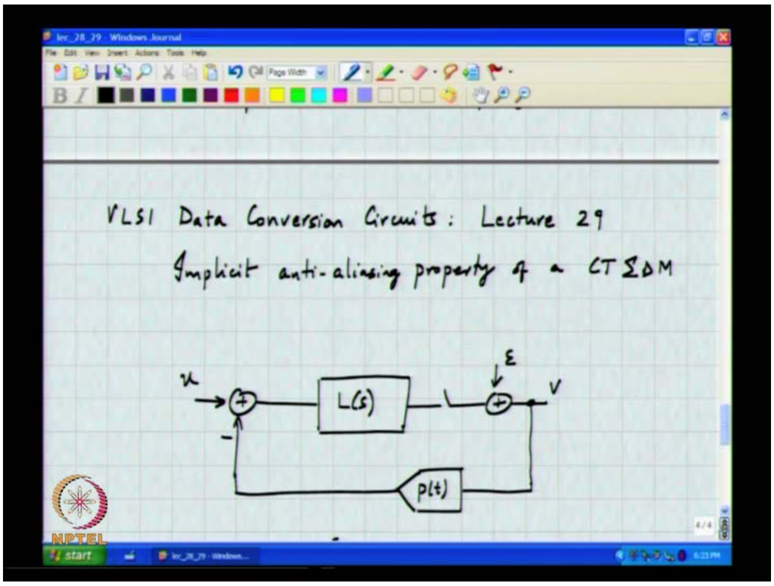
VLSI Data Conversion Circuits
Prof. Shanthi Pavan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture - 29
Modulators with NRZ and Impulsive DACs

This is VLSI data conversion circuits – lecture twenty nine.

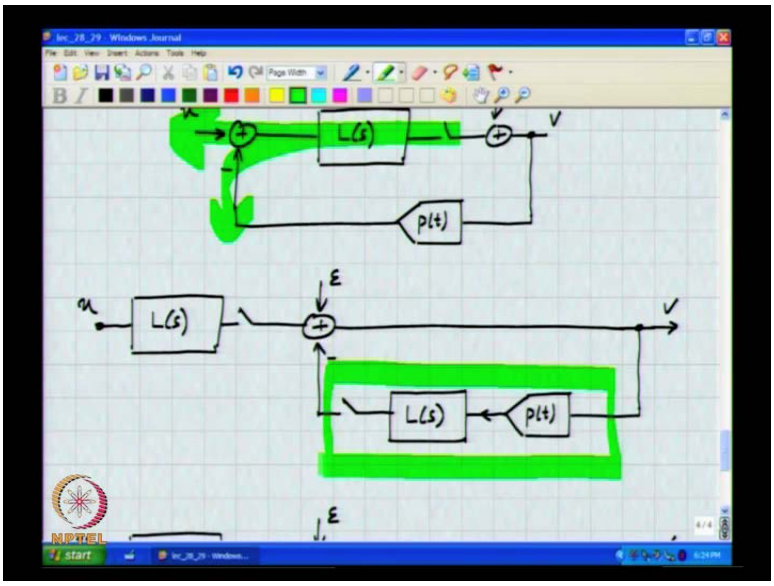
(Refer Slide Time: 00:14)



In the last class, we saw what was called the implicit anti-aliasing property of a continuous-time delta sigma modulator, where we saw this specific case of a first order loop filter, where the noise transfer function was 1 minus z inverse. And, we saw that, precisely at multiples of the sampling rate of the modulator, the magnitude of the signal transfer function has actually 0. And, around multiples of the sampling rate, the signal transfer function is small. And therefore, the gain for frequencies, which can potentially alias into the signal band, is very small. So, it is as if one made a regular A to D converter and put an anti-aliasing filter upfront. Here we see that, all that stuff is combined into one structure.

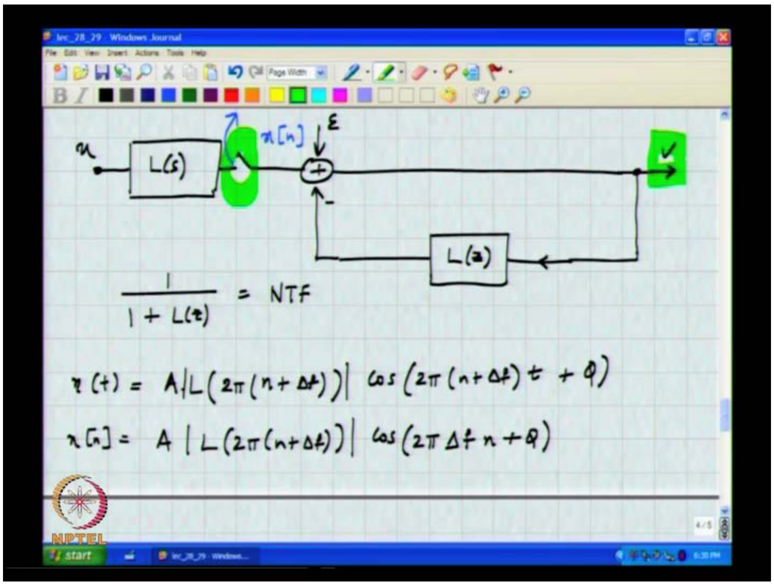
Now, let us generalize a little bit and look at what happens in a continuous-time delta sigma modulator, where the loop filter transfer function is L of s; where again as I mentioned, L of s convolved with p of t and sampled at the sampling rate must have the same response as L of z; where, L of z is being derived from the desired noise transfer function. Correct?

(Refer Slide Time: 02:16)



So, like in the first order example, we can manipulate this system into an equivalent mathematical model by separating the discrete-time and the continuous-time parts. We pull the loop filter out in this fashion. So, all I have done is move the loop filter through this summer like this; and, also pull the sampling operation out of the loop, which results in this diagram. Then, we identify this animal with L of z. Therefore, one ends up with this equivalent of the modulator.

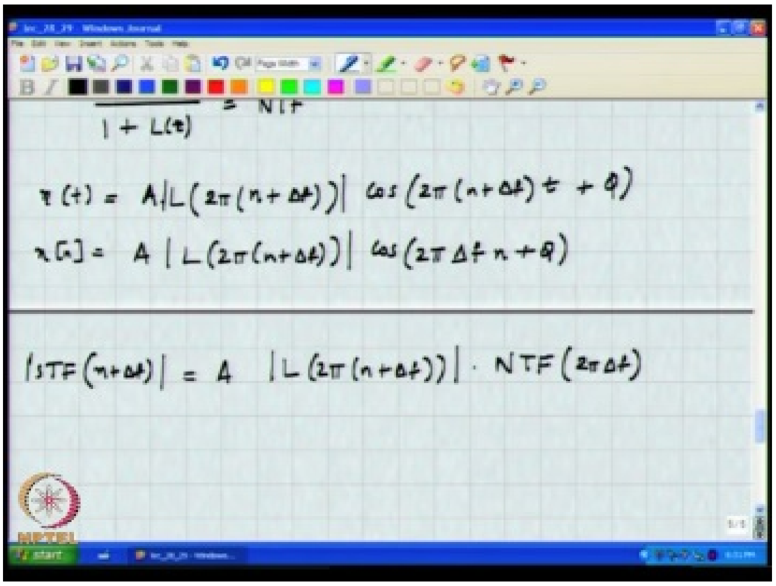
(Refer Slide Time: 03:32)



By definition, if I call 1 by 1 plus L of z as the NTF; then, let me call this x of t; this is x of n. And, when I am interested in finding the STF as the function of the input frequency, what am I doing? I am putting in a sinusoid at some frequency f i; this f i need not be confined to 0 to f s by 2. So, this point is exploring the response of this modulator to any possible input frequency. Correct? If the input frequency happens to be greater than f s by 2, then due to the sampling operation here, it will alias to a frequency, which is less than f s by 2. Specifically, frequencies in the neighborhood of f s will alias to frequencies around DC, which correspond to the desired signal band. Correct? And therefore, plotting the magnitude of the signal transfer function as a function of input frequency; where, input frequency is swept from 0 all the way to infinity will tell us what the response of this modulator loop looks like for frequencies in the alias bands.

And, as I mentioned, it makes sense to work with normalized quantities, where the sampling rate is 1 hertz; in which case, all the alias bands will be of the form n plus delta f in hertz; where does the delta f is a very small number; and, the maximum it can be is 1 by 2 OSR. Does it make sense? And, as we saw with the first order example, the procedure for calculating the signal transfer function is very straightforward. All that one does is finds the loop filter transfer function evaluated at say 2 pi into n plus delta f. Correct? This is the magnitude of the loop filter transfer function at a frequency n plus delta f hertz. Correct? This times whatever – cos of 2 pi into n plus delta f times t is plus some phi is x of t. So, if I assume the input is of the form A times cos 2 pi n plus delta f into t; the output of the loop filter is nothing but a sinusoid with the same input frequency, but whose magnitude and phase are modified according to the transfer function of the loop filter in this manner. Is this clear? So, x of n is simply the sampled version of this sinusoid, which will be A times magnitude of L of 2 pi into n plus delta f cos of 2 pi delta f into n plus phi. Does it make sense?

(Refer Slide Time: 08:26)



Therefore, the amplitude at V is simply the amplitude of the sinusoid at x, which happens to be A mod L 2 pi n plus delta f multiplied by 1 by 1 plus L of z; which happens to be the NTF evaluated at?

Student: ((Refer Slide Time: 08:53))

At?

Student: Delta f

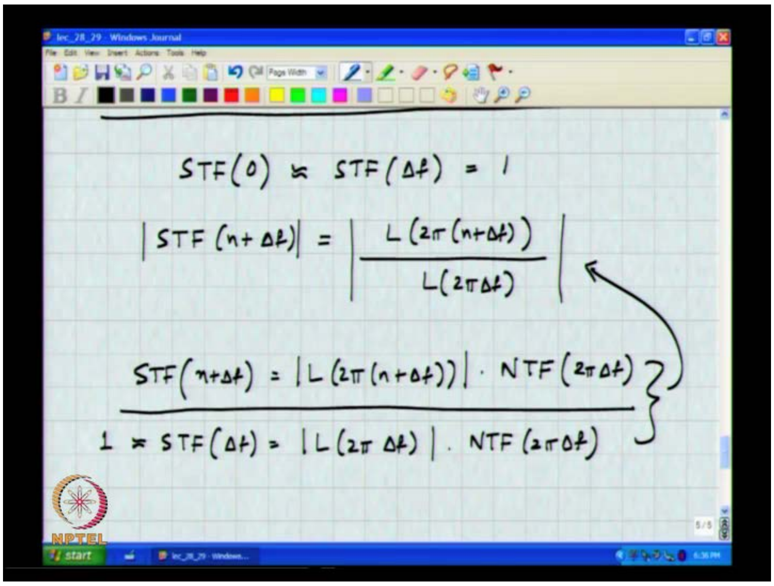
Delta f. Please note that, this has to be in terms of omega. Correct? So, half corresponds to… Or, 1 corresponds to 2 pi. So, delta f must correspond to?

Student: 2 pi delta f

2 pi delta f. If one talked in terms of radiance throughout, then it will be A times the L of 2 pi whatever – n plus… The sampling rate is n hertz; it will be 2 pi n plus delta omega.

And, you multiply with NTF of delta omega. And, this will give you the magnitude of the signal transfer function at n plus delta f; and, divide down by the input amplitude.

(Refer Slide Time: 10:15)



So, the magnitude of the signal transfer function is this. So, at DC, what is STF? It will be?

Student: ((Refer Slide Time: 10:49))

It will be equal to?

Student: 1

1. So, STF of 0 will be approximately STF of delta f; where, delta f is small; which is approximately 1. And, why does this make sense? Because if you want a high-order noise transfer function; at low frequencies, NTF will go as omega to the power n; which means that, in the loop filter, you must have?

Student: 1 by omega…

1 by omega to the power n. Correct? Which means that, how many integrators must be there in the loop?

Student: N integrators

N integrators. So, the DC gain of the loop filter will be very very high; which means that, the STF at DC is loop gained by 1 plus loop gain; which will practically be 1. Is this clear? So, the STF at n plus delta f therefore, is nothing but the magnitude of the loop gain at delta f divided by the magnitude of the loop gain at n plus delta f. Does it make sense?

Since I see many blank faces, let me... So, in this equation, I will simply substitute STF of delta f is nothing but mod L of 2 pi delta f times the NTF of 2 pi delta f. So, you divide one by the other; the noise transfer function goes away in both cases. That makes sense because whether you put in n plus delta f or you put in delta f; after sampling, both of them look like discrete-times sinusoids with the?

Student: Same frequency

Same frequency; correct? So, it only makes sense that, this NTF of 2 pi delta f goes away and you get STF evaluated at n plus delta f is nothing but the loop filter transfer function evaluated at delta f divided by loop filter transfer function evaluated at n plus delta f multiplied by the STF evaluated at delta f. The STF evaluated at delta f is approximately 1; correct? which means that, you divide the two and then you get this equation.

Student: ((Refer Slide Time: 14:44)) Reverse

Yes; 2 pi into n plus delta f by L of 2 pi delta f. Is this clear? So, do you think for a second order modulator, the alias rejection around the sampling rate – is it more than what you would get for the first order modulator or less than what you would get for a first order modulator?

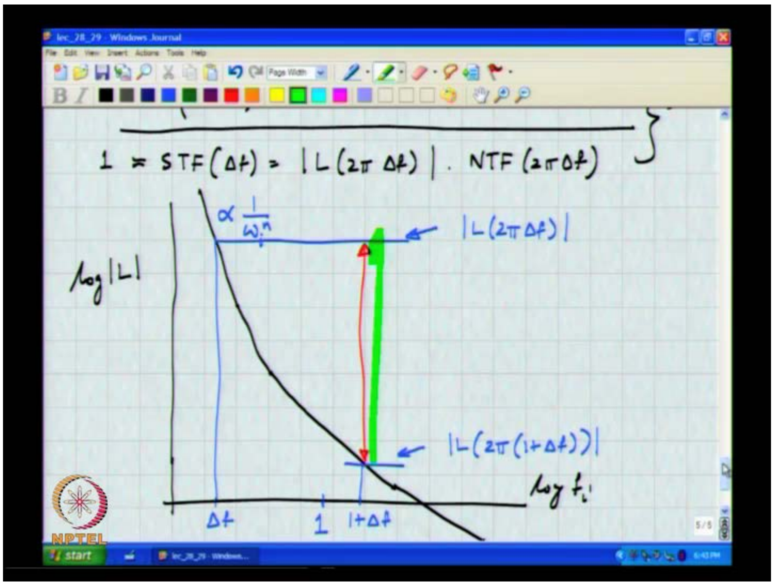
Student: More

It is?

Student: More

Definitely more. And, for a higher order modulator, the alias rejection keeps getting better and better.

(Refer Slide Time: 15:47)



If you have a high order loop filter, at low frequency, it will behave like… If you have an n-th order loop filter; at low frequency, how will it behave like?

Student: 1 by omega to the power n

1 by omega to the power n; so, this for example, here; this will be proportional to 1 by omega i to the power n. As frequency goes on increasing, what do you think will happen?

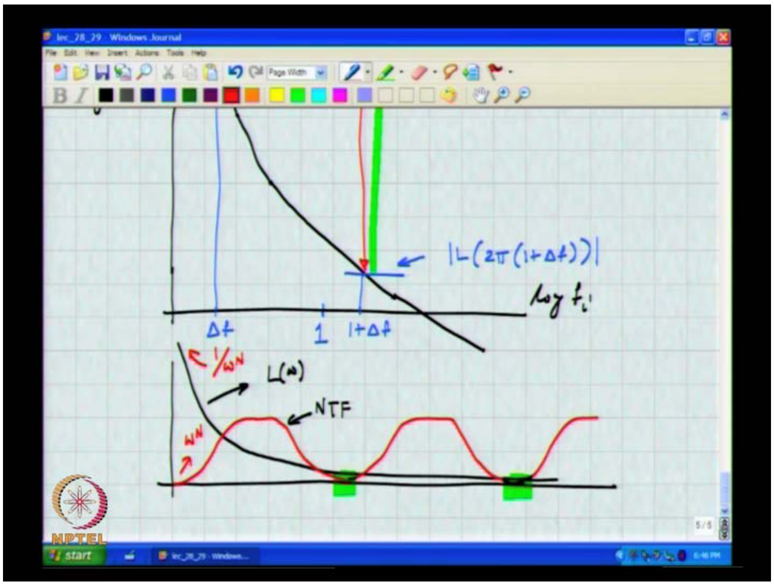
Student: It will fall off.

You would expect it to fall off. The rate at which it falls off will not go as 1 by omega to the power n, but something smaller. At this point, it is perhaps not obvious to you; but, in the feedback control classes, if we have a high order transfer function, the loop gain for a system to be stable – when it crosses the 0 dB line, it must fall as?

Student: ((Refer Slide Time: 17:01))

At 20 dB per decay; so, however high order the system is, at low frequency, the loop gain can be very high. But, as it keeps falling, eventually it must fall off as 1 by omega when it crosses the 0 dB line. So, the similar thing happens here too even though the loop here is a lot more complicated because of a sampling inside the loop. But, intuitively at least, you see that, you cannot have this gain, which goes off, which falls off all the time as omega to the n. So, if I want to find the alias rejection at… Let us say this is the sampling rate; and, this is 1 plus delta f; and, delta f is somewhere here; do not choose this ((Refer Slide Time: 18:05)) Delta f is somewhere here. So, the alias rejection at 1 plus delta f is given by this amount, because this value is mod L evaluated at 2 pi delta f; and, this is mod L evaluated at 2 pi 1 plus delta f; correct? And therefore, the STF for a frequency delta f is higher than the STF for a frequency 1 plus delta f by this amount. Correct? Because of this relationship; do you understand? What is this equation telling us? The STF at n plus delta f is the gain of the loop filter at n plus delta f divided by the gain at delta f. On a log scale, it is simply the difference between the magnitudes of the loop filter gain at f and 1 plus delta f. Do you understand?

(Refer Slide Time: 20:22)

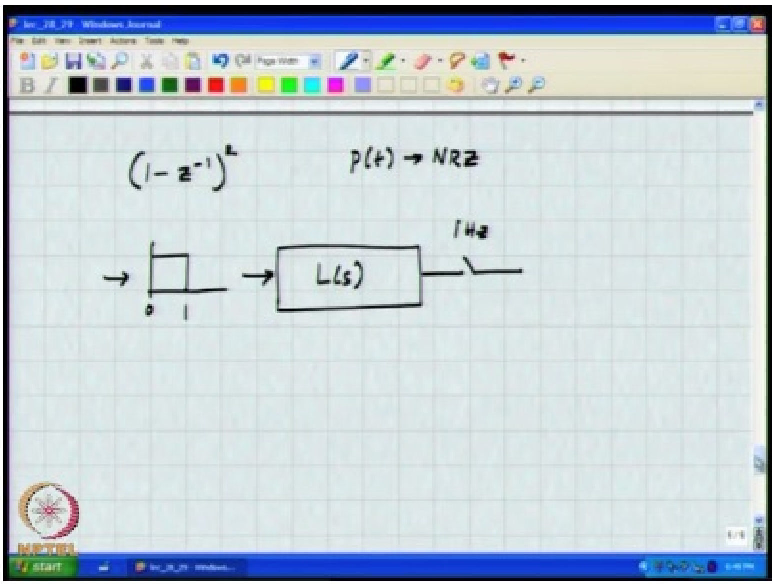


So, the actual STF of course will perhaps look like this. So, the loop filter does something like this. A high order noise transfer function will do something like this. So, this will go as 1 by omega to the n; this goes as omega to the n. So, at DC, the two of them will cancel and you will get an STF of 1. But, to get the complete STF, as we saw, we need to multiply L of omega with a periodic extension of the noise transfer function. So, you will see periodic nulls at all multiples of the sampling rate. And therefore, in all the alias bands, you will see a very low value for the STF. And, the higher the order of the loop filter, the higher the reaction. Does it make sense? So, intuitively, of course as I was mentioning in the last class, this makes a lot of sense, because fundamentally, when compared to a discrete-time loop, the treatment meted out to a sinusoid at n plus delta f, is very different from the treatment meted out to?

Student: ((Refer Slide Time: 22:32))

A sinusoid at delta f as can be seen from this equivalence. Is this clear? And therefore, it makes sense that, the response to high frequencies or response to frequencies around the sampling rate, is actually very very small. So, this is one of the very unique and interesting features of a continuous-time delta sigma modulator, where these things feature anti-aliasing, is an implicit feature of the modulator architecture itself. Or, it is an implicit feature of the A to D converter architecture itself. And therefore, one need not worry about having to put a… At least in principle, one need not worry about having to put an explicit anti-aliasing filter upfront; which is something that would be needed in a discrete-time design. So, that is all I had to say about one of the specialties of implementing the loop filter in continuous time. So, now, let us actually go ahead and try and figure out what the loop filter must be in order to realize more complicated noise transfer functions.

(Refer Slide Time: 24:36)

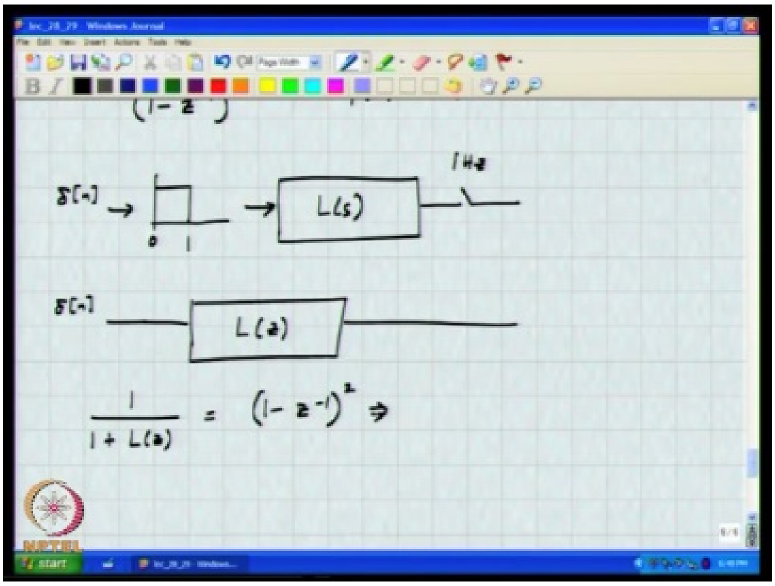


So, we were done with the first order system; now, let us try and realize a noise transfer function of the form 1 minus z inverse the whole square. And, the problem as I mentioned is to determine L of s such that this noise transfer function is realized. And, what all do I need to know?

Student: Pulse shape

So, we need to know the pulse shape. Let us assume the pulse shape is an NRZ pulse. So, the pulse shape when pushed into L of s and sampled at 1 hertz must result in the same sequence that one would get with a discrete-time loop filter.

(Refer Slide Time: 25:50)

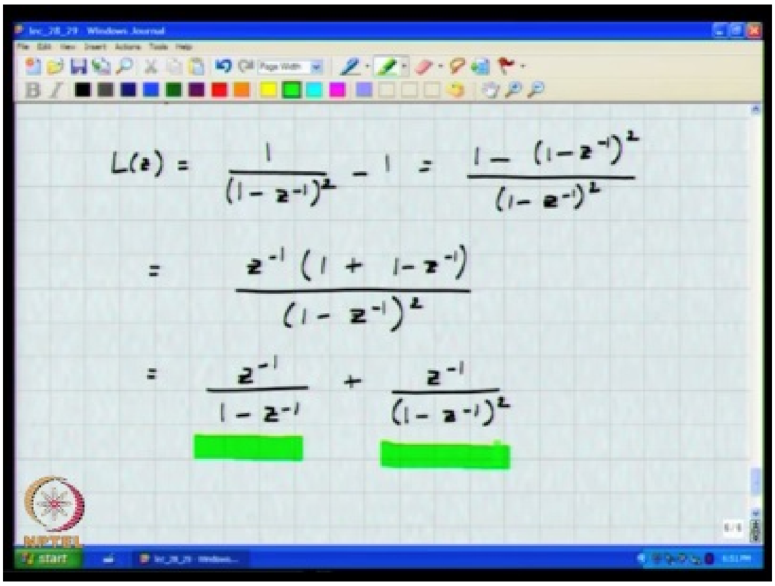


And, what is the discrete-time L of z?

Student: Because it is not…

So, what is the… So, 1 by 1 plus L of z must be 1 minus z inverse the whole square; correct?

(Refer Slide Time: 26:24)



Which means that, L of z must be 1 by 1 minus z inverse the whole square minus 1; which can be written as 1 minus 1 minus z inverse the whole square divided by 1 minus z inverse the whole square; which is z inverse times 1 plus 1 minus z inverse divided by 1 minus z inverse the whole square; which is z inverse by 1 minus z inverse plus z inverse divided by 1 minus z inverse the whole square. This is what…

Student: G of n minus 1

Pardon

Student: G of n minus 1

What kind of building block is this?

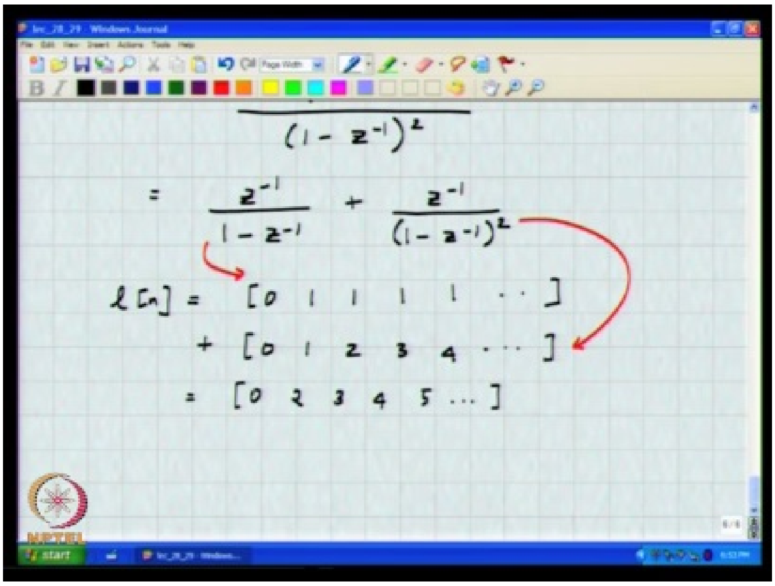
Student: Accumulator and…

It is a discrete time… It is an accumulator or discrete time integrator. So, what would 1 minus z inverse whole square be?

Student: Cascade of two accumulators

It is cascade of two accumulators or cascade of two integrators. This makes sense, because if you want second order rejection, you must have two integrators in the loop filter. Does it satisfy… What must be the first sample of the impulse response of the loop filter? 0. And, is that satisfied here? Yes, why? There is a z inverse multiplying everything. So… So, this is L of z. So, what must L of n be?

(Refer Slide Time: 28:42)



What is the discrete time sequence, which corresponds to this z transform? It is 0, 1, 1, 1, 1 and so on, which is this sequence. And, how does this look like? 0…

Student: 1

It is simply in accumulation of…

Student: 2…

4 and so on. So, L of n corresponds to 0, 2, 3, 4, 5 and so on. Now, let us get to the… We need now to find a continuous-time loop filter, which when excited by a rectangular pulse and sampled, will give you what samples?

Student: ((Refer Slide Time: 30:02))

These samples. So, at this point, we have no clue of what L of s looks like. But, we know one thing, that is, how many integrators do you think you need?

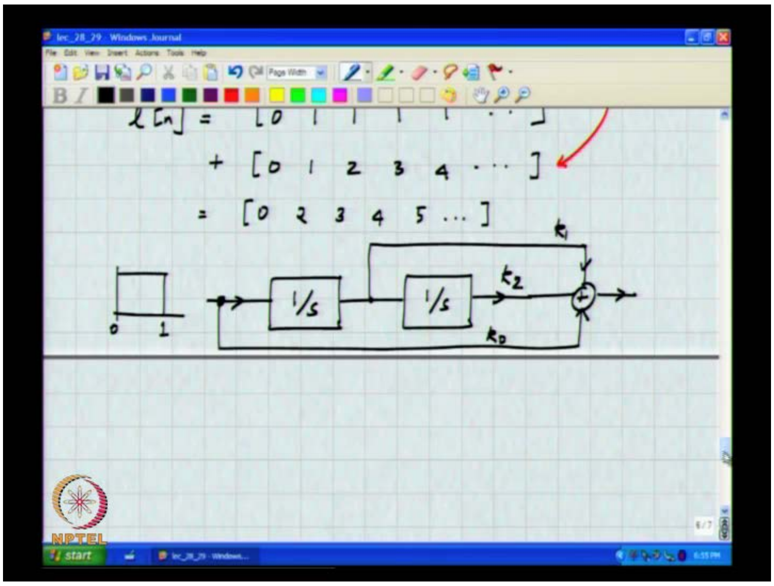
Student: Two

You will need?

Student: Two

Two integrators

(Refer Slide Time: 30:19)



So, in general, let us say you have 1 over s. And, in general, the loop filter should be of the form? At low frequencies, it must be of the form 1 by s square. So, in general what do you think will be the form of the loop filter transfer function?

Student: 1 by s square by omega square ((Refer Slide Time: 31:09))

At DC, it must go as 1 by s square; correct? So, the loop filter must definitely have 1 by s square terms. Then, what else?

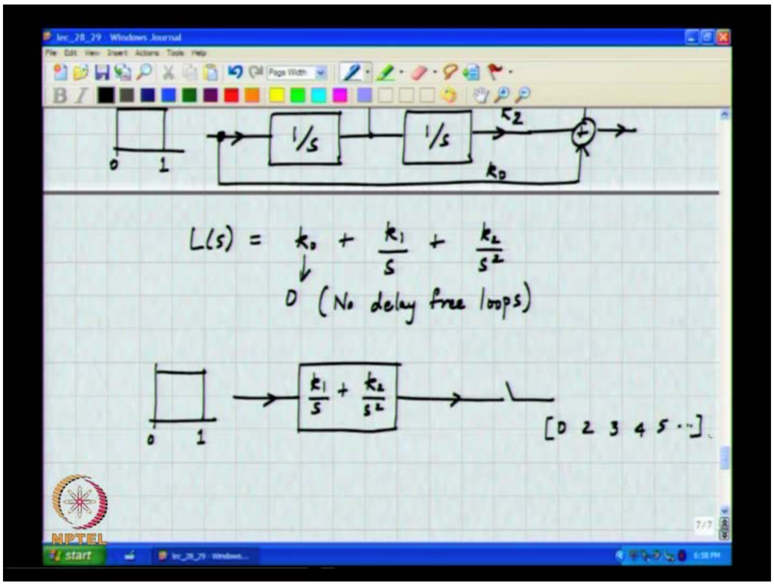
Student: At high frequency, one should go on…

If you have 1 by… If you have two integrators, in general, the output of the loop filter is a linear combination of the outputs of?

Student: Two integrators

All the integrators you have. So, for example, I will call this k 2; I will call this k 1; add this up here. How about k naught? Yes, any comments on this?

(Refer Slide Time: 32:44)



In other words, all I am saying is the form of the loop filter transfer function L of s is some k naught plus k 1 by s plus k 2 by s square. And, what we are interested in finding are? What are we trying to find?

Student: You feed an NRZ pulse transfer…

So you feed an NRZ pulse into L of s; sample the output of the loop filter.

Student: At 2 equal to two portions…

And, what are we trying to determine?

Student: k naught, k 1…

k naught, k 1 and?

Student: k 2

k 2; is this clear? So, can somebody say something right away about k naught?

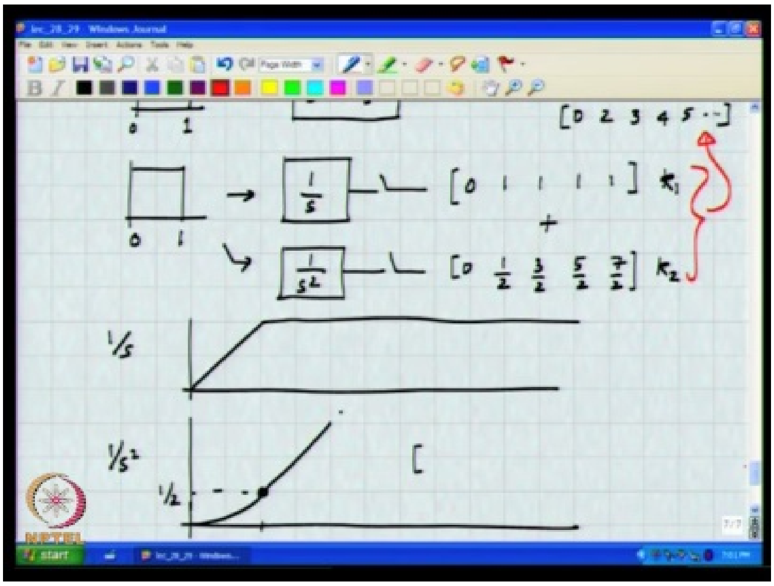
Student: ((Refer Slide Time: 33:28))

k naught must be 0; and, why?

Student: Because of the first sample of the…

If k naught is not 0, then if you look into the modulator, there is a delay-free loop, because it is like having a direct feedback path around the quantizer. So, k naught must be 0, because you must have no delay-free loops. Therefore, our problem reduces to finding k 1 and k 2, so that if a rectangular pulse is pushed into the loop filter and the output of the loop filter is sampled, I must get 0, 2, 3, 4, 5 and so on; correct?

(Refer Slide Time: 34:56)



So, let us do this step by step. If a rectangular pulse goes into 1 by s, what will I get? And, if I sample the output of 1 by s, what will I get?

Student: 0 samples.

0?

Student: 1, 1, 1, 1…

1, 1, 1, 1 and so on. And, if this goes into 1 by s square; and, if I sample the output, what happens?

Student: 0, 1…

No, no, no.

Student: ((Refer Slide Time: 35:40))

Please note that, putting it into 1 by s square is not equivalent to taking that discrete-time sequence and accumulating it. What you need to do is to find the continuous-time waveform and sample it; not the other way around. So, how will the… If you accumulate the pulse, what will you get? You will get something like this. This is the output of the 1 by s. Correct? This has to be accumulated to get the output of the 1 by s square term. So, what will it be? After 1 second, it will be?

Student: Half

Half. And, how will the waveform look?

Student: Parabola

It will look like a parabola. And, after that?

Student: ((Refer Slide Time: 36:40))

It will look like a ramp; and, keep doing this. So, the samples of the 1 by s square will be?

Student: 0

0 of course…

Student: 1 by 2...

This one which is half; then?

Student: ((Refer Slide Time: 37:04))

After that, the input to the second integrator is a constant value equal to 1. So, it must raise at?

Student: Slope 1

At slope of 1. So, this must be 3 by 2, 5 by 2, 7 by 2 and so on. Does it make sense? So, the output of the integrator must be multiplied by k 1. The output of the two integrator cascade must be multiplied by k 2. And, I need to determine k 1 and k 2 such that the sum of these sequences must exactly be equal to this character. So, clearly, is this an under constraint problem or an over constraint one? How many equations do we have and how many variables do we need to find?

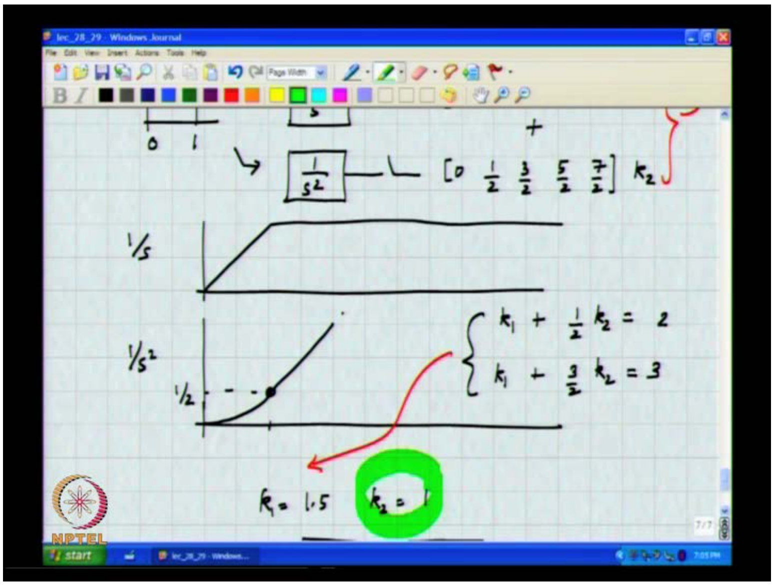
Student: More equations are ((Refer Slide Time: 38:22))

We have?

Student: More equations than unknowns.

More equations than unknowns; so, what are the equations?

(Refer Slide Time: 38:34)

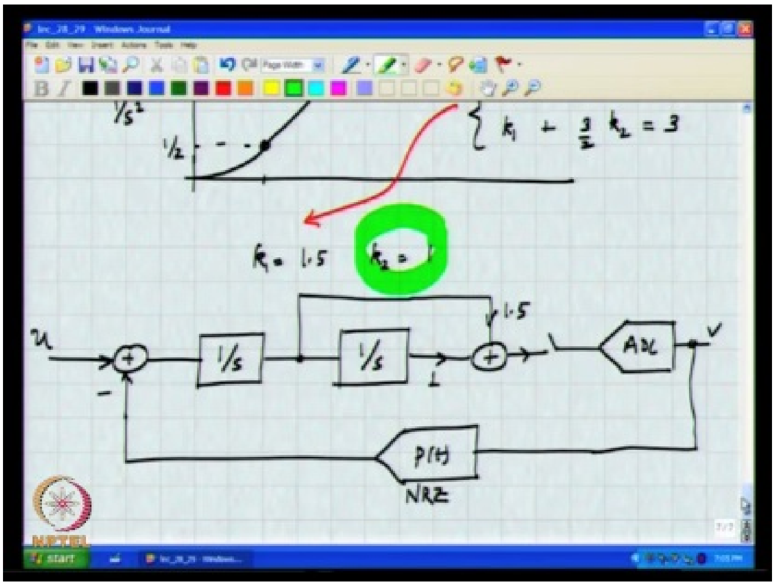


k 1 plus half k 2 is 2. k 1 plus 3-halves k 2 is 3; from which we obtained k 2 is 1. You subtract the two equations. k 1 goes away and k 2 simply becomes equal to 1. And, what about k 1?

Student: 1.1

Is one and a half. So, k 1 is one and half; k 2 is 1.

(Refer Slide Time: 39:53)



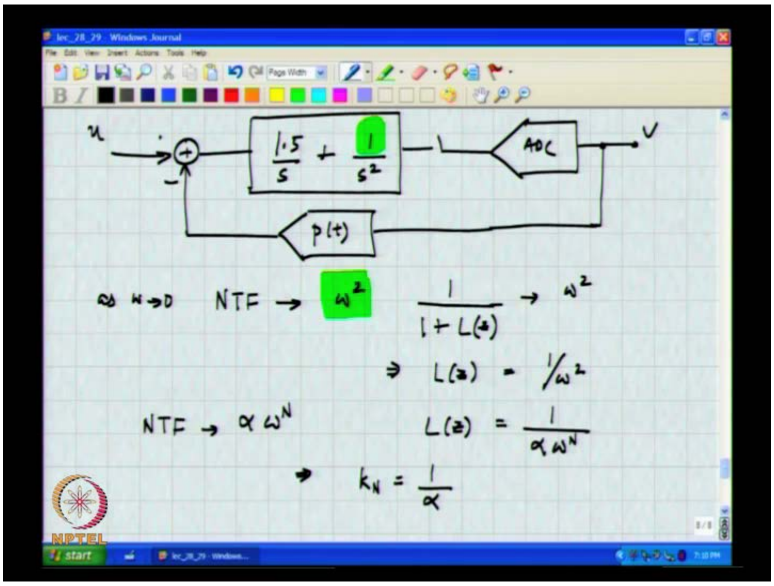
So, 1 by s; 1 by s; 1; 1.5; this denotes the DAC pulse shape. This is the ADC. And, if this… Please note that, if p of t is an NRZ pulse only; you will get 1 minus z inverse the whole square. If the pulse is changed, what should happen?

Student: We will get…

We will get different values for k 1 and k 2. Does it make sense? Because the coefficients are intimately tied with the pulse shape of the DAC. Now, let us see if we can get any more intuition from these coefficients. Can somebody comment on the value of k 2? Of course one can say it comes out of the math; but, is there intuition to k 2 being equal to 1?

Student: We have DC.

(Refer Slide Time: 42:13)



Let me just draw the loop filter as 1.5 by s plus 1 by s square is the loop filter. And, this is p of t. So, why does it make sense that, this coefficient is 1?

Student: No, in discrete, for DC is 1.

Sure?

Student: And, in the last lecture, we saw that, it is NTF into the loop filter is the effective STF. Then, because the NTF goes high as omega square…

Correct.

Student: …near DC and this L of loop filter transfer function should go as it is k 2 by omega square. So, k 2 by omega square into omega square is k 2. So, which should be 1. So, k 2 has to be 1.

That is right. So, the NTF at small frequencies, the NTF goes as omega square; correct? 1 minus z inverse the whole square. So, at small frequencies, it goes as omega square; which means that, 1 by 1 plus L of z at small frequencies must go as omega square; which means that, L of z must be proportional to or must go as 1 by omega square; correct? which means that, at low frequency, the magnitude of the loop filter must go as 1 by omega square; which means that, it must go as that, k 2 must be?

Student: 1

1; do you understand? And, that happens simply because in our particular NTF, we have chosen the NTF goes as omega square. In general, for an n-th order NTF, at low frequencies, NTF will go as omega to the power n divided by some alpha; where, alpha is greater than 1 or less than 1? Or, let me call this alpha times omega to the power n. And, why is this alpha coming?

Student: Because of the poles not being with the…

So, because we saw that, if we just make an NTF of the form 1 minus z inverse to the n, then the maximum stable range – maximum stable amplitude will be very small; or, stable range of the modulator will be very small due to overloading of the quantizer. So, we moved the poles of the denominator away from z equal to 0; thereby reducing the gain of the noise transfer function at omega equal to pi; correct? which means that, in the signal band, at low frequencies, the gain of the NTF is not omega to the power n, but something?

Student: More than 1

Something more than 1. So, this alpha is greater than 1. So, if the NTF goes as alpha times omega to the N, then L of z at low frequencies must go as?

Student: 1 by alpha omega…

1 by?

Student: Alpha omega…

Alpha times omega to the N; which means that, k N should be?

Student: Less than 1

No, it is of course less than 1, but is there an exact answer?

Student: Alpha

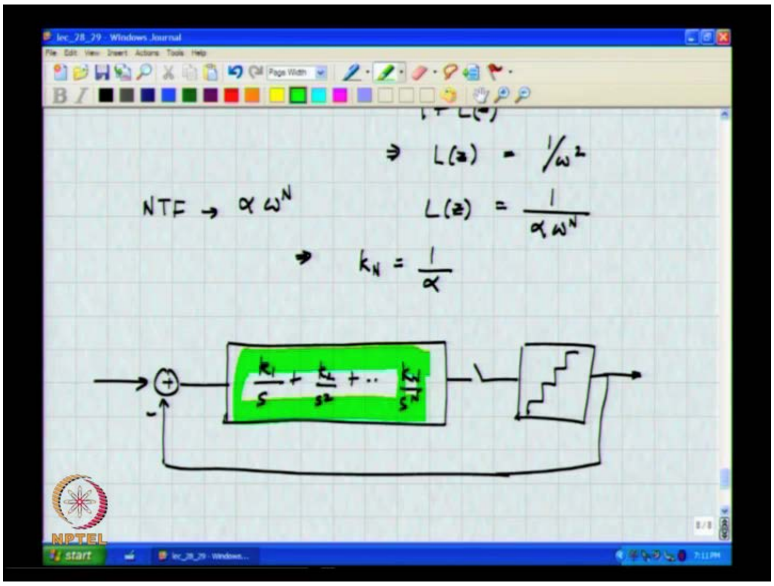
It is?

Student: Alpha

Student: 1 by alpha

1 by alpha; correct?

(Refer Slide Time: 47:11)



So, if somebody showed you a loop filter at let us say… Let me… You can immediately calculate the in-band quantization noise by looking at just the last coefficients, because at low frequencies… Please note that, quantization is getting shaped or pushed out of the signal band simply by virtue of the gain of the loop filter at within the signal band or at low frequencies. At low frequencies, which of these terms will contribute the largest to the gain of the loop filter?

Student: k at the k N by s power N.

The k at the k N by s power N; so, at low frequencies, the loop filter gain is k N by s power N; which means that, the noise transfer function will be of the form omega to the power N divided by?

Student: k N.

k N; and, you can integrate this within the signal band and estimate the in-band?

Student: Quantization noise

Quantization noise due to this noise transfer function. Is this clear? So, this therefore… That is why this k 2 equal to 1 makes sense, because in our case, the NTF at low frequencies is precisely omega square. So, the next thing to do is to say now, we have looked at first order and second order noise transfer function; what would you do with a higher order noise transfer function, where the denominator is not 1, but some d of z inverse. Again, in general, the loop filter will be of this form. And therefore, instead of having two equations, you will need to solve a system of n equations. One thing that nobody asked me was how come…

Student: System of ((Refer Slide Time: 50:17)) k 1, k 2 are solved in the sense…

One thing that we noticed was there are lot more equations than variables; correct? And, we seemed to have arbitrarily chosen the first two equations to solve k 1 and k 2. But, if you pay close attention, regardless of which two equations you choose…

Student: Same

…you get the?

Student: Same…

Same solution. So, one question is – is it some special property of this noise transfer function that this is happening; or, is it a more fundamental thing; and, will you be able to get the same solution for a higher order modulator? In other words, there are infinite number of equations; there are only a finite number of variables. So, in principle, choosing any choice of n equations will give you the coefficients k 1 through k N. One question that needs to be answered is can I arbitrarily choose the equations and will I get the same coefficients? If I do not get the same coefficients for an arbitrary choice of equations, then… As a designer, my confidence in my whole process is completely shattered, because I cannot keep having different coefficients if I choose different sets of equations; correct? So, we will have to figure this out and we will have to see how to go forward and find these coefficients for an arbitrary noise transfer function. Do you understand? So, you do this in the next class. Thank you.

Student: Sir

Yes

Student: These coefficients will also dependent upon what ((Refer Slide Time: 52:10))

Of course, absolutely; as I said, the coefficients you get are intimately linked to?

Student: Pulse

The pulse shape; without knowing the pulse shape, you cannot do anything at all. A very quick and dirty thing is if the pulse shape was impulsive, what coefficients will we get? If the pulse shape is impulsive, the no… Assume a small delay, so that you do not get… The first sample is 0. So, the 1 by s will give you 0, 1, 1, 1, 1; the 1 by s square will give you?

Student: 0, 1, 2, 3, 4

0, 1, 2, 3, 4; correct? So, what will you get?

Student: 1, 0. Some cannot… 1, 1…

You will get?

Student: 1, 1

1 and 1; k 1 and k 2 will both be 1. Correct? And, this makes intuitive sense. k 2 is still 1. That makes intuitive sense as we have just seen. But, k 1 being smaller than one and half, why does that make sense? See the first sample of the loop filter impulse response is coming from partly from the 1 by s term and partly from the 1 by s square term. Correct? If the DAC is an impulse, then the 1 by s square term will contribute?

Student: 1

1; which means that the contribution from the 1 by s term can be smaller; do you understand? You need to get 2; if you had an NRZ pulse shape, with k 2 equal to 1, the 1 by s square path is only contributing half. So, the 1 by s path has to contribute one and half to get 2. On the other hand, if the DAC was impulsive, the 1 by s square path will contribute 1; in which case, the 1 by s path has to only contribute a smaller amount, which is 1. Do you understand? So, as you can see, it is very straightforward to see that, the coefficients indeed depend on?

Student: Pulse shape

The pulse shape.