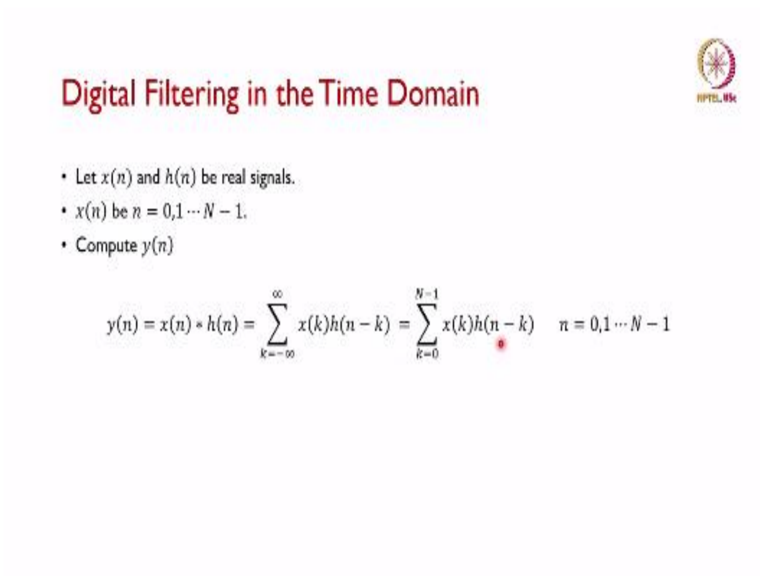
Real – Time Digital Signal Processing
Prof. Rathna G N
Department of Electrical Engineering
Indian Institute of Science - Bengaluru

Lecture – 22
Complexity of Filtering and the FFT

Welcome back to real time digital signal processing course. So, last class we discussed about the Discrete Fourier Transform. In this today's class, we will see the complexity of filtering, and then how FFT is going to be derived.

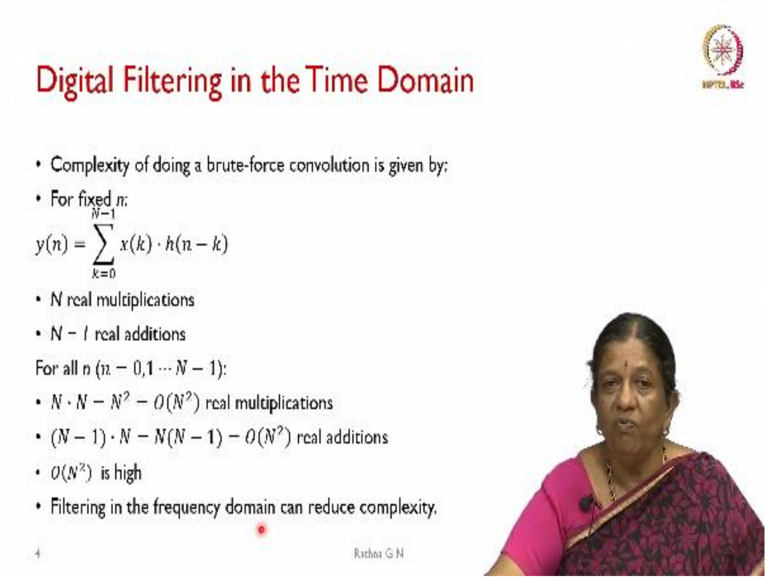
(Refer Slide Time: 00:40)



So, how we represent our digital filtering, we know that equation for our FIR filter for simplicity what we will take it here. So, $x(n)$ is our input and $h(n)$ be real signals that is 2 signals, if we want to take it or the impulse response of the filter, what we will consider it as $h(n)$. So, if we assume the coefficients, it becomes $v(n)$in FIR filter, so $x(n)$ is we representing it as 0 to $N-1$, then we have to compute our $y(n)$ using the as we can see here, the star indicates the linear convolution.

So, how we can represent k will be varying between -∞ to ∞, $x(k) \cdot h(n-k)$ or we can represent k = 0 to $N-1$, because this is what our length of sequence input sequence, then what we will show is $x(k) \cdot h(n-k)$ or we can as you know that, since this is a LTI system, so we can have $x(n-k)$ for my x and then $h(k)$ for my coefficients, what we have seen in our DSP implementation, so both the ways are correct. Here, we will be assuming at present to derive $x(k) \cdot h(n-k)$.

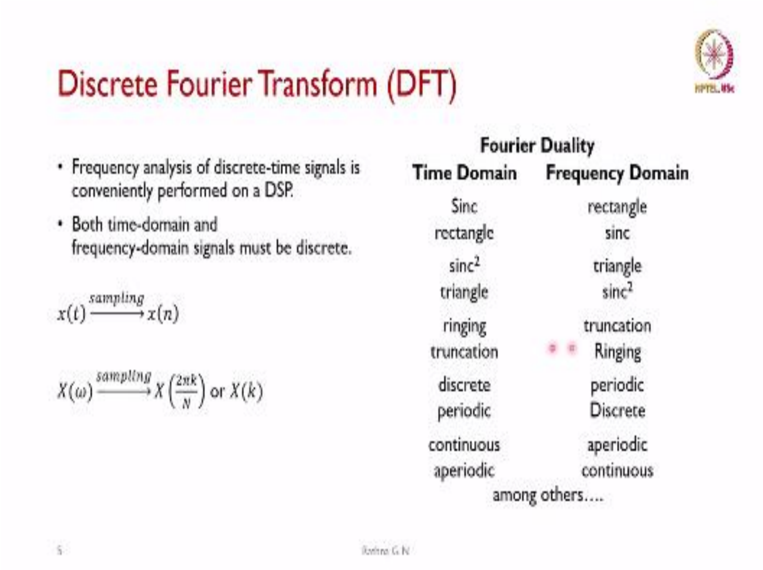
(Refer Slide Time: 02:18)



So, what is it? So, we have to say that what is the complexity of computation of this linear convolution, to make it simpler, instead of convolution, we will assume it as multiplication at present, k will be varying between 0 to *N* - *1* and then $x(k) \cdot h(n-k)$. So, how many real multiplications I am going to do it because it is 0 to *N* - *1* and *N* real multiplications. And with $x(k) \cdot h(n-k)$ and then we have to have the summation which is going to be 1 less we will be assuming that *N* - *1* real additions are required.

For all *n* varying between 0 to *N* - *1*, then if we assume that my *n* is also length of 0 to *N* - *1*, then total number of multiplications for filtering required is $N \cdot N = N^2$, we say order of *N* square real multiplications are required. And then addition we know $(N-1) \cdot N$ additions which is nothing but this also results in order of *N* square real additions. So, total number of multiplications and additions are required is order of N^2, which we consider it as very high. So, how we can do filtering in the frequency domain which can reduce or complexity what we will be looking in a few slides.

(Refer Slide Time: 03:55)

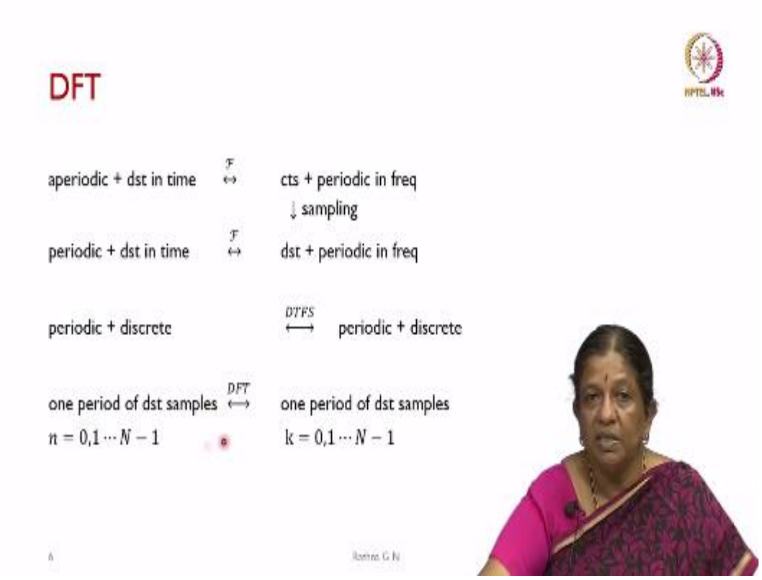


So, what is it? We will say discrete Fourier transform we say it is a frequency analysis of discrete time signal. It is how we can perform it on the DSP that is what we will be looking at. So, both time domain and frequency domain, the signals must be in the discrete format. So, $x(t)$ is our analog input. After sampling, I will be getting $x(n)$ in the digital domain, and $X(\omega)$ which is ω is continuous after sampling it, we will be getting $X\left(\frac{2\pi k}{N}\right)$, or we call it as $X(k)$. So, we will be representing $\frac{2\pi k}{N}$ as K in this case, that is K.

So, we will be seeing some of the duality of the Fourier domain thing one is in the time domain, what it is represented in the frequency domain, if it is a sine function in time domain, which is going to be rectangle in frequency domain. So, if it is rectangle in our distinct time domain, it becomes a sine function in our frequency domain. So, I think some of you would have heard sine square function, it becomes triangle and frequency domain vice versa.

And we will be seeing that it becomes a ringing in case of time domain, then it becomes a truncation in the frequency so, the other way around. So, if it is discrete in time domain, which becomes periodic in our frequency domain, and if the input signal is periodic, then it becomes discrete in our frequency domain. So, if it is continuous, it becomes aperiodic in frequency and if it is aperiodic in time domain it will be continuous and then you can have many more like this in the duality, what you can consider. A few of it which is required for our derivation what we will be using it here.

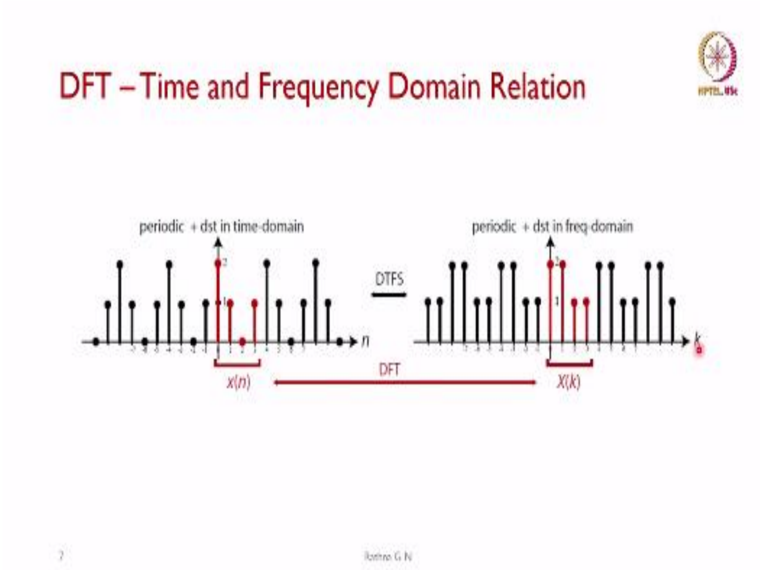
(Refer Slide Time: 05:59)



So, coming to DFT so, we know that, if the signal is aperiodic plus discrete in time domain, then if I take the Fourier transform of this, so that means to say if $x(n)$ is aperiodic and my $h(n)$ is discrete in time domain, what I will be getting the result is continuous time signal in Fourier domain plus periodic in frequency what I will get it So, by doing the sampling, I can reach this discrete signal what I can get plus periodic in frequency what I can achieve from this stage.

Or I can have a periodic signal in the time domain plus dst in this think discrete time signal here, then I can get discrete plus periodic in frequency when I do the Fourier transform, or if the signal is periodic plus discrete, then if I take the discrete time Fourier series, basically, if I convert it, then the output is going to be periodic plus discrete. So, we will be seeing that one period of our say discrete sample DFT, I will be getting one period of discrete samples here. So, here $n = 0,1 \cdots N - 1$, in this case, we will be having k will be varying between 0 to N - 1.

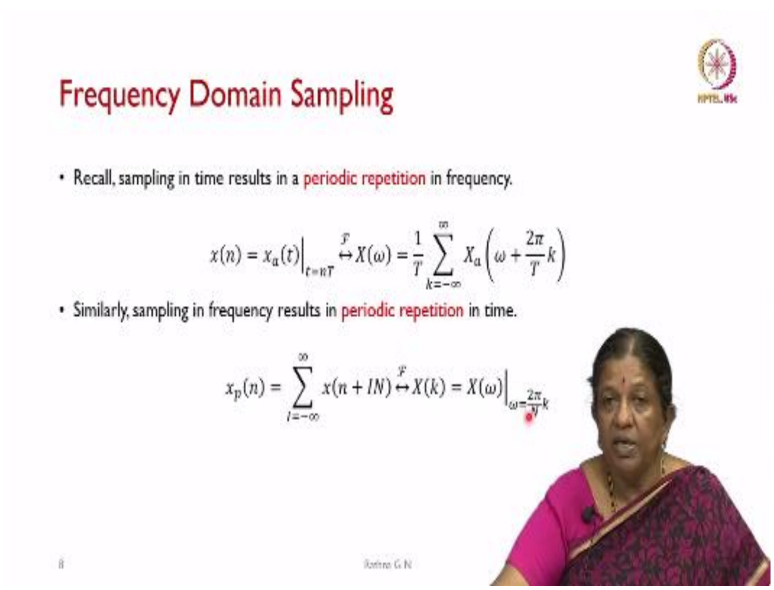
(Refer Slide Time: 07:34)



So, now, how to represent it both in time and frequency domain relation what it is shown here. So, you will be seeing that this is a periodic plus discrete time sample what it is taken in the time domain. So, you are seeing that this is my $x(n)$and then, when I take the DFT so, I will be getting $X(K)$ here, if I take the discrete time Fourier series, then what you will be seeing is periodic plus discrete in this thing time domain.

So, this is how the samples have been represented in that, so, this will be x axis will be k and then our y axis will be representing periodic plus DST in frequency domain.

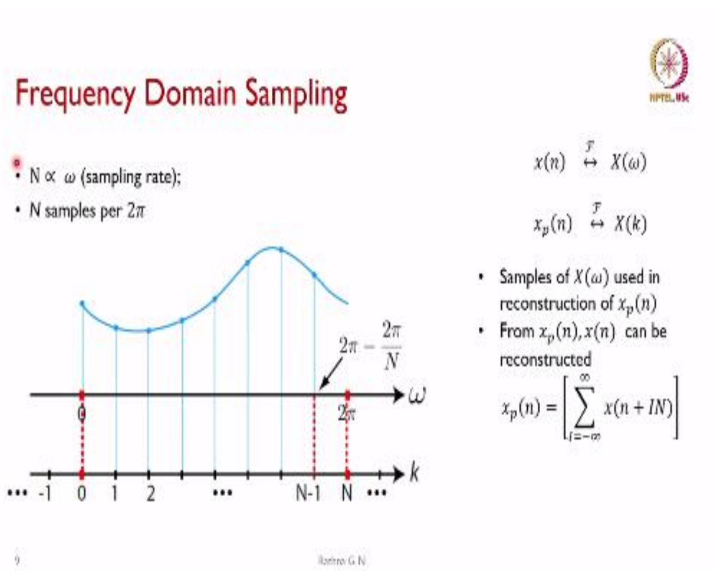
(Refer Slide Time: 08:23)



So, coming with the think how we are going to do the frequency domain sampling what we are going to achieve. So, that is we have already said sampling in time domain we said it is going to result in periodic repetition in frequency. So, when I represent $x(n)$, so, we say that it is a repetition of it $x_a(t)$ we represent $t = nT$, then take the Fourier transform we result in $X(\omega)$ which is nothing but $\frac{1}{T}\sum_{k=-\infty}^{\infty} X_a\left(\omega + \frac{2\pi}{T}k\right)$.

So, similarly, sampling in frequency results in periodic repetition in time what we will be seeing it this is the periodic signal what we are representing as $x_p(n)$ in the time domain, then it is going to result $I = -\infty$ to ∞, $x(n + IN)$ and then I will be taking the Fourier transform which is equivalent into $X(k)$, which is nothing but $X(\omega)|_{\omega=\frac{2\pi}{N}k}$.

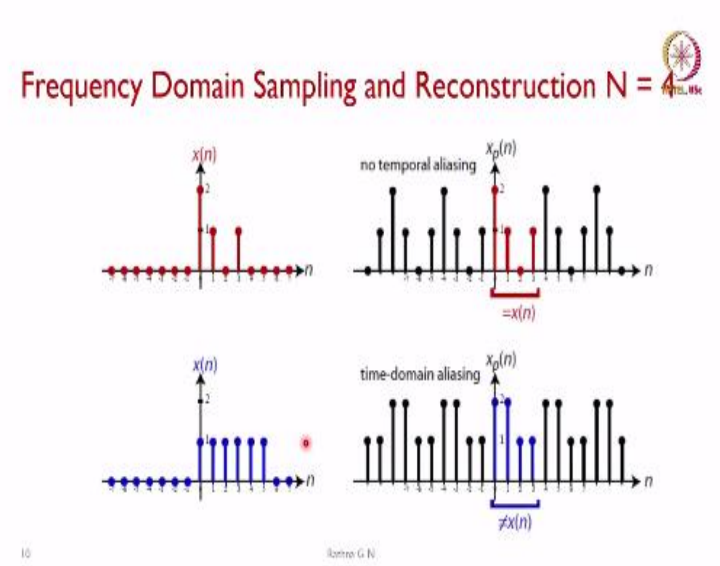
(Refer Slide Time: 09:48)



So, now we will see the thing how the sampling is going to happen. So, we know that N $\propto$ ω that is sampling rate. So, that is N samples per 2π, what we are going to have it in this region is 0 to 2 π, we will have N samples. So, we know that $x(n)$ when we take the Fourier transform, we will be getting $X(\omega)$. So, the periodic representation $x_p(n)$ if we take it Fourier transform, we will be getting $X(k)$ in this case. So, we are seeing that is samples of $X(\omega)$ used in the reconstruction of $x_p(n)$.

And from $x_p(n)$, we can get $x(n)$ can be reconstructed in this fashion. So, we will be seeing that this is what we are varying 0 to N - 1. Next one will be the N sample k is going here. And this is our omega x axis. So, here it is, you are seeing when $N = 2\pi$, this is the axis and then N - 1 point will be 2π- 2π/N. So, this is the resolution what we will be calling depending on our N samples.

(Refer Slide Time: 11:13)



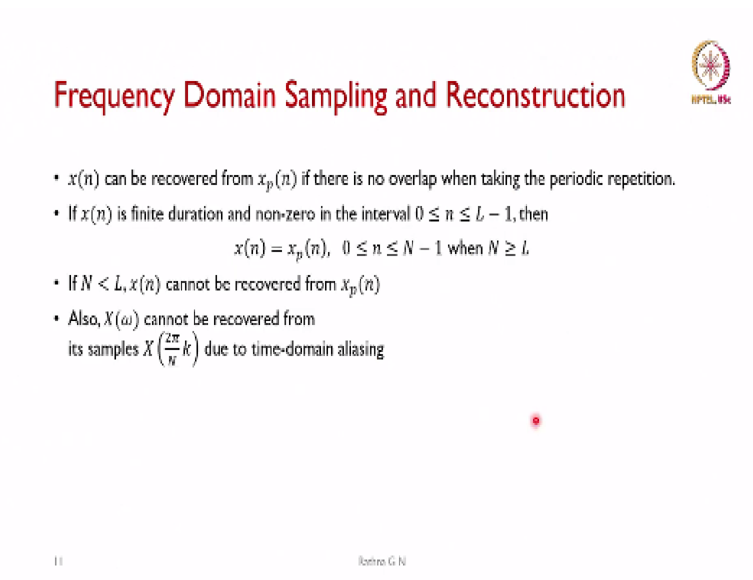
So, coming to the how we can do the reconstruction, because we have gone from time domain to frequency domain, but my output I want it in the time domain. So, how we can do the reconstruction? So, this is shown with $N = 4$ in this case. So, you can see that this is my $x(n)$, which has 4 samples, the magnitude, you will be seeing that this is 0101, what you have taken rest of them are zeros, both the sides negative and then positive side, this is my n basically.

And then we will see that we are assuming because I have N = 4. So, we do not have any temporal aliasing in this case. So, this is my $x(n)$ basically, this is the periodicity what I have taken. So, you will be seeing that the magnitude sorry, at x is 0 it is 2 and then $x = 1$ it is 1 and then $x = 2$ it is 0 and then this is 1, so you will be replicating them on both the sides as you can see it. So, these 4 samples, you will be seeing that it becomes 2 1 0 1 2 again, so on the negative axis also what you will be doing the repetition, so we do not have any aliasing.

Because they are distinct in nature. So, you will be seeing that x of n is a unit step response, all of them are 1 then what happens in that time domain. So, we said that if it is a rectangular window, it should result in your sine function. So, here you can see that what is the happening is in the periodic if I take the thing, then you will be seeing that these 2 are 1s and then that is magnitude is 2 and then 1 and then which is not equivalent to my $x(n)$. So, the sampling rate, if you have $N = 4$ you have kept it.

But x of n is what you have given more than n so I would not be able to reconstruct the signal. So, you are seeing that aliasing happening instead of 1 magnitude of what I have supposed to get it. So, 2 of the samples which have got alias and then which has gone to pick magnitude 2 in this case.

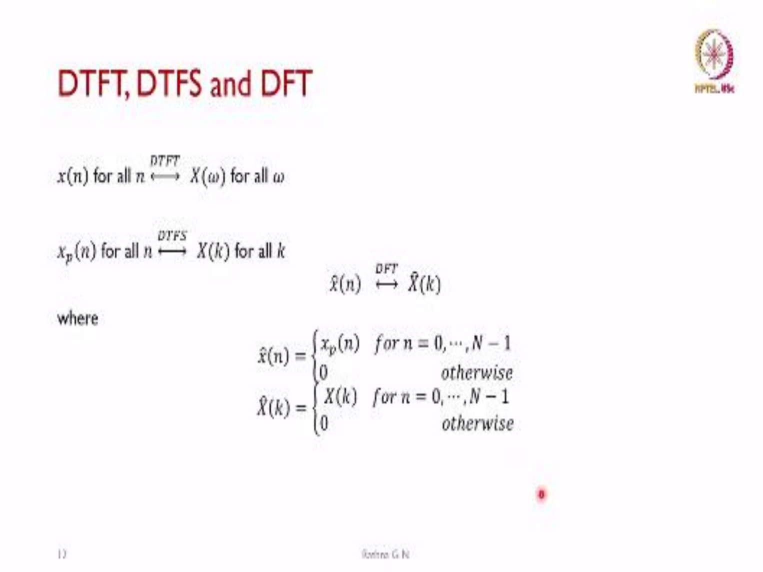
(Refer Slide Time: 13:47)



So, what is sampling and then reconstruction how they are related in the frequency domain, we will see it so $x(n)$ can be recovered from our periodic $x_p(n)$, if there is no overlap when taking the periodic repetition. So, if $x(n)$ is finite duration and then non zero in the interval 0 to $L - 1$, then what we say is $x(n) = x_p(n)$ in this domain 0 to $N - 1$ when N is greater than or equal to L. So, if it is less than then we know that aliasing is going to happen for the periodic signal.

So, if N is less than L, that is what it says cannot be recovered from our periodic $x_p(n)$. Also $X(\omega)$ cannot be recovered from its samples that is $X\left(\frac{2\pi}{N}k\right)$ due to time domain aliasing what has happened, as the previous slide shows that, what was the initial thing but here all of them are 1. So, we are not getting back in this case.

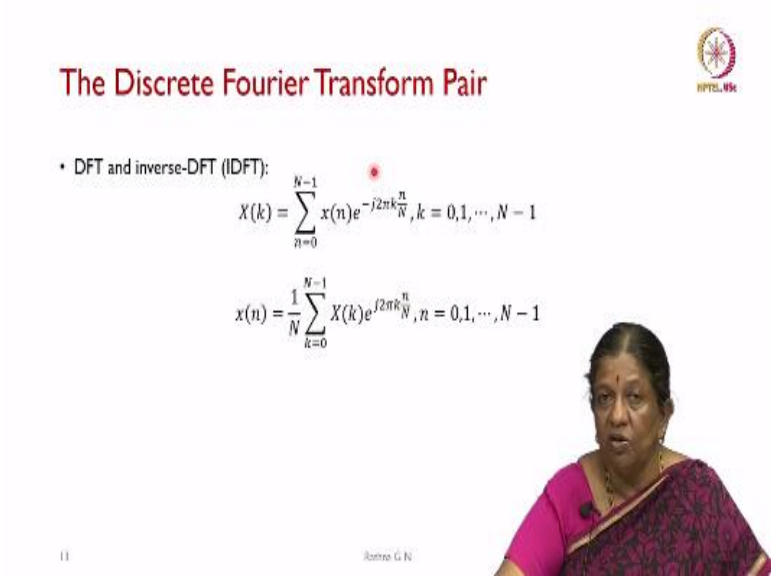
(Refer Slide Time: 15:05)



So, now, what is the relation between our DTFT, DTFS and then DFT. So, $x(n)$ for all n if I take the DTFT then it becomes $X(\omega)$ for all omega and then the periodic signal in the time domain $x_p(n)$ for all n and if you take a discrete time Fourier series, then it is going to become $x(k)$ for all k. So, we say that $\hat{x}(n)$ that is periodic DFT if I take it results in the periodic $X(k)$. So, here $x(n)$ periodic what we are represented instead of $x_p(n)$ so, for $n = 0$ to $N - 1$ it will be 0 otherwise.

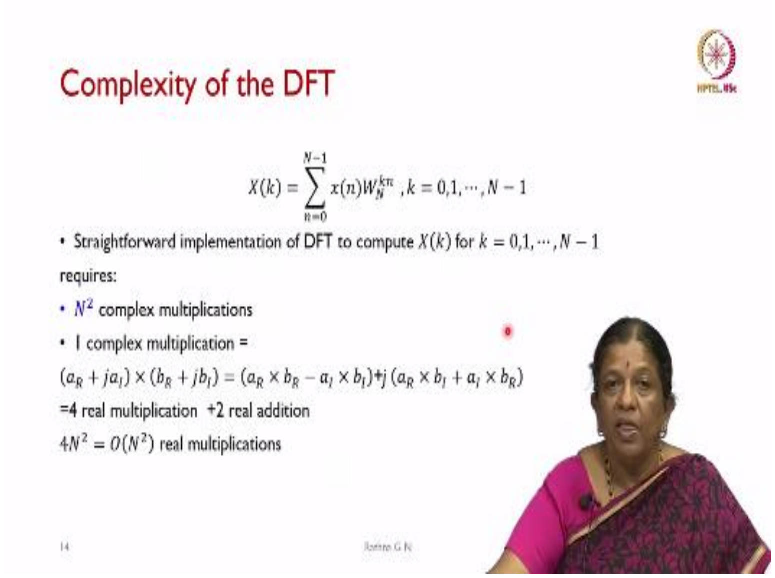
And then what happens in the frequency domain $\hat{X}(k)$ will be $X(k)$ in this 0 to $N - 1$ other thing is 0.

(Refer Slide Time: 16:06)



So, what we have seen the DTFT pair. So, now, we will see the DFT pair. So, that is we have taken the example and we took the IDFT so, we call them as pair. So, this is the analysis function and this is the synthesis what we call it that is $X(k)$ is given by $n = 0$ to $N - 1$ $x(n)e^{-j2\pi k\frac{n}{N}}$ whatever way you are represent it k will be varying between 0 to N – 1. The N was what we have seen it as x of n is nothing but 1 by N. So, $k = 0$ to $N - 1$ $X(k)e^{j2\pi k\frac{n}{N}}$, instead of negative here it becomes positive. In this case n will be varying between 0 to $N - 1$.

(Refer Slide Time: 17:01)



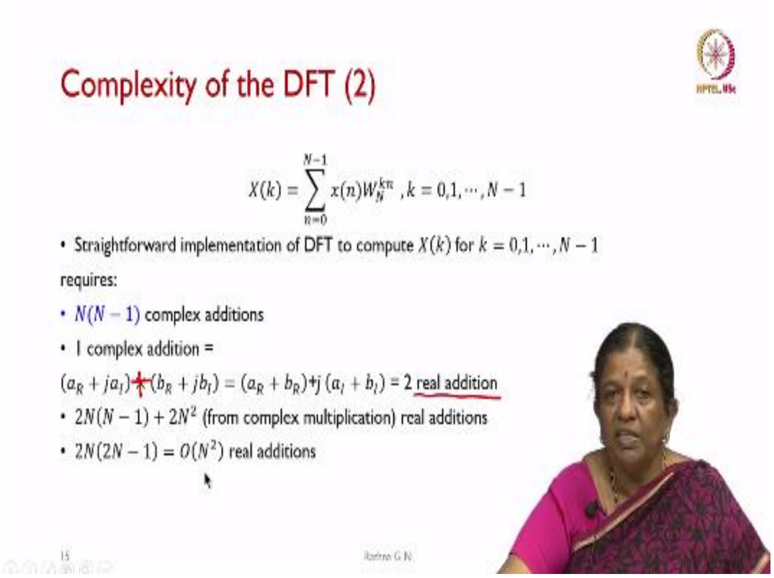
- So, what is the complexity of DFT what we have seen the thing. So, with respect to our signal processing hardware, we will see what is the complexity of our DFT, complexity of filter what we have seen the thing we have to see the complexity of our the discrete Fourier transform $n = 0$ to N - 1. So, we know that $x(n)W_N^{kn}$ will be varying between 0 to $N - 1$. So, we know that a straightforward implementation of DFT to compute $X(k)$ for $k = 0,1,\cdots,N-1$ requires just like our filter order of N square here it is not real multiplications.

It is going to be complex multiplications because $x(n)$ has to be multiplied with cos and $x(n)$ has to be separately multiplied with our sine with his imaginary part and they have to be kept separately. So, that is why he needs complex multiplications in this case, and then we need one complex multiplication is equivalent to what we call it as $(a_R + ja_I) \times (b_R + jb_I)$, if I do it, this is what, what I am doing. So, both $x(n)$ and then my coefficients are we call it as complex $x(n)$ is also complex, we have assumed.

When we do this multiplication, you will be seeing that it is nothing but $(a_R \times b_R - a_I \times b_I)$+$j$ $(a_R \times b_I + a_I \times b_R)$. So, what does it show? 1 2 3 and then 4 real multiplications so, one complex multiplication is equivalent to 4 multiplication plus, I had to have even we take it as subtraction or addition, because we do in 2's complement if you call back your number system, so, we have 2 additions, real additions.

So, what happens to our total computation time which is equivalent into $4N^2$, which is nothing but $O(N^2)$ real multiplications what we need it, so, which is same as that of our filtering.

(Refer Slide Time: 19:29)

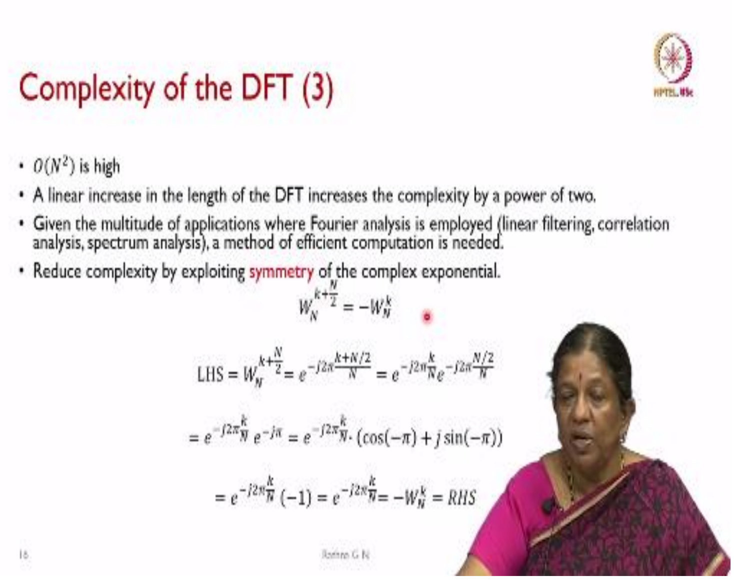


So, we will see that how we can reduce or take that forward. So, in this case, we said only multiplication we covered the thing. Now, we have to see from the addition point of view. So, what is it our filtering was needing $N(N-1)$ real time additions, but in this case it is going to be $N(N-1)$ complex additions. How do we represent one complex addition? That is $(a_R + ja_I)$ this is what we say is plus $(b_R + jb_I)$. So, this is my equation, $(a_R + ja_I) + (b_R + jb_I)$. So, it will be $(a_R + b_R)$+j $(a_I + b_I)$. So, that is how we will be resulting in 2 real additions.

And then when I want to do the total complex multiplication, and then we take that $2N(N-1)$, so we will be deriving it once again, when we take up the problem. So, it will be coming it $2N(N-1) + 2N^2$, that is, which is arriving from my complex multiplication. So, some of the additions taken from there so, this many number of what I need is real additions. So, what does it mean? So, this is the complex addition. So, 2 times because each complex multiplication we are taking it as 2 real additions.

So, which results in $2N(N-1)$ from this stage and then we know that from the complex multiplication, we have addition which is resulting so, which comes to $2N^2$. So, that is what, what it is rendered from complex multiplication, I have to take this also into account. So, this is what our total number of additions required for computing my DFT. So, in that case, the maximum is $2N(N-1)$ if I take it if we observe this also inside. So, additions also what we need is $O(N^2)$, that is the complexity of DFT both real multiplication and real additions are $O(N^2)$.

(Refer Slide Time: 21:59)

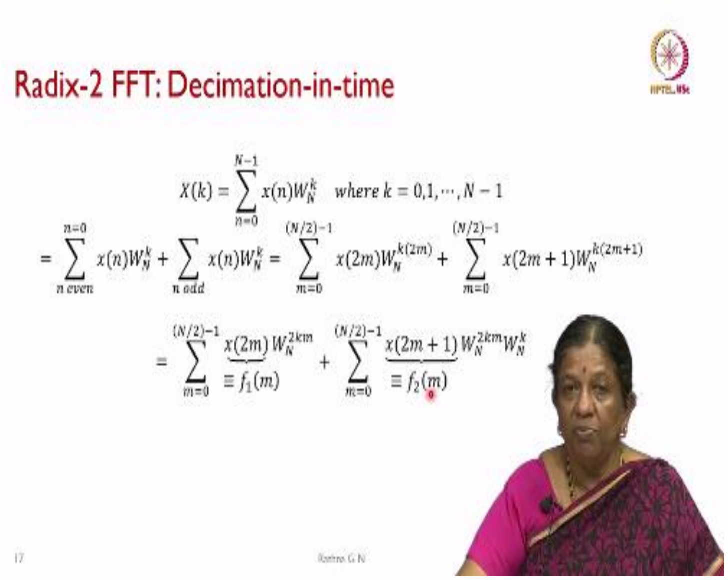


So, we know that already we have pointed out in the filtering, it is too high. So, linear increase in the length of the DFT increases the complexity by a power of 2 basically. And if you are given them this thing, what is it? The multitude of applications that is number of them where Fourier analysis is employed. That is, we will call it as linear filtering or correlation analysis, which will be taking it up little later, and then do the spectral analysis, what are the intention of our applications, then, we will say that how many this thing what is the complexity of it.

That is efficient computation is required. So, that is, reduce the complexity by exploring the symmetry property of our complex exponential. So, how we can do that? We have that twiddle factor, we saw that $W_N^{k+\frac{N}{2}} = -W_N^k$. So, we will see the left hand side is nothing but $= W_N^{k+\frac{N}{2}} =$ $e^{-j2\pi}$ substituting k with $\frac{k+N/2}{N}$ which is nothing but $e^{-j2\pi\frac{k}{N}}e^{-j2\pi\frac{N/2}{N}}$. So, this results in -1 that is what, what is shown here $e^{-j2\pi\frac{k}{N}}e^{-j\pi}$.

Which is nothing but this $(\cos(-\pi) + j\sin(-\pi))$ when we expand $= e^{-j2\pi}$ and then we know that $\cos(-\pi)$ is -1 and $\sin(\pi)$ is going to be 0, π and $-\pi$ is 0 so, which will be - 1. So, when we represent this is nothing but $-W_N^k$. So, that is how the derivation between the 2 has been arrived at so, LHS = RHS. So, whatever we are representing that is symmetry property is true.

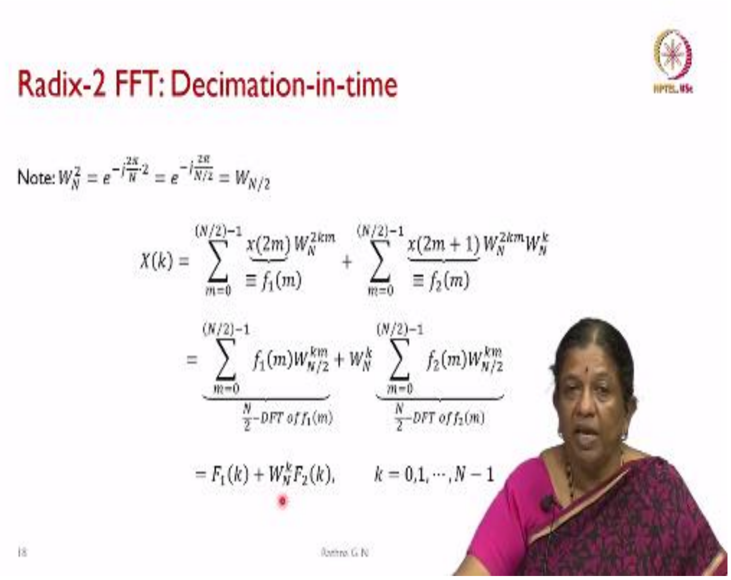
(Refer Slide Time: 24:30)



So, now, how we can extend this usually we call it as decimation in time basically, that is $X(k) = \sum_{n=0}^{N-1} x(n)W_N^k$ and this is our normal DFT equation k will be varying between 0 to $N - 1$. So, now, what happens? So, if n is even then $n = 0$ what will be taking it $x(n)W_N^k$ plus that is we are considering even and then odd parts next n is odd $x(n)W_N^k$. So, here it is going to be, we are changing the little bit of notation from n actually $\sum_{m=0}^{(N/2)-1} x(2m)W_N^{k(2m)}$ because it is a even and $W_N^{k(2m)}$, n is substituted with m in this case.

And this is our odd part $\sum_{m=0}^{(N/2)-1} x(2m+1)W_N^{k(2m+1)}$. So, this is how we can split our DFT equation into even and then odd parts. So, when you can see that we represent $x(2m)$ as $f_1(m)$ and we are seeing that this has become already W_N^{2km} and then this we call $x(2m+1)$ as $f_2(m)$ equivalent to and then this is $W_N^{2km}W_N^k$.

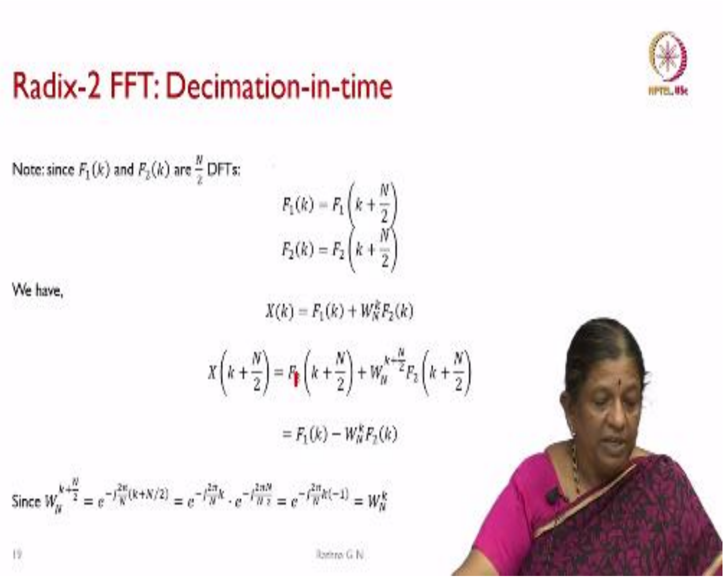
(Refer Slide Time: 26:16)



So, we are going to split like this decimation in time and then we know that W_N^2 is nothing but by substituting $k = e^{-j\frac{2\pi}{N}\cdot 2}$. So, which is nothing but $e^{-j\frac{2\pi}{N/2}}$ what I can take it so, the twiddle factor it is going to be $W_{N/2}$. So, then we will represent $X(k) =$ what we have is $f_1(m)$ and then $f_2(m)$. So, by putting substituting this $f_1(m)$ and then we are substituting our W_N^2 actually with respect to this, it becomes $W_{N/2}$ into km plus we know that W_N^k is independent of N.

So, which we will be taking it out W_N^k and this is our $\sum_{m=0}^{(N/2)-1} f_2(m) W_{N/2}^{km}$. So, you will be seeing that this is $\frac{N}{2} - DFT\ of f_1(m)$ that is what, what we have it then if we represent this as this is $\frac{N}{2} - DFT\ of f_2(m)$ if we substitute that as $F_1(k) + W_N^k F_2(k)$, k will be varying between 0 to $N - 1$.

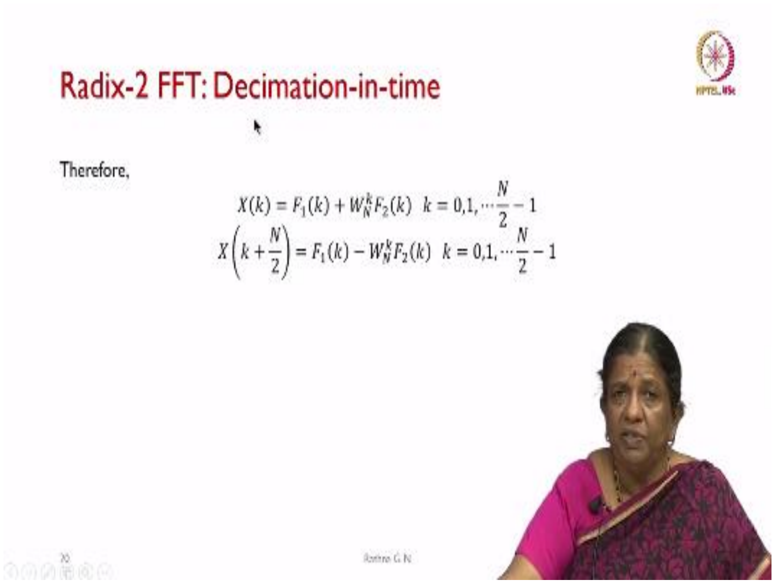
(Refer Slide Time: 27:49)



And then we know that $F_1(k)$ and $F_2(k)$ are $\frac{N}{2}$ DFTs. So, you will be seeing that $F_1\left(k+\frac{N}{2}\right)$ and $F_2\left(k+\frac{N}{2}\right)$ and $X(k)$ is going to be that is DFT of it will be $F_1(k) + W_N^k F_2(k)$. So, now, what we are going to represent that k also will be taking the symmetry property, we will do it as $k+\frac{N}{2}$. So, we can have it as $F_1\left(k+\frac{N}{2}\right) + W_N^{k+\frac{N}{2}} F_2\left(k+\frac{N}{2}\right)$. So, here it is $F_1\left(k+\frac{N}{2}\right) + W_N^{k+\frac{N}{2}} F_2\left(k+\frac{N}{2}\right)$, so, which is nothing but $F_1(k) - W_N^k F_2(k)$.

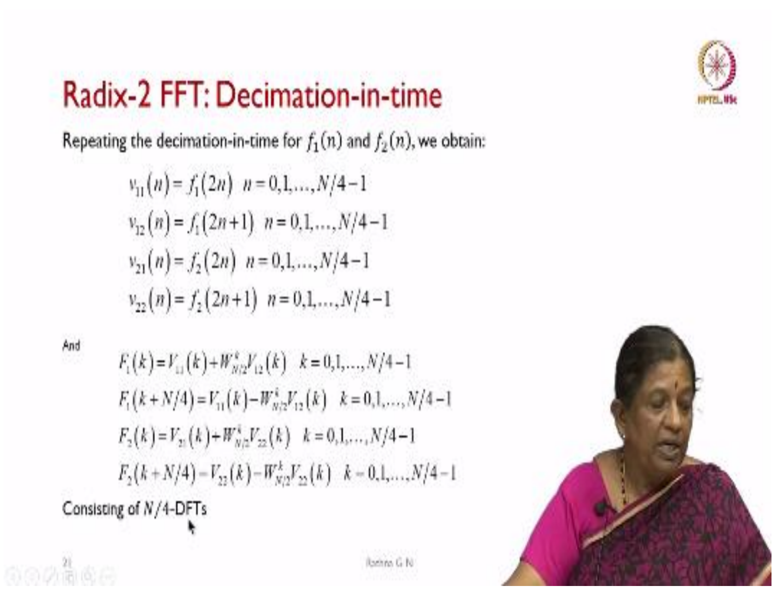
So, how this has come you will be seeing that $W_N^{k+\frac{N}{2}}$ what we have to solve the thing which is nothing but $e^{-j\frac{2\pi}{N}(k+N/2)}$ which you solve the thing. So, which is nothing but $e^{-j\frac{2\pi}{N}k(-1)}$, so, this is our W N k. So, you can see that $k+N/2$ results in W_N^k.

(Refer Slide Time: 29:38)



So, we can split this decimation in time further. So, that is $X(k)$ is nothing but $F_1(k) + W_N^k F_2(k)$, k also we will be going between 0 to 1 to $(N/2) - 1$ and then $X\left(k + \frac{N}{2}\right)$ is given by this equation as we have already computed $-W_N^k F_2(k)$ $k = 0,1, \cdots \frac{N}{2} - 1$. Now, this is what we call it as radix 2 FFT that is decimation in time what is happening step by step till we go up to last stages too.

(Refer Slide Time: 30:14)

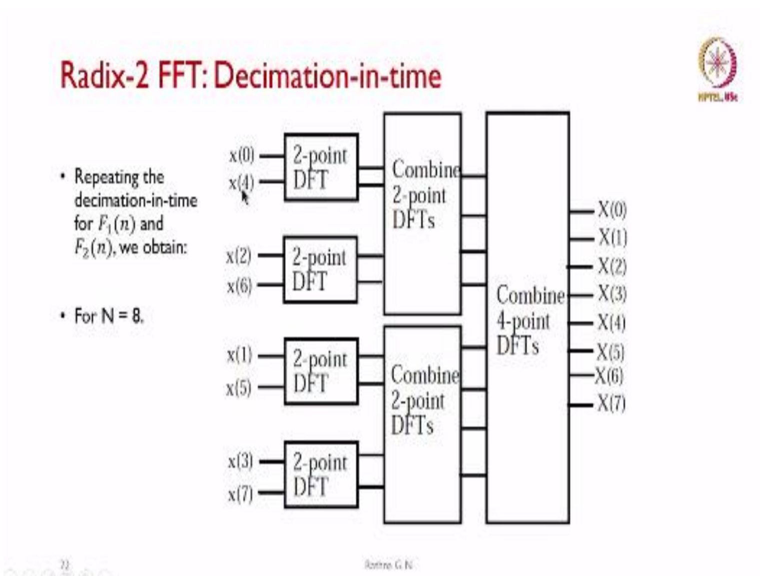


So, you will be seeing that repeating decimation in time $f_1(n)$ and $f_2(n)$. So, we will be obtaining $f_1(2n)$ and $f_1(2n+1)$, we call it as $v_{11}(n)$ and $v_{12}(n)$ so, you will be going $n = 0,1, \ldots, \frac{N}{4} - 1$ x stage and then $v_{21}(n)$ is nothing but $f_2(2n)$ and then $f_2(2n+1)$. So, which will be going by $\frac{N}{4} -$

1 when k is equal to divided by 2 further. So, that is how you will be continuously going on splitting the thing that is $F_1(k)$ is nothing but $V_{11}(k) + W_{\frac{N}{2}}^{k} V_{12}(k)$ so, which is going between this then $F_1\left(k + \frac{N}{4}\right)$ if you take the thing.

This is what, what you will be resulted and then that is the next F_2 what is split into this so, which has $N/4$ DFTs in this case.

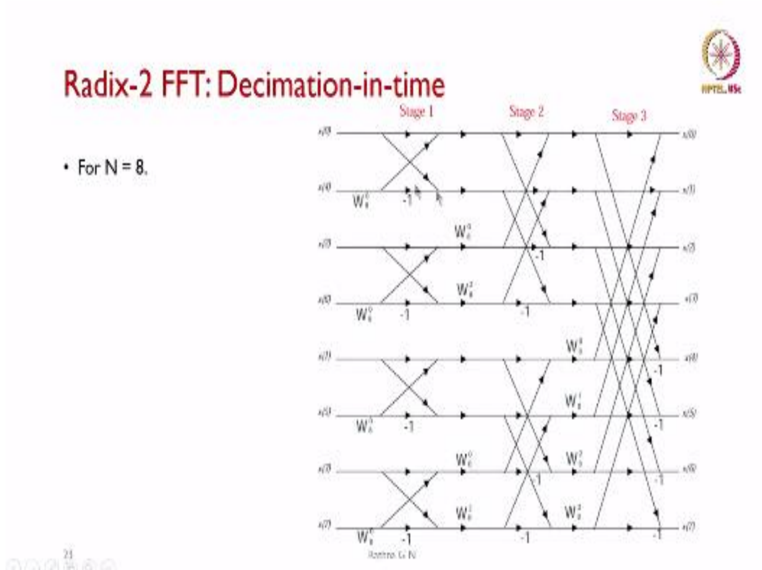
(Refer Slide Time: 31:24)



How we can represent in decimation in time, this is my for $N = 8$ it is a simple one to consider, we have considered it. So, the first one is $x(0)$, the next value is what we need is $x(4)$so, we will be doing the 2 point DFT and then $x(2)$ and $x(6)$ we will be doing that 2 point DFT which is going to be combined with 2 point DFTs here. And then the other 2 odd part what you can see it $x(1)$ and $x(5)$, $x(3)$ and $x(7)$ other 2 point DFT what you can do it and then combine them and then finally, you will be combining as a 4 point DFT.

So, you will be getting output as $X(0)$ to $X(7)$. So, you can see that input is bit reversed, what we have considered in the number system and DSP architecture we said we need input in the bit reverse format, output will be in in-order. So, we have seen the example how to generate the bit reversed also using hardware adder.

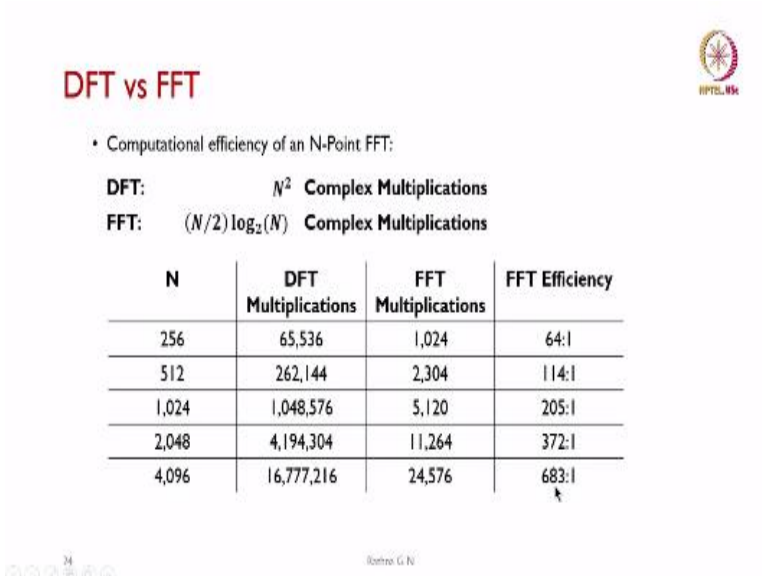
(Refer Slide Time: 32:42)



So, when we represent this as the, from the previous thing, in terms of what we call it as this is the butterfly structure, so this is my $x(0)$ and $x(4)$. So, this will be my weight W_8^0 , and this is my minus W_N^k what we have it here it is going to be -1, and then we will be combining these 2 and which goes to the stage 2 this is stage 1, this is stage 2, this is stage 3, where all the things are combined. Whereas in the second stage my twiddle factors, what do I need is W_8^0 and W_8^2. So, in both the cases, and we know that W_8^0 is 1.

I had to compute only this twiddle factors W_8^2. Whereas in the last stage, we need 3 twiddle factors, that is W_8^1, W_8^2 and W_8^3 has to be computed and then we as usual, $W_8^0 = 1$. So, we will be getting the output in order. So, to do FFT computation for $N = 8$, we know that $\mathbf{\log_2(N)}$ 8 is nothing but 3, we need 3 stages. So, we will be seeing that how we are going to reduce the thing computation?

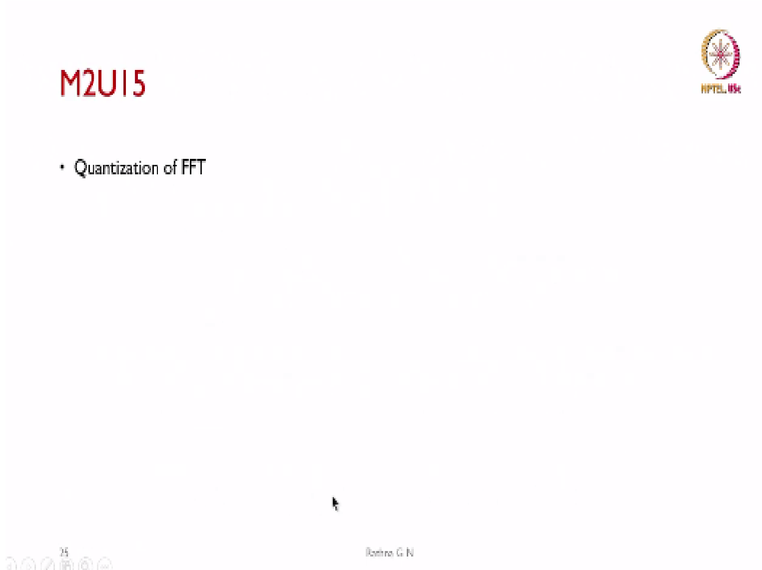
(Refer Slide Time: 34:08)



So, we will be seeing that with respect to this what we have given computational efficiency of N point FFT, we will see it, we know that it is $\boldsymbol{N^2}$ complex multiplication and addition what we needed and FFT, we need $(\boldsymbol{N/2})\,\mathbf{log_2}(\boldsymbol{N})$. So, we said that for 8 point it was 3 stages. So, if it is N point, then we need $\mathbf{log_2}(\boldsymbol{N})$and we can use a symmetry property it is going to be $(\boldsymbol{N/2})$ complex multiplications what we need it using FFT. So, you will be seeing some N points and then what is the DFT multiplication and FFT multiplications with respect to N.

So, if it is 256, we see that 65,536, here what we need is 1024. So, FFT efficiency, we compute it as 64 is to 1 for our DFT so as in when our N point increases, so you will be seeing that computation of DFT increasing very much. And then FFT you will be seeing low and you will be seeing that 683 is to 1 is the ratio for 4096.

(Refer Slide Time: 35:27)



So, this is how we do the computation of FFT. So, in the next class, we will be seeing quantization of FFT, how it is going to effect our word length and then even the coefficient has to be quantized. So, number of stages is going to increase. So, it is pipeline structure what we are going to have it. So, we will see what are the quantization effects in FFT in the next class. Thank you.