Signal Processing Techniques and Its Applications

Dr. Shyamal Kumar Das Mandal
Advanced Technology Development Centre
Indian Institute of Technology, Kharagpur

Lecture - 35
Digital Filter

So last week, we talked about the first Fourier transform and how that efficient calculation of discrete Fourier transform using divide concur methods radix 2 algorithm. So, this week, let us start with another topic, which is digital filters. So, as you know, we want to implement the filter in the digital domain.

So, the input of the input signal of the filter is digital, the output signal is also digital, and the filter is implemented digitally. So, as you are aware, there are active and passive filters, which are all designed here using the analogue domain. I think if you have completed your analogue electronics course, you have already designed an active low pass filter and an active high pass filter using op-amp circuits.

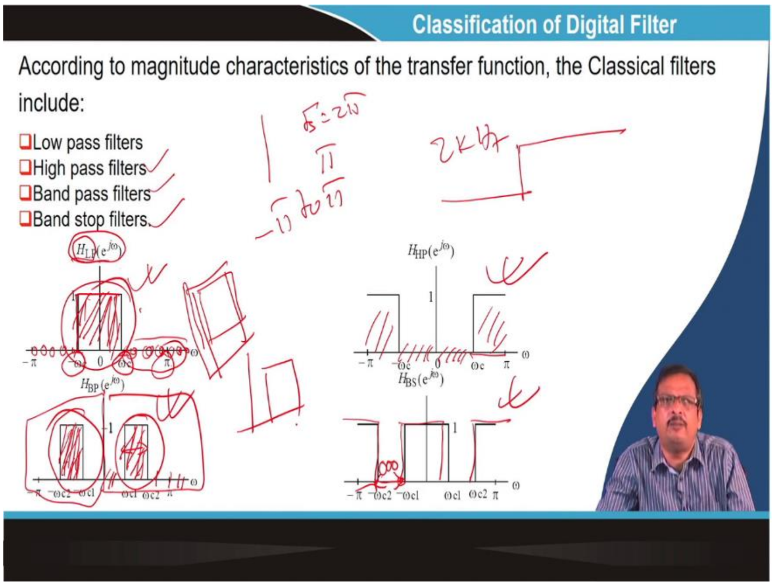
So, those are analog filters. So, the input signal is an analogue signal; it passes through this circuit and then produces an output signal, which is also an analogue. Here we talk about the digital filter. As you know, why digital filter, why not analog filter; you know that the digital signal process advantage is that it is nothing but an algorithm.

So, when I talk about the digital filter, it also is nothing but an algorithm; instead, you can say the component, and then there will be an active device and a passive device. instead of that, here I have an already digital signal. I pass through an algorithm, which will act as a digital filter, and the output will be in digital. Then, at the output, we can apply that analogue to the digital-to-analogue conversion, produce the analogue signal, and feed it to that natural system.

Let us think about that as your equalizer. Suppose you have a sound system, and you want to design an equalizer whose job is to emphasize certain frequencies and may be attenuated by certain frequency components, which is nothing but a filter. So, you can say that hearing aids emphasize a particular frequency band. So, we required a filter, but that filter is a digital filter.

So, the signal is digital, the filter is nothing but an algorithm, and the output is also a digital signal. So, when you talk about the digital filter, then it is an algorithm. So, how do I implement that algorithm? So, there may be a different kind of digital filters.

(Refer Slide Time: 02:44)



So, let us use the normal filter if I am talking about the normal filter. So, what is there? According to the magnitude characteristics of the transfer function, as you know, the function and the classification of the filters are different kinds of filter; maybe the low pass filter, there is a low pass filter, there is a high pass filter, there is a band pass filter, band stop filter.

What is a low pass filter? So, it will allow this frequency only; elsewhere than the attenuation, the amplitude of the magnitude response is 0. So, the signal will exist only in those components. And so, this is not a signal. This is a frequency characteristic, s, that is why H L P $e^{j\omega}$. So, it is nothing but a frequency magnitude plot, or I can use the spectrum of the digital filter.

So, this is the spectrum, where that c filter only exists minus ω c 2 plus ω c; ω c is the cut-off frequency. And you know that digital frequency, π is normalized digital frequency, which is nothing but; because you know that Fs is equal to 2 π, so the maximum frequency of possible in the digital signal is π, maximum oscillation -π to π.

So, if I define the magnitude response of the filter, it is nothing but a ω c and minus ω c. This is called a low-pass filter because low frequencies are only passed and high frequencies are stopped, with magnitudes of 0. So, high-frequency stops and low-frequency passes. That is why it is called a low-pass filter. Similarly, there is a high pass filter unless it is a bandpass filter. So, in the bandpass filter, if you see the particular band of frequency is passed, you know that if I say frequency transform, there is a negative axis and a positive axis.

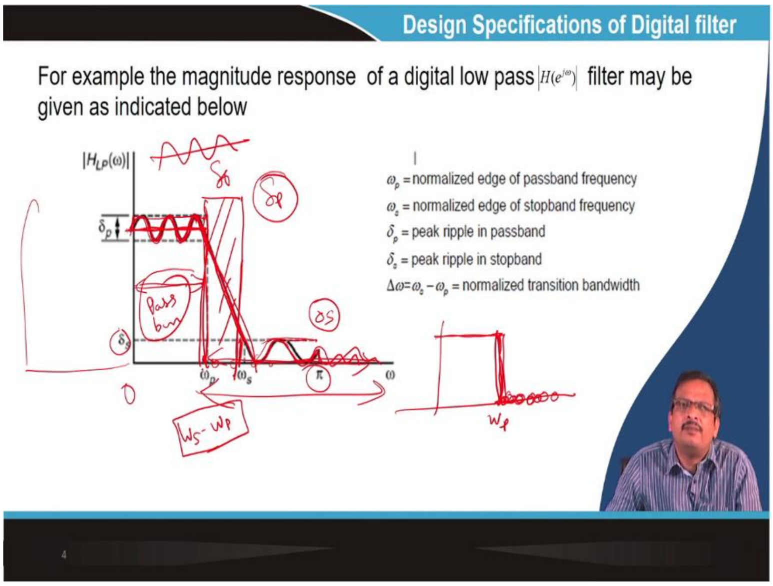
In a real application, I may only look into this side. This side I may agree, but in theory, there will be both sides minus ω c plus ω c. Similarly, here, there will be a band and minus axis and also, because the frequency response is -π 2 π. So, that is why I can say this band of frequency will pass; the frequency response is like this: this is the positive side, and this is the negative side of the frequency, and this is the band. So this is called the pass band, and the rest is elsewhere at 0. So, this is a bandpass filter.

What is a high pass filter? What is the maximum frequency in the digital signal is the pi; π to -π. So, if I say high pass filter means, let us say, 2-kilo hertz high pass filter, that means the after 2-kilohertz signal will exist, below 2-kilowatt signal should not exist. So, that is why the below 2 kilohertz, below ω c signal, is 0, and after that, it exists up to pi; because the maximum oscillation is possible -π to π.

So, that is a high-pass filter. Then I say band stop filter, which means a particular band will be stopped; that means a particular band frequency. So, this to this band; these bands of frequency, signal should not exist. So, it is a combination of, I can say, a low pass filter and a high pass filter; it is nothing but a band pass filter. So, basically, if I am able to design, if you see that if I am able to design this low pass filter, low pass filter; I can implement band pass filter, I can implement high pass filter, I can implement band stop filter.

So, ultimately, it boils down to how I implement and design this low-pass filter. What algorithm should be used to design this low-pass filter? That is our target, okay? So, I got different kinds of filters and their frequency response.

(Refer Slide Time: 06:52)



Now, ideally, it is possible that is ideally I want a low pass filter. Let's say I am not drawing that negative side; let's say talk about only the positive frequency 0 to π only; let us only think about the one band, not two bands; one band 0 to π is a positive frequency. So, ideally, in a low pass filter, if my cut-off frequency is ωp, it should be sub-transition. So, that is after ωp, then and there it should be 0.

But ideally, it is not possible to implement an ideal filter, which is just after ωp. All frequencies will be 0. So, instead of ideal, when we want to implement an algorithm, it will become some sort of black line with this kind of frequency characteristics. So, what is the meaning of this? This means that I can say this is my ωp; ωp is supposed to be 1. Let's say ωp is supposed to be 1 in here, ok, supposed to be 1.

This is a dotted line in the ideal filter that I want, but I have implemented the solid black line filter; that means there is some variation here, and there will be instead of a transition, there will be a gradual transition, and there will be a variation in here also. So, this part is called pass band, pass band. So this part is called the passband; this part is the frequency component. The power will exist; elsewhere, it is supposed to be 0.

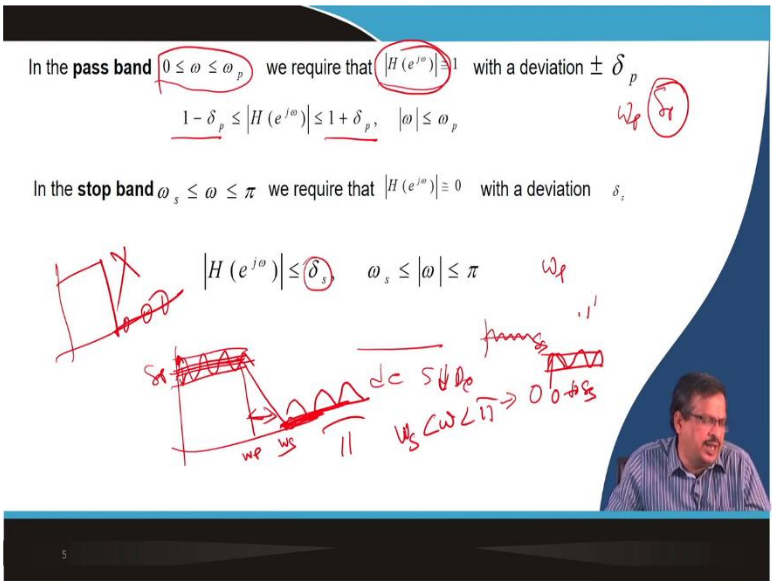
But, so this is pass band, and this is stop band, and this portion is stop band. But if you see, even if the passband, there is some variation of the signal, which is called δp, δp is the variation plus minus variation. If I saw that there is a middle line here, I can say there is a variation in plus δp and minus δp. So, this δp is called passband ripple; δp is called passband ripple. There will be a ripple, which means variation; instead of a straight line, there will be a variation; that variation is called a ripple.

So, since this ripple exists in the pass band, that is why it is called passband ripple. Similarly, in the stop band, there will also be a variation. That is why, instead of 0, there will be some value. So, it is nothing but the δs. So, this δs is called peak stop band ripple. So, δp is the peak pass band ripple; δs is the peak stop band ripple; because this band is the stop band, after ωp, we call it the stop band.

Let's say the stop band is defined here; instead of ωp directly and real filter, this transition is gradual. So, this part ωp, ωs minus ωp is called the transition band. So, this is my transition bandwidth. So, in any filter, when I want to design, there will be a pass band ripple, there will be a stop band ripple, and there will be a transition bandwidth. And ωp, ω s; at the ωp is normalized stop band frequency, and ω is normalized edge frequency of the stop band, normalized edge of stopband frequency.

So, what is δp? How do I define what kind of practical specification will be given to me when designing the filter? So, what is there?

(Refer Slide Time: 11:14)



If I say the pass band is my pass band, ωp is my passband frequency. So, the pass band is 0 to ωp is my pass band. So, within 0 to ωp, what I want is the amplitude to be 1 unity, but there will be a deviation with δp, which is called peak ripple in the pass band. So, instead of 1, instead of this, I will get a variation. So, that variation, this peak-to-peak variation if it is δp, then I call this passband ripple.

So, I can say instead of 1, my filter magnitude filter frequency magnitude is $1-\delta p$ and $1+\delta p$; this is 1, $1-\delta p$ this side and $1+\delta p$ this side peak to peak, δp will be varying. So, my magnitude response of the filter, including ripple, is not 1; it should be $1-\delta p$, or it may be $1+\delta p$; I do not exactly get 1. So, δp is called passband ripple. As you say, when you design, and let's say, I think most of the electronics engineers will be there.

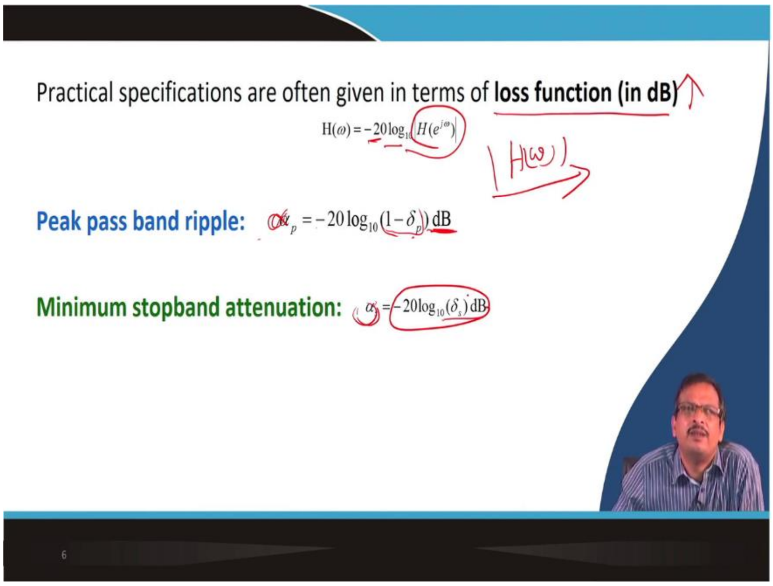
So, as you know, there is a ripple in the DC converter and DC power supply; what is the ripple? I want a constant source, a DC source 5 volt DC, but since I derive that 5-volt DC from an AC source, there will be some ripple. So, ripple is nothing but a top-up at 5 volts, but plus or minus, some variation will be there; that is called ripple. So, here also I want 1, but there will be a plus-minus variation, which is called passband ripple. Similarly, what do I want?

If my stop band ωp is my passband, then the highest frequency is my π the maximum frequency possible is π. So, my pass band varies from ωs to ω 2 π because after that signal is not there. So, in that region, what do I want? I want 0; I want it to be 0. So, if this is my pass band and this is my stop band in here, ωs and this is my ωp.

So, this portion ωs minus ωp is called transition bandwidth, and this ωs is my stopband edge frequency. So, in the stop band, what do I want? I want 0, but instead of 0, there will be some variation. However, this variation is not on both sides because the minimum is 0. So, I can get a kind of sign variation, which is minima, which is 0. So, if the peak is δs, I can say it varies from 0 to δs.

So, instead of 0, it will be δs. So, the maximum value is δs, and the minimum value will be 0. So, this δs is called stopband ripple, do you understand? So, when we design a filter, instead of the ideal filter, an the ideal filter, stopband ripple is 0, and passband ripple is 0. So, I want one, then the transition bandwidth is 0 and then all are 0, but this is not possible. So, instead of that, what I get, I will get a pass band ripple, stopband ripple, and transition bandwidth.

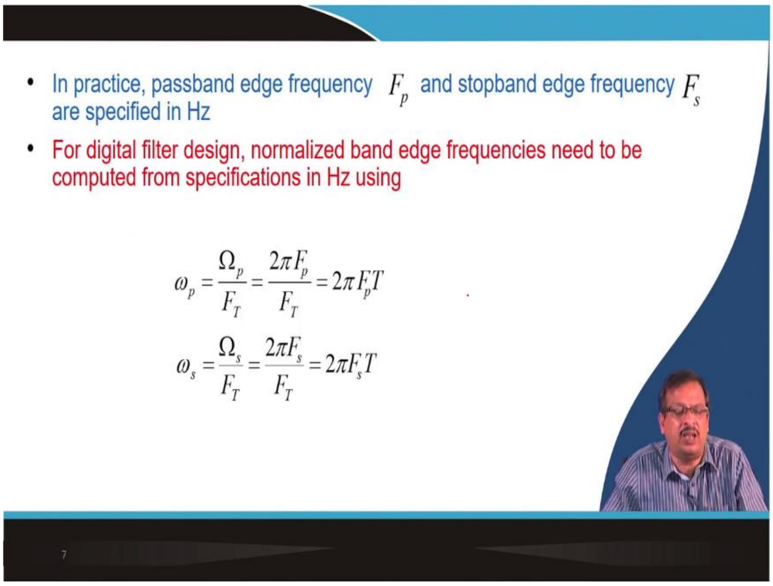
(Refer Slide Time: 15:15)



So, how will this specification be given when you want to design a filter? So, in practical specification, the loss function will be defined in d B. So, if I say what is the value of the amplitude of my pass band H, mod of H ω, this is this has to be expressed in d B. So, it is in voltage. So, if it is square, then it will be power. So, if it is voltage, then I know it is 20 log 10 mod of H is ω, which is the magnitude response of the filter.

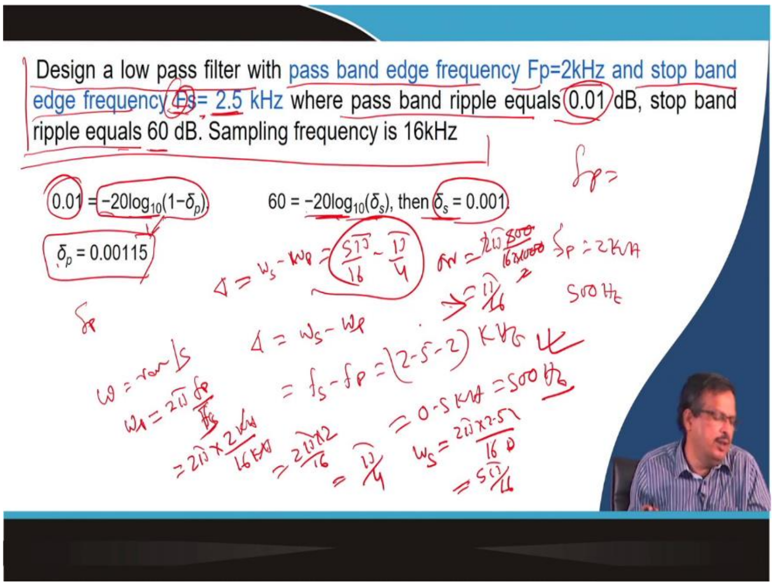
Now, what is pass band ripple? The pass band ripple is α p in d B. So, δp in d B. So, it is nothing but a minus 20 log 10 1-δp in d B; minimum stopband attenuation α s will be given in d B, which is minus 20 log 10 δs will be given in d B.

(Refer Slide Time: 16:22)



So, when you talk about the practical specification of a filter, this will look like this.

(Refer Slide Time: 16:24)



That design of a low pass filter with pass band edge frequency Fp equal to 2 kilohertz and stopband edge frequency Fs equal to 2.5 kilohertz; then what is the transition bandwidth? The transition bandwidth is delta is equal to ωs minus ωp; in hertz, I can say it is nothing but Fs minus Fp, which is nothing but a 2.5 minus 2 kilohertz. So, it is nothing but a 0.5 kilohertz, which is nothing but a 500 hertz.

So, my transition bandwidth is 500 hertz. So, instead of stopband edge frequency, they may say I require a stop pass band frequency or cut off frequency Fp is equal to 2 kilohertz and transition bandwidth is 500 hertz. So, they can define either transition bandwidth or stop band.

So, this is the practical problem given where the passband ripple is equal to 0.01 d B and the stop band is equal to my 60 d B. The sampling frequency is 16 kilohertz, so I have to design the filter. So, what do I have to require? I have to find out the δp. How do I find out the δp?

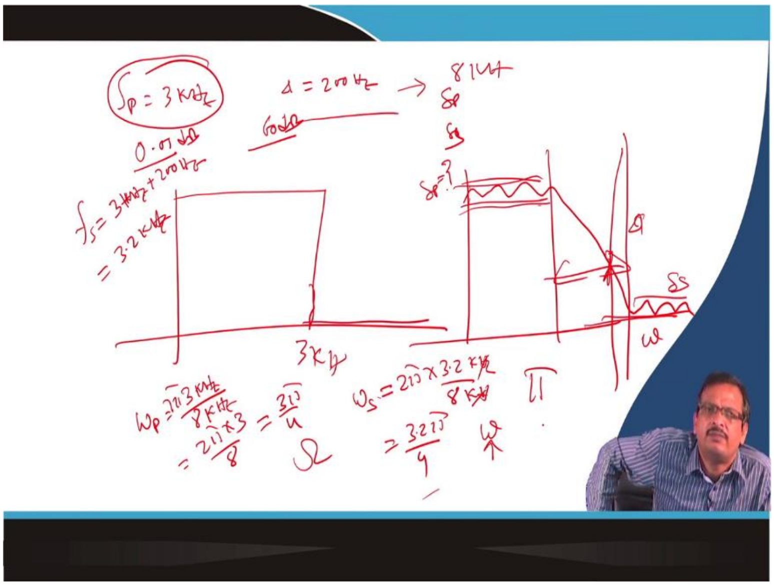
I know δp is minus 20 log 10 1-δp is equal to 0.01. So, from there, I can calculate δp. Similarly, I can calculate δs 60 minus 20 log 10 δs, then δs is equal to this one. So, I know δp and δs, and I know transition bandwidth. But when we design a digital filter, I have to express in normalized frequency, normalized discrete frequency; that in my first lecture, if you remember, I said in the case of the digital domain, the ω is radian per sample; it is not radian per second.

So, if I want to find out ωp, what is the value of ωp then? So, ωp is nothing but a 2 π Fp by f s; this Fs is not this f s, sampling frequency capital F s. So, I can say ωp is equal to 2 π into 2-kilo hertz. In this case, divided by Fs is 16-kilo hertz. So, 2 kilohertz, 16 kilohertz, so kilo hertz kilo hertz cancel, I can say 2 π into 2 divided by 16. So, it is equal to π by 4.

Then what is ωs? Ωs is equal to again; it is nothing but a 2 π into 2.5-kilo hertz divided by 16-kilo hertz. So, a kilo hertz is not required. So, I can say it is nothing but a 5 π by 16. What is the transition bandwidth in delta del delta transition bandwidth? It is 500 hertz, I can directly say that ωs minus ωp, which is nothing but a π by 16 -π by 4. Or I can say, let's say, instead of giving F s, they have given that transition bandwidth 500 hertz; again, I can say 2 π 500 hertz divided by one 16 kilohertz.

So, I can say it is nothing but a π by 16. So, this difference and this one will be a match; this one also will be π by 16. So, when practical things will be given; so if I told you, suppose let us I take a slide.

(Refer Slide Time: 20:35)



Suppose somebody told you to design a digital low pass filter whose cut-off frequency S p is given 3 kilohertz, the transition bandwidth delta is equal to 200 hertz and the stop pass band ripple is 0.01. Let us say d B and stop band ripple is 60. d B. So, passband ripple from d B, I can calculate δp and δs. That is straightforward, but I have to. So, when I want to draw this filter precipitation, I use a low-pass filter.

So, ideally, it should look like this, 3-kilo hertz and the rest will be 0. This is ideal, but instead of the ideal scenario, what will I get? I will get some variation here as well. So, I can say this is nothing but a δp. So, δp I can calculate; this is nothing but a δs I can calculate; this is my transition bandwidth. Sorry, this is my transition bandwidth.

So, I can calculate the delta for transition bandwidth. So, delta, once I say it is ω, then I have to normalize the frequency. So, what is my ωp? Ωp is nothing but the 3 kilohertz divided by the sampling frequency. Let us the sampling frequency is 8 kilo hertz, then I can say 8 kilo hertz; 2 π into 3 kilo hertz divided by 8 kilo hertz. So, it is nothing but a 2 π into 3 divided by 8 is equal to 3 π by 4.

Then what is ωs? Ωs is equal to. So, what is ω? So, Fp is given, what is f s? Fs is equal to stop band frequency s frequency. It is nothing but a 3-kilo hertz plus a 3-kilo hertz plus transition bandwidth of 200 hertz, so it is nothing but the 3.2-kilo hertz. So, ωs nothing but a 2 π into 3.2-kilo hertz divided by 8-kilo hertz. So, I can say 3.2 π divided by 4.

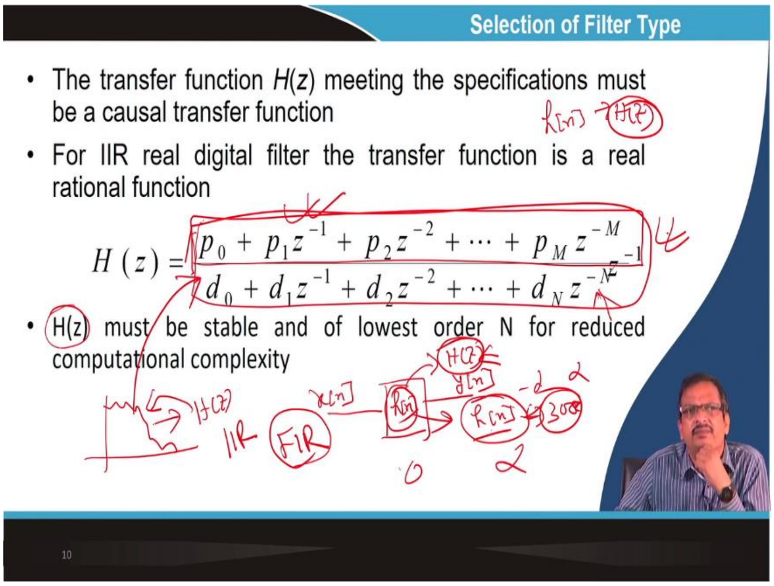
Now, I get the filter specification. So, I know ω Fp, I know everything; now I have to design that low pass filter. So, how do you interpret the filter specification? That is very important. Let us give you an example; suppose I told you to design a bandpass filter whose lower cutoff frequency is, let's say, 1 kilohertz and whose upper cutoff frequency is, let's say 3 kilohertz. So, the pass band width is 2-kilo hertz, the transition bandwidth is 200 hertz, and the pass band ripple and stopband ripple are given in d B.

So, hertz to ω conversion is nothing but a normalized discrete frequency radian per second, normally hertz that when I say the discrete domain; why it is maximum frequency is π, because π radian per sample, it is not radian per second, that is why we write in small ω. When I say analogue frequency, we write in this ω, this is radian per second, and this is radian per sample, that is why we divided by f s, f by f s, ok.

So, I think you have to understand this portion. So, what is there? I have to design a digital filter. So, what is a digital filter is nothing but an algorithm. So, what will be given to me? They will give you a, provide a specification of the filter, which will be described in term of frequency ripple and transition bandwidth. Now, from there, I have to derive the filter parameter; I have to derive ωp, ωs, passband ripple, and stopband ripple, and then I have to go for the design of the filter.

So, before designing the filter, I have to draw the frequency response of the filter and the frequency response of the filter is only provided by the specification. Is it clear? So, you have all are clear; that passband ripple, stopband ripple and all those things.

(Refer Slide Time: 25:10)



Now, I have come to how I implement this filter. So, if I say digital filter, it is nothing but a system. So, a digital system whose sampling in unit impulse is h[n] and my signal is nothing but a, let's say, x[n] passes through h[n], I get y n. So, I can say if I z domain, the h[n] is nothing but the H(z). So, H(z) is the transfer function of the filter that meets the specification. So, a frequency response is given for this H(z).

So, transfer function frequency response is given. So, whatever people give you, design this kind of filter, so from there, I have to derive H(z). So, H(z) must meet the specification given by the user, and it has to be a causal system; if it is non-causal, then how do they practically use it? Now, there are two kinds of filters; one is called the IIR filter, and another is called the FIR filter. What is IIR?

Infinite impulse response filter. What do you mean by infinite impulse? If you see if h[n] is the impulse response of the filter, then h[n] has an infinite very infinite the; this impulse response is infinite, it varies from minus infinity to infinity, n varies from minus infinity to infinity; if it is causal, it is varied from 0 to infinity. So, it is an infinite impulse there, so if I implement an infinite impulse response.

So, if I am able to implement h[n], which consists of infinite impulse response, then it is called IIR filter, infinite impulse response filter. Now, if it h[n] instead of infinite, let us that infinity; I said no, I do not consider the infinite response, let us I consider only 300 in point of this impulse response. So, h[n] 0 to infinity, but I am considering only 300 points, then it is called FIR filter, finite impulse response filter.

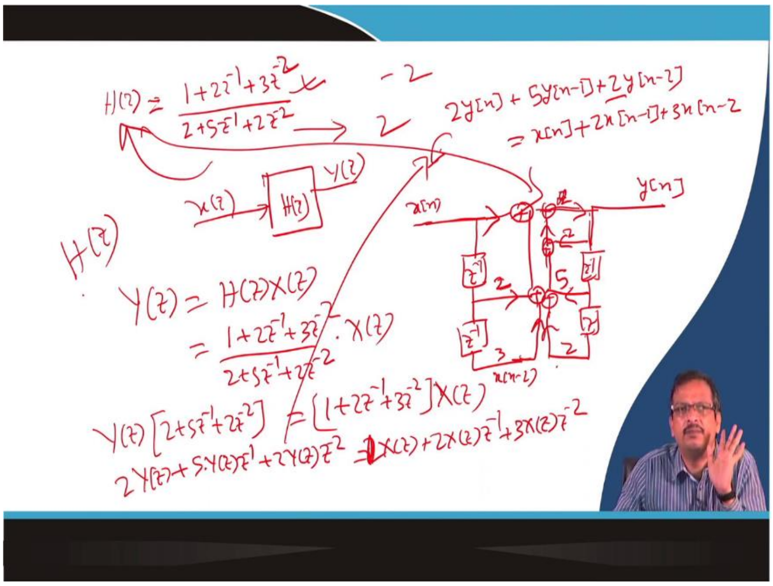
So, there are two kinds of filters: IIR filter, when it is an infinite impulse response, and FIR filter, where it is a finite impulse response. So, when I say IIR filter, then I can say that h[n] in the z domain is nothing but a H(z). So, H(z) is a transfer function, which is nothing but a ratio of a two polynomial, a ratio of a two polynomial. So, I can say as I know that from the z in the z transform view, the upper side solution of the upper side polynomial gives me the 0, and the solution of the lower side polynomial gives me the pole.

That is why H(z) is called the pole-zero filter. So, if I am able to implement the transfer function directly in digital methods, then I am implementing the IIR filter. But what is the condition? H(z) must be stable.

So, in the z domain, I have to derive the transfer function from the given specification of the frequency response, and that transfer function must be stable in the z domain. As far as possible, the number of orders of this polynomial as much as possible I have to keep low because if the order of the polynomial increases, then the complexity of implementing H(z) increases.

So, if I give you, let's say, I give you the H(z) transfer function. So, if I ask you that I have already discussed the structure one and structure two implementations, let's say I give you a filter transformation in polynomial form H(z).

(Refer Slide Time: 29:50)



Let's say H(z) is

$$H(z) = \frac{1+2z^{-1}+3z^{-2}}{2+5z^{-1}+2z^{-2}}$$

So, what is the order of the upper polynomial? Second order. What is the order of the lower polynomial? Second order. So, in this transformation, by the 2 poles and 2 zeros. Now, if I told you, can you implement this H(z) digitally implement H(z) using the structure 2 method or the structure 1 method? Let's say you implement the structure 1 method.

So, how do I implement the structure one realization of this transfer function H(z)? So, what is there? I know this is nothing but the H(z), and there will be an X(z), and I get Y(z). So, x I know, so Y(z) is equal to H(z) multiplied by X(z). So, if I say that, so H(z) I know Now, if I simplify it, Y(z) then

$$Y(z) = \left(\frac{1+2z^{-1}+3z^{-2}}{2+5z^{-1}+2z^{-2}}\right) X(z)$$

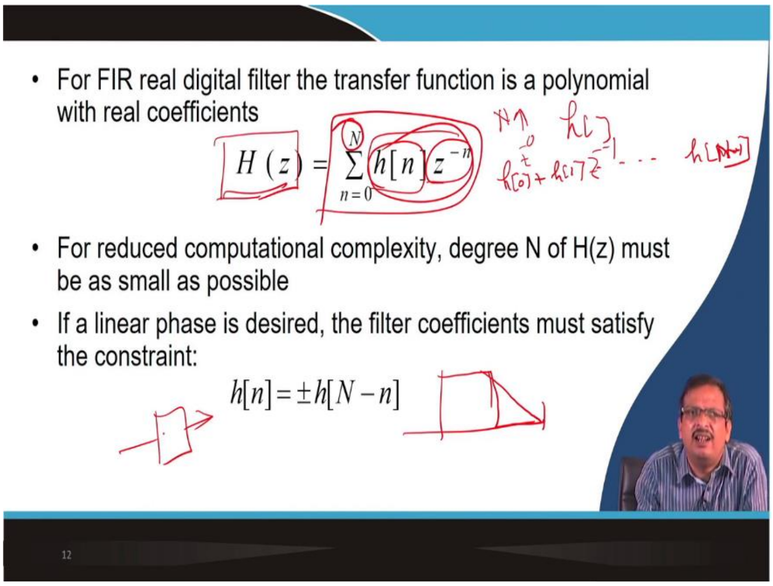
Now, if I do this, what is there; Y(z) 2 into Y(z) plus 5 into Y(z) z^{-1} plus 2 into Y(z) z^{-2} is equal to X(z) plus 2 into X(z) z^{-1} plus 3 into X(z) z^{-2}. Now, if I see it, what is the impulse response? I know it is nothing but the 2 into y n plus. So, z^{-1} signal is delayed by one sample y n minus 1 plus 2 into y n minus 2, which is equal to x[n], so 2 into x[n]. So, it will 2, this is now this will be 2. So, this is x[n]. This is not 2; x[n] plus 2 into x[n] minus 1 plus 3 into x[n] minus 2.

Now, I can say, can I implement this one in a block diagram structure 1; I know this is my x[n] and this is my y[n]. So, x[n] is delayed by z^{-1}, then again z^{-1}. So, this one is x[n] minus 2. So, this will be multiplied by 3 and x[n] minus 1 will multiply by 2, and both will be added together, ok. And then it will be added with x x[n] ok and then it will be fitted to y[n].

Then y[n] is again has to be multiplied by 2 ok, y[n] is to be multiplied. So, y[n] then z^{-1}, z^{-2}; let's say not 2, put it 2 in here, put 2 in here. So, y[n] plus this side, z^{-1} this will be multiplied by 5 added to here, and this will be multiplied by what, 2, 2 and added to here. So, this is the structure 1 implementation of this transfer function. So, if I am able to derive the transfer function from the given specification, then I can implement the IIR filter using structure 1 or structure 2 realization if possible.

However, the condition is that H(z) must be stable in the z domain. So, stability means all poles should be inside the unit circle. So, that is all that condition I have to catch. So, there is a procedure to derive the transfer function from the given specifications, which we will discuss during the design of the IIR filter.

(Refer Slide Time: 34:58)



Now, I said if I truncate that frequency response, instead of infinite impulse, let's say I n a number of impulses. So, I can say h h 0, h 1, h 2 dot dot dot up to h infinite; instead of infinite, I only consider N minus 1. So, N number of impulse responses I have taken. So, it is a finite impulse response filter. So, h 0 will multiply z to the power minus 0, h 1 will be multiplied z^{-1}. So, I can say h[n] into z^{-1}. So, that is my H(z).

So, here, it is not polynomial; I have not taken an infinite impulse response; I have truncated that impulse response; I took a certain number, which is N, and this N is called the degree or order of the filter. Now, if you see if I increase the N, then if you see a number of multiplication also increases, z is a complex number, so a number of complex multiplication increases. So, I have to put N as much as possible, but when I put the lowest, it will affect my filter characteristics.

So, what will be the effect on filter characteristics; that means if I take infinite impulse, then suppose it is an ideal filter. Now, once I reduce the order of the filter, the transition bandwidth will increase. So, there is a relationship between the transition bandwidth and the order of the filter. If you decrease the order of the filter, your transition bandwidth will be. So, if I require a very short cut-off, then my filter order must be increased.

If I require a shallow cut-off wide, ok, no problem, then the order of the filter will be very low. Another condition is called the phase linear phase. So, there is a concept of phase in the filter. So, h[n] is nothing but a system. So, the requirement is that it must be a linear phase. So, what is the concept of linear phase, and how is it possible to implement it in an FIR filter? We will discuss this in the next class.

Thank you.