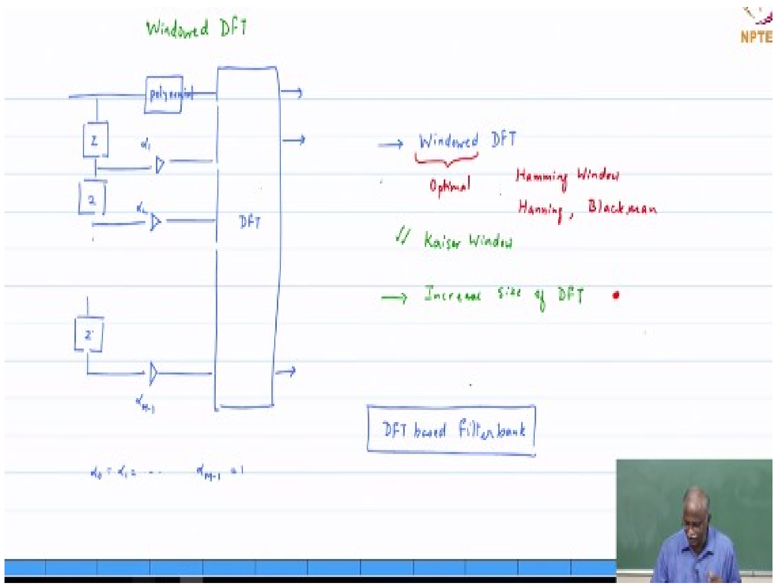
Multirate Digital Signal Processing
Prof. David Koilpillai
Department of Electrical Engineering
Indian Institute of Technology – Madras

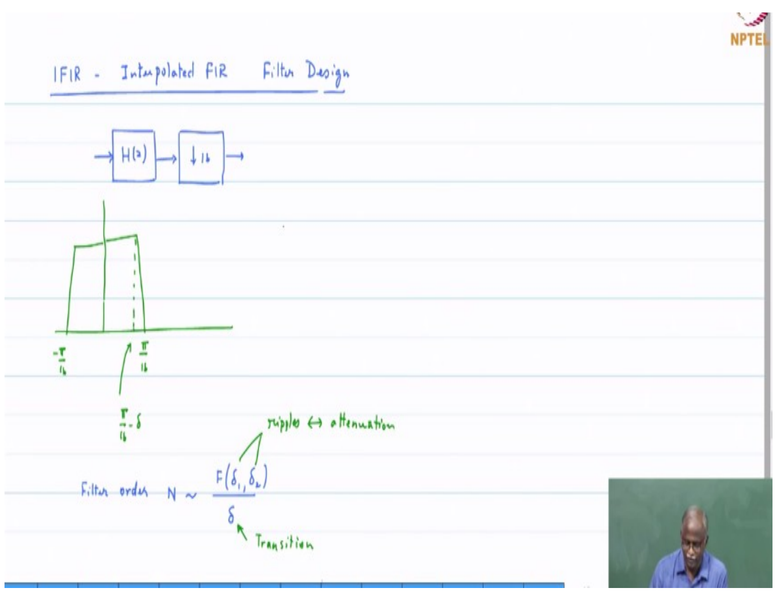
Lecture – 18 (Part-2)
DFT and High Resolution Spectral Analysis – Part2

(Refer Slide Time: 00:16)



Okay, now let us sort of put aside the DFT let us go back to some;

(Refer Slide Time: 00:23)



Underlying tools that we can develop in the context of multirate. Okay, this method or technique is called again the reason for this name will become very clear in a very short time. It is called IFIR Interpolated FIR. Interpolated Finite Impulse Response and this is in the context of filter design. IFIR Filter design and it turns out that this is a very innovative method of reducing the complexity of the design of filters.

Okay. So let us take an example and then justify what we have just now said. Supposing I had to down sample by a factor of 16, H of z this has to; this is an anti-aliasing filter for down sample by a factor of 16, okay. Now can you help me design what this filter looks like? I do not want any aliasing. So basically there is a passband; there has to be a stopband and the stopband is Pi/16. Am I right? And it is a real coefficient filter so it is from –pi/16 to pi/16. Okay

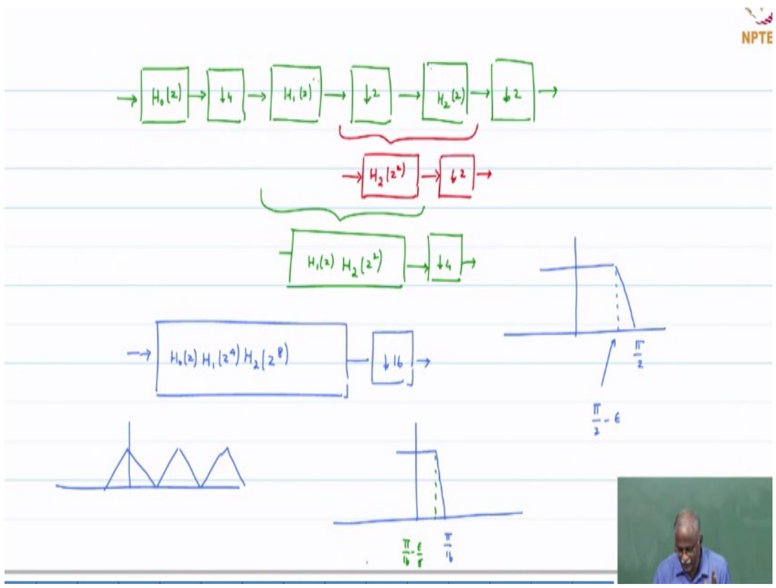
And so the passband is at pi/16 -delta some transition band delta is the width of the transition band, okay. And if you recall that the filter order needed to design a filter of this form the filter order depends let us call that as N, it is a function of the following; it is inversely proportional to the transition band; the narrower you want your transition band the more higher the filter order is going to be. So delta is actually going to sit in the denominator, okay.

And what is there in the numerator it is some function of what sort of ripple you can tolerate in this passband and stopband if delta 1 is the ripple of the passband and delta 2 is the ripple of the stopband. Let me just mention that these are the ripples; the smaller you want them higher is going to be the filter order and this is the transition band, okay. So this is well-known and therefore ripple can also be interpreted as attenuation.

Lower the; higher the attenuation lower the ripples will be, okay. So basically this is the form that we have. So pi/16 itself is a narrow filter. Now on top of that you have to have a transition band which means that this filter is probably going to be of very high order. So anytime you are going to have up sampling or down sampling by a large number you're going to have to design a very narrow filter with very tight specification and therefore the filter length will be; and I'm sure I mean several of you have discussed the computer assignment with me.

You said you design filter of length 100; some people said I designed even longer to basically make sure that your aliasing is done in a very compact manner. Now keep this picture in mind.

(Refer Slide Time: 04:13)



Now I am going to do the following; I want to down sample by a factor of 16, I am going to do the following method. I do not, I do an anti-aliasing filter H0 of z and then down sample by a factor of 4. Then pass through another anti-aliasing filter; we will call it H1 of z down sample by a factor of 2 and a 3rd filter H2 of z down sample by a factor of 2, okay. Again overall down sampling factor is 16; 4 times, 2 times 2, 16 so that that is not an issue.

But I want to know is there any difference any advantage; this is a cascaded down conversion in fact you also done it in your computer assignment. It is supposed to be exactly the same as doing a single step effectively the same. But let us see what the filters actually show up as. If you combine these two using the noble identities you will get H2 of z squared followed by down sampling by a factor of 2.

Basically a moving the down sampling factor all the way to the right just for understanding this. And so basically what you will then be able to do is combine these two and say that there is a filter effective filter which is H1 of z; H2 of z squared and then followed by a down sampling factor of 4, okay. So that is; this portion of the latter portion of the structure comes out. Now if I were to tell you draw the please draw the equivalent of the entire thing.

This is equivalent to a single filter which is a cascade of these three filters effectively moving the down sampling this becomes H0 of z; H1 of z power 4 and H2 of z power 8 followed by a down sampling by a factor of 16 and at first glance you may end up saying okay well whatever you called as H of z in the previous case is equal to H0 of z; H1 of z power 4 and H2 of z power 8, okay.

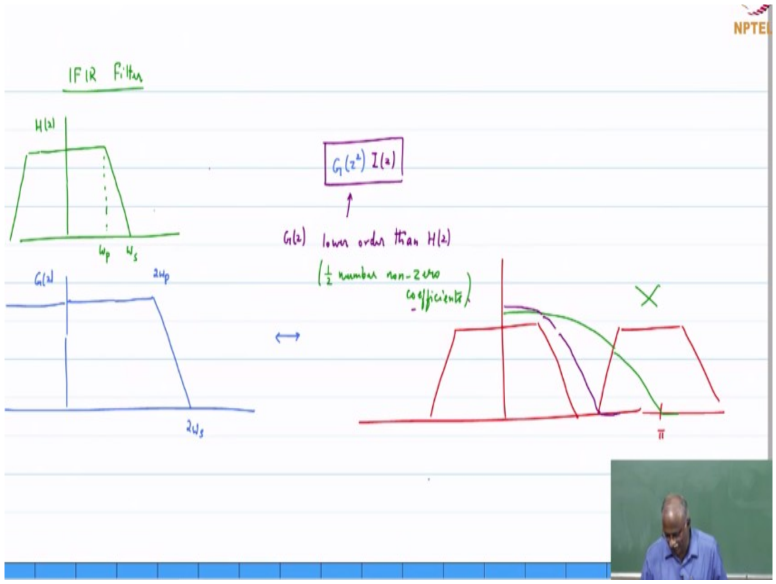
Now a very, very important element in this, what is the frequency response of H2 of z power 8? Something which; anything H2 to the power 8, so let me just not take; make it look like a filter. Anything which is z power 8, if this was the input spectrum originally and then you had z power 8 that means interpolation by a factor of 8 you will have 7 replicas of the spectrum. Am I correct? 7 replicas of the spectrum, okay. So here is where the key thing comes about.

So if I want to down sample by a factor of 2 my filter as many of you have already designed this will be pi/2, the stopband will be at pi/2. And let me call this as pi/2- Epsilon; pi/2- Epsilon. So that was what H2 of z would have looked like. But in the process of doing this cascade it became H2 z power 8. Can you tell me what H2 z power 8 will look, like just the first copy of the spectrum? Pi/16 and the transition band this will be pi/16- Epsilon by 8.

Did you have to do anything for it, came free, right. Okay. All you did was; you know move the decimators around. And you see that oh wow, okay. Now what is the price that you paid? It came along with spectral images. Now how do you kill the spectral images that is what the other filters are therefore. So the crux of the matter is whenever there is interpolation or a form of interpolation which when you get z power 4 or a power 8 then automatically there is a compression of the spectrum.

And you can actually take advantage of it to design high very narrow filters, okay. So once you have this principle in mind.

(Refer Slide Time: 09:24)



Then the filter design people said, okay we will now develop something called the Interpolated FIR filter. And so the question; let us say that the filter design specification where as follows. I had a filter which had a passband Omega P and a fairly tight transition band to Omega S. Now we know that the filter design is going to be inversely proportional to the transition band. So let us say that this is H of z this is my requirement, okay and symmetric real coefficient filter.

Now interpolated technique says, "Hey, do not design H of z", okay. Let us do something we just now learnt a trick in multirate. We are going to design not H of z but something slightly different. I am going to design a low-pass filter with two times Omega P as the passband; 2 times Omega S as the stopband and so this filter order is going to be less than the original filter order. And I am going to do the following; let us call this is G of z, I am out of space.

I cannot draw the; so what I am going to do is I am going to interpolate this by a factor of 2, G of z squared then when this gets compressed when this gets compressed then I automatically get a filter of the form H of z except that; so this when I compress will give me the following. It is going to be the filter that I am looking for plus it will have a component at pi, right because of the compression of the; so this is pi. Okay. And of course it repeats with the 2pi as well.

So now I need to get rid of this image. This one has to be eliminated. So then what you do is, okay design a filter that will basically kill this. You do not want any trace of the any trace of this so maybe the right way to design would be something that is flat in this portion and then goes down to 0 so that eliminate this completely, so again different forms are possible to get rid of that.

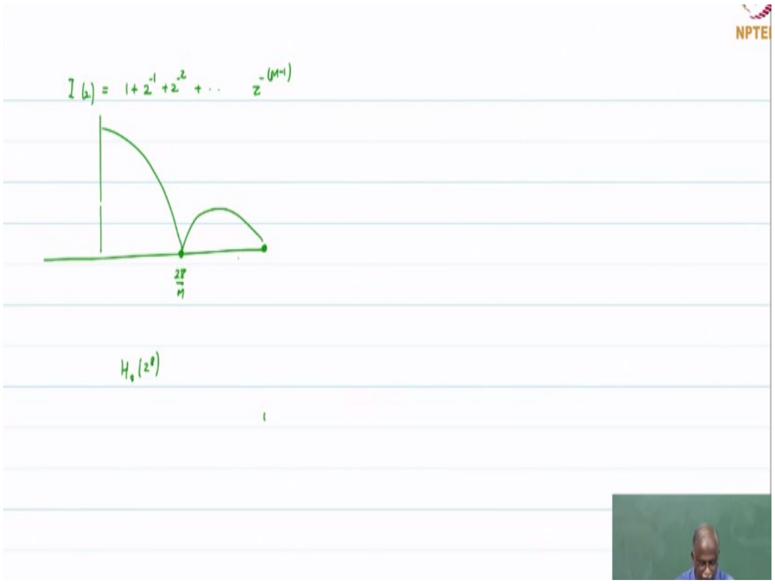
So we are going to call this filter that removes the image as I of z. Why I of z? It is an interpolation filter. Basically, when you did the interpolation it you got some unwanted images up sampling and you would use a filter to get rid of the images and this is the method that we want to call as the interpolated FIR technique, okay. Now what is the advantages, this one had lower order than H of z. So it has a lower order G of z it has got lower order.

But G of z squared that means the order becomes double right when I write G of z squared the polynomial order so G of z has got lower order than H of z but G of z squared I cannot make an argument but the advantages it has got only half the number of non-zero coefficients. So it has got only half the number of non-zero coefficients because G of z squared means every other coefficient is 0. Half the number of non-zero coefficients.

So in terms of non-zero coefficients because those are the ones that you would actually have to implement as multipliers from a hardware perspective that is going to be that is going to be the key factor, right. So what have we done we have intern obtained a tool from multirate signal processing which says that when I do global identities I can move the blocks around and actually I get an advantage because my spectrum gets compressed whenever I move them you know, when I get a filter of the form H of z raised to some power L.

But this notion of saying that I have got a lower complexity filter people may not buy it and the reason is it is not just G of z squared there is one I of z also coming and say you come on at the end of the day you design two filters and; did you really reduce the complexity. And DSP people are quite smart, they said okay wait, wait, wait. There is there is; what does this filter need to do? It needs to kill the image, right it needs to kill this image.

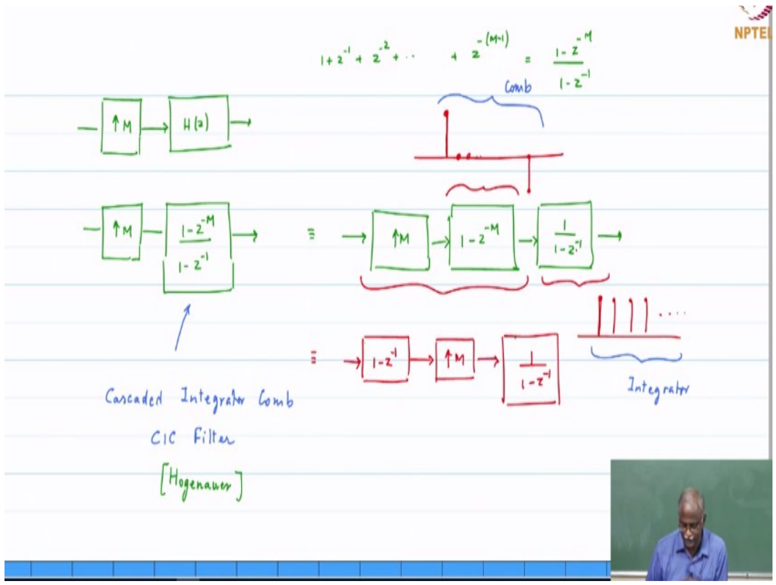
(Refer Slide Time: 14:26)



And it turns out that DSP we in fact even in just in today's class we have studied that this filter H of z let us call this I of z; if I chose it to be 1+ z inverse + z-2… z power M-1 okay. So this is a multiplierless filter, so very low complexity. And what is it good for it will have a 0 at 2pi/M Okay. Now I can design this filter such that I get a 0 exactly where I wanted to kill the response, right. So that is one observation.

So let us say that I did H0 of z power 8; H naught of z power 8 that means I will get 7 images and I will choose M=8 and I will basically kill all of the unwanted images by placing an exact 0 at those frequencies. Am I right? Are you satisfied with the design? Wait a minute. Wait a minute. Now this is not a does not have a very good stopband, so you know you do not by the fact that it is a good filter.

(Refer Slide Time: 15:53)



But before we get to that let us make the let us make the following observation. So supposing I am doing an interpolation, interpolation by a factor of L and I have to do and post filter H of z, right. This is similar to the problem that we are trying to address right now. Because when you move it around you will basically get H of z power 8.

Now the key point is when you have a structure of this form the filter that we just now talked about; this is a rectangular window has got a lot of advantages. And how is the advantages highlighted, it is in the following way. So if you have something of this form 1+z inverse + z-2 + …+ z power m-1. If you do a geometric series what is this? 1-z power -1 divided by 1-z inverse. So this filter, okay I am going to change this to M, so you do not have to change the others.

So I have up sampling by a factor of M and I have to do the following filter; 1-z power –M divided by 1-z inverse, okay. I am going to split this; this is identically equal to up sampling by a factor of M followed by 1-z power –M followed by 1- 1-z inverse. This is a cascade form, perfectly allowed LTI systems. Now I combine these two to get a very convenient structure which is given by 1-z inverse up sample by a factor of M followed by 1/1-z inverse, okay.

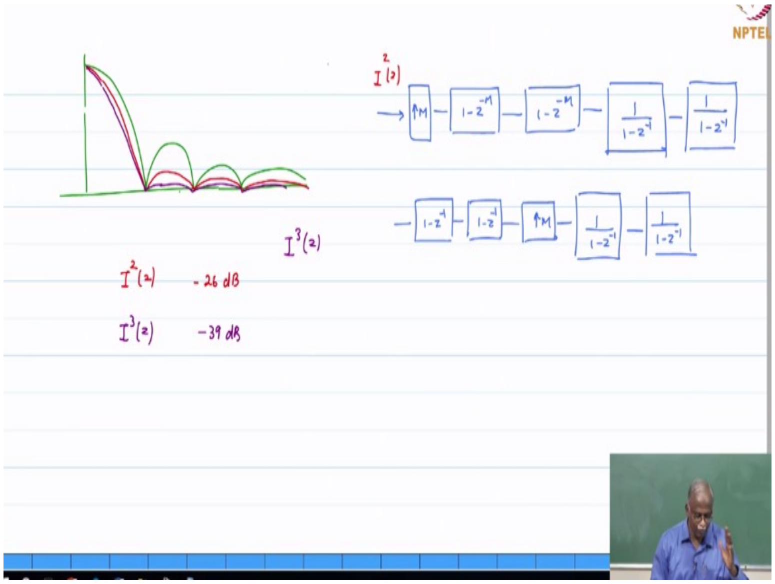
And of course you can see that we can do the same thing if you are doing the down sampling it will be on the other side; you can move these things around. Okay. Now if were to ask if somebody were to ask you what is the impulse response of this block? What would it be? You will get a 1 at time index 0; you will get 0 valued samples and then at M you get -1. That is the impulse response of this block.

What is the impulse response of this block? It is a 1 1 1 1… okay and this is a integrator. Basically adds all the past samples, so this is what you would call as a integrator. Whenever you have some non-zero samples and then interspersed with 0 valued samples we tend to refer to them as some form of a comb, okay you may have both of them positive; one of them shift around does not matter this is a comb.

And so this particular filter is called it is got a very popular name it is called a Cascaded Integrator Comb filter CIC Filter very, very popular ASIC designers will all the time talk about CIC Filters. Why are they so happy about it? It is multiplierless. It is got some very interesting properties especially when you want to do sampling rate change. It has also got another name within the DSP community based on one of the people who contributed a lot to the understanding of it they also call Hogenauer filters.

But CIC would be a more descriptive name that that you can use. Now put all the pieces together. This multiplierless filter can be interpreted as a finite length rectangular window or it can be interpreted as a cascade of a comb and an integrator. Now it turns out that the integrator comb interpretation is useful for us when there is a sampling rate conversion that is also present in the system. Now very important element that comes about in this context.

(Refer Slide Time: 21:18)



They said, okay I am; it is good that you have this type of filter multiplierless filter it has but the only problem that I have with this it is stopband attenuation. It is got this problem, okay. And I cannot really use it for either anti-aliasing or for interpolation because the side lobes are too high I need to improve the side lobes. Then the ASIC people say, "Hey, not a problem. We can solve that problem for you." You had I of z, correct? What about if you had I squared of z?

What is the frequency response? It will have the 0, 0 zero crossings will not change but when I square this is the number less than 1 so therefore it becomes smaller, I squared of z. Does it still have the advantages that we saw for a sampling rate conversion actually it does because I can write this in the following form 1-z power –M cascaded with 1-z power –M cascaded with 1/1-z inverse; 1/1-z inverse and up sampling by a factor of M.

I can move it past the first block; I can move it past the second block. This can actually be written as 1-z inverse; 1-z inverse up sample by a factor of M and then have the two integrators cascaded with each other 1- z inverse; 1-z inverse. Okay, so the CIC filters are actually quite powerful because you can; if you tell me that reduce the attenuation even more not a problem I can do I cubed, I cubed of z, I cubed of z you will find that there is a faster droop.

But the advantage is that your response goes down, okay. So I squared of z, what is the stopband attenuation? It was 13 dB to begin with now it will become 26 dB, right -26 dB basically it is a square of that into 2. So now when you have three stages I cubed, I cubed of z you have -39 dB maybe this is already enough or you want to go to I4 not a problem it can easily be at accommodate that, okay.

So for interpolation looks like we may have a good option. Now is this also good for anti-aliasing? Passband ripple is an issue. The what is where is the first 0 coming? 2pi/M. But if you wanted to do down sampling by a factor of M what should you do, it should be at pi/M. So what do I do? Well, if I multiply two of them I will still get the same the 0s will be; take a 2M length filter, yeah.

"Professor to Student conversation starts" Phase problem? See again these are effectively linear face filters because all the coefficient are equal to 1. So anytime you square them the face is still linear face but with a different slope. So these are actually filters which are very well behaved in terms of phase as well. So they are, in other words con constant group delay filters which give you multiplierless responses and very good performance. **"Professor to student conversation ends"**.

So we will talk a little bit more about CIC filters but the important thing to recognize is that this is; where did all of this come from, because you wanted to do something of this form where you wanted to design a cascaded integrated filter and so you wanted to do G's of z squared I of z. You wanted to reduce the overall complexity. So we said we will take complete advantage of the multirate part but then you needed an interpolation filter that will kill the other the response.

And if that is going to be a filter with a lot of non-zero coefficients it is not going to help, so therefore I of z has to be taken into account. Last point, very quickly before we finish. There is a droop in the passband. Am I right? The droop in the passband. Now how do I, how do you take care of that? I am going to use a CIC filter as my integrated filter. How do I take care of this? I will design this not in this form.

Basically when you specify the response magnitude response that you need for the low-pass filter you can specify it with a certain deviation or basically you can specify the magnitude response.

This is very easy to incorporate if you use parks mcclellan method or any of the other FIR filter design methods. So you basically, you know what the droop is going to be you pre-compensate it in your design of your filter and then you can still take advantage of that, okay.

Let me stop here and pick it up from there in the next class. So basically the two channel Maximally Decimated Filter Bank is what we will pick up in the next class. Thank you.