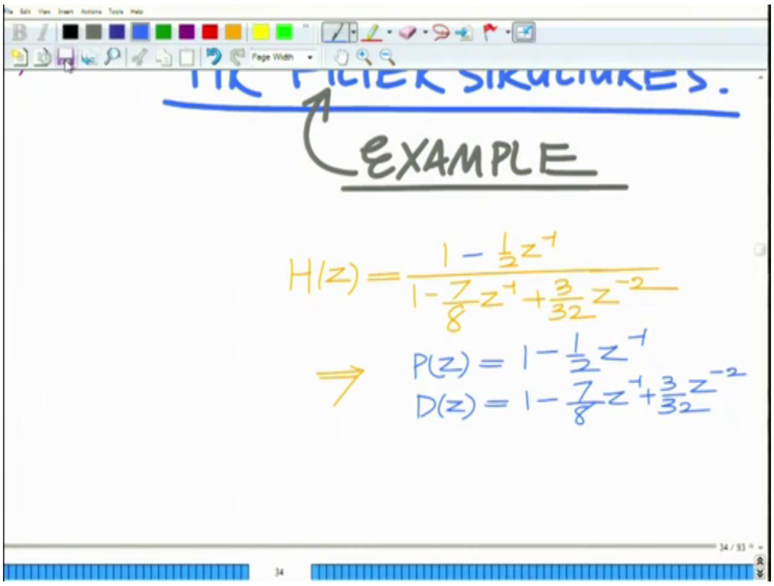
Principles of Signals and Systems
Prof. Aditya K. Jagannatham
Department of Electrical Engineering
Indian Institute of Technology, Kanpur

Lecture – 73
IIR Filter Structures: Example

Hello welcome to another module in this massive open online course. So, we are looking at IIR filter structures. And their implementation we have seen various forms such as the direct form one, direct form two, direct form one transpose and direct form 2 transpose. Let us now look at several examples to understand this the implementation ok.

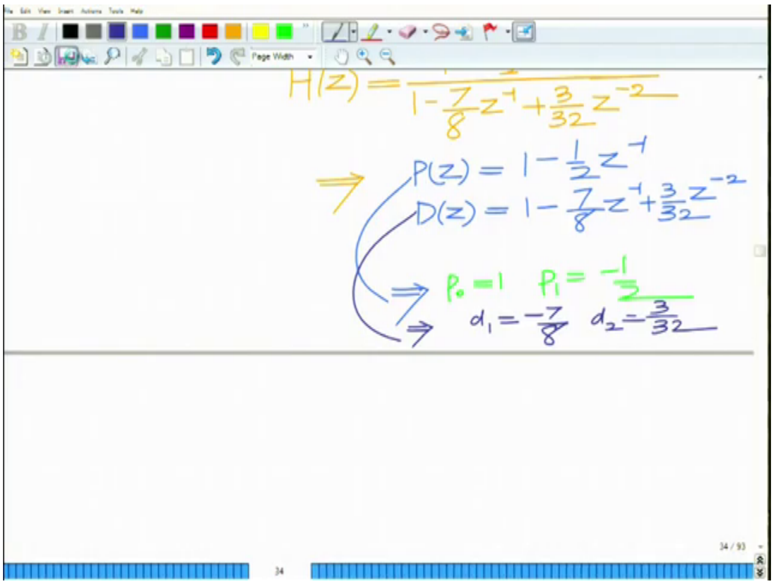
(Refer Slide Time: 00:35)



So, want to look at FIR filter structures. I am sorry IIR filter structures and consider and we want to do an example problem to understand this better.

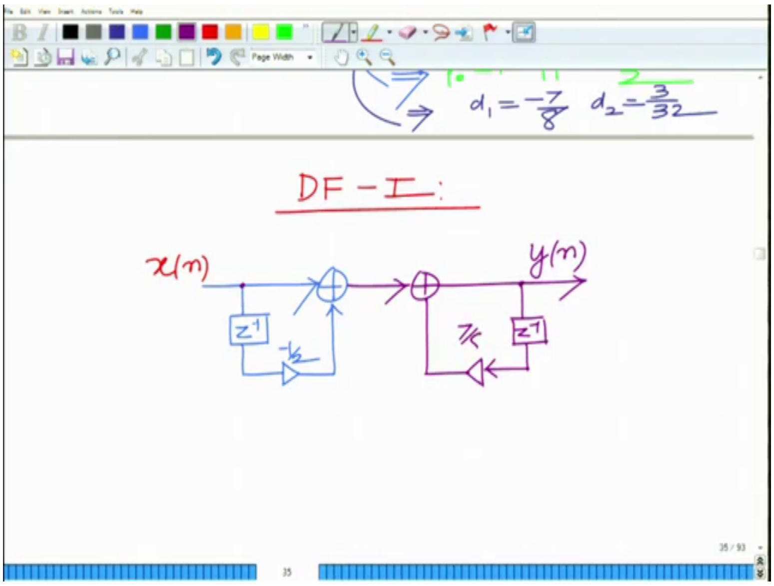
So, what we want to do is let us consider this example where we have our H z equals 1 minus half z inverse over 1 minus 7 by 8 z inverse plus 3 by 32 z raised to minus 2. So, this implies you will look at this our numerator polynomial is P z which is 1, I am sorry this is 1 minus; minus half z inverse and D z equals 1 minus 7 by 8 3 by 32 z raised to minus 2. So, now if you look at P z, now what you can see is this implies basically that P naught equals 1 and P 1 equals minus half and if you look at the D z polynomial you can see

(Refer Slide Time: 02:17)



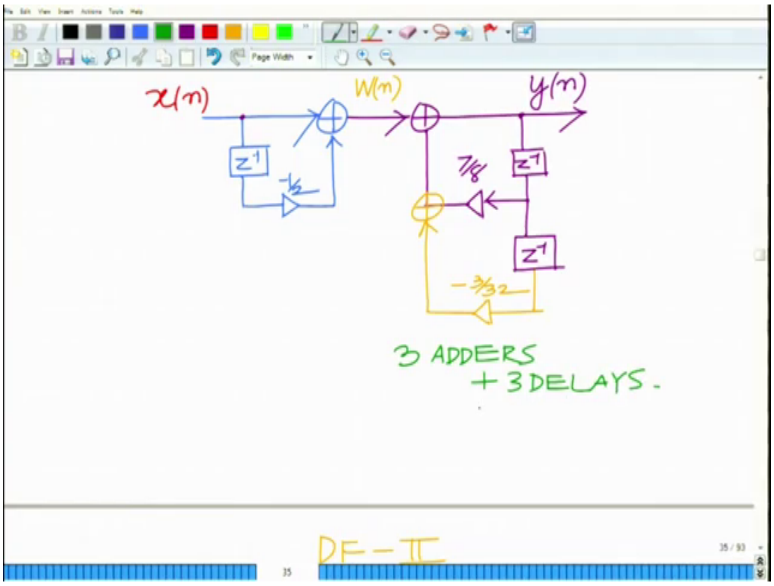
this implies that well the d naught is 1, d 1 equals minus 7 by 8 coefficient of z inverse and d 2 equals 3 by 32. And we can now develop the several representations. Remember we have seen 4 representations; DF 1 direct form 1, direct form 2, direct form 1 transpose and direct form 2 transpose and let us now develop the structure for each of these different realizations ok..

(Refer Slide Time: 03:02).



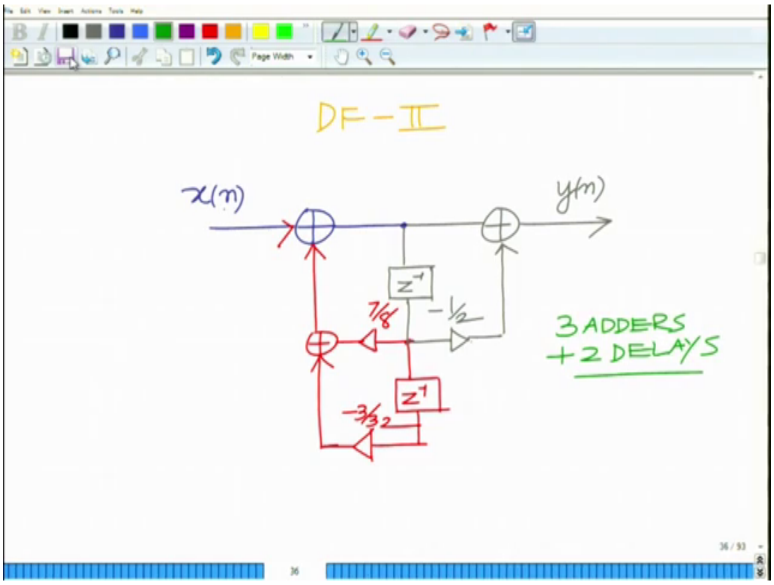
So, we have DF 1, DF 2 ok. So, our DF 1 direct form 1 will be as follows. That will be you have x n fed to an adder and you will have z inverse and this will go to the gain of minus half and come into this adder, this will be x n minus half, x n minus 1; that will be given that will be w n. Remember this is the DF 1; the output of this goes through another adder. Finally, you have y n and you have the branch here that goes through z inverse and the gain here is minus d1 that is 7 by 8. Further you have another delay. And this goes through another gain that is and here you have another adder and this will be minus.

(Refer Slide Time: 04:09)



So, we have minus d1 and then we have minus d2 ok. So, and this will be your DF 1 realization corresponding to the system and this is your w n ok. And now we want to develop the so, this is basically your DF 1 is a direct form one realization for the given example of the transfer function ok. Now let us look at the direct form 2; direct form 2 remember again is obtained by interchanging the branches and then merging the common delays in this case all right, in the case of the direct in case of the normal realization ok.

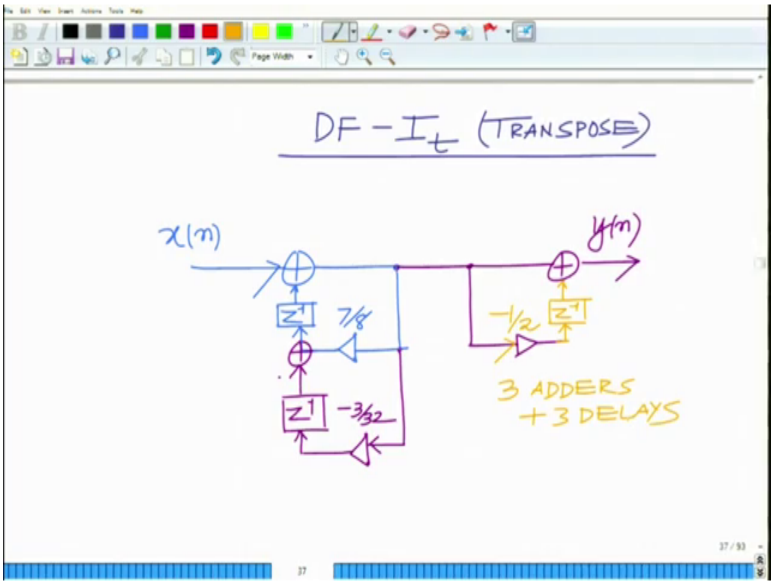
(Refer Slide Time: 05:13)



So, we have the direct form 2; the DF 2 realization and that is given as follows. So, we have x n; remember the delays in this along the central path. So, this will be z inverse minus this is minus this is just P1 that is your minus half correct and this will go as input to the adder and on the other side you have minus d1; that is minus 7 by 8. So, this will be added to x n and then you have another delay here and you have you have another delay here. And you have the corresponding gain minus 3 by 32, I am sorry this will only be 7 by 8; this is minus d1, this is minus D 2 and this will be fed to the adder above all right.

And this is basically you can see you have 3 adders and 2 delays. So, this is so this the DF 1 has a 3 adders and 3 delays and this you can see the DF 2 realization has 3 adders, but only 2 delays. This has 3 adders and only 2 delays. So, you can see it results in a significant saving because delays are typically difficult to implement. So, this results in a saving of one delay element ok.

(Refer Slide Time: 08:11)

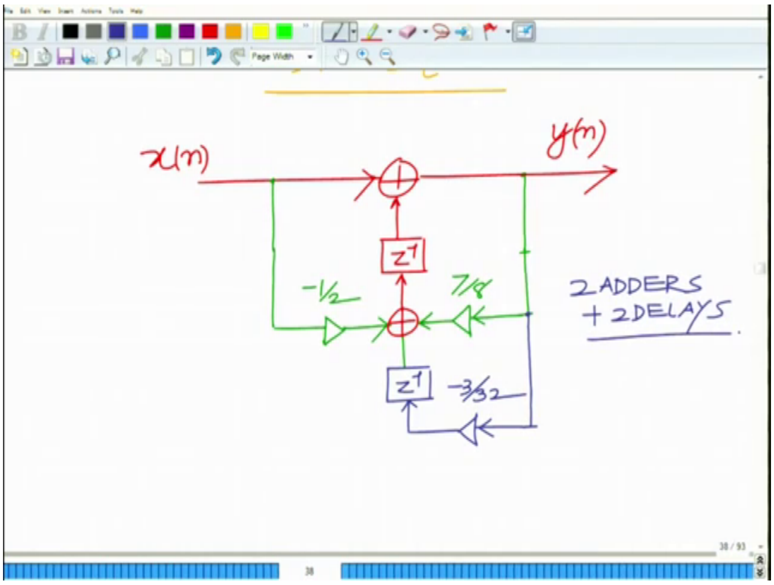


And now let us look at the transpose versions. So, we have the DF 1 transpose DF 1 transpose we have again. Let us say you have your x n that will be you have the adder element and the adder followed by this. So, let me just make a little room here. So, you have x n adder and then you have to pass it through the gain stage. So, this is a delay and here you have the gain stage and the gain in this case remember this will be minus d1 that is 7 by 8 and that is also passed through another gain stage; that will be so, here you have an addition and that will be passed through. And here you have another delay and another gain stage that corresponds to minus d2 that is minus 3 over 32 and this will be here ok.

And therefore, what you have and then this is your w n all right and this is now passed through another gain stage that is there is an adder here and you have y n and you have another delay element here. This will be, just draw it a little bit more and this will be minus half and therefore, this is your transpose. And this can also you can also see that this also has 3 adders plus 3 3 adders plus 3 delay elements. So, you can also see this also has 3 adders and 3 delay elements.

And now finally, let us see the direct form 2 transpose which is obtained by interchanging the branches and the direct form 1 transpose and then merging the adders and as well as the delay elements.

(Refer Slide Time: 11:25)



So, we have the DF 2 transpose which can be derived as x n adder that gives y n ok; z inverse and you have here another adder and from here what you have you have the gain. This is minus d1 7 by 8 and this gain here this gain here is minus half that is your P1 and finally, you have here z inverse. I am sorry this has to be just slightly corrected; this has to be your the gain stages are given to the adder ok.

And this has a gain 7 by 8 and the gain minus half and finally, what I have over here is I have another delay element z inverse and I have another gain that is minus 3 by 32 and I have over here correct and this is your gain. So, this is minus d1, minus d2 is minus 3 by sorry 32, minus 3 by 32 and yeah. So, this is minus half z inverse adder is minus 3 by 32 and this is your DF 2 transpose form. And you can see this has the fewest number, this has 2 adders plus 2 delays ok.

So, this has the minimum this has the representation that gives the minimum number of adders and delay. So, let me just briefly check this. So, I have my DF 1 is minus half 7 by 8 minus 3 by 32 that is correct this is 7 by DF 2 minus 7 by 8 minus half minus 3 by 32; that is the DF 1 transpose which is 7 by 8 minus 3 by 32 minus half and then finally, the DF 2 transpose which has a minus half 7 by 8 minus 3 by 32 and that also seems correct ok. So, we have 4 representations; the direct forms, all right because here, the coefficients are directly given in terms of the coefficients of the actual polynomials all right.

So, you have the direct form direct form 1, direct form 2, direct form 1 transpose, direct form 2 transpose. Direct form 1 transpose has the minimum number of delay elements, direct form 2 trans direct I am sorry direct form 2 has the minimum number of delay elements, direct form 2 transpose has the minimum number of adders as well as delay elements all right.

And we have seen an example we also demonstrated in the previous modules we have also seen the theory to derive these structures and now we also demonstrated the construction of these structures using a practical example all right. So, let us stop here and we will look at other forms other IIR structures in the subsequent modules.

Thank you very much.