Signal Processing Techniques and its Applications
Dr. Shyamal Kumar Das Mandal
Advanced Technology Development Centre
Indian Institute of Technology, Kharagpur

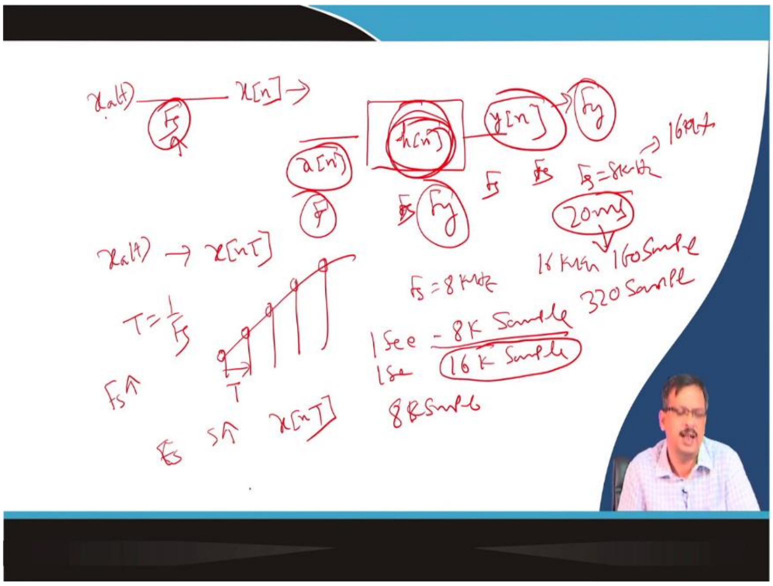
Lecture - 50
Instruction to Multirate Signal Processing

So, this week, we will talk about Multirate Signal Processing. So, the title of the talk of this week will be Multi-rate Signal Processing.

(Refer Slide Time: 00:24)

So, what do you mean by multirate signal processing? What is the meaning of multirate the word multirate? So, what is signal processing? When I say the multirate, what do you mean by the signal processing rate of the signal? What do you mean by that? So, let us talk about the digital signal processing.

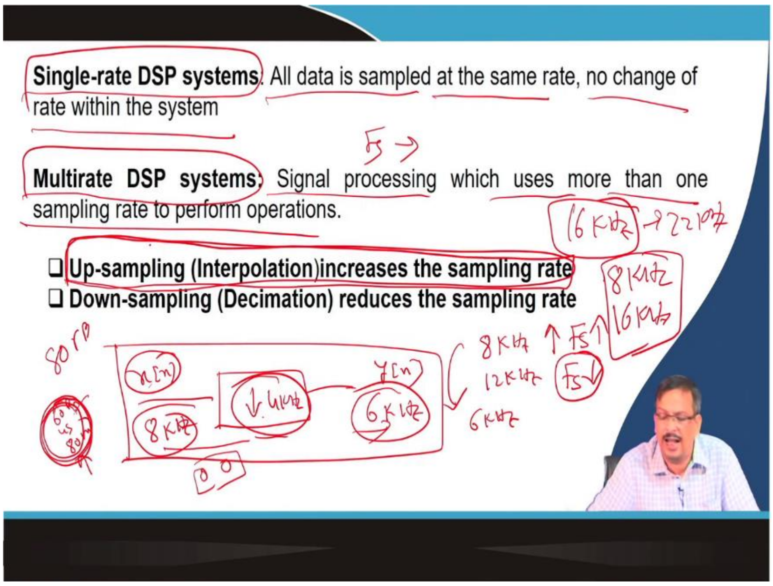
(Refer Slide Time: 00:54)



So, in digital signal processing, when I say, then I say, let us say xa(t) is a time domain signal, and xa(t) is a time domain signal. When I say that the xa(t) is sampled with a sampling frequency Fs, then I know that I get the digital signal x[n] after quantization. Now, then, what is the rate? The rate means the rate at which I collect the sample. So, if I say that my whole signal is in a single rate, that means the xa(t) is sampled in Fs, then I call this a single-rate DSP single-rate signal processing. What is the meaning?

This means that let us say I have a system, I have an input x[n], I have a system, let us say h[n], then I get a y[n]. Now, if I say the sampling frequency of x n, sampling frequency for h[n], and sampling frequency of y[n] are all the same as Fs, then I call it a single rate signal processing. So, I have not changed the rate of the signal, which means the signal's sampling frequency. So, x[n] is sampled as Fs; h[n] or the processing is also done on sampling frequency Fs and output is also in Fs, then I said it is a single rate signal processing.

(Refer Slide Time: 02:28)



So, a single rate means all the data is sampled at the same rate, with no change of rate within the system. So, when I apply x[n] in the system, I do something and get y[n]. So, in this process, I never changed the rate of the sample. So, the sampling frequency will remain the same. So, that is called a single-rate DSP system. Now, once I call the multi-rate DSP system, in this case, signal processing, which will use more than one sampling rate. So, in that case, in multi-rate, it is not that x[n] may be in Fs y[n] may. Be I want F y.

So, the sampling frequency of y[n] is different from sampling frequency x[n], or I can process it using a different sampling frequency, F y, and then I get the output. So, when I say that in a DSP system, when I use multiple sampling frequencies inside the system, then I call it multirate signal processing.

So, in single-rate signal processing throughout the DSP system, my sampling frequency is Fs; in multi-rate signal processing throughout the DSP system, a different sampling frequency will be used. That is why it is called multirate signal processing.

Now what is sampling? As we studied in week number 1, I have an analogue signal xa(t), then I have converted to x[n] capital T where capital T is equal to 1 by F s. That means a continuous time is discretized or a discrete signal I am making discretized. So, continuous time is discretized using an interval T. So, after every T interval where T is equal to 1 by Fs, I am collecting the sample.

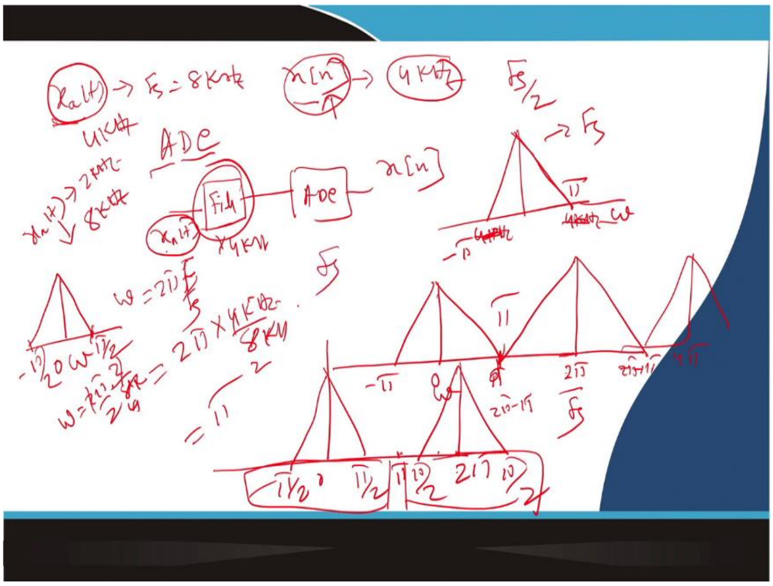
So, if I say my Fs sampling frequency is 8 kilohertz, then I know that within 1 second, I will get an 8 K sample. So, in 1 second, I will get 8 K samples. Now, once I say that Fs is equal to 16 kilohertz, then within 1 second, I will get a 16 K sample, so if you see that I increase the sampling frequency, the number of samples increases, and the number of samples within a unit of time increases.

If I say I have a signal that is sampled is equal to 8 kilohertz and I collected a 20-millisecond signal, then within 20 milliseconds, there will be a 160 sample. But when I said the same signal, if I sampled at 16 kilohertz and now I am collecting 20-millisecond signals, then I know within 20 milliseconds, there will be a 320 sample.

So, when I increase the sampling rate, the number of samples per second increases at the number of seconds within a time. So, that means if the Fs is high, then the number of samples also increases. Then, if I say ok, I know the sampling frequency I know.

So, what is sampling frequency is nothing but the x[n] T; T is equal to 1 by F s. So, in 1 second for 8-kilohertz sampling frequency, I got an 8 K sample of 8 in an 8-kilo sample now if I want to know what is inside the frequency domain.

(Refer Slide Time: 06:41)



Let us say I have a signal xa(t) whose sampling frequency is Fs is equal to 8 kilohertz, and I get x[n]. Then, as per the Nyquist rate, what is the maximum frequency contained in the x[n]? What is the maximum baseband frequency? So, I know the maximum baseband frequency is 4 kilohertz. So, if the sampling frequency is 8 kilohertz, then I can say that the maximum frequency contained in x[n] is equal to 4 kilowatts, which is nothing but an Fs by 2.

Now, when I convert xa(t) to x[n], that is why the analogue to digital conversion A do C A to A ADC analogue to digital conversion. What did I use? I use the anti-aliasing filter. So, here is a; xa(t), there is a filter which restricted, or I can say the band limited the input signal with 4 kilohertz. Then I convert to ADC, and I get x[n]. So, there is an anti-aliasing filter in which the band limits the signal xa(t) to 4 kilohertz if the sampling frequency is 8 kilohertz.

So, I can say that when I say digital signal, it is automatically a band-limited signal. Whatever the signal's highest frequency will be, Fs y 2, it is limited by an aliasing filter, so if I say that I have a signal which has a 4 kilohertz some 4 kilohertz frequency. So, up to 4 kilohertz, I have a signal, or if I say this axis is ω. So, what is ω? Ω is radian per sample. So, ω is equal to 2π F by F, the analog frequency content divided by F s.

If I say that I have a signal whose maximum frequency is 4 kilohertz and sampling frequency is 8 kilohertz 2π into 4 kilohertz divided by 8 kilohertz. So, I can say this is nothing but a 2, so, it is nothing but a pi. So, maximum rate of oscillation exist in the baseband signal is π 2 minus π that is my band limited signal. So, if I sampled it Fs then what is the frequency domain representation this axis is ω, this is 0, so this is pi, this is minus pi. Then I can say here I have a Fs; Fs is nothing but a 2 pi.

So, there will be an π and there will be an minus pi. So, it is 2π plus π and 2π this point is 2π minus π which is nothing but a pi. Now, suppose I have a signal let us say I have an signal xa(t) whose highest frequency content is 2 kilohertz and I sample this signal with 8 kilohertz. Then if I want to draw the baseband signal, so this axis is ω, this is my 0; then I know what is the value of this point this point is ω is equal to 2π 2 kilohertz by 8 kilohertz. So, 1 by 4 2π/2, this is π/2, minus π/2.

So, if I want again to draw this signal against the sampling after sampling frequency. I add the 0; I will get π/2, which is also π/2, minus π/2. Let this be 2π, and this point is 2π, and this point is pi. So, I can get here also π/2 minus π/2, so I know this is not aliasing. So, this is ok, and this is the maximum frequency of the content. So, that is the frequency domain representation of the sampling. So, when I sample a signal, I get this kind of representation in the frequency domain 2π is nothing but an Fs then I will get another component here, which may be 4π twice Fs that way, I will get it.

So, Fs is the sampling frequency. Now I said I want to use a DSP system where instead of single Fs, I have multiple sampling rates and multiple sampling rates. That is why it is called a multi rate DSP system. So, let us say I have a system like this, and this input is x[n] in 8 kilohertz. I want to process it in, let us say, 4 kilohertz, and then I want to get I want to get the output in 6 kilohertz, y[n] is in 6 kilohertz.

So, if you see the sampling rate for x[n] is 8 kilohertz, the sampling rate for processing is 4 kilohertz, and the sampling rate for output is 6 kilohertz. So, I have different sampling rates I have defined and that is why the whole system is called a multirate DSP system or multirate signal processing. Now, if I say, suppose I have a signal at 8 kilohertz. So, what I can do is either up the sampling rate. So, instead of 8 kilohertz, I may say 12 kilohertz I can want, or I can down the sampling rate or 8 kilohertz I want 6 kilohertz, let us say. So, either I can increase the Fs, or I can decrease the Fs.

If I increase the Fs, this is called up sampling; when I increase the sampling frequency, this is called up sampling. When I decrease the sampling frequency, this is called down sampling. So, to make it multirate signal processing, I may required to increase the sampling frequency, or I may be required to decrease the sampling frequency. When I increase the sampling frequency, this is called up sampling; when I want to decrease the sampling frequency, this is called down sampling. Practical life example.

What will happen if I make it up sampling without restricting that? Suppose I have recorded a signal with 16 kilohertz? Let us say you recorded audio in 16 kilohertz; if I play it with 22 kilohertz, then what will happen? Or earlier example, suppose if you see in a gramophone record that every record contains a fixed rpm, in the top of the record, it is written 60 rpm 45 rpm or 80 rpm; all rpm is written.

So, in the gramophone record using a pin, I am collecting the audio information 60 rpm; that means, the p the rotation speed of the gramophone records is 60 rpm. So, if I have a gramophone recorded at 60 rpm. If I play with it at 80 rpm, that means I am upsampling the recorded signal; when I up sampling it, the quality of the signal will be changed because I am not converting it.

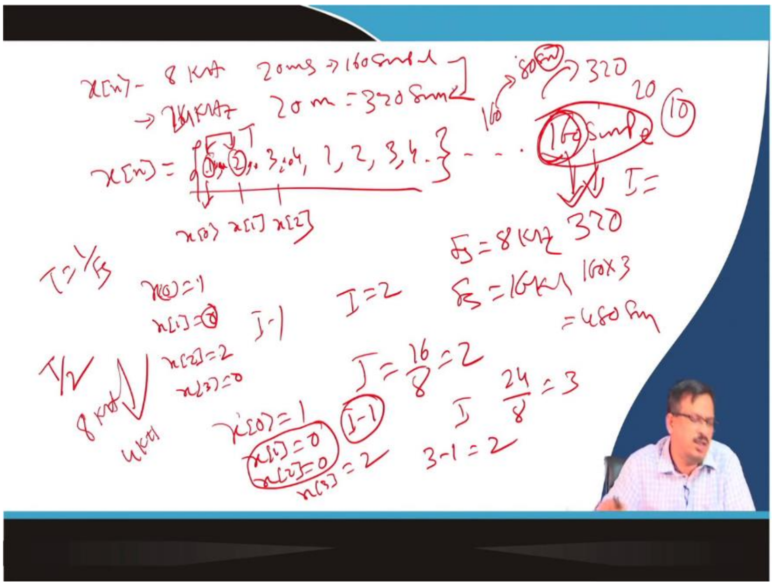
In your tape recorder, also, what is their motor speed? The tape recorder, if you see that record the tape is rotating using a motor inside the tape recorder. If you increase the motor speed, what will happen? With a female voice, a male voice becomes a female voice; if you decrease the speed, the female voice becomes a male voice you can see it. So, a lot of things will happen.

So, I can say that if I recorded a signal an 8-kilohertz audio signal, I have to play with 8 kilohertz then I get the original sound. If I recorded a signal in 8 kilohertz, if I play it at 16 kilohertz, then I require I am not only playing, I am not processing it. So, when I say upsampling, I am not only playing it, I am processing it. So, what I want in case of upsampling also I do not want the signal quality should change.

So, in the case of the upsampling process, I have an 8-kilohertz sampling frequency; I want to make it 16 kilohertz. I do not want a change in the signal quality or change of the signal. So, it is not the example of like the tape recorder. If I do in a tape recorder, I am basically not sampling; I am not completing the process; I am playing it only with a very high sampling rate.

So, there is a change of frequency is happening. So, upsampling also there is a concern. So, once I say the upsampling. So, can I up-sample it without knowing what is happening in the signal? That means I am saying that suppose I have a signal of 8 kilohertz? So, when I say I want to change the sampling frequency, let us say I have a signal with 8 kilohertz. I want to change the signal to 16 kilohertz.

(Refer Slide Time: 17:13)



So, I have an x[n] recorded with 8 kilohertz. So, in 20 milliseconds, I have 160 samples, but if I want to change the Fs is equal to 16 kilohertz. Then, in 20 milliseconds, I have 320 samples; that means the number of samples increases by 2 times. So, suppose I have an x[n], which is nothing, but let us say 1, 2, 3, 4; 1, 2, 3, 4 all are the sample values. So, this is x 0, this is x 1, this is x 2.

Now, if I say when I say let us say this kind of 160 samples is there, now I want to make it 320 samples within the same time. So, how can I do that? So, I can say that in between 2 samples, I can put 1 sample 1, 0. So, I can say x[0] is equal to 1, but x[1] is equal to 0; x[2] is equal to 2; x 3 is equal to 0. So, what am I doing? Since I have to increase the 2 times, so let us say the 2 to I is equal to 2. So, the upsampling factor, I call I am equal to 2.

So, the sampling factor is 2 because I have 160 samples; I want to make it 320. So, in sampling frequency, I have an Fs equal to 8 kilohertz; I want to make it Fs is equal to 16 kilohertz; that means I is equal to 16 by 8 is equal to 2. So, how do I do that? I can put it this way. So, instead of 160 samples, I get 320 samples. So, I put a 0 in between a sample of how many 0; I have to put I minus 1.

If I want to make it, Fs is equal to 24 kilohertz. Then, the number of samples will be how much? 160 into 3; 3 times I am up sampling 3 times, so; that means 480 samples. So, if I want to make it a 480 sample, that between x[0] is equal to 1, then x[1] will be 0, and x[2] will also be 0, and then x 3 will be equal to 2. So, I am putting in between two samples minus 1 number of 0. So, in the case of here, it is 24 by 8 is equal to 3; I is equal to 3.

So, I am putting I 3 minus 1; 2 number of 0 in between 2 samples. So, when I make upsampling, I put 0, which is called interpolation; upsampling sometimes I sometimes is called interpolation. So, I am interpolating the signal between them. So, what is the new T? So, in the case of 1 and 2, the distance between the 2 sample time distances is T; T is equal to 1 by F s.

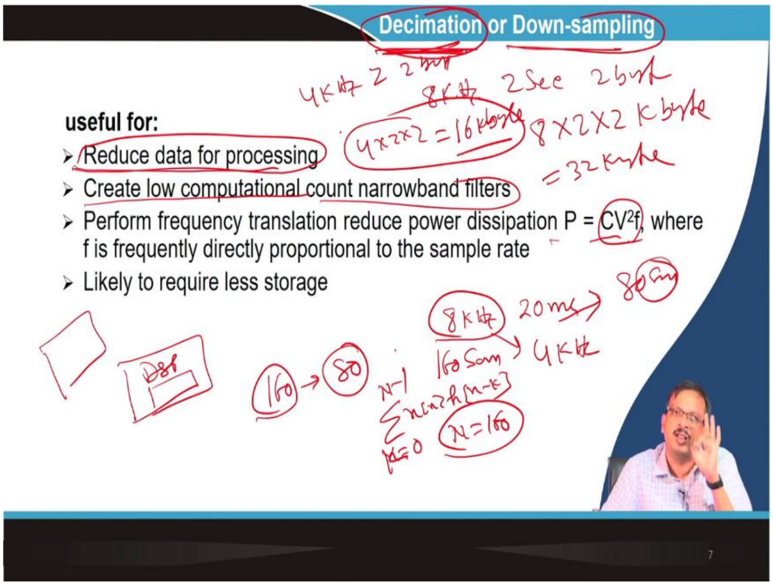
Now, once I say it is upsampled, that means Fs is equal to 8 kilohertz. Now, once it is Fs equal to 2 times, that means T is reduced by 2 times. That is why I have to put another sample in here; this is called upsampling. So, now, I put a so put 0 here. So, what will happen in the frequency domain? I have to think about that also. Unless what will happen? If I put it, the sampling frequency changes.

But if I so when I say when I give the example of tape recorder, I am not inserting 0. What am I doing? I have recorded the sample 160 sample I have recorded. So, within 20 milliseconds, I am supposed to play 160 samples, but since my speed is double, I am playing it within 10 milliseconds; that means a 20-millisecond signal is squeezed into a 10-millisecond signal.

So, the frequency domain will be changed, but what do I want? I do not want that frequency domain to be changed. The next one supposes I have an 8-kilohertz signal I want to downsample; that means I want to decrease the sampling rate, let us say, to 4 kilohertz, so up by 2, down by 2 also. So, I can either down-sample it or up-sample it.

So, a downsample means I have to. So, instead of 160 samples in 20 milliseconds, there will be only 80 samples. So, what is it? Out of 160 samples, I have to pick up every alternative sample. So, I get the 180 samples. So that is called downsampling. Now, we understand down sampling upsampling.

(Refer Slide Time: 22:47)



Then why do I do that? Why do I do downsampling, and why do I do upsampling? So, it is useful for dissemination or down-sampling. Why is it called dissemination? I am saying that out of 160 samples, I want to make 80 samples. That means some of the samples I am discarding. That is why it is called dissemination. So, downsampling is called dissemination. So, from 160 samples to 80 samples, the number of samples is reduced.

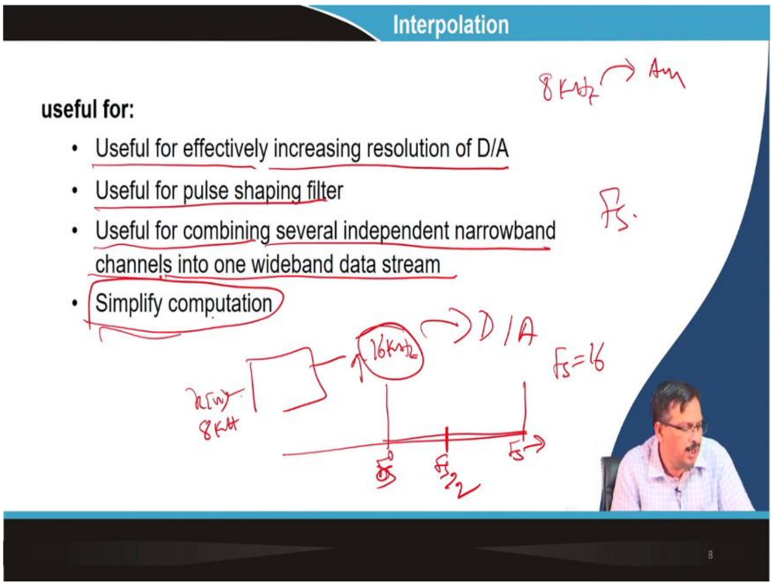
So, if it is reduced, then the data processing is reduced. Suppose I want to implement a convolution, and I have a signal that is 80 kilohertz as the sampling frequency, and I take a window size equal to 20 milliseconds. So, when I say that, I said 160 samples are there. So, I know convolution equal to n equal to 0 to N minus 1; capital N is equal to 160 x[n] into let us say h[n] let us say it is n minus k. So, this is k, this is not 1. So, k equals n, which is equal to 160 samples.

So, when I say the convolution, if my size is data size increases, the complexity increases. If the data size is instead of 8 kilohertz, if I say the sampling frequency is 4 kilohertz, then I know within 20 milliseconds, I have only 80 samples, so my processing time will decrease. So, it reduces data processing, creates a low computational count or narrow band filter, and also reduces power dissipation.

So, you know that for any handheld device, when I say the power dissipation factor, the P is equal to CV square into f; f is the frequency. So, if my processing speed is slow, then the power dissipation will be lower. So, when I process the data inside the DSP processor, if I say I want to process it with a low sampling frequency, that means the rate of data is low, and it also requires less memory.

Suppose I said I have a sampling frequency of 8 kilohertz. If I want 2-second data, I require each sample to be 2 bytes. Then I know I required 8 into 2 into 2 kilobyte space, which is equal to 32 kilobyte. Now, let us say the same signal I sampled at 4 kilohertz 2 second data 2 bytes each sample. So, 4 into 2 into 2, I required 16 kilobytes, so now storage space is reduced. So, when I down-sample the signal, the storage space is also reduced.

(Refer Slide Time: 26:03)



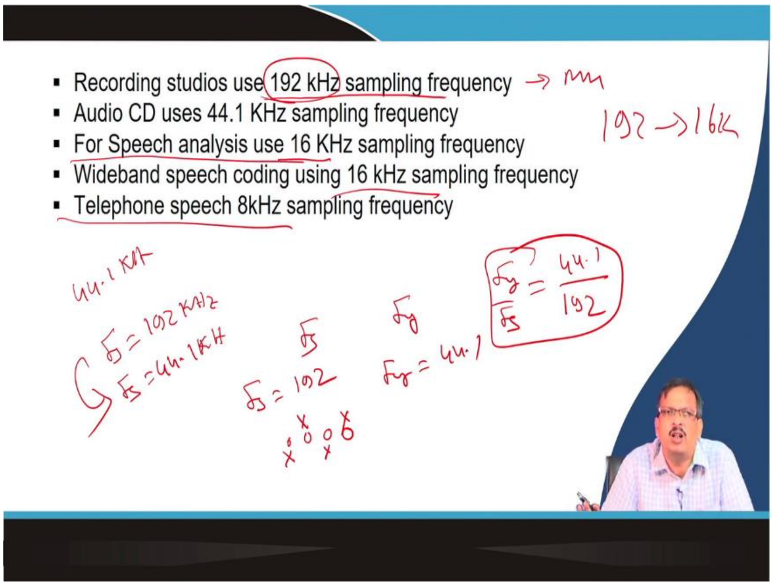
Similarly, in the case of upsampling or interpolation, it is useful for effectively increasing the resolution of the DA digital to analogue converter. So, suppose you know that I have an 8-kilohertz sampling frequency. I want an exact copy of that input signal when I want to convert to an analogue signal. So, if the number of samples within a time increases the efficiency or effect, I can say there is an increase in the resolution of the digital-to-analogue converter.

So, suppose I have a signal x[n] in 8 kilohertz. I process it within 8 kilohertz, and then output I increase to 16-kilohertz sample it and then apply digital to analogue converter. Then what will happen? Effectively, my sampling frequency is increasing, which is why the resolution of my digital-to-analog converter will be increasing. Useful for pulse shaping filter, I will come to that point.

It is useful for combining several independent narrowband channels into one wideband data stream; suppose what I want. Let us say I have a signal. So, you know if I have an Fs, the Fs equals 8 kilohertz. So, you know that the distance between 0 to this is 2 I Fs; Fs is 8 kilohertz. So, what is the bandwidth? Bandwidth is basically the signal I can keep within that Fs is Fs by 2, so 4 kilohertz. Now, once I increase the Fs, that means Fs is equal to 16 kilohertz, then the effective bandwidth of the baseband signal is Fs by 2; 8 kilohertz.

So, instead of 4 kilohertz, I can put 8 kilohertz. So, I can put a lot of narrowband channels into a single F s. It also sometimes simplifies computation; if I increase the sampling frequency, sometimes not the computational complexity computation. So, like in the case of filtering, let us say if the transition bandwidth increases, then the order of the filter decreases. So, somehow, if I use the sampling process to help me increase the transition bandwidth, then I can say the order of the filter decreases. I will give you one example of this.

(Refer Slide Time: 28:51)



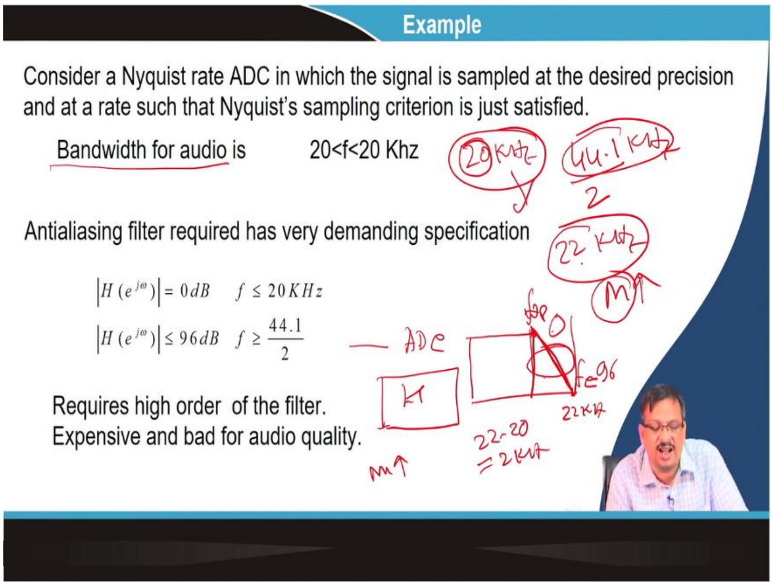
Let us give another example: if you go to any recording studio, you can see the recording studio uses 192 kilohertz sampling frequency for master voice. That means when the recording student studio records that audio signal for the master cassette. They use 192 kilohertz sampling frequency, but for any audio CD, the sampling frequency is 44.1 kilohertz. So, I have an F, which is 192 kilohertz.

Now I want 2 Fs, which is equal to 44.1 kilohertz. So, can I have to sample it down, I have to downsample it? So, Fs is there, and I require an F y. So, I have to downsample it. So, Fs is equal to 192, and Fy I require 44.1. So, how much down-sampling factor do I require? I can easily calculate Fy by Fs. Fy by Fs is nothing but a 4-point 44.1 divided by 192.

So, that is my fraction of, but downsampling when I say that, can I select? So, when I say there is a 4 sample, I want to reduce the downsample by 2. So, there were 2 samples; that means I selected this sample and this sample. I discarded this sample in this sample, but I cannot say 1.5 samples. How can I discard it? However, this can be easily implemented using up-sampling and down-sampling. I am coming into details later.

For speech analysis, we used 16 kilohertz sampling frequency. So, 192 has to be reduced to 16 kilohertz, for speech coding 16 kilohertz, and for telephone bandwidth 8 kilohertz. So, from master wires, can I down-sample it that way or up-sample it down sampling? So, I required many applications, where I required a down sampling and an up sampling. So, how do I do that?

(Refer Slide Time: 31:06)



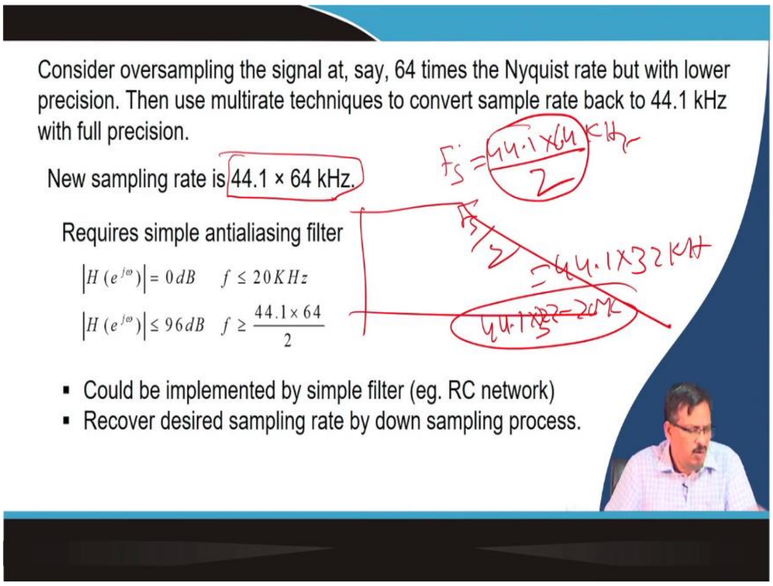
For the next example, I said that sometimes sampling decreases the complexity of the system. What is the example? Let us say I have an audio bandwidth. You know the audio bandwidth is 20 kilohertz, which is the maximum bandwidth of the audio signal perception of the human being; a human being can perceive audio up to 20 hertz. So, if it is 20 kilohertz, then I know the audio CD sampling frequency is 44.1 kilohertz. What is Fs by 2? 44.1 divided by 2; that means roughly 22 kilohertz.

So, the maximum signal can be contained if the sampling frequency is 44.1 kilohertz, which equals 22 kilohertz. I know my audio bandwidth is 20 kilohertz. So, when I design the analog-to-digital converter, I first use an aliasing filter. So, what is the property of this low pass filter, low pass anti-aliasing filter? I required up to 20 kilohertz flat response. Then I know that within 22 kilohertz, it must be sufficiently attenuated. So, that aliasing can be avoided.

So, I can say within the transition bandwidth, 22 minus 20. So, this is ω P the f P, and this is f C. So, I can say f C minus f P 20; that means, within 2 kilohertz, this has to be sufficiently attenuated, let us say 96 dB attenuated, so it is minus 96 dB. So, the transition bandwidth of the filter is 2 kilohertz. Now you know the transition bandwidth and order of the filters are inversely proportional.

If I increase the order of the filter, then the transition bandwidth decreases. So, if the transition bandwidth is low, then the order of the filter will be very high, ok or not. So, the order of the filter will be very high. So, within 2 kilohertz, if I have to design the FIR filter, I know I require a high M will be very high.

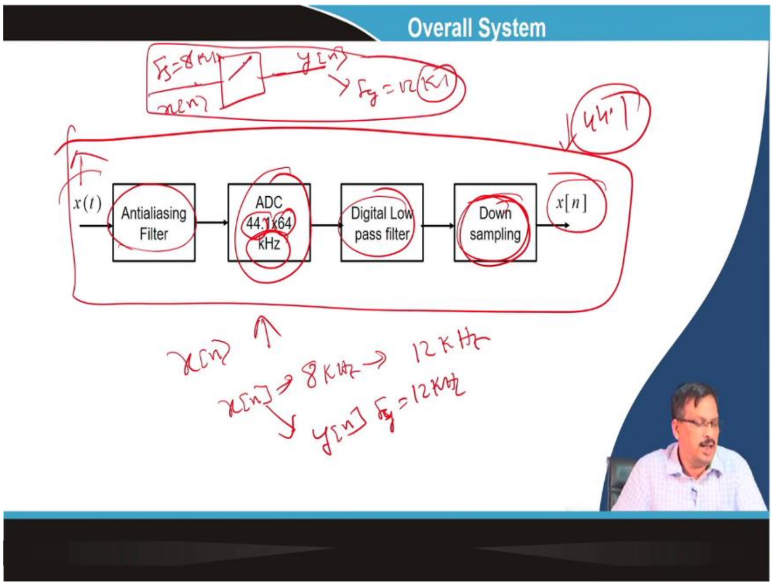
(Refer Slide Time: 33:47)



But let us say instead of designing this, I let us say instead of now my sampling rate is Fs instead of 40 kilohertz; I said 44.1 kilohertz. I said it is 44.1 into 64 kilohertz. So, 64 times, I up-sampled the signal. So, 64 times I upsample the signal; that means, once I do 64 times sampling, what are the Fs by 2 values? This is divided by 2, which is nothing but 44.1 into 32 kilohertz. So, if I say 44.1 into 32 minus 20 kilohertz. So, there is a huge long transition bandwidth.

So, if the transition bandwidth is very long, then I know the order of the filter is very low. But the problem is I have to sample the signal from 44.1 into 64 into 2 kilohertz. Somehow I had to sample that signal, so that also required another end. So, if I up-sample it, but that is already done, and by an anti-aliasing filter, it can be done. This is notional. So, when I design a filter, sometimes upsampling can reduce the complexity of the filter.

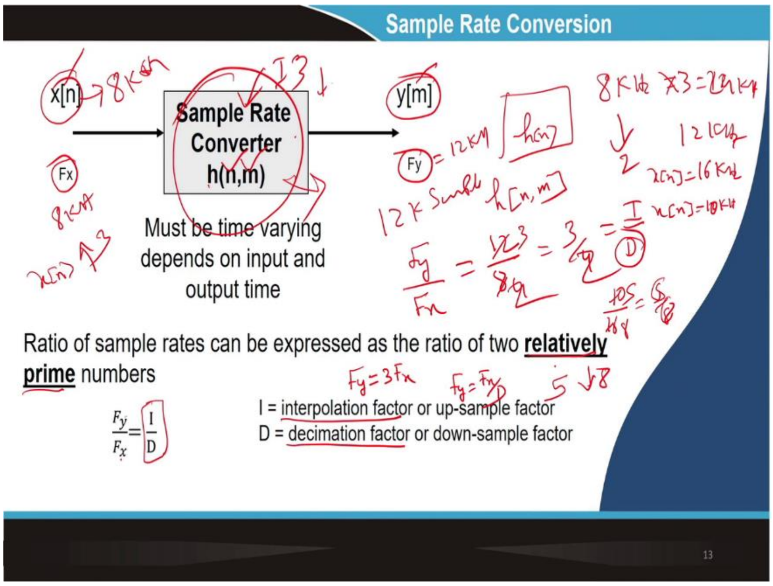
(Refer Slide Time: 35:25)



Now, I come to that overall system. So, when I say I have an x[2], I have an anti-aliasing filter, digital low pass filter, and downsample, and then I get x n. Because I have an x[n], I up-sampled it, but again, I have to revert back to the system. So, suppose I said, let us say so x t; ADC is 44.1 into 64 kilohertz. I have up-sampled it. But I have to sample it because I want x[n] to be in 44.1 kilohertz. So, I have to sample it down. So, the overall system diagram will look like this. Now, suppose I have given you a problem.

Suppose I have a signal with 8 kilohertz and want to sample it up. Let us say I have a signal x[n] that is sampled at 8 kilohertz. I want to upsample it to 20 kilohertz a 12 kilohertz. I want the signal y[n] from x[n], which is Fs, and Fy equals 12 kilohertz. Have you understood or not what I am saying? I have a signal. Let me have a signal whose Fs is equal to 8 kilohertz x[n]. I want another signal using the signal processing term block. So, my output of the system is y[n] and whose sampling frequency Fy is equal to 12 kilohertz. So that is called a sampling rate converter.

(Refer Slide Time: 37:19)



So, when I say the sampling rate converter, the sampling rate converter is nothing but x[n], which is the input whose sampling frequency is Fx. I want a y[n] as an output whose sampling frequency is Fy. So, the process of the system or inside that what I have to do for that whole system is called sampling rate conversion, or the system is called sampling rate converter.

So, in any system, I have an impulse response. So, the input index is n, and the output index is m, and that is why the sampling rate converters impulse response is represented by h, n, m. Because the index is different, if I say Fx is equal to 8 kilohertz, then in 1 second of x[n], I get an 8 K sample. Let us say Fy is equal to 12 kilohertz, and then I know that in 1 second, I get a 12 K sample. So, the index m and n are different. So, how do I do that? So, what I want is Fy, which I gave Fx. So, what is the ratio?

The ratio is nothing but a 12 divided by 8, so I can say 3, 4. So, 3 by 4 is called I by D; I is the upsampling factor, D is called the downsampling factor or sometimes called the interpolation factor and D is the decimation factor. So, what will I do? I will get an x n.

First, I will up-sample the x[n] 3 times. So, 8 kilohertz is up the sample by 3 times. So, it has become 24 kilohertz, and now it is 2. It is downsampled by 2 times, so it is 12 kilohertz. So, I can say if it is upsampling, let us say upsampling, let us say this is an up sampler of I equal to 3. Then, I can say Fy is equal to 3 into F x.

If it is a down sampler of 2, I can say Fy is equal to sorry D Fx by D. Let us say I have given another example. Suppose I said I have a signal x[n] whose sampling frequency is, let us say, 16 kilohertz. I want to make x[n] equal to, let us say, 12 kilohertz or 10 kilohertz. What is the up-sampling factor, and what is the down-sampling factor? So, the input is Fx is nothing but 16 kilohertz, and Fy is equal to 10 kilohertz. So, it is nothing but the 10 by 16. So, I can say it is 5, it is 8, so it is 5 by 8.

So, I can say this Fy by Fy by Fx is equal to I by D, which is relatively prime, so this is prime, so 5, 8. So, I have to upsample by 5 times and downsample by 8 times, ok. So, if I give you a problem, let us say I have a signal frequency of 4 kilohertz, but at the output I want, it should be 44 kilohertz.

So, what should be the up-sampling factor and what should be the down-sampling factor? You can easily calculate ok. So, I will stop here, and in the next class, I will talk about what upsampling is and what downsampling ok.

Thank you.