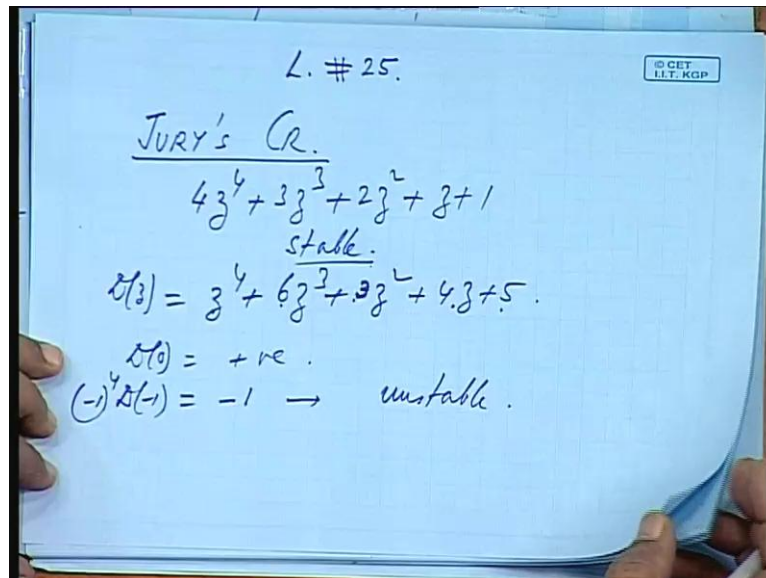


Digital Signal Processing
Prof. T. K. Basu
Department of Electrical Engineering
Indian Institute of Technology, Kharagpur

Lecture - 25
Effects of Quantization

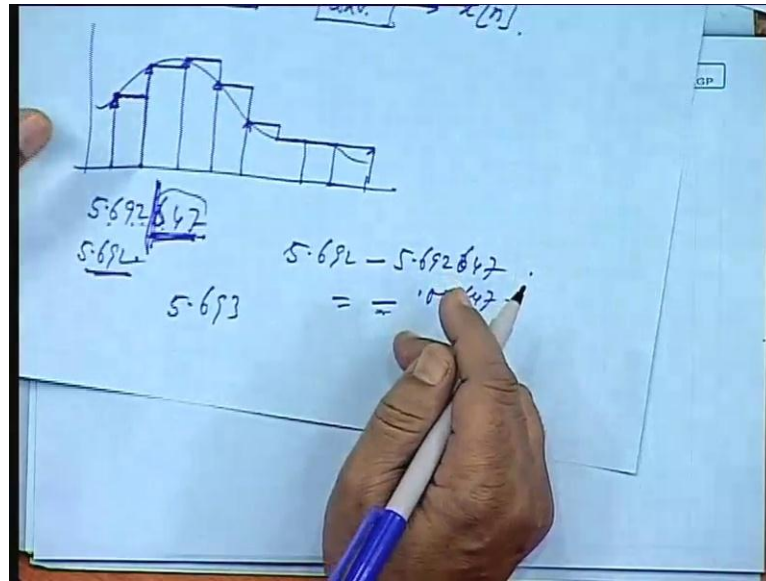
(Refer Slide Time: 00:47)



In the last class, we were discussing about Jury's criteria for stability. We saw with one example $4z^4 + 3z^3 + 2z^2 + z + 1$, we arrange the rows and we found that the system is stable. Let us take another example, this was meeting all the criteria. Let us have the denominator polynomial as $z^4 + 6z^3 + 3z^2 + 4z + 5$ is the system stable, first you apply the condition $D(0)$ is $1 + 6 + 3 + 4 + 5$, so it is positive.

What about $D(-1)$ $1 - 6 + 3 - 4 + 5$ that is $1 - 6 + 3 - 4 + 5$ plus 3 plus 1, so that gives me minus 1, so this condition is evaluated you need not go for further trial, you do not have to make the rows. If in the first two test you find the conditions are not made, then it is unstable will take up a few problems in the tutorial class and see how to apply in jury's criteria for stability. Today, we shall be considering the Effect of Quantization.

(Refer Slide Time: 02:43)



Before we go for quantization, I will just give you the basic idea about t to a conversion, you have x a t that is an analog signal, then you have sample and hold, then actually A to D conversion takes place, so you get the converted value x_n . Now, the analog signal may be like this, if you just sample it, it is not an instantaneous process you sample it for then this sample value will be represented by binary digits by any particular code.

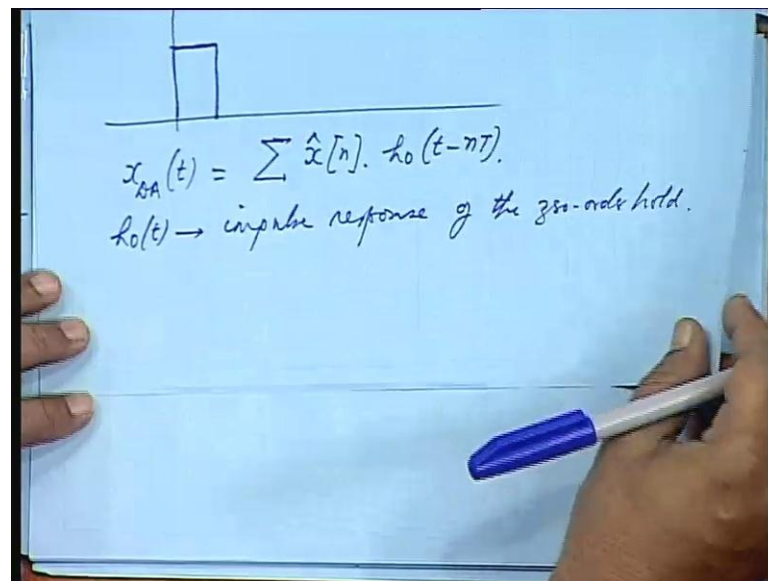
And then till that particular value is approximately closed that sample value it will not stopped, so it will require continuously a fixed input, if the input keeps on changing then it will be disturbed. So, this level has to be maintained for a period t , so this is the sampling period and within this time the computation has to be completed, the conversion from that exact value here to the approximate value that binary digit has to be computed within this time.

So, that is how the sampler time is fixed and then the next value comes, so it will stay there, then this value will stay here, then this value will stay here, this value will stay here and so on. So, there will be some errors of course, we are quantizing the quantity say 5.692347 , now if I in a decimal system, if I quantize it there are two types of reducing this one is suppose I have three decimal place approximation, then I will just truncate it.

So, it will be 5.692 , so the error is always $0.34, 0.0003447$, if I have say 647 , then also if I truncate it then this would be the error, so truncated value is always less than the actual

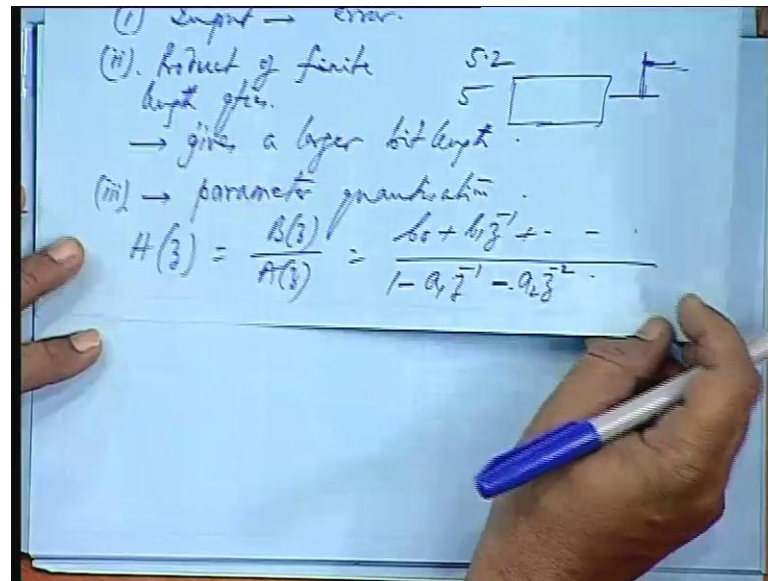
value, it can be at the most equal. So, the error is always positive, error is one side it actually error is actual value to be subtracted from the final result, so it is 5.692 minus 5.692647, so error is minus 0.000647, so it is always in one side. But, if you round it off, then it will be a 5.693, so it can be higher, so round of error can be both positive and negative, whereas truncation error will be one side it will be always less, the error will be always negative.

(Refer Slide Time: 06:40)



So, this hold circuit is having a value 1 for this period and 0 otherwise, so the whole circuit output is basically like this, one may write x_D to A converted value as $x[n] h_0(t - nT)$, so $h_0(t)$ is the impulse response of the hold circuit, of the 0 order hold.

(Refer Slide Time: 08:12)



Now, shall take up a little bit of simple mathematics that you all know. How to represent the numbers in binary from by different conventions. So, DSP algorithm will be use for special purpose DSP chips or the programs that will be quantizing either the signal or the parameters. Now, there are three places where this quantization is coming, so error will be generated at three points, one is input, input first of all you have to scale it to that particular level and then the input has to be quantized.

So, the input error, now input is a signal which really does not alter the system property, it will give you some output say instead of 5.2, suppose you are giving an input 5, so had everything else been ideal, then the output is just proportionately reduced. So, input error is going to give you error in the output, but it is not going to affect the property of the system, then or the quality of the output, then you have product you are having multiplication inside.

So, you are having a finite bit length for the representation of any quantity say 8 bit, so an 8 bit number is multiplied by another 8 bit number, it may go to 16 bit, then you are truncating it, so this is going to cause more of errors. Product of finite length quantities or variables, that is you can multiply by a multiplier or just two quantities, two variables may be multiplied, finite length quantities.

So, the product will have this gives a larger bit length, see you have to truncate it the third one, which is the most important one is the parameter quantization. Parameter

quantization, you can see if we change the parameter the roots may change. In z we say B by A z , we may write b_0 plus $b_1 z^{-1}$ and so on, $1 - a_1 z^{-1} - a_2 z^{-2}$ and so on.

Normally, we put a negative sign because it is an I mean this coefficients will come to this and if they are transferred you can write plus and minus it really does not matter, many books they follow this as a standard form. Any way these b_0 , b_1 , a_1 , a_2 , etcetera; these are the constants which if you quantize will be changing the distribution of roots, you must have studied in control systems root locus technique. So, they are the last coefficient.

Again if you keep on changing the roots keep on changing, now here you are having multiple changes, changes in all the coefficients, so you can imagine all the roots will be changing in a very complex way and sometimes they may go out of the unit circle. So, that may reduce the stability of the system, it is not only going to the unstable is one, even if they change their positions the response is drastically affected that will see very soon.

(Refer Slide Time: 12:58)

radix $\rightarrow r$

$$25.564 = \sum C r^i$$

$$= 2 \times 10^1 + 5 \times 10^0 + 5 \times 10^{-1} + 6 \times 10^{-2} + 4 \times 10^{-3}$$

for $r=2$

$$10.00 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} = 6.25$$

So, round off noise which may be positive or negative, they may cause limit cycle oscillation and noise, random noise; now representation of numbers you may write a number in this form, where C is the coefficient associated with a particular radix r . So, in a decimal system we have r is equal to 10 in a binary system, we have r is equal to 2, so

25.564, you write as $C_i 10^i$ to the power i , so that can be written as $2^i 10^i$ to the power i plus $0 \cdot 10^0$ plus $5 \cdot 10^{-1}$ plus $6 \cdot 10^{-2}$ plus $4 \cdot 10^{-3}$. If r is equal to 2, you write 110.010 that represents $1 \cdot 2^2$ plus $1 \cdot 2^1$ plus $0 \cdot 2^0$ plus $0 \cdot 2^{-1}$ plus $1 \cdot 2^{-2}$ plus $0 \cdot 2^{-3}$. So, if you work it 2^4 plus to $6 \cdot 0$ this is 146.25 .

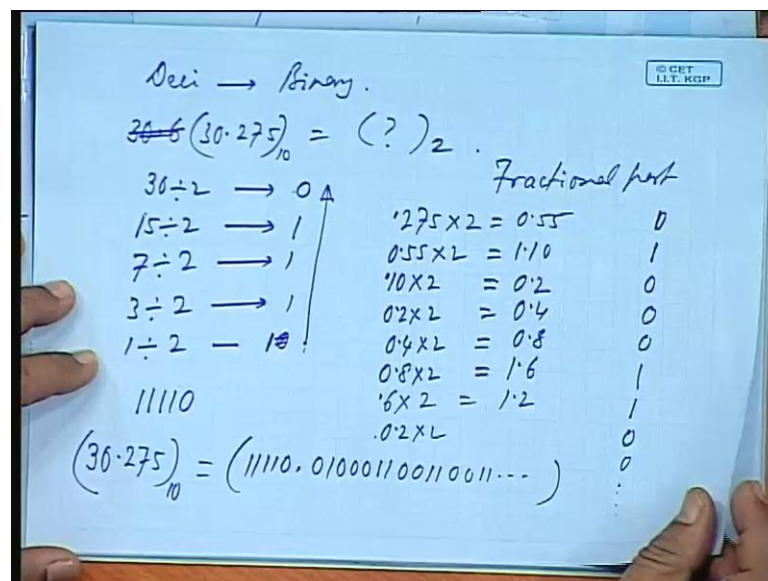
Student: ((Refer Time: 15:27))

0, which one $5 \cdot 10^{-1}$

Student: ((Refer Time: 15:36))

0 into, not $20 \cdot 5 \cdot 10^0$, thank you very much, I was writing something I was writing 20 in my mind, so this is 6.25 .

(Refer Slide Time: 16:03)



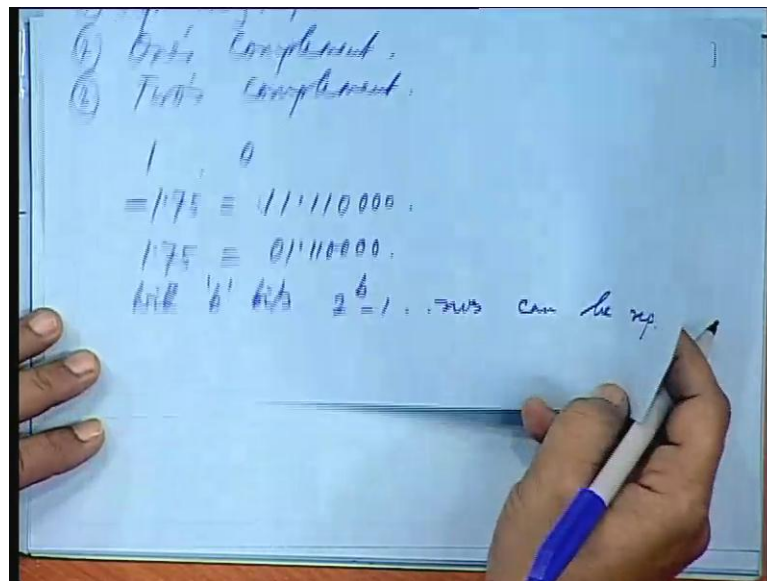
So, for conversion from decimal to binary we have say floating point arrangement like this 30.275 how much is it, in the binary system, so what we do this part we divide by 2 and take the remainder 0. Then, that gives me 15 divided by 2 the remainder is 1, the quotient is 7, 7 divided by 2 the remainder is 1, then 3 divided by 2 remainder is 1, 1 divided by 2 remainder is 0, so we have 1 and quotient is 0.

So, we take it from this way will be 11110 and the fractional part we keep on multiplying by 2 and starts removing the integral part, so 0.275 multiplied by 2. And we go downward is equal to 0.5, so this is 0, 0.55, 0.55 multiplied by 2, so that is 1.10, so that is 0.10 into 2 how much 0.2 0, 0.2 into 2 0.4 that is 0, 0.4 into 2, 0.8 into 2 is 1.6, so 1. And then 0.6 into 2 is 1.2 is 1, again 0.2, so if it is 0.2 you are coming to this same place, so it will continue again 0 0 like that 1, 1, and so on. So, 30.275 will be 11110 point will 0 1, go downward 00011001100 will it be two zeros are three zeros check.

Student: Two zeros.

Two zeros 1100, so it is like a your recording decimal, so it is not only in decimal system also in binary system, you can have a number which may not end it may be endless.

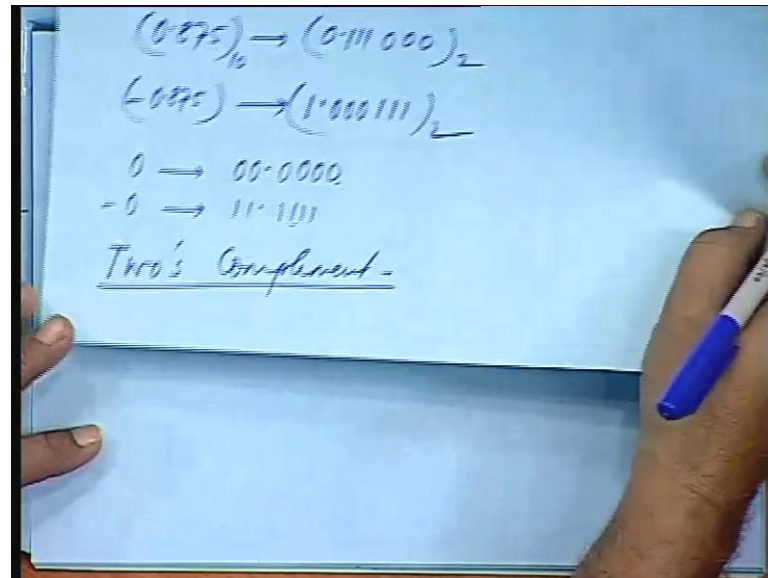
(Refer Slide Time: 19:39)



So, there are three types of representation in the binary system, one is sign magnitude, one is one's complement and the third one is two's complement, some of you have studied this in digital electronics. So, the benefit of those who have not gone through this course, I will just briefly go through this will not go in to the details of this representation. So, the negative number is always given by a 1 and the positive number were 0 in the MSD more significant date, so say minus 1.75 will be 11.110000, 1.75 is 0 1, so this 1 is for the sign this 0 is for positive sign 0.110000, 0.75 is written like this.

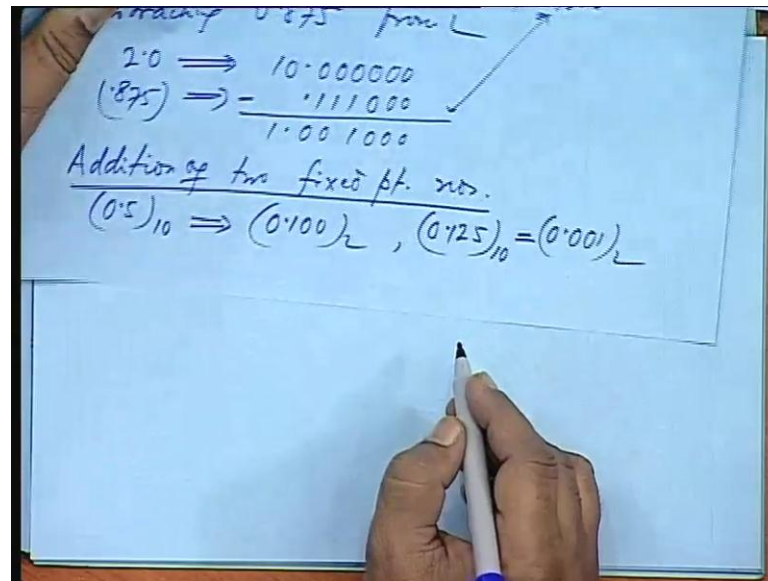
So, this is half to the power 1 half to the powers 2 and so on, so that gives me this, so with b bits you have 2 to the power b minus 1 number of numbers can be represented, so this is a very simple form.

(Refer Slide Time: 21:54)



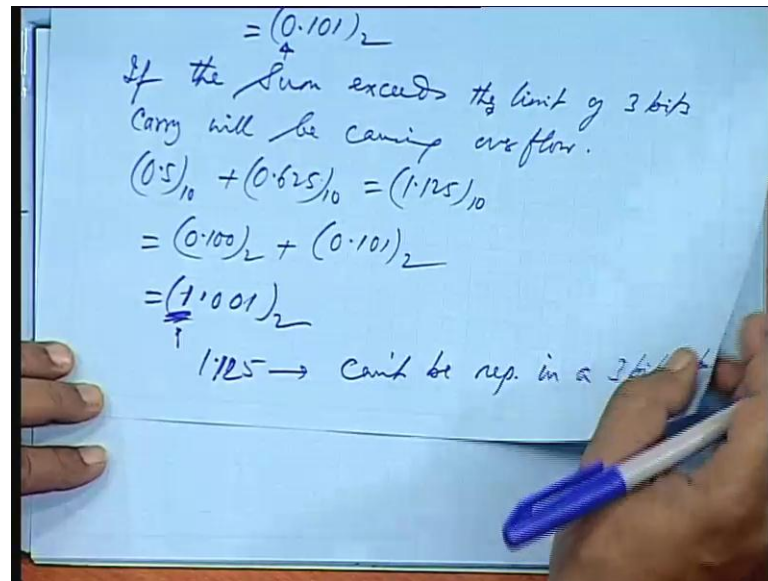
In 1's complement form for positive numbers it is same, for positive numbers it is same as the earlier one that is sign magnitude form, for negative numbers say 0.875, if you write it will be 0.111000 in the binary form, so for minus 0.875 you just complement each bit, so it will be 1.000111. Then, 0 can be represented in this 1's complement form as 00.0000 say we have 6 bit, I can write also minus 0 that will be 11.1111. So, this and this both will be representing 0's, in the 1's complement form, in two's complement form, let me write on this side.

(Refer Slide Time: 23:50)



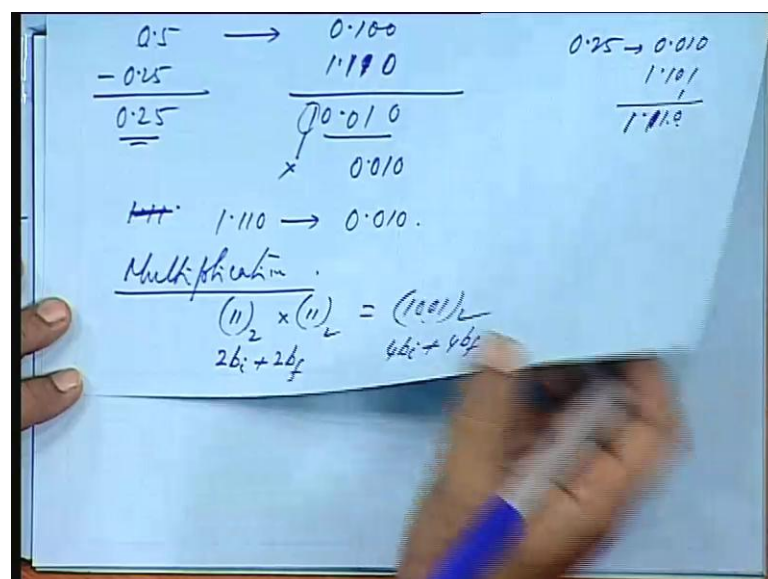
Say the same number 875 is positive number is once again same, so minus 0.875 you just take the complement and then add the 1 at the last bit, so it was 1.000111, so add 1, so that gives me 001 plus 1000, then 100, so it will be 1.001000. So, this will be the representation in the binary form for 2's complement, this is as good as this is same as subtracting 0.875 from 2, so 2 means 10.000000 and 0.875 is 0.111000 and you are subtracting this, so that will give you 1.00100, so this is same as this. Addition of two fixed point numbers, if it is 0.5 which is 0.100 in the binary form and 0.125, which is 0.001 in the binary form if you add them together, if the total does not exceed 1 there is no problem.

(Refer Slide Time: 26:54)



So, 0.625 is the addition of 0.5 plus 0.125 in the decimal system, and it will be 0.100 plus 0.001, so that gives me 0.101 this sign bit is not changing, but if the sum exceeds 1 that is rather exceeds the limit of 3 bits, here the 3 bit representation if it exceeds that, then we have problems. That carry will be causing overflow say 0.5 plus 0.6 to 5, if we had this is 0.100 and 0.6 to 5 is 0.101, if you add this is 100.1, which is causing a negative sign basically. So, this will be causing an overflow which is here confuse with a sign bit, so one point actual sum is 1.125, so 1.125 cannot be represented in a 3 bit system.

(Refer Slide Time: 29:18)



The subtraction of two fix points number, this we can do by 2's complement say 0.5, we want to subtract 0.25 from here, so that gives me 0.25, so 0.5 in 2's complement it is 0.100 whether it is a positive number there is no problem. For a negative number 0.25 is how much and then complement it will be 1.101, so 1.101. We have to add 1 for 2's complement this is 1's complement, this 2's complement 1 1 1 0 1 and 1, so 1.110 add them straight 0 1 0 0 1 and then neglect this.

So, you drop this, so 0.010 this is for 0.25, so this carry is dropped, otherwise 1.110, let us say 1.110 what will be 2's complement, 0.010 you can do by a simple subtraction also. The 2's complement is simpler you just take the negative quantity in 2's complement, positive quantity in remains as it is, so add them together that gives you the result. Multiplication this is somewhat involved we see that, suppose we have a 2 bit representation of 2 numbers, the product is a 4 bit number, when you multiply specially you get this problem. So, if you have 2 bit number for the integral part plus 2 bit number in the fraction part, then the product may give you 4 bit part in the integral part and 4 bit part in the fraction part.

(Refer Slide Time: 33:27)

$\frac{1}{2} \leq M \leq 1$ $M \rightarrow$ Mantissa
 $C = +ve$ or $-ve$

$4.5 = 2^3 \times \left(\frac{6.5}{8}\right)$
 $= 2^3 \times 0.8125 = 2^{011} \times 0.1101$

$1.5 = 2^1 \times \frac{1.5}{2} = 2^1 \times 75 = 2^{001} \times 0.1100$

$6.5 = 2^2 \times 8.125 = 2^{011} \times 0.1101$

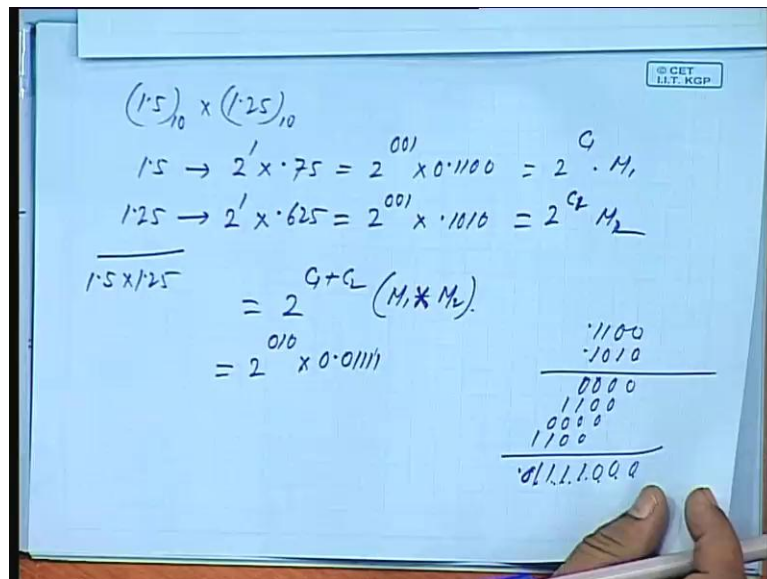
$0.625 = 2^0 \times 0.625 = 2^{000} \times 0.1010$

So, in a floating representation we write a number in this form, where M is written half and 1, M is a mantissa, so C is either positive or negative, for example 4.5 we should write as 2 to the power how much, if I write 2 to the power 2, then I will get a quantity m which will be more than 1. So, it should be always next higher number which is some

power of 2, next higher number is 8, so 8 into something 8 into 4.5 by 8, so it is 2 to the power 3 multiplied by 0.5625.

So, it will be 2 to the power this 3 is 011 multiplied by 0.5625 if you write in a 4 bit representation it will be like this 0.5 plus 0.25 and so on. Similarly, 1.5 will be next number is 2, so 2 to the power 1 into 1.5 by 2, so that is 2 to the power 1 into 0.75 and that is 2 to the power 1, so it will be 001 into 0.750 0.1100. 6.5 2 to the power 3 into 0.8125, so that gives me 2 to the power 001, to the power 3 011 into 0.1101, 0.625 it will be 2 to the power 0 into 0.625 is to be 2 to the power 000 into 0 0.1010. So, like this any number you can express, in terms of decimal binary forms.

(Refer Slide Time: 36:33)



Let us now multiply 1.5 into 1.25, so 1.5 is 2 to the power 1 into 0.75, that is 2 to the power 001 into 0.1100. So 2 to the power C 1 into M 1, 1.25 is 2 to the power 1 into 0.625 that is 2 to the power 001 into 0.1010, so that is 2 to the power C 2 in to M 1, 2 to the power C 1, C 1 is equal to C 2 M 2. So, multiplication will give me 2 to the power C 1 that is common plus C 2, 2 to the power C 1 plus 2 to the power C 2 got it, into M 2. So, 2 to the power 010 into 0 product of these 2, so 0.1100 0.1010 what is the product, 0000 1100, then 0000 1100, if I add 111. Now, 1 2 3 4 5 6 7 8, 1 2 3 4 7 8, so 1 0, so 0.0111 1 1 2 4, if it is a 4 bit representation then you have to truncate.

(Refer Slide Time: 39:06)

$3.0 = 2^2 \times 0.75 = 2^{010} \times 0.110000$
 $0.125 = 2^{-2} \times 0.03125 = 2^{010} \times 0.0000100$
 $3.0 + 0.125 \rightarrow 2^{010} \times (1.100100)$

Quantization error.
Direct form \rightarrow
Cascaded form, parallel form

Now, if you have to add two floating point numbers for addition or subtraction for that matter, for addition of two floating point numbers, see when you add 10 with 7, basically you put a 0 here. So, both of them are brought to the same level of representation same number of digits, so you have to bring them to the same number of digits and then only you can add.

So, 3.0 plus 0.125 what will be the addition of these two, so 3.0 is 2 to the power 2 into 0.75, so 2 to the power 010 into 0.110000, 0.125 is 2 to the power 2 you have to bring the same index into 0.125 by 4, so 0.03125 which gives me 2 to the power 010 into 0.0000100. So, if you add 3.0 plus 0.125 that gives me, correct me if I am wrong 0.125 by 4, so that gives me this, so this will be 2 to the power 010 into product of this two...

Student: ((Refer Time: 41:01))

Will become... this is sum, so this will give me 0.1100 1100100, so will take it up in further details later on if time permits, otherwise I leave at this, you can read it off from the books for further details of this. Now, quantization error as we are discuss in earlier, in the direct form the coefficients of the polynomials will be appearing in the structure as constants as a multiplier, is it not. In the other forms that is if you have cascaded form or parallel form the constant of these are not going to affect all the roots, they are going to affect only that particular pair.

(Refer Slide Time: 42:40)

$$\frac{B(z)}{A(z)} = \prod_{i=1}^K \frac{b_{i0} + b_{i1}z^{-1} + b_{i2}z^{-2}}{1 - a_{i1}z^{-1} - a_{i2}z^{-2}}$$

$$= \frac{K_1}{1 - a_{11}z^{-1} - a_{12}z^{-2}} + \frac{K_2}{1 - a_{21}z^{-1} - a_{22}z^{-2}} + \dots$$

$$\frac{B(z)}{A(z)} = \frac{b_0 + b_1z^{-1} + b_2z^{-2}}{1 - a_1z^{-1} - a_2z^{-2}}$$


For example, if I have say $A(z)$, $B(z)$ by $A(z)$ as product of some b_{i0} plus $b_{i1}z^{-1}$ inverse plus $b_{i2}z^{-2}$ to the power minus 2 divided by $1 - a_{i1}z^{-1} - a_{i2}z^{-2}$ to the power minus 2 and i varying from say 1 to K . That means, there are many bike wards I have got in cascade, I could have also written this as some like a constant K_1 by $1 - a_{11}z^{-1} - a_{12}z^{-2}$ to the power minus 2 plus a K_2 minus $a_{21}z^{-1} - a_{22}z^{-2}$ to the power minus 2, and so on, I could have put in parallel forms.

So, here the two 0's and two poles only will be affected by quantization of these coefficients, whereas in the direct form all the coefficients that is $B(z)$ by $A(z)$ say b_0 plus b_1z^{-1} inverse plus b_2z^{-2} to the power minus 2 and so on. If you take the polynomial as it is and quantization of this will be affecting all the poles and 0's, here quantization of a coefficients, suppose forget about others just one coefficient is changed only this whole pair will be affected, others will not be affected.

So, here we are restricting it within a certain group, restricting the sensitivity of a particular group of poles and 0's with the variation of these coefficients. So, this is much more robust, however in these case the poles are restricted, 0's will be 0's are not known they are in a very complex mode, because if you add up K_1 in to a_{21} , a_{22} , K_2 into a a_{11} , a_{12} and so on, all this terms will appear. So, here only the poles will be affected by rather the effect of variation of the poles, these parameters will be restricted to poles, but these parameters will not be a affecting the 0's independently.

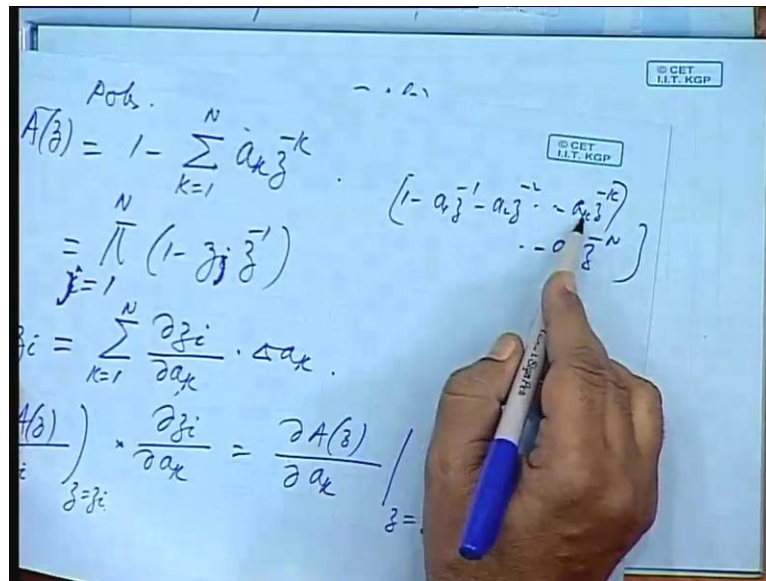
(Refer Slide Time: 45:59)

smaller bit rep.
computation is more.

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 - \sum_{k=1}^N a_k z^{-k}}$$
$$\hat{H}(z) = \frac{\sum_{k=0}^M \hat{b}_k z^{-k}}{1 - \sum_{k=1}^N \hat{a}_k z^{-k}}$$
$$\hat{b}_k = b_k + \Delta b_k$$
$$\hat{a}_k = a_k + \Delta a_k$$


In normalized lattice, if you are having a smaller bit representation, then this is a very robust one, very useful for smaller bit representation, but it needs lot of computation, computation is more. ((Refer Time: 46:31)) In the other two cases the number of elements required will be same as a direct form a little bit of extra computation will be involved. So, these three representation that is normalize lattice or cascade or parallel forms, they will be more robust compare to a direct form. Now, will see the effect of quantization, let us consider $H(z)$ as K varying from 1 to N , K varying from 0 to M , so $\hat{H}(z)$ will be $\hat{b}_k z^{-k}$ to the power minus K , \hat{b}_k is b_k plus Δb_k the error due to quantization, actual value plus the error.

(Refer Slide Time: 48:16)



Let us consider $A(z)$, the poles we can write this as $1 - a_k z^{-k}$, k equal to 1 to N , this can be factorized. And I can write this as $1 - z_j z^{-1}$, j varying from 1 to N , N there are its an N th order polynomial having N roots. Now, let us see the sensitivity, let us see the change Δz_i is $\sum \Delta z_i$ by Δa_k in to Δa_k , that is shift in the sensitivity of the i th root with respect to any one constant, multiplied by the change in that constant that will be the small change due to a particular element, particular parameter.

Similarly, if you sum up, then that will be the total change in the position of a particular root, each root is dependent on all the parameters a_1, a_2, a_3, a_4 , so you make a small change in a_1, a_2 and so on. So, collect the total change, so the total change in that i th root will be so much, again k varying from 1 to N . Now, $\Delta A(z)$ by Δz_i multiplied by Δz_i by Δa_k is equal to $\Delta A(z)$ by Δa_k $z = z_i$. So, change in this, a partial derivative of this with respect to any root z_i , and if you compute that slope at $z = z_i$, that multiplied by the sensitivity of z_i with respect to a_k will be the sensitivity of this that partial derivative of $A(z)$ with respect to the coefficient a_k .

(Refer Slide Time: 51:36)

The image shows a handwritten derivation on a blue background. At the top, it says $\frac{\partial a_k}{\partial z^i}$ and $\left[\frac{\partial A(z)}{\partial z^i} \right]_{z=z^i}$. Below this, the expression is written as $= \frac{+ z^{-k} \cdot z^i}{\prod_{\substack{j=1 \\ j \neq i}}^N (z^i - z_j)}$. To the right of the equation is a pole-zero plot on a complex plane. A circle is drawn around the origin. Several poles are marked with 'x' and zeros with 'o'. A specific pole is highlighted with a red dot, and a vector is drawn from the origin to this pole. Another vector is drawn from the origin to a zero. The plot illustrates the relationship between the poles and zeros in the complex plane.

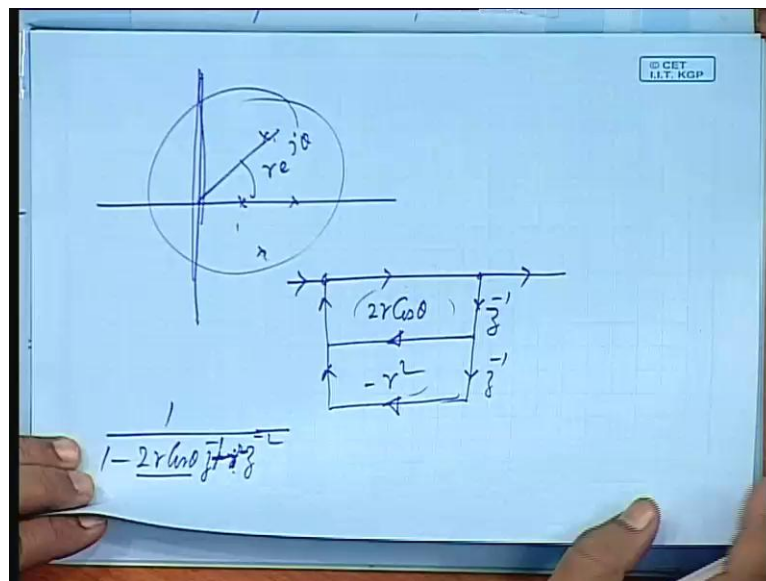
So, from here $\frac{\partial z^i}{\partial a_k}$ will be this divided by this, if you expand this $\frac{\partial z^i}{\partial a_k}$ by $\frac{\partial a_k}{\partial z^i}$, ((Refer Time: 52:11)) you can just see take the derivative with respect to a_k . So that will give me what it is $1 - 1 z$ to the power minus 1, so a $2 z$ to the power minus 2, and so on. If I take a $k z$ to the power minus k minus a $N z$ to the power minus N , if I take the derivative with respect to a_k it will be z to the power minus k with a negative sign.

So, this will be z to the power minus K , similarly if you do this will get z to the power i and then $z^i - z^j$ if you evaluate this, j equal to 1 to N , j not equal to i and there will be a negative sign here, so that will make it positive. Now, you see this term what does it mean? Suppose the poles we are now considering only the denominator factor poles, if the poles are very tightly clustered, there will be complex conjugates know, there will be always appearing in the form of complex conjugate.

Now, these are the vectors $z^i - z^j$, say suppose this is the i th one, so these are the $z^i - z^j$ quantities $z^i - z^j$, now if you take the products if this z^i shifts a little it is a quite a big change. It is somewhere a 0.01 to 0.02, the difference is say if it is at a distance of 5.1, if it shifts to 5.2, there is not much of a change 5.1 to 5.2, but the distance between the two poles if there all in close cluster, it may be from 0.01 to 0.02 it is doubled, do you get my point.

So, the denominator which is the product of such vectors, finally will give you a big change, so they are very, very sensitive specially when they are in close cluster same is the case with the 0's, this is a narrow band, band pass filter basically. If you are having close cluster here that will correspond to a low pass filter, narrow band low pass filter; low pass filter roots are somewhere close to the real axis and band pass filters here away from the real axis, now let us see what would be the effect on this.

(Refer Slide Time: 55:43)



We have say for a quadratic representation, we may have something like this, you can write a quadratic form in this fashion also, that is $1 - 2r \cos \theta z^{-1} + r^2 z^{-2}$. Suppose, we have the bike ward representation like this, $1 - 2r \cos \theta z^{-1} + r^2 z^{-2}$ it is in a quadratic form, if the roots are complex.

If the roots are simple on the real axis, then there is no problem the roots are here, if they are complex then we can represent by this form, so the roots are at $r e^{j\theta}$ and $r e^{-j\theta}$. Suppose, we quantize these two quantities, then what are the possible values, so will take it up in the next class, the time is just over, what will be the effect on quantization of these quantities.

Thank you very much, I will take it up on.