

Digital Voice and Picture Communication
Prof. S. Sengupta
Department of Electronics and Communication Engineering
Indian Institute of Technology, Kharagpur
Lecture - 8
LDM and ADM

Having seen the improvements that one can get through the differential quantization now we are going to see some specific implementations of the differential quantization and today we are going to discuss about the linear delta modulator; in short form we are calling it as LDM; and then we will see that what are the deficiencies of the LDM; and then we will go in for the adaptive delta modulation that is ADM. So this we are going to cover today

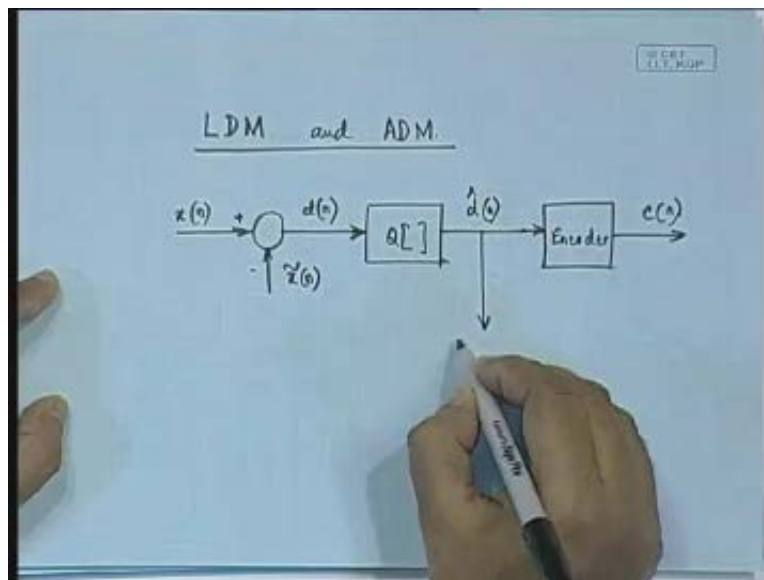
Since we have already known the advantages that one gets in the process of differential quantization especially when there is a strong correlation between the adjacent samples, between the samples which are only one unit delay apart that is why one can take a great advantage of this fact. Any questions? [Conversation between Student and Professor – Not audible ((00:02:07 – 2:25 min))] Differential quantization, okay what is your doubt please? Amplitude is analogy in nature, exactly.

See, when we are having x of n , x of n is a sampled value of the analog waveform. So x of n can have any value in the range that the analog signal is going to have. x cap of n is a quantized version so even that is also in a sense analog. It is only when we have c of n there we are having digital. I suppose you followed this point. So fine; any other doubt that you may be having pertaining to the previous lecture or any previous discussion? Okay. So we can go over to the LDM and the ADM.

First let us see that what exactly are we going to have for the linear delta modulator. Now, for that purpose let us first see the linear delta modulator and its block diagram would be something like this that at the input to the linear differential linear delta modulator we are going to have x of

n and then we will be having a summer block which is having plus and minus over here; in this input (Refer Slide Time: 4:03) we are going to have $\tilde{x}(n)$ which is the predicted value of the sample and then the difference is $d(n)$; this is same as what we had seen for the differential quantization. So there will be Q that is the quantizer module and here we are going to get $\hat{d}(n)$ and then we are going to have the encoder, and then the encoded output will be $c(n)$.

(Refer Slide Time: 4:33)

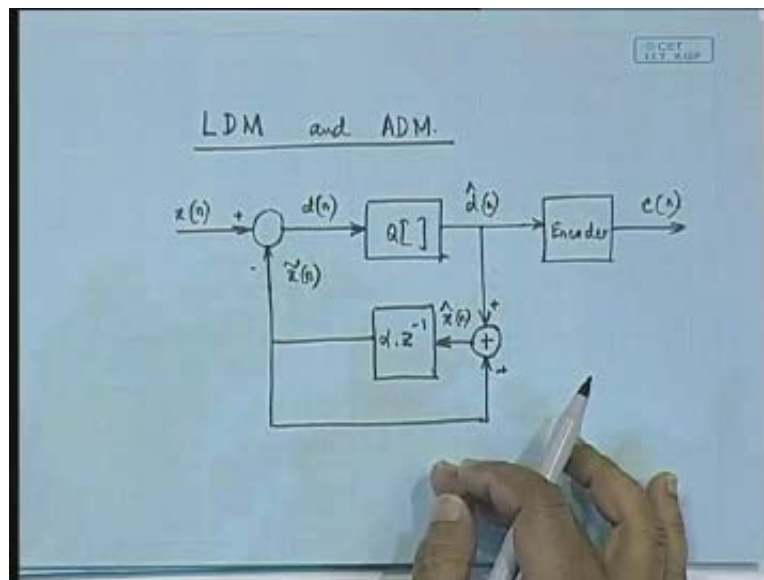


This is nothing new for us. I mean, even we know that what is going to happen after this. So we are going to have yet another summer block but this time this $\hat{d}(n)$ should be added to $\tilde{x}(n)$ and in the process what we are going to get is the $\hat{x}(n)$. So here (Refer Slide Time: 5:01) we are going to get $\hat{x}(n)$ and it is this $\hat{x}(n)$ which will be put through a predictor block.

Now, so far we have seen only the block, I mean, yesterday we were writing that as P the predictor block and now specifically we will implement this predictor and if it is only based on the previous sample; if the estimate is based only on the previous sample then we can take only one previous sample which means to say that if we multiply this by α and put it through a unit delay which is given by z^{-1} then after putting this $\hat{x}(n)$ through

alpha times z to the power minus 1 we will be able to generate the predicted signal which is $\hat{x}(n)$. So this is the basic implementation of the differential coding scheme.

(Refer Slide Time: 6:01)

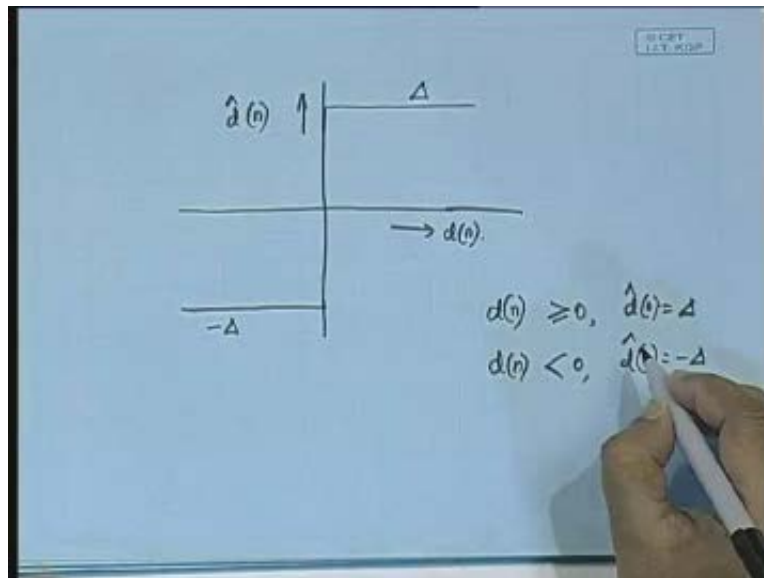


In fact you will find a very similar thing when we are going in for the differential pulse code modulation; the block diagram as drawn like this is no different from this especially when we are having only the estimate based on the previous sample; that is to say what is called as the first order predictor. In fact the order of predictor is defined like this that it basically refers to the number of previous samples that you are using for the implementation of the..... **what you say** for the implementation of this prediction part.

So now let us go over to the specific aspect of the LDM and there what we have to do is that we should have the differential signal. So, if we say $d(n)$; so here if we say $d(n)$ then we are going to have the quantized output that we are going to say as $\hat{d}(n)$. So we have to plot that what will be the characteristic between $d(n)$ and $\hat{d}(n)$ that is what is going to give us the quantizer characteristic. and for the case of delta modulator, **you must be knowing from your knowledge of the Digital Communication course**, that we are going to have these levels as plus delta and minus delta which means to say that as long as we have $d(n)$ to be greater than or

equal to 0 then $\hat{d}(n)$ is going to be plus delta and if $d(n)$ is going to be less than 0 in that case we are going to have the $\hat{d}(n)$ to be equal to minus delta. So it is plus delta and minus delta these are the only two possible levels that we are going to have with the delta modulator.

(Refer Slide Time: 8:36)

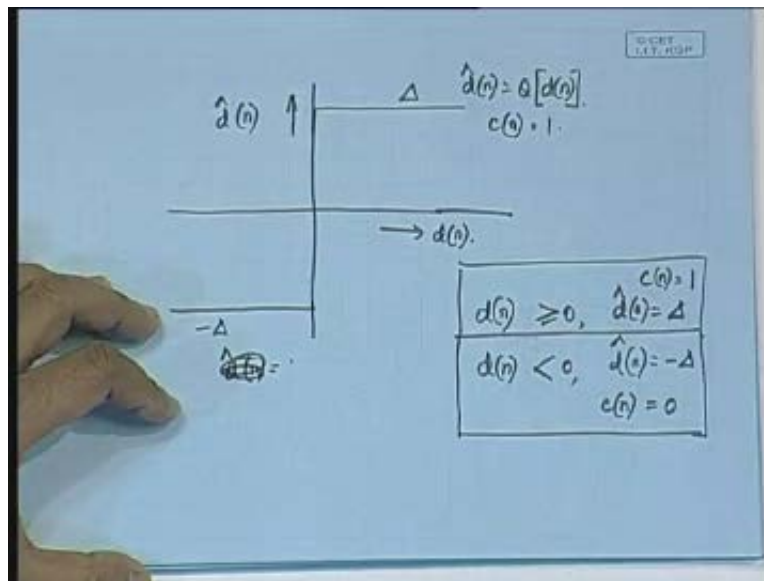


Now look at this that this is basically a two level quantization; either plus delta or minus delta. two level quantization means that how many bits do we require to represent these samples; **only 1 bit is essential** only 1 bit is enough because it is two levels. Therefore, when the output, the digital output c of n is going to be 1 in that case we can say that c of n we will represent as 1 when $\hat{d}(n)$ is equal to plus delta and when $\hat{d}(n)$ is equal to minus delta, we can represent the digital output by $c(n)$ equal to 0.

Hence, when $d(n)$ is greater than or equal to 0 we not only have $\hat{d}(n)$ equal to delta but at the same time we are also going to have $c(n)$ is equal to plus 1 so this for the positive $d(n)$ case and for the negative $d(n)$ case we are going to have $c(n)$ to be equal to 0. So these are the only two possibilities that we are going to have. So we can say that $\hat{d}(n)$, this is delta, so $\hat{d}(n)$ of n is nothing but it is the quantized version of d of n and we are going to have in this case the $c(n)$; actually it is only a matter of convention; you can choose $c(n)$ to be..... if you choose

$c(n)$ to be 1 over here then you have to choose this $c(n)$ to be 0; if you choose this as 0 (Refer Slide Time: 10:20) then you have to choose this to be 1. So anyway that is up to the convention or that that is up to our choice that is what we can have. Hence, this is the delta modulator characteristics.

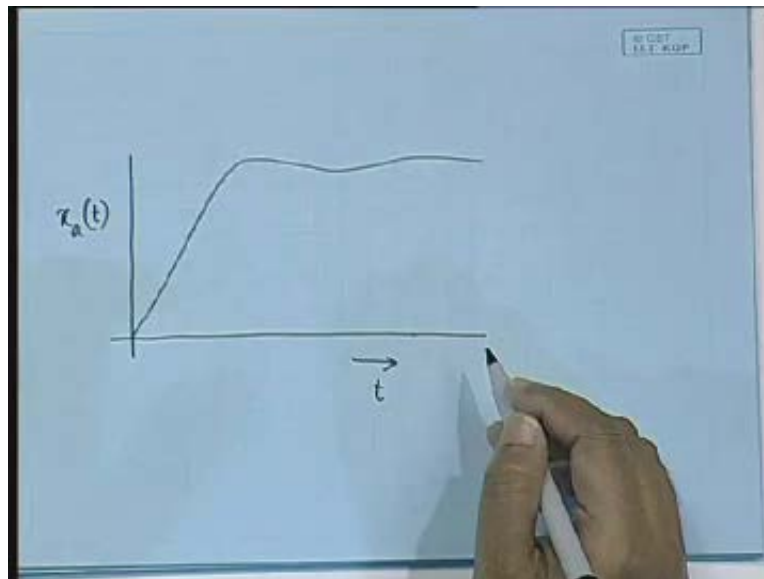
(Refer Slide Time: 10:39)



Now, delta modulator looks pretty attractive in the sense that it just requires 1 bit per sample which means to say that if we have a sampling rate of speed at 8 kHz in that case a simple linear delta modulator would mean that we will be encoding the signal at 8 kilobits per second. But let us see that whether this much of bit rate improvement is really going to help us or not; I mean, whether this improvement is always good for us or are we sacrificing on the performance.

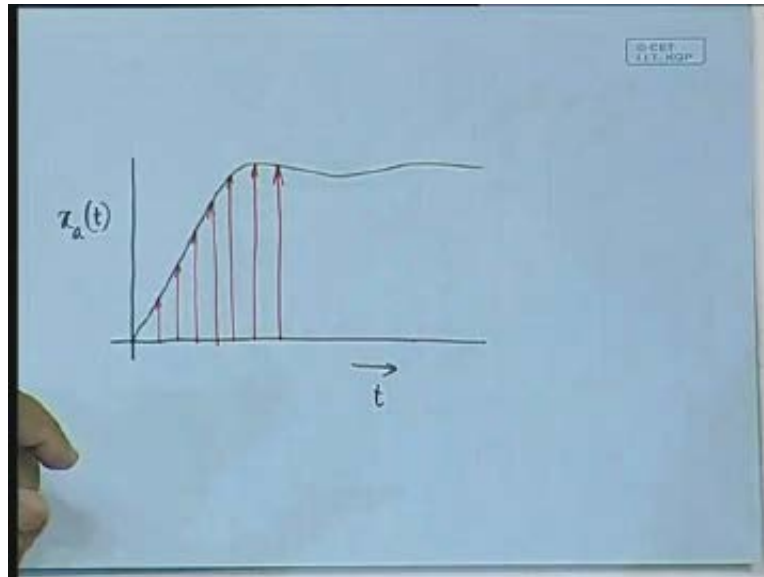
Now for that let us go back to the analog waveform, the original analog waveform that we are going to sample and then going to encode through this linear delta modulator. Let us see that the analog signal; so what we are going to plot is like this that we are going to plot the analog signal; let us write it as x subscript we are giving a just to indicate that it is analog being used as a function of t ; t is plotted in the x axis so horizontal axis indicates the t and the vertical axis that indicates the value of x a of t and let us say that x a of t varies something like this.

(Refer Slide Time: 12:25)



This is a typical variation we have considered with respect to time. The analog waveform $x_a(t)$ has varied with respect to the time. Now let us see what output are we going to expect. When $c(n)$ is..... now we are going to have $c(n)$ according to this; we are going to have $c(n)$ equal to 1 if $d(n)$ is going to be greater than or equal to 0 and we are going to have $c(n)$ to be equal to 0 otherwise which means to say that if we..... let us say that we are taking samples at some intervals; say this is one sample we take, this is the next sample we take (Refer Slide Time: 13:25) let us say that these are the successive samples that we are taking for the signal, for the analog signal.

(Refer Slide Time: 13:40)



Now the first sample is taken over here. Now the second sample is having amplitude of this much. But what we are going to have? We are going to predict the second sample based on the first sample. Because in our block diagram the first sample is over here and the second sample is here (Refer Slide Time: 14:07). So the second sample's prediction will be based on the first sample delayed version. So the system will assume that this sample would be close to what we have over here. Thus, as a result the predicted value will mean that x of n will be higher than \tilde{x} of n because \tilde{x} of n will be based on the past sample and x of n will be based on the present samples so x of n is clearly higher in this case. Since it is higher what is going to be the value of the d cap of n ? d cap of n is going to be Δ . Therefore, here we are going to have a value of Δ in our step size and if d cap of n is equal to Δ what are we going to have from this block; what is going to be \hat{x} of n (Refer Slide Time: 15:13)?

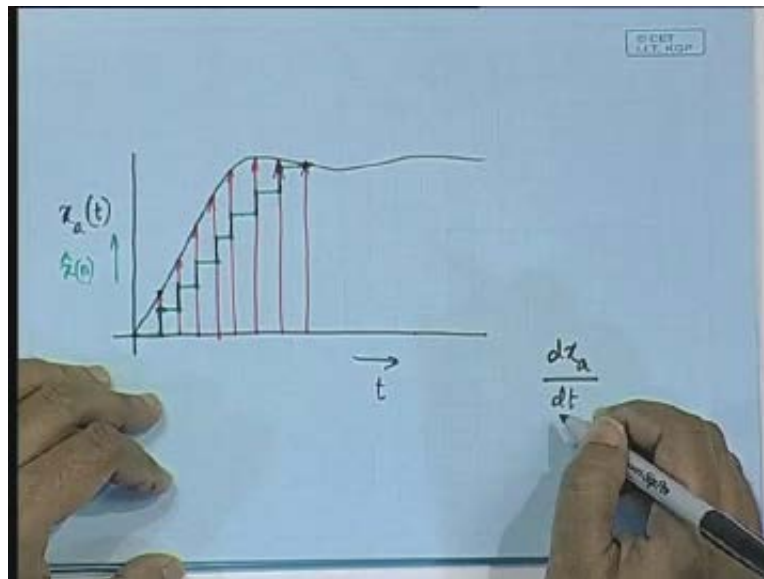
Now the previous sample was zero so zero gets added to Δ which means to say that \hat{x} cap n now becomes equal to Δ . So \hat{x} cap n is Δ , so, if I plot on this axis using the green lines I plot the \hat{x} cap of n . Therefore, if I plot \hat{x} cap of n then here, supposing this is the height that we are having for the step size Δ , so here the value of \hat{x} cap n will be this much and till the next sample arrives the value that we are assuming for the sampled signal will be this much.

When the next sample arrives the previous sample is this. Now based on this it is going to predict the next sample. So the next sample will have yet another increment by delta in its estimate. Therefore, now this becomes 2 delta with respect to this. Now the next sample will be like this; the next sample would be like this; the next sample like this, every time we are increasing by delta, we are very clearly seeing that whatever estimate we are making based on the previous sample it is always lower as compared to the x of n . So we are trying to track down whatever x a of t the analog waveform is there we are trying to track it down through this \hat{x} of n . Now we are trying to track it out by increasing the delta in every step. Now here also we increase by yet another delta. Now only when we come to the next sample perhaps say the next sample would be taken at this instant (Refer Slide Time: 17:32) only here we can see that there is a scope for a change over.

Now the estimated signal can be higher as compared to the incoming signal because there is some droop that this characteristic is indicating or you can say that more or less it is having a uniform value.

Now, before we go into the uniform part of the waveform, let us see that how good is our tracking over here. We are not been able to track it satisfactorily. The analog waveform is varying like this; the estimated waveform or the coded waveform rather because if this is the manner in which \hat{x} of n is going to be there the same will be followed in c of n . Hence, when we are trying to reconstruct the signal at the corresponding decoder delta modulated decoder there also we are going to have the same difficulty.

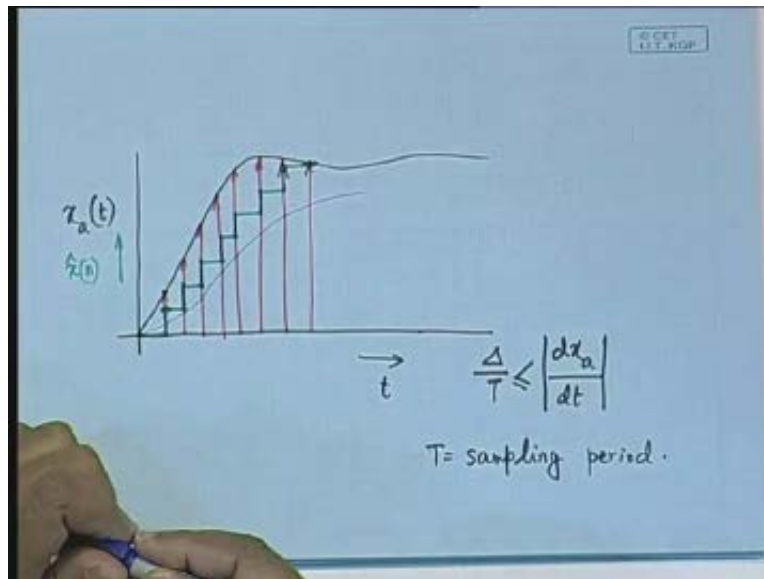
(Refer Slide Time: 18:54)



And why this is happening? Would it have happened if the derivative the time derivative of this waveform $\frac{dx_a}{dt}$; if this was less as compared to the time delta by T where T is nothing but the sampling period? So if we take capital T to be the sampling period and we take delta to be this step size in that case delta upon T , if we have delta upon T to be less than or equal to $\frac{dx_a}{dt}$ then what would have happened? Then could be have tracked it down properly?

We could not have tracked down properly. If delta by T is less than or equal to $\frac{dx_a}{dt}$ it would have meant what we have actually speaking. So here the time derivative of this step size increment is less as compared to the time derivative of the signal. If the signal would have been further steeper like this then it would have failed considerably, then the error in prediction would have been much more.

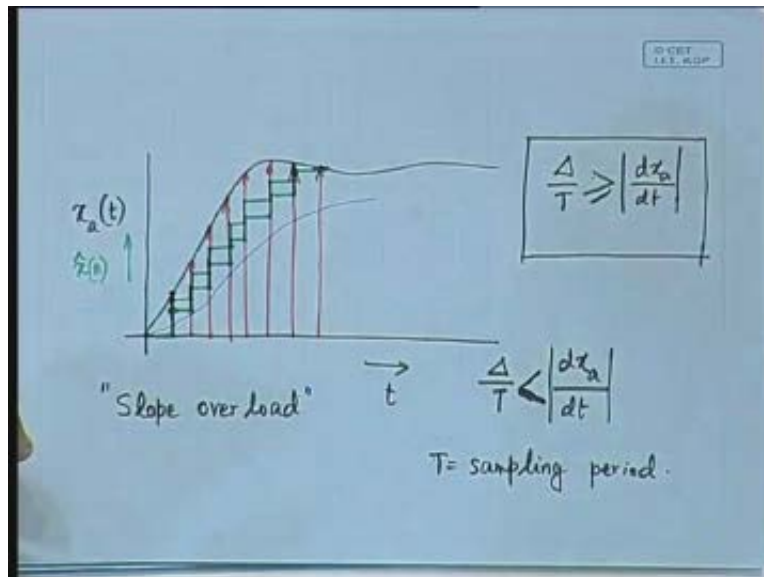
(Refer Slide Time: 20:15)



On the other hand, if it is a slow varying signal; so instead of this if we had observed something like this (Refer Slide Time: 20:21) in that case could it have tracked? It could have, because here the time derivative is much less. In that case we can say that delta upon T greater than mod of d x a upon dt; in fact we should say..... if we say this to be less then we should say this delta by T greater than or equal to d x a by dt should be the condition which must be satisfied so that the analog signal is tracked down properly by the encoded signal.

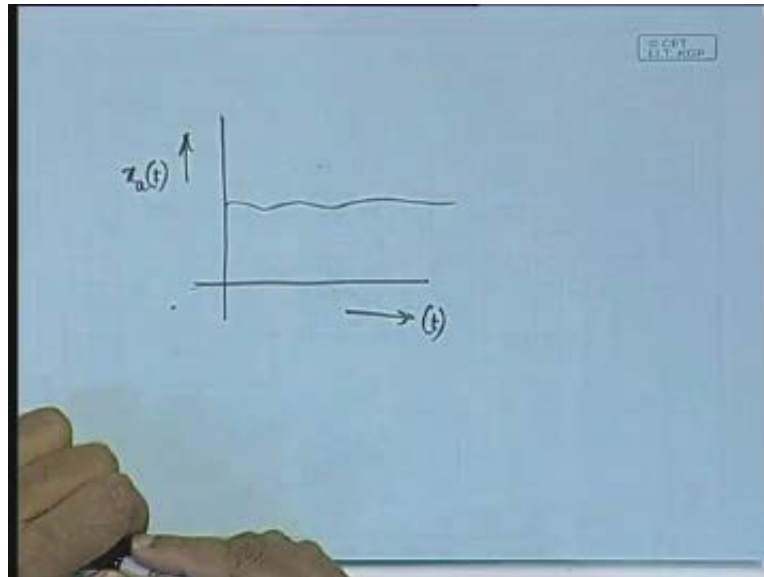
Now this sort of a situation, when this happens the example which we have shown, this refers to phenomena called as slope over load so this is called as slope over load and this is slope over load happens when we have delta by T less than d x a dt. And to prevent the slope over load we should have delta by T as greater than d x a dt. Now a very simple solution we can suggest that why not have an increased value of delta. Your sampling time T is fixed so you increase delta. If you increase delta you are increasing delta by T so at some stage you can definitely exceed d x a dt. So the safest thing is to have increased value of delta.

(Refer Slide Time: 22:34)



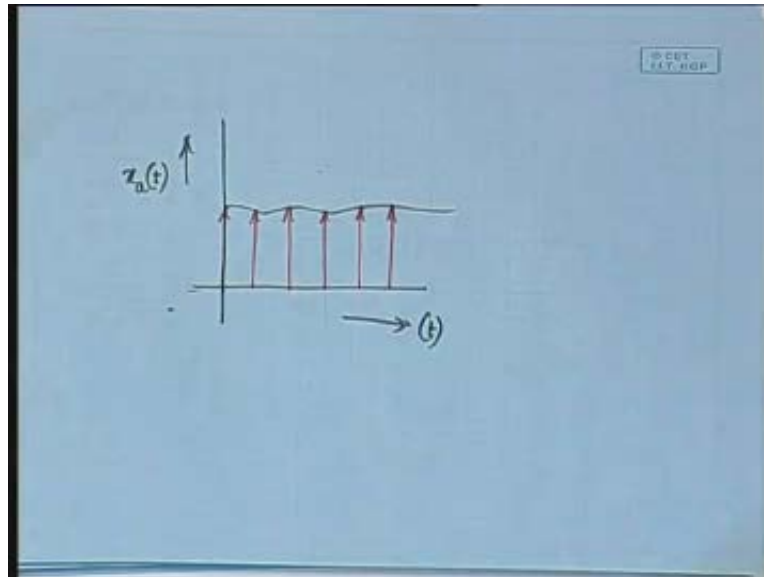
Had it been an increased value of delta then we would have definitely been better off because in that case our encoded waveform would have been a step higher than this so we could have obtained something like this (Refer Slide Time: 22:27). If we increase delta even further we could have tracked down the waveform that would not have been of any difficulty. But look at the uniform part of the waveform now. For better clarity I should not be cluttering this diagram any further so let me draw another waveform of x_a as a function of t versus time where we take this x_a of t to be more or less a uniform value say this is the uniform value.

(Refer Slide Time: 23:06)



Now the first sample let us say is here, so this is where the first sample is (Refer Slide Time: 23:12); this is where the second sample; third sample; the fourth sample; the fifth sample; the sixth sample like this, and mind you, in order to track down the analog waveform we have decided to increase delta. And now with increased delta let us see that how will be our performance for a more or less uniformly varying waveform.

(Refer Slide Time: 23:42)

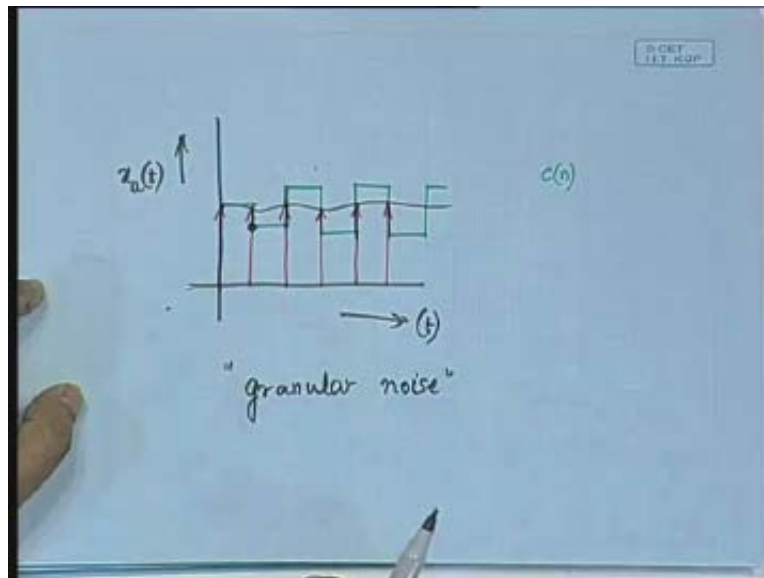


Now, when the second sample comes, we are going to predict the second sample based on the first one. Now clearly the first sample is slightly higher than the second sample so we can say that the \tilde{x}_n will be slightly higher as compared to the x_n ; and slightly higher means that immediately we can conclude that d_n is going to be negative. And because we are having a characteristic which is like this (Refer Slide Time: 24:22); slightly positive or slightly negative does not matter because it is a function like this, so slightly negative means it immediately goes into minus delta. So what we are going to have is that because the signal has reduced we are going to predict this sample as this. This will be the predicted value. So we are going to have it like this then this will be the predicted sample.

Now **the predicted sample** the predicted sample is going to be lower than the next sample so we increase it by delta; and mind you, we have decided to have higher value of delta. So a higher value of delta it goes up, next time it goes down by higher delta, next time it goes up by delta, next time it goes down by delta like this so it is going to alternate; your c_n is going to alternate between 1 and 0. So once we will be having the estimate higher than x_n , next sample we are going to have the estimate lower than x_n because our prediction accuracy is by the factor plus minus delta only. Therefore, if delta is large then for the uniformly varying signal

portion we are going to have a very good amount of error; good amount of estimation error will be there and this error **anybody knowing**? It is called as the granular noise. So this results in what is called as granular noise.

(Refer Slide Time: 26:19)



Hence, we have to face two situations: smaller value of delta would lead to possibilities of slope over load (Refer Slide Time: 26:27) and larger value of delta would lead to the possibilities of granular noise in the slowly varying portion.

So what is the solution?

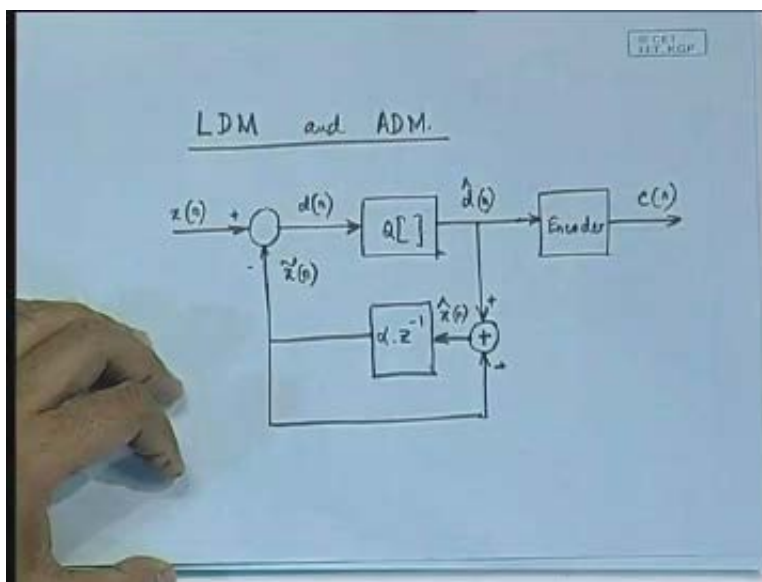
The solution should be that for fast varying portion of the waveform we should have higher step size and for slowly varying waveform we should have a lower step size which means to say that the steps size, we should be prepared to vary the step size accordingly that is why we are saying that again we should not be talking in terms of a fixed delta, we should be talking in terms of delta as a function of n. So the step size now should be made to vary in accordance with the samples. So we should have delta of of n and that is what leads to the adaptive delta modulator.

Now the question of this..... now adaptive delta modulator, in fact, is further more required for the speech signal. You see, for non-speech kind of waveforms, this sort of situation that sometimes it should be fast varying, sometimes it should be slowly varying, these situations may not arise that frequently but look at the speech waveform. Whenever you are having voiced speech you are going to have sharp variations in the signal so there the slope over load is very important. And look at the unvoiced part; for unvoiced part your waveform is going to be more or less uniform and there your reduction of granular noise very important. So for speech we should try to think in terms of the adaptive delta modulator.

How it should be made adaptive?

The first thing is that the block diagram-wise, I suppose that you can make a guess that what change we are going to have in the block diagram. So let us look at the original block diagram of the linear delta modulator. This is the block diagram of the linear delta modulator.

(Refer Slide Time: 29:08)



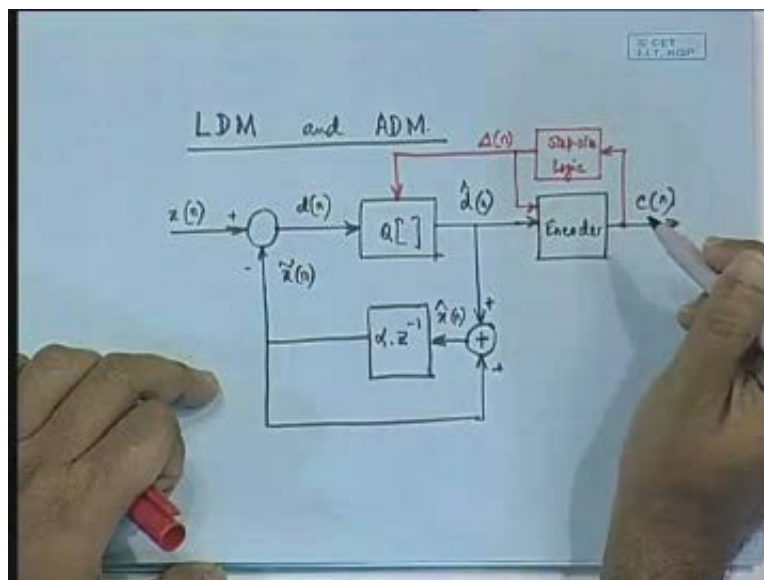
Now, what are we going to have?

Now instead of a fixed delta; so this was having a fixed value of delta. But now instead of a fixed value of delta we should add a block so as to adapt the step size. Now again here we can either

adapt it through a feedforward mechanism or we can adapt it through the feedback mechanism. Let us that we have a feedback mechanism. So, **instead of redrawing the circuit** instead of redrawing the block, I suppose, only the added elements if I draw in a different color that would be more prominent to you.

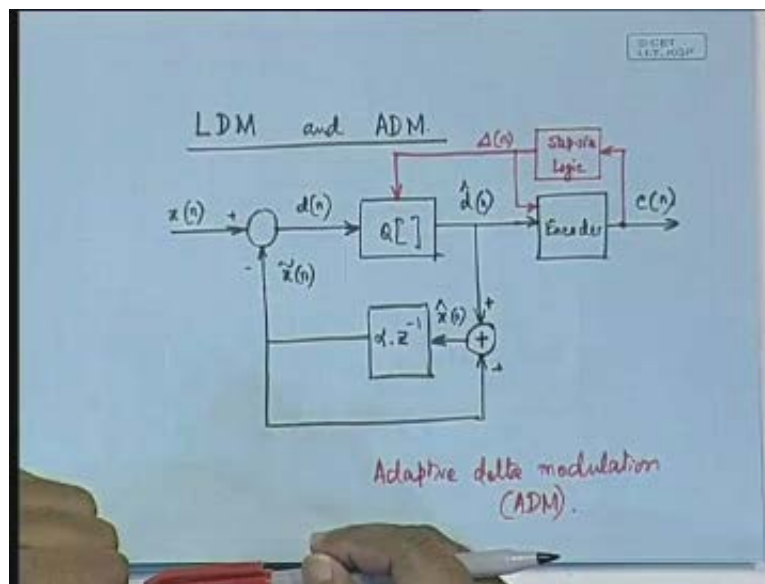
Now we take a feedback from this c of n and then we have a step size logic, so this (Refer Slide Time: 30:09) we are calling as the step size logic. This logic is ultimately going to dictate that whether the step size should be increased or the step size should be reduced. and **if increased** if it is to be increased or if it is to be reduced, in that case to what extent it should be increased or reduced that also should be decided by this step size logic and the output of this step size logic obtains the delta, not delta but delta of n . So this delta of n should be the parameter that we are going to feed to the quantizer and not only that, this delta n should also dictate the encoder because in accordance with this delta n this d cap of n will be decided. So the delta n goes like this. This is with the feedback adaptation. With the red block drawn like this it is a feedback adaptation.

(Refer Slide Time: 31:10)



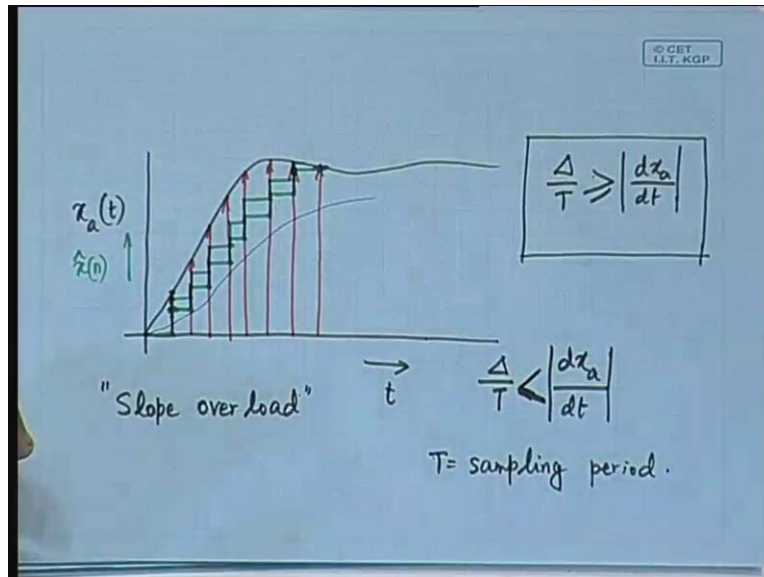
Now I suppose that I need not have to tell you explicitly that if I want to change over from the feedback adaptation to the feedforward adaptation what I should do I should do in a very similar way; the only difference is that there instead of taking the step size logic signal from c of n we should get it from x of n . So x of n followed by the step size logic block and then the output will be $\Delta(n)$ which will control this quantizer and the encoder. That would have been the only difference which we would have got. so this is our so Hence, with this red block addition this becomes our adaptive delta modulation or what we are going to describe in short form as ADM.

(Refer Slide Time: 32:23)



Now how it can be adaptive that is the million dollar question. Now let us see some kind of an intuitive logic that when we should increase the step size and when we should reduce the step size. You see, again let us go back to our example.

(Refer Slide Time: 00:32:44 min)



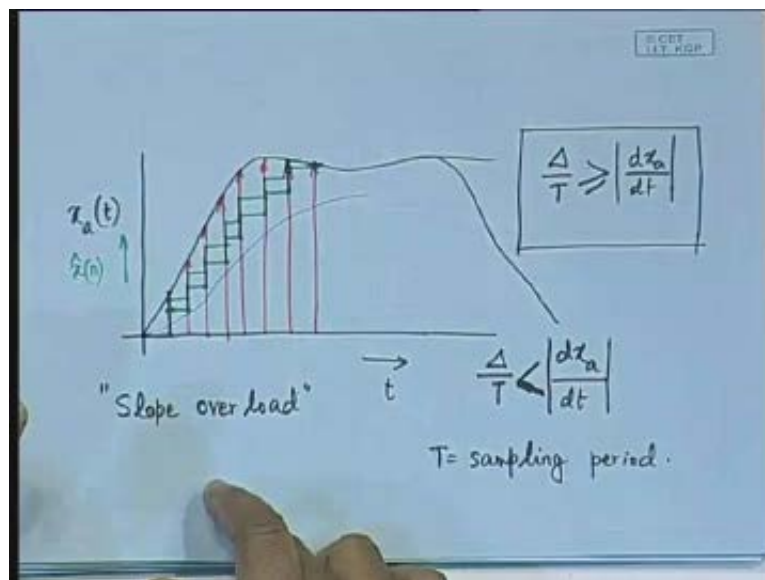
We have two examples: one is where the signal is changing considerably (Refer Slide Time: 32:51) and the other is where the signal is more or less uniform in time. Now when it is changing, supposing I mean, look at our original curve with the value of delta; supposing we see our original delta, now what happens? In this case the value of $c(n)$ is going to be a continuous string of..... in this case x of n is always higher as compared to the x tilde of n so that is why d of n is always going to be higher, positive, which means to say that c of n is going to be 1. So c of n will be a string of 1. Now, when c of n is a string of 1 it means to say that the function is increasing.

On the other hand, if we had a waveform that was like this that supposing after sometime the waveform makes a downward fall like this in that case in this range we would have got $c(n)$ to be equal to 0 and if it is able to track down, if the x cap of n is able to track down the input waveform in that case we would have got a change over from $c(n)$ is equal to 0 to $c(n)$ is equal to 1. But if a condition like a slope over load happens, then we would be having a continuous string of $c(n)$ is equal to 1 for the positively increasing signal or $c(n)$ is equal to 0 for continuously reducing signal. So increasing or decreasing signal in the case of slope over load would definitely give rise to a continuous string of $c(n)$ as 1 or continuous string of $c(n)$ as 0.

If we monitor; now since we are monitoring the $c(n)$ if we find that the past $c(n)$ and the present $c(n)$ (Refer Slide Time: 35:13) is having the same value, if past $c(n)$ is 1 and the present $c(n)$ is also 1, in that case what decision should we take about delta; should we increase or should we decrease?

Yes, definitely the answer is correct which many of you had given that in this case delta n should be increased when we observe that $c(n)$ is remaining the same which means to say that there is a need to increase the step size that is what we tried to do over here. Because when we found that there was a slope over load we were trying to increase the value of delta and trying to track down the waveform. So if we increase delta and see that still the value of past $c(n)$ and the present $c(n)$ happens to be the same, in that case what we should do, we should further increase delta, still if we find we should further increase delta, but okay there should be some limit because ultimately we are going to use some electronic circuitry to do that so the circuitry must be having some dynamic range that is why there be should be an upper limit to which you can increase the value of delta.

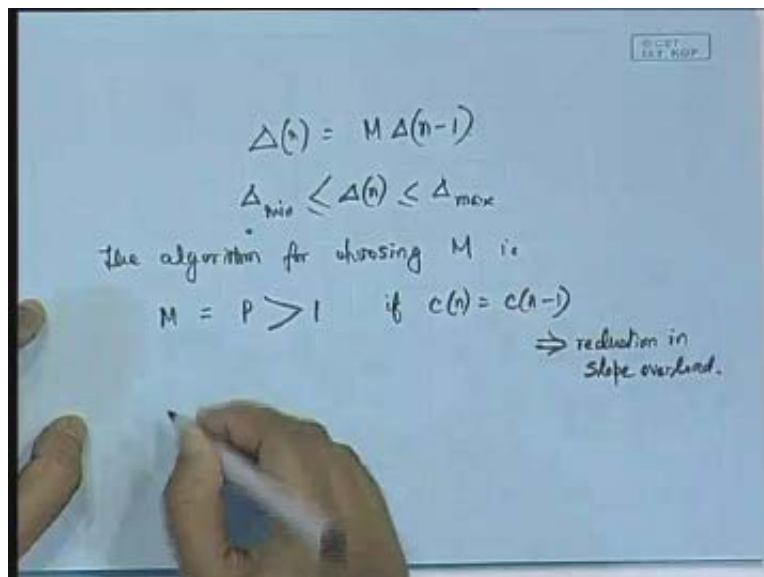
(Refer Slide Time: 436:14)



Now again look at the other situation; that is to say that **when we were considering the slope over load** when we were considering the granular noise (Refer Slide Time: 37:03). Now here ideally what we would have wanted; we would have wanted a small value of delta. So if we observe a granular noise, to reduce the granular noise we should reduce delta n so we should multiply the present delta n by a factor which should be less than unity. Now again in this case how do we know that there is going to be a granular noise? Well, again by observing the $c(n)$. If you are having the $c(n)$ to be alternating between 1s and 0s that is a clear indication that there could be granular noise and when it is alternating between 1 and 0 that means to say that you have tracked down properly, you are close to the waveform. So if you are close to the waveform reduce delta n so that you reduce the noise also. So that should be the basic philosophy.

Again to what extent you can reduce? Can you make it zero? Again circuit-wise it should be difficult to make it exactly zero so we should say that there should be a lower limit to which the delta can be adjusted and let us say that it is the delta min and whereas in order to reduce the slope over load let us say that we could go up to delta max. So the value of delta should lie between delta min and delta max.

(Refer Slide Time: 440:42)



Therefore, now we can state our algorithm to be like this that we can say that Δn is equal to M times Δn minus 1 and the Δn should lie between Δn_{max} and Δn_{min} . So what is M ? M is nothing but a multiplying factor for the step size. Whatever the previous step size was that should be multiplied by M . sometimes we should multiply it by M greater than 1 and sometimes we should multiply by M less than 1. So, the algorithm for choosing the step size multiplier is for choosing the step size multiplier M is that we choose M as P which is a quantity greater than 1 if we have c of n is equal to c of n minus 1. This implies that the slope over load will be reduced because we are increasing by a factor which is greater than 1; the same thing, whatever I have talked of qualitatively now I am just putting it in a quantitative form by a factor P . So P greater than unity when we observe $c(n)$ is equal to $c(n)$ minus 1 so this implies reduced slope over load so it implies reduction in slope over load and M should be equal to Q which is less than unity if c of n is not equal to c of n minus 1 and in that case there is a reduction in granular noise. So you can have a value of P ; you can have a value of Q because ideally you can say that why should we fix two numbers P and Q why not dynamically adjust it.

Okay, implementation-wise it could be difficult, your algorithm also will be complex so instead in a very simplified form of implementation you can say that there are two fixed quantities P and Q , P is a number which is higher than 1 and Q is a number which is lower than 1 so whenever I need to increase my Δn I will increase by P and whenever I need to reduce the Δn I need to multiply it by a number which is less than 1 so multiply it by Q .

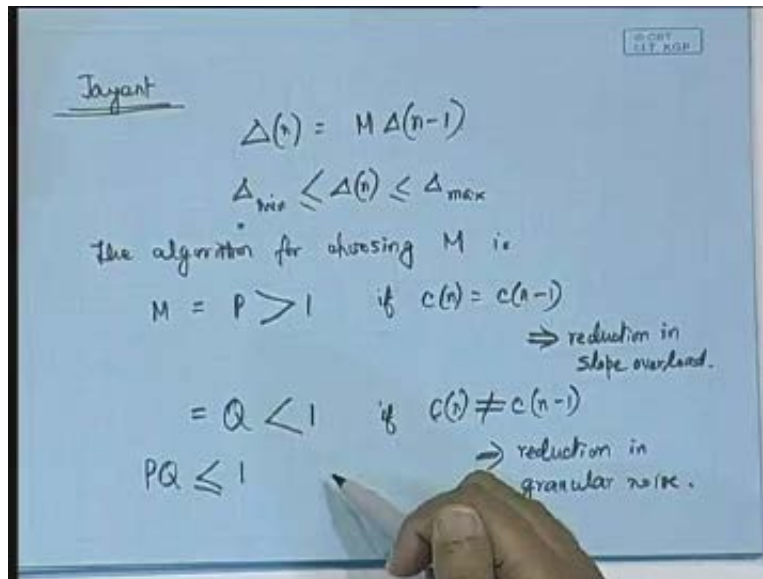
So P and Q actually, this kind of a very simple algorithm in fact was proposed by Jayant. This is known as Jayant's algorithm. In fact Jayant happens to be a pioneer in this 'Theories of Adaptive Quantization' and the 'Adaptive Delta Modulation' so based on Jayant's work, one can have this. And in fact Jayant also showed that, to have a stability, one should have the value of PQ P multiplied by Q should be less than or equal to 1. People sometimes implement with PQ equal to 1 also. So PQ equal to 1 means; a simple example that if you are taking P is equal to 2; so if you put the multiplying factor to be P is equal to 2 and if you take the so then correspondingly you have to take Q as half in order to maintain that PQ is equal to 1. You can choose equal to 1, you can choose slightly less than 1 also.

What are the questions; let us answer to some questions. [Conversation between Student and Professor – Not audible ((00:43:25 - 43:39 min))] Yes, we can say $c(n)$ is equal to $c(n)$ minus 1 if we are predicting predicting x of n plus 1. Anyway your observation is 100 percent correct that it should be used to predict the next one, yes. So we will be observing the previous and the previous..... absolutely.

Any other question? [Conversation between Student and Professor – Not audible ((00:44:13 min))] Yes, we can take but as I was mentioning that that will make the algorithm fairly complicated. I mean, if you are making P and Q to be continuously adaptive the thing is that, at least for the case of this, at least for the case of this feedback adaptation whenever we have an algorithm like this it looks difficult because what are the possibilities; either $c(n)$ and $c(n)$ minus 1 are the same or they are different so we just used two numbers over here that if they are same use P , if they are different use Q . But complication in algorithm should be possible; say if you try to monitor even several past c 's.

If you say that if $c(n)$ is equal to $c(n)$ minus 1 and is also equal to $c(n)$ minus 2 then you should increase the value of m , you should increase m by a further factor. In fact to make it very clear that there has been considerable research efforts; following Jayant's work there has been considerable research about what different adaptive mechanisms one can have so all these variations have been tried out and some are giving marginally better results but not very significantly. This this sort of a scheme is something which gives reasonably good results for all practical speech processing applications.

(Refer Slide Time: 445:25)



Any other questions?

Yes, [Conversation between Student and Professor – Not audible ((00:46:08 – 46:25 min))] The bit error rate, see the bit error rate; what happens here is it is like this that ADN is saving; ultimately ADN is saving the..... is going to have a faster adaptation and in that process one can introduce more error robustness or more error resiliency by trying to incorporate some extra bits because you are saving number of bits; you can use extra bits in order to have error protection.

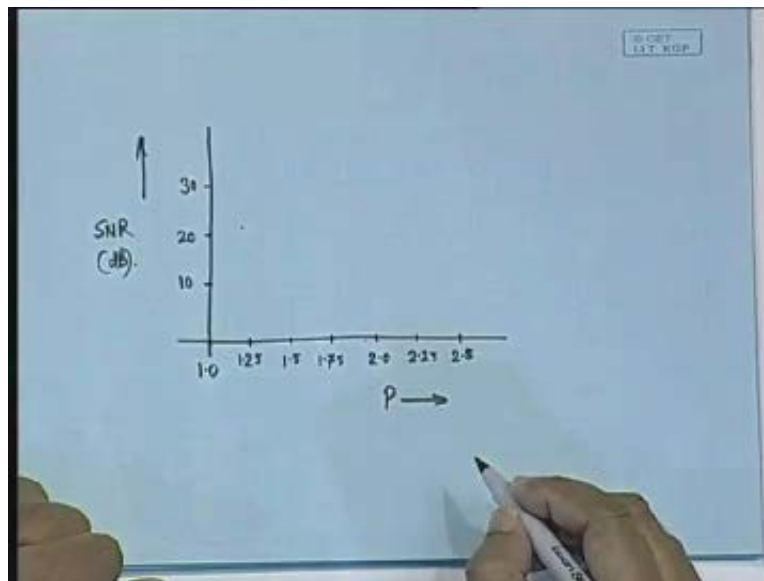
[Conversation between Student and Professor – Not audible ((00:47:17 min))] Right. So at this point of time, I do not find it very relevant pertaining to our discussion to address this aspect but we will be definitely discussing this aspect about that error robustness etc when we come to the discussion pertaining to that. So at this moment we are not considering the channel aspect. So let us consider the channel effects somewhat later. We will **we will** definitely do that.

Any other questions?

Now, just to show you the performance, let us show that what kind of performance curves one gets by varying this P and what kind of SNRs we obtain.

The characteristic is somewhat similar to this that on the x axis let us plot P; P being the multiplier, so P is the quantity which is greater than 1. So we start P is equal to 1 in this case and then say this is 1.25; this is 1.5, 1.75, 2, 2.25, 2.5 and on the SNR side let us say that this is 10, dB this is 20 dB, this is 30 dB so on this axis we are plotting the SNR in dB.

(Refer Slide Time: 49:11)



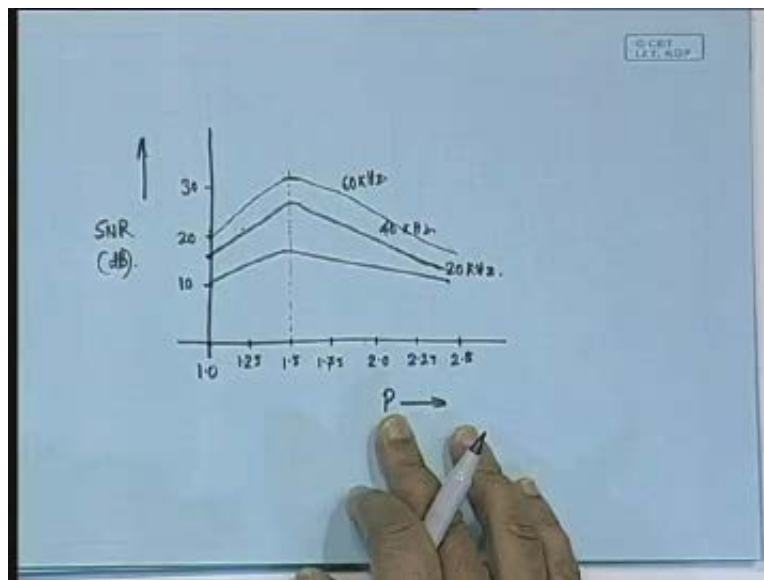
So what do we observe for the variation in SNR against the variation in **b**. So we find something like this that let us take three different sampling rates. So we first take a rate is equal to 20 kHz; so for 20 kHz one observes a plot as something like this (Refer Slide Time: 49:46). This is what we are going to have for sampling frequency 20 kHz and if we double the sampling rate let us say from 20 kHz over to 40 in that case the curve would look something like this. It will never intersect of course. May be that **after that it will suit** like this and then if we make the sampling rate three times as that of the original then one can have like this, so this curve we will be getting for a sampling rate of 60 kHz.

So what observation are we making?

When we have P equal to 1.5 roughly, through experimentation one observes that the signal to noise ratio reaches its peak at P equal to 1.5. This was observed for some typical speech signal.

So this can change from speaker to speaker and this can also change from one speech content to the other. This is only some kind of a rough idea, just to give you some rough idea.

(Refer Slide Time: 51:29)



Now why is it that the increase of P is leading to optimal SNR and then with further increase of P the SNR is dropping down?

Again you will see that when we have the value of P to be quite on the higher side in that case it may track down fast but in that process it will lead to granular noise initially. But okay, again granular noise can be reduced by quickly having a very low value of Q . because if you are taking say P value to be equal to 2 in that case Q value becomes equal to 0.5; say that if we are drawing this characteristic with PQ is equal to 1 in that case Q becomes equal to 0.5 so next time we will reduce it but initially we will be having a higher value of P and that would result in large amount of prediction error so even that prediction error also will contribute and try to lower down the signal to noise ratio. So there is a stage where we reach this.

Now look at the case for P is equal to 1. What is P is equal to 1? What is the interpretation of P is equal to 1? [Conversation between Student and Professor – Not audible ((00:52:55 min))]

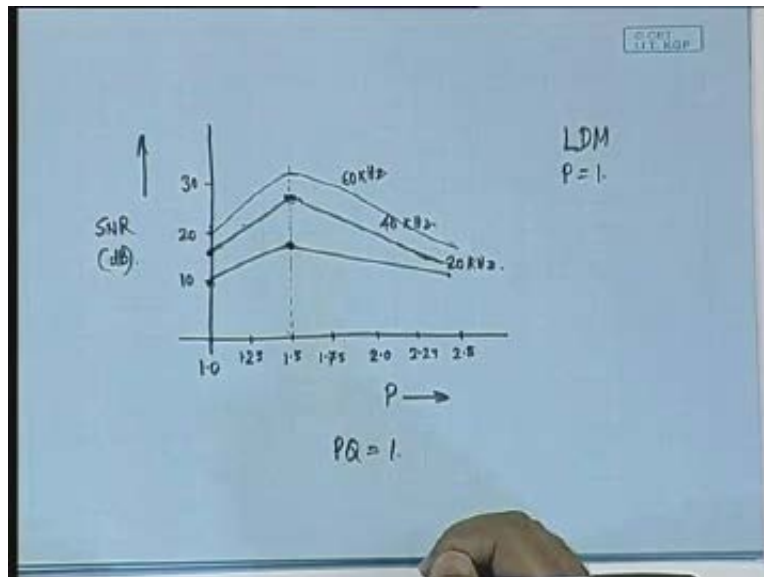
P is equal to 1 is a degenerated case of LDM, the linear delta modulator. The linear delta modulator is observed with P is equal to 1. Now what we observe in P is equal to 1 is that in this case you see that at 20 kHz, sampling rate it is 10 dB with 40 kHz sampling rate it is 16 dB because you know that when we increase the sampling rate by 2; whether we increase the bits per sample by 2 or whether we increase the sampling rate itself by 2 we will be improving the signal to noise ratio by around 6 dB. So if this is 10 this is equal to 16 and if it is increased from 40 kHz to 80 kHz we would have observed 22 but it is little lower than 22, it is 20 because it is not 80 but it is 60 kHz of sampling rate.

Now see, observe one more thing that when P is equal to 1.5 roughly you may find that this is equal to 14 dB whereas with 40 kHz sampling rate this is around 24 dB can you see that; so that means to say that here with P is equal to 1.5 doubling of sampling rate leads to an increased value of SNR. So here every bit increase results in 6 dB rise whereas here every bit increase results in 10 dB rise.

In fact, now I think I will be able to throw some better light to the question that was asked sometime back.

If with every bit increase we have or if with higher sampling rate we are going to have 10 dB of increase in that case definitely one can have a better error protection performance in the sense that just to maintain the same SNR we can have lesser number of bits and then one can use that for error protection. Anyway that aspect we will talk about later on. But it suffices to say at this point that we are going to have 10 dB of increase. This is something which is very striking.

(Refer Slide Time: 55:49)



Therefore, this talks about the adaptive delta modulator performance and there are other forms of algorithm but we just happen to present the one that was presented by Jayant .

Any other questions at this stage?

So let us now stop at this point and in the next lecture we are going to talk about the differential pulse code modulation which is an extension of this adaptive delta modulator or this delta modulator. We are going to have the DPCN and the adaptive DPCN and also the adaptive prediction aspect that will be covered in the next class. Thank you.