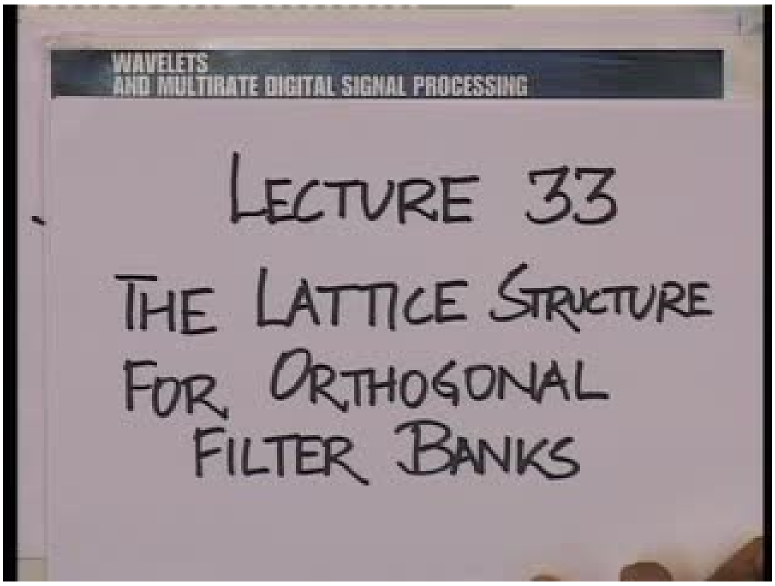
Advanced Digital Signal Processing – Wavelets and Mutlirate
Prof. V. M. Gadre
Department of Electrical Engineering
Indian Institute of Technology, Bombay

Model No # 01
Lecture No # 33
The Lattice Structure for Orthogonal Filter Banks

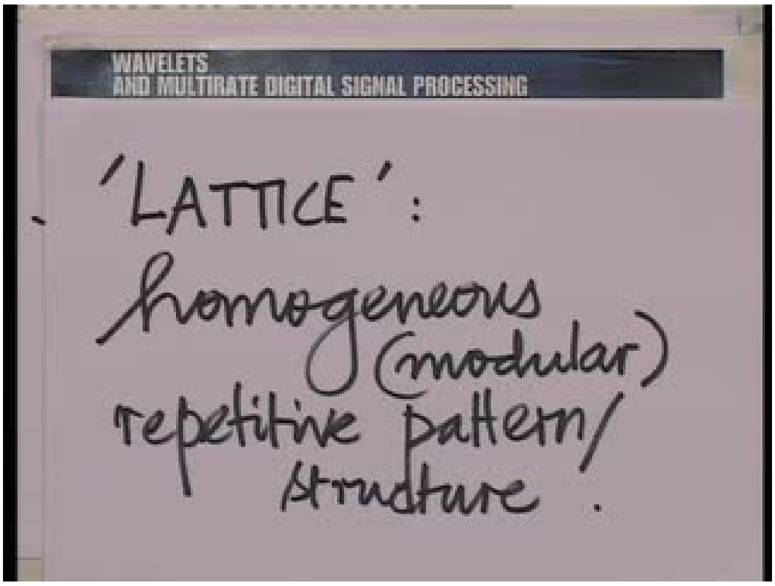
A warm welcome to the thirty-third lecture on the subject of Wavelets, Multirate and Digital Signal Processing. We had brought out the noble identities (quote unquote) the noble identities in the previous lecture, where the aim was to replace a cascade of down samplers or up samplers and filters with equivalent structures, in which these elements were interchanged in order.

Now, in the previous lecture, the aim of interchanging those components was to bring out the wave packet transform, to bring out an equivalent filter, which could represent the coefficients, in terms of an expansion of the wave packet basis, in terms of higher order sub spaces. What I mean by that is, when we take two steps of decomposition, so, we go from v 2 to v 1 and then to v 0. If we simply use the discrete wavelet transformation, we go from v 2 to v 1 and w 1 and from v 1 to v 0 and w 0 and we finally, end up with v 0 w 0 and w 1. In the wave packet transform, we decompose w 1 again into w 1 0 and w 1 1 and what we do using this pair of noble identities, is to establish a set of coefficients that relate the basis in these four sub spaces at the same level, w 1 0, w 1 1, w 0 and v 0 to the basis of the space v 2.

(Refer Slide Time: 02:44)



(Refer Slide Time: 03:07)



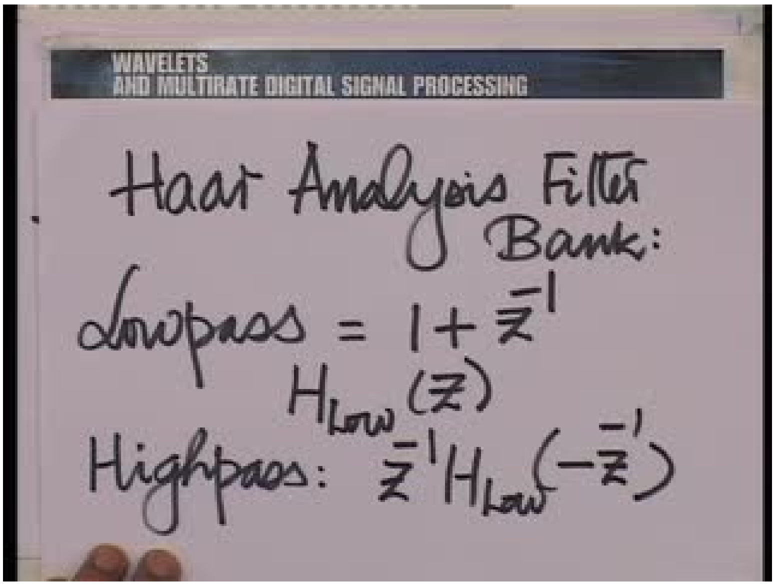
Now, we had also hinted, in the previous lecture, the same noble identities could also be used to bring in computational efficiency. And in fact, we are going to build on that idea in greater depth today. And therefore, in the lecture today, we shall be looking at a structure, which makes it efficient to realize orthogonal filter banks called the lattice structure. Now, the word lattice is a uniform term used for a homogeneous repetitive pattern, a homogeneous modular repetitive pattern or structure.

In fact, the word lattice has been used more commonly in the context of material sciences. So, there we know that, we could think of the composition of materials in terms of lattices, periodic repetitions of certain arrangements of atoms and molecules, for example. Now, we can carry that analogy a little further, and bring it into signal processing.

So, in signal processing, the idea of a lattice could be essentially, to repeat certain patterns involving the basic elements, namely delays, multipliers, adders, in such a way that, we can realize a higher order structure from a lower order one.

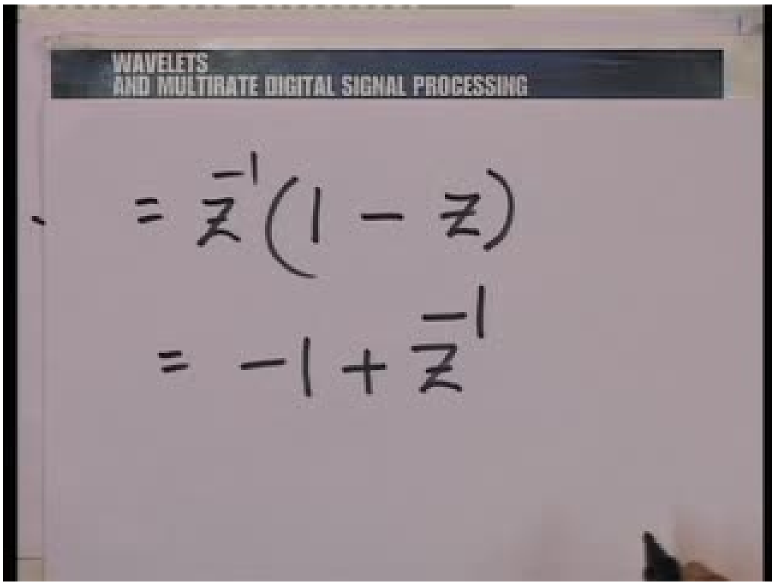
What we are going to do today, is to build a lattice structure for orthogonal filter banks. Now, towards that objective, what we shall first do, is to look at, as usual the Haar multi resolution analysis and we shall essentially bring out the lattice structure from the Haar analysis side, by invoking the requirement for computational efficiency.

(Refer Slide Time: 05:17)

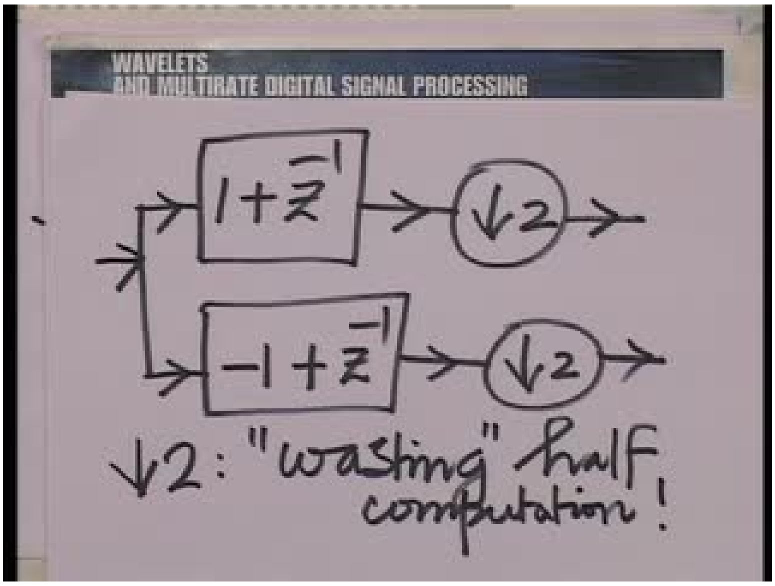


So, let us go straight down to putting down the Haar analysis filter bank. Now, on the analysis side, in the Haar, the low pass filter is essentially 1 plus Z inverse and we shall refer to this as H l Z or H low Z, let us say.

(Refer Slide Time: 06:19)



(Refer Slide Time: 06:35)



The corresponding high pass filter would be Z inverse H low minus Z inverse. And that is easily seen to be, Z inverse 1 minus Z, which is minus 1 plus Z inverse. And therefore, all in all, we have the following structure for the Haar analysis side.

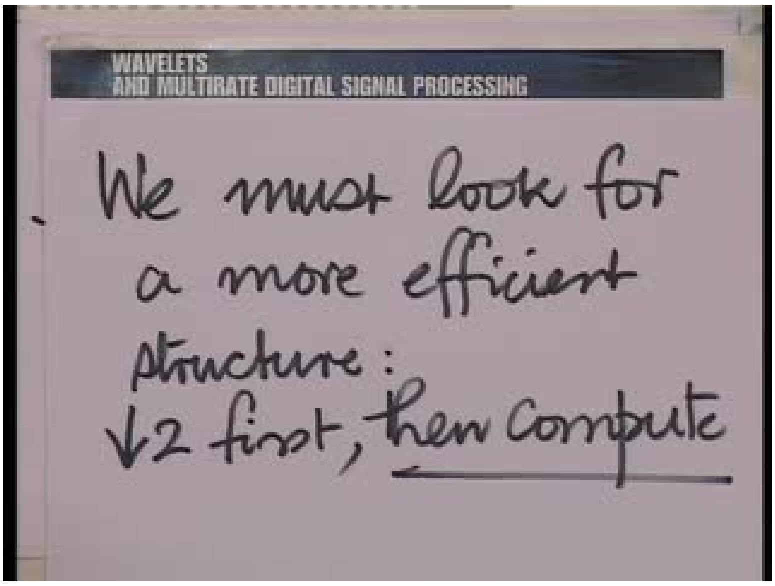
Now, you know there are different ways in which we can look at this structure. One of them is of course, from the point of view of the filter design problem. So, here, we have designed this low pass filter and designed the high pass filter and down sampled by 2, so much so, from the point of view of frequency domain interpretation. But when we look at what we are doing in this structure computationally, we would be aghast to see the wastage. Let us look at that structure from the point of view of computation.

Now, what are we doing in the structure here? We are taking the input, we are subjecting it to a convolution here, let us look at this branch, for example, we are subjecting it to a convolution here, all be it a small convolution, and then down sampling.

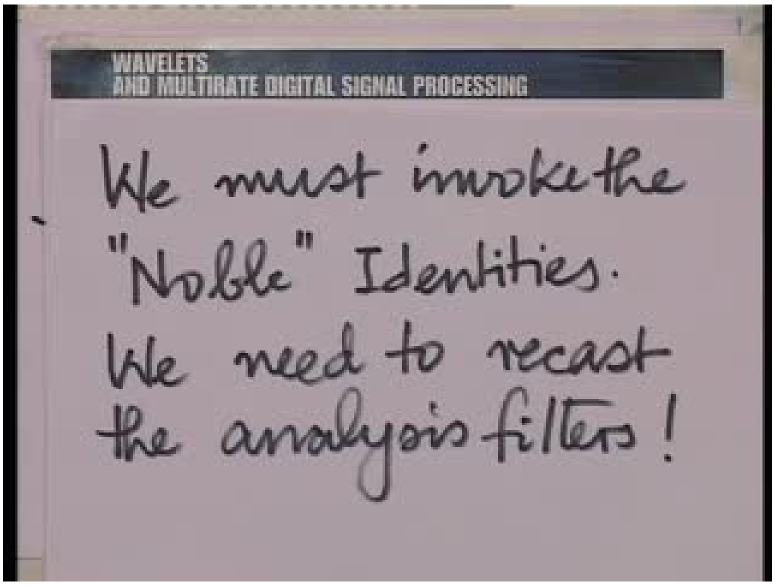
Now, you are calculating all samples, all outputs of the convolutions here, as also here, and when you down sample, you are losing half of those samples, half of that computation is entirely wasted. So, let us make a remark, down by 2 means wasting half the computation.

Every other sample is dropped here, but, you have computed it here. So, it is a terrible thing to do. It would have been so much easier, if we did this a little more efficiently, a little more strategically, by down sampling first and then doing the computation that is required. And that is not too difficult, if we only rearrange this part of the computation, in terms of a crisscross structure, as follows.

(Refer Slide Time: 09:19)



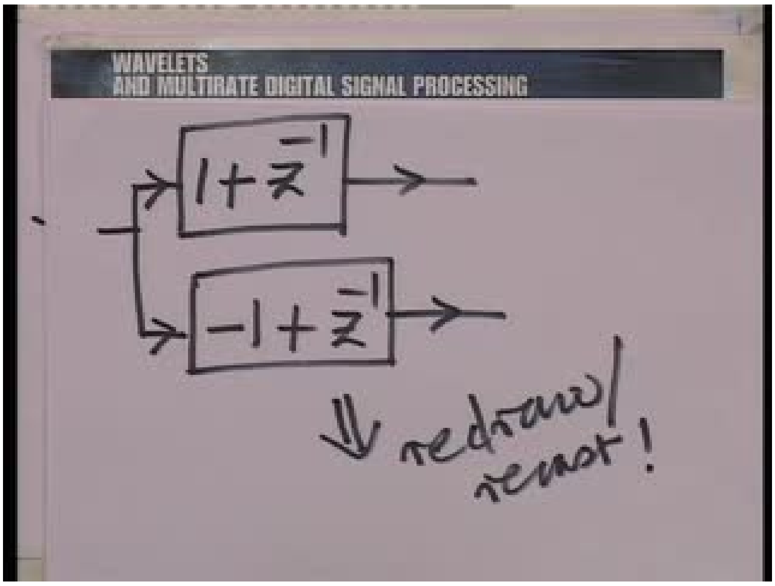
(Refer Slide Time: 09:59)



So, what we are trying to do is, we must look for a more efficient structure. And what should an efficient structure have? It must have down sample first and then compute. How do we do that? We have to invoke the noble identities. And to invoke the noble identities, we need to recast the analysis filters.

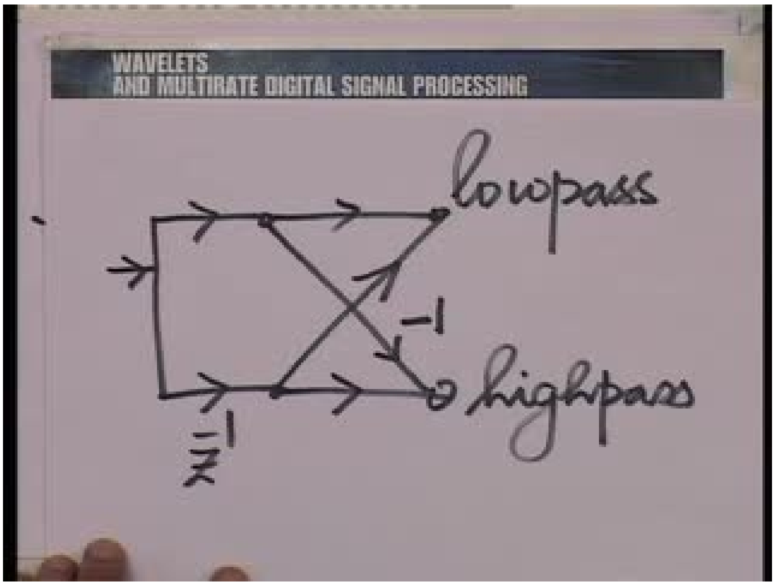
How do we recast the analysis filters? We must recast them in such a way, that we can interchange the down sampler and the up, the down sampler and the filtering operations.

(Refer Slide Time: 11:09)



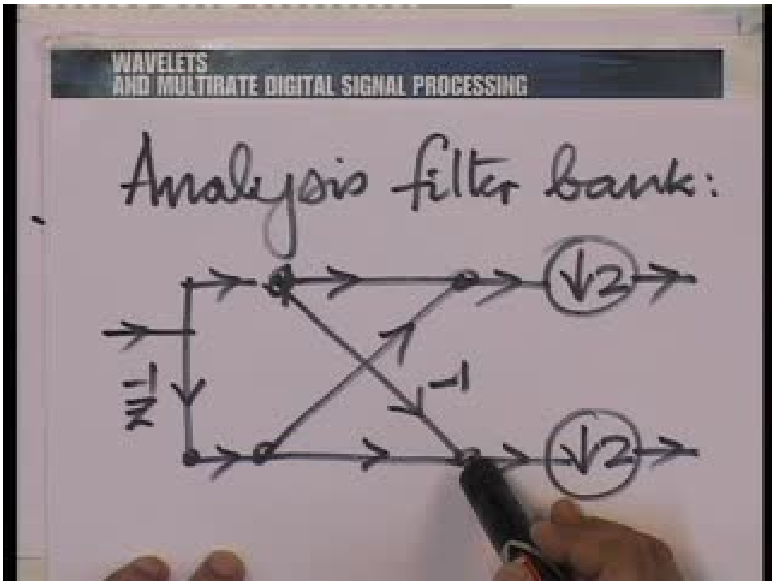
Now, in this case, let us rewrite the 2 analysis filters in such a way, that we have operations that are interchangeable, with down samplers. So, to do that, let us look at this structure carefully once again. Let us look at the pair of filters here. 1 plus Z inverse, minus 1 plus Z inverse. You know, the Z inverse operation is common. So, we can rewrite this, or redraw this, recast this, as follows.

(Refer Slide Time: 11:46)



We could first have 2 branches, one on which we pass the signal as it is and the other, where we pass the signal delayed by 1. Now, we add these 2, to get the upper filter. So, this gives us the low pass branch. And we subtract these 2, use a minus 1 here, to get the high pass operation.

(Refer Slide Time: 12:34)

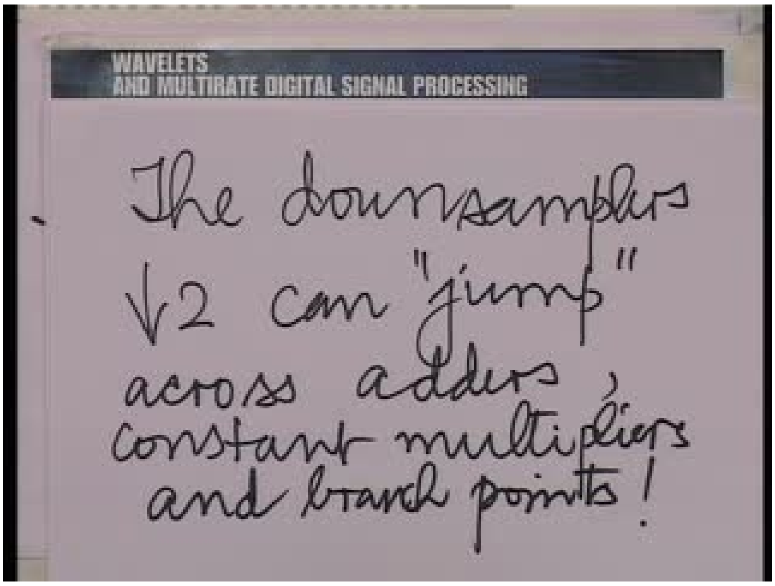


Now, once we have done this, we are in a position to put back the down sampler and the up and the down sampler for the moment. So, overall the analysis structure is like this. Let us put the Z inverse here. We have, followed by 2 down samplers and now, it is obvious, what we can do.

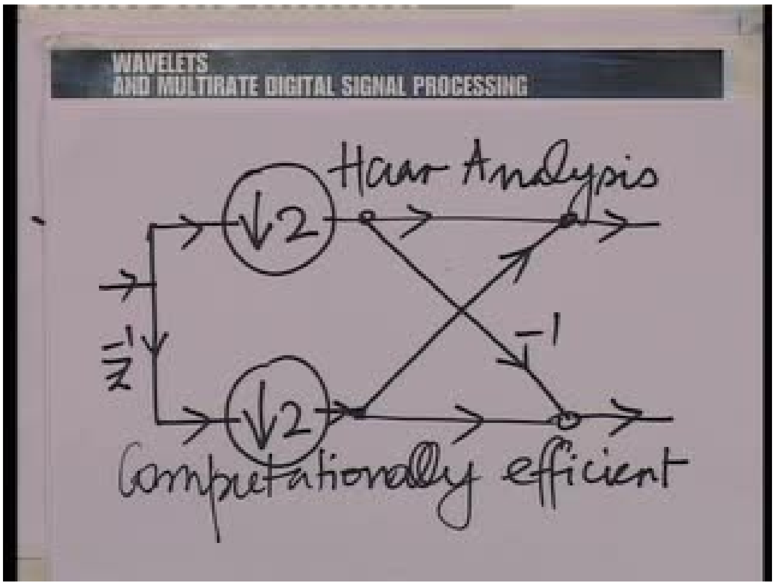
You see, you are down sampling after an addition here, and you are down sampling after and another one more addition here. These are of course, constant multipliers. All of these are in fact, these are trivial. There is no multiplication here. This is a trivial multiplier and therefore, it is very obvious that, this down sampler can jump across the adder here. So, you could bring this down sampler here, before the addition and in fact once you do that, then you can put both the down samplers all the way up to here. So, you know, you have a down sampler here, you have a down sampler there, so, you do not need to first branch and then down sample again.

You can first down sample and then branch. So, you can see what is happening. You have this down sampler, it jumps here. So, there is a down sampler here and a down sampler here. Similarly, this down sampler jumps on to these 2 branches. There is a down sampler here and 1 here. So, you have 1 down sampler on this branch and another sampler on this branch, but, then, you are branching and then down sampling, that is wasteful. You might as well first down sample and then branch. So, these 2 down samplers can come all the way up to here. This down sampler could come here and this down sampler could come here.

(Refer Slide Time: 14:55)



(Refer Slide Time: 15:35)



Let us draw the structure with that change. So, essentially the, the main theme here is, the down samplers can jump across adders, constant multipliers and branching. And doing that, we have a down sampler here, a down sampler there, an adder there, a minus 1 there and an adder there and this now becomes a computationally efficient structure for the Haar analysis side.

Why is it computationally efficient? Because you have down sampled first and then you are doing the calculations, additions, multiplications, corresponding to the filtering operations. So, there is no wasteful computation here, unlike the earlier structure, where you filtered first, that means you convolve first and then down sampled.

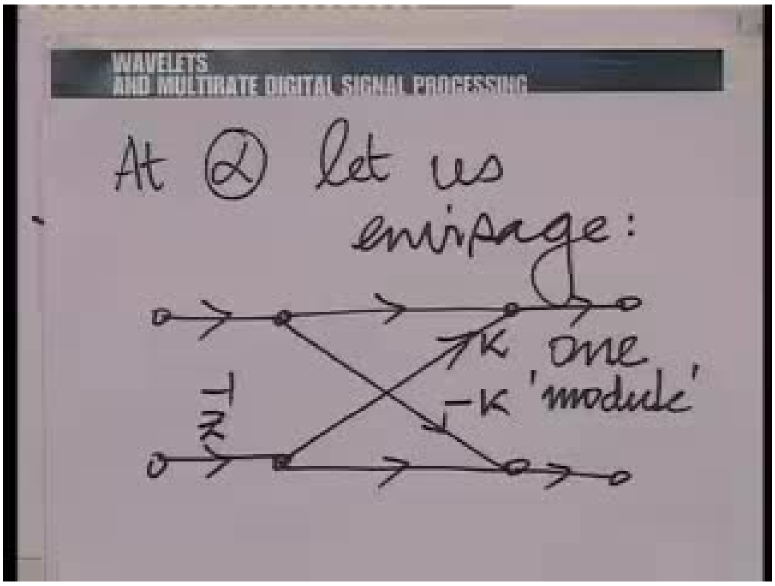
Now, the next question that we need to address, you see we have built the structure for the Haar. In fact, we would now like to extend this idea even further. The Haar is the baby of the Daubechies family, so to speak. It gives you analysis filters and therefore, synthesis filters of length 2.

Can we build upon the structure that we have here? To go to the Daubechies 4, Daubechies 6, and so on. In other words, can we take inspiration from the structure that we have just drawn here, to get a modular structure, which could give us the same computational efficiency and an increase in order, step by step? In other words, we wish to come out with the most parsimonious step, the smallest step, that we put one after the other, in repetition, so as to build the higher members of the Daubechies family for example, or for that matter, any family of conjugate quadrature filters.

Now, towards that objective, let us look at the structure that we have just drawn. So, in fact, if we look at this analysis computationally efficient structure that we have here, what we notice is essentially the following: one is, that this is rather very, very specific to Haar on account of the multiplier here and the multiplier here, being the same. Not forgetting that the multipliers here and here, are again a function of these 2 multipliers, because the filter that is created here is not independent of the filter here. In fact, the filter that is created here can be obtained in terms of the filter that was here.

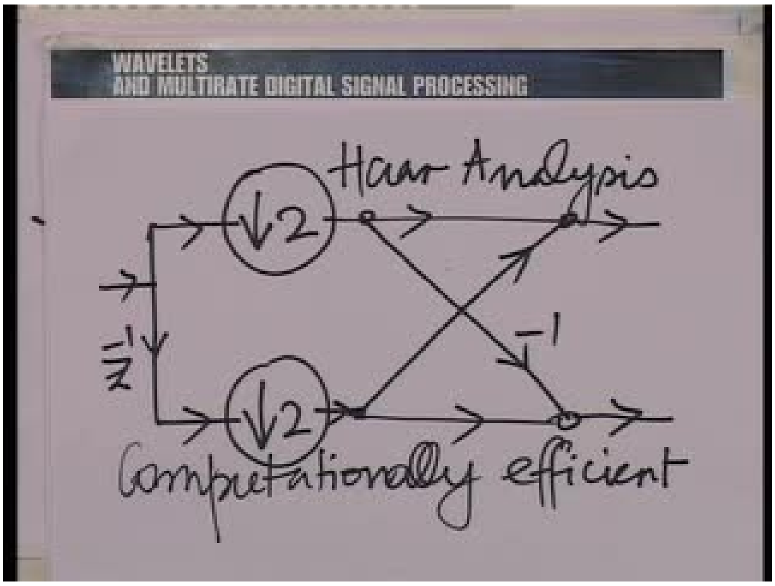
Now, what is it that we need, to free or to vary or to make more general, in this structure, to build one step of a module, or one modular pattern, or one step of the lattice, one stage of the lattice? Well, all that we need to do is, to bring in a relative variation between the coefficients here. So, leaving this coefficient, as it is 1, we could bring in a coefficient K here, and K can be varied to bring in a variety in different modules. And if you bring in K here, we must then bring in minus K there. That is the central idea behind building one stage of the lattice, so to speak.

(Refer Slide Time: 20:09)



Let me draw that structure before you. So, what we are saying is something like this. For the moment, close this. We are saying, for the addition and multiplication part, essentially use a structure like this, this plus K times this, minus K times this, plus this. But when we do this, unfortunately, we are restricted in terms of the order. No matter how many such stages we put, I mean, let me put them before you. Let me put back the previous structure before you.

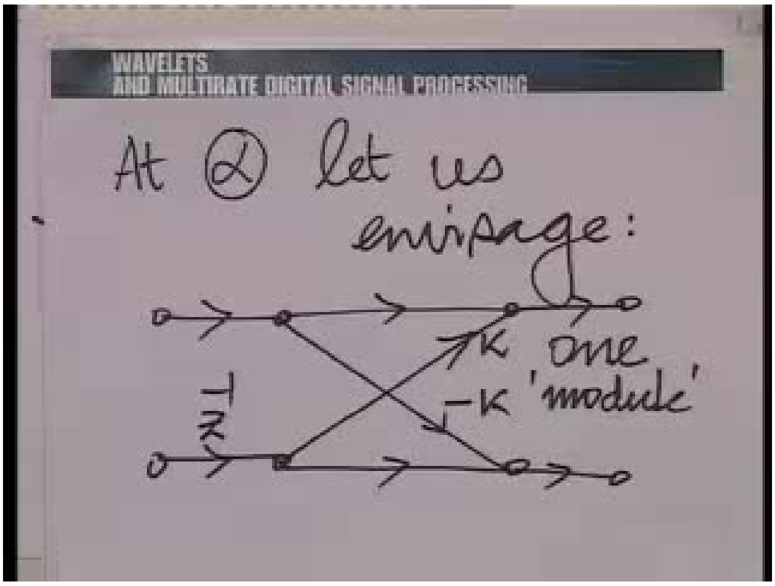
(Refer Slide Time: 15:35)



So, if I have this structure here and if I do nothing, except to put such stages, I mean, you know, in the module that I just showed you, if I were to forget this part, and just take this stage and put this stage repeatedly here. So, I put this stage repeatedly after this. All that will happen, means, all this will essentially be equivalent to some linear combination of this and this, one linear combination coming up here, another linear combination coming up here, there is no increase of order.

Now, how can I get an increase of order? An increase of order is going to be possible, if I bring in a delay element. So, I must put some delay element after the down sampler 2. Now, this is where I need to make an observation, about the noble identities, about which we talked earlier on in this lecture. If we put a delay, somewhere after the down sampler, recall from the noble identities, that it is equivalent to a delay of double the number of time units prior to the down sampler.

(Refer Slide Time: 20:09)



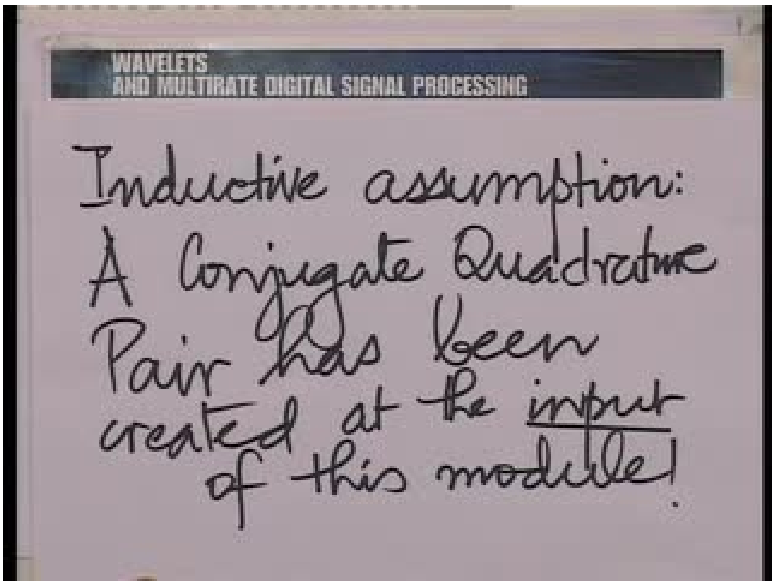
So, what I am saying is the following. If I add to this module, a delay prior to that, so as to increase the order. A delay here, which actually comes after the down samplers, is equivalent to a delay of 2 units before the down samplers.

So, let us put the whole structure down for discussion. Let us build an inductive argument to analyze the structure. Let me recall, before you, the principle of mathematical induction. Mathematical induction has 2 parts in a proffer in a construction. So, we are trying to make both an inductive construction and an inductive proof here. It has a basis step.

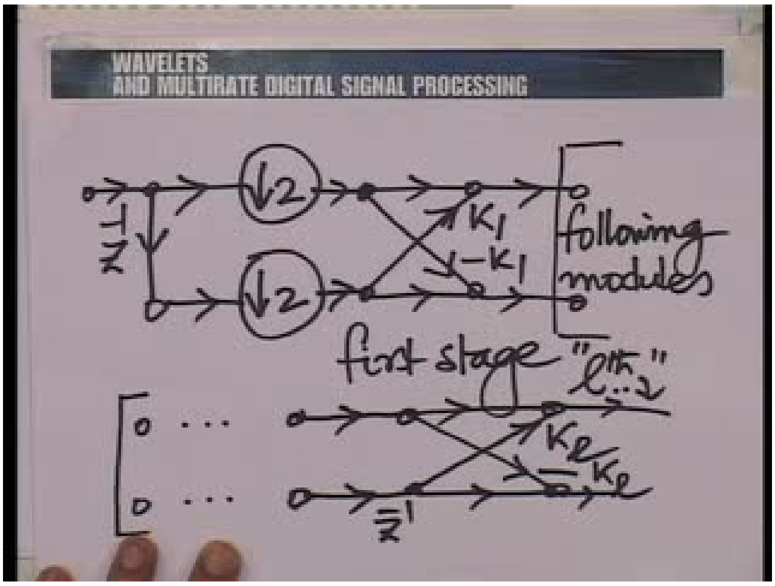
The basis step is one, where you build the lowest order structure or the lowest example of an entity hierarchy. In this case, it is a question of conjugate quadrature filters of a certain family

Now, you know the basis step here would essentially be a filter bank of length 2, with the filters of length 2. The Haar is an example. But in general, a filter bank of length 2 would essentially be just one stage of this, the Haar with the multiplier freed, that means the multiplier made variable. The second part of an inductive construction or inductive argument, is what is called inductive step. In the inductive step, there is an inductive assumption. What is the inductive assumption here? We need to put it down very clearly.

(Refer Slide Time: 24:12)

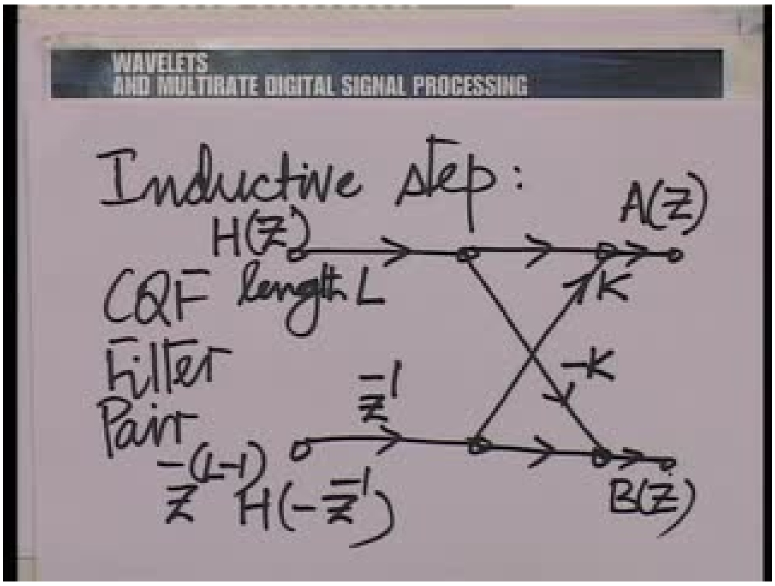


(Refer Slide Time: 24:36)



The inductive assumption here, would relate to having built a conjugate quadrature filter structure, of a certain order. So, we will say that prior to one such stage, a conjugate quadrature pair has been created, at the input of the module. Let me put that down graphically. So, what I am saying is, something like this. I am saying, I have this first basis stage here. So, I have down samplers, then I have essentially this stage, where this is, you know, notice this is very much like a Haar stage, except that, I have freed this multiplier here. I have put a K 1 and a minus K 1 there. In the Haar case, K 1 is equal to 1.

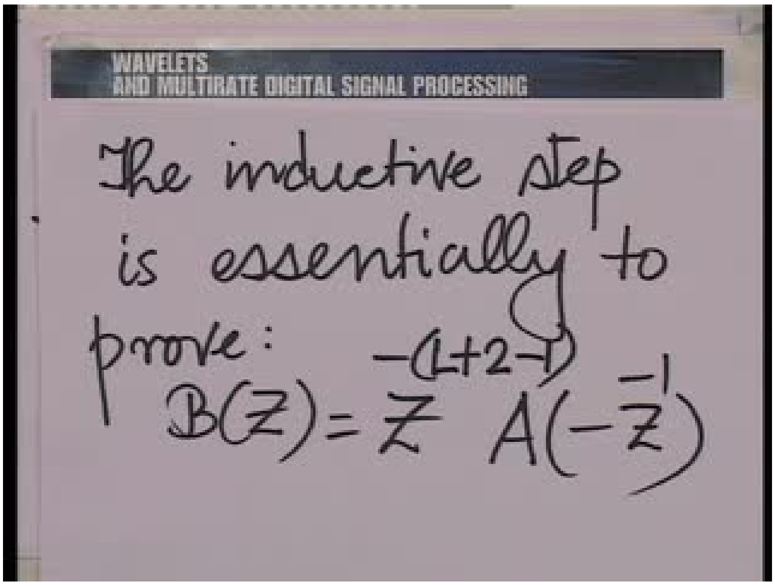
(Refer Slide Time: 25:55)



Now, after this stage, I have put several modules of the kind shown here. So, I have a Z inverse followed by a so called lattice constant stage. So, multiplier K l there, minus K l here, a summation here, and a summation here. Let me expand this stage. So, what I am going to show you, is the inductive assumption across this stage. The inductive step, therefore, can be put down as follows here.

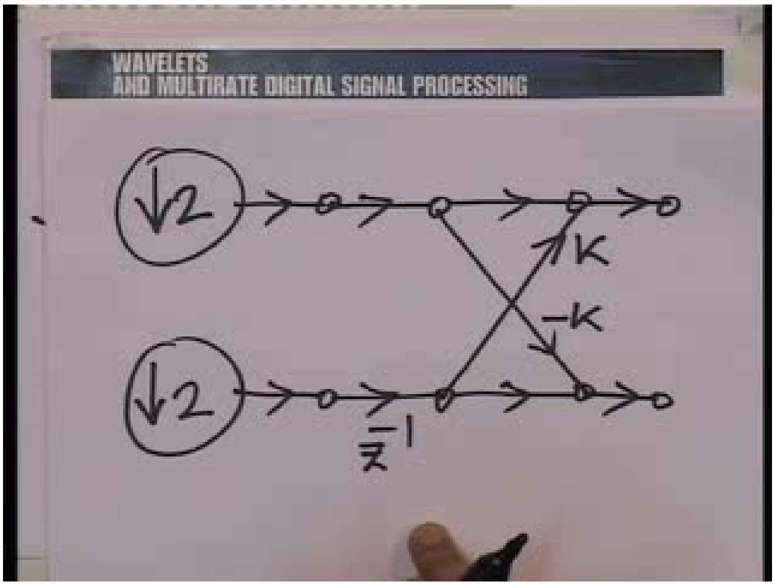
We have the conjugate filter pair. Let us call it H Z. So, up to here, let us say, we have got an H Z and here if H Z is of length L, then we have Z raised to the power minus L minus 1, H of minus Z inverse here and then we followed it with this one module that we have shown, a Z inverse here, a lattice addition there. So, I have a K, a minus K and summation here. And I am saying at this point, let the system function effectively be A Z and let the, let the system function at this point be B Z.

(Refer Slide Time: 27:34)



And we are now required to show, a relation between B Z and A Z, of the same kind as existed at the input here. So, therefore, the inductive step is essentially to prove B Z is Z raised to the power minus L plus 2, so, the length has increased by 2, minus 1 times A of minus Z inverse. This is what we need to show.

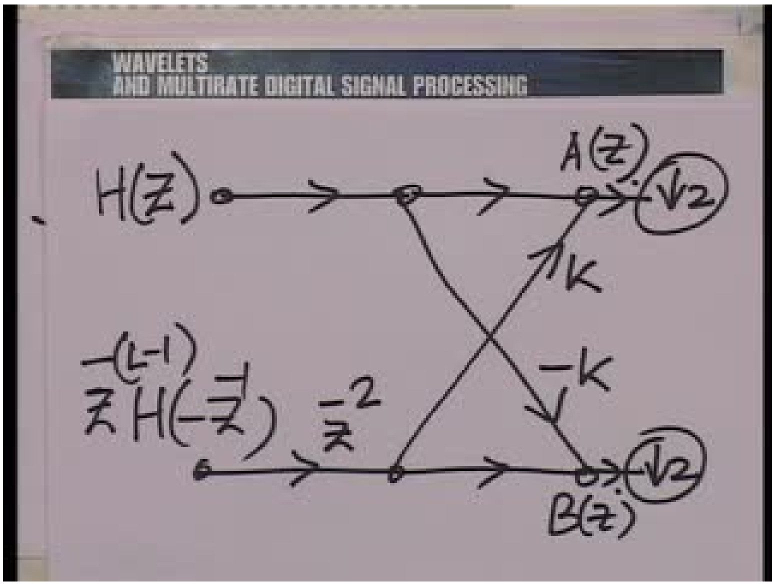
(Refer Slide Time: 28:47)



Now, you know, again, it is easier to use the noble identities at this point. So, you know, when we say that, we have built the inductive step up to here, we made this inductive assumption, we made it assuming that, the down sampler has come all the way up to the input point here. So, we are saying, we have the down sampler here and then we followed it with this stage.

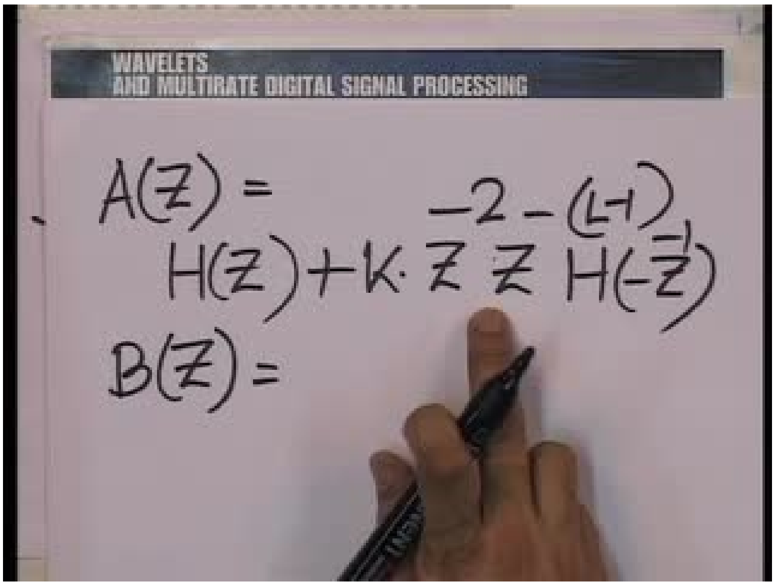
And the proof that we need to complete, is when we bring the down sampler from here to here, and that can be done using the noble identities. So, if you use the noble identities, if you take this down sampler and make it jump across this delay here, this delay is replaced by Z raised to the power minus 2 and then of course, you can keep jumping the down sampler across these branches. So, down sampler can jump across the branch point. It can jump across the addition, it can jump across the multiplication and all in all we have H Z here,

(Refer Slide Time: 30:05)

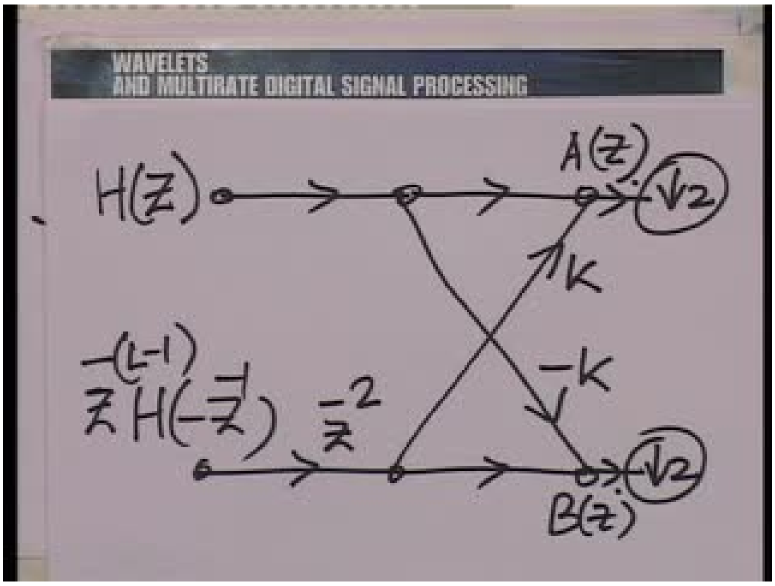


Z raised to the power minus L minus 1, H minus Z inverse, there, a Z raised to the power minus 2 here, a K there, a minus K there and the down samplers back again here. And we need to analyze the A Z created here and the B Z created here, which is now very easy to do, because I can write A Z in terms of these 2 and I can write B Z in terms of these 2. Let me do that.

(Refer Slide Time: 31:07)

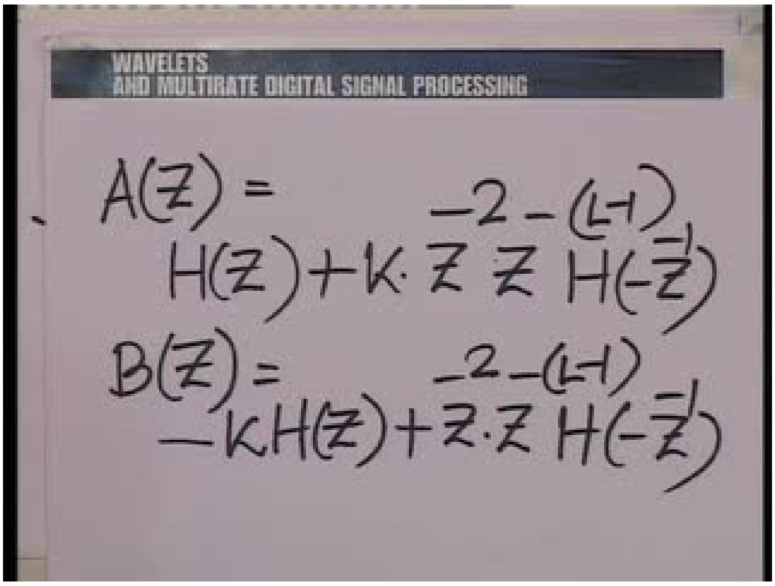


(Refer Slide Time: 30:05)



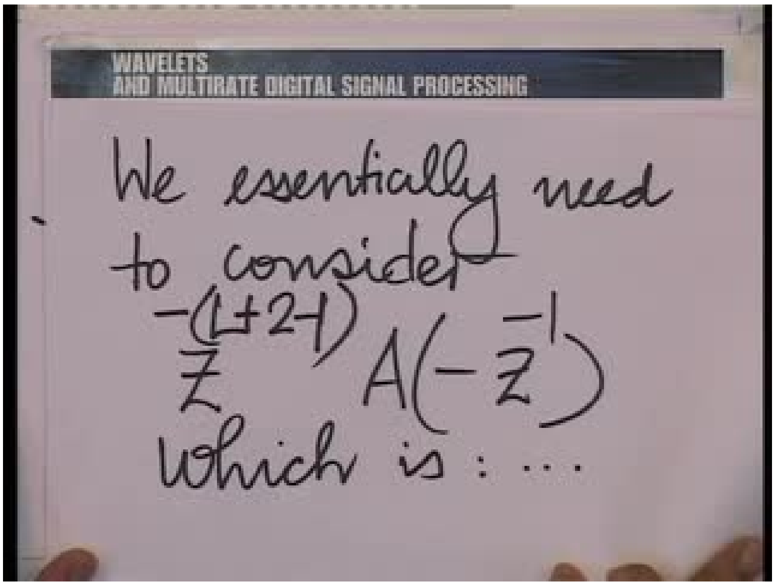
So, I have, A Z is very clearly, H Z plus K times Z raised to the minus 2 times Z raised to the power minus L minus 1 H minus Z inverse. And B Z is very clearly, in fact, let me just put back the figure before you to explain this. I am saying A Z is H Z plus K times Z to the power minus 2 into Z to the power minus L minus 1, H of minus Z inverse. So, H of this multiplied by Z raised to the power minus 2 multiplied by K and added here.

(Refer Slide Time: 32:14)

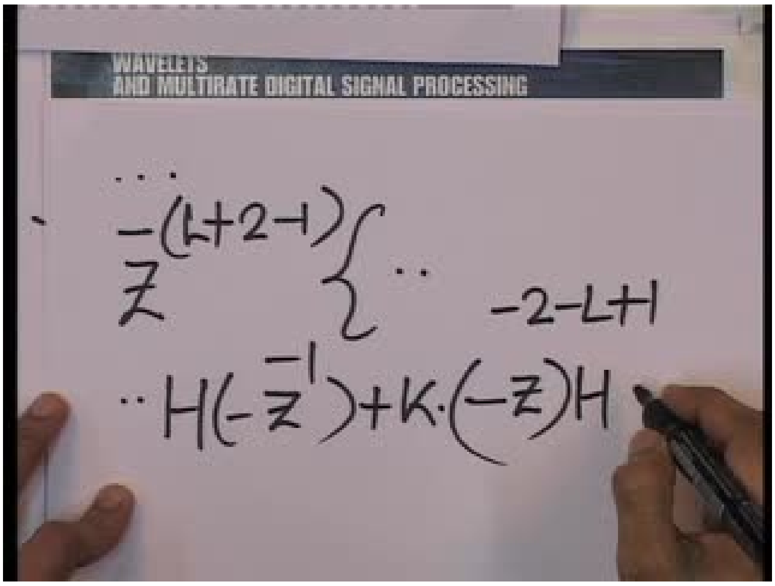


Similarly, here it is minus K times H Z plus Z raised to the power of minus 2 times this, added together to give B Z. So, B Z is minus K times H Z plus Z raised to the power minus 2 times Z raised to the power minus L minus 1 H minus Z inverse. Now, all that we need to do, is to consider the corresponding high pass filter, assuming that A Z is low pass.

(Refer Slide Time: 32:50)

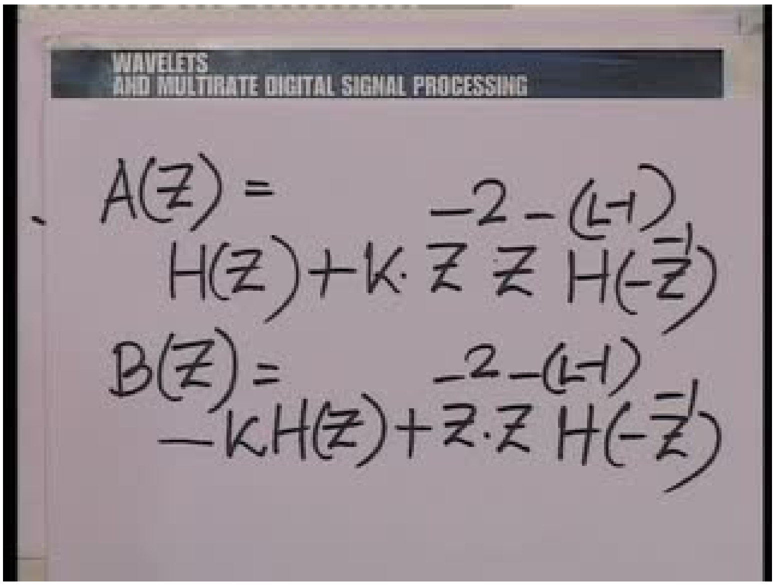


(Refer Slide Time: 33:52)

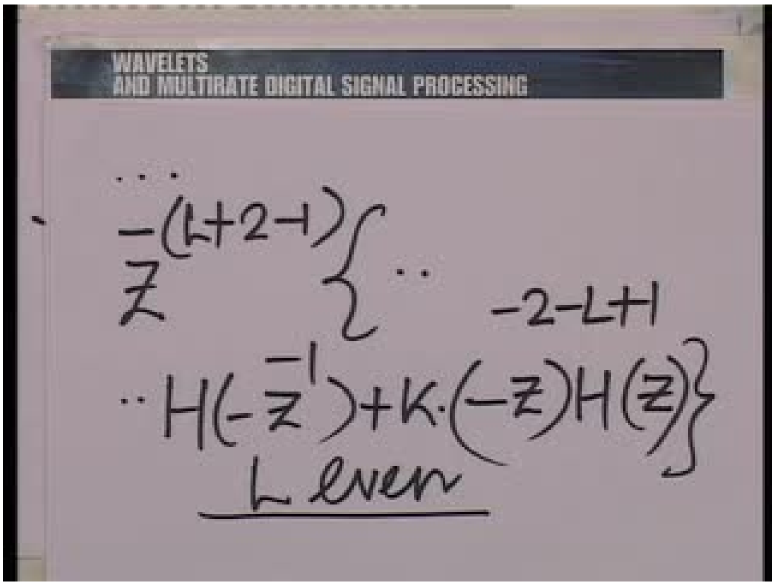


So, we essentially need to consider A Z replaced by A minus Z inverse first, and then noting that this would become non-causal, delay it, now note, delay it, but, assuming that the length has gone up by 2, so, Z raised to the power minus L plus 2 minus 1. Let us do that. So, let us use the expression for A Z here. And let us make the transformation Z replaced by minus Z inverse and then this, multiplied by Z raised to the power minus L plus 2 minus 1, as we have here, which is, well, Z raised to the power minus L plus 2 minus 1 times H minus Z inverse, now, plus K into minus Z to the power minus 2 minus L plus 1 times H.

(Refer Slide Time: 32:14)



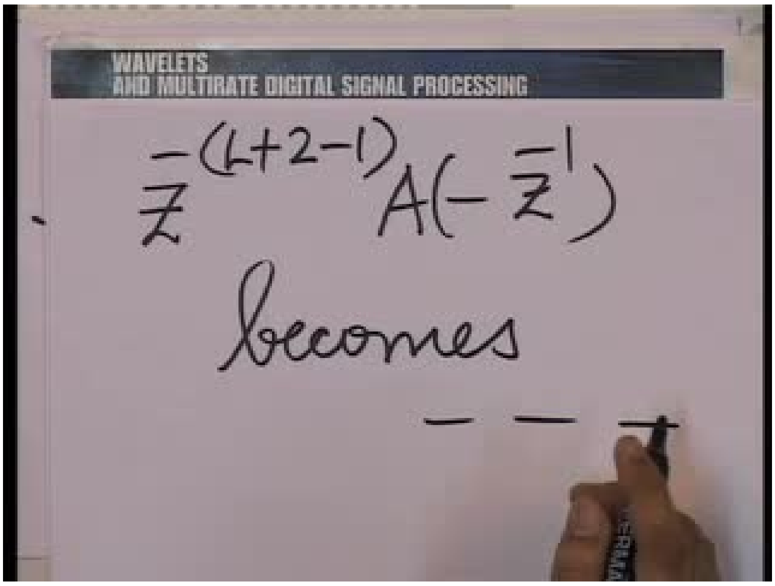
(Refer Slide Time: 34:44)



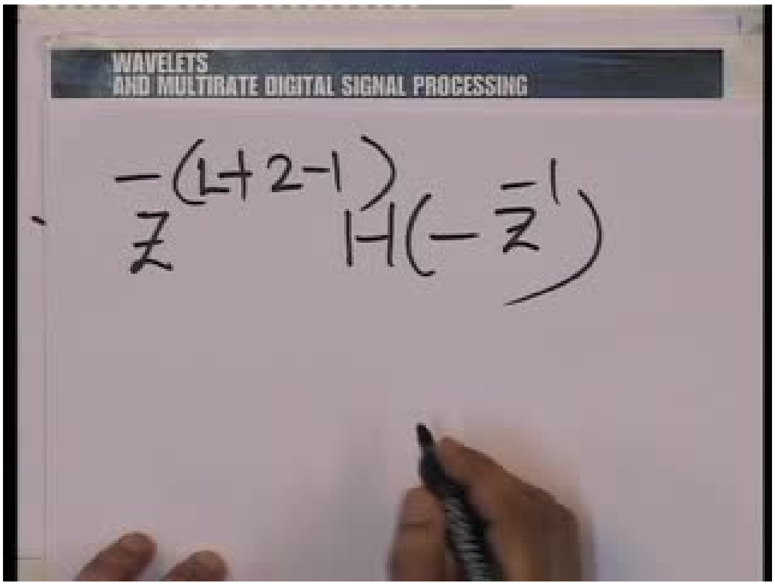
Now, here we have the argument, minus Z inverse ((sign)) and when we replace Z by minus Z inverse once again, this whole thing simply becomes Z here. So, it is H Z. And we close them.

Now, this is easy to expand. You see, you have minus Z raised to the power minus 2 minus L plus 1. And please remember, L has been assumed to be even. Let us make a note of that. L is even by inductive assumption. You know, the first filter of length 2 is L equal to 2 and each time you add, you are assuming the step is 2, you are increasing the filter length by 2, so, L is even.

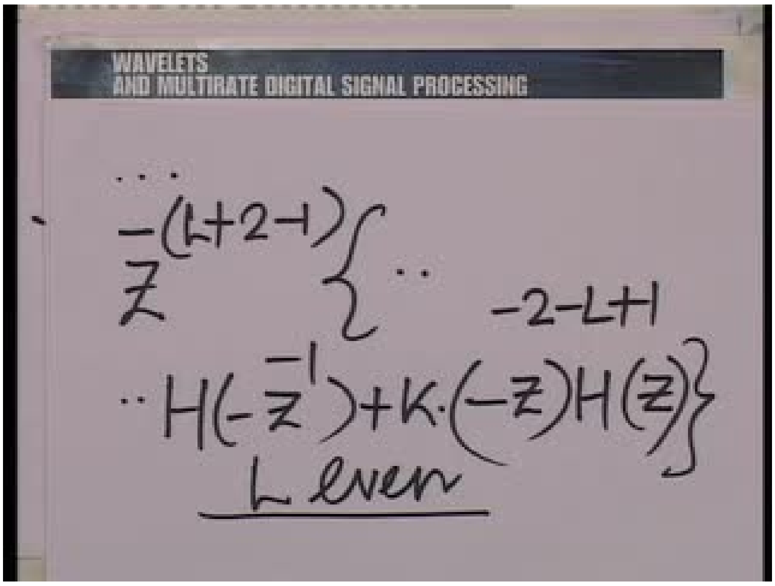
(Refer Slide Time: 35:48)



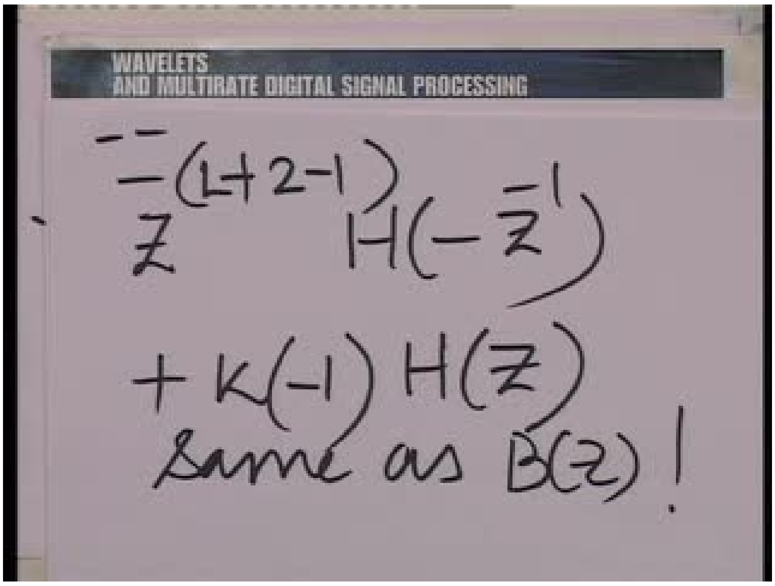
(Refer Slide Time: 36:03)



(Refer Slide Time: 34:44)

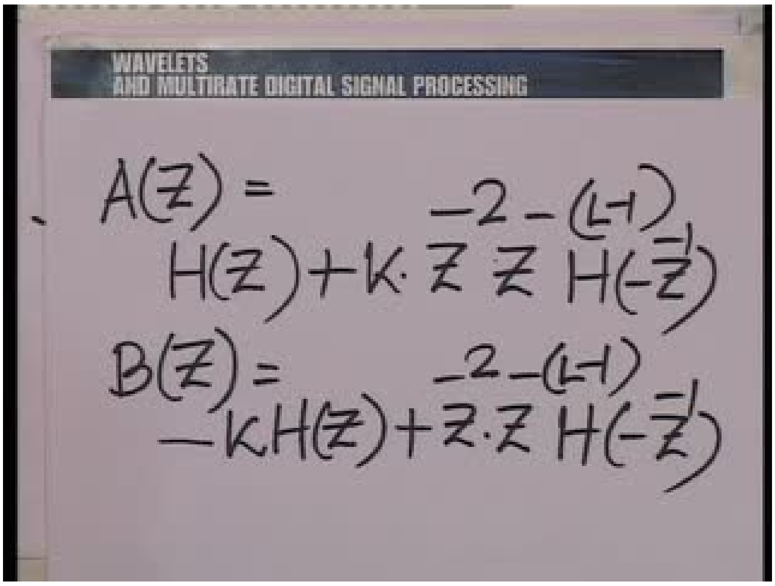


(Refer Slide Time: 36:36)



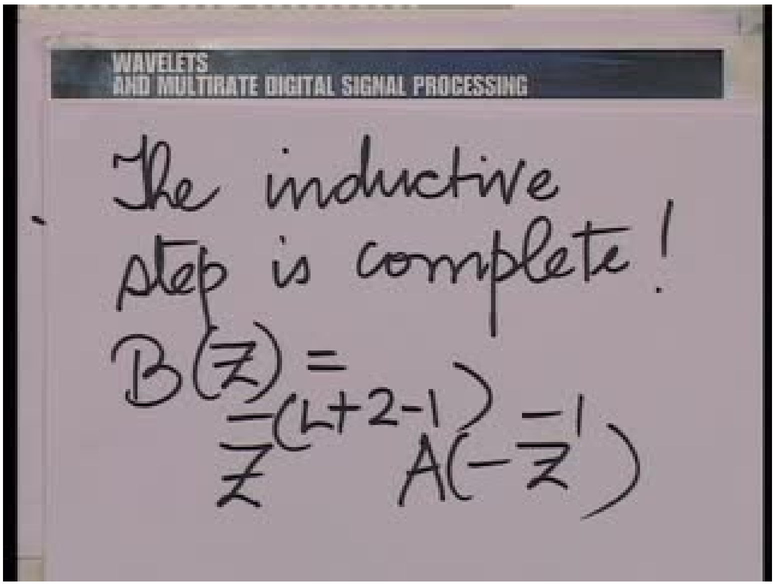
So, in other words, Z raised to the power minus L plus 2 minus 1 times A of minus Z inverse becomes Z raised to the power minus L plus 2 minus 1 times H of minus Z inverse. If we simply simplify this, plus K into, now, here you have a minus 1 raised to the power of minus 2 minus L plus 1. This must be odd. L is even. So, this must be odd. So, minus 1 raised to the odd integer, leaves you with a minus 1 here. And then, of course, you have a Z raised to the power minus L plus 2 minus 1 here and a Z raised to the power minus, you know, you could take this into brackets. So, this becomes L plus 2 minus 1 in brackets, with a minus sign. So, Z here and a Z inverse there, these cancel and that leaves you with minus 1 here and H Z there.

(Refer Slide Time: 32:14)



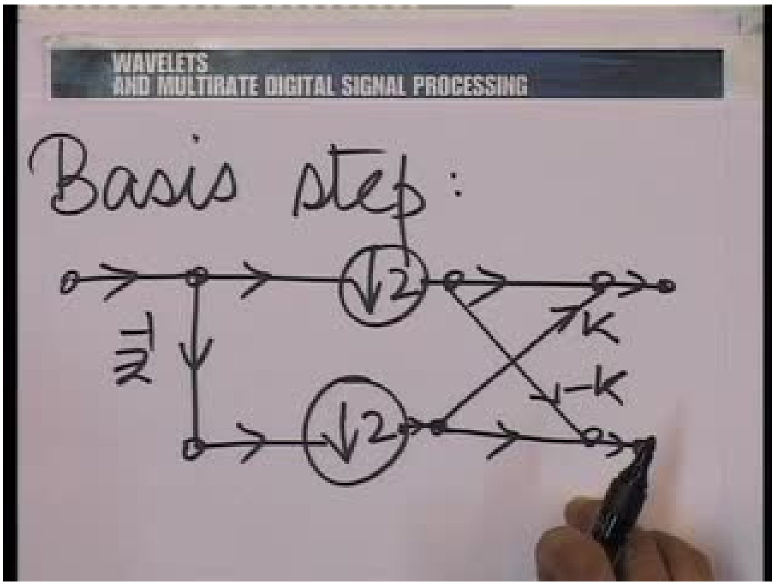
Continued from the previous. So, it is very clear that, what we have here is, essentially the same as B Z. Indeed, let me put back the expression for B Z before you, for comparison. B Z was minus K times H Z plus Z raised to the power minus 2 Z raised to the power minus L minus 1 H minus Z inverse and look at what we have here, minus K times H Z plus Z raised to the power minus L plus 2 plus minus 1 H of minus Z inverse. Perfect match, this and this. So, in fact, we have completed the inductive step.

(Refer Slide Time: 38:08)

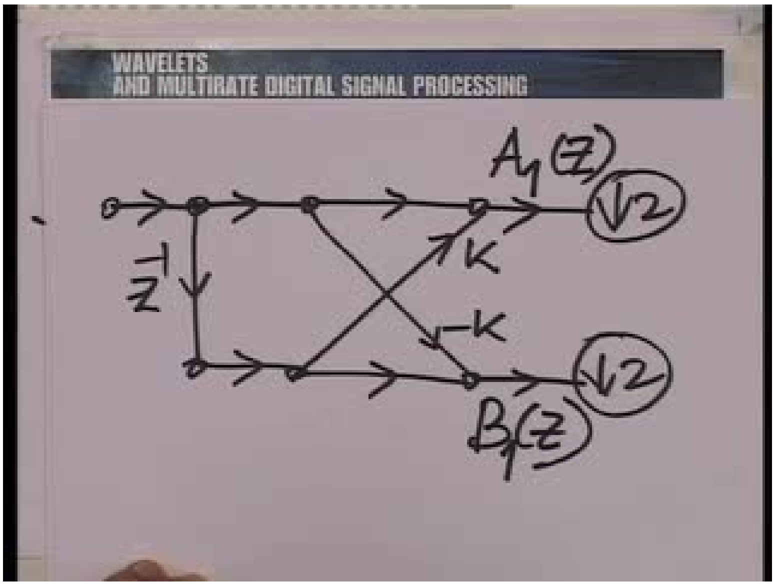


The inductive step is complete. B Z is Z raised to the power minus L plus 2 minus 1 A minus Z inverse. So, what we are saying is, the effect is that, the conjugate quadrature relation that we require on the analysis side is replicated one module later. So, each time we put a module like this, we at least replicate the relationship between the low and high pass filter in a conjugate quadrature structure analysis side. Now, for the basis step. That is easy.

(Refer Slide Time: 39:02)



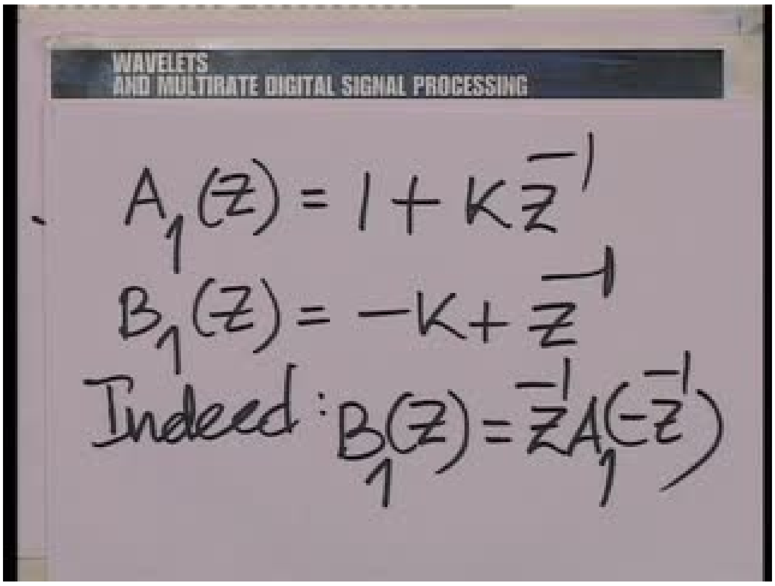
(Refer Slide Time: 39:55)



Well, the basis steps needs to begin from here, down sample there and then the lattice here, K times there, minus K times here, a summation here and we need to jump these down samplers here and study the relationship between these. So, we need to look at the following structures.

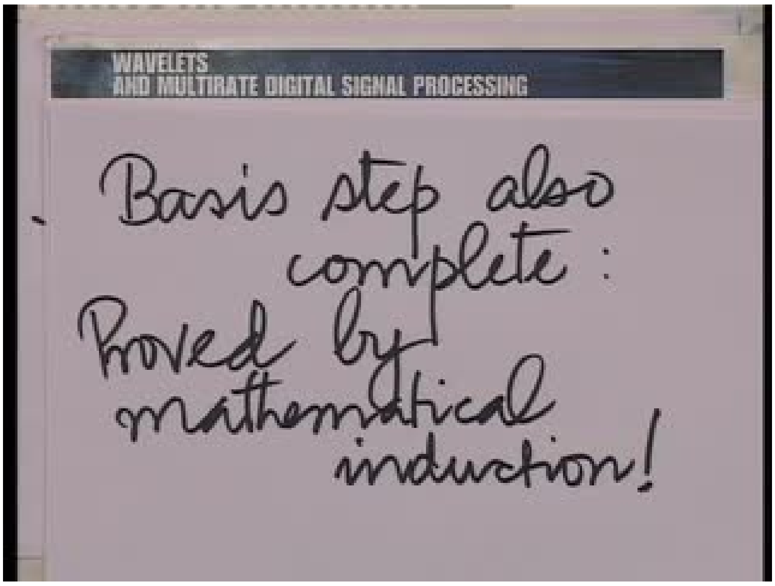
A K there, a minus K here, and down samplers following. Let us put down the system functions. Let us call them A 1 Z and B 1 Z here. The system function, up to this point, is essentially 1 plus Z inverse times K. Simple enough, is it not? A 1 Z is 1 plus Z inverse times K, looking at all the forward paths. Notice that, there are no loops here.

(Refer Slide Time: 41:14)



There is no question of a denominator. B 1 Z is similarly, minus K plus Z inverse. Let us write these 2 down. And indeed, it is very obvious that, B 1 Z is Z inverse A 1 minus Z inverse. Simple enough. It does not require too much of computation to show that, if you replace Z by minus Z inverse here, you get a minus K and then if you, you know, essentially multiplied by Z inverse here, you get what you require.

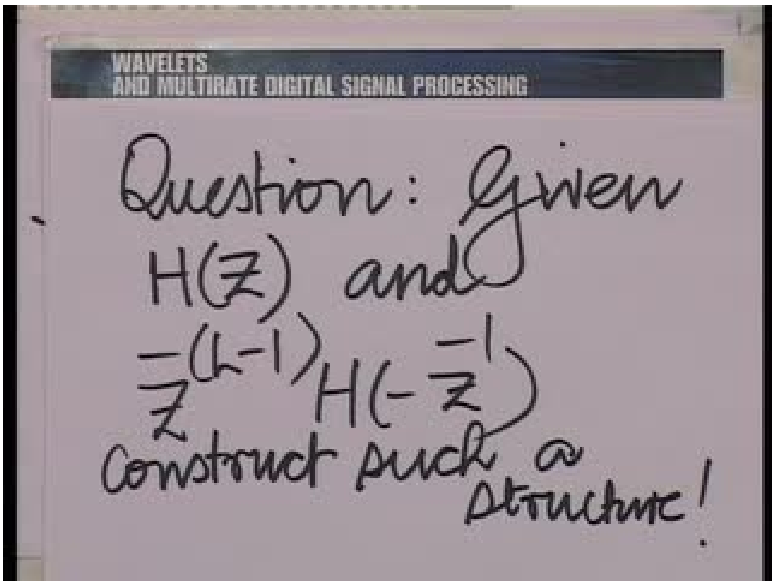
(Refer Slide Time: 42:04)



So, the basis step is complete as well. Therefore, proved by mathematical induction. What exactly have we proved? Let us spend a minute in reflecting on that. What we have proved is that, this repetitive structure that we have here, generates a pair of conjugate quadrature filters on the analysis side.

Now, the proof is analytic. So, assuming a modular structure, it shows you that, the conjugate quadrature relationship is maintained. It is not synthetic, meaning, if I already have a pair of conjugate quadrature filters, can I construct such a modular structure to realize that pair of conjugate quadrature filters? This question needs to be answered now. The synthesis or the construction.

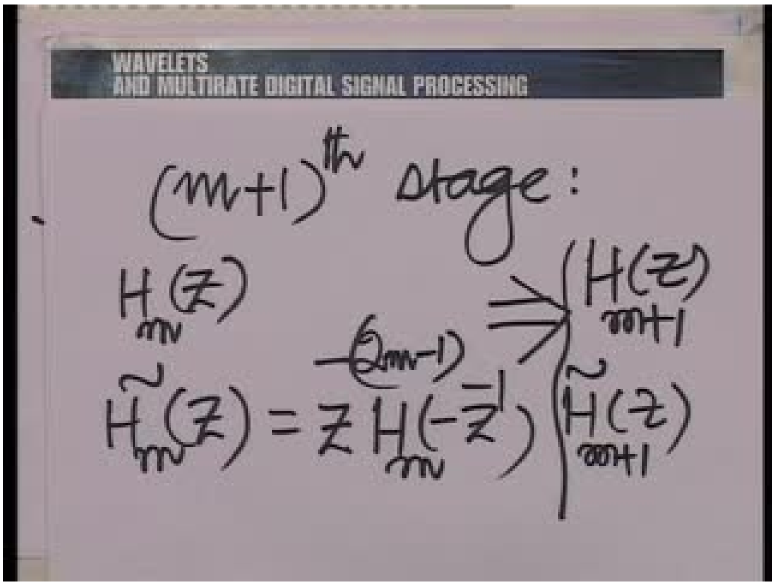
(Refer Slide Time: 43:18)



So, the question that needs to be answered is, given H Z and Z raised to the power minus L minus 1 H of minus Z inverse, construct such a structure. Now, towards that objective, we would need essentially, to go down 1 step.

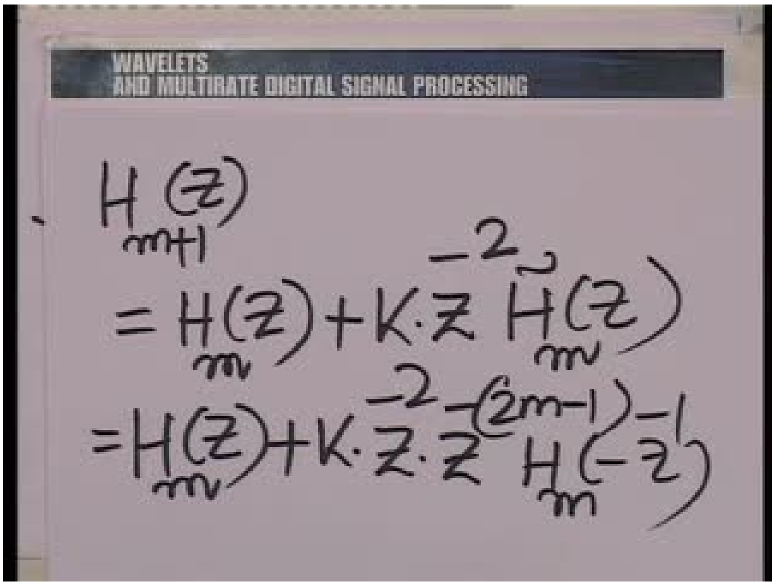
So, assuming that, we have filter lengths of more than 2, we need to show, how we can peel off, you know, you, you could think of this lattice structure as 1 stage surrounding the next, to make the filter lengths larger and larger. If you need to show a mechanism by which you can peel off the outer stage to go 1 stage inwards, jump backwards across 1 module. Now, to do that, we need to write down the 2 relationships once again.

(Refer Slide Time: 44:44)



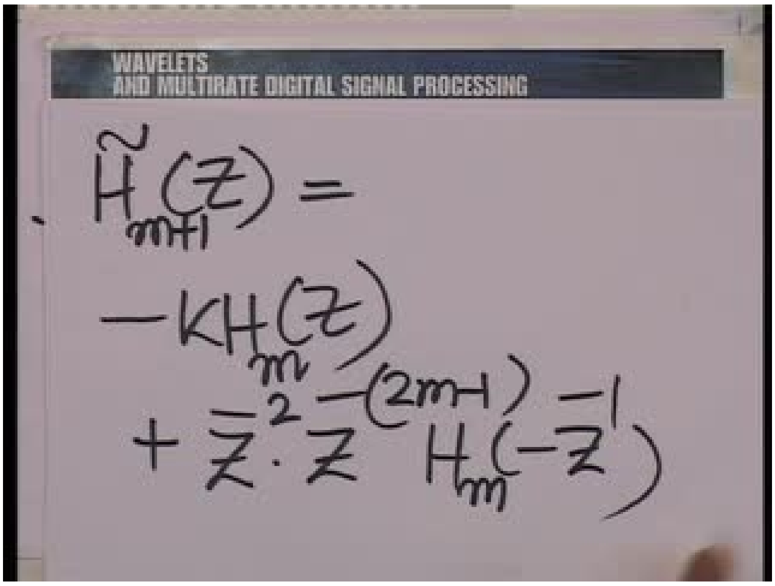
So, we have, let us consider the m plus 1 th stage, so to speak, where you have H m Z coming in here and H m tilde. Now, we will introduce this notation H m tilde Z, H m tilde Z being Z raised to the power minus 2 m. So, you know at the mth stage, you have a length of 2 m. So, 2 m minus 1, H m minus Z inverse here and the m plus 1th stage takes you from these to H m plus 1 Z and H m plus 1 tilde Z. And of course, in the m plus 1 stage, as you know, you have a Z to the power minus 2 there and then a multiplier and adder stage. So, what we have in general, in the m plus 1 stage, is the following relationship.

(Refer Slide Time: 46:04)

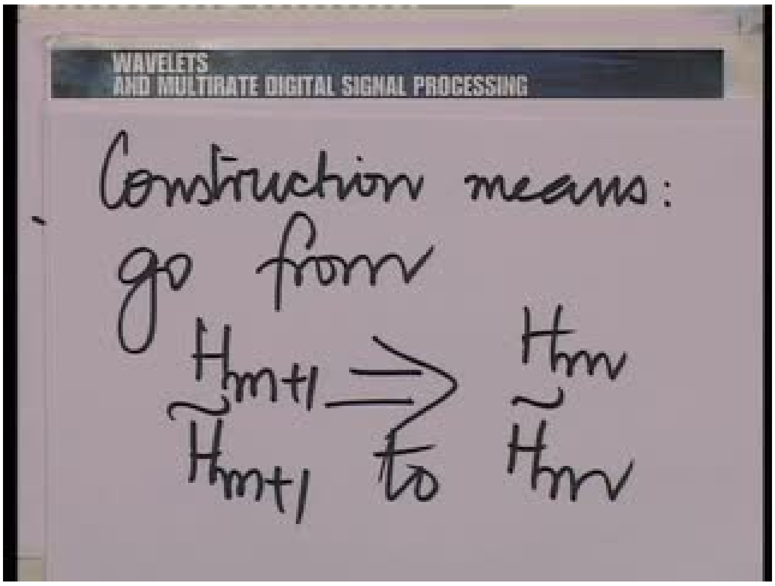


So, we have H m plus 1 Z is H m Z plus K times Z raised to the power minus 2 H m tilde Z, which of course, also becomes H m Z plus K times Z raised to the power minus 2 Z raised to the power minus 2 m minus 1 H m minus Z inverse. And we have assumed that, this structure, well, we have proved by a mathematical induction, after an inductive assumption that, a modular structure like this continues the conjugate quadracy.

(Refer Slide Time: 47:10)

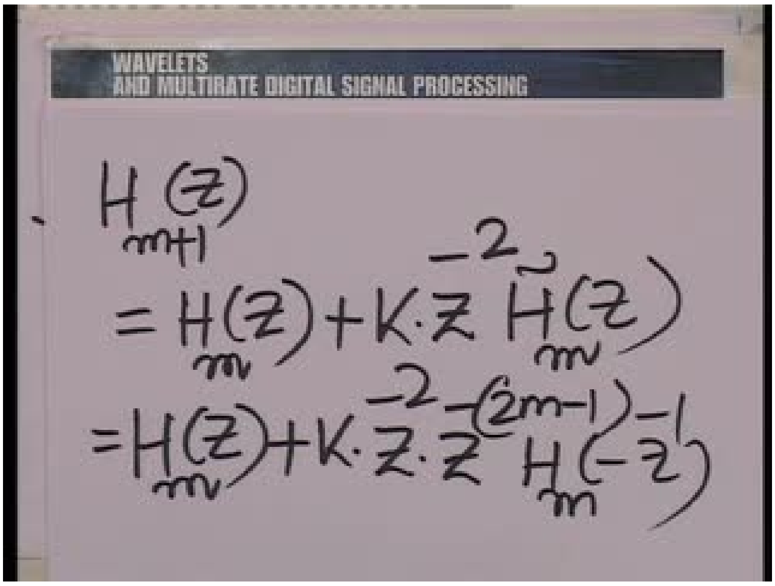


(Refer Slide Time: 47:42)



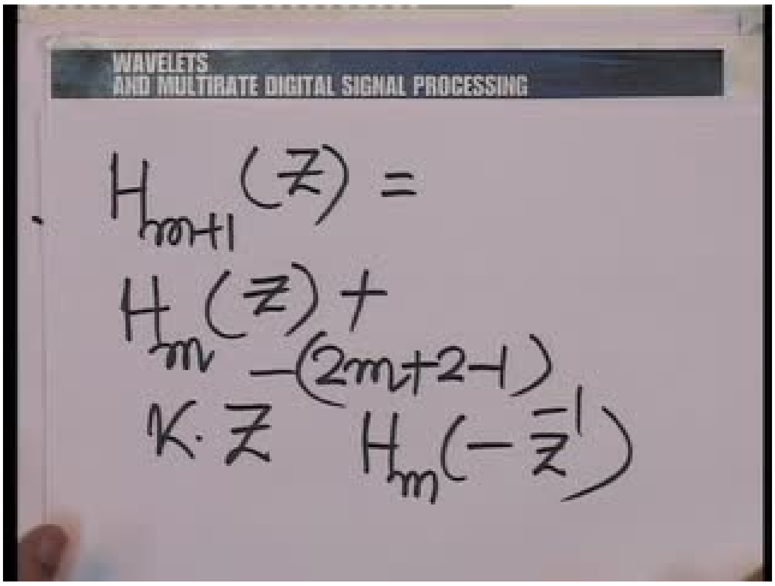
So, H m plus 1 tilde Z, which we assume is available to us, is equal to minus K times H m Z plus Z raised to the power minus 2 Z raised to the power minus 2 m minus 1 H m minus Z inverse. Now, our objective in the synthesis or the construction, construction means, go from H m plus 1, H m plus 1 tilde to H m and H m tilde.

(Refer Slide Time: 46:04)

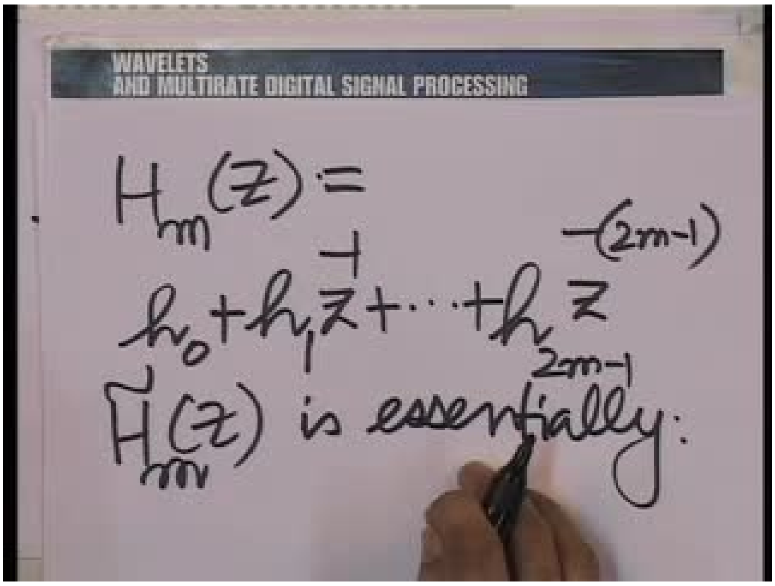


And that essentially means, extract the K, the K m plus 1. You know, if you know the K m plus 1, you also know what that stage is. There is nothing, there is nothing new in each stage, except the K m plus 1. So, we need to put down the value of K explicitly. So, for that purpose, we will need to go back to these 2 equations once again. Let us write these 2 equations down explicitly. H m plus 1 is H m plus K times all this. In fact, if you look at this expression carefully, we have the answer to what K is. Let us focus our attention only on this expression here.

(Refer Slide Time: 49:01)

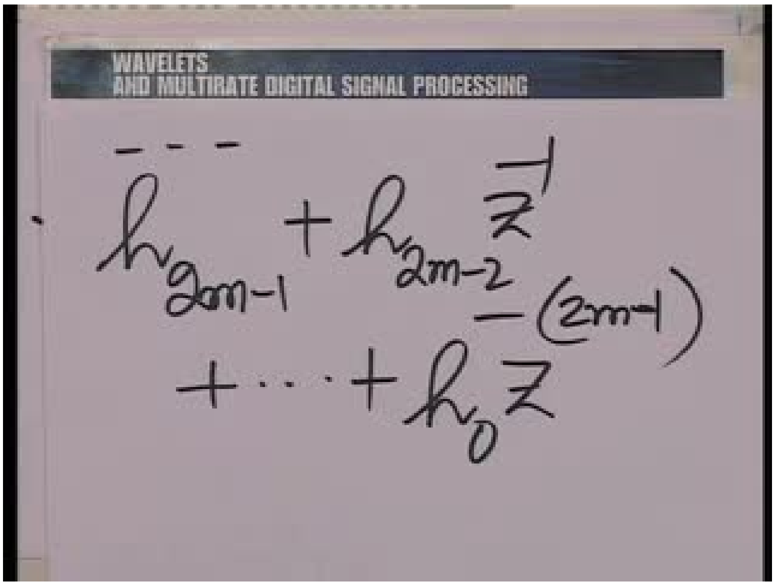


(Refer Slide Time: 49:32)



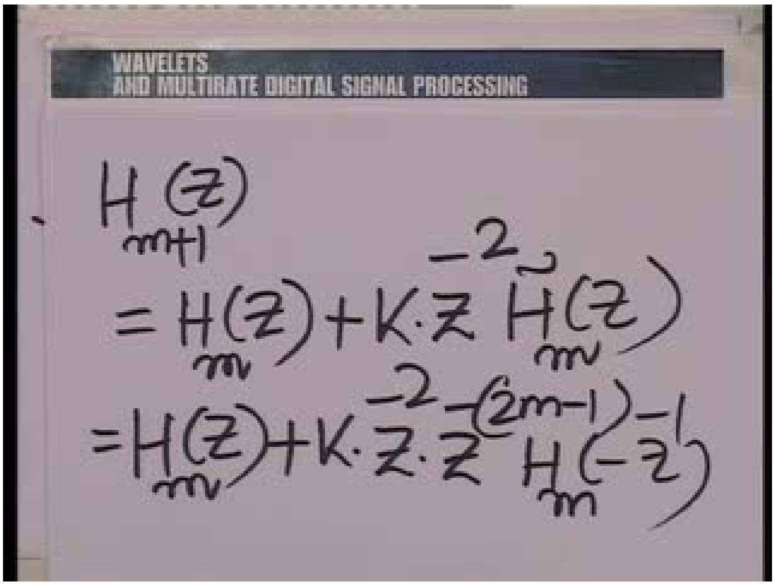
Now, you know, we know the length of H m Z. H m Z is of length 2 m. So, it is like this. It has, you know, if you look at H m Z, it is going to be of the form h 0 plus h 1 Z inverse plus, plus h 2 m minus 1 Z raised to the power minus 2 m minus 1.

(Refer Slide Time: 50:53)

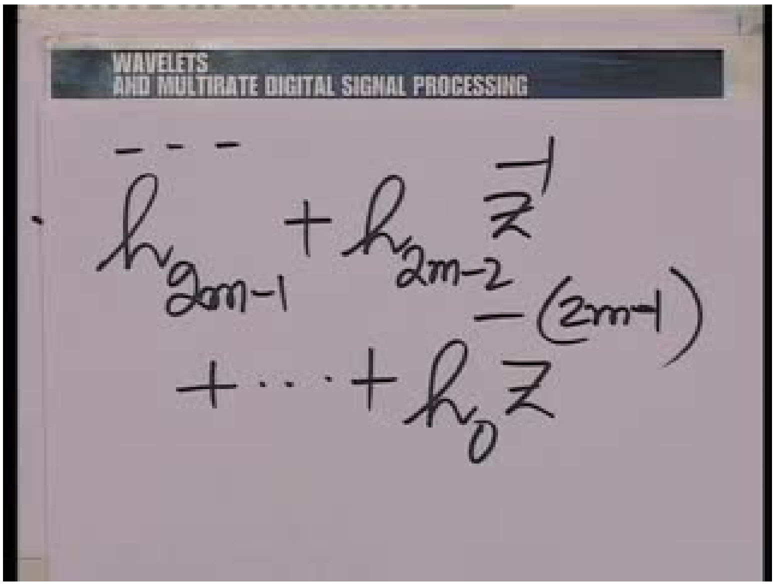


So, let us simply put down these coefficients and if we look at H m minus Z inverse, what we are going to have here, is essentially Z replaced by Z inverse everywhere. So, you have h 0 plus h 1 Z plus and so on h 2, 2 m minus 1 Z to the power 2 m minus 1, not minus 2 m minus 1 here. And then, when you multiply it by Z raised to the power minus 2 m minus 1, when you take this part, it essentially amounts to a reversal of these coefficients here. So, H m tilde Z, is essentially h 2 m minus 1 plus h 2 m minus 2 Z inverse plus and plus h 0 Z raised to the power minus 2 m minus 1.

(Refer Slide Time: 46:04)

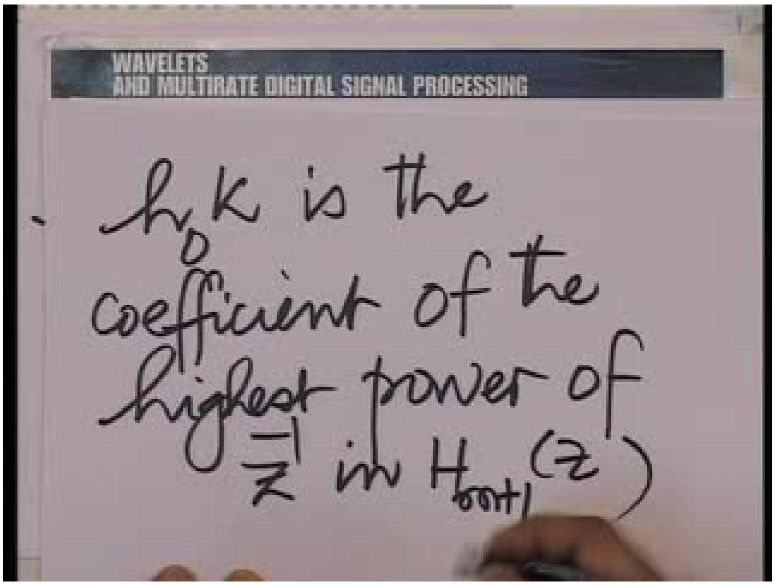


(Refer Slide Time: 50:53)

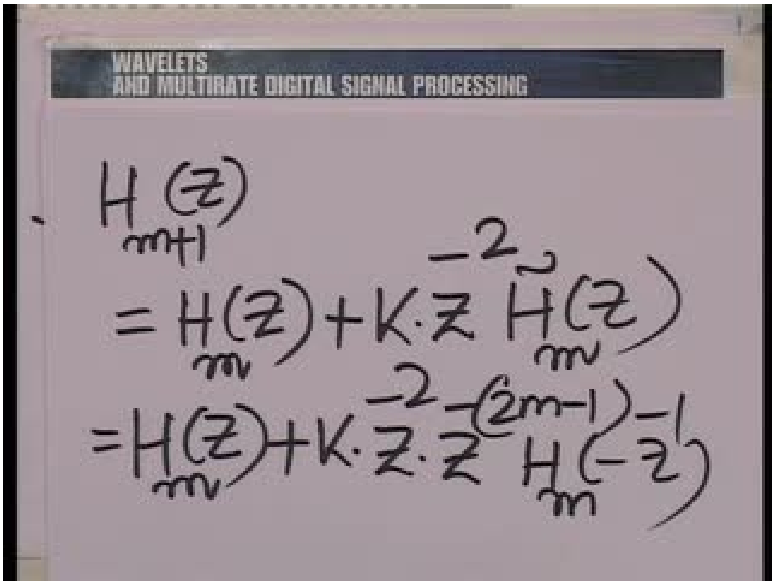


Now, we can see what happens. When we multiply again by this… So, we know what this is. Now, when you multiply that by Z raised to the power minus 2 and multiply by K once again, you are essentially shifting all these by 2 powers of Z and the multiplication by K ensures that the, the coefficient of the highest power here, is essentially K times h 0.

(Refer Slide Time: 51:53)

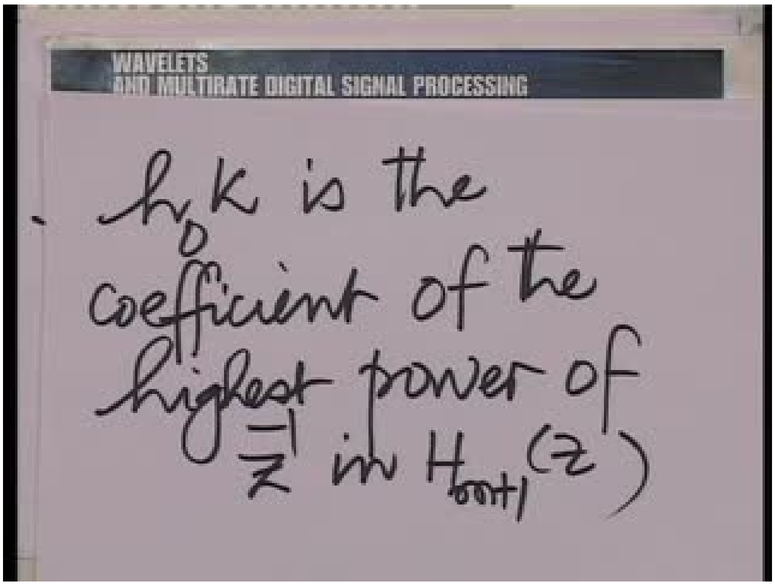


(Refer Slide Time: 46:04)



So, h 0 times K is the coefficient of the highest power of Z inverse in H m plus 1 Z. Let me put this back before you. What I am saying is, in H m plus 1 Z, K can only come with a highest power of Z inverse from here. So, therefore, we have now a mechanism to obtain K.

(Refer Slide Time: 52:30)



Once we have K, then, we now need to build a mechanism to peel off that last stage. That means, to go from, to use that K to build H m and H m tilde, from H m plus 1 and H m plus 1 tilde. We shall continue to do this in the next lecture and complete the construction of the lattice stage. We shall also extend this lattice structure to the synthesis filter bank and then complete the discussion on the lattice structure in the next lecture. Thank you.