Digital Signal Processing

Prof: S. C. Dutta Roy

Department of Electrical Engineering

Indian Institute of Technology, Delhi

Lecture – 31

Lattice Synthesis (Contd...)

This is the 31st lecture and we continue our discussion on Lattice Synthesis.

(Refer Slide Time: 01:13)

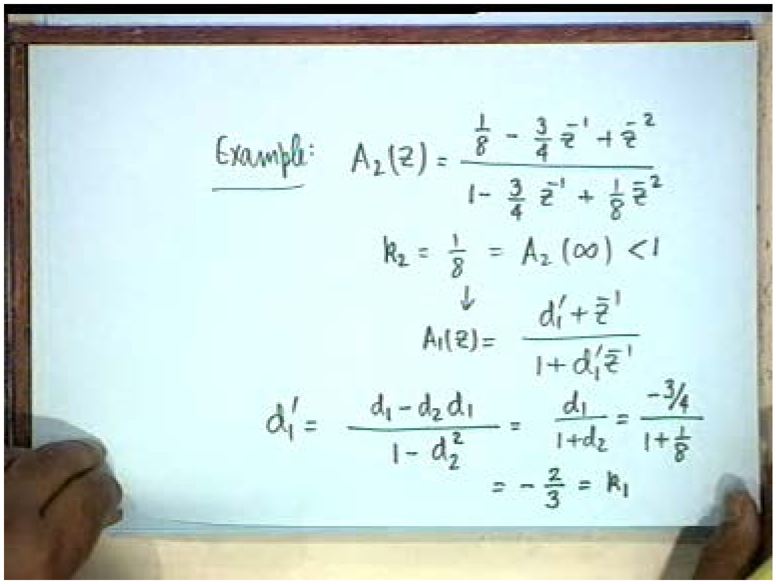
In the 29th and 30th lectures we talked about various forms of IIR realizations namely direct, cascade and parallel. And in case of parallel IIR, you speed up the processing because instead of a number of delays equal to the order of the transfer function, you can afford to use only two delays per channel so that the minimum processing time is twice the sampling interval. You have to add to this the time required for multiplications but addition is almost instantaneous. It is the multiplication which is time consuming. Then we switched over to all pass IIR realization and the first approach that we used was multiplier extraction approach. The aim was to make the realization canonic in multipliers as well as delays. Ordinary IIR direct form realizations do not achieve canonic multiplication. First order requires two multipliers; second order requires four multipliers because in the denominator one term is unity and in the numerator the coefficient of z^{-2} = 1, so there are four multipliers.

On the other hand, using the multiplier extraction approach, you can use one multiplier and one delay for the first order and two multipliers and two delays for the second order. Next we considered digital two pair extraction approach, and this is only applicable to All Pass Filters. This approach leads to a modular structure, that is a repeated structure, in which only one parameter is variable in each module, called a lattice. Each lattice can be realized either with two multipliers or with a single multiplier. But for some reasons, particularly historical and VLSI implementation, two multiplier realizations are preferred except in dedicated programmable DSP chips like the tunable filters.

We will start with tunable filters today. Let us first consider an example, then we will go to the tunable filters. Here is a second order example: $A_2(z) = [(1/8) – (3/4)z^{–1} + z^{–2}]/[1 – (3z^{–1}/4) + (1/8)z^{–2})]$. This is an all pass filter. If you wish to realize by a lattice you require two lattices in cascade terminated in a straight connection, i.e.$Y_2 = X_2$ which is the terminating condition. The first lattice has $k_2 = 1/8$, because $k_2 = A_2(\infty)$. You can see that the coefficient of the highest power term in the denominator is the same as this, but then you must make sure that the constant term in the denominator is unity. So $k_2 = 1/8 < 1$. I also told you that if at any intermediate stage, k comes out as equal to or greater than 1, then your job is simplified. You do not proceed further because the system becomes unstable and an unstable filter cannot perform a useful job unless you design an oscillator.

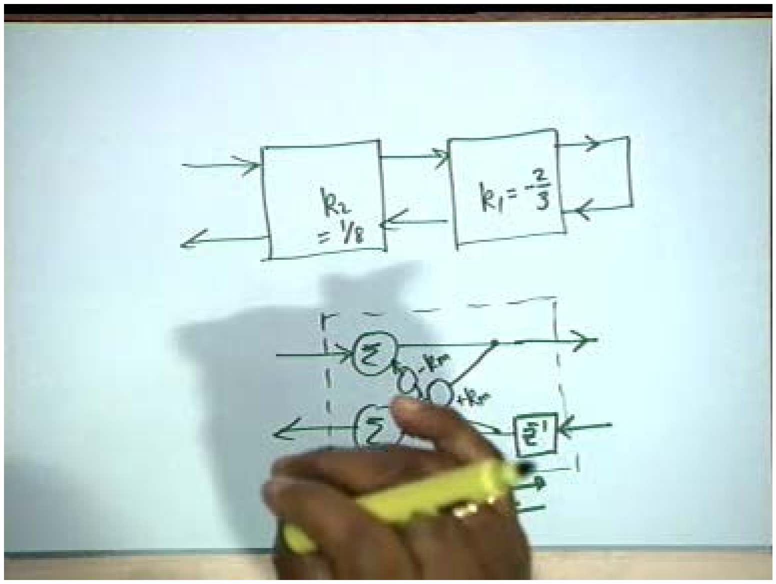
The oscillator is an unstable system with which you generate sinusoidal or other kinds of waveforms. To find out the other lattice parameter k_1, you have to find out $A_1(z)$ of the form $(d_1’+ z^{–1})/(1+ d_1’z^{–1})$. Now $d_1’$ from the recursion formula is $(d_1 – d_2 d_1)/(1 – d_2^2) = d_1/(1 + d_2)$. This is an important feature which shall always occur in the last stage of lattice synthesis. Whenever you are going from the second order to a first order all pass, this shall always occur and the formula is simplified. And since $d_1 = -3/4$ and $d_2 = 1/8$, $d_1' = k_1$ comes out as $-2/3$; k_1 magnitude is less than unity therefore the system is stable.

(Refer Slide Time: 4:49 – 8:36)



Once you have found k_2 and k_1, all that remains to do is to draw the filter. I draw the lattice simply as a box starting with k_2 then k_1 and the output is terminated in a straight connection, as shown in the figure. Put the values of k_2 and k_1 in these boxes. Each stands for a lattice with 2 summers and 2 multipliers, $\pm\ k_m$ in the m^{th} order stage. In addition to these, you have a delay. You must indicate signal flow directions by appropriate arrows. We have compressed these details into a box so that we do not have to draw these again and again. This modularity leads to simplified fabrication in VLSI. Without modularity, processing steps change at every step and it becomes a costly proposition.

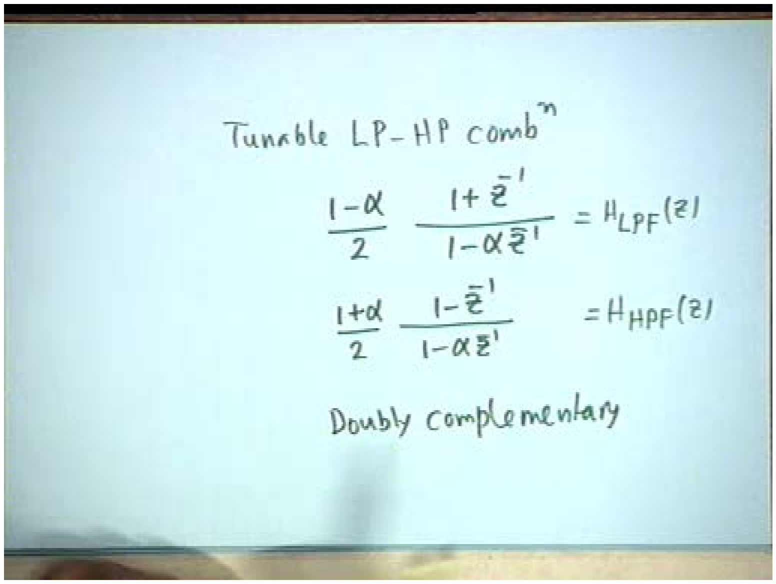
(Refer Slide Time: 08: 41 – 10:40)



On the other hand, all that you have to change in the processing of a lattice structure is the value of k for each stage and therefore it facilitates VLSI implementation.

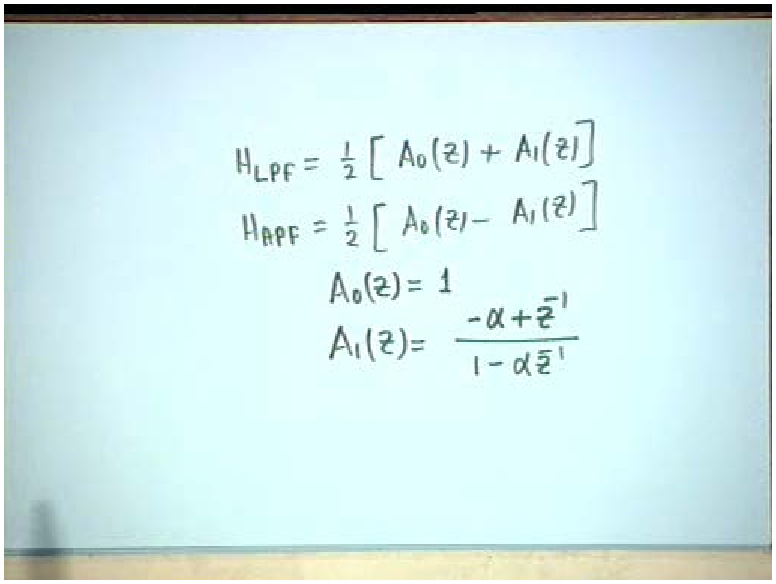
Now let us consider the tunable low pass, high pass combination, starting with the first order. If you recall, the transfer function is $((1 - \alpha)/2)(1 + z^{-1})/(1 - \alpha z^{-1})$ for a first order low pass filter $H_{LPF}(z)$ and its complementary all pass as well as power complementary filter is the high pass filter $H_{HPF}(z) = ((1 + \alpha)/2)(1 - z^{-1})/(1 - \alpha z^{-1})$. The sum of the two is an All Pass Filter and they are also power complementary. That is, if I take $|H_{HPF}(z)|^2 + |H_{LPF}(z)|^2$ for $z = e^{j\omega}$, the sum shall be unity. These are doubly complementary filters. The sum is all pass and the magnitude squares sum up to unity.

(Refer Slide Time: 11:03 – 12:43)



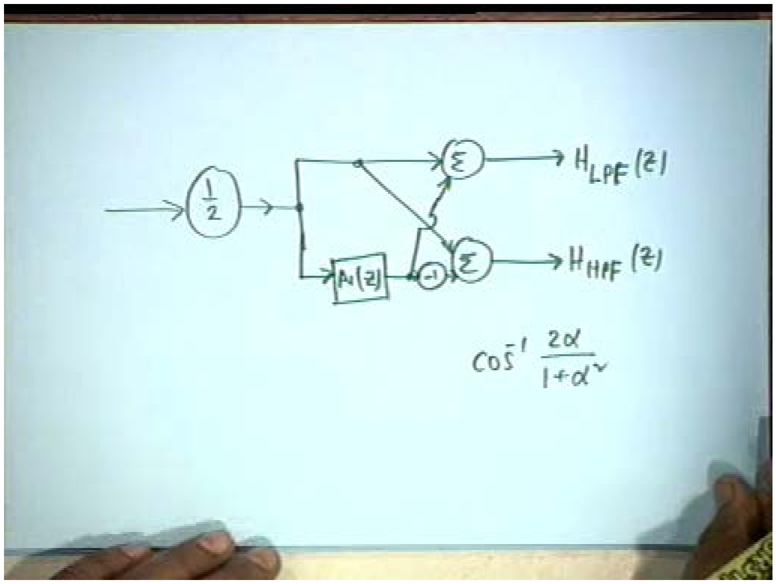
In our previous discussion we also showed that these low pass and high pass filters can be expressed as the sum and difference of the two All Pass Filters. $H_{LPF}(z) = ½ [A_0(z) + A_1(z)]$ and $H_{HPF}(z) = ½ [A_0(z) – A_1(z)]$ where $A_0(z) = 1$ i.e. it is a straight connection and $A_1(z)$ is the first order All Pass Filter $(– \alpha + z^{-1})/(1 – \alpha z^{-1})$. Therefore the realization of first order All Pass Filter suffices to make a multi output system. One of the outputs is low pass and the other is high pass and this is one of the most popular filters in digital stereo. The low pass one goes to one channel of the stereo while and the other stereo channel is the high pass one. So low frequencies and high frequencies are separated out and that makes a stereo record.

(Refer Slide Time: 12:47 – 14:08)



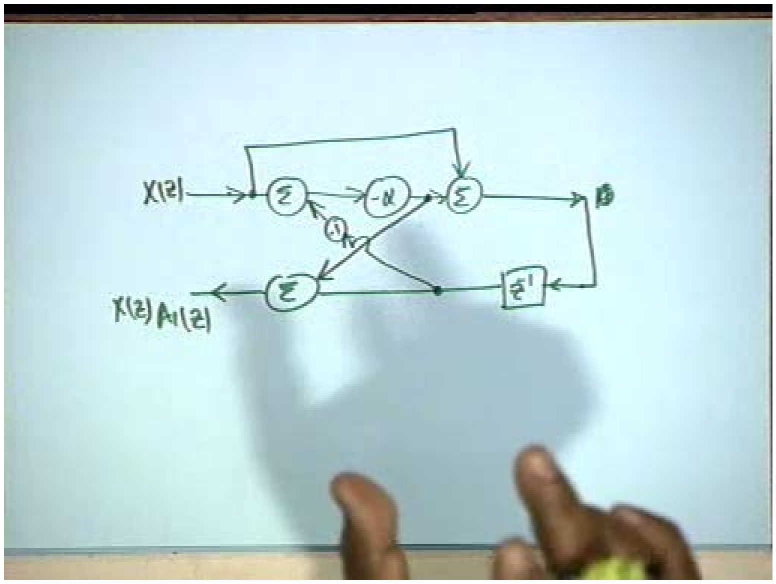
We also discussed the implementation earlier. Take the factor 1/2 out which is a shift, not multiplication, and then you have two channels. One of them goes straight because the all pass function $A_0(z)$ is 1, which feeds both the summers. Then you shall have $A_1(z)$, which feeds a summer to form ½ [$A_0(z)$ + $A_1(z)$], thus realizing $H_{LPF}(z)$. In order to create $H_{HPF}(z)$, you shall have to multiply $A_1(z)$ output by – 1, which is not also a multiplication but a change in the sign. This total thing is being realized by only a single delay and a single multiplier, and can be fabricated by a dedicated chip, in which α is programmable. You get two functions, low pass and high pass, for splitting a given audio channel into its low frequency and high frequency components. What does ά control? ά changes the bandwidth which is $\cos^{-1} 2ά/(1 + ά^2)$ radians for both the channels.

(Refer Slide Time: 14:19 – 16:22)



Therefore for $A_1(z)$, a single multiplier lattice structure shall do and this is available commercially as a programmable chip. Programmability only involves changing the bandwidth ά. When I draw a lattice I will simply put it in a box and put k inside the box. k is the lattice parameter, which is – ά in this case. The detailed diagram is shown in this figure. This is the single multiplier lattice structure which makes sure that even if α changes, the all pass property is not destroyed. This will not be the case with a two multiplier structure, because the two multipliers, even if identical (except for sign) in the design, may not remain so after quantization.

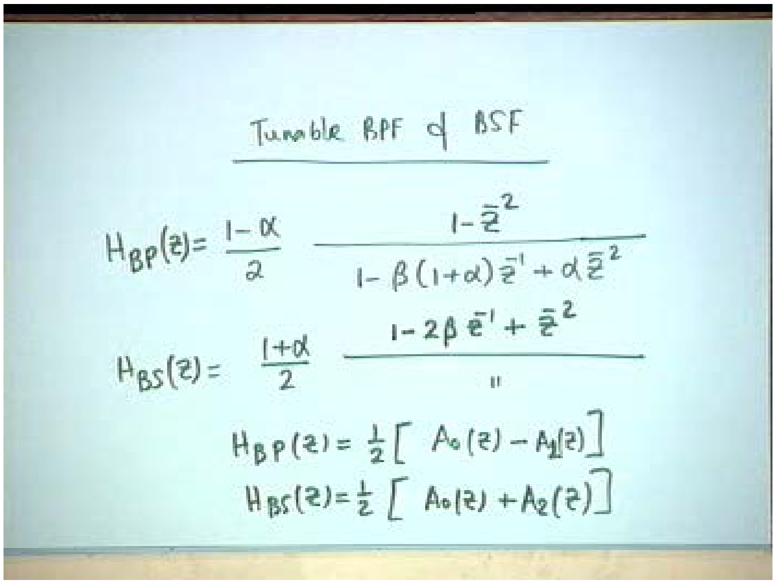
(Refer Slide Time: 16:41 – 18:33)



So this is one of the structures for tunable filters. The other is tunable band pass and band stop combination. Band pass and band stop cannot be made in first order and the minimum order needed is two. Recall that the second order band pass transfer function was $[(1 - \alpha)/2]\ (1 - z^{-2})/[1 - \beta(1 + \alpha)z^{-1} + \alpha z^{-2}]$. The numerator is $1 - z^{-2}$ which makes the DC response as well as the response at π equal to zero. That is what is needed in a bandpass filter.

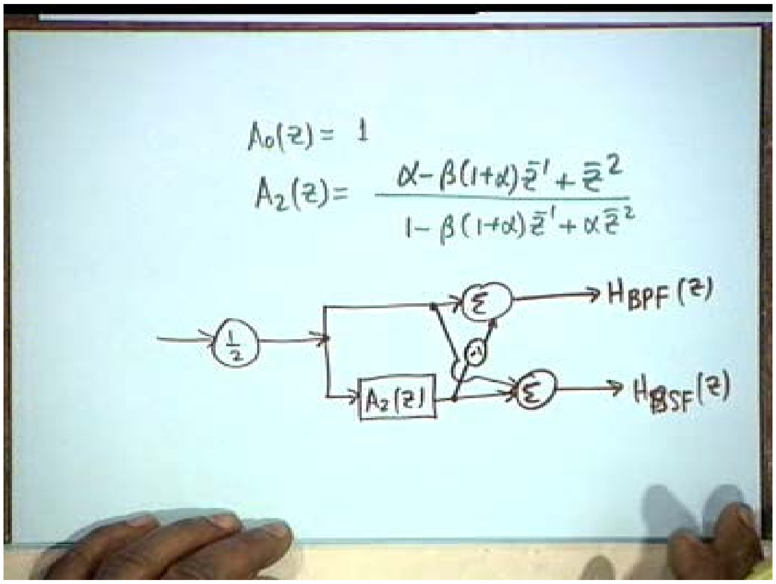
Now we shall appreciate why the denominator is written in this form and not in the form $1 - \beta z^{-1} + \alpha z^{-2}$ after we derive the lattice realization of the corresponding All Pass Filter. Let me also write the band stop transfer function, which is $[(1 + \alpha)/2]\ (1 - 2\beta z^{-1} + z^{-2})/[1 - \beta(1 + \alpha)z^{-1} + \alpha z^{-2}]$. These are also doubly complementary, that is all pass as well as power complementary filters. We shall show that these two filters can also be realized in a single chip using only one second order All Pass Filter whose denominator is the same as that of either filter. When you actually derive the two lattices, you shall see that one of them has a parameter ά while the other has the parameter $-\beta$. If you recall our previous discussion on these two filters, ά and β independently control the bandwidth and the center frequency. The center frequency in case of band pass is the frequency of maximum response and in the case of band stop it is the frequency of rejection. $H_{BP}(z)$ and $H_{BS}(z)$ can also be written as the sum and difference of two all pass functions. But here unlike low pass and high pass, the band pass the difference ½ [$A_0(z)$– $A_2(z)$], whereas the band stop is the summation of the two ½ [$A_0(z) + A_2(z)$].

(Refer Slide Time: 18:49 – 22: 37)



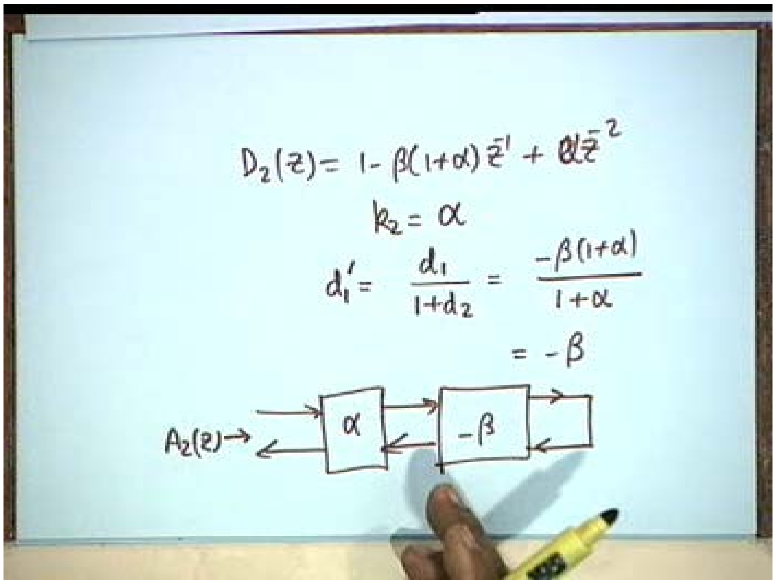
I repeat, the band pass requires a difference and the band stop requires an addition of the two all pass functions $A_0(z) = 1$ and $A_2(z) = [\alpha - \beta(1 + \alpha)z^{-1} + z^{-2}]/[(1 - \beta(1 + \alpha)z^{-1} + \alpha z^{-2}]$. The structure would be very similar to the first order case. A multiplier 1/2 is followed by two channels, one leading to $A_2(z)$ which now has to be multiplied by – 1 to get a band pass filter $H_{BPF}(z)$, and the other a straight connection. To get a band stop filter $H_{BSF}(z)$, $A_2(z)$ output goes straight to the summer. $A_2(z)$ obviously shall require two lattice sections. Let us see what the lattice parameters are.

(Refer Slide Time: 22:42 – 24:06)



$A_2(z)$ denominator is $D_2(z) = 1 - \beta(1 + \acute{\alpha})\, z^{-1} + \acute{\alpha}\, z^{-2}$ so $k_2 = \acute{\alpha}$. For finding k_1, you have to find d_1' and if you recall, d_1' would be d_1 divided by $(1 + d_2)$, which is simply– β. In other words, what we require is a lattice with a parameter ά, and another with the parameter (– β). The second section is terminated in a straight connection. Each of the two boxes in the figure have been realized by a single multiplier structure. This gives you a programmable DSP chip with two functional outputs where one is the band pass and one is the band stop. The center frequency can be controlled by one multiplier β which has to be programmable. The bandwidth is changed by ά. And these are independent of each other that is when you change ά the center frequency does not change. I wish to remind you at this point that the term bandwidth, applied to the band-stop filter is not the stop bandwidth, but is simply the difference between the 3-dB frequencies of the two passbands. The stop bandwidth is determined by the specified tolerance in the stop band. It is difficult to attain this feature in analog filters. Digital filters totally isolate the bandwidth control and the center frequency control. This set of bandpass – bandstop combination is available as a commercial chip.

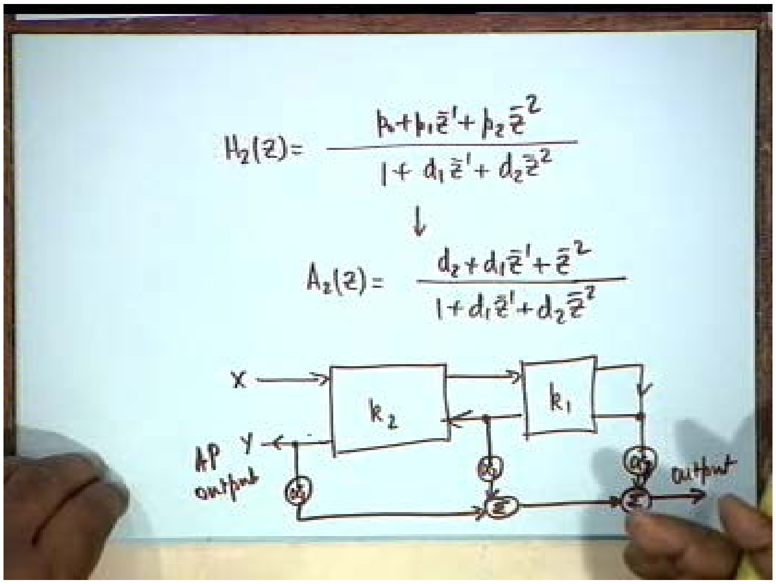
(Refer Slide Time: 24:27 – 26:40)



So far we have talked about only All Pass Filters. We have also used All Pass Filters to obtain a low pass, a high pass, a band pass and the band stop filter, can the lattice realization of all pass filters be so modified to obtain a general IIR transfer function? Let us see. Consider a general transfer function for which the denominator is $1 + d_1z^{-1} + d_2z^{-2}$ and the numerator is general, of the form $p_0 + p_1z^{-1} + p_2z^{-2}$. Can it be realized by a lattice structure that would make the lattice a more versatile structure rather than only being limited to All Pass Filters?

Now, in order to realize this, the first step is to find out the corresponding all pass and realize it. That is, realize the transfer function $[d_2 + d_1z^{-1} + z^{-2}]/[1 + d_1z^{-1}+ d_2z^{-2}]$. For the corresponding lattice structure you tap some of the outputs. The tapping is done in the lower line as shown in the figure. These tapped signals are weighted, that is multiply them by $ά_1$, $ά_2$ and $ά_3$, and add them together, two at a time, to get the final output. I have to realize three coefficients p_0, p_1 and p_2, and I have used three weights $ά_1$, $ά_2$ and $ά_3$. Now I shall find out the output numerator polynomial and then match the coefficients to find out the required weights $ά_1$, $ά_2$, $ά_3$.

(Refer Slide Time: 27: 25 – 30: 39)

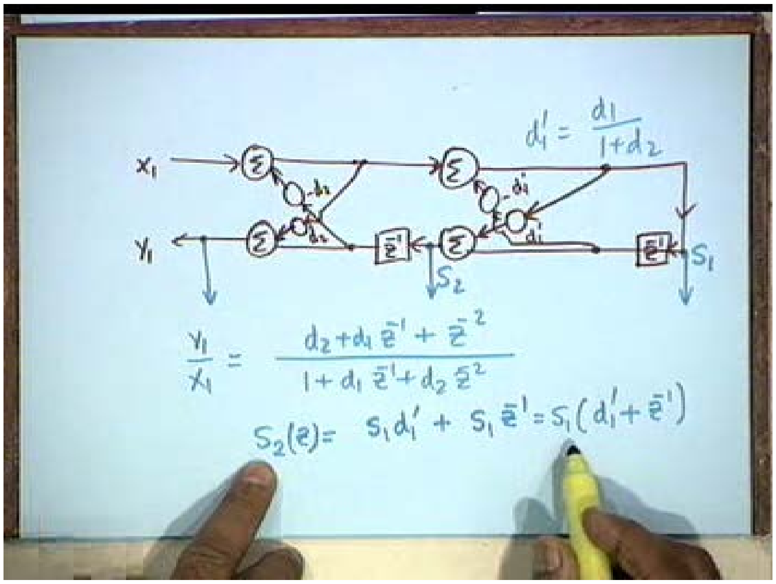


Let us carry this out completely then we shall see what these multipliers $ά_1$, $ά_2$ and $ά_3$ should be. Is it guaranteed that we shall get a numerator from which I can choose three unknown parameters? Yes, it should be possible.

Once we are sure it can be done, then we proceed. Now I have to draw the total structure because I have to analyze it. First we shall draw the all pass structure and then we shall see the taps. This has been shown in the next figure. Let us call the tapped signals as S_1, S_2 and S_3, which of course is Y_1.

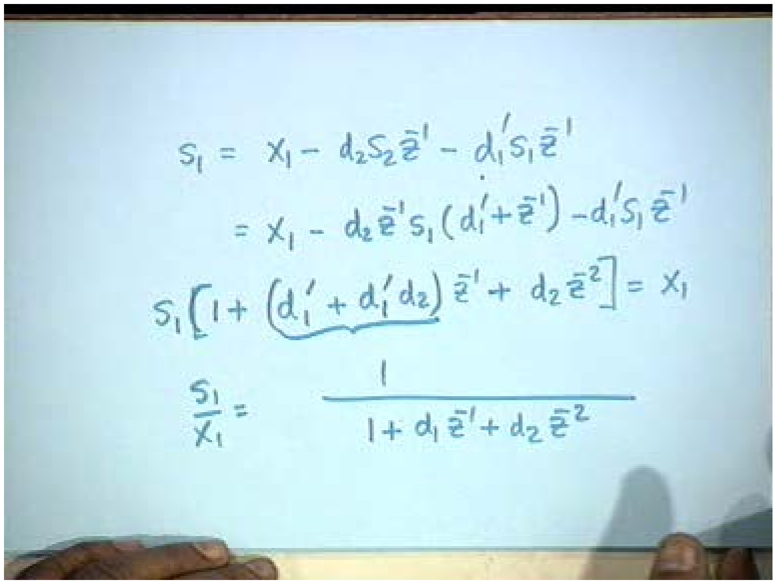
Now you know that $Y_1/X_1 = (d_2 + d_1z^{-1} + z^{-2})/(1 + d_1z^{-1} + d_2z^{-2})$. Let us look at what is S_1/X_1 and S_2/X_1. As far as S_2 is concerned, you see that $S_2(z)$ is very simply S_1 (d_1' + z^{-1}). What is this polynomial d_1' + z^{-1}? It is the numerator of the first order all pass function where d_1'= d_1/(1 + d_2). Incidentally, this method of tapping is a very beautiful concept first given by Gray and Markel. This is their very elegant contribution to DSP literature that an all pass lattice can be used to realize any arbitrary IIR transfer function and you will see how things fit into place. We have established a relationship between S_2 and S_1.Let us proceed further.

(Refer Slide Time: 31: 43 – 36: 07)



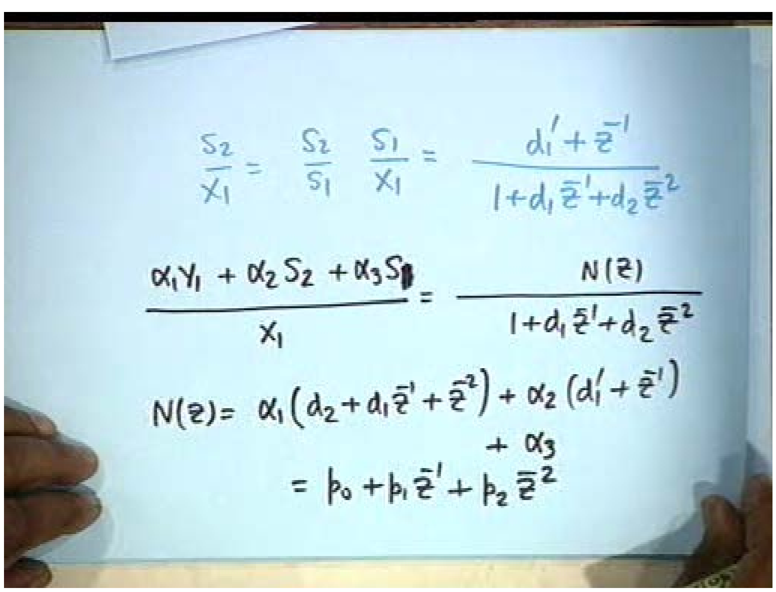
Let us find out S_1. S_1 obviously is contributed by three signals where one is X_1, the other is $S_2z^{-1}(-d_2)$ and the third one is $S_1z^{-1}(-d_1')$. Let us write this down: $S_1 = X_1 - d_2S_2z^{-1} - d_1'S_1z^{-1}$. And we have already obtained a relationship between S_2 and S_1 therefore, by elimination, I get $S_1/X_1 = 1/(1 + d_1z^{-1} + d_2z^{-2})$.

(Refer Slide Time: 36:14 – 39: 17)



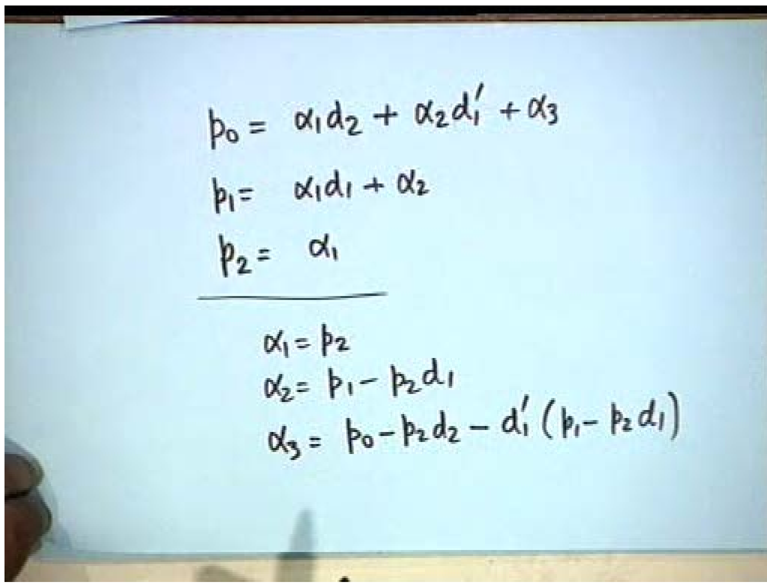
My desired output is $\acute{\alpha}_1Y_1 + \acute{\alpha}_2S_2 + \acute{\alpha}_3S_1$. The denominator shall be the same $1 + d_1z^{-1} + d_2^{-2}$ and the numerator is $N(z)\ X_1$ where $N(z) = \acute{\alpha}_1(d_2 + d_1z^{-1} + z^{-2}) + \acute{\alpha}_2\ (d_1' + z^{-1}) + \acute{\alpha}_3$, which should be equal to $p_0 + p_1z^{-1} + p_2z^{-2}$. Now you match coefficients to get the required multipliers.

(Refer Slide Time: 39: 24 – 42: 12)



Let us match the coefficients and see the results. Obviously $p_0 = \acute{\alpha}_1 d_2 + \acute{\alpha}_2 d_1' + \acute{\alpha}_3$, $p_1 = \acute{\alpha}_1 d_1 + \acute{\alpha}_2$ and $p_2 = \acute{\alpha}_1$. Now you can go from the last one $\acute{\alpha}_1 = p_2$, and since you know $\acute{\alpha}_1$, you can find $\acute{\alpha}_2$ and since you know $\acute{\alpha}_1$ and $\acute{\alpha}_2$ you can find out $\acute{\alpha}_3$. And the results are: $\acute{\alpha}_1 = p_2$ and $\acute{\alpha}_2 = p_1 - p_2 d_1$, $\acute{\alpha}_3 = p_0 - p_2 d_2 - d_1'(p_1 - p_2 d_1)$. Thus any arbitrary IIR filter can be realized with all pass lattice.

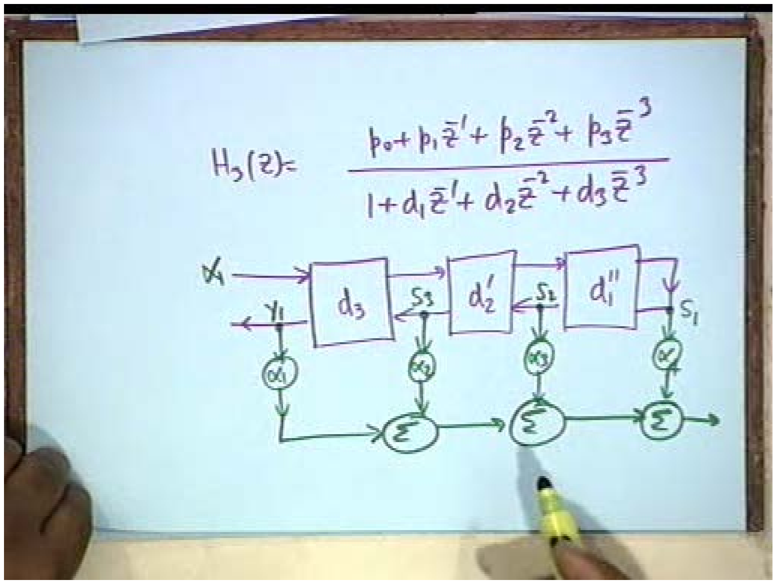
(Refer Slide Time: 42: 35 – 44: 09)



)

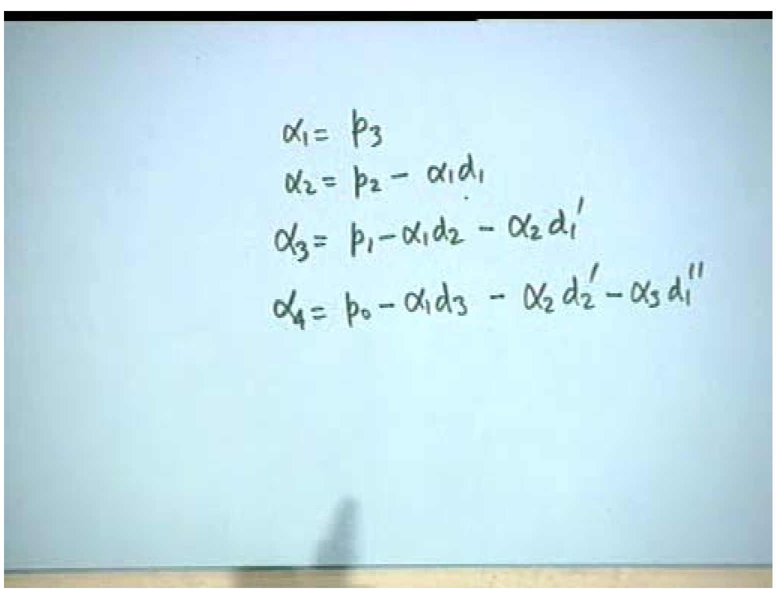
Lattice in that sense is very versatile. Any arbitrary IIR transfer function can be obtained from the lattice structure of a versatile element, namely the All Pass Filter. That is why DSP is obsessed with All Pass Filters. If you have a third order transfer function $(p_0 + p_1 z^{-1} + p_2 z^{-2} + p_3 z^{-3})/(1 + d_1 z^{-1} + d_2 z^{-2} + d_3 z^{-3})$, then you would have realized the lattice with three parameters d_3, d_2' and d_1'' and then a straight connection. Now you take taps with weights $\acute{\alpha}_1$, $\acute{\alpha}_2$, $\acute{\alpha}_3$ and $\acute{\alpha}_4$ and add them together. By observation and by looking at the second order realization can you tell me what these signal would be? Let us call them S_1, S_2, S_3 and Y_1. The numerator of $S_1/X_1 = 1$, the denominator is the same, the numerator of S_2 would be $d_1'' + z^{-1}$, and for S_3/X_1, it would be $d_2' + d_1' z^{-1} + z^{-2}$. For Y_1/X_1, the numerator of course would be $d_3 + d_2 z^{-1} + d_1 z^{-2} + z^{-3}$; so you can find out the required coefficients.

(Refer Slide Time: 44: 37 – 47: 02)



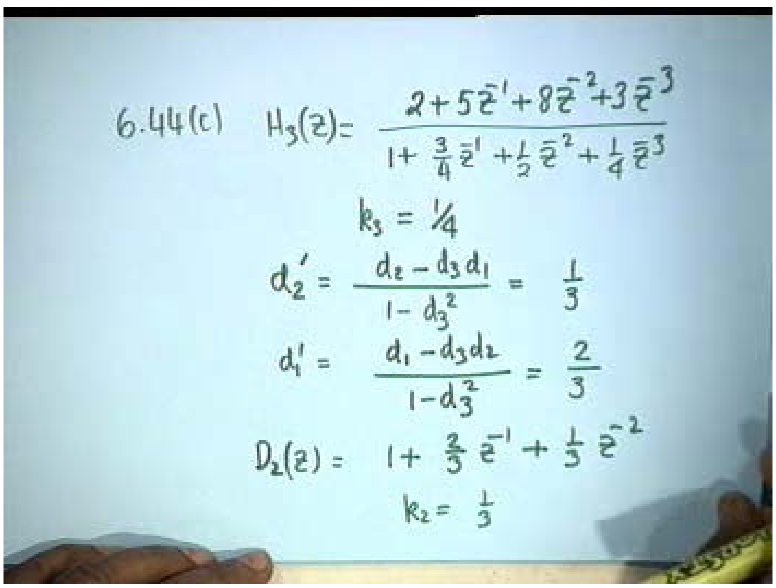
Please do verify that these coefficients are $\acute{\alpha}_1 = p_3$, $\acute{\alpha}_2 = p_2 - \acute{\alpha}_1 d_1$, $\acute{\alpha}_3 = p_1 - \acute{\alpha}_1 d_2 - \acute{\alpha}_2 d_1'$ and $\acute{\alpha}_4 = p_0 - \acute{\alpha}_1 d_3 - \acute{\alpha}_2 d_2' - \acute{\alpha}_3 d_1''$. We shall work out an example of a general transfer function.

(Refer Slide Time: 47:12 – 47: 54)



This example I have chosen intentionally so that the results obtained here can be utilized later also. This is Prob. 6.44c in Mitra and it says realize $H_3(z)$ by Gray Markel structure, where it is given as $[1 + (3/4)z^{-1} + (1/2)z^{-2}\ (1/4)z^{-3}]^{-1}\ (2 + 5z^{-1} + 8z^{-2} + 3z^{-3})$. The first thing to do is to realize an All Pass Filter with this denominator and the numerator $z^{-3}D_3(z^{-1})$. You do not have to write this. You can see that for the all pass $k_3 = ¼$; then you have to find $D_2(z)$. In finding $D_2(z)$, your coefficients are $d_2' = (d_2 - d_3d_1)/(1 - d_3^2)$ and that comes out as 1/3. Also, $d_1' = (d_1 - d_3d_2)/(1 - d_3^2)$ and that comes out as 2/3. Just write the denominator $D_2(z) = 1 + (2/3)\ z^{-1} + (1/3)\ z^{-2}$; therefore $k_2 = 1/3 < 1$. If at any point, k_i comes equal to 1 or greater than 1 you give up say it is not possible.

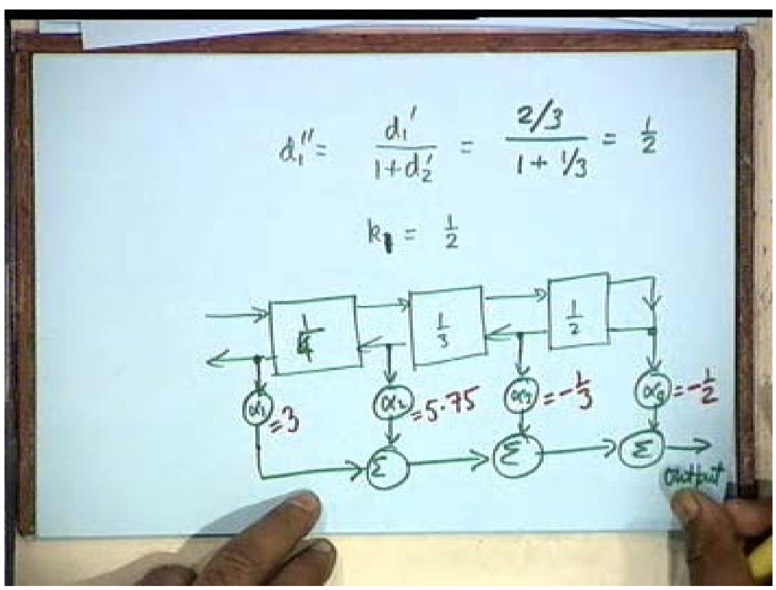
(Refer Slide Time: 48:22 – 50: 57)



And finally in calculating $D_1(z)$, all that we have to find out is d_1'' and this you know is $d_1'/(1 + d_2') = 1/2$. Therefore k_1 is simply ½. Now I can draw my all-pass lattice. The next job is to calculate α_1, α_2, α_3 and α_4 and then add them to get my desired output. If you proceed as in the third order case, then α_1 comes as 3, α_2 is 5.75, α_3 is – 1/3 and α_4 is – ½. Now in these multipliers, the only problem is this multiplier – 1/3 because whatever way you quantize, you are going to have an error. On the other hand, 1/2 is no multiplication, 5.75 is no multiplication because 5 is 4 +1 and .75 is ¾ = (1/2) + (1/4); these shifts shall be done in parallel. 3 is no problem, it is 2 + 1; so you shift one channel and put the other one directly. Only – 1/3 is a problem.

While appreciating the Gray-Marvel innovation, a question which has always troubled me is: can we not synthesize an arbitrary IIR transfer function by a true lattice only? That is, can we avoid the tappings and the additions? Also, can we obtain it canonically, without going through the Mitra procedure of a feedforward in every section? I obtained the answers to both the questions in the affirmative after several years of investigation. The method turns out to be very simple indeed. The results were published in two papers: 1) IETE Journal of Research, Jan-Feb 2007 issue and 2) IETE Journal of Research, Jan-Feb 2008 issue. The second paper, in fact presents a new method of realization of an arbitrary IIR transfer function in terms of a simpler IIR transfer function with an FIR feedback path. These will hopefully appear in future text books.

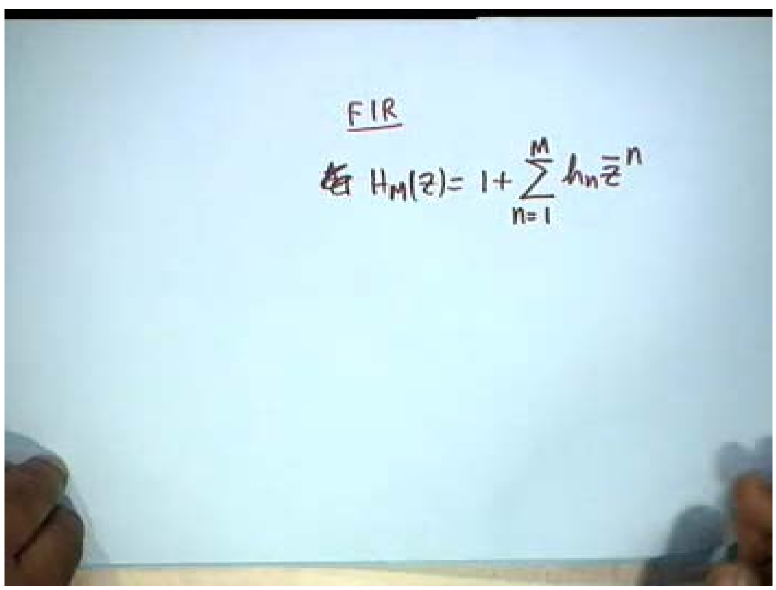
(Refer Slide Time: 51: 03 - 54:13)



This leads us now to the question: what about FIR lattice? Can we realize an FIR filter also by a lattice structure? It is a historical fact that FIR lattices were not synthesized; they were obtained by assuming the structure and then carrying out analysis. We shall follow the same procedure.

Do you understand the methodology? Synthesis can be done ab initio; that is, we do not know the structure and try to get it as we did in the case of IIR lattice starting from digital two pair extraction approach. In FIR case, traditionally and historically, we start from the structure, analyze it, compare it with the given requirement and match things; this is called synthesis by analysis. For example, if you want a second order analog band stop filter then you know various structures that you can use, for example the parallel T RC network. So you draw the structure and then match the coefficients with the values of the resistors and capacitors. This is called synthesis by analysis. So we will follow synthesis by analysis for the FIR structure. I will first draw the complete structure for a transfer function $H_M(z) = 1 + \sum\ h_n z^{-n}$, n = 1 to M where M is the order of the filter. There is no denominator as it is FIR. We shall take the constant term as 1, always.

(Refer Slide Time: 54:33 – 56:46)



The structure looks like the one shown in the next slide (lecture 32). Because it is FIR, there is no feedback. It should be a non recursive structure. So, start with input X_0 (the notations also change), delay it by one sample, and then the lattice starts. The lattice is similar to the IIR lattice; there is a criss-cross, but two multipliers are all in the forward direction. The difference between IIR lattice and FIR lattice is that the two multipliers are identical. This can also be realized by a single multiplier structure. It requires a little bit of innovation. The two outputs of the first section shall be called as X_1 and X_1'. Both of them shall be useful. One thing that I forgot to mention in the general IIR structure or the all pass IIR structure is that if you fabricate a chip and take outputs from S_1, S_2, S_3 and so on, this becomes a multifunction chip. For example, S_1/X_1 is an all pole filter; the numerator is 1. And S_2/X_1has one 0 and the same denominator as that of the desired transfer function, and so on; so it becomes a multifunction device. This is the beauty of DSP in that one chip can perform a variety of functions.