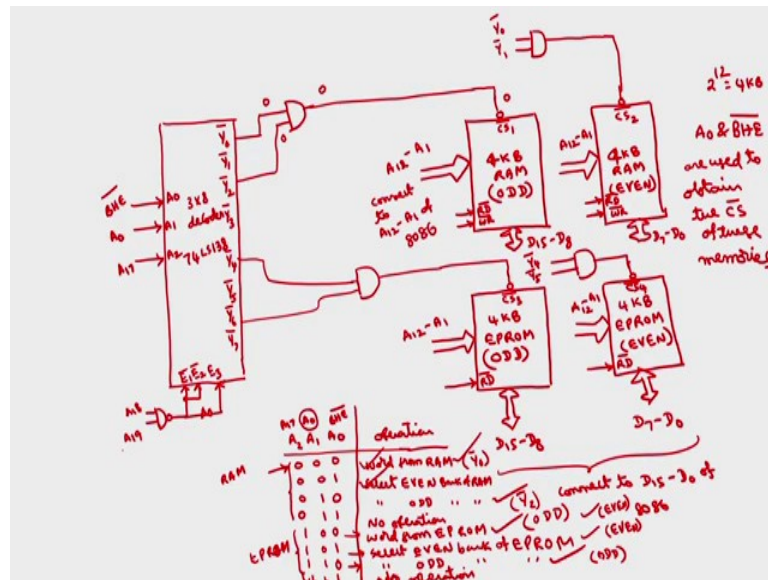
So, we have here 8 kilo bytes of the RAM ODD 8 kilo bytes of the memory read only memory EVEN. Similarly, we have sorry this will be 4 kilo bytes only because we have divided into EVEN and ODD half of this memory will be available. Then we have 4 kilo bytes of EPROM ODD 4 kilo bytes of EPROM EVEN.

So, what are the different signals for this 4 K RAMs and 4 K RAMs. So, we will be having how many data lines? And how many address lines? So, this will have 12 address lines because 2 raise to the power of 12 is 4 K, so this is A 11 to A 12 to A 1. So, 1 to 12 is total twelve similarly here also we will be having A 12 to A 1, why you have not used A naught is? A naught we will use to select the banks for EVEN bank A naught is always 0.

So, we will use this A naught to select the bank C to find out the chip selects of this. If we call this one as chip select 1 bar, chip select 2 bar, chip select 3 bar, chip select 4 bar to determine this values we will use A naught and BHE bar. A naught and BHE bar bus high enable bar are used to obtain the chip selects of these memories, okay.

Then here also we will be having 12 address lines, here also we will be having 12 address lines A 12 to A 1, A 12 to A 1. This you have to connect to A 12 to A 1 of 8086. So, all this 4 I mean memories address lines you have to connect together and that has to be connected to A 12 to A 1 of 8086 ok.

Similarly, data this will give you 8 data lines this will give 8 data lines. So, here this ODD will give D 15 to D 8, EVEN will give D 7 to D naught total we have 16 bit data. If you want 16 bit data you have to enable both EVEN and ODD banks, if you want only 8 bit we have to enable either of these banks. Similarly, here also you will be having two data buses this will give D 15 to D 8, this will give D 7 to D naught. This you have to connect to D 15 to D naught of 8086 and EPROM will be having only read bar signal this is read bar signal. This is also having read bar signal you connect this to read bar signal of 8086 ok.

So, while discussing about the pin diagram we have discussed that 8086 will be having read bar signal. Whereas, in case of RAM will be having both signals read bar as well as write bar you connect to the corresponding read bar and write bars of 8086. So, these are the different signals. Now, how to select the chips, so that this memory RAM will operate at the address that have specified in the next slide which is from C0000H to DFFFH. And EPROM will interface starting from the address E0000H to FFFFH ok.

So, if you observe here see in this memory map, so this A 19, A 18, A 17 they are all A 19, A 18 are one in all the cases ok. So, whereas, this A 17 to A 13 is 0 all zeros A 17 to A 13 for the last address is 01111 ok. So, we can use here because we have 4 banks okay and we have A naught as well as BHE bar to select this chip selects we can take a 3 to 8 decoder. So, I am going to take a 3 to 8 decoder here, so 3 to 8 decoder.

So, if we take 74138 is the IC 74 LS 138 is 3 to 8 decoder. So, this will have 3 enable signals E 1 bar, E 2 bar, E 3, E 1 bar and E 2 bar are active low signals and E 3 is active high signal. And we will be having three inputs A 2, A 1, A naught and 8 outputs Y 0 bar, Y 1 bar, Y 2 bar, Y 3 bar, Y 4 bar, Y 5 bar, Y 6 bar and Y 7 bar.

So, in order to make this E 1 bar, E 2 bar, E 3, so you have to use this A 2, A 1, A naught from the memory map. So, in case of memory map if I choose this ram, so in case of RAM A 17 will be 00 always whereas, here A 17 is 11 ok. So, the only difference between this RAM memory map and EPROM memory map is A 17 is 00 from starting to ending address of RAM, and A 17 is 11 for a starting to ending address of EPROM.

So, here if we observe this A 17 A 17 is 0 in case of RAM, A 17 is 1 in case of EPROM ok. This A 17 I am going to connect to A 2 of this decoder and A 0 and BHE bar as I

have told we have to use for selecting the chip select. So, I will connect here A 0 here BHE bar, then what about the remaining address lines A 7 I have to use it.

So, the remaining are A 13, A 14, A 15, A 16, A 18 and A 19 okay. A 18 and A 19 are always ones only ok. So, in here in case of EPROM this A 13 to A 16 starting address they are 0 ending address they are ones ok. So, I can use this A 18 and A 19 for generating this both are ones to generate 0, so we can pass through the NAND gate. And if we pass through the NOT gate this will becomes 1, this is A 18, A 19 ok.

Now, here how to select CS 1, CS 2, CS 3, CS 4? So, this depends upon this A 0 and BHE bar, so I can form a table now. So, this BHE bar A 2, A 1, A naught of decoder; this has been connected to A naught is connected to BHE bar of microprocessor. And this A 1 is connected to A naught of the microprocessor and A 2 is connected to A 17 of the microprocessor ok. So, if we have 000 we have total 8 combinations 000 say if all are 000 Y naught will be selected ok.

So, Y naught becomes 0, output will be Y naught will be 0. So, if it is 000 what is the operation? So, A naught is 0 this A naught is 0, BHE bar is 0 means it is a word operation. And because A 17 is in case of RAM A 17 is 0, in case of EPROM A 17 is 1, 0 means RAM 1 means EPROM, so A 17 is 0 means this represents RAM. So, word from RAM this is the operation here, word from RAM means you have to select both the banks ok.

Then if I take 001 what happens? BHE bar is 1, so bus high enable will be disabled okay only EVEN bank will be selected because A naught is equal to 0 and 0 means RAM ok. So, this will be EVEN bank of RAM, so it will selected, select EVEN bank of RAM fine, then 010. Now, BHE bar is 0 means bus high enable is 0, A naught is 1 means ODD bank, A 2 is 0, so RAM.

So, select ODD bank from RAM only. 011 this is no operation because both A naught and BHE bar is disabled, so not at all selected. So, this will not be allowed and if I take this 4 will be for RAM. Now, next 4 will be for EPROM will be 100, 101, 110, 1 1 1 these are all for EPROM. So, O 2 is 1 means EPROM by default because of that address map that I have discussed in the previous slide.

So, A naught is 0 means A naught is 0 means this one will be word from EPROM, because both BHE bar is 0 as well as A naught is 0 ok. 01 means select EVEN bank of EPROM, 10 means select ODD bank of EPROM and here no operation.

Now, how to connect CS 1? So, CS 1 in order to select CS 1. So, what is CS 1? the ODD bank of RAM ODD bank of Random Access Memory ok. So, in which cases this ODD bank of random-access memory has to be selected? selected means chip select should be 0 ok. So, here if 0000 is there. So, if I apply 0000 here, then Y 0 will be 0 remaining all will be 1 because this is active low signals Y 0 bar, Y 1 bar, Y 2 bar and so on.

Now, so in order to select CS 1 how to go I mean obtain the expression Boolean expression for CS 1 ok. This is 4 kilo bytes of the ODD RAM bank, this is ODD RAM bank. Here also this ODD RAM bank has to be selected as well as here also, because if I want to select the word both EVEN and ODD banks has to be selected.

So, in the first combination also we need the first RAM ODD bank and in the third combination also we need this. So, in the first combination which of the output will be selected Y 0 bar this is Y 0 bar, and this is Y 2 bar. So, whenever either Y 0 bar is 0 or Y 2 bar is 0 then CS 1 of the first, I mean a 4 K RAM ODD has to be 0 ok. So, what will be the logic? You simply connect Y 0 and Y 2 bar through NAND gate to the chip select of this.

So, if either this is 0 what will be the output? Output will be sorry this is not NAND gate this is should be AND gate. So, if Y 0 bar is 0 output is 0, if only Y one bar is 0 output is 0 if both are 0 also output is 0 ok. So, if a both are ones then output is 1 ok. So, in order to select this ODD bank of the RAM either Y 0 bar should be 0 or Y 2 bar should be 0 or both should be zeros in any case chip for as is selected.

So, this is how we can derive the expression for chip select of the first ODD RAM bank. Now, what about CS 2 bar? How to generate this signal for CS 2 bar? In which cases CS 2 bar has to be selected? CS 2 bar is EVEN RAM bank. So, this can be either the first one or the second one. In the first its word, so both EVEN and ODD has to be selected in the second it is only exclusive EVEN of RAM ok, so in these two cases.

So, what will be this output of this chip select signal? So, here also you have to connect to AND gate the output of the AND gate you have to connect to the chip select of this

EVEN RAM. What will be the inputs now? Because the first two first two are Y 0 bar and Y 1 bar, clear this is about the RAM selection. Now, coming for the EPROM, so in the EPROM, what about the ODD ? chip select of ODD bank?

So, chip select of ODD bank will be enabled in this case as well as in this case these two for ODD. Whereas, these two for EVEN, the first I mean the 100 combination will be selected for both EVEN as well as ODD. So, what will be this inputs for the AND gate? Here also we have to connect the output of AND gate to chip select of this.

So, what will be the input? For the ODD bank either this can be 100 or it can be 110 means 4 and 6, 4 and 6. Similarly, for chip select 4 another AND gate output you have to connect chip select what will be the inputs 4 and 5, 4 and 5 this will be Y 4 bar, Y 5 bar. So, you see about the complete interfacing of 8 kilo bytes of the RAM and 8 kilo bytes of the EPROM to the 8086.

(Refer Slide Time: 26:54)
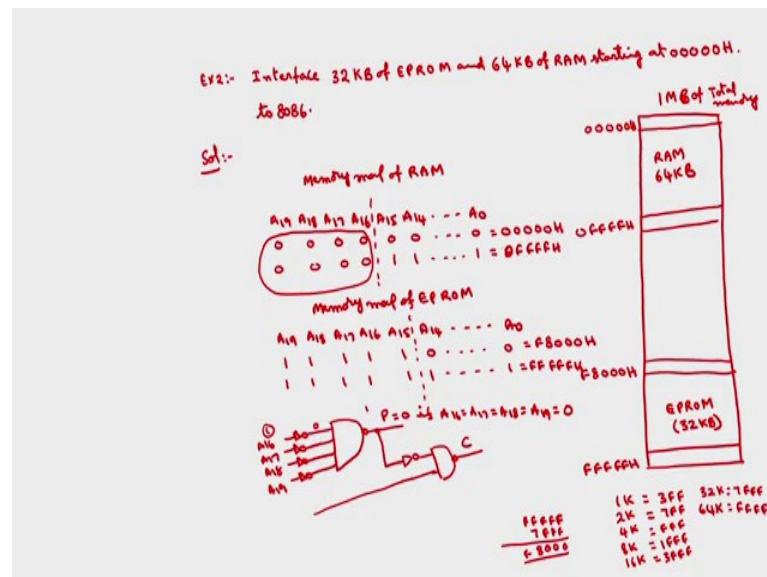
So, basically you have to draw the memory map and from that you have to choose the corresponding bits. I will take one more example where if we have the different sizes of the RAM and ROM. Interface 32 kilo bytes of EPROM and 64 kilo bytes of random access memory starting at 00000H to 8086 microprocessor.

So, the memory map of 64 kilo bytes of RAM is given the starting address. Whereas, the memory map of EPROM is not specified by default as I have discussed in the earlier

class also you have to connect EPROM at the end of the memory. Because whenever the microprocessor is reset the microprocessor will goes to FFFF0 location.

So, by default if address is not given you have to connect the EPROM in the last locations of its total memory map here you have to connect a EPROM. So, what is the ending address is FFFFFH this total 32 kilo bytes. So, what will be the starting address? As I have discussed in the earlier class 1 K means it is 3FF, 2 K means total you have to add 7FF, 4 K means FFF, 8 K means 1FFF, 16 K means 3FFF, 32 K means 7FFF. And 64 K means FFFF, these are the values that you have to add to the starting address to get the ending address correspond 1 K, 2 K, 4 K, 8 K, 16 K, 32 K and 64 K ok.

So, if the ending address FFFF this is 32 kilo bytes to get the starting address you have to subtract 7FFF ok. So, what will be this 7FFF if you subtract? From FFFFF if you subtract 7FFFH, 0008F F8000 H. And RAM it is given that you have to connect at the starting of the this total 1 megabyte of the memory 00000H.

And what is the capacity 64 means ending address will be 0FFFFH, this is RAM of capacity 64 kilobytes. Now, in order to connect this again you have to draw the memory map. So, what is the memory map of RAM? Starting address is 0000 A 19, A 18, A 17, A 16, A 15, A 14 so on up to A 0.

The starting address all zeros is 00000H, ending address is 0FFF H 0 is 4 zeros FF means all ones is FFFFFH. And in this particular 64 kilo bytes of the memory will be having how many address lines? 16 address lines. This is the memory map of, so A 0 to A 15 you have to connect to A 0 to A 15 of 8086.

Now, only this address line that are available to find the chip select are A 16 to A 19. Similarly, what is the memory map of EPROM? A 19, A 18, A 17, A 16, A 15, A 14 so on up to A 0. Starting address is F8 F8000H F is 1 1 1 1, 8 is 100 all remaining all are zeros. This is F 8000H and ending address is all ones 1 1 1 1 1 1 all are ones this is FFFFFH. Now, here how many address lines will be there for 32 kilo bytes? Only 15 address lines, so A 0 to A 14 ok. Now, in order to find out the chip selects of this one ok.
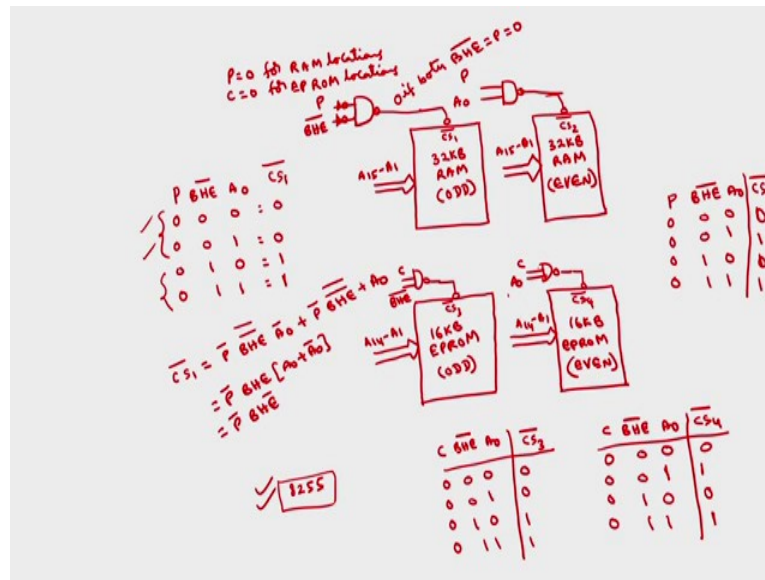
So, you have to use a four memories total 2 EVEN and 2 ODD memories, similar to the previous example. Out of 64 kilobytes of the RAM, 32 kilobytes of the RAM we have to use as ODD bank. And 32 kilobytes of the RAM you have use as EVEN bank. Similarly, out of 32 kilobytes of EPROM you have to use 16 kilobytes of EPROM as ODD bank and 16 kilobytes of the EPROM for EVEN bank for EVEN bank.

So, what are the corresponding address lines? We have to connect to the corresponding address lines of the 8086; 32 kilobyte will be having 15 address lines A 15 to A 1. This is also A 15 to A 1 this you have to connect to the A 15 to A 1 of 8086. Here also you are not using A naught, because A naught we are going to use to find out the chip select of these ICs.

Similarly, here this will be having only 14 address lines, so you have to connect from A 14 to A 1 this you have to connect to corresponding microprocessor. So, all the 4 buses you are going to connect to the common bus of 8086. Now, the question is, so how does this 4 devices will be different, at any time only one of this memory will be selected, so the data transfer will be taken place with that particular memory ok.

Even this common bus of 8086 is connected to all the 4 memories at any time only one of the memory is selected. So, that that the data transfer takes place between the selected memory ok. Here you can use a decoder or else use the other concept directly by using the gates ok, because here RAM and ROM are having different memory sizes.

So, RAM size is 64 kilo bytes EPROM size is 32 kilobyte ok. So, here you can observe that, so in case of random-access memory this A 16 to A 19 will be always zeros ok. So, I will select a signal using NAND gate whose output will be 0 if A 16 to A 19 is 0. So, you can connect like this NOT gate you give the input as A 16, A 17, A 18, A 19 and you call this output as some P.

So, P is equal to 0 if A 16 is equal to A 17 is equal to A 18 is equal to A 19 is equal to 0 otherwise P is equal to 1. If anyone of this A 16 to A 19 is 1 output will be 1 ok, because if anyone of this one is say A 16 is equal to 1. For example, this will be 0 if one input is 0 for the NAND gate output will be one regardless of the other inputs ok.

So, here this P is equal to 0 only if A 16 to A 19 are zeroes. This A 16 to A 19 will be 0 for this combinations of the memory location 00000H to FFFFF H sorry this is 0, 0FF FFFH ok. Similarly, I am going to I mean a derive another signal using this P, so this P I will take as one input. So, what are the additional signals for the EPROM? EPROM the signals that you are going to connect to the microprocessor are A 0 to A 14 ok, A 15 to A 19 are available.

So, here out of this I have connected A 16 to A 19, so which will generates 0 here. Now, we will connect this A 19 as another input for this if I call this as C NAND gate okay. So, when does this C is equal to 0 if A 15 to A 19 if all are the ones what happens? If any one of this A 15 to A 19 is 0 otherwise I will generate this C separately instead of generating from P, I will generate separately in a similar manner.

(Refer Slide Time: 39:11)
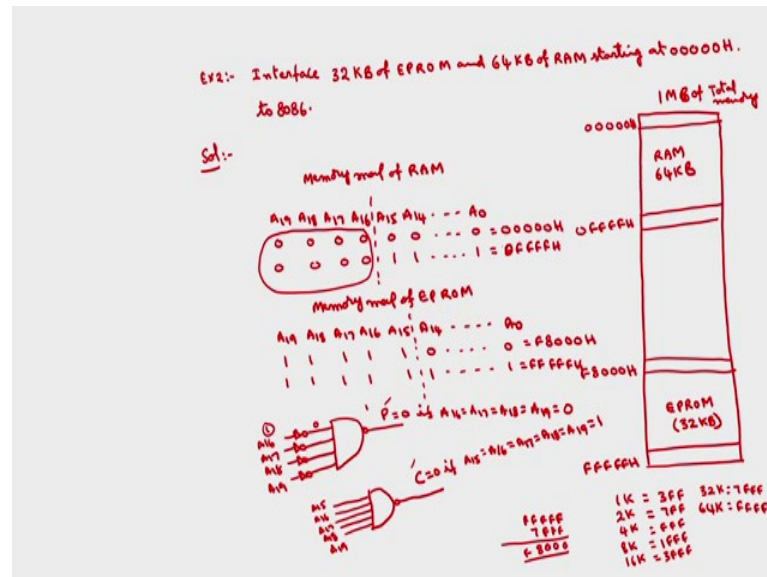
So, I will take a NAND gate, but I will not take the inverters now, I will call this one as C, the input are A 15 to A 19; A 15; A 16; A 17, A 18, A 19. So, C is equal to 0 if A 15 is equal to A 16 is equal to A 17 is equal to A 18 is equal to A 19 is equal to 1. If all are ones then only output is 0 if anyone of this A 15 to A 19 is 0 output is 1. So, I am using 2 signals P and C, using this P and C, I will select the memory banks; P is 0 for RAM locations all the RAM locations and C is 0 for EPROM locations okay.

Now, how to select this chip selects of this 4 ICs, chip select 1 bar, chip select 2 bar, chip select 3 bar, chip select 4 bar. So, in order to select this EPROM, so how to select this chip select 1 bar? So, for chip select 1 bar it should be ODD of RAM ok. So, P and BHE bar and A naught will decide about this CS 1 ok. So, if P is 0 P is 0 and A 0 is this can be 00.

So, what should be this CS 1 bar? CS 1 bar be CS 1 bar has to be selected in case of word selection from the RAM or only ODD bank ok. So, this P is equal to 0 by default this is for RAM, so no need to consider you can write here itself P is equal to 0. Or BHE 0 A 00 means chip select bar has to be selected because this is word from the RAM okay.

And when P is equal to 0 this is 0 1, what should be CS 1 bar? This is BHE bar high enable is 0 means high bank we are not going to select only low bank. So, this high enable is 0 means high bank which is ODD bank has to be selected this will be 0,

because BHE bar is 0 high bank means it is a ODD bank ODD bank has to be selected, so 0, 0 1 0, so this is EVEN bank.

So, this will not be selected this is 0 1 1 no bank will be selected, so this will be 1. So, only in these 2 cases only CS bar should be equal to 0. So, what is the Boolean expression for CS 1 bar? CS 1 bar is equal to P bar BHE double bar, A 0 bar plus P bar BHE double bar plus A 0 ok.

So, you have this A 0 is 1 in 1 case another 0 in another case. If I take this P bar and BHE double bar of course BHE double bar becomes BHE. If I take this as a common, we will get A 0 plus A 0 bar this is equal to simply P bar BHE bar. So, what will be chip select of this one? P and BHE signals you have to connect to this, so when both are zeros only output should be 0.

So, what should be the gate? AND gate; this is P and BHE bar you see here. So, in these two cases P is 0, BHE bar is 0 so output is 0, this chip will be selected. So, in these two cases, so this will not be selected because BHE bar is 1, P is 0, so output will be 0, so it will be selected, so you have to exclude this. So, for that you have to take NAND gate right, you have to invert here otherwise ok, now I think its correct.

So, if both are zeros then here these two will be 1, so output will be 1, so it will not be selected it should be NAND gate. So, this is both are 0 means 0 0 this will be 1, 1 1 it will be 0. If any one of the input is 0 then what happens here this will be 1 0, if one of the input is 0 output will be equal to 1 it will not be selected, so this will work out ok.

So, the circuitry will be. So, for P is equal to 0 BHE bar 0 for P is equal to 0 BHE bar 0 this will be always 0, 0 if both BHE bar and P bar is equal to 0 ok. Similarly, you can derive the chip select of EVEN bank. So, what will be chip select of EVEN bank? For that RAM only, so P is equal to 0, P should be 0, BHE bar A 0 bar CS 2 bar ok.

So, it should be 0 0 0 0 1 1 0 1 1. So, bus high enable is 0 means we should not select this chip, because this is only high bank ok. So, this is both are zeros you have to select, here you do not select, here you have to select, here not select. This is EVEN bank, so a 0 equal to 0 this you have to select, this you have to not select ok.

So, here BHE bar will get canceled, so you will get same in terms of P we can verify this that this output will be P and A naught. Similarly, here you can get NAND gate here will get NAND gate outputs. The inputs of this one will be C, C BHE bar and here this will be C A naught in the similar manner you can check for this, so like if you want CS 3 bar.

So, the inputs will be C then BHE bar A 0, so C should be 0 for EPROM. If this is 0 0 what means both the banks has to be selected chip select should be 0, so that it will be selected. 0 1 means is high bank which is ODD it this has to be selected, 1 0 means this is only EVEN bank, so ODD banks should not be selected, 1 1 means no bank will be selected.

So, if I take the Boolean expression here A 0 varies from 0 to one here which will get canceled. So, only CN BHE bar will be there and you have to connect through NAND gate. Similarly, for CS 4 bar you can write C BHE bar A 0 CS 4 bar this should be 0 for EPROM, 0 0 means both the banks. So, this has to be selected, 0 1 means this is high bank only ODD bank.

So, EVEN bank will not be selected means 1. 1 0 means only lower bank which is EVEN banks, so this has to be selected, 1 1 means no bank is selected ok. So, even if we write the Boolean expression, we will get NAND operation of C and A naught. So, this is about the complete memory map of this particular problem ok. So, this is how we can interface the memory to the microprocessor ok. Now, we will discuss the I/O mapping with the microprocessor.

So, in order to map most of the IOs, so we will we will require some extra peripheral pins called there are some like 8255 is programmable peripheral interface. This is required to connect most of the input output devices. And that there are some special ICs to connect keyboard and to connect this printer and all we have DMA, we have different ICs. So, the but most of the useful IC for connecting the IO is 8255, this 8255 I will discuss in the next class.

Thank you.