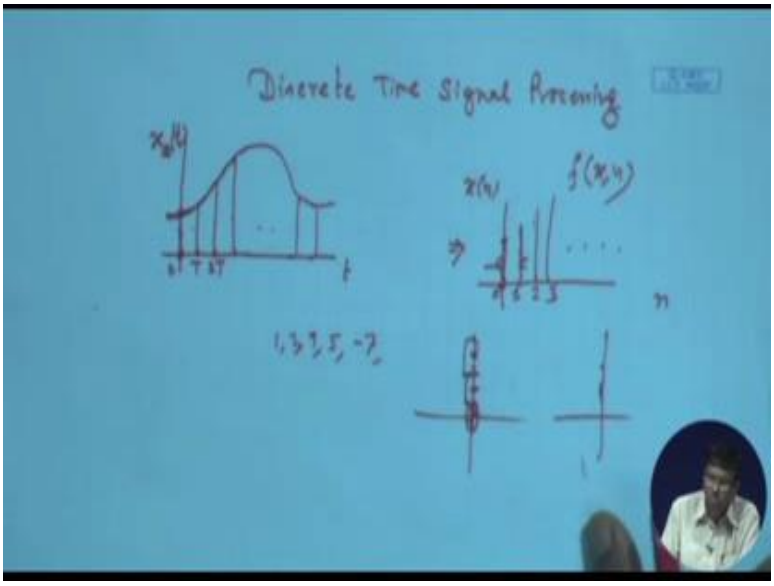
Discrete Time Signal Processing
Prof. Mrityunjoy Chakraborty
Department of Electronics and Electrical Communication Engineering
Indian Institute of Technology, Kharagpur

Lecture – 01
Introduction

So, we begin this course which is given the name Discrete Time Signal Processing.

(Refer Slide Time: 00:24)



It is a discrete time it means the signal normally we come across analog signals analog signals are actually wave forms, wave forms of time in general time it could be space also like if it is an image continuous image it is a function of 2 x's, x and y is a continuous function, that is an analog image. Similarly an analog wave form as a function of time could be like this, there are analog wave forms, and you can call it x subscript for analog as a function of time. So, the function of time t is a continuous variable time t are you plot it is called a analog signal it can come from say somebody's speech like I am saying something, if you would record the wave form as a function of time it will be like this it can get from any source from communication source, from various sources it just convert it into an wave form function of time.

So, for us signal means, analog signal means something which is a function of time of function of one variable either normally we deal with functions of time or in the case of images function of space, but they are at least function of 2 variables x axis, y axis.

Something like if x comma y, x is continuous in x y is continuous at each x y there is a value, pixel value - I mean intensity value and since x and y are continuous there will be a continuous image. There is a 2 dimensional signal, 1 dimensional signal are typically function of time analog (Refer Time: 01:20) these are analog.

But if I discretize the time that is I will not observe it at all points I will observe it at uniformly separated points may be 0 then t then 2 t what capital T is called the period, sampling period - that I am taking a sample here, next sample here, next sample here, next sample here, dot, dot, dot, dot. So, the sample values, when I consider the sample values this axis is no longer time, we in fact, throw away the notion of time from here we only bother about index, this is a 0-th sample so this is n equal to 0, this is the first sample n equal to 1.

So, this gap has nothing to do with capital T you can make it very narrow very wide it gives nothing its only 0 1 2 3, what matters is for this first sample, which is second sample, which is third sample, that is 0-th sample, first sample, second sample, third sample like that. That means, this x axis is discrete and that is why it is discrete time that is if the original signal came from a function of continuous time then if you discretize the axis you just get points 0 1 2 3 and corresponding sample where as you plot you will get a sequence. This sequence is a discrete time sequence we just call it a sequence and I repeat again there is no notion of time here this a n has got no time it is just an integer diamonds are less unit less does n equal to 0, n equal to 1, n equal to 2, n equal to 3. So, who is 0th, who is first, who is before whom, who is after whom, these are the things matters and then I will call it x n this will be a sequence we also call it discrete time signal.

Now, advantage of this thing is this representation is that if you throw either notion of time you can take care of any sequence of numbers, because may be a sequence is obtained just manually not by certainly by sampling an analog signal. The sequence of numbers, them also I can process here like I mean if you get a sequence of numbers 1 2 9 5 minus 7 dot, dot, dot you automatically generate or may be your computer is generating sequence of numbers. I just plot them 1 after another 0th, first second, I put a 0 value I mean I set one origin index origin for 1 of them 0 and then 1 2 3 and put them accordingly. So, what I get is a sequence so, but this time it is not obtained by sampling any analog signal. So, this is more general if I throw a either notion of time just begin index n and call it 0 or 1 or 2 or three I can take care of not only sequences obtained by sampling an analog signal, but any sequence manually generated sequence of numbers, sequence generated by computer, data, generated by computer all, that is why here after discretizing we do not write in terms of time we just write in terms of these. Now the question is why I should suppose I am considering only this analog signal and a sequence obtained by sampling this like this as I discussed.

Suppose now why should we go for this discrete time signal processing that is processing this rather than the analog wave form answer is analog wave form processing means actually what is processing? Processing means I should be able to do various operations as many operations as possible on that input signal. Now suppose these are analog signal I want to multiply, I want to divide I mean by some number some fraction; this is possible in analog domain just take a registering divider r 1 r 2 set the values of r 1 r 2 so that and we get a fraction of that is very easily durable. If you want to multiply by a number this signal in my processing involves multiplying a number greater than one; that means, amplification; that is also very easy. We pass it through a common emitter amplifier or common based amplifier or (Refer Time: 06:26) amplifier whatever adjust again. So, you can do this amplification k. So, this operation where you have to multiply this by a number greater than 1 in analog domain we can do that, no problem, using amplifiers.

Then suppose you want to add 2 such wave forms x 1 x 2, two such wave forms you want to add that is also possible, because what you do there you use an (Refer Time: 06:53) based (Refer Time: 06:54) and thus give the 2 signals you get an output if it is inverted inside you pass it through another inverting amplifier and all that. So, adder is very must there, so I can add I can carry out addition operation and thereby I can carry out subtraction operation also not a problem. Multiplying 2 signals in analog domain that is also possible, there are multipliers and by analog multipliers which can take 2 signals fourth quadrant multipliers and get you out.

Connecting log of a signal that every value of this wave forms when I take the log of it, can I do a logarithmic calculation? Yes, there is something called log amplifier using (Refer Time: 07:34). So, you can carry out log you can carry out anti log. So, these all these are operations and some more operations are possible but point is number of such operations is very limited you cannot go beyond a few for instance if you have to take a say tenth root of these or even square root cube root, can I do that? Not easily, any other operation if I take this value if I have to carry out trigonometric operation. So, every value I have to take it sin or cos, how much? This is very difficult in analog domain you cannot do.

But if I convert them into a sequence, so just sequence of pure numbers and give these numbers to a computer then in a computer I can run an algorithm on them on the sequence and through this algorithm I can do any kind of operation all operations are possible. So, that is digital processing, processing this sampled values. But the moment I make them available in this form and give it to a computer, computer can work only on numbers discrete elements, computer can work on continuous elements. So, discrete numbers 1 after another. So, my algorithm will work out then and give you any operation you want.

So, the range of operation becomes inmates, anything is possible if you go for digital processing. But only thing is you can ask me a question that look here I am not taking the entire wave form, I am not taking the entire plot, I am probably taking sum of some values 1 value, another value, another value, another value, I am leaving out this intermediate wave form part, intermediate values, then am I not losing something, am I losing some important information, because I was crazy to get into some kind of digital processing. Answer is no, this is the beauty of digital signal processing. That if suppose this analog function has got a finite bandwidth then it can be shown this is the fundamental theory I mean DSP verifies, take the samples close enough that is make the sampling period less than equal to some value or sampling frequency greater than equal to some value, there the samples are good enough to carry all the information of the envelope even if the intermediate part is missing.

But (Refer Time: 10:05) actually what is called nyquist sampling rate which I will cover later, that is there is a fundamental limit for band limited signals that limit is called nyquist rate. If you sample this analog signal at a frequency higher than nyquist frequency; that means, sampling period is shorter than that more dates, more close then those samples carry all the information of the analog envelope you do not lose anything that is fundamental. That gives you a permit; a license to replace this analog wave form by a series of discrete samples and which can be easily processed by a computer or by a dedicated digital hardware and any operation can be done.

As you understand in digital processing number of operations can goes to infinity, you can do it for anything you want because it will be basically implementing an algorithm. Algorithm means like in computer where you can do anything the same thing you can work out that, but algorithm can depend on continuous stuff that is why discrete has the big advantage. This is the main motivation for going into discrete time signal processing or rather digital signal, discrete time signal processing right. Now there is one more reason while you will go for this, but before that I am coming to another point - we will coming across 2 phrases: one is discrete time signal processing and sometimes digital signal processing question is are these two same. Actually here in this wave form this axis is discrete 0 1 2 3, but this amplitude axis y axis it is not discrete the sample can take any value.

But why do you give it to computer every sample we pass it through a quantizer this analog to digital converter and then quantizing to a either 16 b you know I mean that will give an 16 bit output or the (Refer Time: 11:59) I mean or 24 bit output or 12 bit output or 32 bit output. So, basically this can take some quantized values only. So, that time this x is also get discretized that is if the y axis is this anything falling in this range may be quantized to will be taken to be 0 value, anything falling in this range may be take into these value, anything falling in this will taken to these value and then they are (Refer Time: 12:29) by number of domain some binary (Refer Time: 12:31). So, that is quantization you are familiar with the additional quantization. So, that time y axis is also discrete. So, the samples actually take discrete values.

So, in pure digital processing this x is also discretized because they (Refer Time: 12:45) take discrete values because after all they are depended by fixed numbers, so 8 bit numbers or 2 to the power 8, 64 possibilities only; so 64 level we will assign. Around the level there will be a band anything falling within the band will be equated to that level that another level around that there is a band anything falling within this band will be equated to this level so on and so forth. That is why this x is also discretized.

This will be a pure digital signal where both x axis that is this axis of n at the y axis they are discretized, but the problem is if you really start using y axis to be discretized also, x axis to be discretized also then analysis other things become very difficult that is why we assume that number of bits, after quantization number of bits is very large. So, therefore, this bands are very narrow, every levels are very close to each other and bands that instead of being 2 levels being here they can be very close to each other almost adjacent to each other and therefore, they are almost like continuous that is if you assume that number of bits is large the gap between 2 levels discrete level will be very small. So, you can assume the (Refer Time: 14:04) axis to be almost continuous and then this becomes just a discrete time signal - continuous amplitude, but discrete time signal.

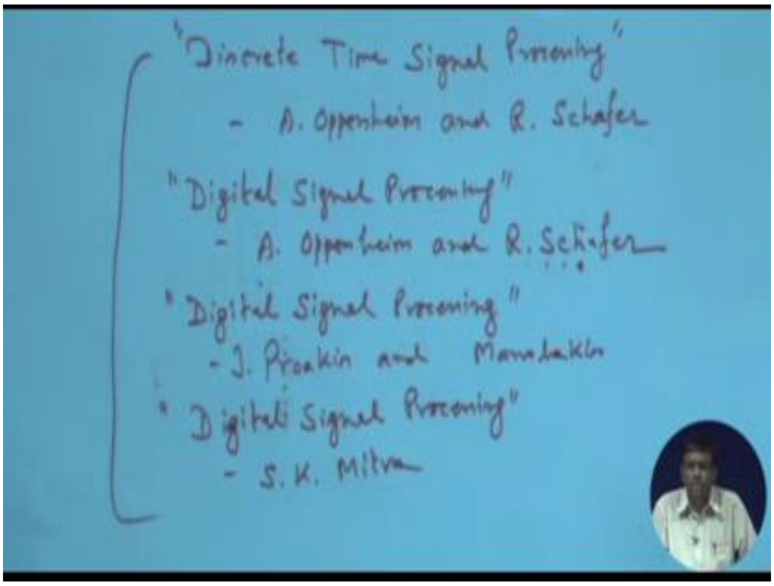
We did this assumption in doing all analysis and design and other things because if you want to carry both the discrete in time, discrete in amplitude then it becomes difficult. But there again the question comes that I am making an assumption there is an approximation, after all y axis is also discretized, but I am making it you know I am assuming it to be continuous under the assumption number of bit is large. But have I not introduced them either as (Refer Time: 14:37), after making this assumption that y axis is continuous and then going (Refer Time: 14:43) with my design and other things then we get (Refer Time: 14:48) take a look and try to estimate how much error in my design has got it because of this assumption that y axis actually is discrete, but I have assumed to be continuous because the levels are very close to each other. So, that analysis is called finite (Refer Time: 15:03) analysis or you know finite (Refer Time: 15:05) analysis that we estimate, because of this is the approximation how much error going (Refer Time: 15:10).

So, again by that we estimate whether that error that is OK or not, but before that till design level we assumed it to be continuous, so this is the approach; why do you go for designing the analysis everything take the y axis to be continuous, x axis to be discrete, it becomes a discrete time only signal - discrete time signal then after doing your after getting your design and all that then carry out some further error analysis where you try to find out the effect of this approximation on the y axis that y axis actually is discrete, but you assume to be continuous. So, what kind of error, how much error does it give raise to, how to keep that error within limit within bound making small and all that, that is a separate chapter that is called finite for length analysis.

So, this is the difference between discrete time signal and digital signal. Now this one more region why this digital signal processing or discrete type signal procedure here became very popular one is of course, the digital domain you can do any processing another is last 30 years or so we have seen tremendous growth in digital VLSI not. So, much in analog VLSI, but digital VLSI and that is why (Refer Time: 16:22) is shifted from processing in analog domain to digital domain because in digital domain you can do millions of operations you know so many gates and others can be (Refer Time: 16:31) and achieved can be made.

So, much of operation can be put in 1 IC. So, that is why (Refer Time: 16:40) is shifted to this. These are the 2 main motivations, I suggest some books also though I suggest I tell you that be careful always follow my lectures because I normally do not follow any book, I teach from my understanding and here and there you will always find some new concepts which I am not giving any book. So, that should be of value addition you should be careful you should take note of this. So, the books are this by Alan Oppenheim and Ron Schafer I do not remember the publisher (Refer Time: 17:43) these all.

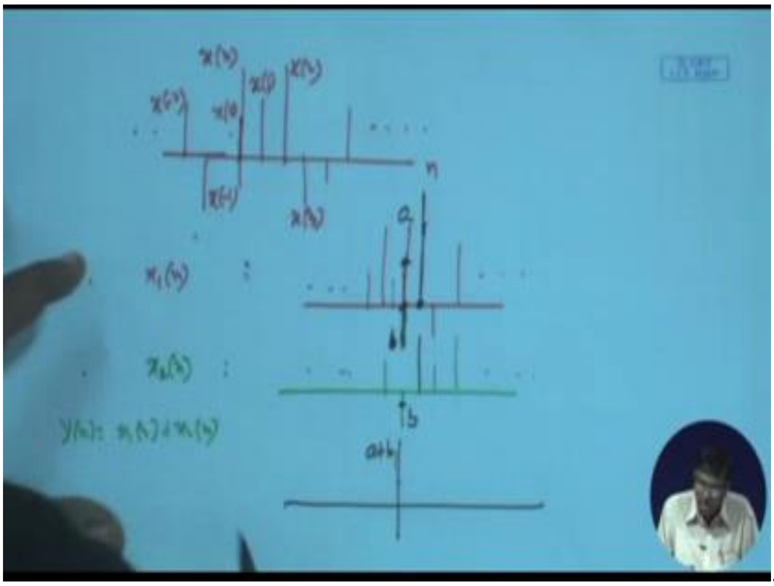
(Refer Slide Time: 17:11)



But this 2 authors you know way back in meet 70s when the DSP topic just came up they wrote a fantastic book that is just digital signal processing, I believe Oppenheim Schafer and there is another per third author that I do not remember, but this is a famous book, but I personally like this book. By the same authors this is a fantastic book much shorter must thinner than that, but this is a fantastic book this is more elaborate and sometimes unnecessarily elaborate unnecessarily in the in buying opinion of course. Then there is a book DSP I do not remember there is a title by John Proakin and Manolakis full name I do not remember its title is Manolakin it is a (Refer Time: 18:58) book, a very good book. Then there is a DSP book by S K Mitra this is also very good. This four book

(Refer Time: 19:21) they are pretty good they are very good at having very good examples and everything, but please follow my lectures because my lectures or for my lectures I do not follow any of them I teach from my own understanding and here and there I give inputs which are not easily found in book and you have to be careful for that.

(Refer Slide Time: 19:47)



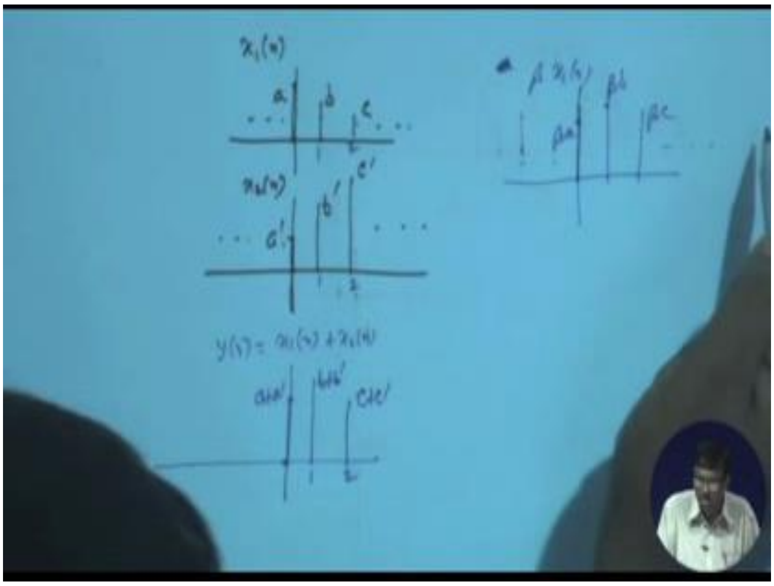
So, with this background I start this course now. So, a starting point as I told you is a sequence, sequence of numbers though this axis is n there is no concept of time here thus compute I mean for processing we take only the numbers even if the analog signal is quantized, even if we started by sampling an analog signal we take every sample by analog to digital converter then digitize the sample see the sample height that the number we process the numbers this sample so much value, next sample so much value, next sample those values those numbers numerical values one after other which is sequence of numerical values then this values only I process that is 1 sample value so much, another sample value, so much another, so much another, so much another, so much another only the values I process, sequence, not so. This time of separation between these 2 samples is not my concern here.

The another digit is that whatever algorithm I want to propose out of them that should be independent of this sampling period that is why we only take the numbers the sample heights and they one after another they come and they form a sequence or denoted by say x n, h n, y n, v n. So, this is my x 0 at 0-th point this is my x 1, this is my x 2, this is my x

3 and dot, dot, dot; I can have on this side also this is my x minus 1, this is my x minus 2 because this is 0, this is 1, this is 2, this is minus 1, minus 2, dot, dot, dot these are typical sequence, sequence of numbers, it could have come by sampling an analog signal. So, and then I just take the sample values the train of samples one after another I do not care for the time of separation there is a sequence who is after whom, who is before whom, who is 0th, who is first, who is second like that or this could be generated just as a sequence of numbers manually by myself or by a computer which is generating data I think like that, this is a sequence.

Sequence is there are some properties first if I give you 1 sequence x 1 n something like this, something like this dot, dot, dot and x 2 n, x 2 n, dot, dot, dot, dot, dot. When I add x 1 n plus x 2 n by this I mean another sequence y n, where y n could be like this 0-th sample. So, this is a 0-th sample and 0-th sample this 2 will be added.

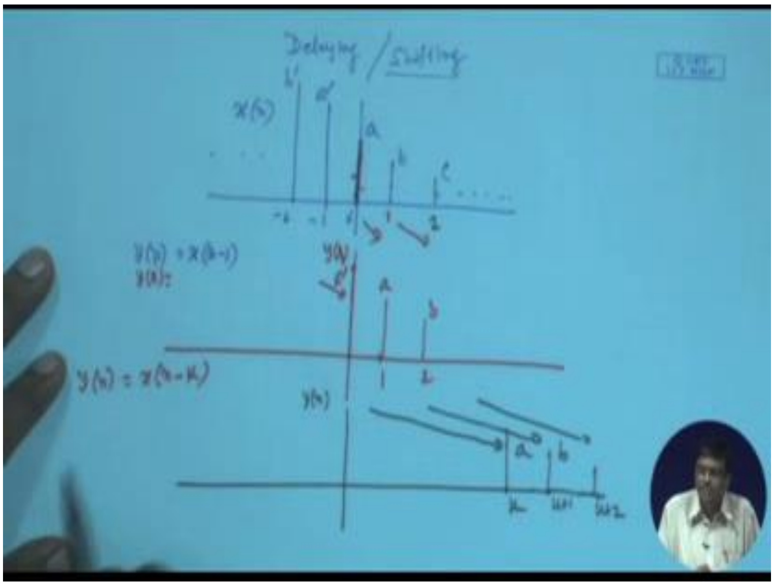
(Refer Slide Time: 23:47)



So, new 0-th sample sorry, this is 0-th sample and this is 0-th sample, this 2 will be added new 0-th sample. So, if it is a, if it is b, if it is b mu value will be a plus b. So, sample wise addition let me draw a better figure this is what clumsy, if it is a sequence like this say a, b, c, dot, dot, dot, dot and I call it x 1 n and another sequence say a prime, b prime, c prime, dot, dot, dot, dot and I call it x 2 n then y n equal to x 1 n plus x 2 n this will be. Thus we have to add sample wise 0-th sample of this fellow 0-th sample of this fellow they have to be added that will be new 0-th sample. So, this will be a plus a prime, a plus a prime together is that you get the new 0-th sample at 0-th point, similarly point number 1 at index 1 b and b prime, so we add at index 1 there will be new value b plus b prime. At index 2 again you have got c c prime, so you add the 2 values you will get c plus c prime at index 2 like that. So, at every index take the sample here, take the sample here at the 2 that will be the new samples at the same index, this is very simple this is called addition.

Then multiplying a sequence say x 1 n you have to multiply by say beta into x 1 n is nothing every sample will be multiplied by beta. So, 0-th sample earlier was a now it will be beta a, next sample was b so it will be beta b, next sample was c now it will be beta c dot, dot, dot, dot. So, scalar multiplication of every sample means multiplying by every sample that is beta times a sequence means in a scalar times a sequence means take every sample multiplied by the scalar, each of them multiplied by the scalar. Then comes a very important thing Shifting, Delaying.

(Refer Slide Time: 26:02)

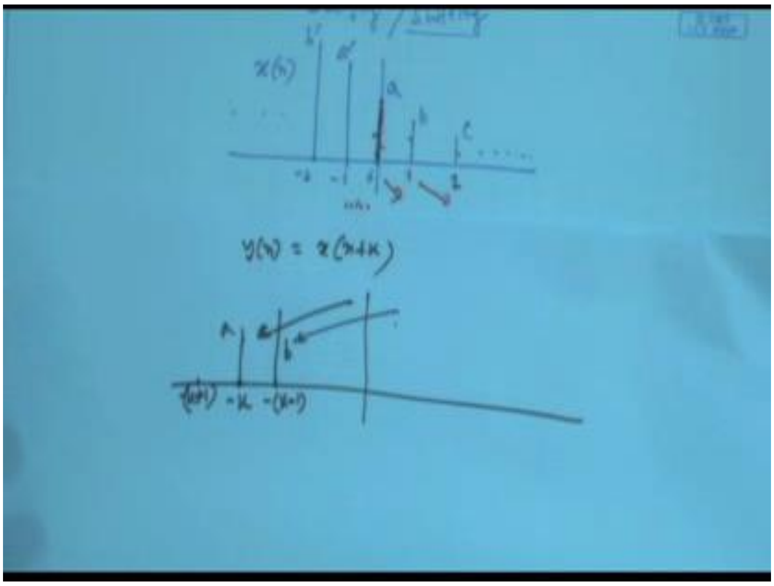


First, what is shifting? Suppose you have been given x of n, it has got a value at 0-th point may be a, then b, then c, dot, dot, dot, dot at minus 1 it has got say may be a prime, at minus 2 it has got say b prime dot, dot, dot, dot suppose this is a sequence. If I have y n is equal to x n minus 1 its n minus 1, if n is an integer n minus x is a function of some integer right, x 0 has a value, x 1 has a value, x 2 has a value, everywhere given. So, if n is an integer n minus 1 also is an integer. So, x of that also has a value from here only I can pick up.

But how will this look like then. So, for that what is y 1 suppose I said example you take y 1 y 1 will be x of 1 minus 1 0. So, at index 1 there is why this I am plotting y n. So, at y 1 value will be x 0. So, this fellow will move here. So, it is moving to the right. What is y 2? It will be x 2 minus 1 x 1. So, this was x 1 this will now move here y 2. So, this is moving to the right, this is moving to the right. What will be y 0? y 0 is x minus 1 that is this - a prime, a prime is also moving from left. So, it is shifting, this sequence is just shifted to the right by 1; if it is n minus 1. If it is y n is equal to x n minus say k where k is an integer how will it look like? Start with y n equal to k, n equal to k means k minus k 0. So, at n equal to k, n equal to k y n that is y k will be x 0. So, x 0 from here into this sample value will move here.

So, it is making a jump from 0-th index to k-th index, it is then y k plus 1 if I put k plus 1 it will be k plus 1 minus k. So, on x 1 from location 1 will move to location k plus 1; y k plus 2 y k plus 2 will be x 2. So, this guy you see will now move here. So, you see they are jumping from 0-th to k-th, from 1-th to k plus 1-th, from 2-th to k plus 2-th. So, this sequence is shifted to the right by k if I write x n minus k.

(Refer Slide Time: 29:54)



On the other hand, if I have y n is equal to x n plus k then what will happen? We will start at n equal to minus k, y minus k will be x minus k plus k that is 0; x of 0 what was x of 0 a. So, this from here will move here to the left side. So, it is moving to the left at minus k plus 1 minus k plus 1 it is further minus y of minus of k plus 1 if you put here you get x minus 1; sorry, same from come here first I will go to their later - minus of k 1 less k minus 1 say it was minus k to the right will be 1 less minus of k minus 1, y of minus of k minus 1. If you put that here at this point you will see it will become x 1, this guy he will come here.

So, as is moving to the left by k, b is moving to the left by k. So, even that way you can verify every circle is moving to the left by k that is sequence is shifted to the left we call it advanced by k. So, if k is negative sequence will be shifted to the right by k, if k is positive it will be shifted to the left by k. The shift has something to do with delaying, if it is a real time signal that is if this sample points corresponds to time axis points then it amounts to delaying or advancing that I am coming to in the next module.

Thank you.