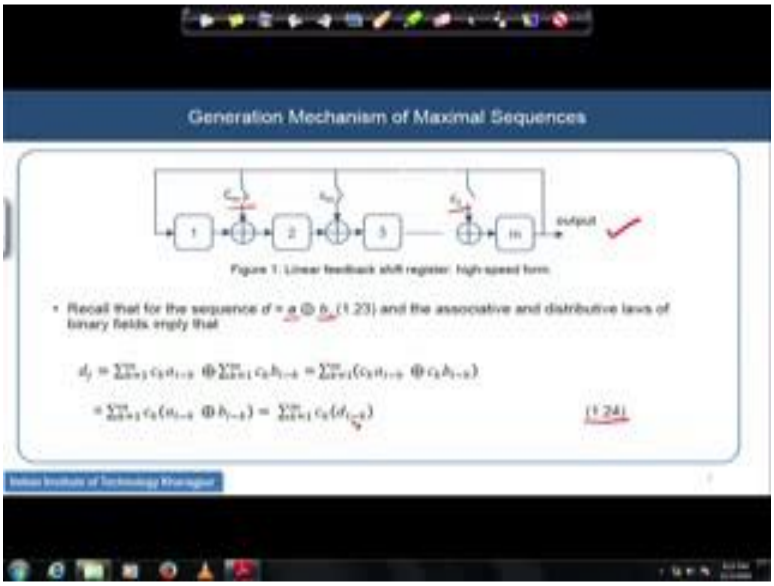
Spread Spectrum Communications and Jamming
Prof. Debabrati Sen
G S Sanyal School of Telecommunications
Indian Institute of Technology, Kharagpur

Lecture - 11
Properties of Spread Spectrum Sequences

Hello students, today we will start learning the properties of the spread spectrum sequences. In the last module, we have learnt the generation mechanism of a spread spectrum sequence based on which is a ML sequence based the linear feedback shift register architecture. We will continue our discussion today following the same architecture; and we will try to see the properties of the spread spectrum sequences generated by this kind of the linear feedback shift architecture, shift register architecture.

(Refer Slide Time: 01:01)



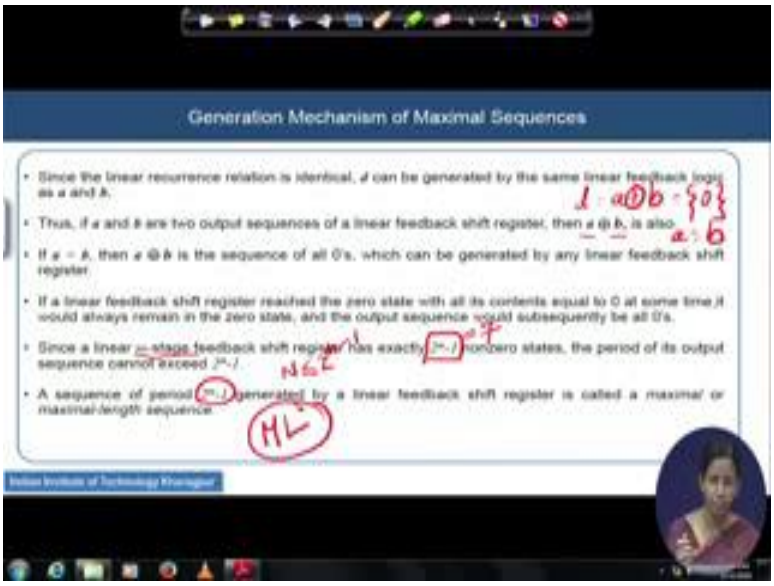
Hope you will remember that we were discussing the structure to be of high-speed form. Here I wish to mention one thing that we initially constructed generalized linear feedback shift register where there was a feedback from the last stage which is the mth stage to the initial stage, stage number 1, and all the feedback path I mean the XOR gates where on the feedback path itself. In the high-speed form we shifted all those XOR gates of the linear logic gates. We and came in your in the forward path which is coming from the stage number one to stage number m in between. And c case where the switches were allowing the feedback of a particular stage to join to give actually tracks attributing to the feedback logic.

Please note that when there was a generalized architecture, the notation of all the switches was a c 1 on the left most side; and a c m minus 1 on the right most side. Once we have shifted to as the high-speed form then nomenclature for those switches should be changed, which user not done in the last module, please correct it. Here I have corrected it already; the nomenclature of the switch should be c 1 on the right most and then the c m minus 1 on the leftmost. All the equation that has been derived in till equation 1.23 till the last module will follow the architecture that is given in this figure that is with the switch nomenclature of c 1 on the rightmost and c m minus 1 on the leftmost.

So, we also recall the fact that we have discussed the point, where the, if a is a shift register generated sequence, and b is another sequence generated by the shift register, maybe the initial condition for generation of a and generation of b are not same. And if we do the XOR operation bit by bit for all the sequence for the sequence of a all the bits for the sequence of a with the all the bits of the sequence of b regenerated another sequence d. We also proved in the last module that the any bit in that generated new sequence d, which can be written in the form of c k a i minus k where i's is the ith pulse and k was actually the number of the pulses or the time instant that you are going back from the ith instant and d j is the value stored in the jth stage of the new sequence d. That will be given by the XOR operation of c k a i minus k with c k d minus k.

After some mathematical simple processing, we ended up with the fact that c d j can be actually written in the form of c k d i minus k, which proved fact that any sequence generator by the linear feedback shift register, and two sequences such two sequences generated by the linear feedback shift register and XOR on both of them. If you are generating a new sequence each and every bit of that new sequence will keep a linear recurrence will show a linear recurrence with the earlier stage values. So, this is the way, but you have the jth stage value of the newly generated sequence d is basically or actually is constructed by a linear recurrence of the stage values which are stored from the i minus kth instant multiplied by their corresponding weight values or the switch values. This was an important observation with which we ended in the last module.

(Refer Slide Time: 05:41)



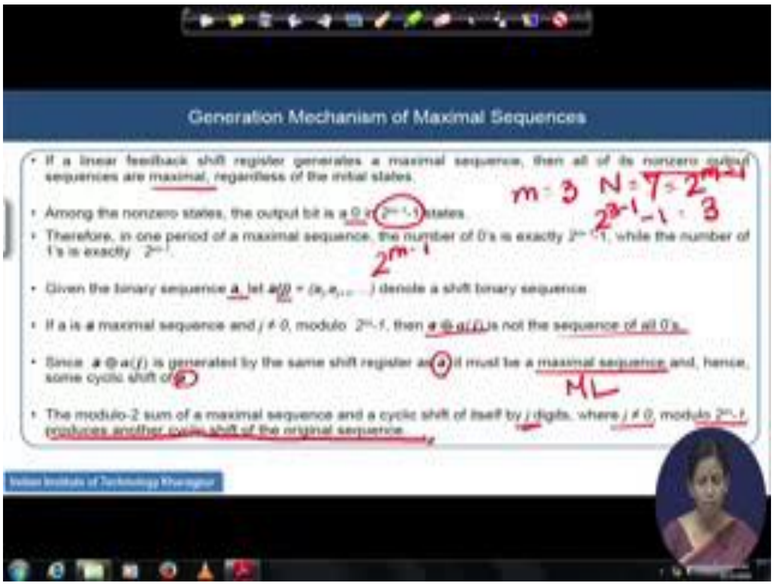
In this case, we will continue from here to understand the further properties of the sequences that we are getting from this kind of the sequences. Apart from the fact that d can be generated by the same linear feedback logic as a and b, and then we can also think that the linear feedback resistor can also generate d, if a is sequence generated by the shift register the same construction can also generate. The construction that is generating a and b that can also construct a new sequence, which is equal to d; and d is nothing but a XOR b. If my a and b, these two sequences are found to be exactly same then obviously, this XOR operation will give you all 0 sequence. And this all 0 sequence they are then actually d which is equal to a XOR b will be exactly equal to all 0's sequence because a is equal to b. And remember such a sequence can be generated by any kind of the linear feedback shift register.

And if you see that all the stages of a linear feedback shift register contains equal to 0 at some points of the time, then this will remain in the stage of 0 and no more output sequences could be obtained from such kind of the architecture. And hence initialization of a shift register sequence of an architecture of a linear feedback shift register is very important, and never initialize a linear feedback shift register with all 0 values.

Since, we are discussing about up in stage feedback shift register and we have seen in the last module taking an example of m is equal to 3 that we could generate exactly 2 to the power m minus 1. That means, in our case it was the 7 nonzero states we could generate 7 nonzero steps, and as it was generated by that hence the period for this output sequence can never be greater than 2 to the power m minus 1. And this rule holds good for all kind of the linear feedback shift register of stage value equal to small m. And if you see that for a typical architecture, you are getting typical length or the period is equal to 2 to the power m minus 1; always the structure is inheriting to 2 to the power m minus 1 then we call that the generator sequence is a maximum length sequence.

Remember I am repeating the feedback shift register architecture can give you less than equal to 2 to the power m minus 1 number of the nonzero states hence we are telling that the period capital N, if capital N is denoted for the period m will be always less than equal to 2 to the power m minus 1. And if by chance you see that N is equal to exactly 2 to the power m minus 1, the stage value is such that the period is equal to 2 to the power m minus 1 then this kind of the sequence will be called maximum length sequence or ML sequence. Very important kind of the sequence in the field of spread spectrum communication because this is the, we will see later on that this is the mother of the sequence generation of any kind that we can think day-to-day.

(Refer Slide Time: 09:29)



A linear feedback shift register sequence that is generating a maximum length sequence if it is generating at all then all of its nonzero output sequence will be maximal and regardless what we were with initial steps. So, you keep on changing the initial state all the nonzero states, you will see it is the maximal one. And among the nonzero states, if you count, you will see that the number of zeros that you are getting exactly the number of the zeros will be 2 to the power m minus 1 minus 1.

Please go back and check the last module where we took the example of small m is equal to 3, and we generated a sequence where n is equal to 7 and which really actually signifies equal to 2 to the power m minus 1 status. And hence we can declare the sequence that we generated there was a ML sequence. Count the number of the zeros you are getting there we will see you are getting exactly 2 to the power 3 minus 1, minus 1 is equal to just 3 number of the zeros in the generator sequence. So, it holds good. It proves that for an ML sequence if the number of the nonzero states, it will get the number of zeros exactly equal to 2 to the power m minus 1 minus 1. And hence the number of the one you will get how many you will get 2 to the power m minus 1.

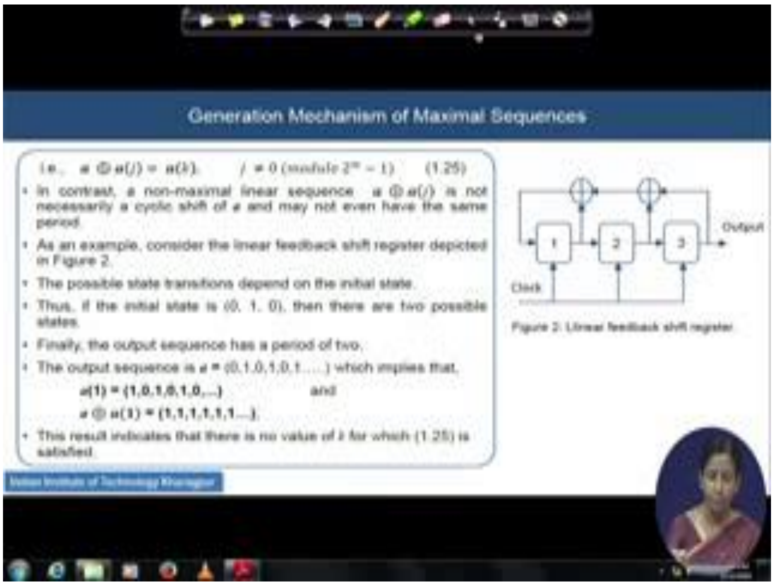
And now with the understanding of this fundamental property of ML sequence let us see what will happen if we are have having a maximum length sequence and we have shifted value of the sequence. Let us assume that we have a shift of an amount j. So, a is generated by a linear feedback shift register, a binary sequence and it is an ML sequence also. And then we have generated another sequence a j by giving a shift of an amount of j to a.

So, a j is a shifted version of a. And the shift is such that the shift is not equal to 0 and we have a modulo 2 to the power m minus 1 value of the j is such that. And then if I do the XOR operation of a with a j then you will definite be sure that there would not be also sequences it will get you will get some sequence again it is not all 0's sequence. And since a XOR a j it is generated by the same shift register where from a was generated, so definitely that a XOR a j will be again an ML sequence. So, this is the very unique property of ML sequence.

So, multiple ML sequences we can generate by giving a shift of the original sequence and doing the XOR operation in between them. And hence we are giving a cyclic shift of a. It is basically a with the cyclic shift of a we are XORing. And if we go ahead with the modulated sum of this ML sequence and the cyclic shift of itself owns sequence by some digit j, where j 0 is not equal to 0 and modulo 2 m minus 1.

So, the conclusion makes that it will always produce another cyclic shift of the original sequence only. So, thing is that three facts one is you have a ML sequence generator from a linear feedback shift register a, you have shifted it by an amount of the j, but j is not equal to 0, and j is 2 to the power m minus 1 having modulo operation 2 to the power m minus 1. And if you are generating another sequence by doing XOR operation by bit by bit of a with a j, and we will be ending up with another ML sequence only. And that ML sequence that now you have generated by operation of a XOR a j luckily you will see that it is actually some cyclic shift version of a only.

(Refer Slide Time: 13:54)



And that is if I now write this in a mathematical form, how will it look like. I have taken a, I have shifted a by amount of j, I am doing the XOR operation I am ending up with an another k. Finally, this k is the amount of the cyclic sheet of a it is such that a, a j, a k all are the ML sequences. But it does not mean that if you are having a non maximal linear sequence, and then if you do this kind of the operation then it will be a cyclic shift of a that is not possible and it would not have the same period of a also.

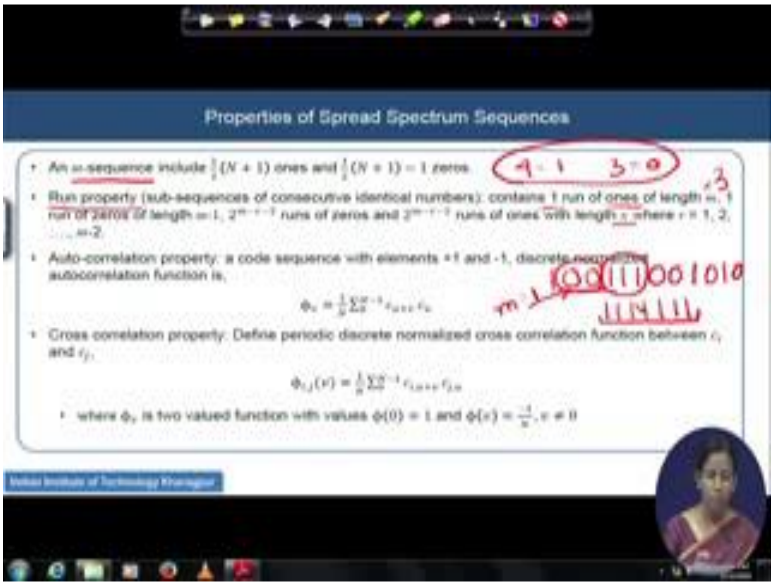
We can take an example to prove it suppose see this linear feedback shift register architecture, where we have three stages unlike the first example why we took the feedback from the stage number two, stage number three and we did the XOR operation here. We are having two XORs in the feedback path where the output from the first stage is also taking part in the feedback logic.

We can see actually if you develop the table of generation the way we did in the last module for the other example if you generate the sequence that the output you will see that there are only two stages possible. You are getting any initial state value you start with you will be ending up with only a period of 2, where the output sequence one output sequence we will get like this a is equal to 0 1 0 1 0 1 going on. And if you give a shift of one of this sequence you will get a another sequence 1 0 1 0 1 0. Now, if I try to do the

XOR between these two you will end with all one values, which is never a shifted some cyclic shift version of the original sequence. So, it means actually there is no such value of the k exist for this example for which this equation holds good.

So, fundamental learning is if you are not starting with a maximal linear sequence, you will never end up with another maximum length sequence by this operation governed by the equation 1.25, so that proves that it is a strictly a unique property of ML sequence. But it does not hold it at all for any other kind of the non-maximal length sequences.

(Refer Slide Time: 16:25)



Now, coming back to the rest of the properties of this ML sequence, if you have generated an m-sequence, the ML sequence is also called as m-sequence for easy understanding. We will be calling it as m-sequence throughout the course. And m-sequence you will always find that it will include N plus 1 by 2 number of ones and N plus 2 by 2 minus 1 number of zeros.

For example, where actually if you take the example of the last module where small m was equal to 4, then you will see that we had four number of ones and we had three number of zeros. So, our period was seven there remember, so that 7 plus 1 by 2, it was generating the four ones; and 7 plus 1 by 2 four minus 1 we had three zeros. So, four ones and three zeros we got in the sequence that we have discussed in the last module and total period was seven. So, it also proves once again the sequence that we generated there is an ML sequence or m-sequence. This property holds good for any kind of the m-sequences. You keep on changing the initial value I advise you to keep on changing the initial value on that architecture given in the last module, and check all this property is that today we are discussing. To prove that really it for the m-sequence, we will always end up with N plus 1 by 2 number of ones, and N plus 1 by 2 minus 1 number of zeros.

Next important property is the run property. What is run, run is how many number of the subsequences of the consecutive identical numbers you are getting. For example, you are supposed generating a very long length ML sequence or m-sequence; and such a way that the sequence is coming like this, like this. Then run is by run we say that how many consecutive same bits are coming. For example, for run property, if it is an ML sequence, it will contain at least one run of ones of length small m.

What is the meaning if I am having two numbers of the stages; in the whole generative sequence you will get at least one run constituting of value equal to one whose length will be equal to 3. So, you will get always a situation where consecutively three ones will appear; if your m values changes to 7, then in the generative sequence, you will get at least one occurrence of seven ones coming parallely coming at a stretch that is standard. You check it, you can try to prove it with the value of initial situation and different value of the small m values and you will realize.

Then for the zeros there is another rule at least one of zeros you will find. This is a run of one you will find at least one run of zeros for length m minus 1. So, if I am having n is equal 3, so I will get at least one run of zeros where the 0 will be m minus 1 means equal to 2 that will appear simultaneously is consecutive of occurrence actually. You should not count suppose 0 here and then another 0 here consecutively they should appear. And and like this actually we will find that 2 to the power m r minus 2 runs of zeros and r minus 2 to the power m minus r minus 2 runs of ones with length equal to smaller r. So, length now if you keep on varying length equal to 1, length equal to 2, length equal to 3, length is equal to m minus 2, if you keep on searching like this length you will see that we are going to get m minus r minus 2 runs of zeros and m minus r minus 2 runs of ones automatically.

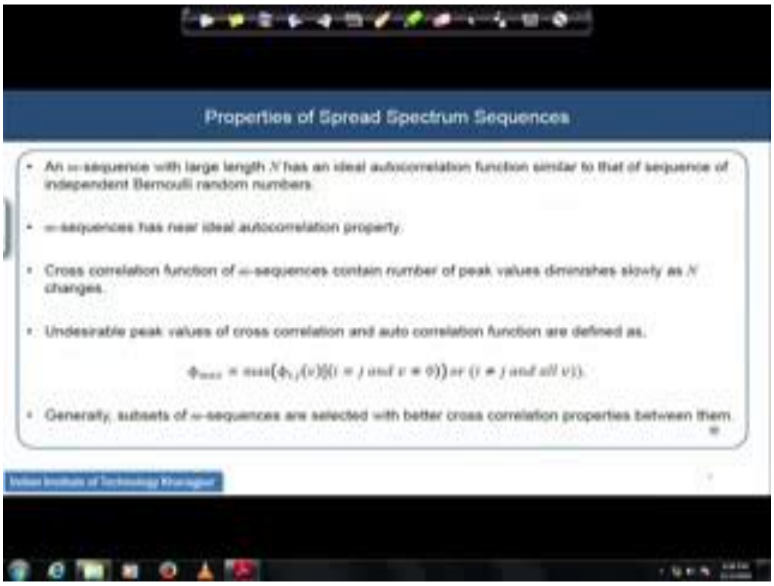
So, if I now start searching actually whether I will get 1, if it is m is equal to 7 say, shall I get actually six number of ones. How many times I will get six number of runs consecutively, it should be equal to 2 to the power 7 minus plus length of equal to 6 minus 2 minus 7 minus 6 minus 2 kind of, and then you will get to the runs of the zeros. So, if you are getting ending up with a positive value, if you are ending up with a positive value then only you will get the existence of it; otherwise you would not get the existence of that.

The autocorrelation property that we will discuss more in the next slides on next module that autocorrelation property of this code sequence ML sequence which is having the element of plus 1 and minus 1 in the discreet normalized fashion will be discussed by 0 to n minus 1, suppose we have the number of the observation samples. And we are having a factor C u is a sequence, ML sequence with us. And nu is taking the delay of it. So, the autocorrelation will be talking about C u minus u plus v in to C u, I mean you take that sequence and original sequence and you delay it that original sequence by an amount nu. And you take the multiplied value and average it over the number of the samples so of that observation. And basically capital N is equal to the length of the sequence at least to take the autocorrelation function with the delay value is equal to nu.

The cross correlation property is defined by this periodic discrete normalized cross correlation function, where actually there are two different state of ML sequences are we are taking, and trying to find out how much they are related, how much matching is possible in between them. So, there we keep a one sequence C j u, and another one is the C i u is having with respect to u, it is having another shift or the delay of say nu. And again we are considering the both of them are having at least same length equal to capital N, if it is average over capital N then this is the autocorrelation cross correlation of the ith sequence with the jth sequence for a delay of nu, we will tell.

And if we find that this autocorrelation value of the two valued function with values phi 0 is equal to 1; and if it is a two valued only two value, then it will be auto correlation value will be all one with the delay 0. And for delay one; with for delay nu it will be always minus 1 by N, and when your delay is not equal to 0. So, this minus 1 by N is a standard and well proven value for ML sequence. You take the cross correlation function of you take any two ML sequences, and try to find out the cross correlation value of each where your delay is not equal to 0. You will always end up with a standard value of minus 1 by capital N, where N is the length of both of the sequences and there we are considering the both the sequences are having same length. So, you will always end up with the value of minus 1 by N, we will see it, we will prove it in the next module.

(Refer Slide Time: 24:48)

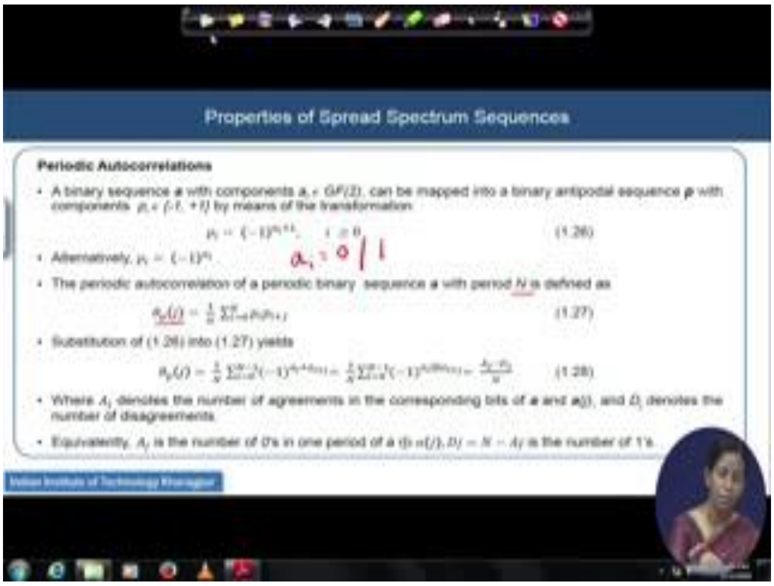


Now, coming up with a little bit more about the properties. If you are considering about the m-sequence with very large length capital N and which has an ideal autocorrelation function which is the similar to have a sequence of the Bernoulli random numbers. M sequences usually give you very near ideal autocorrelation property. What is the ideal autocorrelation property, ideal autocorrelation property starts from the random binary sequence property of the autocorrelation property of the random binary sequence. So, we find that the ML sequence give us a very close autocorrelation property that of a random binary sequence.

Cross correlation function of the m-sequences they contain the number of the peak values, which is the (Refer Time: 25:36) values also, and they are diminishing if you are increasing the value of the capital N. So, if you increase the length of the sequence, you will get the cross correlation value peaks heavily down which is a very good property for the code design, especially with related to the synchronization. Where we do not prefer to have the multiple peaks, then you will get a false lock in the detection of the synchronizing in detecting the synchronized points.

So, this undesirable peak values of this cross correlation and autocorrelation functions, they can be defined as this. Usually, generally, the subsets of any m-sequences they are selected in such a way that you can get always the best autocorrelation and the best cross correlation property. So, ML sequences are having the huge set and you can actually have flexibility to choose a pair of the good sequences, such a way that you get a good autocorrelation property and when I mean very high value of the autocorrelation property and the cross correlation property goes almost close to 0.

(Refer Slide Time: 26:48)



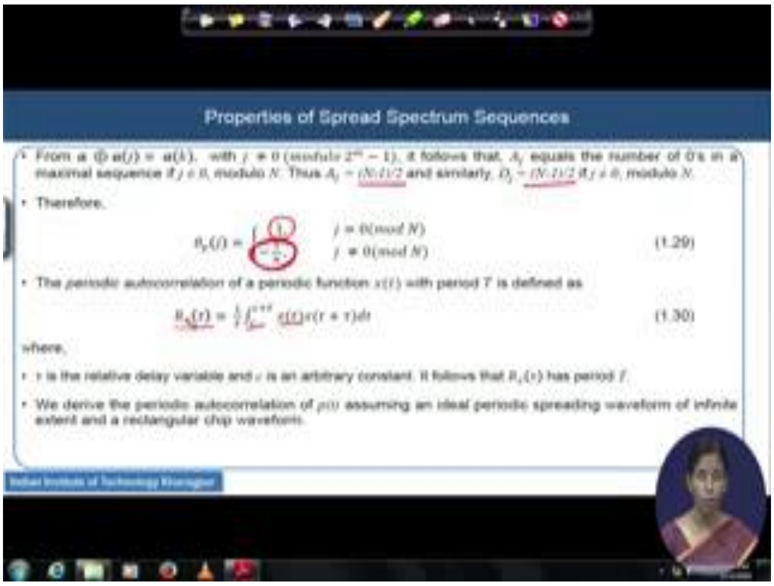
So, now there is we already understood that we do not prefer to go ahead with the random binary sequences that is why we discuss that why the periodic sequences will be preferable in from the view of the synchronizer design in the receiver circuit. That is why we entered into the linear feedback shift register, which helps us to the device periodic sequences. Once we are dealing with the periodic sequence, we have to deal with the periodic autocorrelation. So, autocorrelation cannot have a single periodic function cannot be a single periodic function.

So, let us slowly enter into that periodic autocorrelation values. Suppose, I have a binary sequence a, who is having the components drawn from the Galois field 2, we have the components a i and a i have drawn from the Galois field 2, we understand if it is a Galois field 2 component, it can have only two values 0 and 1. Let we can mapping that 0 and 1 value in a binary antipodal sequence P. In such a way that for 0, we will get minus 1; and for one, we will get plus 1, mathematically, I can generate every component of this p i by taking minus 1 whole to the power a i plus 1 or you write p i equal to minus a whole to the power a i. We can give a proof. See, if a i is equal to 0 I can have two values right a I can be either 0 or 1. So, when a i equal to 0, p i will be minus 1, and when a i equal to 1, it will be when a i equal to 0, it will be plus 1 and when a i equal to 1 it will be minus 1. So, you can generate all the values of the p i based on the value corresponding to the value of a i. So, that is a direct mapping.

Now, when we talk about the periodic autocorrelation of the periodic binary sequence a, we are considering that suppose the period that is associated with this binary sequence a is capital N. And we can define now the periodic autocorrelation of this p is equal to for a shift j is equal to p i p i plus j and average over 1 by N. So, as if p i is one set, p i plus j is the shifted version of this by shift is by j and then this is the periodic autocorrelation function for the period of capital N is yielding. And if I substitute now the values of p i and p j from here, I will be ending up with minus 1 whole to the power a i for p i into minus 1 whole to the power a i plus j for p j p i plus j.

And hence this is the final expression, we are ending up. And basically this is nothing but we understand that there is no within the over a Galois field 2 there is no instance of simple addition basically all are the modulo 2 operation of the XOR operation. And this whole expression now can be written as specially the after summation, it can be simplified in the form of A j minus D j. This A j denotes all the number which agrees with the corresponding bits of a; that means, A j is equal to number of the zeros and D j is the number of the ones. And the relation between them should be like this D j will be N capital N minus A j, which is equal to the number of ones in N for any number for any period capital in this situation holds good.

(Refer Slide Time: 30:33)



Now, question is from the consideration that a XOR operation with a j which yields will definitely yield a k, because we have already seen this proof earlier and with when j is not equal to 0. If it is an ML sequence definitely, it will do like this where a, a j, a k all will be ML sequence. And if j is not equal to 0, and j is modulo 2 to the power m minus 1 and then it follows that all the a j's that means, if this equation holds good. Then a j will be always equal to the number of the zeros in the maximal sequence, if j not equal to 0 modulo capital N and there thus we will be ending up with a value of n minus 1 by 2. Similarly it will be also n minus 1 by 2 if j is not equal to 0.

And therefore, then if we substitute these values in the equation of the theta p j in the last slide that we have shown for j equal to 0, the theta p where there will be perfect correlation and correlation value will go up to 1. And if it is not equal to 0, if you substitute there the values of A j and D j you will end up with the value of minus 1 by N. So, here is that autocorrelation value that we blindly stated as a property of a ML sequence that autocorrelation the cross correlation between when j is not equal to 0; that means, you are plotting the cross correlation values. If it is the cross correlated value output then you will be ending up with the always minus 1 by N.

Remember this periodic autocorrelation function x t, if the x t is having a period of t. Therefore, the classical definition for computing autocorrelation periodic autocorrelation of a periodic function - x t given by x t x t plus tau and where integration is running from c to c plus t; this c is an arbitrary constant, it can be 0 also in your analysis or any point. And basically it is the integration of a one period interval over which the x t is lying and the autocorrelation function by default if you are computing this it is obvious that autocorrelation function also will lie over a period of capital T only.

In the next slide, we will derive, in the next module not the slide actually the next module, we will derive the periodic autocorrelation. And the periodic autocorrelation, we will try to now find out with respect to the p t. We will assume that ideal periodic spreading waveform an infinite extend with rectangular chip imposed on each will be the assumption. And we will see how the periodic autocorrelation of p t will look like, the spreading sequence that we will look like.