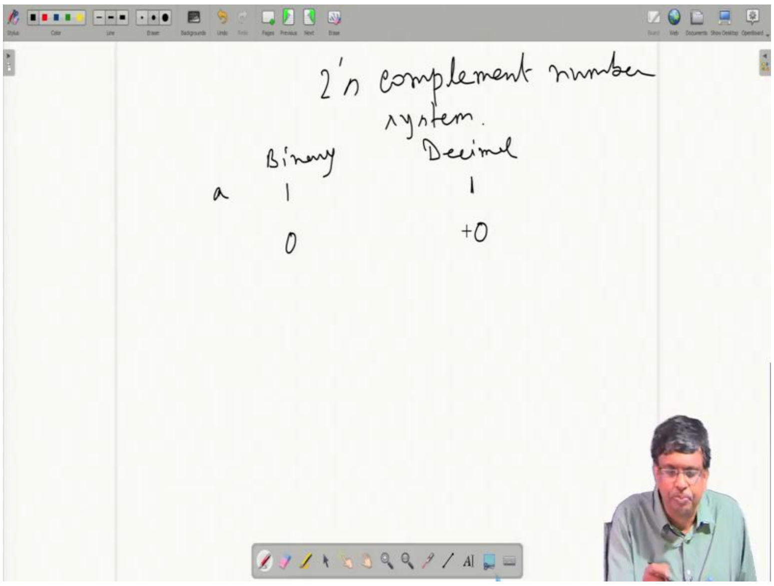
VLSI Signal Processing

Professor Mrityunjoy Chakraborty
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture 33

2’s Complement Number System

(Refer Slide Time: 0:40)



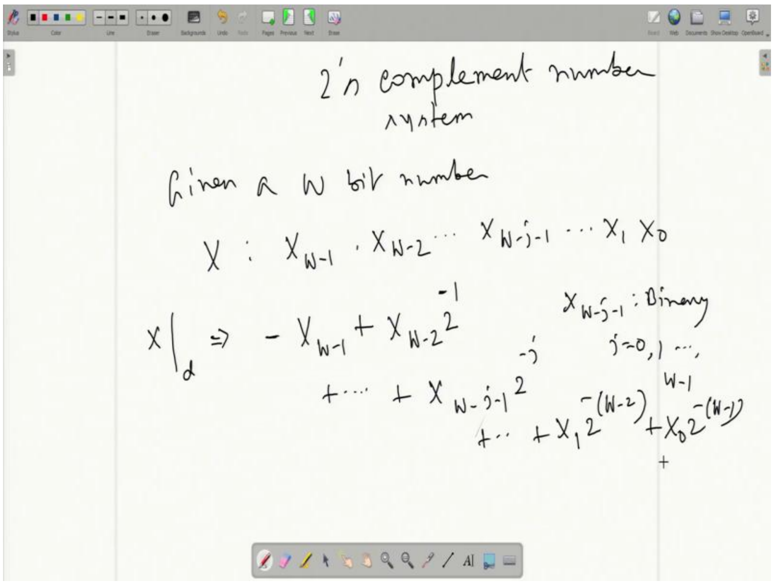
Okay, so in the last class, we discussed we just started the topic of 2’s complement number system. So, it might be a good idea to again start from there. Okay, as I told you in the last class, 2’s complement number system is not just 2’s complementation. You are all familiar with 2’s complementation, that is complement H-bit of a binary word and add a 1 at the LSB, then you get 2’s complement, which you can use in some context the negative of that number, okay. But that is different. 2’s complement number system is system.

Okay, now before get into this, let me, you know, set certain things in the beginning. Suppose, A is there. A is a variable and is a binary variable. In binary domain, it can take binary 1 or binary 0. For, in it is in binary domain, then it follows binary arithmetic, that is 1 plus 0 is 1, 0 plus 0 is 0, 1 plus 1 is 0 it carry 1 and all that. In binary domain, we have only these operations. We cannot have minus 1 or 1 binary 1 into some decimal 5. This has no meaning.

So, what do we do in the binary domain? We take A as binary 1 or binary 0 and follow normal binary operations, as I told you just now, complementation, addition, etc. But same A, when I take it in the decimal domain, then if A is binary 1 here, I will take it as decimal 1 and if it is binary 0, I will take it as decimal 0. In decimal domain, if I am evaluating something, there I can use decimal operations. For instance, 1 into 5, I into 6, because they are decimal numbers. Minus 1 and things like that, okay.

So, depending on which domain I am in, A will be treated as a binary variable, take binary value 1 0 or decimal variable again it is similar, 1 0. But here I can use decimal arithmetic operations. Here I can use only binary arithmetic operations, okay on the same A. We will have to make these distinctions in all what we do because in all the system we discussed, the same variable A will be in originally binary variable but we will also take it in the decimal domain that time it will viewed like this, okay.

(Refer Slide Time: 3:00)



We have given a W, X this is most significant bit, X W minus 1, then X W minus 2, dot, dot, dot. X W minus j minus 1 dot, dot, dot X1 X0. And there is a binary point. The binary point is not hardware. It is in our mind, I mean if you want you may not put it here, you may put at the end. Then it will be an integer. But normally we prefer fractional numbers. We put it here, its actual meaning will come now. But it is again in your mind. It is just a point of reference and nothing else.

Okay, now such and here, every variable whether X W minus 1 or X W minus 2 or these or X1 or X0. Every variable is a binary variable. That is every X W minus j minus 1; binary. j can be 0, then you are here. j can be 1, so W minus 2, you are here dot, dot, dot if j is W minus 1, you are here, total W. But for all of them these variables is in, is binary, okay.

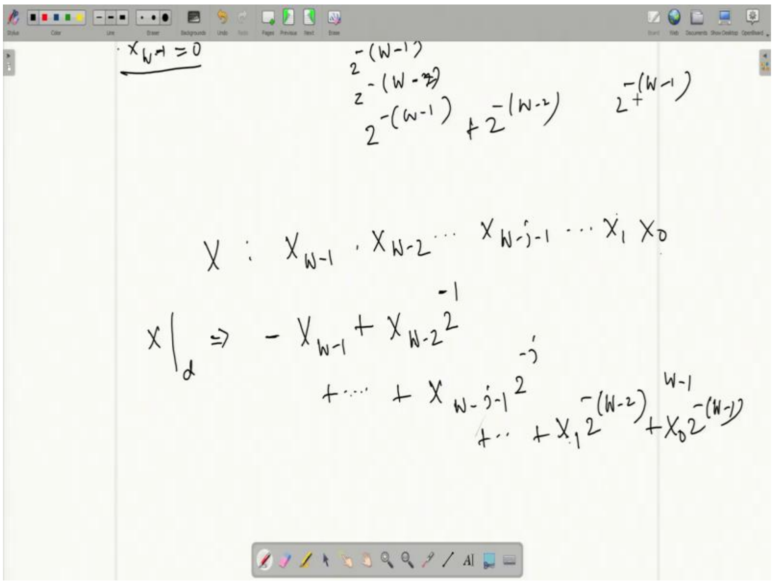
So, either binary 0 or binary 1 and when I am writing the word, in this word, they will follow the binary arithmetic because they all will be treated as a binary variable. So, far so good. But under 2's complement system, if I give you this word, W-bit word, what is its decimal value? That is what defines the 2's complement number system.

Under 2's complement number system; it has a decimal value this d for decimal. So, X has a decimal value and this time I will write an expression, which will be a decimal expression. So, I will use the same variables in this expression. But now, they will be viewed as decimal variables. Not binary variables. They will still take 1 or 0, but I will take them as by decimal 1, decimal 0 in this expression and then I am going to write.

And then, since it is a decimal expression, I can (())(0:05:08) use minus sign, you know, into 2, into 4 like that. So, this decimal expression will be minus X W minus 1. So now, X W minus 1, here it was a binary variable, Binary 1 or binary 0. If it is binary 1, in this expression, it will be decimal 1. These are confidential. If it is binary 0 here, it will be treated as decimal 0 and now minus decimal 1, minus decimal 0 is okay. Decimal arithmetic permits minus sign.

Then next one is this now plus will start coming. This into first negative power, 2 inverse then dot, dot, dot. This will be 2 to the power minus j you can see easily, it will be 2 to the power minus j because if it is W minus 1, there is no 2 to the power, there is 2 to the power of 0. Then if it is 2, W minus 2. It is 2 to the power minus 1. If it is 2, 1 coming here. So, if it is j plus 1, it will be in the bracket, it will be j here. Going that way and last one is this. All right, okay.

(Refer Slide Time: 6:36)

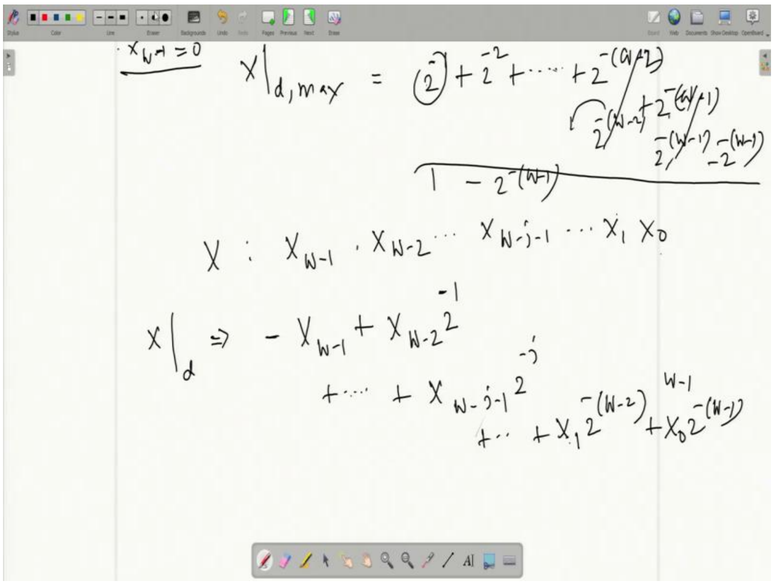


Now, let us see, these are decimal expression. Let us try to understand its meaning. Suppose, this fellow is 0, MSB is 0. If it is 0, then obviously it is a positive number because minus 0 means 0 and all other fellows are positive, so it is a positive number, okay. In that case, minimum will be when this is 0, this is 0, this is 0, this is all are 0 only X0 is 1. So, it will start at this value, 2 to the power while all are 0, only X0 is 1.

Then next will be X1 1, X0 0, all right. So, that time it will be 1 here, so 2 to the power, here that will be 1 1, so it will be summation of the two, so on and so forth. So, you can see that that at every step as I go on increasing from all 0 then 1 here, then1 0, then 1 1, number value is increasing by this step, 2 to the power minus 1. What is this after all? It is twice this. Okay, then if it is 1 1. If it is 1, 1 means it will be, so it is thrice these, okay, so like that.

So, if it is all 0 but 1 here, it will be just this much. If it all 0 but 1 and 0 here, it will be twice this. If it is all 0 but 1 1 here, it will be thrice this. So these are basic units by it keeps increasing.

(Refer Slide Time: 8:31)

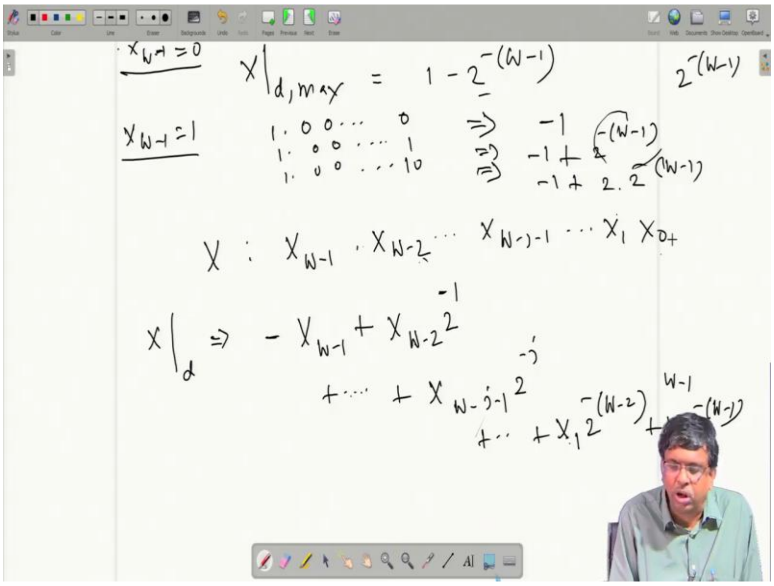


So finally, it will increase maximally up to 1. When MSB that is most significant bit is 0 but all the other fellows are 1 that is the highest positive number. So, highest positive number will be, as per this formula, minus 0 it is 0, then 1 into 2 inverse. That means Xd its max, will be equal to 2 inverse, then 2 inverse 2 dot, dot, dot, dot, 2 inverse W minus 1 sorry, W minus 2 plus 2 inverse W minus 1.

In the last class, I have shown since, it is basically a GP series. You can take out 2 inverse out. So it will be 1 plus 2 inverse plus 2 inverse plus dot, dot, dot, dot. So it is the GP series, you can take the sum of a GP series, okay. Alternatively, the way I do, I add this like this, you know, this I add here and subtract the same thing.

So, if we add them these two, if you add, it is twice of this, so you will get here and if we add this, you move here and like that so and so forth. So finally, twice of this will come, which is 1 and you are left with this minus okay. This, why we are doing? But you can carry out a GP sum, a GP series sum and we will get the same thing. So this is what you have.

(Refer Slide Time: 10:08)



Okay, you are starting at 2 to the power minus W minus 1, then twice that, thrice that, four times that, finally, you are here. These are maximum you can get. If W is large, then 2 to the power minus bracket W minus 1 will be very small. So, it will be approximately equal to 1, but theoretically it is less than or equal to 1, less than 1, always. So it will not exceed 1 under this system. If I imagine a binary point here, okay.

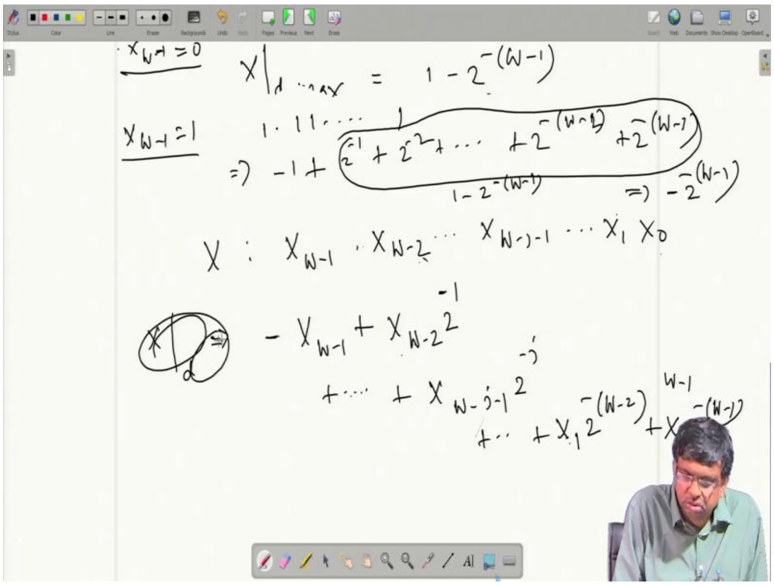
So it will be less than 1, which means, when it is positive at least, it is a fraction. It is less than 1. Between 0 to 1. Very good. Now, we go for the other side. If this fellow is 1. Earlier I took the MSB, this to be 0 and I found the number to be having positive decimal value and we started going up from 2 to the power minus W minus 1, then twice that, thrice that, four times that dot, dot, dot up to this. This comes when all the fellows are 1 and the MSB is 0. This part we have covered.

Now suppose, MSB is 1. Suppose is 1, that I have minus 1 here. Minus 1 if now all the other fellows are 0s, then from all the positives there is no contribution, because 0 here , 0 here, 0 here. So we have got maximally negative value. So if it is 1.00000, it will have a binary value, decimal value equal to minus 1. Minus 1 plus 0 times something, 0 times something, so as 0.

Then if you go up slowly, 1.00 dot, dot, dot, dot 1 that means, it will be minus 1 plus only this last fellow, because 1 is here, others are 0, so this. So from minus 1 we go up in upward direction by this much. If it is 1.00 dot, dot, dot, dot 10, it will be minus 1 plus 1 here and 0 here. So it is twice 2 times this basic quantity and so on and so forth.

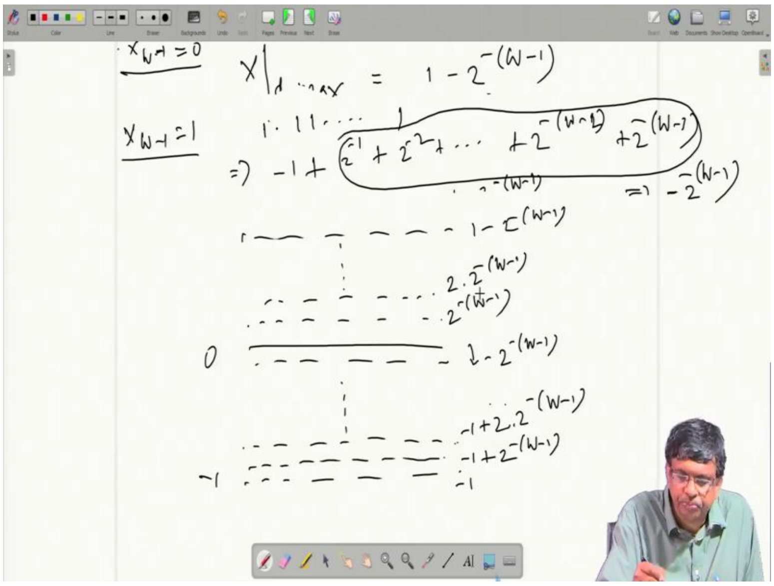
So from minus 1, it starts going up. It steps up 2 to the power minus W minus 1. Okay, as I go along, you know, 1.000 maximally negative, then 0001, slightly less, negative. 00010 further less negative and so on and so forth. It will be 1.000011, so I am going up. It steps up 2 to the power this. It steps up this. 1 times this, 2 times this, 3 times this like that. Finally, a situation will come (())(0:12:51) 1.11111.

(Refer Slide Time: 13:11)



That is we will have here decimal value will be minus 1 and then 1 into 2 inverse then 2 inverse 2 and dot, dot, dot, all are 1, so 2 to the power minus 2 minus 1. But these some I have already worked on, a while ago. This is GP series. Its value is 1 minus 2 to the power minus W minus 1. So, together it will be minus 1 plus 1 this cancel, minus this.

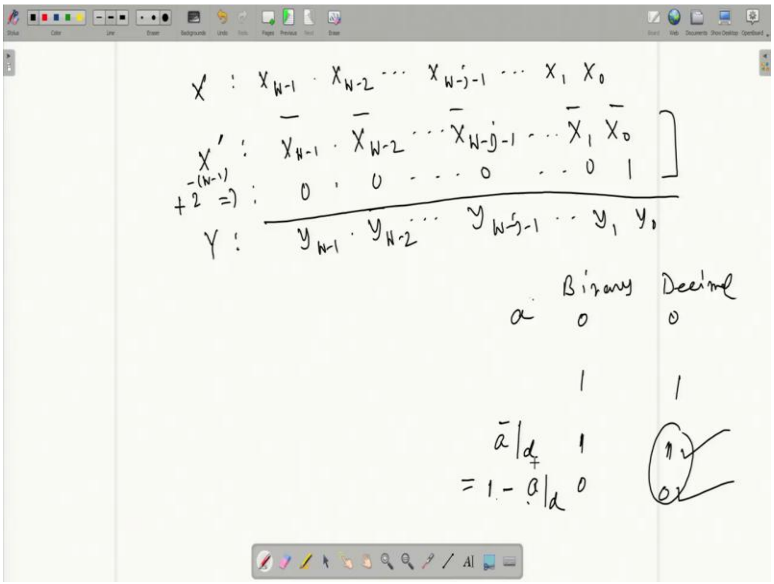
(Refer Slide Time: 13:55)



Which means if it is decimal 0 and it is minus 0 and minus 1, this is minus 1 then you go up. This is minus 1, this is minus 1 plus 2 to the power minus W minus 1, then this is minus 1 plus twice 2 to the power minus W minus 1 and dot, dot, dot. Finally we are here, so this much is minus then if you have next number at 0 point all 0, so you are here. Then 0 point all 0 and 1, then you go to the positive side. It will be just 2 to the power then 0 point all 0, 1 0, it will be twice this, dot, dot, dot, dot.

Finally when it is 0.1111, all 1. You will have the last one, 1 minus this. Now, this is how it goes. These are flow of the 2’s complement number systems. This is the definition that under 2’s complement number system, if I give you a binary word, it has a specific decimal value given by that formula, okay.

(Refer Slide Time: 15:16)



Now, there is an interesting property, which we should discuss now. Suppose, you have given this binary word. This is a binary word, W-bit binary word given to me. We know its decimal value. Suppose, I play a game. What I do is, I form a number binary number, maybe X prime by taking complement of each. If it is binary 1, I take it binary 0. If it is binary 0, I take it binary 1. So, complement. Here also, complement.

So, I just take complement of each bit and then I add this number, binary number 0.000000, just 1. That is actually decimal it is 2 to the power minus this and the resulting number, if we will called it Y, it will be Y W minus 1 dot Y W minus 2 dot, dot, dot Y, if we are adding the two. Question is what is this Y? That is the thing we will evaluate.

But before that, let us examine some basic thing. Suppose there is a binary variable A. binary domain and decimal domain. It can either be 0 or 1. Then A bar if A is 0 then I know what will be A bar. A is 0 1 and here it is 0 1, no problem. If A is 0 in binary it will be treated as decimal bit 0 or in decimal domain, we do not call it bit anymore because in decimal case, even if only single point, it can have more than two possibilities, 0, 1, 2, 3 up to 9. So whenever you have more than two possibilities, we no longer call it bit, we call it a digit.

So it is a digit. But digit value remains same. If it is binary 0, it is decimal 0. If it is binary 1, decimal 1 that is for A. But if it is A bar if A is 0, A bar is 1, if A is 1, it is 0 and obviously, decimal also it will be 1 because if the binary fellow has binary value 1when I am treating it as a decimal value, its weight will not change, it will be 1 it will be 0.

But you see , I can my claim this I can write A bar as A bar in decimal if you want, you know, more clarity I can write A bar in decimal that is A bar here. A bar in decimal is 1 decimal 1 minus A by (())(18:35) d. That is A, A bar see this value, when A is 0 binary, then in decimal it is also 0. So if you put that then 1 minus 0 is 1.

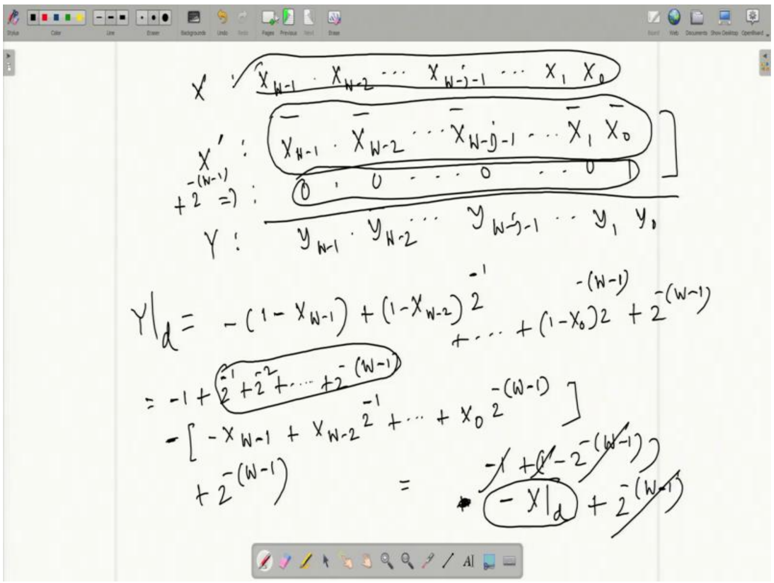
So if A is decimal of 0, A bar is decimal 1 1 minus 0. If A is binary1, in decimal it is decimal 1. If it is decimal 1, then what happens to A bar in decimal? A bar in decimal will be 1 minus by this formula, 1 minus 1 and that is 0 and see you are getting 0. So, this is verified. If A is decimal binary 0 and therefore decimal 0. A bar is 1 minus 0, which is 1 so satisfied. If A is binary 1 and therefore decimal 1 then A bar, by this formula, is 1 minus 1, if A is decimal is 1, so 1 minus 1, which is decimal 0, satisfied.

So, in decimal domain, any A bar will be 1 minus A in decimal domain. That is 1 minus A, A was originally a binary variable but when I write it an expression 1 minus A you do not have to always write A by d to indicate that it is a decimal. You can simply write 1 minus A because we are using minus sign, it is automatically implied. You are dealing with A as a decimal number.

So, if it is a binary 1, it will be decimal 1. If it is binary 0, decimal 0 because its value will not change, but domain changes, domain is decimal, I can put a minus, I can do all these things. And now it is very clear, if A was originally binary 0 and therefore decimal 0, by this formula, 1 minus decimal 0 is 1okay, which is decimal 1, and A bar should be decimal 1 so this is matching.

Similarly, if A was binary 1 therefore decimal 1, by this formula, 1 minus 1 is 0. So A bar by this formula is 0 in decimal domain and that is matching. A bar should be 0 and I am getting 0. So that means any complement will be 1 minus original in decimal domain.

(Refer Slide Time: 21:02)



See, if I add the two, it will be decimal value of this and decimal value of this, okay. Decimal value of this is minus 0 because it is a 2’s complement system, so minus 0 plus 0 into 2 to the power minus 1 plus 0 into 2 to the power minus 2 dot, dot, dot, dot, dot. Nothing. Only this 1. So 1 into 2 to the power minus W minus 1, okay.

And here so I have got let me write this term. This is coming from here and from here, under 2’s complement, it will be minus of this bit. Okay but this will be 1 like I told you A bar is 1 minus A, so in decimal domain, X W minus 1 bar that means, if it is binary 0 and therefore decimal 0 this with bar it will be 1 and vice versa. So that is nothing but 1 minus decimal X W minus 1. Okay, that is this bit, minus because in the 2’s complement formula I mean, decimal value of a 2’s complement number means this has to be treated as a decimal digit with a minus sign. So minus sign, decimal digit.

The moment it is decimal, it is 1 minus decimal of this. That is what is coming. Then plus this into 2 to the power minus 1 but this again to be treated as a decimal digit. The moment it is decimal digit means 1 minus this original in decimal form. Sorry, this one. All right, these are total value. This Y decimal.

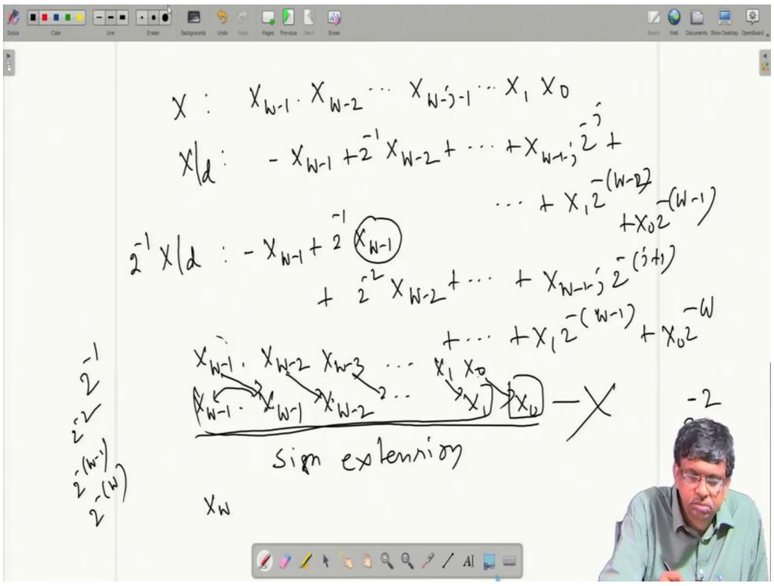
Now, let us try to find out what this is. If you take out the minus 1 and then plus 1 into 2 inverse plus 1 into 2 to the power this under one bracket. So, it will be minus 1 plus 2 inverse, 2 inverse 2 dot, dot, dot, dot 2 inverse W minus 1. And then, minus minus X W minus 1 I am taking the minus out, so this minus is gone out, so it will be plus now, okay.

And lastly this 2 to the power now, how much is this? This summation we have worked out. 1 minus 2 to the power this so this is minus 1 plus 1 minus 2 to the power minus W minus 1. This part is for this. Then you can see this sorry, not minus sign. What is this bracketed quantity? It is nothing but the decimal value of the original number X, this one that is minus of X W minus 1 and all those things, okay.

And lastly, we are left with this single quantity, this. So you can see, this 1 and minus 1 cancels. These two cancels. We are left with this much, which means if you really follow the procedure then you complement each bit of X at a 1 at the LSB and get a new word. The decimal value of that word, under this 2’s complement system will be nothing but negative of the original decimal value, minus X by d. this is the advantage of 2’s complement number system that is if you want to take integration, if you want to do subtraction, it do not need to be a subtractor, you take complement of each bit and 1 at the LSB, the real thing what we have decimal value equal to minus of the original decimal value. So you just simply add with whatever number that will amount to subtraction.

Okay, these are main advantage of 2's complement number system. There are some other advantages also I mean fundamental advantages, which I will tell later in the next class, next section, okay.

(Refer Slide Time: 26:08)



There is one more property. This also very very useful in practice. Suppose X is given to you and it is 2’s complement number. So I know what is X by d. Suppose, this decimal value, I want to divide by 2, then I have to multiply by 2 inverse. This decimal value, I want to divide by 2 or equivalently multiply 2 inverse.

Right hand side, if I multiply 2 inverse it will be minus half first, minus half times this. Minus half I can write as minus 1 plus half, it will be minus half times this and minus half is nothing but minus of 1 then plus half. If we do that, so minus 1 times this will be this and plus half that is 2 inverse 2 inverse times. And then 2 inverse times this, it will become 2 inverse 2 times dot, dot, dot. This will be 2 inverse j plus 1 dot, dot, dot, this will be X1 2 to the power minus and then X0.

Now, if you look at this, this is the decimal value of what under 2's complement system of this word, MSB, so that if the decimal domain, when I take the decimal value minus of this comes which is present. Next bit is what is associated with the power 2 inverse. Next, but that is this.

So, next will be the one associated with 2 inverse 2 that is this. So like that. Finally, X1, X1 is coming with 2 to the power minus W minus 1 and then X0, 2 to the power minus W and then X0 here. So size has gone up. Because this has got repeated, so earlier size was W, now it is W plus 1. These are what whose decimal value is this under 2's complement system. What is this word?

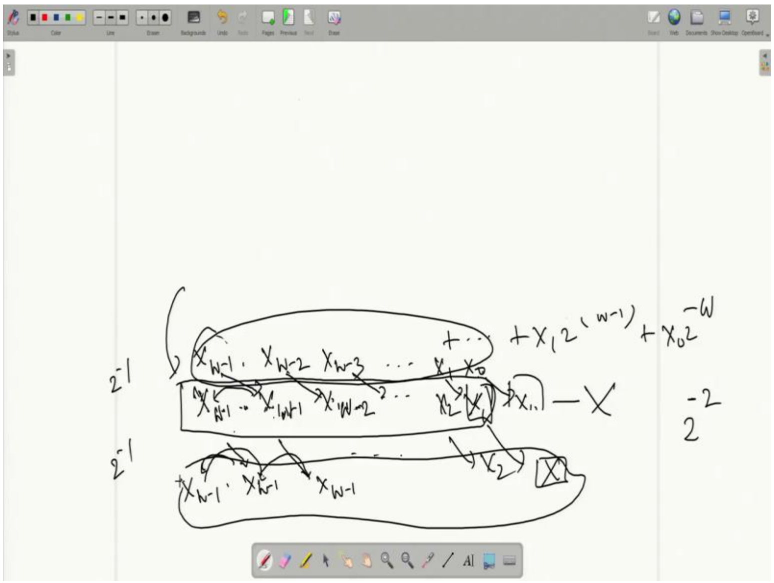
Original word was, these are the original words, W-bit word. What has happened? This got shifted to right by 1, this got shifted to the right by 1, right by 1, this got shifted to the right by 1, this got shifted to the right by 1 and this is what is called sign bit. What is a sign bit? I forgot to mention.

If this bit is 0, I know the number is positive and we have also seen that if this bit is 1, whatever value this bit take, now total number is negative, it is below 0. That is why it is an indicator of whether the number is positive or negative, if it is 0, it is positive, the whole number is positive. If it is 1, whole number is negative that is why is called sign bit.

So, sign bit also comes here sign bit comes and other bits come here and the same thing gets repeated here it is called sign bit extension, sign extension. And it is W plus 1 bit. Now, most of it, we have a finite size resistor. Resistor may be W-bit. So in that case, this part will be trunked out. It will just go out because we will take the higher ones. So this is what we will get, all right.

So, if I multiply by 2 inverse, every bit get shifted to right by 1. LSB, the least significant bit gets out and then sign bit comes back here as an extension. If I now multiply it by another 2 inverse, there is overall 2 inverse 2. Then it will be, this will come, as it is sorry, so this is a word. So this will get shifted.

(Refer Slide Time: 31:00)



So X W minus 1 let me do it here only, this was the word, when I multiply it by 2 inverse. Another 2 inverse, again same thing, this will be shifted, this will come here X W minus 1. Next X W minus 1 that will come to this position and dot, dot, dot. X1 will go out and X2 will come here, and this will get extended here.

So, you see sign bit actually gets extended twice. If you compare with this word, original one and this one, sign bit got shifted twice. All other bits got shifted twice and the sign bit got extended once, extended next time twice. So, if it is 2 inverse 2, all bits get shifted to the right by twice, you got two bits get tucked out and the sign bit then get extended once and one more time, so on and so forth.

This is called sign extension and in we have to implement this in digital hardware when you carry out, you know, multiplications and other circuits, okay. Thank you very much. I will move to the next section. Thank you.