Digital Signal Processing

Prof. S. C. Dutta Roy

Department Of Electrical Engineering

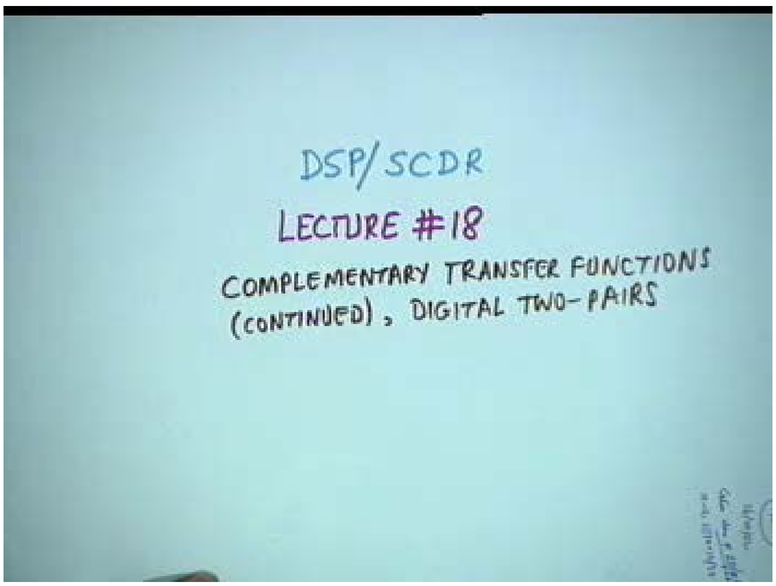
Indian Institute of Technology, Delhi

Lecture - 18

Complementary Transfer Functions

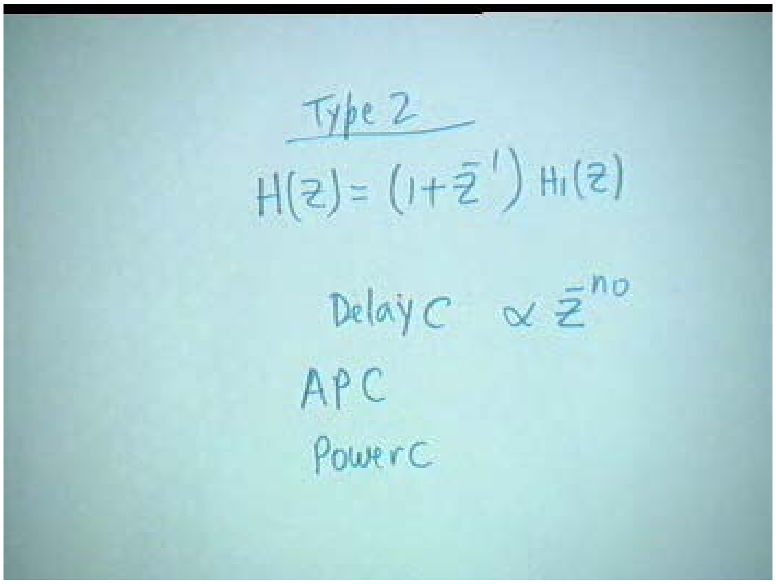
(Continued…) Digital Two-Pairs, Stability

(Refer Slide Time: 1:09)



This is the 18th lecture on DSP and we continue our discussion on complementary transfer functions and introduce an entirely new concept of digital two pairs. In the previous lecture, the 17th, we continued our discussion on linear phase FIR filters and also we gave a summary. We saw that type one linear phase, that is symmetrical impulse response and odd length, (distinguish between length and order: Order is one less than the length because we started at n = 0) is the most versatile. Symmetrical even length impulse response cannot achieve a high pass filter.

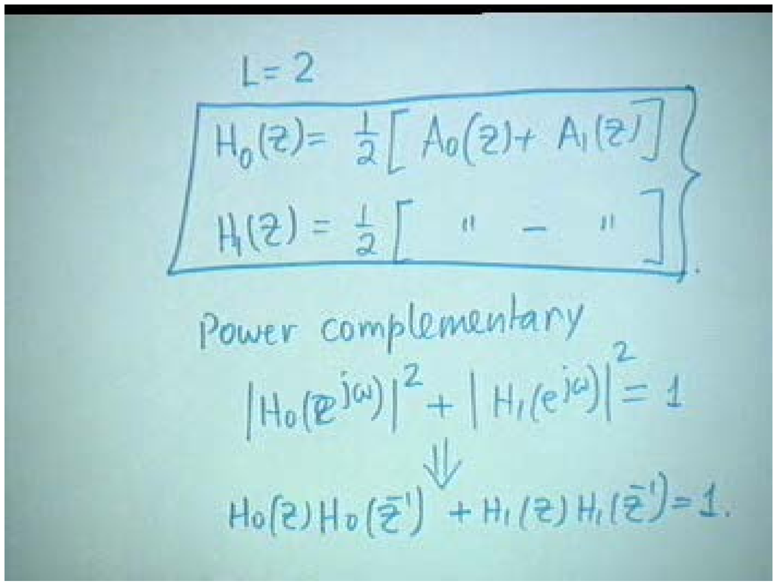
(Refer Slide Time: 02:13 - 5:09)



This is type 2, which has a transfer function of the form $(1 + z^{-1}) H_1(z)$. Type 3 has an anti-symmetrical impulse response and is of odd length; with this, you can neither have HPF nor LPF nor BSF. Type 4 has anti-symmetrical impulse response and is of even length; you cannot have a low pass filter or a band stop filter with this. And then we talked about the location of zeros of the transfer function; we saw that the complex zeros occur in quads, that is, in a group of four. Exceptions are zeros at + 1, – 1, + j and – j. Then we talked about complementary filters. Complementary filters are filters, the addition of transfer functions of which gives rise to a pure delay or all-pass or constant magnitude or constant power.

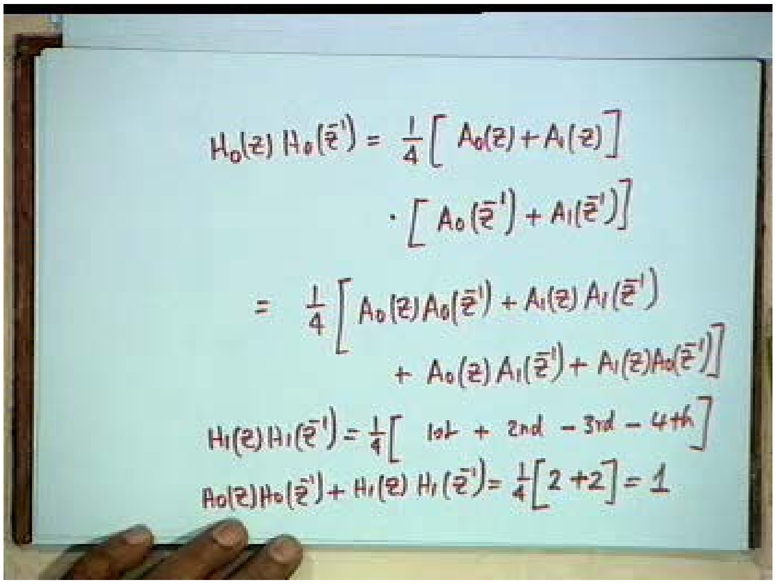
For example, if it is a delay complementary set, then the addition of the transfer functions should give rise to pure delay, may be with a multiplicative constant that can be more than 1, equal to 1, or less than 1. But in delay complementary filter, the sum of the transfer functions should be proportional to z^{-N}. In all-pass complementary, the sum of transfer functions should be equal to an all-pass function. And whenever we talk of all-pass functions, we maintain the discipline that the function is such that its magnitude is unity. Then we talked about power complementary filters, in which the sum of the magnitude squared of the transfer functions should be equal to the normalized value of 1. In magnitude complementary filters, the sum of the magnitudes should be equal to 1. So these are four types of complementary filters. And we looked at some of them.

(Refer Slide Time: 05:20 - 07:10)



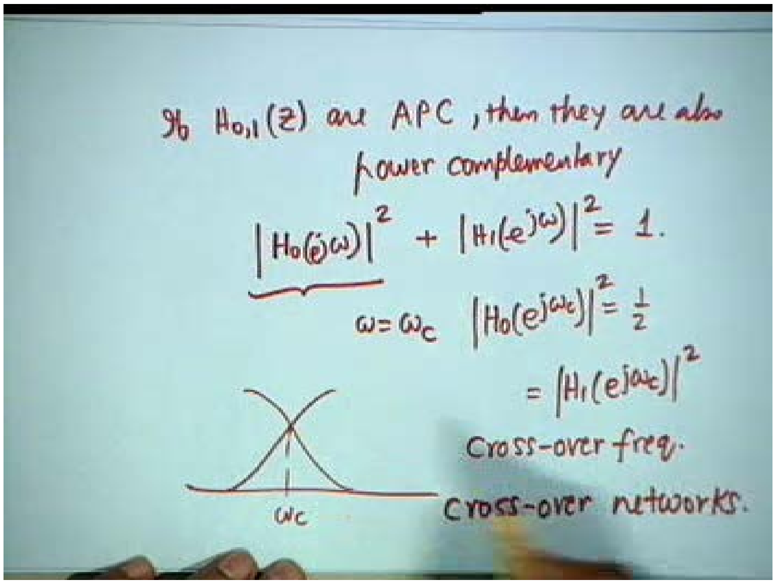
An interesting observation was that $H_0(z) = \frac{1}{2}[A_0(z) + A_1(z)]$ and $H_1(z) = \frac{1}{2}[A_0(z) - A_1(z)]$ are not only all-pass complementary, but they are also power complementary. For two power complementary filters, $|H_0(e^{j\omega})|^2 + |H_1(e^{j\omega})|^2 = 1$; this also implies, by analytic continuation, i.e. by extending this to the complete z plane, $H_0(z)\, H_0(z^{-1}) + H_1(z)\, H_1(z^{-1}) = 1$.

(Refer Slide Time: 07:10 - 9:52)



We now prove the observation that all-pass complementary filters are also power complementary. We have $H_0(z)\ H_0(z^{-1}) = (1/4)\ [A_0\ (z) + A_1(z)] \times [A_0\ (z^{-1}) + A_1(z^{-1})$, Now if you multiply them out then you get ¼ $[A_0(z)\ A_0(z^{-1}) + A_1(z)\ A_1(z^{-1}) + A_0(z)\ A_1(z^{-1}) + A_1(z),\ A_0(z^{-1})]$. And it is very easy to comprehend that $H_1(z)$, $H_1(z^{-1})$ will have the same form except that the last two terms will have negative signs. So the third and fourth terms shall cancel and I shall have $\frac{1}{2}$ $[A_0(z)\ A_0(z^{-1}) + A_1(z)\ A_1(z^{-1})]$. Thus we have proved a very important theorem, namely that if $H_0(z)$, and $H_1(z)$ are all-pass complementary, then they are also power complementary. Therefore these two transfer functions $H_0(z)$ and $H_1(z)$ are very important transfer functions and they solve many problems that arise in practical applications of DSP.

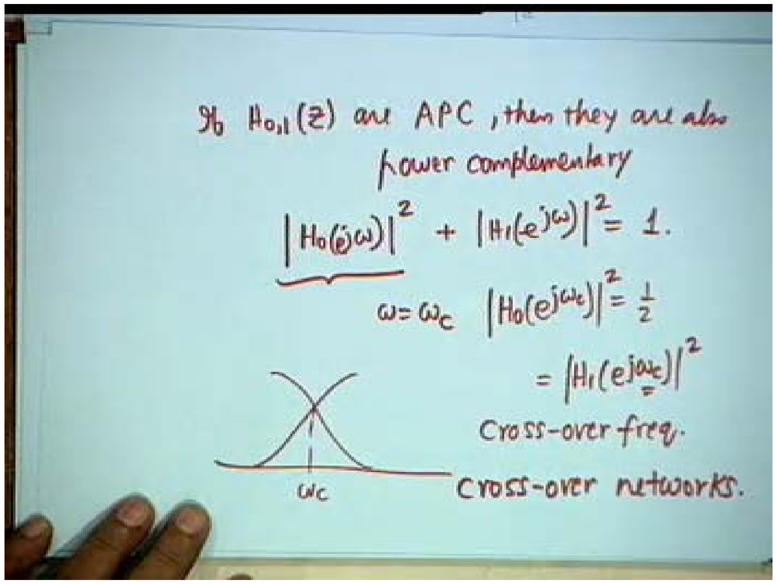
(Refer Slide Time: 09:55 - 13:29)



Let us look at their realization. Before we go to the realization, let us pose a question. If three transfer functions are all-pass complementary, are they also power complementary? Find the answer yourself. In the present case, we have the sum of the two powers equals to one. Now, if at a particular frequency $|H_0|^2 = ½$, let us say at $\acute{\omega} = \acute{\omega}_c$, then does it not follow that $|H_1 e^{J \acute{\omega}_c}|^2$ should also be equal to ½?

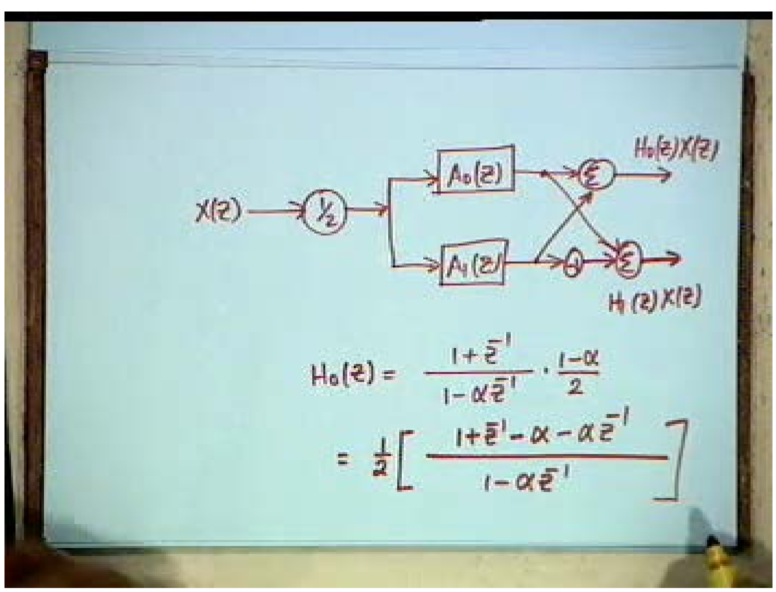
In other words, at $\acute{\omega}_c$, the two transfer function magnitudes are identical and therefore, there is a cross over of these two responses at $\acute{\omega} = \acute{\omega}_c$, and this is called the Cross-over Frequency. And such transfer functions which are all-pass complementary, as well as power complementary, are called cross-over networks or cross-over filters. They have a very important application in digital audio and we shall come to this a little later. This cross-over frequency is important. You notice that this is 3dB frequency of either filter.

(Refer Slide Time: 13:46 - 14:16)



We now look at the possible realization of $H_0(z)$ and $H_1(z)$. We have a given input X(z); we multiply by ½ first. Multiplication by ½ is no multiplication; it is a shift. And then we have two channels: One of them has the all-pass filter $A_0(z)$ and the other has the all-pass filter $A_1(z)$; there are two parallel channels. And if you add the outputs of the two, you get $H_0(z)$ times X(z).

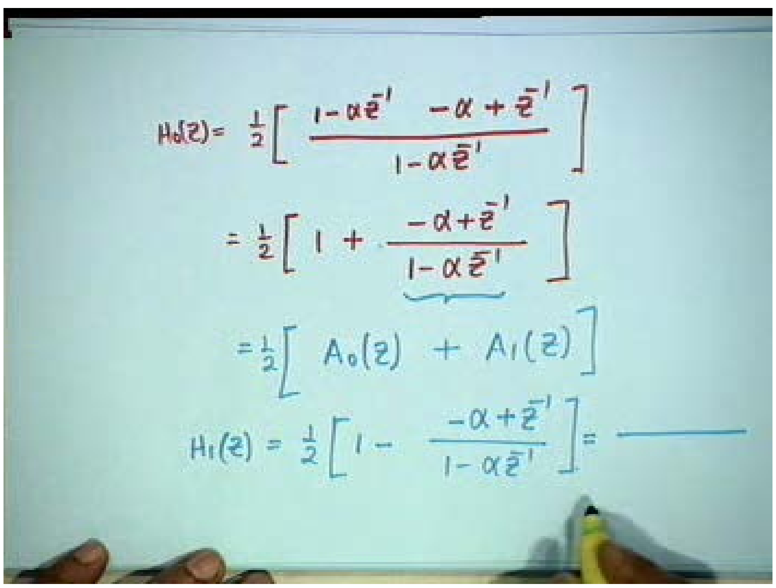
(Refer Slide Time: 14:28 - 17:29)



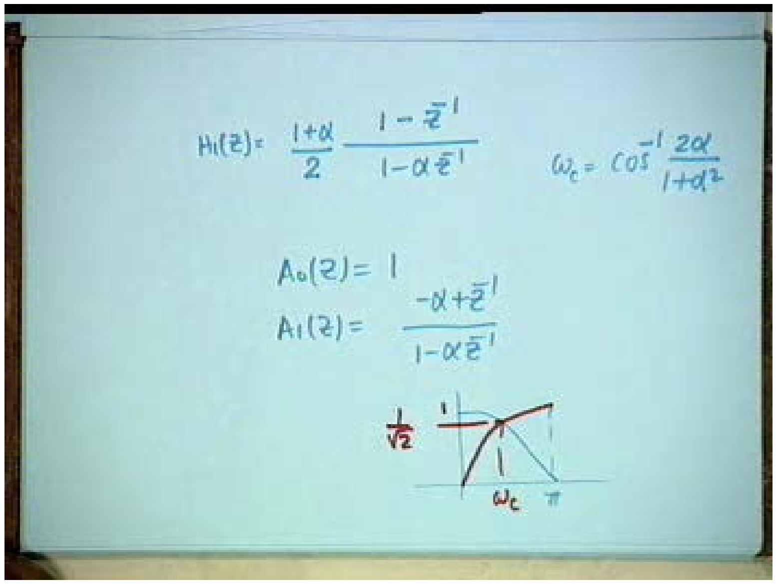
The transfer function from X(z) to the upper output would be simply ½ [$A_0(z) + A_1(z)$], that is $H_0(z)$. And if you multiply $A_1(z)$ output by – 1, (once again, it is no multiplication; it is only change of the sign) and then add the two, the output will be simply $H_1(z)$ X(z). Thus for doubly complementary transfer functions H_0 and H_1, all that you require for realization are two all-pass filters. As an example, suppose $H_0(z)$ is the familiar first order IIR low pass filter, $\frac{1-\alpha}{2}$ $(1 + z^{-1})/(1 - \alpha z^{-1})$. Now this can be written as $\frac{1}{2}[1 + (z^{-1} - \alpha)/(1 - \alpha z^{-1})]$. You can easily identify $A_0(z)$ as 1 and $A_1(z)$ as $(z^{-1} - \alpha)/(1 - \alpha z^{-1})$, which is clearly all-pass. Then $H_1(z)$ will be $\frac{1}{2}[1 - (z^{-1} - \alpha)/(1 - \alpha z^{-1})]$, which, on simplification gives $\frac{1+\alpha}{2}$

$(1 - z^{-1})/(1 - \alpha z^{-1})$. Can you recognize this as the elementary first order IIR high pass filter?

(Refer Slide Time: 17:34 - 19:32)



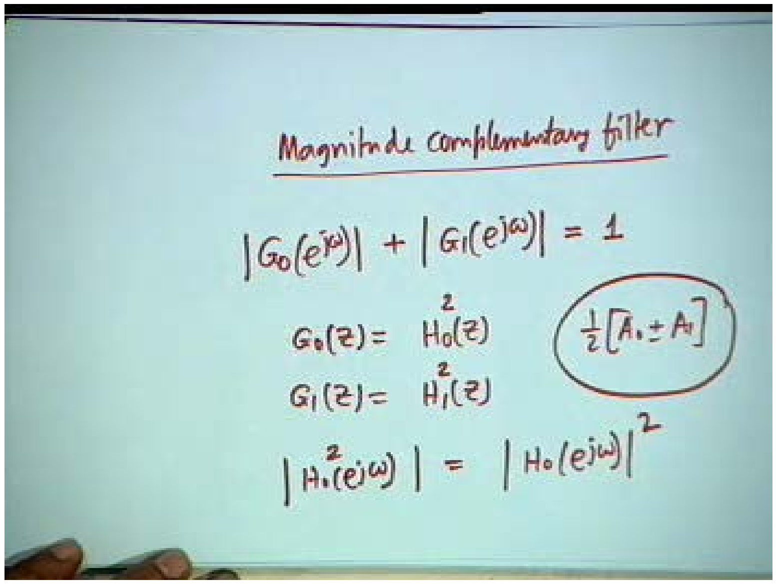
(Refer Slide Time: 19:36 - 23:46)



The cut off frequency of both these filters is $\acute{\omega}_c = \cos^{-1} [2\alpha/(1 + \alpha^2)]$ and therefore our simplest first order IIR low pass filter and the simplest first order IIR high pass filter are all-pass as well as power complementary. $\acute{\omega}_c$ is the cross-over frequency. All that you have to do now is to realize $A_1(z)$, because $A_0(z)$ is a straight connection. We shall see that this realization can be done using only one delay. We shall show later that we can realize $A_1(z)$ with only one multiplying constant and that is α. So it is a beautiful set of filters: first order low pass as well as high pass, realized with only one delay and one multiplier. It is the most economic realization of two doubly complementary filters.

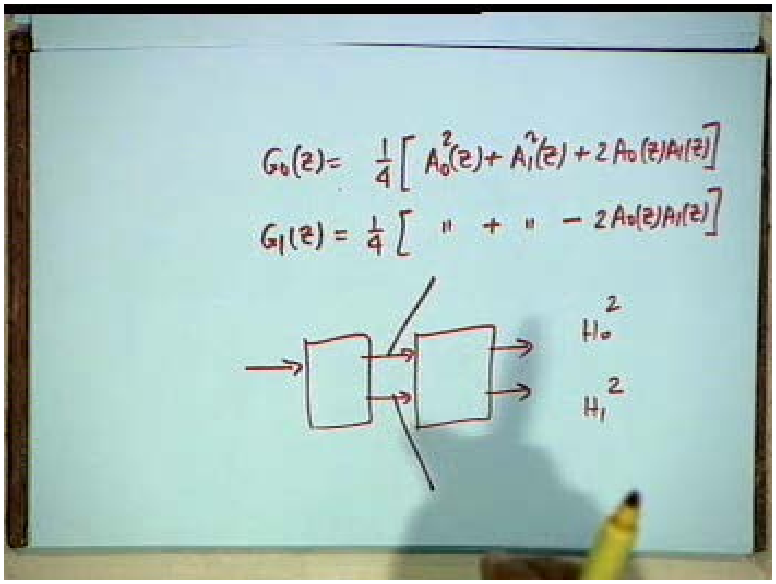
This is one of the most used doubly complimentary filters in the digital audio industry. What they do in producing a stereo music or speech, is to record a single signal and then they separate them into two, where one accentuates the high frequencies, the other accentuates the low frequencies; this is the simplest digital stereo. There may be other processing in between. For example, if you want to enhance the base sound in a tabla, you have to use other filters. The two cross over networks can be implemented most economically by a single chip which contains a realization of first order all-pass filter with one delay and one multiplier. We next consider a magnitude complementary filter.

(Refer Slide Time: 23:54 -26:39)



Consider a set of two filters $G_0(z)$ and $G_1(z)$ such that $|G_0e^{J\omega}| + |G_1(e^{J\omega})| = 1$. Then G_0 and G_1 form a set of magnitude complementary filters. Now, a very simple example of this would be as follows. You take the doubly complementary filters $\frac{1}{2}[A_0(z) \pm A_1(z)]$. And if you make $G_0(z) = H_0^2(z)$ and $G_1(z) = H_1^2(z)$, then you know that $|H_0(e^{J\omega})|^2 + |H_1(e^{J\omega})|^2 = 1$. Also, $|G_0(e^{J\omega})| = |H_0^2(e^{J\omega})| = |H_0(e^{J\omega})|^2$. Similarly, for $|G_1(e^{J\omega})|$. Hence, G_0 and G_1 will be magnitude complementary.

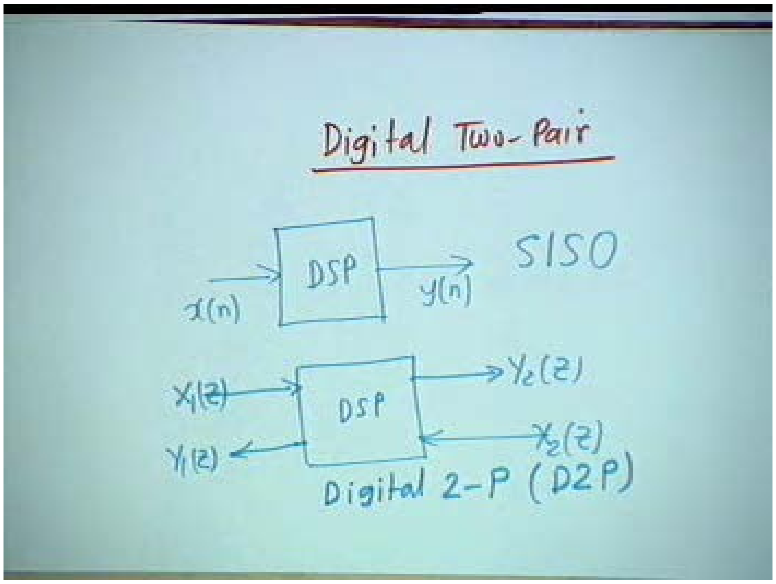
(Refer Slide Time: 26:42 - 30:10)



Are $G_0(z)$ and $G_1(z)$ all-pass or power complementary? Let us see. $G_0(z)$ is (1/4) $[A_0^2(z) + A_1^2(z) + 2A_0(z) + A_1(z)]$. And similarly, $G_1(z)$ = ¼ $[A_0^2(z) + A_1^2(z) – 2A_0(z)\ A_1(z)]$. They are magnitude complementary by design. Are they all-pass complementary? If we add them do you get an all-pass filter? Not necessarily, in general. Are they power complementary? No, they are not. Realizing a magnitude complementary filter is not a problem. You cascade two such blocks, the blocks which realize H_0 and H. you start with same all-pass filter $(– \alpha + z^{-1})/(1 – \alpha z^{–1})$ and we know how to realize H_0 and $H_{1,}$ just add one more of such block. In other words, it is a modular implementation. If you can make an IC chip for the doubly complementary filter, you just have to repeat the process.

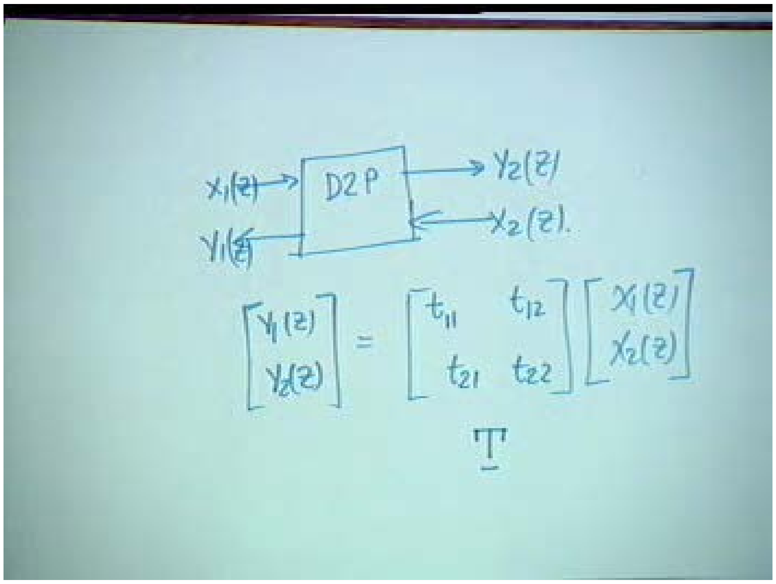
It is also a very economical process. If the IC processing fabrication steps have been standardized, you just repeat this without any change and therefore you can make the cascade of two such filters to get H_0^2 and H_1^2. In between, you get a low pass filter and a high pass filter and you get a set of magnitude complementary filters also. So it is a versatile chip having a number of output pins. One would be marked low pass and one will be marked high pass. You can also get an output from the first order all-pass filter. So you also get a first order all-pass, first order high pass, first order low pass, and in addition a set of doubly complementary filters and a set of magnitude complementary filters. We shall come back to complementary filters at a later date and we shall show why we pay so much importance to all-pass filters. You have seen one example here in which a versatile IC chip requires the design of a single all-pass filter. And if alpha is variable, all of these filters are variable filters. You get a single chip, then you program for α and you get a programmable versatile chip which gives you all these functions: low pass, high pass and all-pass. We will show later that you can also get a band pass and a set of magnitude complementary and doubly complementary filters. We next come to the concept of digital two pairs.

(Refer Slide Time: 31:17 -34:09)



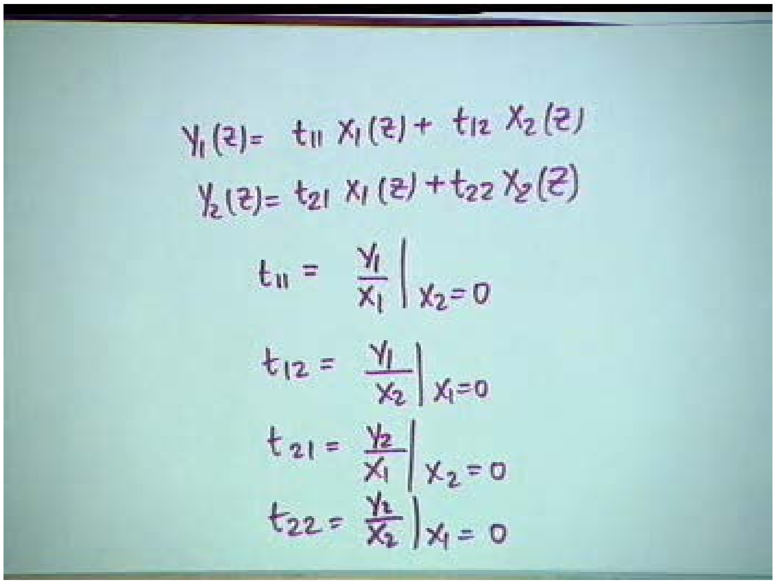
A digital two pair is defined in analogy with analog two ports. In an analog two port, you have an input applied between two terminals constituting a port and an output between two other terminals constituting the second port. But in a digital device, since we are only talking of numbers, you have an input x(n) and you have an output y(n). If a digital signal processor has one input and one output, it is an SISO system. Consider a DSP which has two inputs, and for convenience we shall show the inputs not in the time domain but in the z domain. The two inputs are $X_1(z)$ and $X_2(z)$. For reasons to be made clear a little later, we draw the two inputs on different sides. There is no port concept in DSP, you just extract some numbers or you feed some numbers. The schematic representation is shown in the figure. And the outputs are $Y_1(z)$ and $Y_2(z)$. Inside the block marked DSP we shall have multipliers, and adders and delays where delays stand for retrieval from the storage. Such a device having two inputs and two outputs is called a digital two pair. We shall represent it by D2P where P stands for pair and not port.

(Refer Slide Time: 34:12 - 36:04)



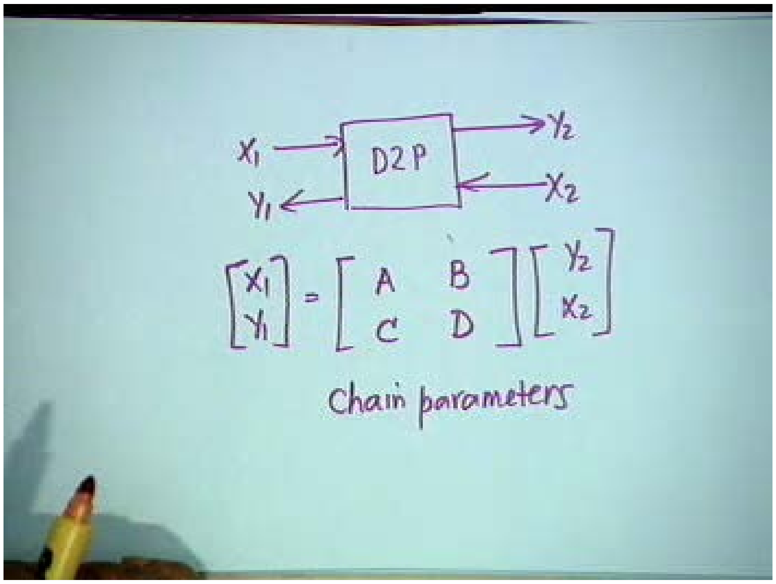
Let us consider a D2P having the defined inputs X_1, X_2 and outputs Y_1, Y_2; they of course are functions of z. In analog two port, we characterize by z-, y-, h- or ABCD parameters. A digital two pair, can be characterized in two different useful ways. In theory, two out of four variables can be chosen in six different ways, but the ones that are useful in practice are, Y_1 and Y_2 expressed in terms of the 2 inputs X_1 and X_2. The corresponding parameters are called transmission parameters, because they show how the two inputs are transmitted to the two outputs. You require four transmission parameters, and the corresponding 2×2 matrix is called the T matrix; the parameters themselves are denoted by t_{11}, t_{12}, t_{21}, t_{22}.

(Refer Slide Time: 36:05 - 37:345)



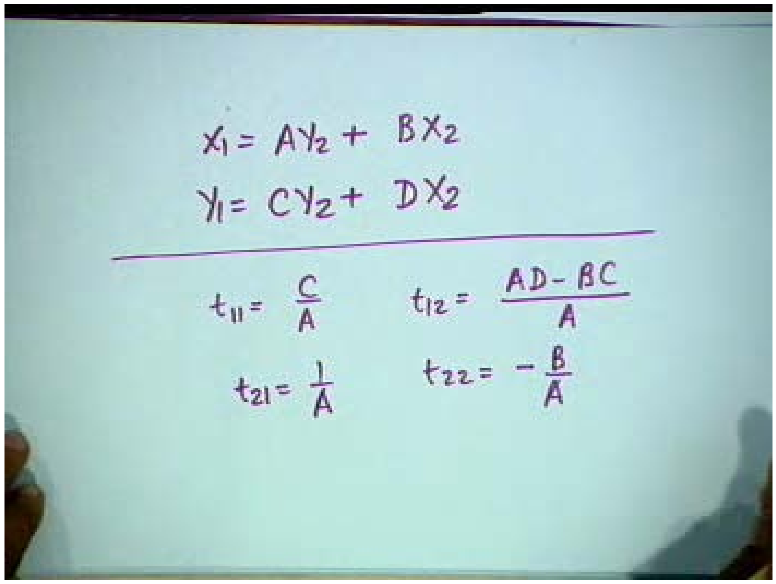
If we write the complete equations, we shall get $Y_1(z) = t_{11} X_1(z) + t_{12} X_2(z)$ and $Y_2(z) = t_{21} X_1(z) + t_{22} X_2(z)$. Exactly as in analog two ports you can define the parameters, t_{11}, t_{12}, t_{21} and t_{22}, as ratios of two variables with a constraint on a third variable. For example, t_{11} is Y_1/X_1 with $X_2 = 0$. If you go back to the schematic diagram X_2 is clamped to zero and Y_2 is left arbitrary; apply an X_1 and find Y_1. Similarly, t_{12} is Y_1/X_2 under the condition $X_1 = 0$; $t_{21} = Y_2/X_1$ with $X_2 = 0$ and finally, $t_{22} = Y_2/X_2$ with $X_1 = 0$. There is no concept of impedance or admittance here. These are pure numbers. The other way that one can characterize a D2P is as follows.

(Refer Slide Time: 37:41 - 39:57)



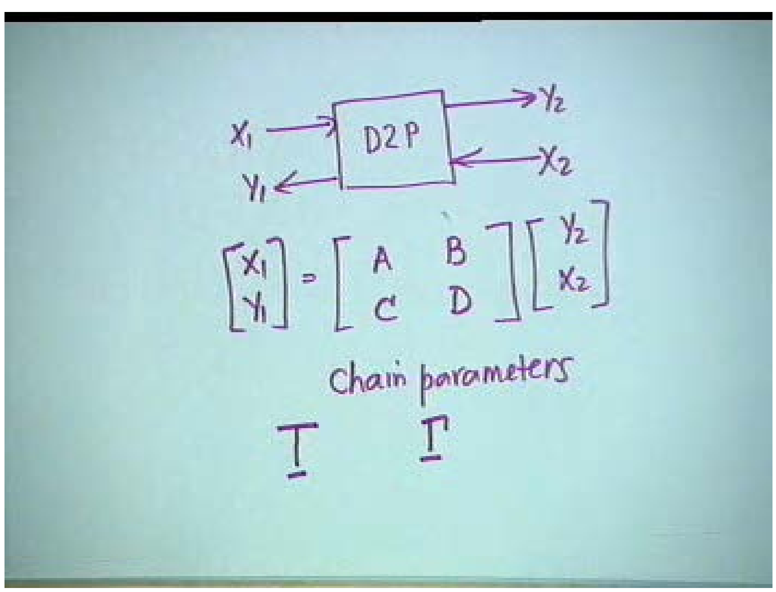
The set [X_1, Y_1] is expressed in terms of the set [Y_2, X_2]. In similarity with analog two ports, the corresponding parameters are called the chain parameters and denoted by ABCD. Why are they called chain parameters? It is because if you have a chain of D2Ps, then the overall ABCD matrix of the cascade shall be the multiplication of the individual ABCD matrices, written in the same order as they are cascaded.

(Refer Slide Time: 40:02 - 42:11)



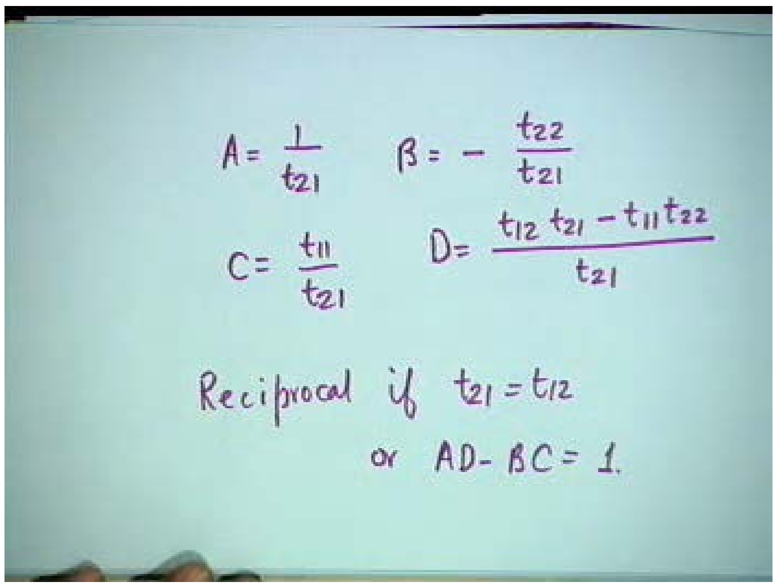
Explicitly, $X_1 = AY_2 + BX_2$ and $Y_1 = CY_2 + DX_2$; you can define or measure ABCD parameters. A, for example, shall be X_1/Y_2 with $X_2 = 0$ and so on. The symbol that is given for the ABCD matrix is capital gamma (Γ).

(Refer Slide Time: 40:31 - 40:58)

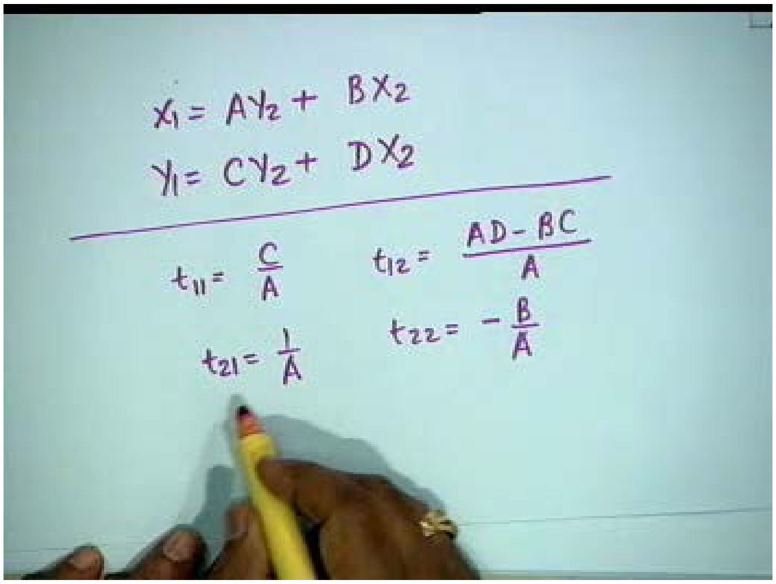


Do not confuse between capital gamma Γ and capital T. In capital Γ one of the hands is chopped off from T. Obviously since the variables are the same (X_1, X_2, Y_1, Y_2), these parameters should be inter-related to each other and one of the inter-relationships is that t_{11} is C/A. You can show this very easily. Also, t_{12} = (AD – BC)/A, t_{21} = 1/A and t_{22} = – B/A. Similarly the ABCD parameters can be expressed in terms of t parameters and the relationships are: A = 1/t_{21} (that should obviously be true because t_{21} is 1/A), B = –t_{22}/t_{21}, C = t_{11}/t_{21} and D = (t_{21} t_{12} – t_{11} t_{22})/t_{21}.

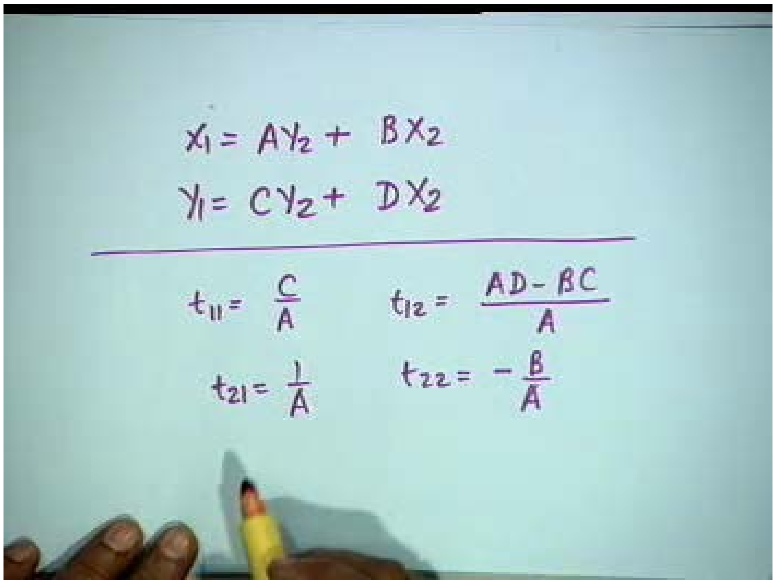
(Refer Slide Time: 42:16 - 44:35)



(Refer Slide Time: 42:23 - 42:25)

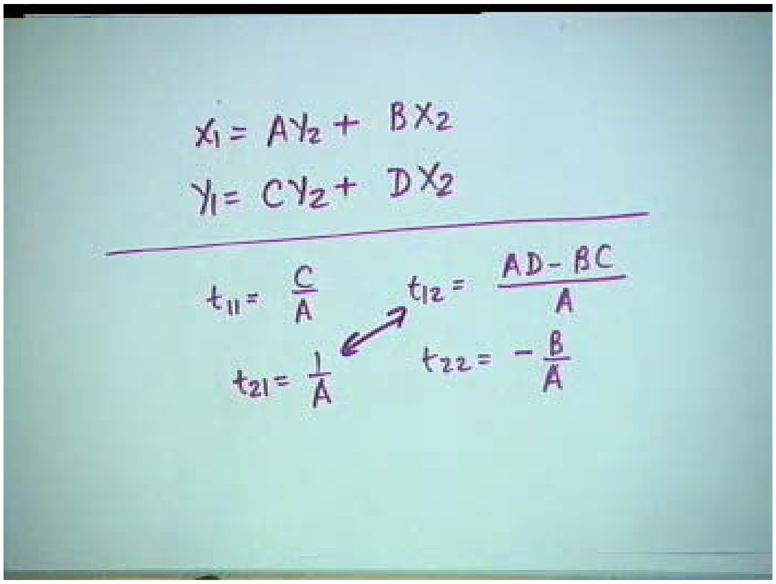


(Refer Slide Time: 42:37 - 42:52)



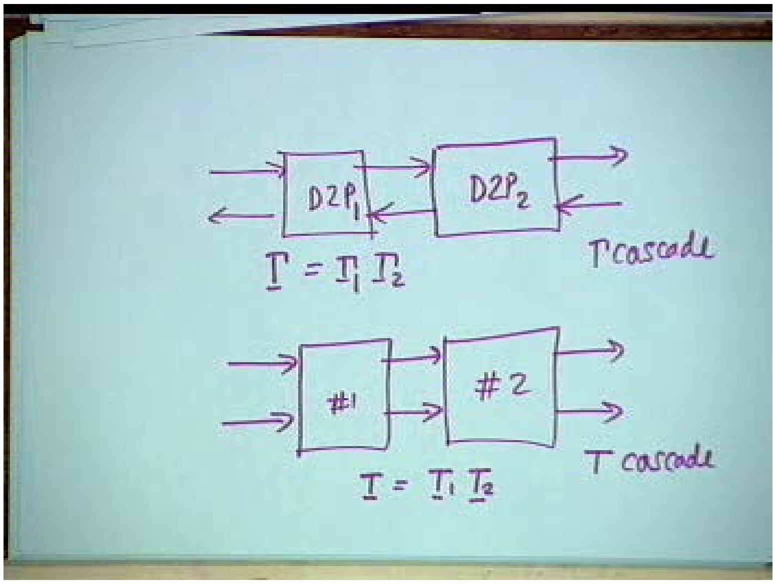
Here one thing you should observe is that the denominator is the same for all the four parameters; this is true in any conversion. Now, there is a concept of reciprocity as in analog two ports. In analog two ports, reciprocity says that if the input and output are interchanged then the ratio of the relevant variables should remain the same if the network is reciprocal. Here also, in terms of the t parameters t_{12} and t_{21} relate input and output and reciprocity demands that $t_{21} = t_{12}$; in terms of the ABCD parameters, reciprocity demands that AD – BC = 1.

(Refer Slide Time: 44:12 - 44:19)



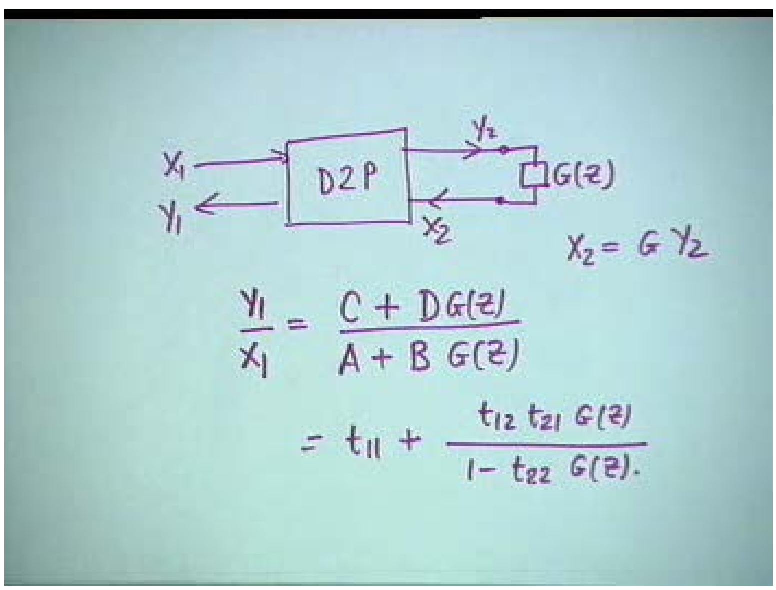
Now, exactly like analog two ports we can cascade digital two pairs in two useful ways.

(Refer Slide Time: 44:42 - 47:15)



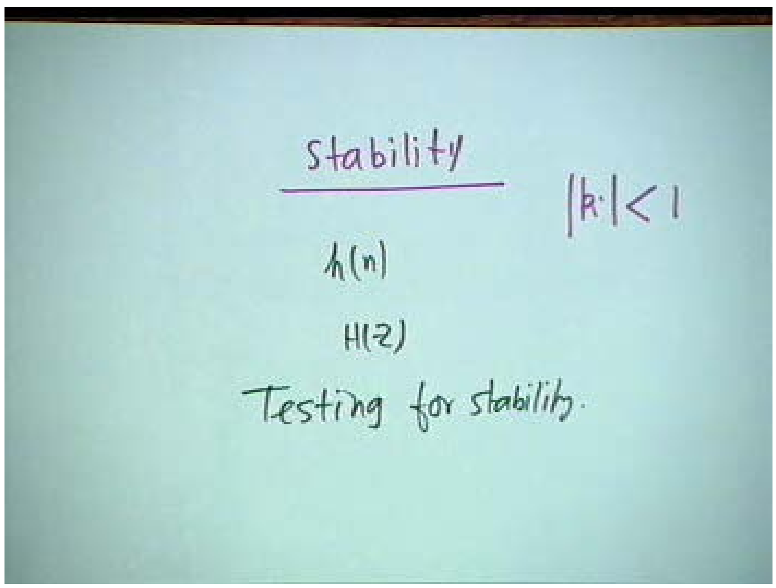
If you take a digital two pair, make a cascade as shown in the first figure, then Γ_1 and Γ_2 multiply to form Γ. This is why they were called chain parameters. But there is nothing sacred about drawing the input X_2 on the right hand side, I could also draw it on the left hand side. If I make a T-cascade as shown in the second figure, then how do you find the overall parameters? Which parameters shall be relevant? Obviously, transmission parameters. So the overall transmission parameter T shall be equal to the product of T_1 and T_2. Does the order of cascading matter? Yes, it does. In Γ cascade, $\Gamma = \Gamma_1 \Gamma_2$ and not $\Gamma_2 \Gamma_1$ because matrix multiplication is not commutative. Similarly, for T-cascade, $T = T_1 T_2$ and not $T_2 T_1$. We shall see later how cascading of D2Ps are useful. One example we have already seen in cascading of two doubly complementary filters giving rise to a magnitude complementary filter set. Finally, we come to the concept of a terminated digital two pair.

(Refer Slide Time: 47:29 -50:02)



I have a digital two pair whose input is X_1 and output is Y_1 with Y_2 connected to X_2 via an SISO system G(z). This is a terminated D2P. Clearly, $X_2 = GY_2$. So if in the two linear equations describing the t and ABCD parameters, you inject this constraint, then you shall be able to find Y_1 and X_1, show that $H(z) = Y_1/X_1 = [C + D\,G(z)]/[A + B\,G(z)]$. Or if you use the t parameters then $H(z) = t_{11} + [t_{12}\, t_{21}\, G(z)]$ divided by $[1 - t_{22}\, G(z)]$. The concept of digital two pair shall be utilized to derive canonic realizations of all-pass filters at some later date. What is a canonic realization? Canonic realization is a realization which uses the minimum number of delays as well as multipliers. The example that I already cited is a first order all-pass filter realized with one delay and one multiplier; it is a canonic realization. We shall also use digital two pairs at a later date in some other context.

(Refer Slide Time: 50:08 - 55:17)

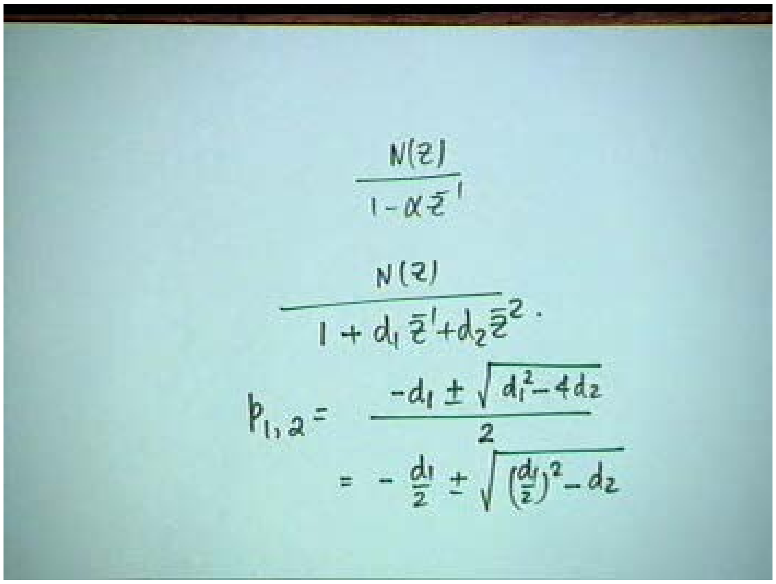


Finally, in this lecture, we go back to a discussion of stability. We have examined stability in two contexts; one is, that an LTI system is stable if and only if the impulse response is absolutely summable. That is, summation mod of h(n) should be less than infinity. The other context we have seen is that the poles should be inside the unit circle. That is, if the poles are at p_i in the z plane, then |pi| should be bounded by unity for all i. Now, this is okay, but as far as testing is concerned, a simple alternative is desirable, particularly if you are given a very high order IIR transfer function. Do you ever have to test an FIR for stability? No, because all poles of FIR are at the origin. Origin is deep inside the unit circle. So FIR is unconditionally stable. This is one great advantage of FIR and the other great advantage is that it can be linear phase. These are the two virtues for which the FIR is given great respect. But at the same time, every positive thing comes with its own disadvantage.

Disadvantage is that you have to increase the complexity; you have to use a high order FIR to achieve what can be achieved by a second order IIR. So there are positive and negative points, and the two together make life comfortable. Now, we have considered stability in terms of two concepts: one is in time domain in terms of h(n) and the other in the frequency domain from the location of the poles in the z plane. We now want to see how to test for stability. If you are given a high order IIR filter, if you want to test in the time domain, then you have to find h(n). And finding h(n) means that you have to invert the z transform transfer function and the inversion is a laborious process, particularly if you have multiple poles. Or alternatively, you can have the transfer function H(z) and factorize its denominator; factorizing the denominator itself requires another sub routine, root finding program. Root finding programs have their own disadvantages, for example, numerical inaccuracy. You cannot find roots up to infinite number of decimals; you truncate somewhere. Then you go and happily design the filter and the filter starts oscillating because of quantization errors! In general, numbers cannot be represented exactly with a finite number of bits.

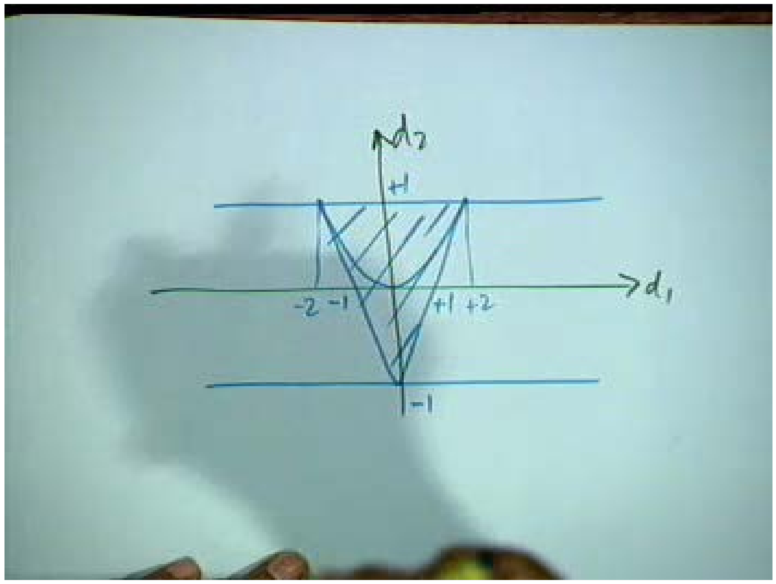
Now, besides implementation, root finding itself is a laborious process and then you have to test whether the root is bounded by unity or not. If it is a complex pole then you have to find the magnitude and see whether it is inside the unit circle or not. But then, testing for stability becomes very easy through an algorithm in terms of all-pass filters. That is, given H(z) to be tested, we first construct an all-pass filter and then you shall see that by a simple algorithm, the derivation of which may not appear that simple but, you can test the stability without finding the roots. Let us first consider the simplest possible first order IIR filter.

(Refer Slide Time: 55:29 - 56:56)



Stability testing is no problem, you have $1 - \alpha z^{-1}$ in the denominator. The pole is at $z = \alpha$ and all we have to do is to find whether α is inside or outside the unit circle. But suppose you have a second order IIR filter, $N(z)/(1 + d_1z^{-1} + d_2z^{-2})$. How do you test the stability of this? Obviously this is also not a problem, you can find poles as $p_{1,2} = -(d_1/2) \pm$ square root of $[(d_1/2)^2 - d_2]$. Now, once again, you will have to find these roots and again numerical accuracy or inaccuracy does come in. But given d_1 and d_2, can you say immediately whether the filter is stable or not? This would be possible if we can have a plot of d_2 versus d_1.

(Refer Slide Time: 57:00 - 59:01)



If we can demarcate a region in this d_1 d_2 plane, which is a parameter plane, such that inside that region, the second order filter is stable, then you do not have to find the roots. Given d_1 and d_2 you can immediately say whether the system is stable or not. We shall show in the next lecture that this region is bounded by + 1 and – 1 on both d_1 and d_2 axes, and that the region has the shape of a triangle, as shown in the figure. We shall derive that if the point (d_1, d_2) lies in this triangular region, then the filter is stable. If d_1 and d_2 or both go outside this triangle, it would be unstable. Not only that, we shall show that if d_1, d_2 lie within the hatched parabolic region but inside the triangle, then the poles would be real. The derivation is not available easily in text books and I shall do it completely in the next lecture. This is where we close today.