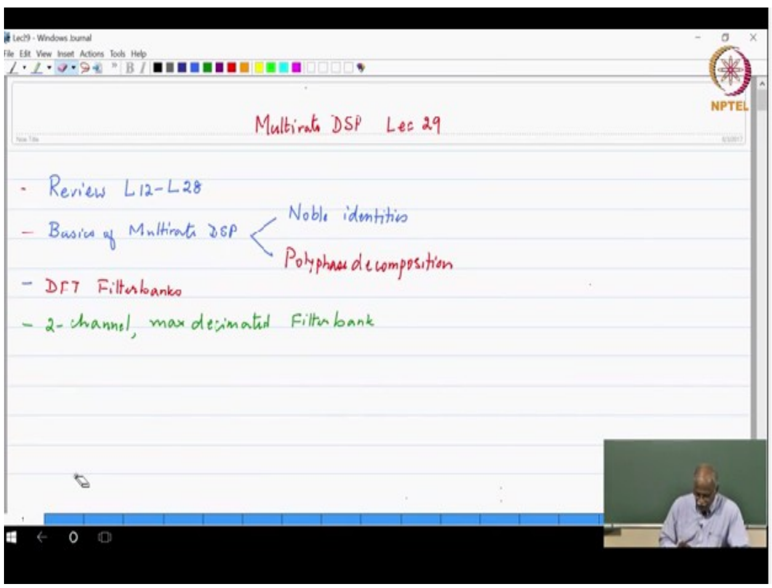
Multirate Digital Signal Processing

Prof. David Koilpillai
Department of Electrical Engineering
Indian Institute of Technology - Madras

Lecture – 29
Review of Lec 1-28

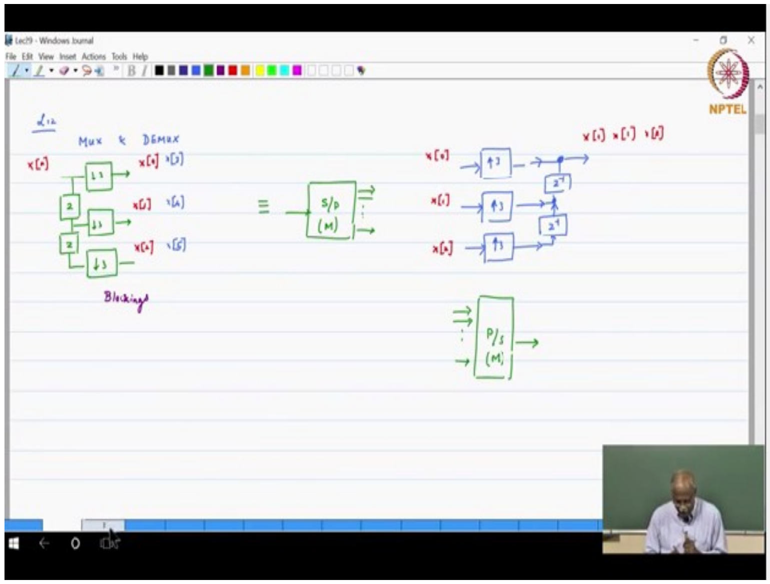
So let us do a quick run through. It is intended to just sort of refresh your memory. And if there are any doubts or clarifications, something in the notation, just ask. We can spend a few minutes. So roughly lecture 12 was where we were introducing the noble identities, started looking at polyphase decomposition, that is more or less start of, end of quiz 1 and the start of the portions after quiz 1.

(Refer Slide Time: 00:32)



Then we talk about the DFT filter banks and their properties. And then the 2 channel maximally decimated filter bank, the various solutions. Actually we have looked at 4 different solutions.

(Refer Slide Time: 00:54)



So the task for us now is to quickly review the items from lectures 12 onwards. So lecture 12, then lecture numbers may be slightly shifted but more or less this is the sequence in which we covered. Lecture number 12, we looked at the structure of a multiplexer and a demultiplexer, okay. And the form of the demultiplexer that is most convenient to us is the following form. Let me take the case of M=3, downsample by a factor of 3, advance operator downsample by a factor of 3.

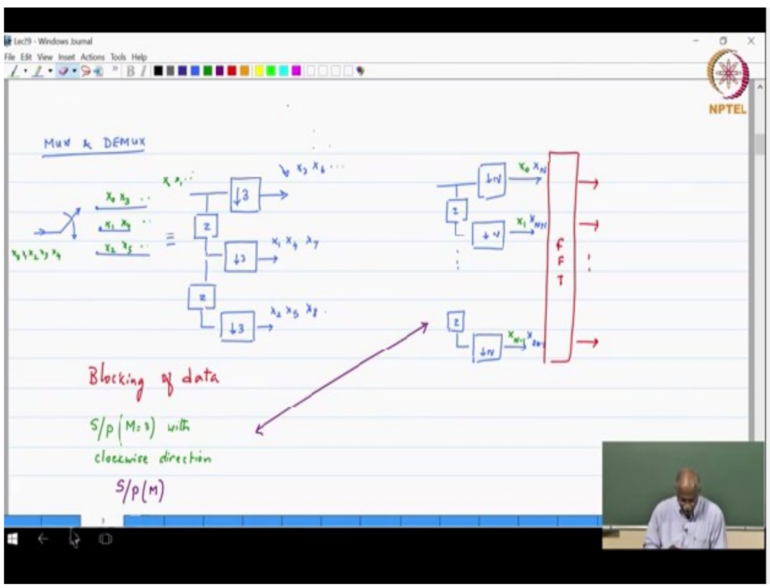
This is also downsample by a factor of 3, okay. So if you were to write down the, if this was X of n and X of 0 was going on the upper branch, then X of 1 will come on the lower branch, X of 2 will come on the second branch. Then the next time instant, it will be X of 3, X of 4, X of 5. This is what we also refer to as the blocking operation, creation of blocks. Notice that these are non-overlapping blocks if you downsample by a number less than 3, then you will get overlapping blocks.

And this is what we designated as a serial to parallel converter of dimension M. One in M out and this is a notation. So if you see serial to parallel with M, it is the one with the delays. Of course, you can have demultiplexer which works with delays. Then the outputs are slightly shuffled and again it is interesting to see what comes out if you replace the advance operators with the delay operator.

And then on the other side, if you want to look at a multiplexer, a multiplexer operation which is most convenient for us, again I will take the case M=3, 3 parallel streams connected or added with a delay operation. So this is a summing node upsample by a factor of 3, a delay and notice the arrows going upward. This is a summing node. So this is the form of the multiplexer that is most convenient.

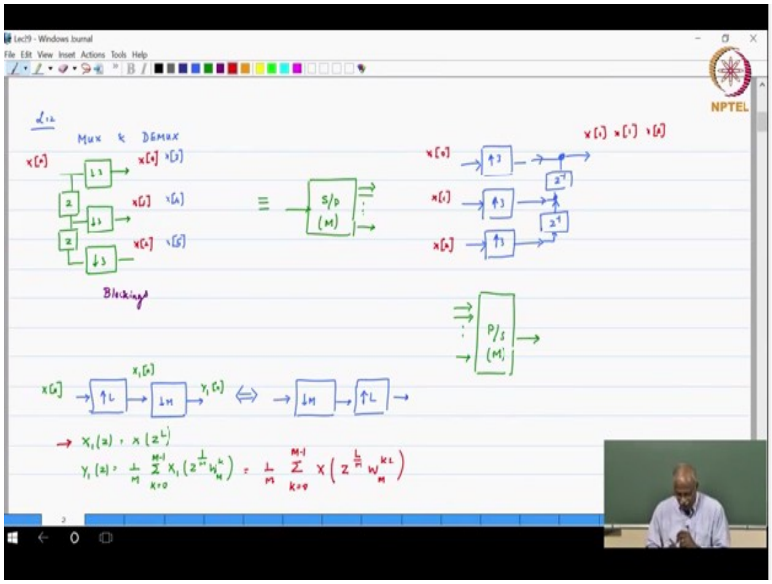
And again if you feed in X of 0, X of 1, X of 2 as a block, what would come out would be X of 0, X of 1, X of 2 which is the logical sequence in which we would expect the data to be present. And the notation for this is M inputs, we refer to this as a parallel to serial converter of dimension M. So again this is the standard definition. Of course, you can mix them up a little bit and get slightly different outputs, okay.

(Refer Slide Time: 04:37)



So if you were look at the movement of the demultiplexer switch, then what you find is that if you have this structure with the advance operators, then it is going in the clockwise direction. Otherwise, you will find that it is going in the anticlockwise direction. So this is a useful form because for example if you wanted to compute the FFT with non-overlapping blocks, this is what you would do. You would do a serial to parallel conversion, downsample by n, so that would give you non-overlapping sets of blocks and we have looked at that type of.

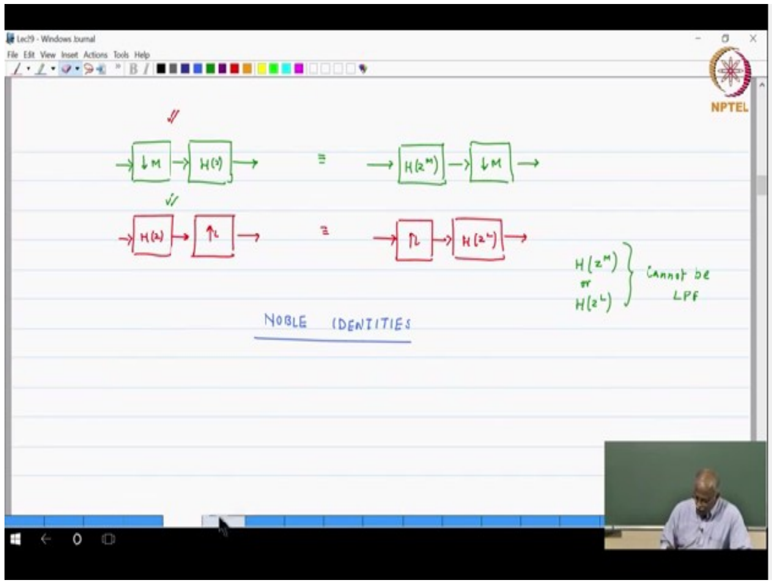
(Refer Slide Time: 05:13)



So let us move on. A couple of other results from lecture 12. We said that the interchanging of multirate blocks upsampling by a factor of L, downsampling by a factor of M, these are equivalent under certain conditions that L and M are prime. So more importantly, more than the result is the ability to show these results mathematically, okay. So let me just write one side of it. So if this is X of n, let me call this as X1 of n, this as Y1 of n. X1 of Z would be X of Z power L because it is the upsampler.

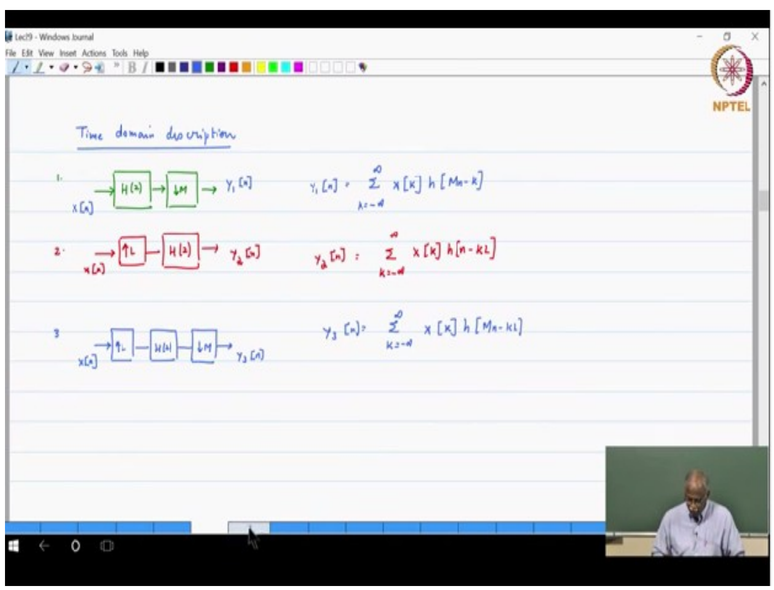
Y1 of Z is the downsampler. So it is 1/M summation K=0 to M-1 X1 of Z power 1/M WM raised to the power K, okay. W is the DFT twiddle factor, e power -J2pi/M. And this can be written as 1/M summation K=0 to M-1, substitute for X1 of Z from the previous equation. So this would be X of Z power L/M WMKL, okay. And similarly we get an expression on the right hand side and make an argument that they are the same, alright.

(Refer Slide Time: 07:14)



So now if we then move on to the next important result, that would be the noble identities. If you have downsampling, sorry, a filtering block, so H of Z power M followed by a downsampling by a factor of M, then you can interchange the two provided we replace H of Z power M with H of Z. And similarly the upsampling also can be replaced and the ones on the left hand side are the more efficient implementation.

(Refer Slide Time: 07:50)



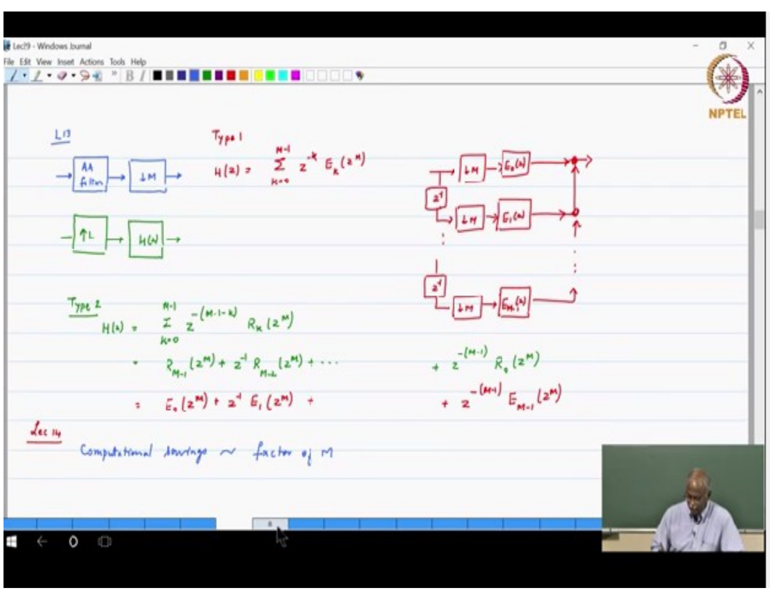
Along with this, we also looked at another important result which we referred to as the time domain representation of the filtering in the context of multirate blocks, time domain description. So this is not an LTI system because we have shown that the upsampler and the downsampler are time variant blocks. However, we are able to write the input and output in terms of something that looks like a convolution and the interesting results are presented here.

So if I have the anti-aliasing filter H of Z followed by a downsampler by a factor of M, okay, so if this was X of n and this is Y1 of n. We have shown that Y1 of n can be written as summation K=-infinity to infinity X of K h of Mn-K, okay. So make sure that you can get the same results. Second result, if we have X of n which is going through an upsampler by a factor of L followed by the interpolation filter, H of Z, Y2 of n.

In this case, we have shown that Y2 of n is summation K= -infinity to infinity X of K h of n-KL, okay. So make sure that we got the correct relationship. And the third one which is the combination of upsampling and downsampling, a fractional sampling rate conversion. So if I have upsampling by a factor of L, followed by filtering, followed by downsampling, again the choice of the filter that is sitting in the middle depends on the values of L and M, whichever one is larger will determine the value.

And in this case, if this is X of n, Y3 of n. Y3 of n we showed is summation K=-infinity to infinity X of K h of Mn-KL. Basically it is a combination of 1 and 2. So this is a way of representing the input output relationship, not strictly convolution but in the form of a convolution type operation, okay.

(Refer Slide Time: 11:03)



Move on to lecture number 13. I am sorry, yes, so again, lecture number 13 may have started a little bit earlier. But probably the important result is the combination of polyphase decomposition and the filtering which is associated with sampling rate change. So anti-aliasing filter followed by the downsampler, we cannot take advantage of the Noble identities except if you do polyphase decomposition.

So type 1 polyphase decomposition is says write H of Z as summation K=0 to M-1 Z power -K EK Z power M, okay. So apply it and then use the Noble identities. What we should get the equivalent structure for this should be downsampling by a factor of M, delay downsample by a factor of M, dot, dot, dot, a delay downsample by a factor of M. This would be E0 of Z E1 of Z, again all the arrows are to pointing to the right.

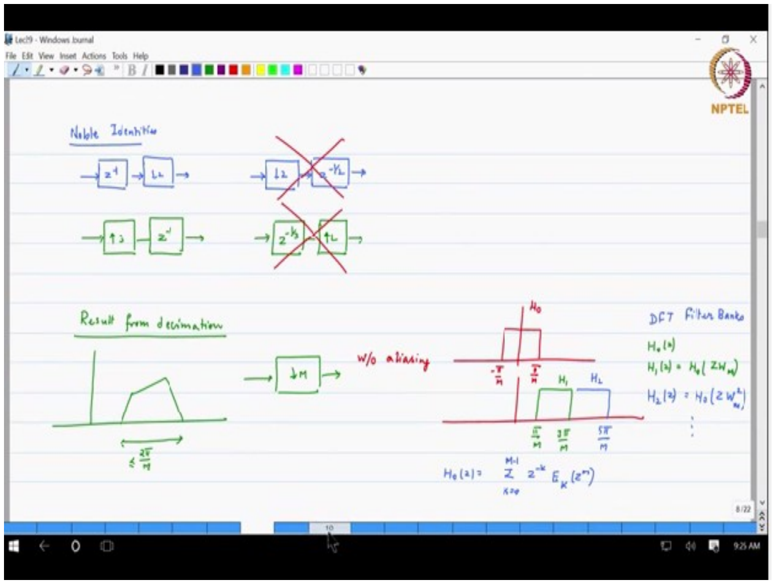
This would be EM-1 of Z and all these outputs get added together, okay. So that would be the way you take advantage for, and likewise the same principle also applies if you wanted to do upsampling by a factor of L, followed by the filtering to remove the images. Then you would do a polyphase decomposition with L polyphase components.

There is a type 2 polyphase decomposition which we have used not in the context of these sampling rate conversion but we have used it in the context of filter banks. So let me just introduce it here itself. So H of Z can also be written as summation K=0 to M-1 Z power -M-1-K*RK of Z power M.

So this is actually RM-1 of Z power M+Z inverse of RM-2 Z power M+, dot, dot, dot, all the way to Z power -M-1R0 of Z power M, okay. Now if I were to just write down the type 1 polyphase decomposition, this would be E0 of Z power M Z inverse E1 of Z power M, dot, dot, dot, Z power -M-1 EM-1 of Z power M, okay. So actually you can then map the equivalences between the polyphase components.

I think somewhere between lecture 13 and 14, we also looked at the computational savings. If you do this, so let me, most probably it was part of lecture 14. We showed that the computational savings if you do the polyphase part is a factor of M than over the case where you did it in the direct form without any, taking any advantage of the fact that you are downsampling. So computational savings. If I am doing for a factor of M, is approximately a factor of M, so huge advantages when you are sampling, making a sampling rate change of large magnitude, okay.

(Refer Slide Time: 15:44)



We also said that the application of the Noble identity cannot produce unimplementable blocks. So Noble identities have to be applied carefully. So for example, this is not permitted. Z inverse followed by a downsampling by a factor of 2, though mathematically is Z power, sorry, downsampling by a factor of 2 Z power -1/2, no, this is not allowed. So this is not allowed, okay. And likewise we cannot do it for the upsampling also.

So upsampling by a factor of 3 followed by Z inverse if I did the Noble identities brute force, then I would get Z power -1/3 upsample by a factor of L and this again is not permitted, okay. Because basically fractional delays are not realizable. Then we introduce the notion or started to introduce the notion of filter banks and there was an important result from decimation. Because these filter banks ideally would like to have maximal decimation.

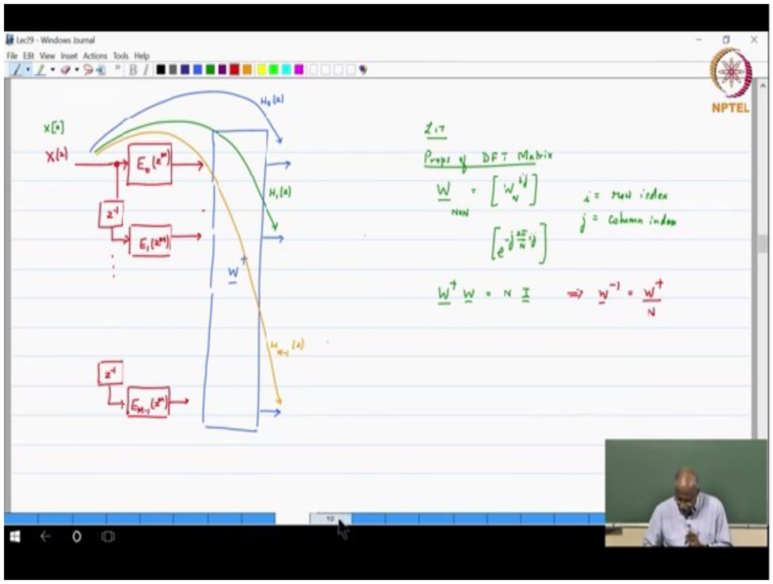
So when can you decimate a signal by a factor of M and not have any aliasing. Basically no aliasing present. So we showed that if you had a spectrum, some complex spectrum. If this width is 2pi/M or less, okay, less than or equal to 2pi/M, then this signal can be downsampled without aliasing. So basically the downsampled signal does not have any aliasing. Important result because what will happen as you will get production of M-1 copies, they will be separated by 2pi/M and because the bandwidth is actually less than 2pi/M, these images will not overlap with each other.

Now this was a very important result which took us to the notion of filter banks. And notion of filter bank starts from having a prototype filter and then producing other filters by shifting this one. So if this is H0, then H1 is here, H2 is here likewise and if these had pi/M as their boundary, as their cut off frequencies, so if this was -pi/M to pi/M, the green filter would have pi/M to 3pi/M.

Notice that both of these will not cause any aliasing. Similarly the blue filter also will not cause any aliasing when downsample by a factor of M. So this was an advantage and there is a particular name for filters that are, filter banks where filters are derived in this fashion. They are basically called the DFT filter banks. The reason for the name is that these are shifted by 2pi/M which is the same as a DFT twiddle factor.

So H0 of Z is your basic filter or prototype filter, H1 of Z is H0 of ZWM. And H, let me use the same colours, blue is H2, H2 of Z is H0 of ZWM squared and so on, okay. So you basically get M-1 filters. If you were to write it down in terms of the polyphase components, it is very attractive, H0 of Z type 1 polyphase components, K=0 to M-1 Z power -K EKZ power M and then apply the DFT filter bank constraints. We get the following result that the, where is my figure, okay.

(Refer Slide Time: 20:51)

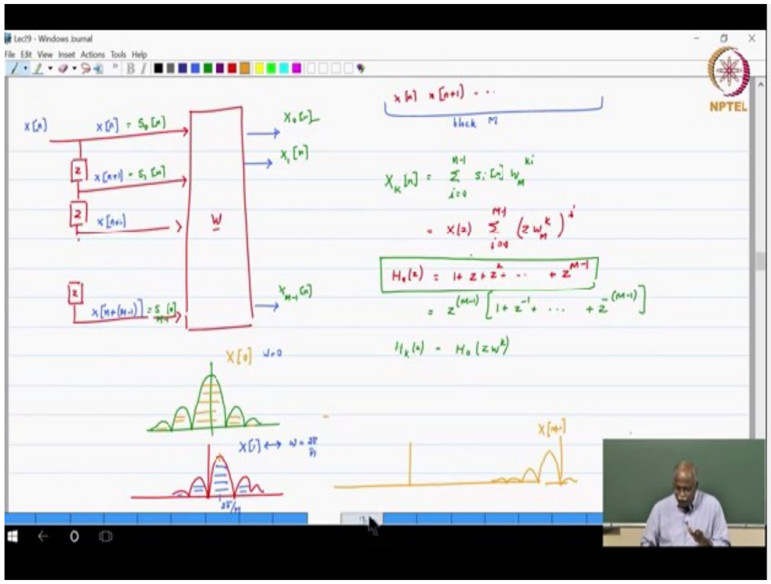


So this is what emerges from the, so if were to apply the polyphase decomposition and apply the DFT filter bank property, then what I would get is a delay chain followed by the polyphase components followed by the transpose conjugate of the DFT matrix. I believe lecture 17 was where, we were at this point. So lecture 17, we also looked at the properties of the DFT matrix belongs to the property or belongs to the family of unitary matrices with suitable normalization. So the notation is W.

We usually talk about an N*N DFT matrix. So the elements of this matrix are W subscript Ni,j. So I is the row index, j is the column index, okay. So it will be e power -j2pi/N*i*j, that would be your entry inside the matrices. And we have shown that through the properties of roots of unity that W dagger*W=N*the identity matrix which implies that W inverse is nothing but the W dagger/N which is precisely the definition of the inverse DFT, okay.

We said that the recognition of the filter banks and the computation of FFT actually have a very strong linkage and that linkage was established through the following analysis where we said that the computation of a DFT requires you to do a serial to parallel conversion, step I, followed by the DFT matrix.

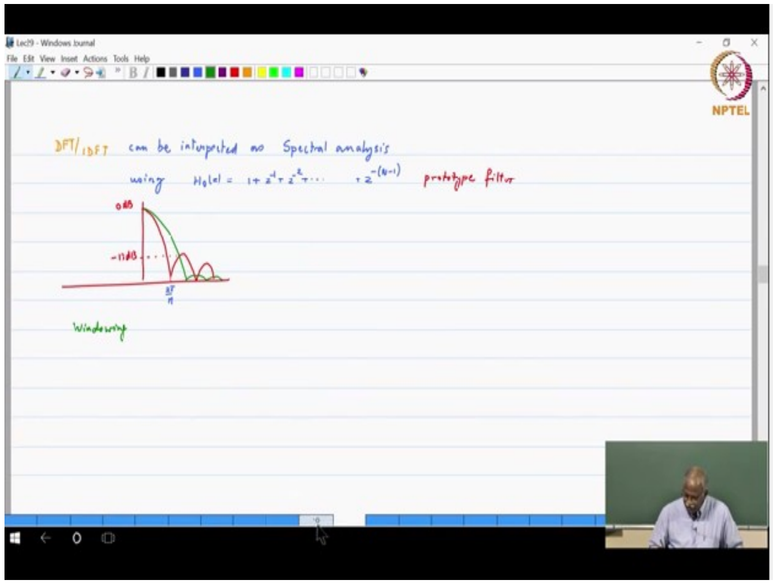
(Refer Slide Time: 23:03)



And then we computed the transfer functions between the input and each of the output. So basically if you were to think of it as the spectral analysis, then the DFT coefficient X0 is the output of A of X of n passing through a filter whose response is given by this expression, 1+, wait, wait, wait. Yes, 1+Z+Z squared + Z power M-1.

So basically if you pull out the appropriate factor, you can show that this is nothing but the rectangular window and likewise you can then show the transfer function between X of n and the second output to be a same filter shifted to the right by 2pi/M and this would be the X of M-1. So this would be X of M-1. So the last but 1 filter. So we say that spectral analysis using the DFT can also be interpreted as a DFT filter bank.

(Refer Slide Time: 24:35)

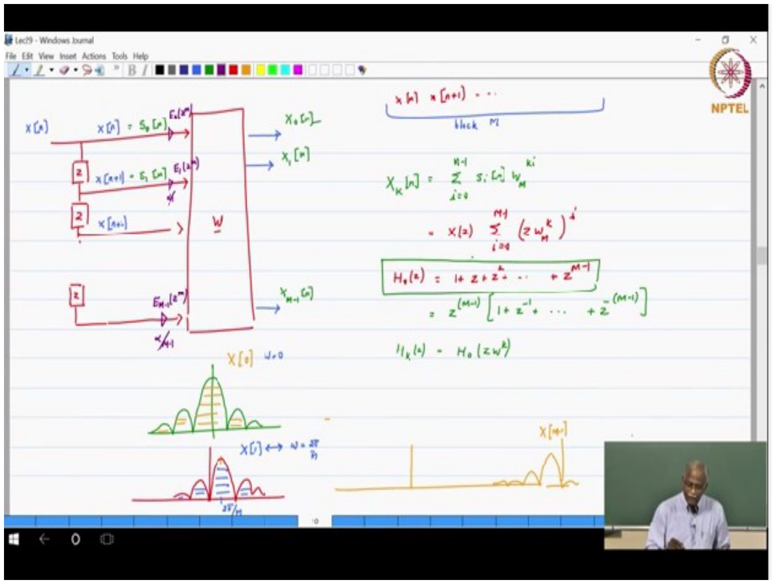


And may be a few statements in this regard would be helpful. Again, I am sure you will remember it once you look at the expression. So the DFT, iDFT, does not matter, both of them can be interpreted as filter banks operation or spectral analysis using the following prototype filter, using H0 of Z 1+Z inverse Z-2 Z power M-1, okay. So again it is a rectangular, this is the prototype filter.

And then all the other filters are DFT shifted versions of it. It is a very useful interpretation. Because this also says that my spectral analysis using the DFT is not a very precise one because the frequency response has got side lobes and the height of the side lobes if this was normalized to 0 dB, this will be at -13 dB and it lead to some erroneous interpretations of the spectrum. And the first 0 crossing happens at 2pi/M, okay.

Now if I wanted to get better spectral resolution in the sense that I do not want spectral leakage, then we said the way to do it would be to introduce windowing, okay, that would help us reduce the spectral leakage. The introduction of a window more or less would do something like this. It would make it a little wider but would make the side lobes much less. How would this actually look like in the DFT structure?

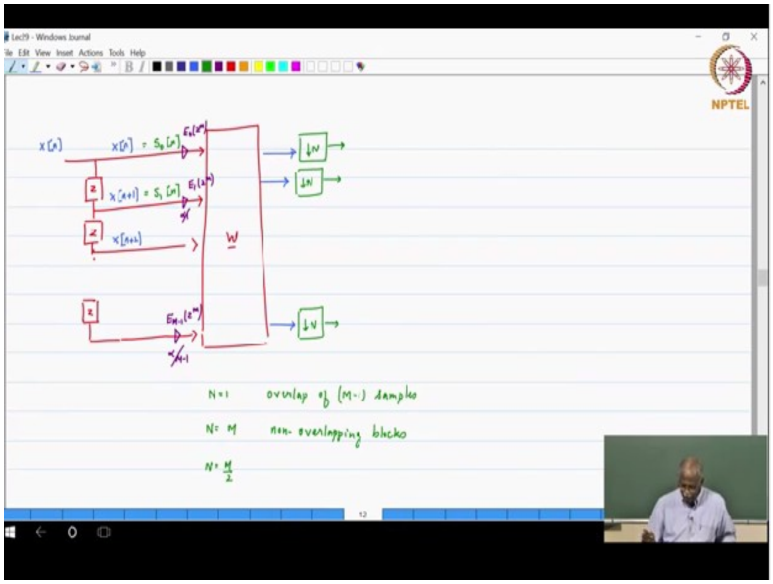
(Refer Slide Time: 26:51)



What would happen is, you would introduce a multiplier here. What colour would be a good one to use? So basically call this as alpha 0, alpha 1, all the way to alpha, have to erase something here, okay. And last one would be alpha n-, I am using M, sorry. M-1, okay. So the windowing helps you with avoiding spectral leakage. Now if you say that I want to have reduced spectral leakage and better spectral resolution, then the only way is to expand the size of the DFT.

And one way to do that without increasing the DFT matrix size itself is to replace these coefficients with the **(()) (28:00) – (()) (28:31) (Corrupted Audio)** that would be a way to do the spectral analysis. Again we mention that this is an important observation. Now in addition to this, there was one more result I believe which was highlighted. I would like to just use the same figure.

(Refer Slide Time: 28:49)

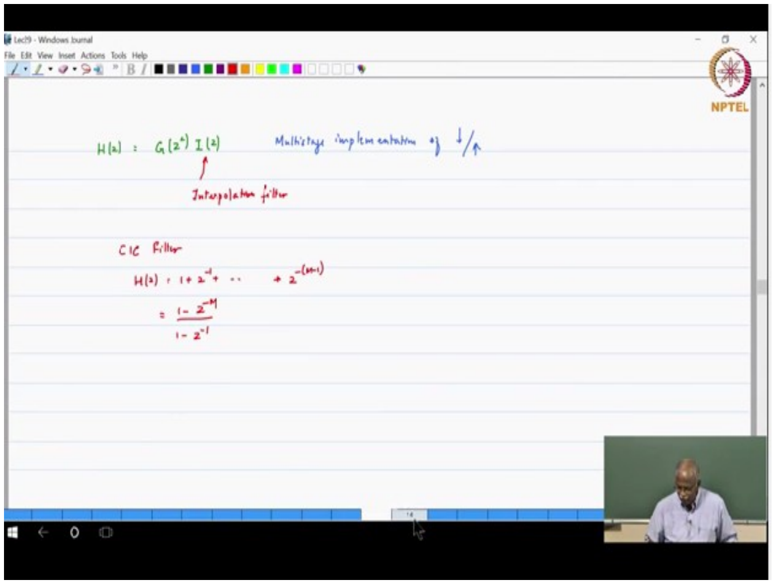


Maybe I will copy it and then, rather than just keep over-writing it, okay. So let us work with this one now. We can also look at the scenario where the output has a downsampler by a factor M, okay. If this is M, take this as N, okay. The role of the downsampler is to reduce the overlap between the subsequent blocks. So if you take N=1, then what you get is just a sliding block where you have an overlap of M-1 samples between successive DFT computations, okay.

If you take N=M, then you get non-overlapping, strictly non-overlapping. And of course anything between N=1 to M, would be various degrees of overlap between the, probably a popular one would be N=M/2, okay. So that will give you sort of 50% overlap, non-overlapping blocks, okay. So again the same structure, the interpretation is that we are trying to do a spectral analysis.

We think of it either as a DFT or we can think of it as a DFT filter bank where you have a prototype filter and then all of the other filters are part of the DFT filter bank. And you know how to improve the filter. You can think of it as a windowing. You can think of it as the polyphase components rather than as constants to be polynomials and therefore, get the advantage of the filters, okay.

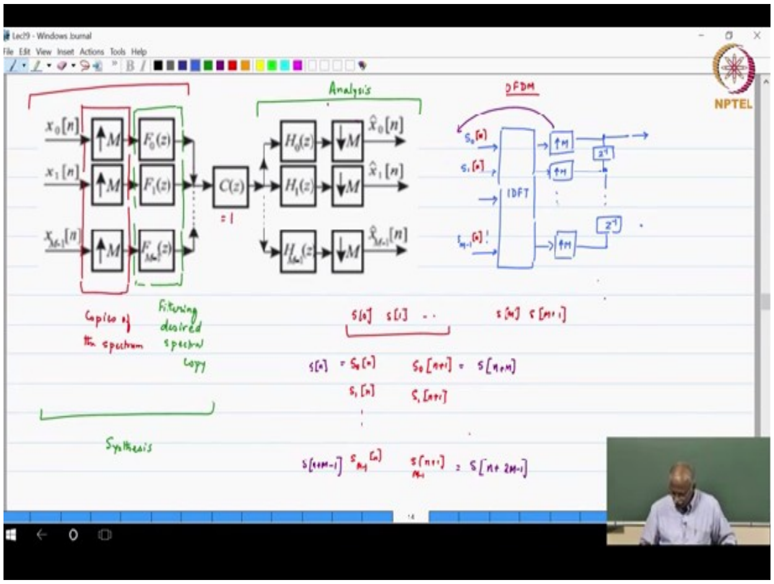
(Refer Slide Time: 31:07)



Let me move on quickly because the next step was the interpolated FIR design and the interpolated FIR design said that we could design filters taking advantage of the cascaded implementation. So H of Z can be implemented as G of Z square*I of Z. And this would be the interpolation filter which removes the unwanted images that are present in G of Z square, interpolation filter, okay. So basically the underlying principle is multistage implementation of the sampling rate conversion, implementation of either upsampling or downsampling. You can take advantage of this, okay.

A filter that is commonly used as interpolation filter is the cascaded integrated comb filter which is nothing but the rectangular window 1+Z inverse+Z power -M-1 1-Z power -M/1-Z inverse. You can apply it for upsampling or downsampling. You can cascade many of these filters and then still take advantage of their computational efficiencies, okay. So that pretty much summarizes the parts on the multirate part.

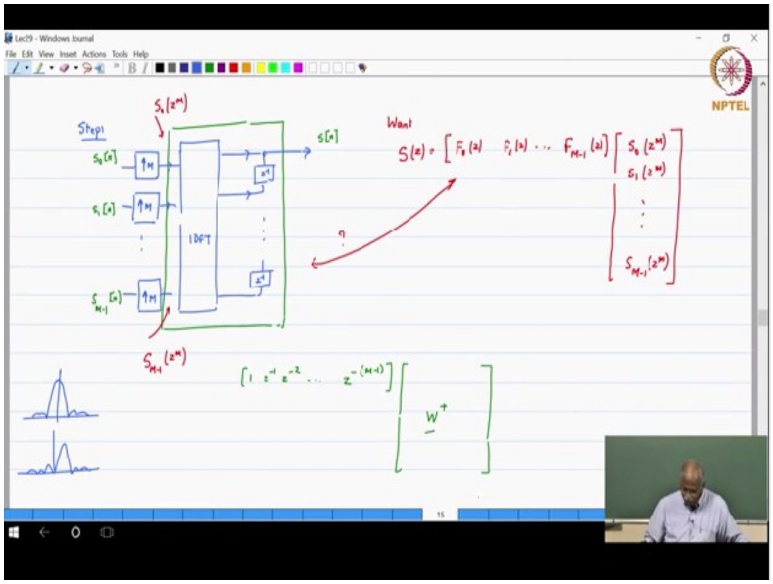
(Refer Slide Time: 32:49)



Now moving quickly into the filter bank portion of it. The first step was to look at the transmultiplexer. The transmultiplexer has the synthesis filters in the beginning followed by the analysis filters. We took the case where C of Z=1. So therefore, nothing is happening in the channel. What is happening is that upsampler produces copies of the spectrum.

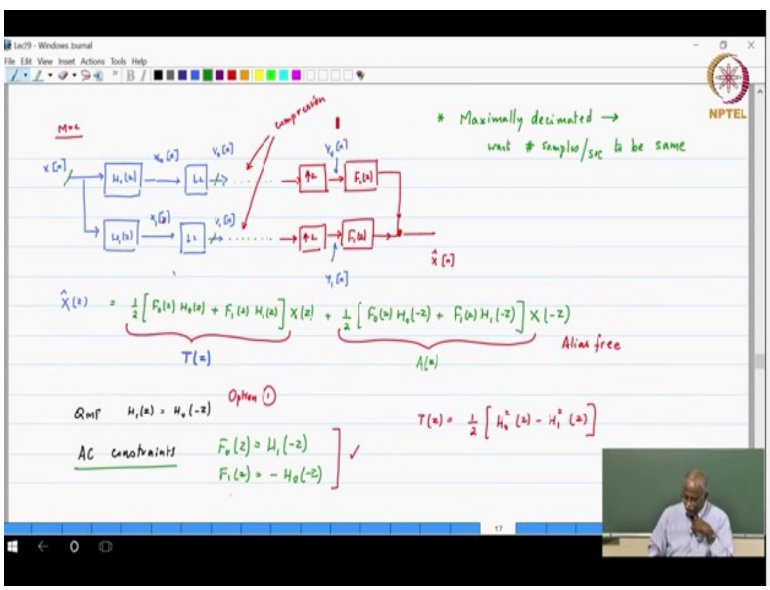
The synthesis filters are picking out the desired responses of the, desired copy of this, of the input signal. We showed the analogy to the OFDM transmitter. OFDM transmitter has an iDFT followed by upsampler followed by a delay chain. The upsampler followed by delay chain is nothing but a parallel to serial converter and then we wanted to analyze what the iDFT actually was doing.

(Refer Slide Time: 33:52)



And we showed that the iDFT is nothing but a filter bank which is exactly the prototype filter, the rectangular window. This second one, second filter is, basically you get a DFT filter bank from this. So the OFDM transmitter is doing exactly the transmultiplexer operation with these F0, F1, FM-1 being part of a DFT filter bank where the underlying prototype filter is the rectangular window, okay.

(Refer Slide Time: 34:31)



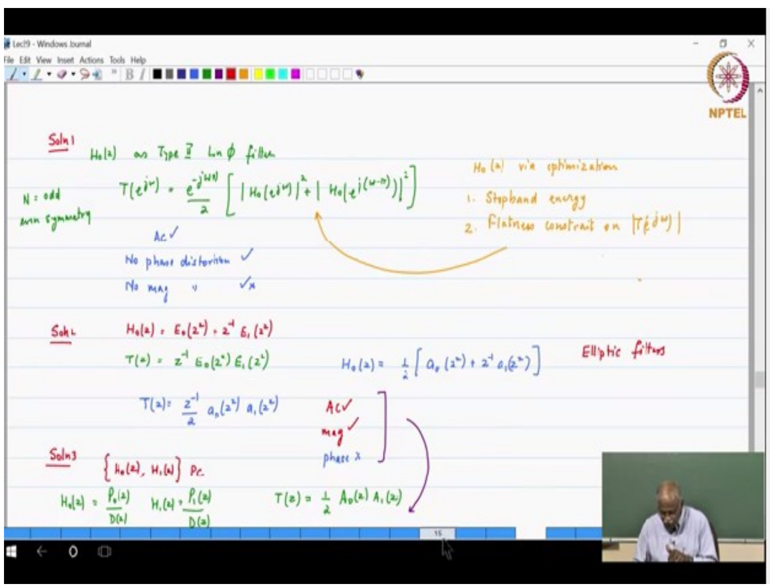
Then we move on to the special case of the 2-channel filter bank. The 2-channel filter bank we have H0 and H1 as the analysis filters, downsampling by a factor of 2, then followed by upsampling by a factor of 2, then the synthesis filters. Very important that we are able to analyze it in detail and get the expressions. We represent output X hat of Z in terms of a T of Z*X of Z.

And another term which is a transfer function for X of -Z.

Now X of -Z is the aliasing copy or the alias version of the input signal and the goal would be to get rid of any contribution of the X of -Z. So the alias cancellation constraints, more or less universally consistent would be F0 of Z=H1 of -Z. F0 should be a low pass filter. H1 is a high pass filter. So if you take -Z, is the shifted version which will become a low pass filter. So it is intuitively that is a sanity check.

Similarly, F1 of Z is H0 of -Z with the minus sign. The minus sign is deliberately to get rid of cancellation, okay. So given this, we are guaranteed that the T of Z can be written as along with the QMF of constraint can be written as H0 squared of Z-H1 squared of Z, okay. Now the rest of the exercise was to see how to get rid of the magnitude and phase distortions in T of Z.

(Refer Slide Time: 36:03)

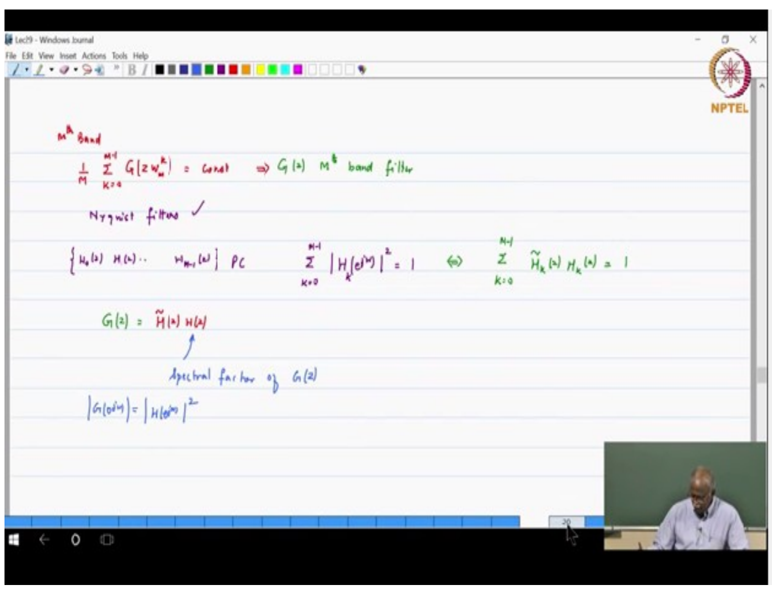


So the first step was to look at a type 2 linear phase filter. Type 2 linear phase filter has got even order, no odd order, yes. So even symmetric, okay. So it is N is odd, has even symmetric, yes. If you then write down the expressions, we get Te of j omega is magnitude H0e of j omega squared+H0e of j omega-pi whole squared. And we said that this could be implemented through the optimization of the stopband energy and a flatness constraint.

So that would give you complete elimination of phase distortion but the magnitude distortion could have some minor ripples, depends on the order of the filter and how well you have done the optimization. The second option was to look at the polyphase decomposition and then show that T of Z actually comes out to be Z inverse e0 of Z squared e1 of Z squared and if we could implement the filter H0 as the sum of 2 all pass functions, which we said was possible. We did not actually prove it but we said that for a class of elliptic filters with appropriate constraints, we could do this factorization.

The important things to know that it exists. Then we could get rid of aliasing, get rid of magnitude distortion but phase distortion would be present. Third method where we actually did the all pass decomposition of the filter, we showed that under very specific constraints of symmetry and order, we could get the transfer function to be A0, not A0 of Z squared. It is A0 of Z+A1 of Z. Sorry *A1 of Z. So that again was case where aliasing was cancelled, magnitude was removed but phase distortion was present, okay.

(Refer Slide Time: 38:41)



Then we came to the fourth element where we said that the QMF of constraint, we are going to apply it slightly differently. And for that, we wanted to go back and look at some other results that were available to us. And that was the property of the Mth band filter. Let me just quickly mention it so that we will have that result handy for us. The Mth band filter.

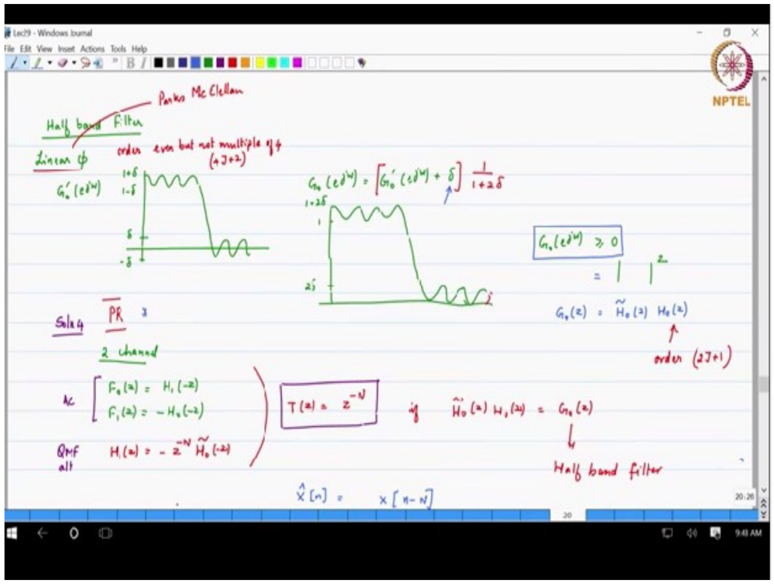
An Mth band filter has the following property that when you add the shifted versions of that filter. So 1/M summation K=0 to M-1 H of ZWMK, if this is equal to a constant, then we call H of Z as a Mth band filter, okay. So we showed that Nyquist filter satisfiez this property. Nyquist filters which are used in our interpolation are actually the ones will satisfy the Mth band property.

Now in addition to this, we also showed that if you had a set of filters which are defined as power complementary, H0 of Z H1 of Z HM-1 of Z as power complementary, this basically means that summation K=0 to M-1 mod HKe of j omega magnitude squared=constant or 1, okay. Now maybe I should have used a slightly different, let me call this as G. Then G of Z is the, the reason we show this side by side is if I have a Mth band filter.

So by the way this can be written as summation K=0 to M-1 HK~ of Z HK of Z=1, that is the para-conjugate representation. Now if we can represent or show that there is a Mth band filter G of Z which can be factorized in the form of H of Z*H of Z, okay, there is a specific name for this. This is called a spectral factor of H of G of Z. Spectral factor of G of Z because magnitude Ge of j omega=magnitude He of j omega magnitude squared.

So basically it is like taking the square root of the magnitude response but under certain conditions you can take it that basically means that your amplitude response must be strictly non-negative and we showed how by lifting the amplitude response, this is possible. Now take this to the case where we wanted to be just the half band filter. S half band filter factorized into this form basically gives us a new way of designing the 2-channel filter bank.

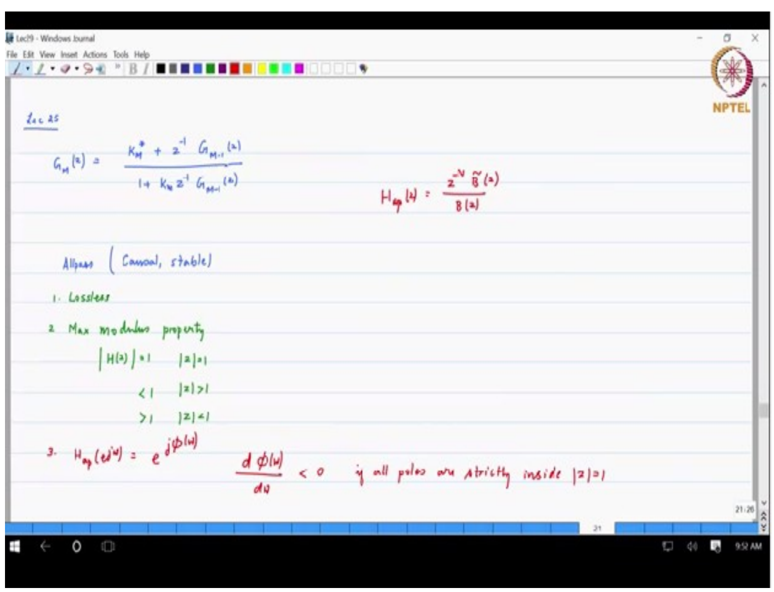
(Refer Slide Time: 42:00)



The aliasing cancellation constraint followed by an alternative way of implementing the quadrature symmetry. H1 of Z is the para-conjugate of H0 of -Z. So para-conjugate with the minus sign. If we, and we showed how to get out to design this filter by designing a half band filter and then lifting the amplitude spectrum. If we do this, then we showed that the overall transfer function comes out to be Z power -N or in other words, we have achieved perfect reconstruction. So this is the option that gives us perfect reconstruction.

Of all the 4, this would be the one that would be the most attractive because it achieves this first.

(Refer Slide Time: 42:54)



Now in the process when we talked about the all pass functions, we did derive the all pass lattice.

I believe that was lecture 25. I will just write that down as the last point. We showed that there is a lattice structure where the order of the filter GM, with M lattice stages, can be related to the transfer function with M-1 lattice stages in the following way, GM-1 of Z/1+KM*Z inverse GM-1 of Z.

And of course, a very interesting property is that if GM-1 is the causal stable all pass and mod KM<1, then GM-1 of Z will also be a causal stable all pass. And likewise if and only if relationship, so if you impose that constraint on GM-1, then automatically this one also becomes the same. So basically we showed that there are structures which will give us structurally stable all pass filters.

Now this was more in the context of showing that you can get all pass filters which we can show to satisfy the factorization of a filter. But again having studied perfect reconstruction, maybe these are limited in terms of their use in the filter bank context. But of course, all pass filters have got several other applications for which this would be a very useful tool as well. So the properties of an all pass, that is the last point I just want to mention, all pass.

So this would be causal stable all pass. They belong to the class of functions which are termed as lossless, input energy=output energy. If it passes through an all pass filter, there is a maximum modulus theorem. This is very important in proving the properties of the lattice, maximum modulus property. We know that the all pass has got magnitude=1 on the unit circle H of Z=1 on the unit circle.

The maximum modulus theorem derived from complex variable theory of analytic functions says that this is strictly less than 1 for mod Z>1 for mod Z<1. Again we have used this property without proof but it is standard result, the maximum modulus property of analytic functions. And of course, there is a third property which we did not explicitly used but they exist for all pass functions that if you have an all pass function.

Usually you write a transfer function in terms of magnitude and phase. Here the magnitude=1. So I can actually write it as e power j phi of omega and if you differentiate d phi of omega/d omega, d phi of omega is a real valued function, can be differentiated. We can show that strictly less than 0, that means it is a strictly decreasing function, monotoned decreasing function if all poles are inside, strictly inside the unit circle, okay.

Under this condition, inside mod Z=1, okay. And I believe that was the span of what we have covered between quiz 1 and quiz 2. And most general form of an all pass filter, Hap of Z is denominator polynomial Z power -N B~ of Z. This also says how will be the poles and 0's of an all pass function. Basically you will get the reciprocal conjugate as the, if you have a pole, you will get a reciprocal conjugate as the 0.

And because of, if you want to impose the condition of real coefficients, then you will get complex conjugate pairs. And of course, the reciprocal conjugate pair will also be present, okay. So this is just the material that we have covered in the class. Of course, if you have any doubts on assignment 4 or anything that we have covered in the lectures, please stop by between 5 and 6, TA's and I will be there. Thank you.