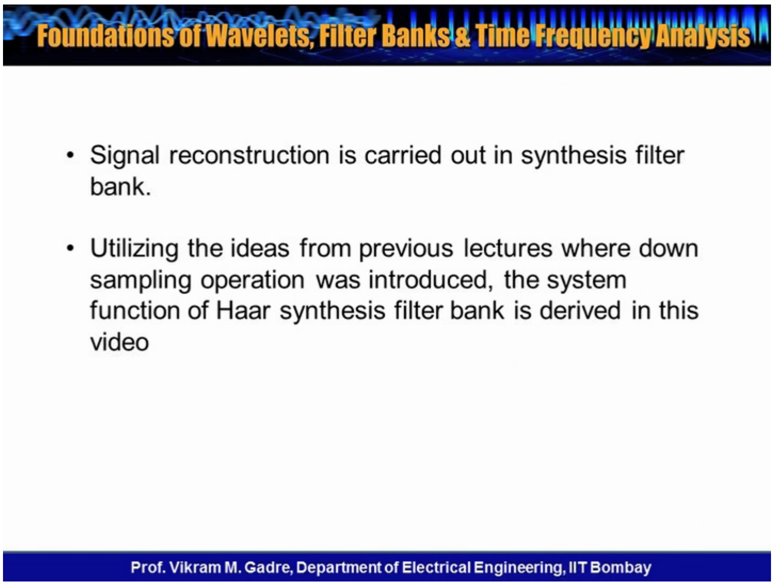
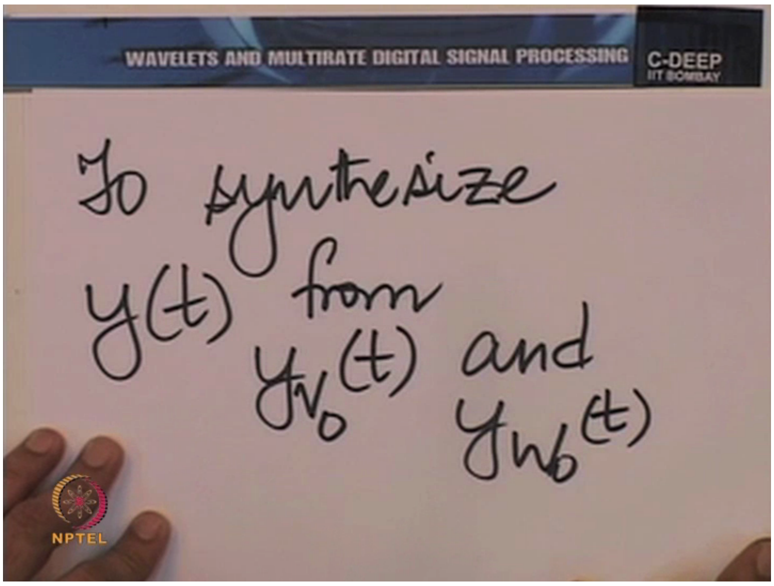
**Fundamentals of Wavelets, Filter Banks and Time Frequency Analysis.**
**Professor Vikram M. Gadre.**
**Department Of Electrical Engineering.**
**Indian Institute of Technology Bombay.**
**Week-2.**
**Lecture-6.3.**
**Hard Synthesis Filter Bank in Z-domain.**

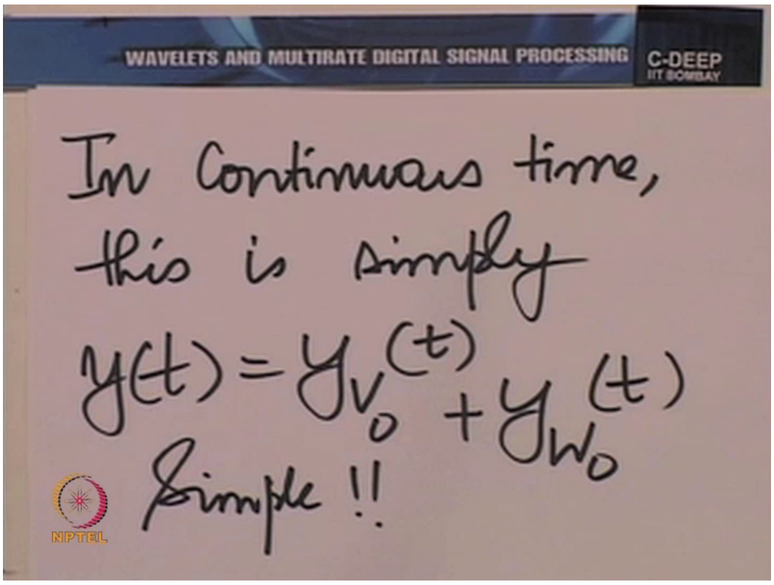
(Refer Slide Time: 0:18)




Now if we spend just a minute in reflecting, intuition tells us that this decimation needs to be outdone when we reconstruct, in some manner. What we mean by outdoing or doing away with the decimation? We will see, what happens as a consequence of that decimation process. In some sense, that decimation process has halved the n index. So, recall that that decimation process essentially brought index 2 to the index 1, index 4 to the index 2, index -2 to index -1 and so on so forth. We need to restore the indices back to their original place, that is at double the value. So, how do we do that?

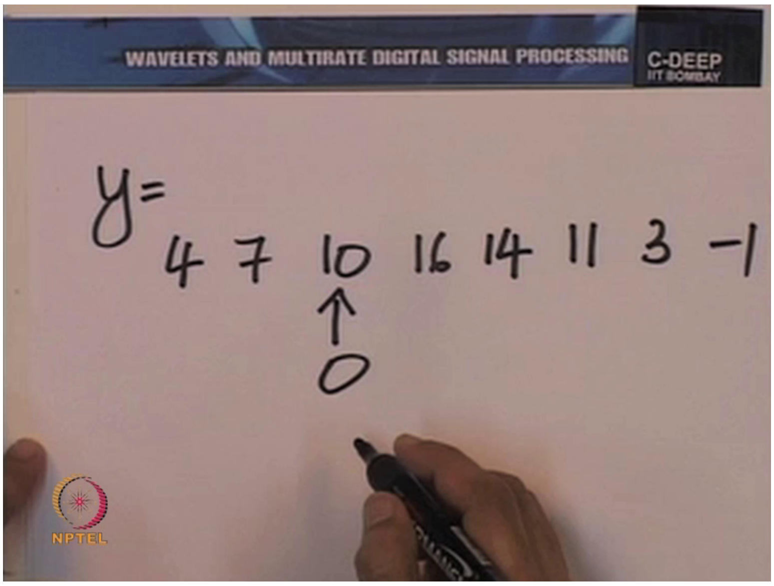
(Refer Slide Time: 1:27)



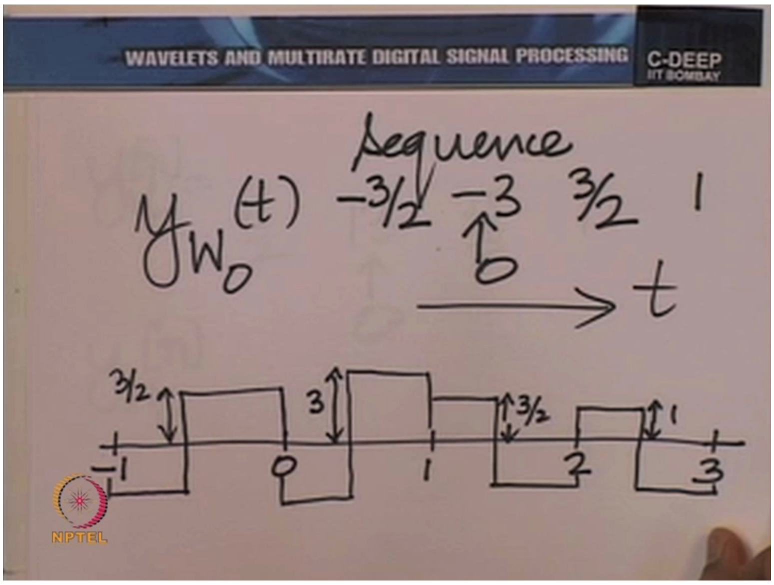

So, essentially, if you wish to construct a synthesis filter bank or to synthesise yt from y V0T and y W0T, actually in Time it is very easy, in continuous Time, it is very easy. In continuous Time, this is simply yt is y V0 t + y W0 t, simple. So simple, you just add them. But that is in continuous Time. When you have to do it in discrete Time, you have to work a little harder. So, in fact it may put back the sequences. You know, let us write the sequences explicitly now. We, what we will do is we will write down the sequences at least in that region from -1 to +3 explicitly for y, for yv0 and y W0.

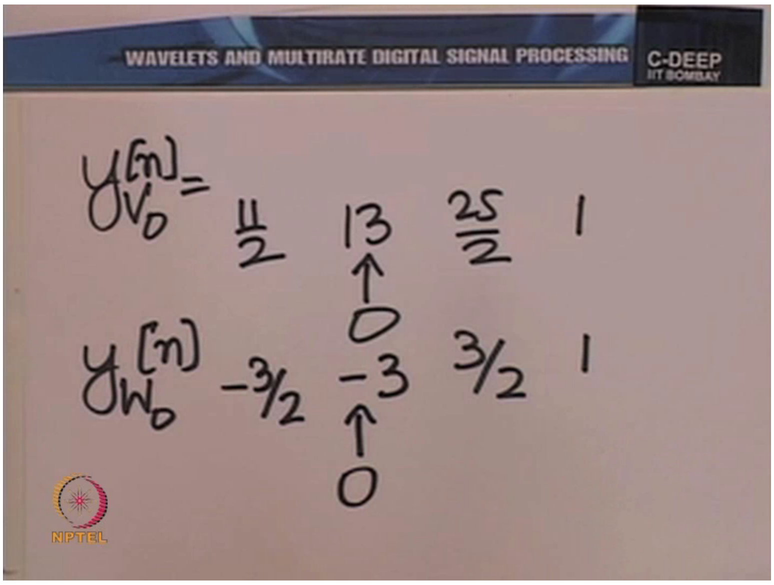
(Refer Slide Time: 2:57)




So, let us write the sequences explicitly. So, for Y, the sequence is essentially 10, 16, 14, 11, 3, -1 here and 7 for there and 0 marked here. Let us on the same reference write the sequence for y V0 and y W0. And then let us put them together. What I will do is when I write the sequence for y V0, I will need to do a little bit of work.

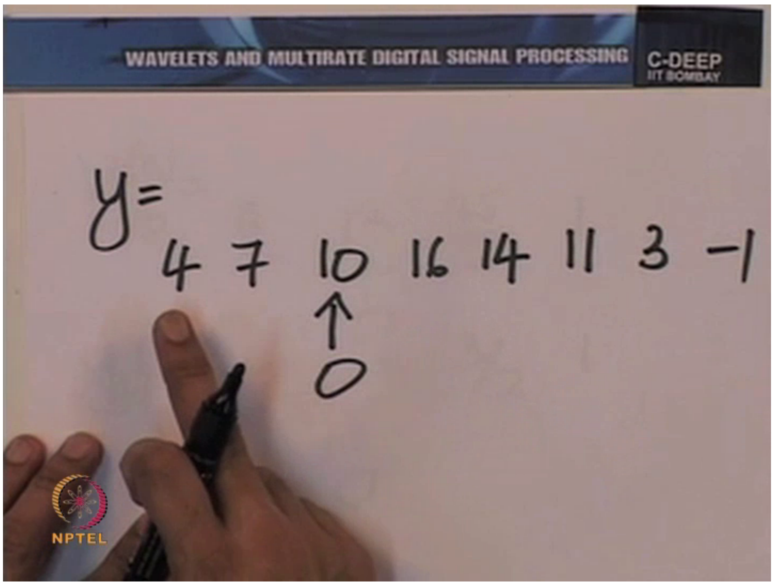
(Refer Slide Time: 3:45)





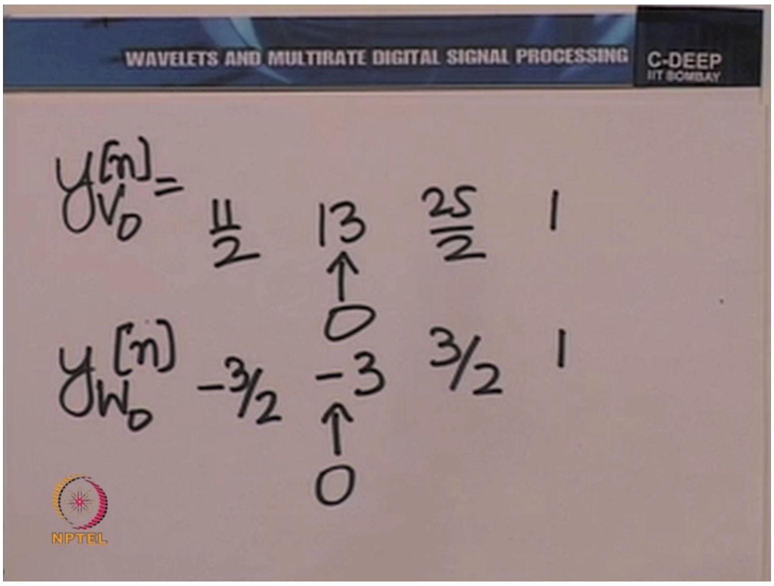

Now, y V0 is going to look like this, so, in fact I have a sequence here, let me rewrite it explicitly. So, I have the sequence for y W0 here and I put it down explicitly. Now you see what I mean.

(Refer Slide Time: 4:50)



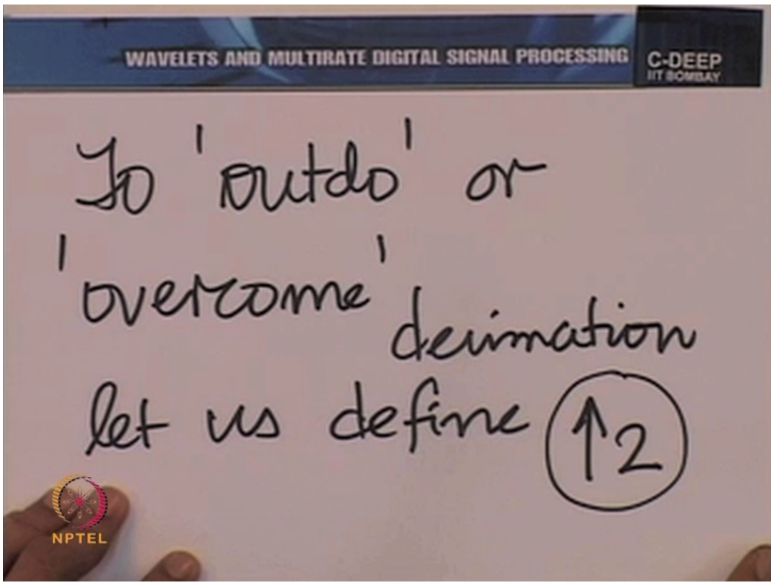

You see if you look at this sequence, it has in the interval from -1 to 3, 2+2+2+2, 8 values. And if you look at these 2, they have only 4. So, somewhere, we have to do an expansion and expansion is required because each of those points in the sequence corresponding to y was actually over half range, half an interval, half unit interval, whereas the sequence points in y V0 and y W0 are on a full unit interval. So, essentially what we need to do is to introduce spurious zeros here.

(Refer Slide Time: 5:30)



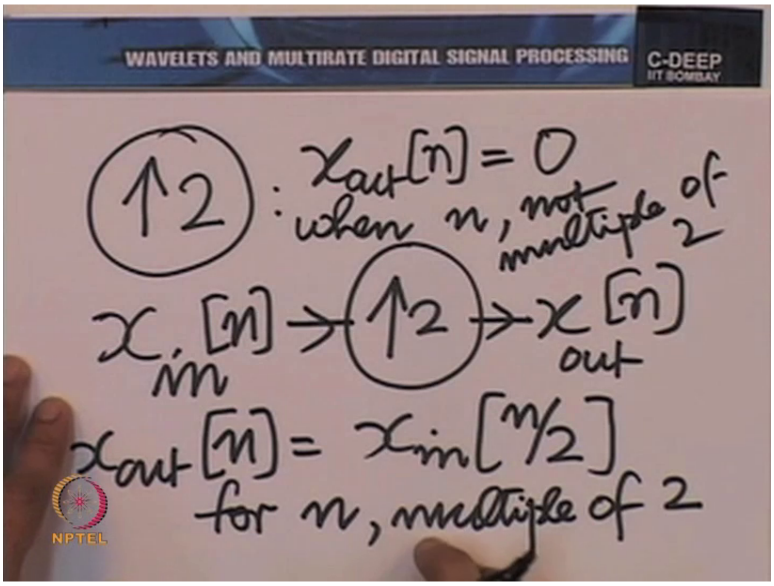

So, let us 1st carry out what is called a process of interpolation or upsampling. So, let us define with the intent of outdoing the operator of decimation, to outdo decimation so to speak to outdo or overcome decimation.

(Refer Slide Time: 5:51)



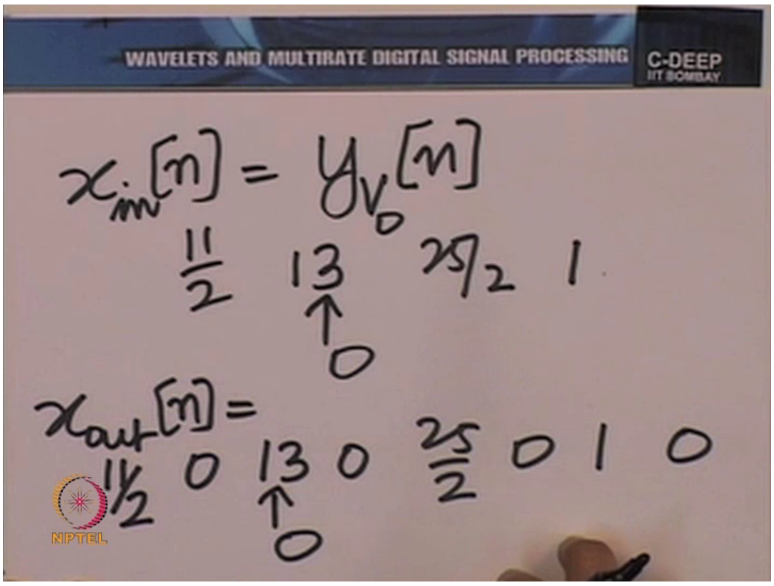

Let us define an operator which we shall denote by up and 2. Up essentially means upsample and what does up and 2 do, well up and 2 essentially introduces a zero between successive samples.

(Refer Slide Time: 6:33)



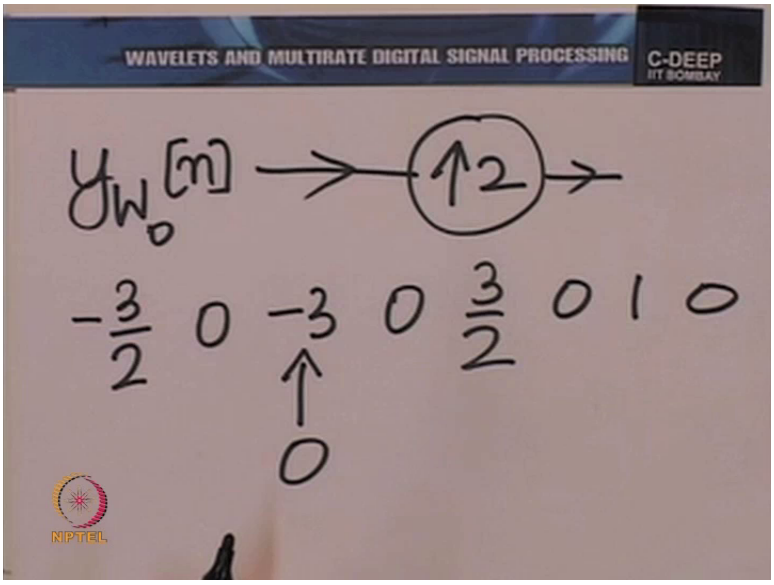

So, if you have x in as a function of n and you subject this to the action of up and 2 to produce xout as a function of n, then xout of n shall be equal to x in of n by 2 for n a multiple of 2 and 0 elsewhere, when n is not divisible by 2 or not a multiple of 2.

(Refer Slide Time: 7:36)



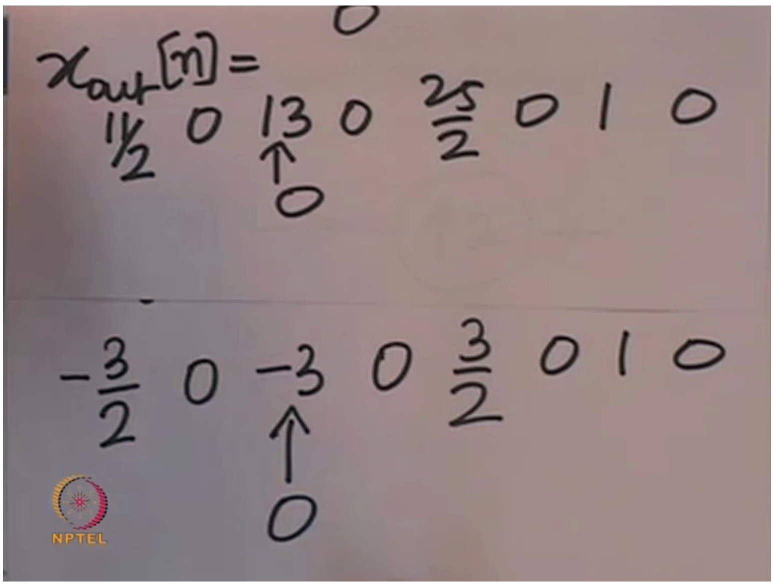

So, let us illustrate this operator. For example, suppose x in of n happens to be the following sequence, let us take the very sequence that we have for y V0, so I have y V0 n. So you have 13, 25 by 2, 1 and 11 by 2 marked with 0. What will xout n look like after upsampling by a factor 2, firstly it will have 8 locations. So, it will be 11by 2, then 0, 13 and then 0, 25by 2 and 0 and 1 and 0 and 13 would of course come back to 0. So, this is how the up 2 sequence looks for the sequence y V0. Similarly we can construct y W0 n upsample by factor of 2. Let us write that down explicitly.

(Refer Slide Time: 8:43)



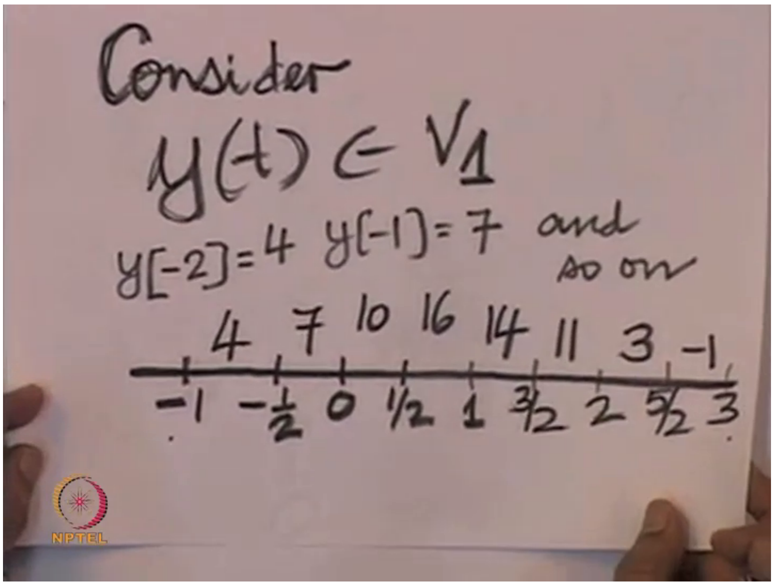

So, if we take y W0 n and upsample it by 2, the way we have done for y V0, we get the following, -3 by 2, 0, -3, 0, 3by 2, 0 and 1, 0 and 3 of course is located at 0. So, now we are in good shape. You see what we now need to do if you think about it, you see, you want to reconstruct y of n from these 2. And for example, let us look at these 2 upsample sequences, so this is the upsample sequence corresponding to V0 and this is the upsample sequence corresponding to W0. Now, how would we get the piecewise constant values from these 2 sequences? Well, let us take the example of the 1st 2 values here.

(Refer Slide Time: 10:40)



The 1st value in upsample sequence of V0 is 11 by 2, so 11 by 2 and 0 and -3by and 0. So, if we simply added them, you see, you can see that, if we simply added them, let me put them together for reference, you know, so, I will just suppress this for the moment. And we just put them together, just these 2 sequences. So, these 2 half intervals, now remember, these actually correspond to half intervals here. 11 by 2, -3 by 2, in other words, when you add these, that is 8 by 2 would give with the value of yt between -1 and half, 4.

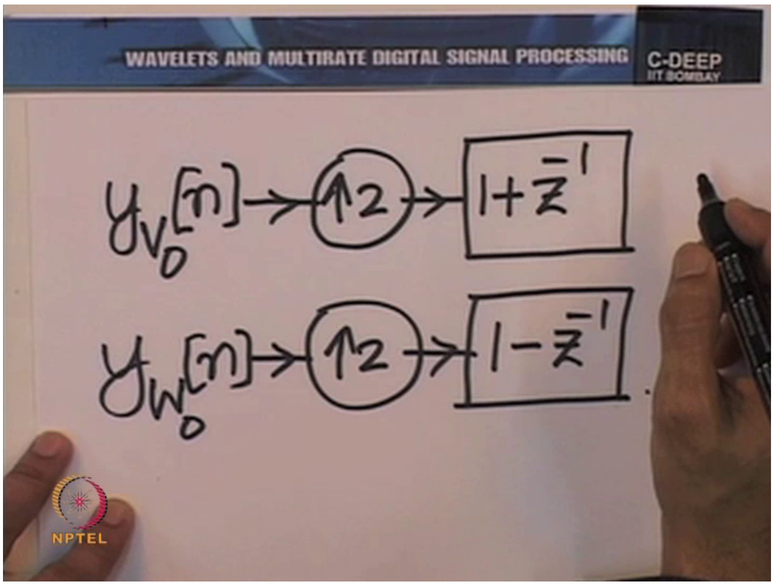
(Refer Slide Time: 11:28)




An 11 by 2 - of -3by 2, in other words, 14 by 2 would give with the value of yt in the interval -1 by2 to 0. So, what you need to do is you need to operate the sum filters, this + this for the 1st half interval and this - this for the 2nd half interval. Let me write this down explicitly. So, you see, it is interesting, what we are doing here, you must understand what is slightly different between analysis and the synthesis of filter bank. In the synthesis filter banks, it is as if one filter operates for one sample and the other filter operates for the next sample.

And therefore you could visualize a situation where you carry out both the filtering operations at once but then allow this sample or the samples from the upper filter to pass in one instance and the sample from the lower filter to pass in the next instance and keep alternating in this way. How do we express this is the language of discrete Time signal processing in the Z domain? Well, let us draw the diagram 1st.

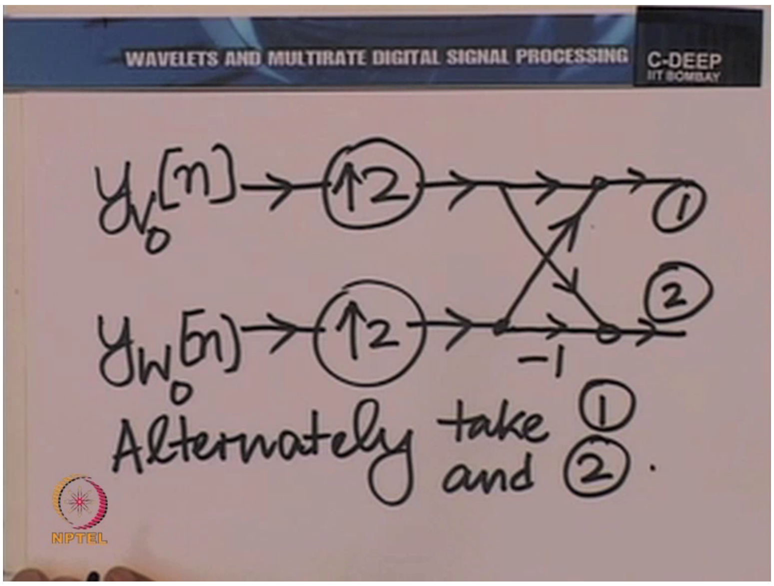
(Refer Slide Time: 12:43)




So, what we are saying is this, I have this sequence corresponding to V0, so y V0n and I have the sequence corresponding to W0, y W0 n, I have upsampled them.

Now, I also subject this, this I subject to the filter 1+ Z inverse, this is the filter where the output Bn and the input An are related according to Bn is An + An -1. And here I subject this to the action of the filter, 1 - Z inverse, alright. Well, I think it might be easier for us, so you know what we want to do I think maybe it will be a little difficult to see this directly. Let me instead put it down as an operation of addition and subtraction directly. So, we make a little change in this.

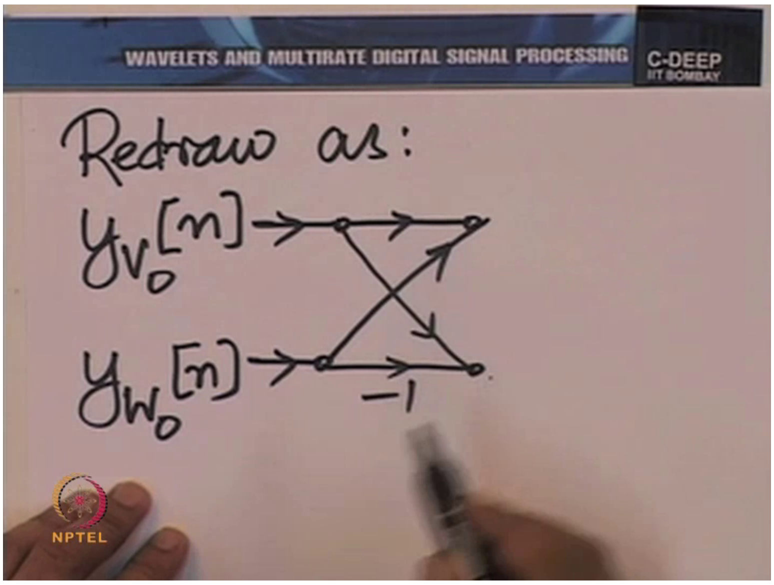
(Refer Slide Time: 14:14)



Let me say instead, I have y V0 n and y W0 n here, I upsample them and I add and subtract. So, now I get the outputs of upsampling, now I add and subtract, so, I have the sum of these outputs in one branch, this + this and this - this on another. This is the sum branch and this is the difference branch. And what I am saying is I pass the sum branch at one instance and I pass the difference branch on the next instance. And remember this is operating at twice the rate. You know, so this upsampling operator needs to be located slightly different, we have to interpret this property.

We are saying, essentially take the sum branch at that 2nth point and the difference branch at the 2n +1th point. So, let us write that down, let us call this the point 1 and the point 2. Alternately take 1 and 2, now how do we express this using the upsampling operator that we have? That is easy, actually if we notice this upsampling operator can commute with this, so you know, it does not matter if you 1[st] upsample and then add and subtract or if you 1[st] add and subtract and then upsample. In fact we would find it convenient to add and separate and then upsample. So, let us replace this.

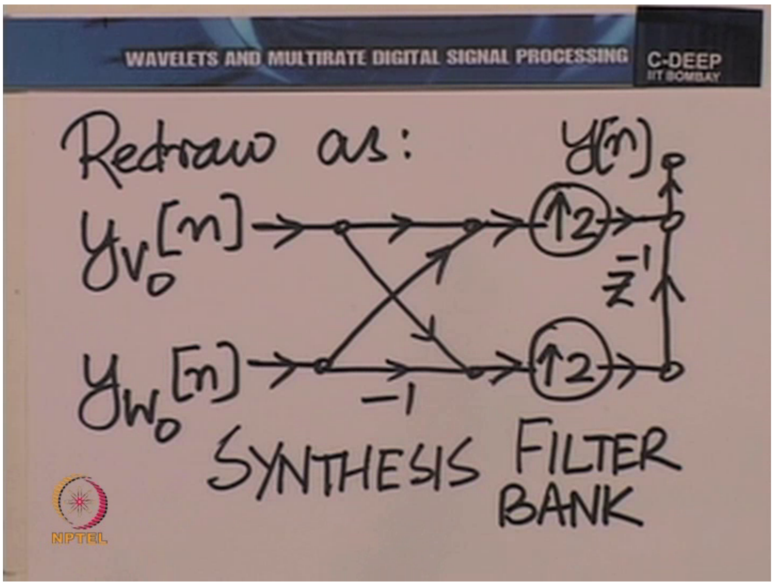
(Refer Slide Time: 16:31)




Let us redraw this. So, we have a sum and difference branch here. Recall this is what we call a signal flow graph. In a signal flow graph, we have nodes and we have edges, a signal flow graph is a convenient way of showing computation. You will recall that whenever you have a node at which you have multiple edges coming together, the content of the edges is added together, it is as if each edge starts from its source node and goes to its destination node multiplying what it carries from the source node by the multiplier on the edge and deposits it at the destination node. So, destination node, any node which has edges coming into it is equal to the sum of all the deposits coming from the edges.

And if many edges go out from a given node, all of them carry the value of that node with them multiplied by the multiplier on the edge.

(Refer Slide Time: 18:19)




Just a recapitulation of the signal flow graph notation. But coming back to this signal flow graph that we have drawn here, I have essentially these 2 nodes where 2 edges come in appropriately with multipliers, these are all multipliers of 1, this one of -1 and now I put an upsampler of 2 here for convenience. And indeed, what I want to do is to pass this as it is for the even instances but this I wish to pass for the odd instances.

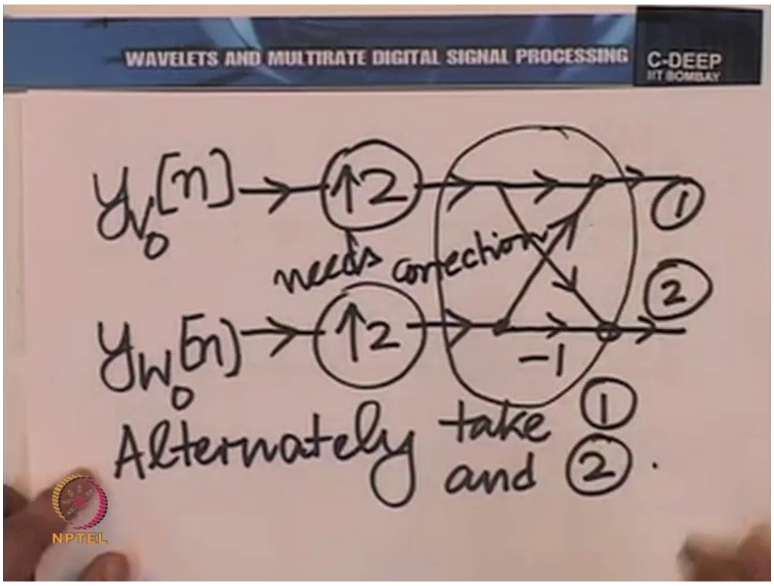
So, how do I express that process of passing for the odd instances? I express that by delaying this by 1 sample on the upsample rate, so here I need to put a delay operator in the Z domain which is Z to the power -1. What does this Z to the power -1 do, it shifts this by one step forward.

So, if you notice at the 2 nth point, it is this which will come and this would come not a 2 nth point but at the next point, 2 n +1 th point. And then you would have this together giving you the sequence that you desire, the sequence corresponding to Y, which is Yn. So, this is the synthesis, the synthesis process or the synthesis filter bank. It is interesting, you know if you notice, in the analysis filter bank, we put down the filters explicitly and then had a downsampling operation.

Here we seem to have the filters implicitly, so you have a combination operation being done 1st, essentially an add subtract kind of operation followed by an upsampler and then followed by some operation in the Z domain. So, in fact what we have done without realising it, is also brought out an efficient way of implementing the synthesis structure. I told you somewhere earlier in this course that the Haar multi-resolution analysis and its derivates, that is the filter bank coming out of the Haar and the other concepts that I had stated from the Haar, illustrate several different ideas, several different common concepts in multiresolution analysis very very lucidly and this is one example.

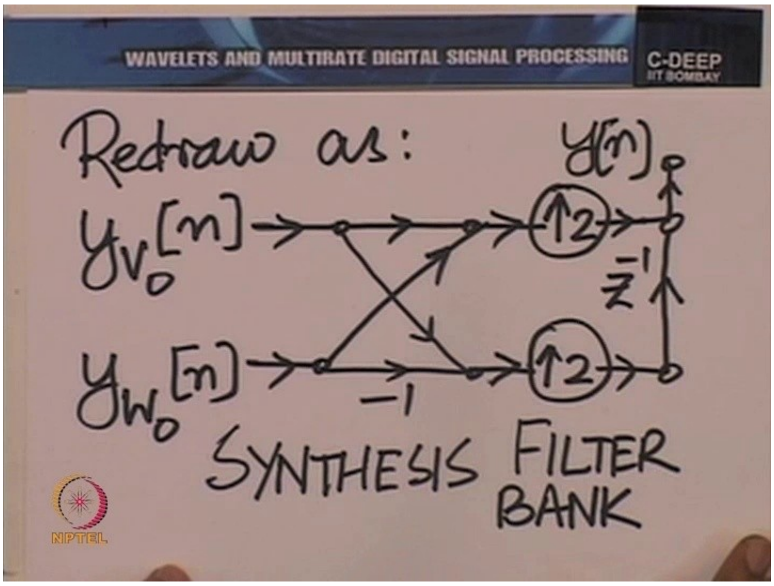
Here I have a beautiful decomposition of the synthesis filter bank into what is called its polyphase components lying ready for me right here. Now this word polyphase we will illustrate in greater detail later. What I am trying to say is that the structure which I have drawn here for the synthesis filter bank actually gives us an efficient way of computing or representing the computation of the synthesis filter bank. And if we only take a minute to understand that this upsampling operation can jump back and forth, you know, it can jump back and forth here.

(Refer Slide Time: 22:14)



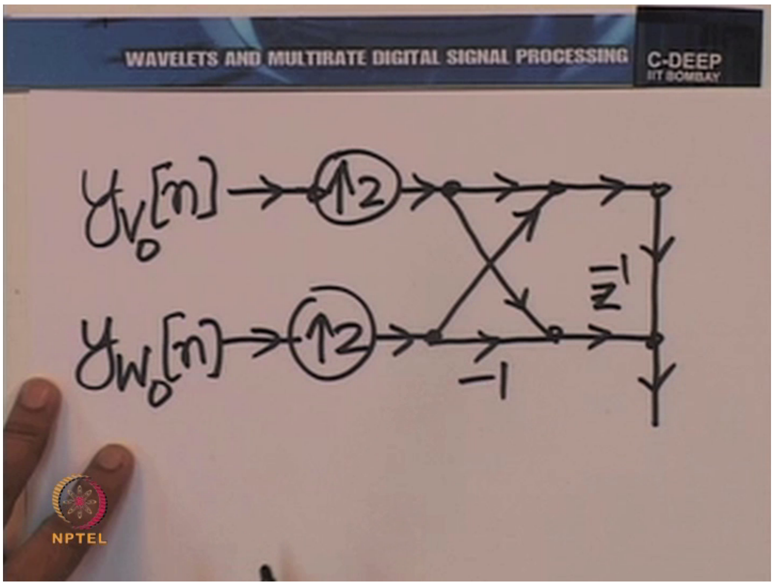

So, you know, here, when we drew this diagram, we were not quite correct. This is intuitively okay but this needs to be corrected, needs correction.

(Refer Slide Time: 22:33)



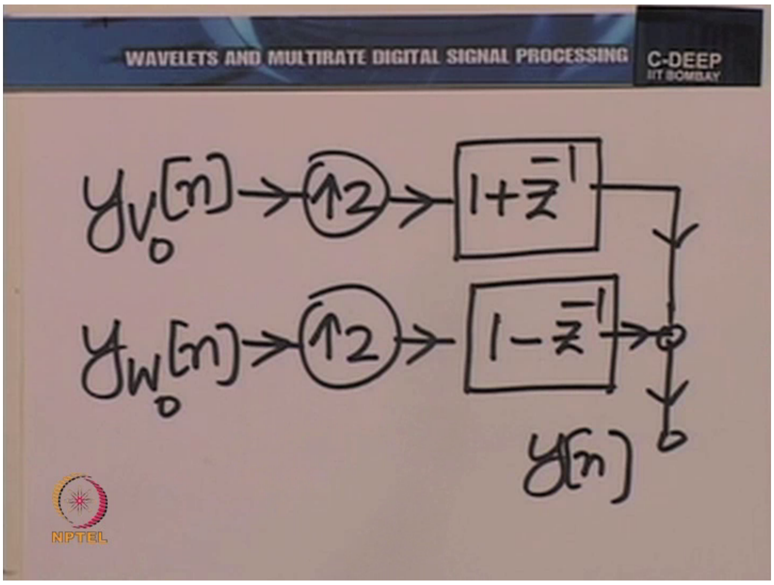

In fact now we have the corrected version emerging from here and let us draw that corrected version by placing the upsamplers back here, remember the upsampler is commuted with this operation here. So, if we do that, we have the following structure.

(Refer Slide Time: 22:47)




Y V0 n, y W0 n, upsample by 2, follow it with a sum and difference operator but remember there is Z inverse there and this is what is added. So, if you see, if you like, you can bring the addition here, does not matter. So, look at what is happening. What is the operator acting on y V0, what is the operator acting on this branch? This branch comes here with a transmissal of 1 like this and a transmissal of Z inverse like this and this branch comes here with the transmissal of 1 like this and transmissal of - Z inverse like this. So, in other words, what we have is the following structure with which we shall then conclude today's lecture and look more deeply in the next.

(Refer Slide Time: 24:30)




The structure which we have is this. We have an upsampler, we have a 1 + Z inverse operator coming there and a - Z inverse +1 operator coming here. And these are being added to produce yn. We shall delve further into this structure in the next lecture. But what we have done just done a minute ago is to construct a synthetic filter bank for the Haar multiresolution analysis. We shall build this further in the lecture to follow. Thank you.

"Professor – student conversation starts"

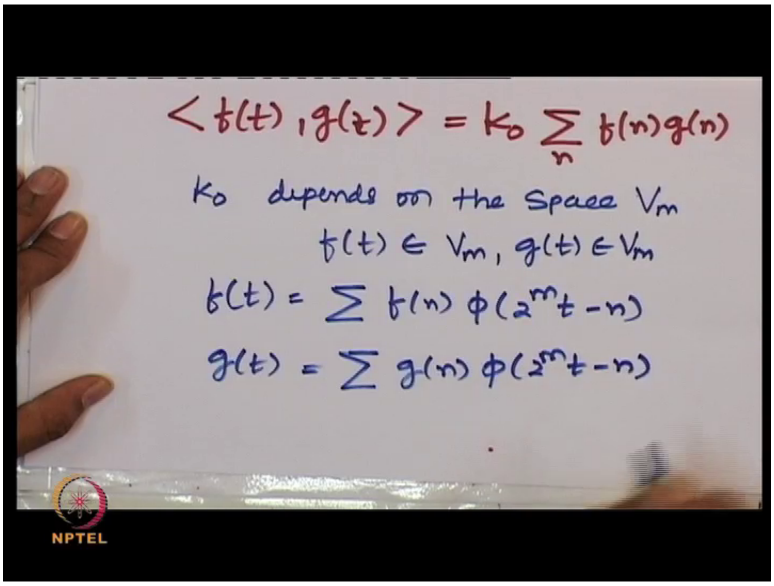
Hello everyone, I am Nikunj Patel. You can ask any questions or doubts if you have regarding the material covered in the class.

Hi Nikunj, in the equivalence relationship between functions and sequences, can you elaborate on the constant term that appears in the inner product of sequences?

Yah, we can derive a constant term identically using the fact that the basis functions for the sequences are orthonormal.

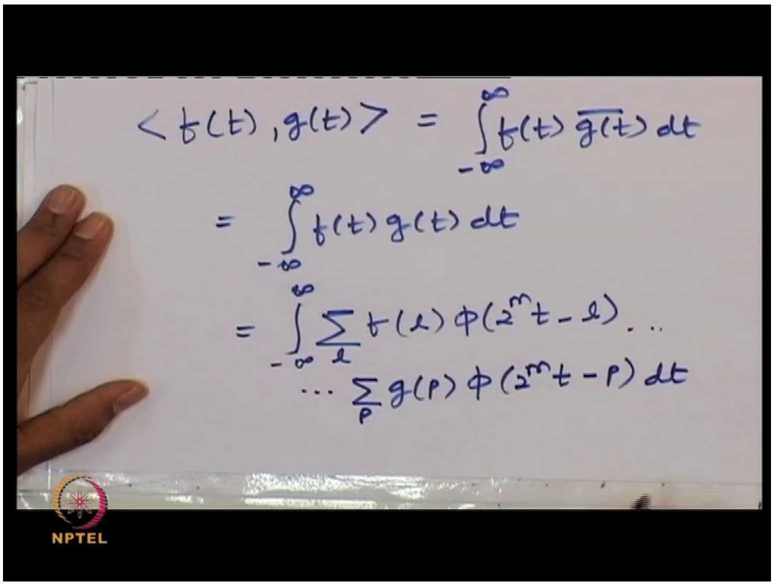
We want to derive the value of K0 that exists in the equivalence relation between inner product of functions and sequences.

(Refer Slide Time: 26:13)



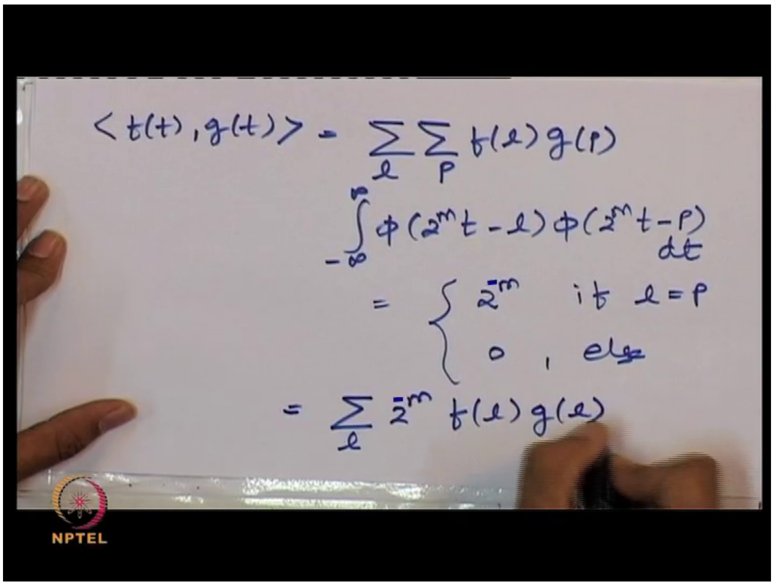

K0 generally depends on the space Vm in which our ft and gt belongs. ft is equal to Sigma fn phi 2 raised to mT - m. gt is equal to Sigma gn phi 2 raised to mT - m.

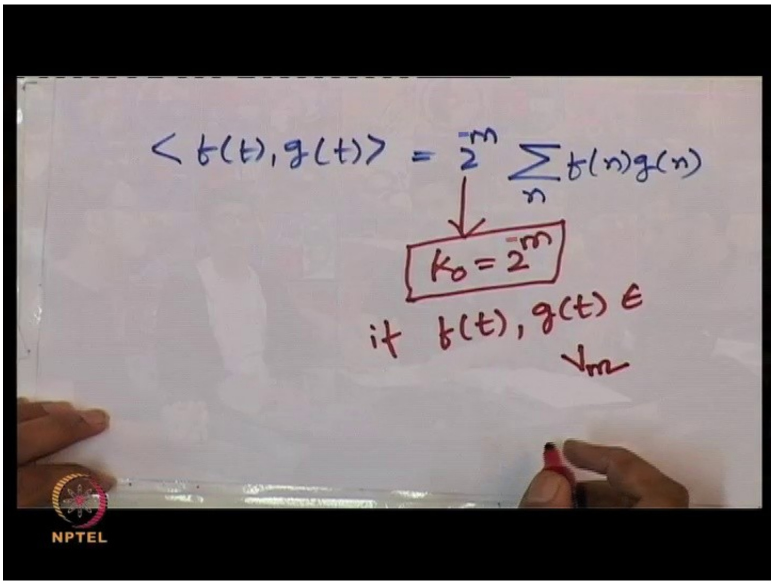
(Refer Slide Time: 27:25)




Inner product between ft and gt is equal to integral of ft g conjugate t dt but here we are assuming ft and gt are real sequences and complex conjugate will be same as gt. So, it is equal to ft gt dt, we will try to substitute ft and gt in terms of the basis function. L will be dummy variable phi 2 raised to mT - L and Sigma P GP phi 2 raised to mT - P dt.

(Refer Slide Time: 28:45)

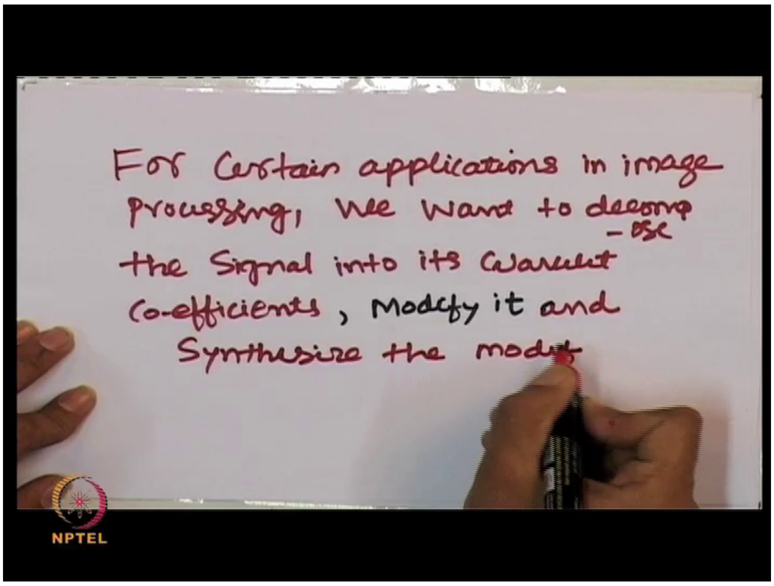



We will take gp and fl common and this results in double summation fl gp and their inverse running t equal to… We know that phi is an orthonormal basis for VM and this is equal to 2 raised to - m if and only if L is equal to P, 0 else. And hence this double summation results into single summation. 2 raised to - m fl gl, hence we proved that F, inner product of ft gt is equal to 2 raised to - m Times Sigma, we will replace this dummy variable L with m. This is the value of K0 that we want to find out. K0 is equal to 2 power - m is ft and gt both belongs to VM. I had a query that in the Haar filter bank, why do we analyse this over, only to synthesise later?

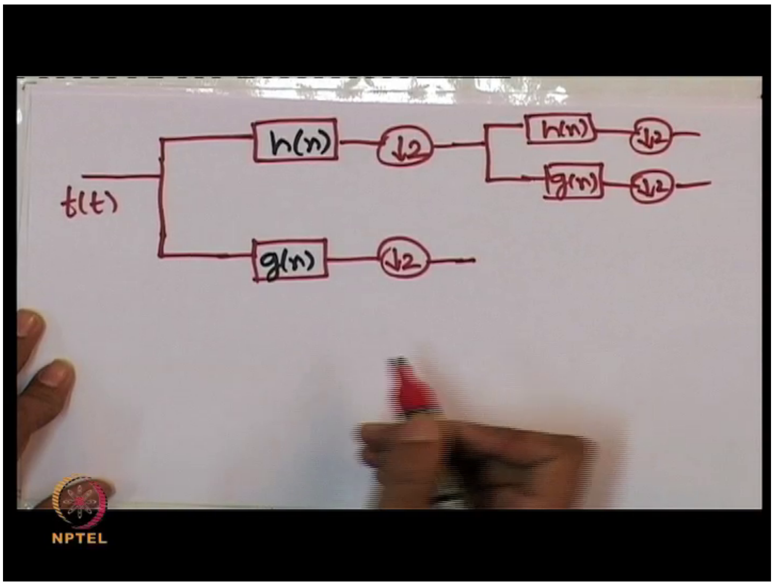
It generally depends on the application. After analysing the signal using wavelets, we are modifying the signal and then we are synthesising, means it depends whether we are doing denoising, compression or any other application. So, some applications we are $1^{st}$ analysing, processing the processions and then we synthesising the signal bank, so that we get the desired signal.

(Refer Slide Time: 30:55)



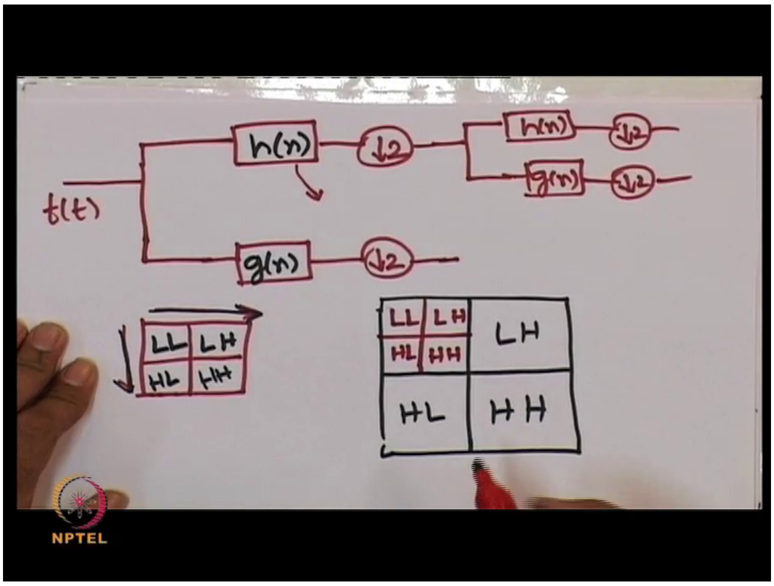

For certain applications, we want to decompose the signal, in for example image processing, we want to decompose the signal into its wavelet coefficient, modify it depending on application. And synthesise the modified coefficients.

(Refer Slide Time: 32:07)




For example, consider 2 band filter bank, we can further analyse this lock depending on the application. Henc,e for example, if we apply this filter bank to the two-dimensional signals such as image, then we will apply this filter hn to the image row wise and then column wise, for example if we apply 1st hn row wise, downsample it and then apply hn again column wise downsample it, then we will get the low low band of image.

(Refer Slide Time: 33:33)

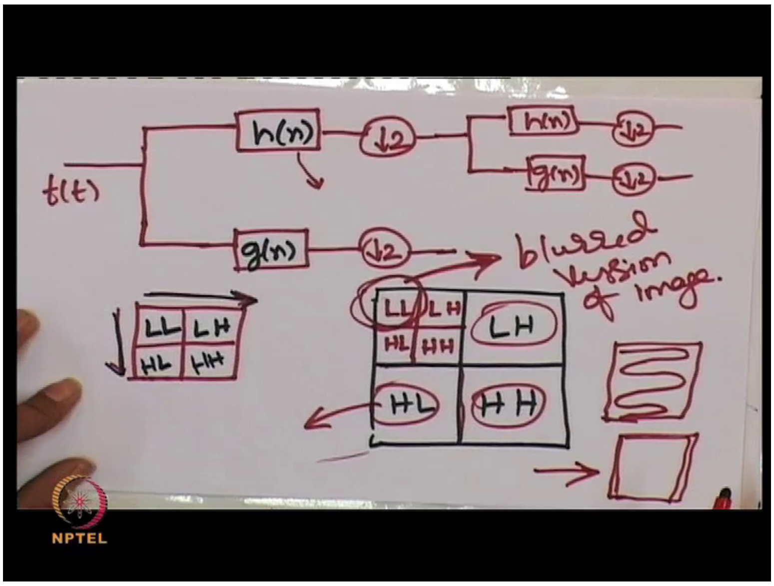



L L, L for lowpass because we are applying hn which is a lowpass analysis filter twice. If we apply lowpass filter hn provides downsample it and then apply high pass filter gn column wise, then we will get LH band. We will repeat this for high pass filter row wise and lowpass filter column wise and we will get HL. And if we apply high pass filter both row wise and column wise, then we will get HH band. We can further decompose this LL band into its sub bands.

What is the importance of this, for example if you have low resolution phone and your phone cannot display high-resolution images, then in communications, we are transmitting only the LL band of an image, it will be sufficient for a phone, for a low resolution phone to display this image using low, using less number of bits.

What if you have a high resolution phone, then we are sending the whole image as such and if you want to reconstruct the high-resolution image, then you can add this detailed components into the image and reconstruct it. If you want to see a low resolution, then you can see this LL band, it will look like an image but the details are missing in this, it is somewhat a blurred version of the image.

(Refer Slide Time: 35:35)




This has a very good application in Web, for example in 1990s, you were seeing that the image in a phone comes like from top to bottom, it is coming like this. And at the end, you are going to see the whole image.

But nowadays you all see that on Facebook or Whatsapp, they are showing you a blurred version of image. Means 1st the LL band is transmitted and then this LH band is transmitted 1st, so if you receive the LL band 1st, then you will get a blurred version, blurred version is displayed 1st. And after receiving this detailed band, they will add the detailed band to the image and you will get the full detailed high-resolution image. Thank you.

"Professor – student conversation ends."