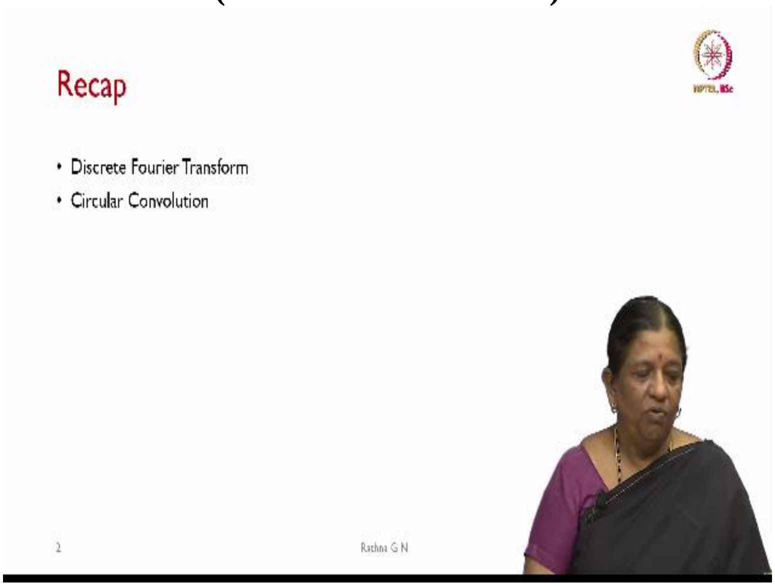
Real - Time Digital Signal Processing
Prof. Rathna G N
Department of Electrical Engineering
Indian Institute of Science - Bengaluru

Lecture - 25
FFT - 1

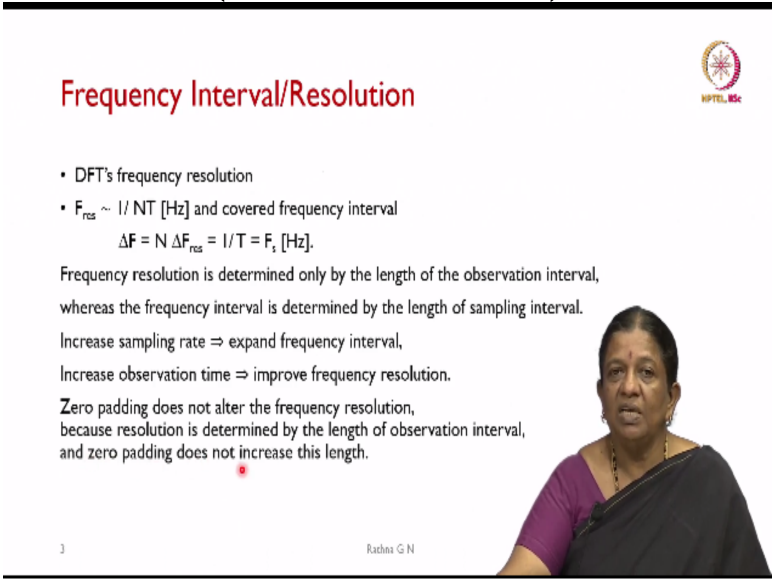
Come back to real time digital signal processing course. So, we have discussed about DFT in the last class, today we will try to cover FFT.

(Refer Slide Time: 00:32)



So, to have a recap we discussed about Discrete Fourier Transform in the last class and then how to use that, for circular convolution to give a flavour of it.

(Refer Slide Time: 00:43)

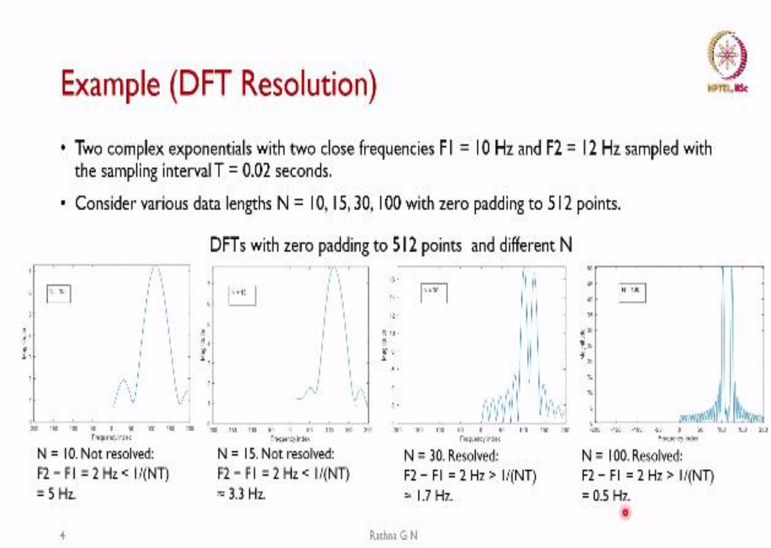


So, we will see the resolution what we discussed, how the interval or resolution in the frequency domain is going to get affected, whether it gets affected by padding 0s or not what we will look at it. So, what we have here is how to check the frequency domain resolution. So, we call F as the resolution which is approximated as 1/ NT, we call this as hertz T is a sampling period and then N is the number of DFT points what we will be taking it.

So, which we can give it as ΔF as the resolution, name with F resolution then we will be having N ΔF_{res}. So, this is the time period between 2 samples. So, which is nothing but 1/ T which is nothing but F_s [Hz]. So, what does it say frequency resolution is determined only by the length of the observation interval that is what we call it as N here? Whereas, the; frequency interval is determined by the length of the sampling interval.

So, increasing the sampling rate, it means that; expand the frequency interval and increase observation time, which will help improving the frequency resolution. So, 0 padding does not alter the frequency resolution. So, because resolution is going to be determined by the length of the observation interval that is whatever N what we have chosen and 0 padding does not increase this length.

(Refer Slide Time: 02:38)



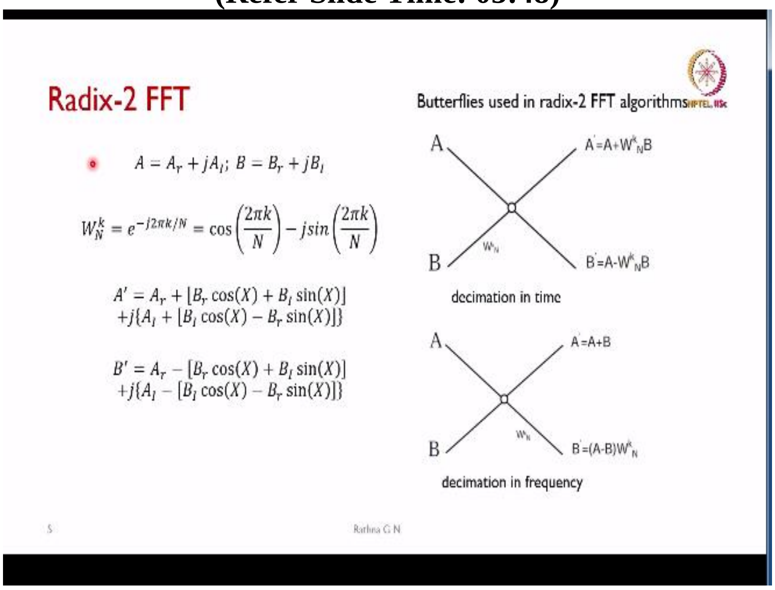
So, to show this, so one can run MATLAB equation at hope see that how it is going to affect the resolution so, we will take 2 complex signals with close frequencies that one is with 10 hertz and one F2 with 12 hertz, sample with the sampling interval T= 0.02. So, it is going to be approximately 50 hertz. So, consider various data length N = 10, 15, 30 and then 100 and we will pad these N with rest of the values as 0. So, that it becomes 512 points and then run it in MATLAB as you can see, for N = 10.

So, what is the frequency we are going to have F2 - F1 is what we have is 2 hertz. So, what is our resolution it is 1/(NT), which is equivalent to 5 hertz. So, hence these 2 signals are not bifurcated, as you can see only 1 is shown. So, as this thing what it is showing 10 times this basically as a 10 hertz thing, so, you are seeing it as 100 here in this. So, coming to the next one when we substitute N = 15 in algorithm and then run still if this is not resolved, why because, 1 by NT = 3.3 hertz.

So, that is the 2 hertz is less than the interval between the 2 samples. So, still you represent your 10 and 12 as approximately 10 hertz itself and coming with N = 30. So, partly as we can see it is resolved so, we are going to see 2 peaks in DFT domain within 30 samples and then rest of them are 0 padded to make it 512 points. So, it is what is it resolution and 1/(NT) which is equal to 1.7 hertz. So, you will be which is 2 hertz is greater than this. So, hence you will be seeing 2 waveforms peaks here which is appearing.

So, when we go to N = 100 so, you will be clearly seeing them 2 peaks. So, it is dropping down to 0 and then again raising. So, this will be at 10 hertz and this will be at 12 hertz. So, what is the resolution here we have it, 1/(NT) is 0.5 hertz. So, as you can see that whatever we claimed in the previous slide that is 0 padding is not going to alter the frequency resolution, that is what is shown in the next slide.

(Refer Slide Time: 05:48)



So, coming with the next why we have to go for FFT. So, initially we will take up a radix-2 FFT. So, will see that how we are going to derive fast Fourier transform algorithm for DFT as we discussed in the last class, the computation of DFT is order of N squared actually that is complex multiplication and addition. So, we will be seeing that how we can reduce it using butterfly structure. So, we are representing 2 butterfly ways of representing it, one is in the decimation in time what we will be looking at it?

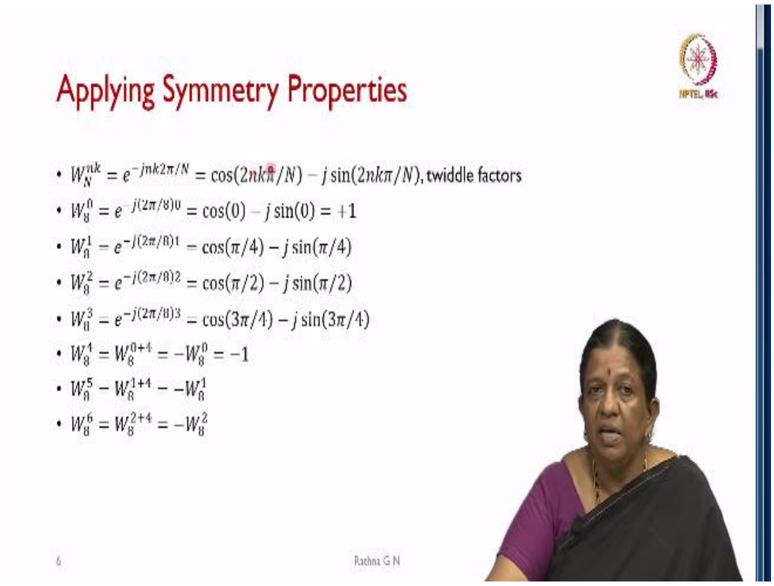
So, your twiddle factors W_N^k is going to be fed here. And whereas, in the decimation in frequency it is at the output what you will be feeding it in so both have same concept and then what is the output here, so, we know that A will have a real component plus the imaginary component what we are feeding A is equal to and then B value also will be providing real and imaginary that is, inputs are complex in nature. So, what will be the output we will see in a while which we call it as A' here.

So, we will be seeing that twiddle factors W_N^k is nothing but $e^{-j2\pi k/N}$. So, which is expanded in cos and sine you will be seeing it $\cos\left(\frac{2\pi k}{N}\right) - jsin\left(\frac{2\pi k}{N}\right)$. So, output A' what we call it when we expand this A with real and then imaginary and B also with the real and imaginary and W_N^k with the cos and sine function. So, this is what, what we will be getting it that is $A_r + [B_r \cos(X) + B_I \sin(X)] + j\{A_I + [B_I \cos(X) - B_r \sin(X)]\}$.

So, you will be seeing that same way even the B' is going to be represented with a real and then imaginary part separately. So, from this you will be seeing that how many multiplications that is what we have real multiplication and addition what you can count from it? So, I will be having 1, 2, 3 and then 4 multiplications and then you will be seeing that 1, 2, 3 and then for the real part 2 additions and imaginary part 2 additions even taking into account my subtraction also.

So, this is how what you can separate and then B' also will be having the same amount. So, what we say is one complex multiplication leads to 4 real multiplications and 2 additions what we are going to have it.

(Refer Slide Time: 09:04)

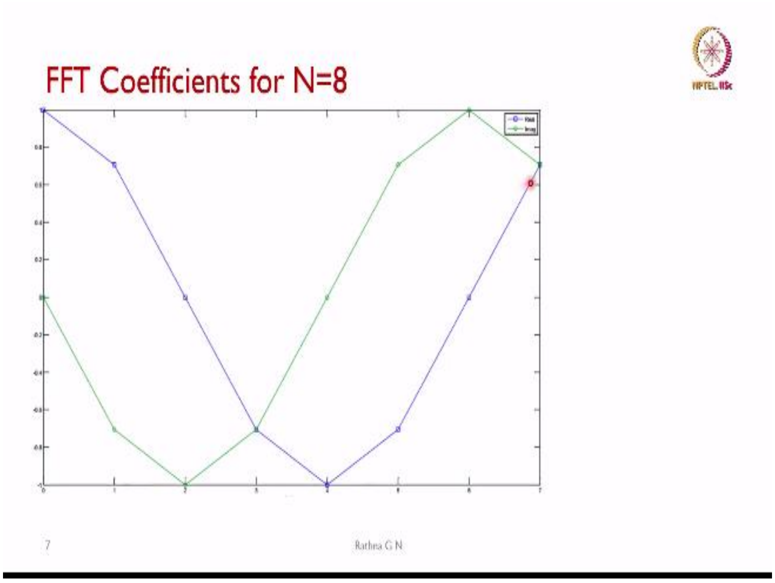


So, if we apply the symmetry property, so then we will see how we are going to make twiddle factors look like. So, W_N^{nk} is given as we know that it is $e^{-jnk2\pi/N}$. So, nothing but cos and sine what we have taken the thing, so how we represent W_8^0. So, to show that butterfly structure we will be taking N = 8. So, this will be $e^{-j(2\pi/8)0}$, which is nothing but $\cos(0) - j\sin(0)$. So, we know that $\sin(0) = 0$ so $\cos(0) = +1$ so we will be getting 1.

And then W_8^1 so what it is going to be $\cos(\pi/4) - j\sin(\pi/4)$ and then W_8^2 will be represented as this and then this is nothing but $\cos(\pi/2) - j\sin(\pi/2)$. So, and then W_8^3 what we have is $e^{-j(2\pi/8)3}$, which is nothing but $\cos(3\pi/4) - j\sin(3\pi/4)$. So, how many coefficients we have calculated $N/2$ in this case for N = 8. So, the rest of the thing that is 0 to 3 you can compute.

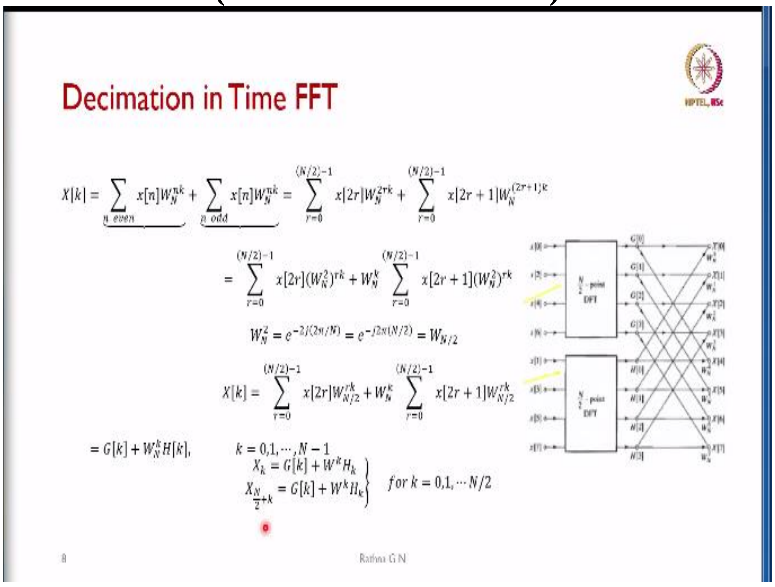
A rest of them what you can do is, you can calculate it as W_8^4 is nothing but W_8^{0+4}. So, which is nothing but $-W_8^0 = -1$. So, you will be using the symmetry property here to get the rest of the coefficients same way W_8^5 and W_8^6 are calculated and then W_8^7 remains same.

(Refer Slide Time: 11:09)



So, to see that how the coefficients are calculated both cos and sine coefficients for N = 8 has been plotted in this case. So, we know that sine starts from 0 and then $\pi/4$ here and then $\pi/2$ and then next is $2\pi/3$ and then how we will be going with values? And then cos we know that it starts from 1 basically, and then how rest of the thing $\pi/4$ and then now $\pi/2$ what you are seeing it in this case?

(Refer Slide Time: 11:55)



So, these are the 8 coefficients, what we will be having it for 8 point DFT that is discrete Fourier transform, which will be deriving it for Fast Fourier Transform basically. So, we will see first decimation in time. So, we said there are 2 ways of doing it. One is decimation in time, the other one is decimation in frequency, so first we will take the time part of it. So, how we are going to bifurcate that?

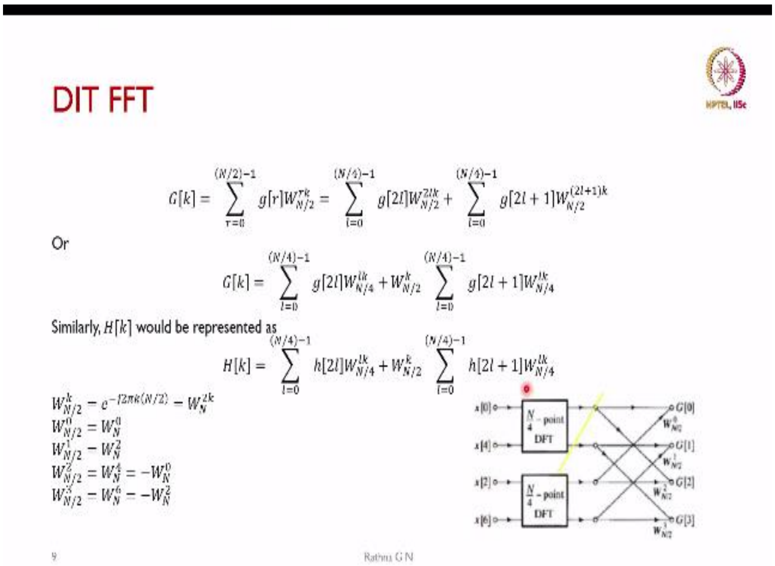
So, $X[k] =$ we can take even parts and then the odd part separately. So, that is what, what we need, so 2 summations what we will be splitting it into, so, which is equivalent to $\sum_{r=0}^{(N/2)-1} x[2r]$ here N is replaced with r basically. For even it is going to be 2r, for odd part of it, it is going to be 2r + 1. So, this is $(N/2) - 1$ and here also it is going to be $(N/2) - 1$ and we have substituted r = 2r.

So, this will be W_N^{2rk} and here what we will be having $W_N^{(2r+1)k}$. So, now we will see that by combining it that is 2 we are taking it out inside in this equation and rk goes here we have separated it out with $(W_N^2)^{rk}$, the other term is W_N^k, basically, so which is not dependent on r, so we can take it as a constant and then take it outside in this equation. So, now we will see what is going to happen with W_N^2, which is nothing but $e^{-2j(2\pi/N)}$, which is when you split the thing, it is going to be $e^{-j2\pi(N/2)}$.

So, we will name this twiddle factor as $W_{N/2}$. So, when we substitute that, so, we will be ending up with $x[2r]W_{N/2}^{rk}$ and W_N^k the other one is $x[2r+1]W_{N/2}^{rk}$. So, we named this as $G[k]$ and this is $H[k]$, multiplied with W_N^k what is shown in here. So, now, what is it k is still going between 0 to N - 1. So, whether we can also split this that is what, what we are looking at, then what happens $X_k = G[k] + W^k H_k$ and then if we go k to $\frac{N}{2}$ then it is going to be $X_{\frac{N}{2}+k} = G[k] + W^k H_k$.

So, these are the 2 terms what we will get it when we split k 2 0 to $\frac{N}{2}$ and the other parties $\frac{N}{2} + k2$ the values what will be representing it.

(Refer Slide Time: 15:18)



So, now continuing with the thing the it part of it, so, we are naming $G[k]$ is this will give it as $\sum_{r=0}^{(N/2)-1} g[r]W_{N/2}^{rk}$ and then we are going to split this into 2 parts again even and then odd part $G[k]$. So, then we can give a new notation that is l actually. So, this will be $\sum_{l=0}^{(N/4)-1} g[2l]W_{N/2}^{2lk}$. So, we are bifurcating, dividing by 2. So, it is going to be $(N/4) - 1$ and then this is going to be $\sum_{l=0}^{(N/4)-1} g[2l+1]W_{N/2}^{(2l+1)k}$.

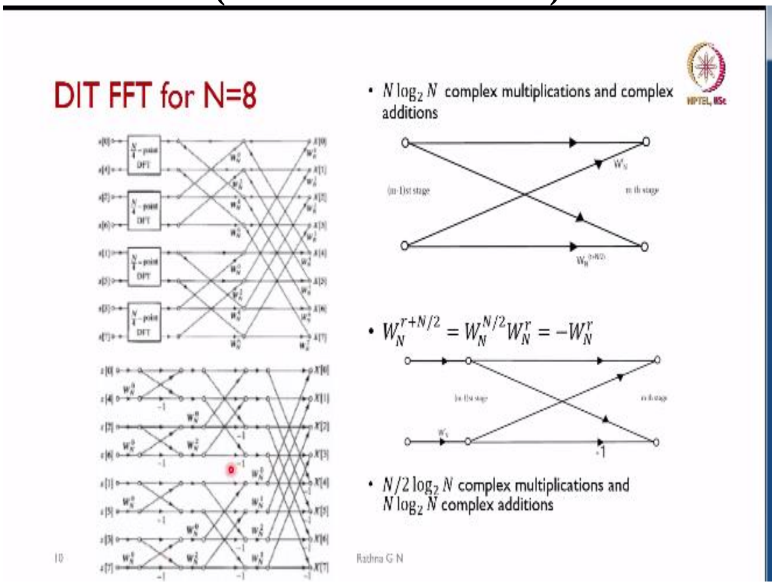
So, same way if a represent the thing; $G[k]$ now so we can have it as earlier splitting. So, we will be putting it as $W_{N/4}^{lk}$ you have to recall that we named it as $W_{N/2}^{k}$ earlier now, it will be lk bottom multiplied by 2 it is going to be $N/4$ and then this is $W_{N/2}^{k}$ and again $g[2l+1]W_{N/4}^{lk}$. So, same way, we can split $H[k]$ so, because we had both $G[k]$ and then $H[k]$ here. So, first we have bifurcated $G[k]$ into 2 parts $H[k]$ we have to bifurcate.

So, when we do that, it will be ending up in $h[2l]W_{N/4}^{lk}$and then this becomes as you can see, this is this was W_N^k. Then when we split the thing, so, it will be becoming $W_{N/2}^{k}$ this is the constant what we will have it twiddle factor and this is normal. So, then what happens, so, how we are going to define are all these coefficients $W_{N/2}^{k}$ is nothing but $e^{-j2\pi k(N/2)}$, which is nothing but W_N^{2k}.

And same way $W_{N/2}^{0} = W_N^0$, $W_{N/2}^{1} = W_N^2$ and $W_{N/2}^{2} = W_N^4$, which is nothing but $-W_N^0$. And $W_{N/2}^{3}$ is nothing but $W_N^6 = -W_N^2$. So, you are seeing that this $G[k]$ is this $N/4$ point DFT. And then $H[k]$ the odd part of it is shown here.

And further, we will be getting $G[0]$, $G[1]$, $G[2]$, $G[3]$. So, you are seeing this as input is $x(0)$ and then here $x(4)$, and then $x(2)$ and then $x(6)$ for the even part where you are bifurcating $G[k]$ into even and then odd in this case.

(Refer Slide Time: 18:42)



So, coming for N = 8 now we will split the thing. So, what we have is, this is my $N/4$ point DFT. What you have it on the left hand side and then we will be combining weights in this case, and then this is the final output. So, how we will be feeding it? So, we had for the 2 of it. So, the other $H[k]$ is split into 2 stages, as you can see $x(1)$ and $x(5)$ and then the other one is $x(3)$ and $x(7)$. So, we will be having combining the thing. This is output $x(0)$ to $x(7)$.

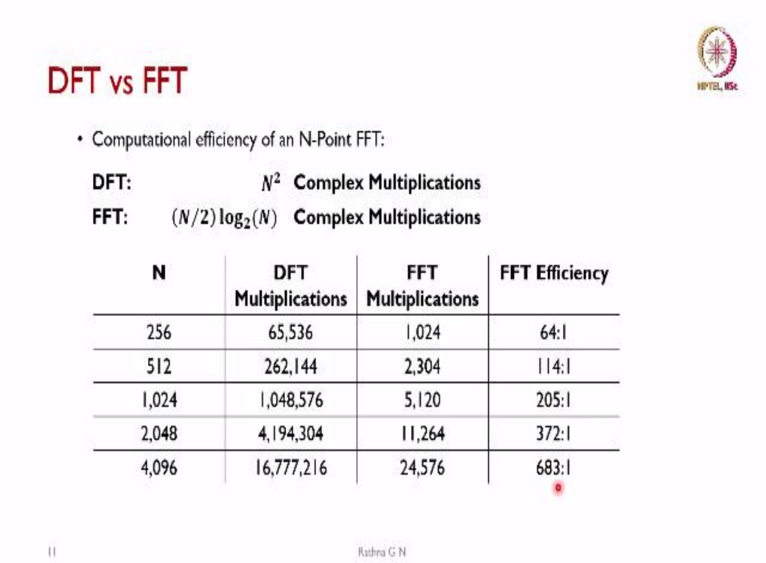
So, now what we will do is we have all inputs here. So, this will see $N/4$ point is nothing but only a single butterfly in this case, so we will replace this with a single butterfly and then put the weights here so we know that $W_N^0 = 1$. So, we can avoid the multiplication here and then we have -1 basically, that is how did we get the -1 here? So, you can you had to go back and then look into the thing.

So, that is W_8^4, which is equal to $-W_8^0$ which is nothing but -1. So, you will be seeing all of them will have the butterfly at the bottom -1 here. What we have is W_N^k here $W_N^{}$ this is $\frac{N}{2}$what we have the thing? So, when we this needs $N \log_2 N$ complex multiplications and complex additions, when I have the twiddle factors sets in this way, that is we are representing m minus first stage in this manner.

So, now, after splitting the thing that is ${W_N^{r+N/2}}$ is nothing but ${W_N^{N/2}}W_N^r$ so, which is nothing $-W_N^r$. So, I can provide what is it, so, this is equivalent to -1 now, so this is -1 here, and I can shift my W_N^k to here in the input of one of the input, then what happens? So, computation is going to reduce to half, so then we will be needing $N/2\log_2 N$ complex multiplications and $N\log_2 N$ complex additions.

So, this shows for N = 8 the complete butterfly structures in the second stage, you will be seeing that, we need 2 coefficients because the other one I have taken it as 1 = 1 and here also what you will be needing its 2 coefficients, whereas in the last stage, what we have is 4 coefficients what is represented but W_8^0 is going to be 1. So, we need 3 coefficients in this case, that way some of the multiplication complex multiplications, you can avoid. So, we will see that the computation complexity in a while.

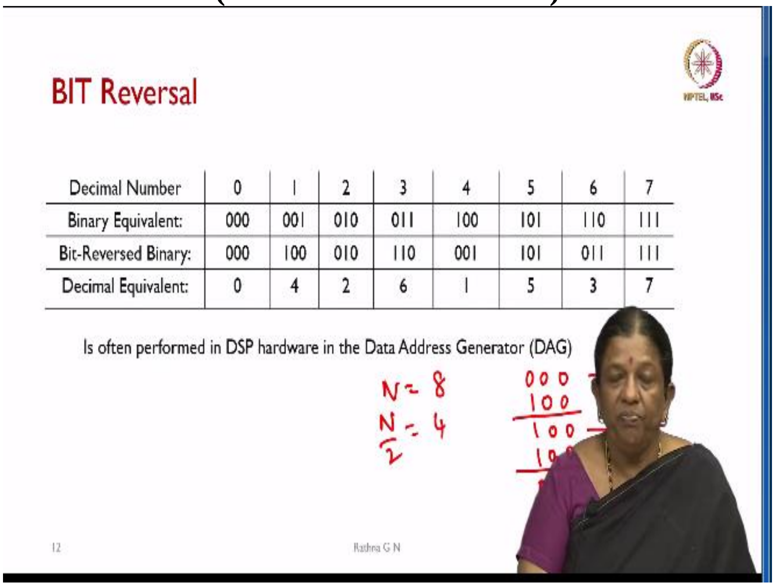
(Refer Slide Time: 22:13)



So, now, what we have compared when we discussed about DFT versus FFT. Now, after doing the FFT, it is splitting of it, so, we will be seeing that how we have reduced it? So, we know that we need in DFT N square complex multiplications and whereas, in FFT with the shifting twiddle factor to the input stage, so, we have achieved $(N/2)\log_2(N)$ complex multiplications. So, with respect to that, we know that when $\boldsymbol{N^2}$.

So, what will be the computations, number of multiplications required by DFT and number of multiplications required by FFT. And only with respect to this we will be comparing efficiency. So, it goes from 64 is to 1 to 683 as the power of the computation we have to do increases by compared to base 2, 2^N. So, as it is increasing, so, efficiency is also going to increase.

(Refer Slide Time: 23:24)

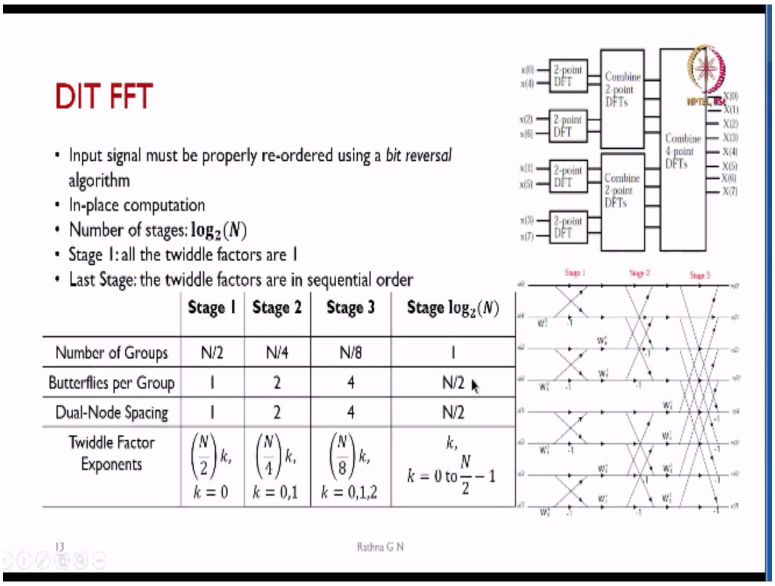


So, now, one of the thing what we have seen is, when we are doing this bifurcation, what has happened to here it is going to show us clearly, so, what I have is first is $x(0)$, next one what I want is $x(4)$ next I want to $x(2)$ and then $x(6)$. So, we have done the bit reversal in this case, so, as you will be seeing in the equation here. So, what we want is decimal numbers 0 to 7 what we have it when N = 8 binary equivalent is as you will be seeing 00 to 111.

But, input in the order what we want is 0 next is 4 what we want it, which is equal to 100 and then 010 and the next is 110 which is equivalent to 6 so on in the odd side part. So, as we said that in DSP hardware, we use the data address generator to do that, I think if you have to recall your architecture, I had given how to do this addition in hardware, as we know N is equal to in this case 8 so N/2 = 4. So, we know that the first one we are starting with 00 then we add 100. So that is from left to right, what will be adding it?

So, which gives me first digit is 0, next one is 4. Now I add 100. So, I said, we will be adding it in this way, 1 + 1 is 0 will be carrying 1 to here 010, which will give me 2. So, you can go on doing that till the end. So, this is the data address generator, which will be incorporating bit reversal as the input to FFT computation.

(Refer Slide Time: 25:33)



So, now we will see the, how complex it is how many groups we have, and then how many butterflies per group what we need it? So, that is we said input signal must be properly reordered using a bit reversal. That is what, what we want it in the input. So, you can say that this is a 2 point DFT, which we can represent it as just like butterfly when it is a 2 stage point DFT. So, the next one is combined these 2 points DFT is from each stage, we will be having the next stage, and then combine these 2 point DFTs.

And then finally, we will be combining 4 points here from each stage to get my 8 bit output. So, when we represent N = 8 as you can see the thing this is the first stage, this is the second stage what we have it, this is the third stage. So, now we call this thing as, if we want to have we can do in place computation. What do I mean by in place computation? That means these are the inputs, I can over write with these values with respect to this.

Because I do not need these values in the later stages that is why we call it as in place computation, so that I need not how to store my input more than 8 bits. And then next is how many stages we have to do? We say N = 8 we have seen that we need 3 stages that is $\log_2(N)$. So, which is going to be 3 and stage 1 so we know that all the twiddle factors are 1, all are W_8^0, so which is one in this case and last stage, that twiddle factors are in sequential order.

As we can see that here it is W_8^0 and W_8^2 what we need it. Here, what we need is W_8^0 W_8^1 W_8^2 and then W_8^3, that is what the twiddle factors in the last stage is going to be in order. Now we will see that some of the parameters with respect to butterfly structures, number of groups, how many I am going to have it? So here, you can see that we call each one as a group. So, each butterfly basically, so in the first stage, we have $N/2$, as we have taken, N = 8, $\frac{8}{2} = 4$. So, you will be seeing 1, 2, 3, 4 groups are there in this case.

Now in the next stage, I am going to have $N/4$ which is nothing but 2, in this case, N = 8, you will be seeing this is 1 group, this is the other group. And in the last stage, we will have 1 group, so $8/8 = 1$. So, this is you are having only 1 group in the stage 3 in this case, now we will see how many butterflies per group are going to be there? In the first stage, we have 1 butterfly. So, in all the groups we have only 1 butterfly.

Coming to stage 2 so, as you can see these are the 2 groups, I have 2 butterflies in this group, and here also we have 2 butterflies, when I come to the next stage, stage 3, then we are going to have and the how many butterflies 4 butterflies in the group here. So, you will be seeing that this is 1 butterfly, second, third and then fourth butterfly. So, that is what it is going to be and number of stages or more, then what we will say in the stage $\log_2(N)$ will have $N/2$ butterflies.

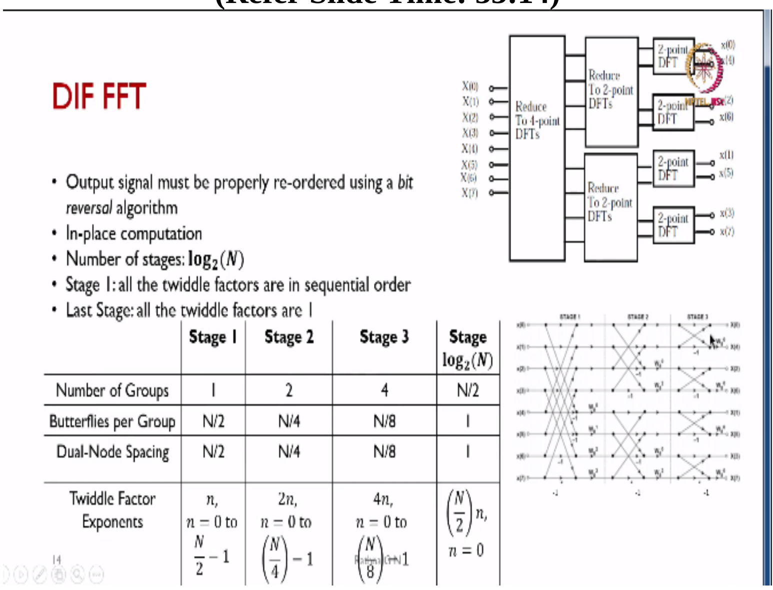
So, something should be striking if it is N = 16 then it is going to be 8 butterflies in the last stage. And then what is it? We call it as dual node spacing. So, between the 2 nodes, what is the spacing, I am going to have it here 1, because $x(0)$ and $x(4)$ these are the 2 nodes, the spacing is only 1 between them. When I come to stage 2, we will be seeing that the spacing is going to be 2 nodes. So, I am taking one from here, next one is coming from this. So, there will be 2 node differences what I will be giving it as input this?

Same way you can calculate everywhere and in the stage 3 I will be having this thing dual node spacing is going to be 4 I think it should be something striking equivalent to your butterflies per group and then node spacing is also same these 2 columns almost repeat it. So, you will be seeing that this is the first input the next input from the fourth one. So, if I take in this one 2 3 0 1 2 3 and then the fourth one, from here, what I will be taking it so there will be spacing of 4 nodes for this.

Now comes the twiddle factor exponents, how we are going to put it in each node? So, we will be having $\frac{N}{2} + k$ in this case in the stage 1k will be equal to 0. So that means to say I will be having 8 W_N^k is which are all 1. So, that is the reason why we do not have any we can remove this $W_8^0 = 1$. So, 8 of them are there. So which is equivalent to 1 in this case. So, the next one in W_N^k what we have put it $k = 0$ that is why it becomes W_8^0 and which is equivalent to 1.

So, in the second stage, what we are going to have is $\left(\frac{N}{4}\right)k$, k will be going between 0 and 1. So, you will be seeing that this is W_8^0 and then k will be 1, then it is going to be W_8^0 into 2 in this case. So, the last stage, it will have stage 3 in this case is going to be last stage N/2 into k 0, 1 and then 2 are the values so, W_8^0, W_8^1, W_8^2 and W_8^3. So, we will be having 4 of them here and when we have more than that, what will be the thing k will be going between 0 to N/2 - 1. So, you will be seeing here N/2 is 4, 4 - 1 is 3. So, that is why we have k = 0, 1, 2 when N = 8.

(Refer Slide Time: 33:14)



So, coming with this thing, I as we said we can have the computation either in the time domain or in the frequency domain. So, that is we call it as decimation in frequency FFT. So, we say that output signal must be properly reordered using a bit reversal algorithm that means to say input is in order and output will be in the bit reversal order in the DIF case. So, here also it is in place computation compared with your DIT decimation in time and here also number of stages is going to be $\log_2(N)$.

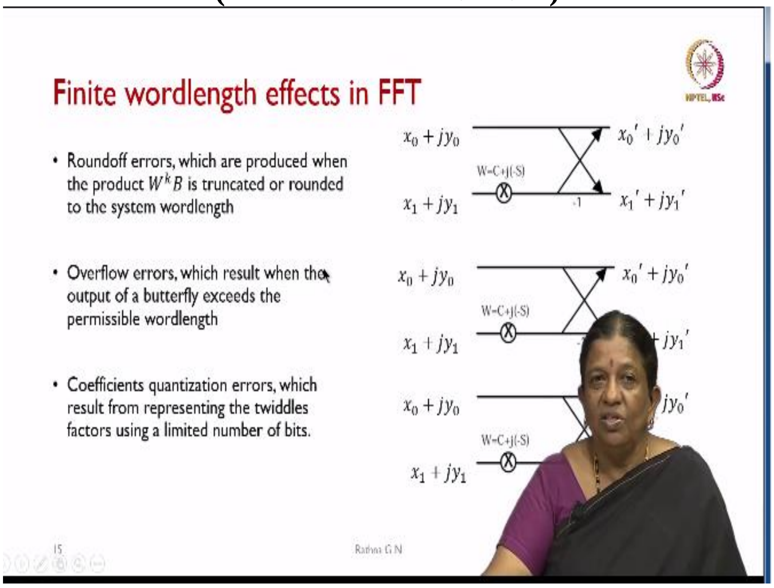
And what we say stage 1 all that twiddle factors are in sequential order and last stage all the twiddle factors are going to be 1 in this case. So, you will be seeing that in decimation in frequency is equivalent to reverse of your decimation in time. So, how are the number of stages and other things, it looks like you will be seeing it, number of groups in this case is going to be

1 for N = 8, but it is shown and then number of groups in stage 2 is going to be 2 and then a last stage in this case is 4 so, it is equivalent to $\log_2(N)$ stages.

The stage thing will have N/2 number of groups and then you will be seeing the butterflies per group will be N/2, N/4 and N/2 and last one, we will have 1 basically per group and then even dual node is equivalent to the previous case basically, this is the one. Now, only the thing is twiddle factor exponents. So, you will be seeing that n in this case there it was a k was the variable here it is going to be n, n will be going between 0 to N/2 – 1 will be your twiddle factor first stage.

And in the second stage it is 0 to N/4 - 1 and so on what you will have it in the last stage you will be having n = 0 and N/2 twiddle factors, which becomes equivalent to 1 in this case also as you can see it.

(Refer Slide Time: 35:38)



So, now we will look at the finite wordlength effects in FFT. Thank you. So, we have seen how to get FFT using butterfly. So, in the next class, we will look at the finite wordlength effects in FFT, how, it will be effecting our output and what is curve we have to take it. Thank you.