CH5230: System Identification

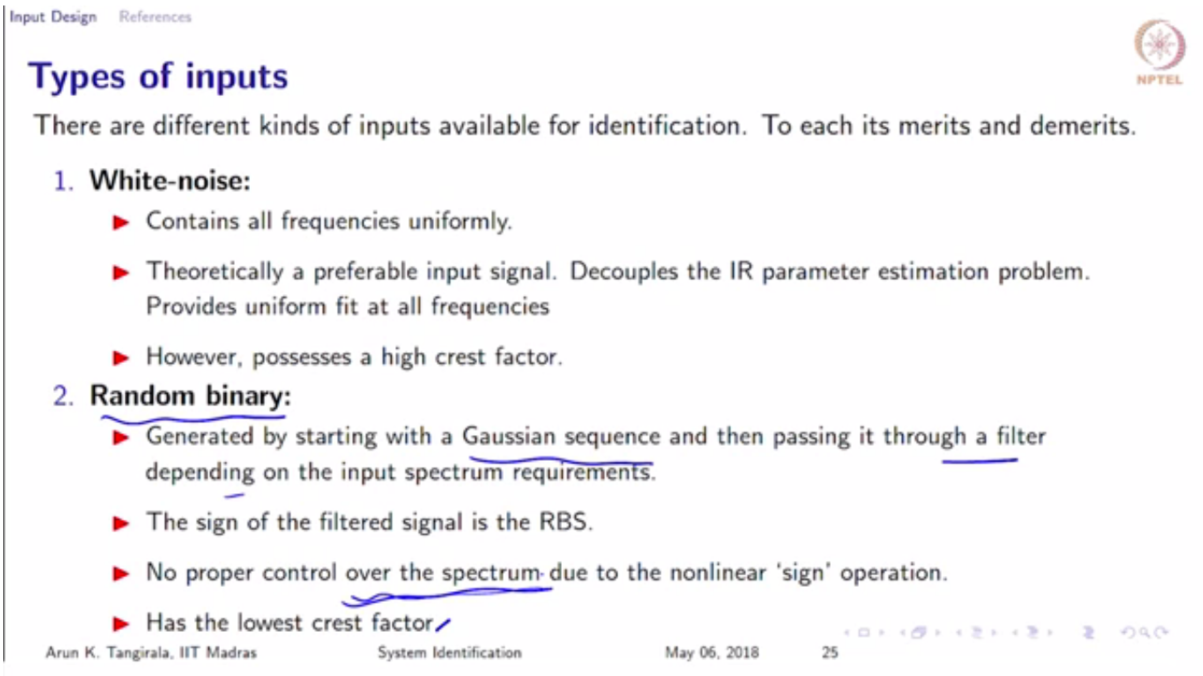
Inputs for identification

Part 3

So let us discuss this different waveforms quickly and then we'll talk about general guidelines for experimental design. So white-noise is a very frequently used input. Does it satisfy the persistent excitation. Yes. Because it has power at all frequencies. It is theoretically a very nice signal. In fact quite nice signals are used heavily in theoretical analysis. But it has a high crest factor as I said typically. When I say white-noise it could be Gaussian white-noise, typically we use Gaussian white-noise.

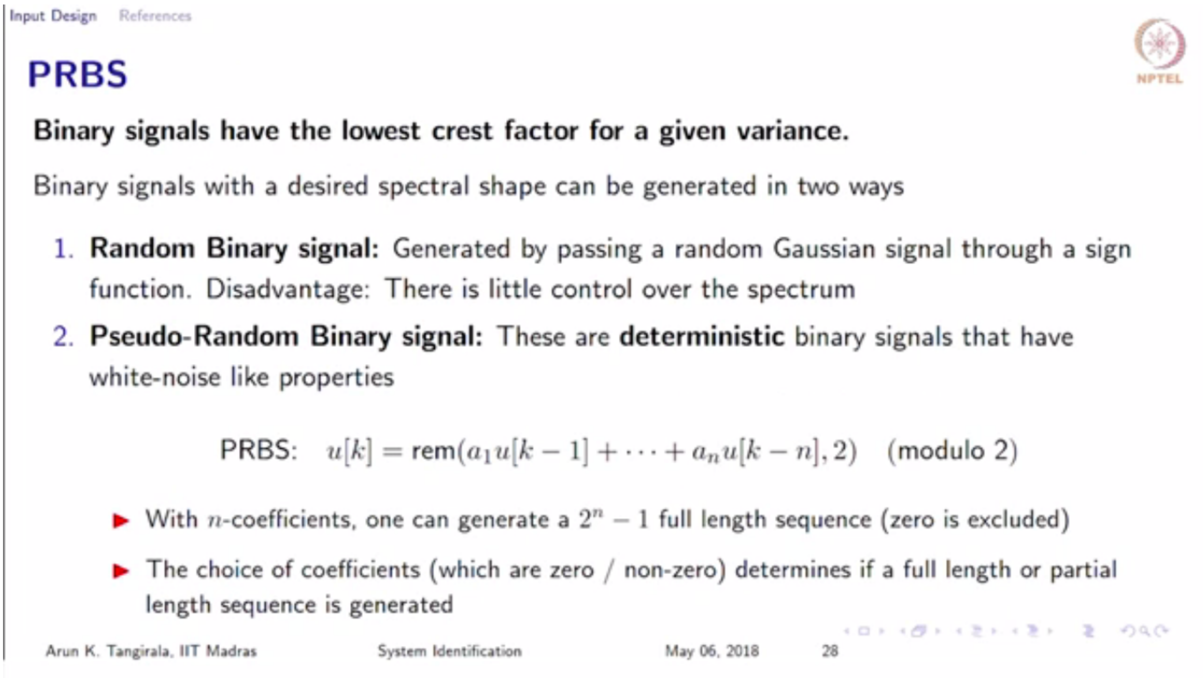
And on the other hand if you look at random binary signals which are generated by starting with the Gaussian sequence and then passing it through a filter. So you generate a random Gaussian noise. Then you pass it through a filter which looks at the sign of the number and gives it a 1 or minus 1. So it's like a non-linear operation because it's a sign operation, which means that the spectral characteristics are going to change. And it's hard to calculate what is a spectral content or it's hard to control, you can say the spectral content of the random binary signal when it is generated this way. Okay. So that is one of the main drawbacks. It has a very low crest factor. So it's almost there but the ability to design an RBS of a desired spectrum because very often we want the input to contain frequencies more or less within the bandwidth of the system. We don't want to inject power into the system where the system is not going to respond. I'm only going to hear noise. In a interview if I ask questions that I know for sure you will not have an answer to. Suppose you've come for an interview for, let us say data science. You want to [02:11 inaudible] there's a data scientist and you applied for it. And I ask you quantum chemistry. Then you'll just. I will ask you. You will stare at me quietly. No response. I will only hear the noise of the AC. That is what will happen in reality. I mean, I may ask you for example about psychology for data scientist. Oh my God. I mean, no response again. So these are outside the syllabus, outside the band width of the candidate. So don't excite the system outside, the way outside the bandwidth. You will only get noise and it's only going to spoil your parameter estimation problem. Therefore we want to design the inputs such that it has a certain spectrum and RBS does not give you that freedom that comfort.

(Refer Slide Time: 02:55)



On the other hand Pseudo-RBS, which is generated using a Linear Feedback Shift Register (LFSR). I'll Just explain to you what it is and then we'll come back.

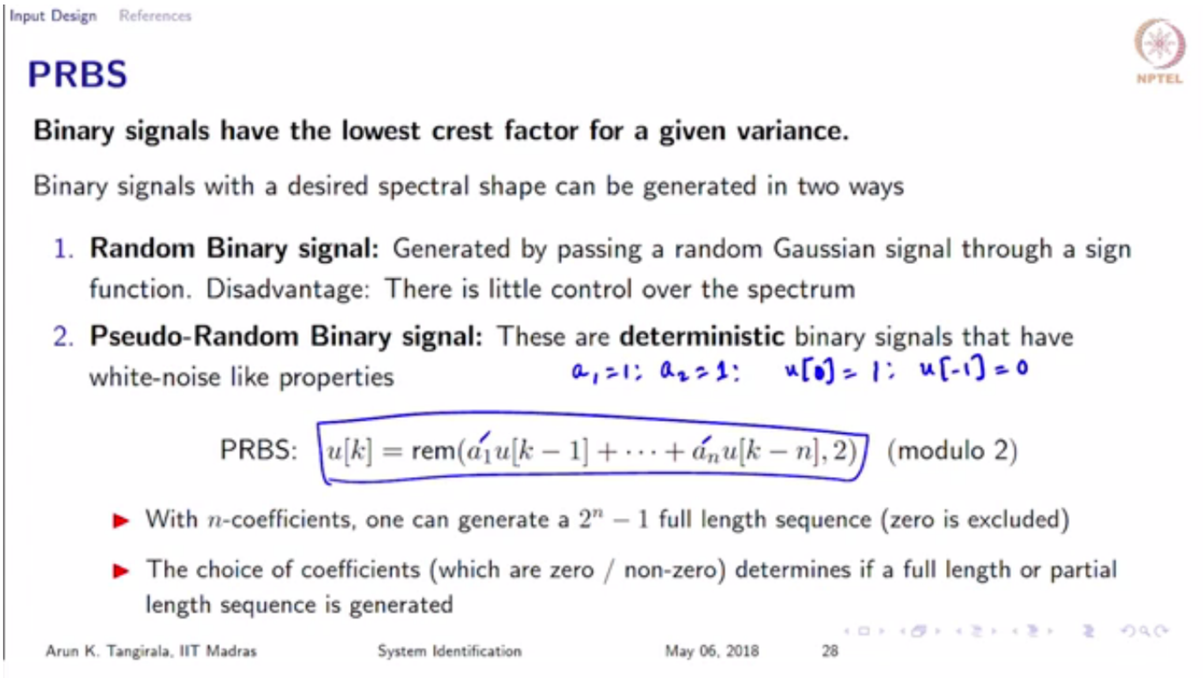
(Refer Slide Time: 03:08)



So the Pseudo random binary signal is generated using this equation. It's a modulo two operation. It keeps switching between 1 and 0. You can always scale it to have 1 on minus 1. As you can see essentially it is taking a polynomial, simple polynomial, the different people look at it differently. Mathematician looks at this as a modulo two operation and signal processing people may look at it as, a computer science person who is looking at logic operations may look at it as an operation using linear feedback shift registers, which are implemented using XOR gates. So doesn't matter eventually this is the question that is being implemented. And you have to initialize U K so that you can get started. Now, what is this a1 an, these are coefficients that you can choose and they are chosen in such a way that you generate what are known as the full length PRBS. It turns out that when you run through this operation after a certain time, certain value in the sequence. What you do is you keep running this operation, after a certain k, you will see the sequence repeats. So PRBS is a periodic signal, you should remember that.

So let's take a simple example here. Suppose a1 is 1 and a2 is 1. So, which means I'm just looking at two coefficient model. And let's say, I initialize my input with u 0 as 1 and u at minus 1. Because I will require u0 and u minus 1. Let us say this is 0. If you initially, if you start with both 0's, you'll remain at 0. So you should not do that. Then calculate. Let us calculate.

(Refer Slide Time: 05:14)



So this equation says, u1 is remainder of . So let's take here a1 equals 1, a2 equals 1, u 0 as 1, u minus 1equals to 0. Then let's start our calculation. U at time 1 is remainder of a1 u k minus 1. Which means a1 u0 plus a2 u minus 1. And remainder with respect to 2. It's a modulo two operation. What do I get? I get the remainder, what do I get here, 1,2 and what is the remainder, 1. Right.

Then I go to u2. What do I get here? Remainder a1 u1 plus a2 u0 and what do I get? I get remainder, what is u0?

1.

What is u1?

1.

So the remainder 1 plus 1, 2,2 and what is that?

0.

Now, I go to u3? Remainder a1 u2 plus a2 u1, what do I get here remainder, 1, 2, 1. Now, you can show that then you to u4, you'll keep shuttling. Okay. So, again here you do all of this. What do we expect to see?

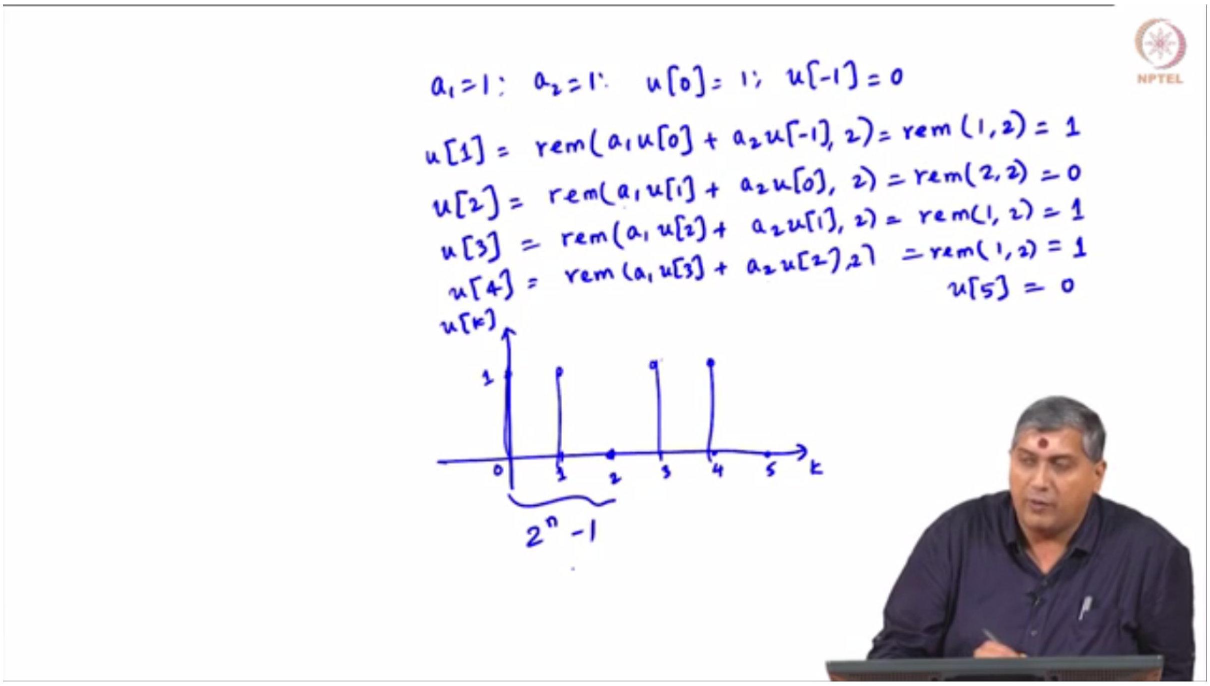
0.

Right.

So if you were to plot this. So let's say this is u k. It depends on where k begins from. If your k is beginning from 0, then it is 1 1 0. You started with u 0 1, in fact u4 here, a1 u3 plus a2, yeah, u2. Correct. So I get here 0. Do I get 0? I think I get 1. Remainder. So a1 u3 is 1, a2 u2 is 0. So, I will get here 1. Now, u5 would be 0. Do you understand. So what will be u5? u5 will be 0. So I keep getting 1 1 0, 1 1 0, so at u0 I have 1. And at u1, what is the value? I have 1. And so this is k. And at u2 0. And then u3 1, u4 1, u5 0. So you can see here. So this is the reputation period. In fact, 2 to the power of n minus 1 is a length.

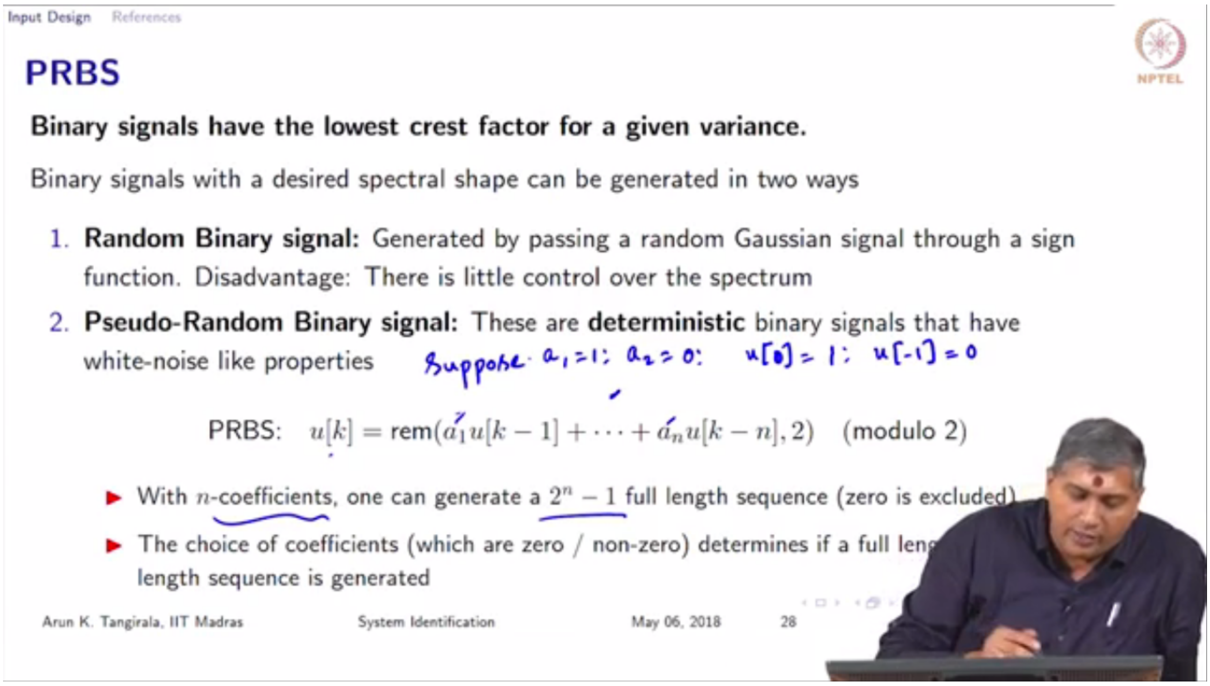
So the PRBS repeats after 2 to the power of n minus 1. That is, if you're using an N length polynomial there. Okay.

(Refer Slide Time: 09:09)



So going back, with n coefficients we can generate 2 to the power of n minus 1, so-called full length. What do we mean by full length? I'll leave it to you. Suppose I chose a1 equals 1 and a2 equals 0. Suppose I chose this. What would happen. You will not. You will see the repetition quicker, all right. So depending on how you choose your coefficients. So that means also we learn that we are to choose our coefficients, so that we maximize the period of PRBS, so that we don't see the reputation as much as possible. Repetition cannot be avoided. But we want to push the period to 2 to the power of n minus 1. And you can show that the full length PRBS has white noise like properties.

(Refer Slide Time: 09:55)



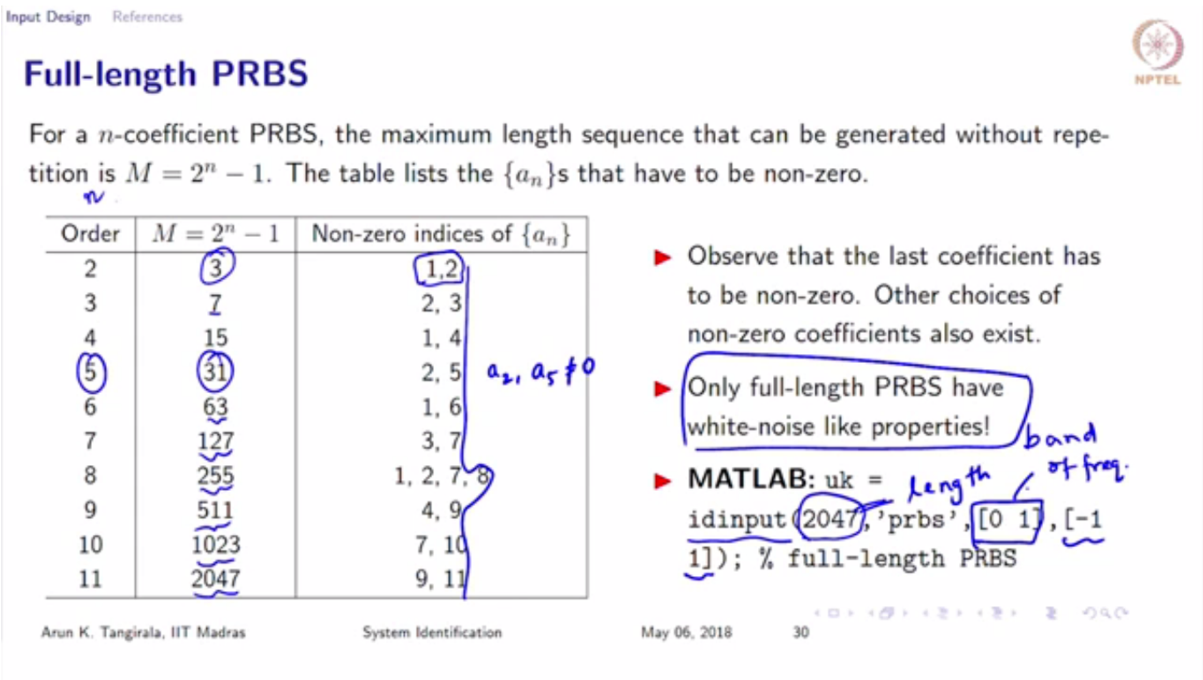
This is an important property to remember. And what I've given in this table is depending on what you choose as n, suppose this is order. n is 2, PRBS of order 2 then you'll get full length 3 and both these coefficients have to be non-zero. I chose a1 equals 1, a2 equals 1. If you choose one of them to be 0, you will not get a full length sequence. So let's pick for example 5. Order 5, this is the length of the full length PRBS and you have to choose a2 and a5 not equal to 0 for sure. The remaining three is your choice. You can set a1, a3, a4 zero or you can choose them to be non-zero. But definitely a2 and a5 had to be non-zero, if you want to full length PRBS. And definitely you see interestingly all these last coefficients have to be non-zero. If you want a full length PRBS. Right. That is what we see common across all orders. That means if I'm looking at the PRBS of order, let's say 13 or 17. Certainly a17 has to be non-zero. In addition a few also have to be non-zero but certainly a17 has to be non-zero.

So in MATLAB you generate full length PRBS using this command. idinput, this is the full length 2047 is one full length. That means you can pick any of these values, if you want a full length PRBS by looking at this if you've figured out what is M. It will add 1 to it and then take the log 2. And then it runs through the polynomial using the coefficients. And how does it know it, you are asking for a full white noise or a full length thing. This is only telling me the length of the input sequence. How does it know? You're asking a full length by looking at this argument here. This is the band of frequencies that you want the PRBS to contain. Just know I said a full length PRBS has white noise like properties.

So it looks at the band 0 to 1 means it understands that you want all the frequencies. One corresponds to the [12:21 inaudible] frequency. 0 corresponds to the DC. The moment it looks at it, [12:26 inaudible]. The user wants a full length. I am going to add 1 to this 2047, and take log 2 of it and use the corresponding polynomial and give it to you. And this argument here is simply to choose the state's. Typical standard PRBS switch between 0 and 1. But you can change it to minus 1 and 1. That's not a big deal. It's just a linear scaling.

Now the band here is an important thing. We have just discussed that we may not want inputs to have white noise like properties. We don't want the full excitation, we want partial excitation, band limited input signals.

(Refer Slide Time: 13:03)



With PRBS, it's easy to design a band limited PRBS by all you have to do is by sampling, re-sampling P times faster than the frequency at which the PRBS is generated. So when in PRBS, if you go into the full details on some website will tell you, there are two clocks, one clock which is keeping track of when you are going to update your input and the other clock is when the flip flops that are implemented are changing their states. So in a full length PRBS this two clocks are in sync. The flip flops chain is and clock also. When the clock changes the flip flops also changes. Suppose I want to ban limited PRBS. What do we mean by band limited PRBS, we don't want the highest frequency to be the [13:55 inaudible] frequency. So how does a typical PRBS look like? You can you can recall in many of the simulation examples we have seen. It would look like this. So very narrow ones as well.

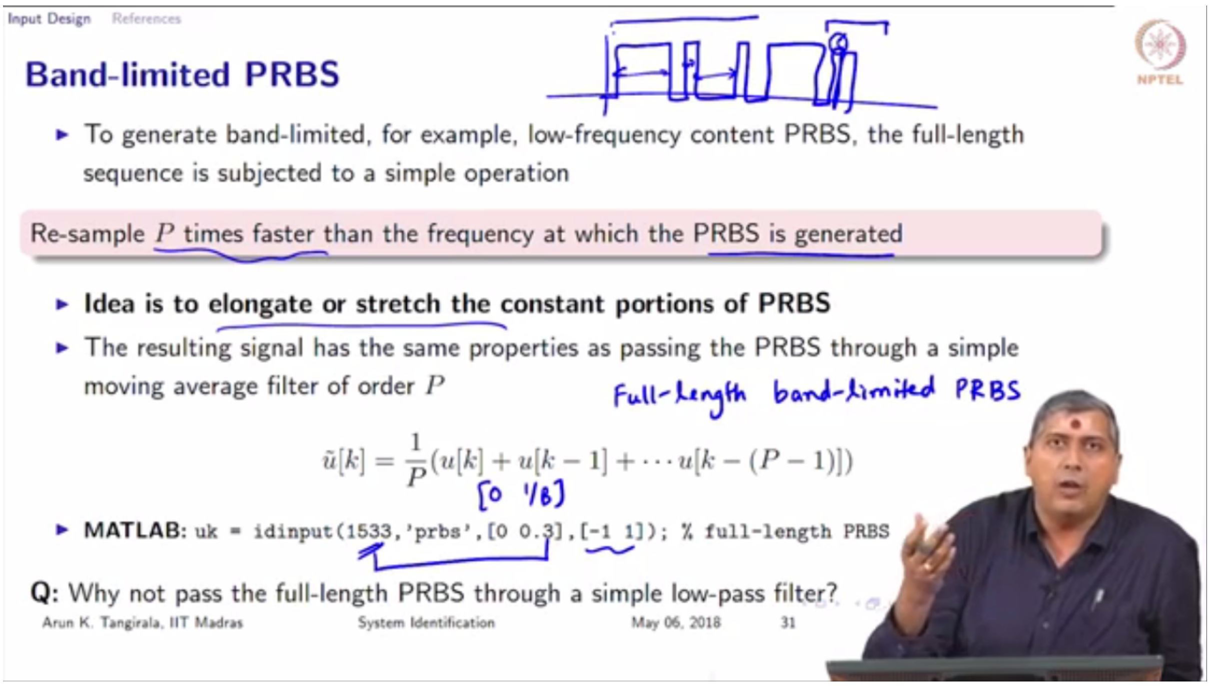
So we want the narrowest one to be stretched. In fact every, all pulse is here, this is one pulse, small, short pulse [14:25 inaudible] wider one, this is a short one, this is a medium one and so on. So we want the highest frequency to be clipped and the width of that pulse determines the frequency content. So the narrowest pulse that you have should now become wider. And everything else also should be stretched. So what do you need to do is, you need to sample much faster than the rate at which the flip flop is changing its state. That means the flip flop stays at this stage for a while, you sample re-sample it much faster, so that this one looks as if it was of this state for this long.

So this has been stretched, everything else get stretched this also now get stretched. So that is the idea you re-sample P times faster than the frequency at which the PRBS is generated or the idea is to elongate or stretch the constant portions of PRBS. So that is conveyed through this command here. In fact through this argument typically you'll see 0 to 1 over B. In fact you will specify here 1 over B. B is the number of times that you want to sample fast. So if B is 4 its point 25. Okay. If B is 2 then it becomes 0 point 5 and so on. So according to this argument and this length? Now what happens is we have discussed, what is the full-length for a full band PRBS?

Likewise there exists also PRBS is anyway periodic. Even band limited PRBS are periodic. We are not going over the theory, but that exists again as an extension of what we have just discussed. A full-length band limited PRBS. So it is possible to have full-length band limited PRBS, which means band limited PRBS also are periodic and we want that. So that the full-length band limited PRBS will have the desired spectral content. Now there is a formula which is not available in the documentation, MATLAB documentation. But if you choose the wrong sample size here. Wrong meaning. That doesn't generate the full-length band limited, essentially these two have to be here the length and the band have to be consistent with each other, if you want full-length PRBS.

So, 1533 happens to be the full-length one for 0 to point 3. You try changing this in MATLAB you'll get a warning message and it will tell you what should be the full-length for a given band. For a full band the formula has been given. That is the basic idea. Okay. So you have full-length white PRBS, your full-length band limited PRBS. Full length white PRBS can be easily calculated. Here there is a formula there full-length the band limited PRBS has a slightly more complicated formula. This are still remains the levels. So that is how a band limited PRBS is generated. And PRBS are very good because like RBS they have the lowest crest factor and as you have seen it's very easy to shape the spectrum of the PRBS. Just you have to change the 1 over B. That's all. Okay.

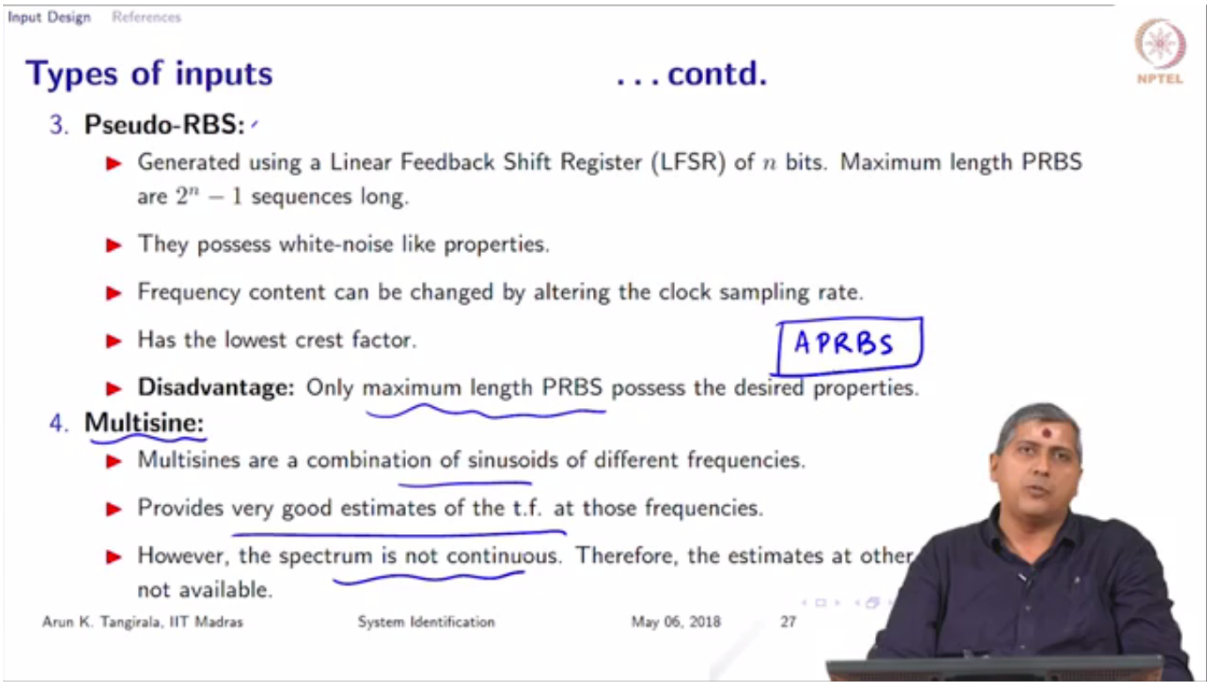
(Refer Slide Time: 18:04)



So let's quickly wind-up on the different signals so we have talked about PRBS and so on. Now always you want to work with full-length PRBS as much as possible but sometimes you may not be because the experimental duration may not allow you to implement the full-length PRBS. And the other limitation of PRBS is it's only good for linear systems. For non-linear systems there is some modification called the APRBS you can refer to the book by [18:38 inaudible] on non-linear system identification, which talks about how to generate this APRBS. Linear systems obviously it is good because it switches only between two states and that's good for linear systems, but for non-linear systems that is not good enough and APRBS over comes that to a certain extent. So the last class of signals that we want to talk about are multisine. Multisines are by the very name mixture of sine waves of different frequencies. They're a combination of sinusoids of different frequencies and they provide very good estimates of the FRF at those frequencies. But the spectrum is not continuous if you want a continuous spectrum one, then you have to mix infinitely many sinusoids.

And the biggest challenge in designing multisine is to how to phase? What is a phase difference within the sines that you mix? So there are some calculations in the literature there are some methods, which tell you how to adjust the phases of multisines. There is a [19:33 inaudible] formula which tells that you want to keep them as much as out of phase as possible to keep the crest factor low. So if you are mixing two sines keep them as much as out of phase, so that the amplitudes don't add up. If the amplitudes add up at some point in time, then the peak will rise in the crest factor will increase. So that is the reason why you want the sinusoids to be off, out of sync or out of phase as much as possible. Anyway, so we will not get into the other aspects of design of multisine, but you should know that is a major challenge. And they are good for systems which are model. That means where you're interested only in the systems knowledge at specific frequencies.

(Refer Slide Time: 20:11)



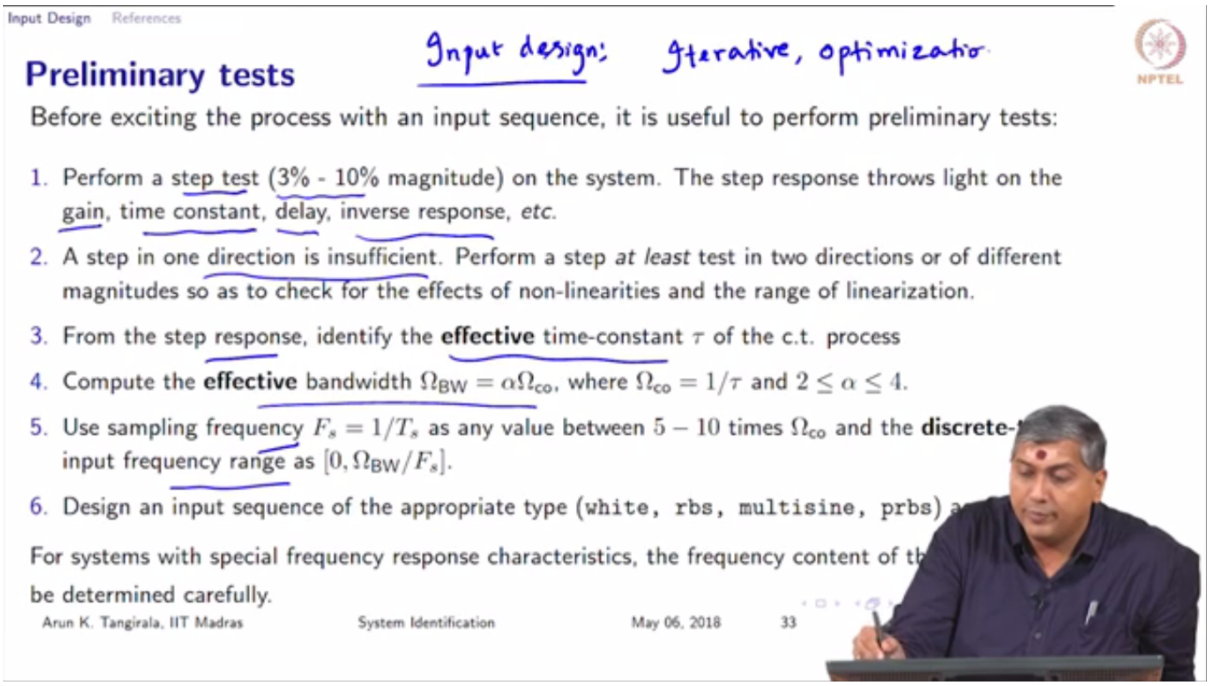
We have talked about this we will close the essentially now we have talked about persistent excitation general requirements for input design, different commonly used signals particularly we spend time on PRBS. We will close this lecture and the course also on a simple guideline as to how you design an input for a general system.

Generally you start with a step test. Not too much of a magnitude of the step 3 to 10% depending on the system. And typically a step in one direction may be insufficient. We may have to do at least two steps, one in the positive and negative or a staircase like signal. All of this is to get some preliminary knowledge of the system. This is like candidate walks in you say introduce yourself. Okay. There the candidate is able to speak. Here the process you cannot go to the process and say, hey, please tell me something about yourself. Will have to do a step test and this step test will give you very good idea of the game of the time constant of the delay or if there is an inverse response and so on. So you get a lot of valuable information like when you say introduce yourself, you get to know how well the candidate is able to communicate, how comfortable the candidate is in the medium that you want the person to speak, language in he want to speak and so many other important details, whether the candidate can hear and so on.

So from the step response you can find effective time constant if it is a first order behaviour. If it is a second order like behaviour that is underdone behaviour, then you can get the time constant and the damping ratio or the raise time. And then you can also calculate the effective bandwidth. And from the effective time content you can calculate the sampling frequency and the range of the input frequencies you want that is the idea. And once you have the sampling frequency and the frequencies that you want to inject into the input then you design your input typically using a PRBS if it's a linear one or a multi sign also if you want only at specific frequencies. Okay. So you go through all this preliminary analysis in general input design is an iterative exercise and a lot of the modern literature on input design essentially involves solution to optimization problems. And what does it optimize?

It actually maximizes the information content as quantified by [officials] information. Because the [23:08 inaudible] says that the precision of the estimates is directly proportional to the official's information or the variability's inversely proportional. So you want to minimize the covariance of parameter estimates so the trace of it. Or you want to maximize [officials] information and then maybe other constraints also on the inputs the values cannot move beyond something or there may be some physical constraints the rate at which the value can change. All these other constraints have to be taken into account which we have not talked about in our Vanille input design lecture that we just had. So overall the input design problem is an iterative optimization problem you should remember. Input design is an iterative, because you have to first know something about the system then only you can design the input. And unless you design the input you will not know something about the system. So it's an iterative one. And optimization problem, you should remember this.

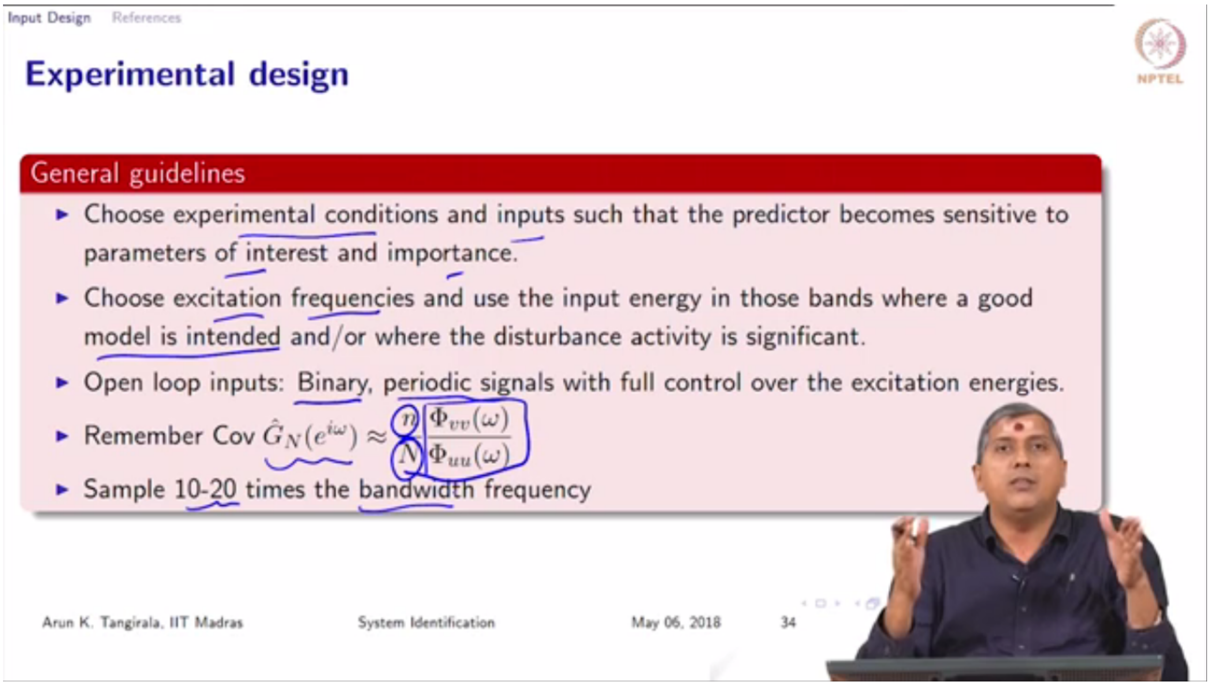
(Refer Slide Time: 24:11)



So general experimental guidelines, of course, I'm not gone go over all of this but you want to choose your experimental conditions and inputs, such that they are aligned with the parameters of interest and importance. And as we have just spoken you want to excite the system at those frequencies, where a good model is intended. So sometimes I may not [24:39 inaudible] model, good model over all frequencies. So you have to put in that into your input design thing. And General the thing to remember binary or periodic signals maybe preferred. And always remember the signal to noise ratio thing. There covariance of the frequency response that variance you can think of it as variance, if you like it. The variance of the FRF estimate is inversely proportional to the signal to noise ratio.

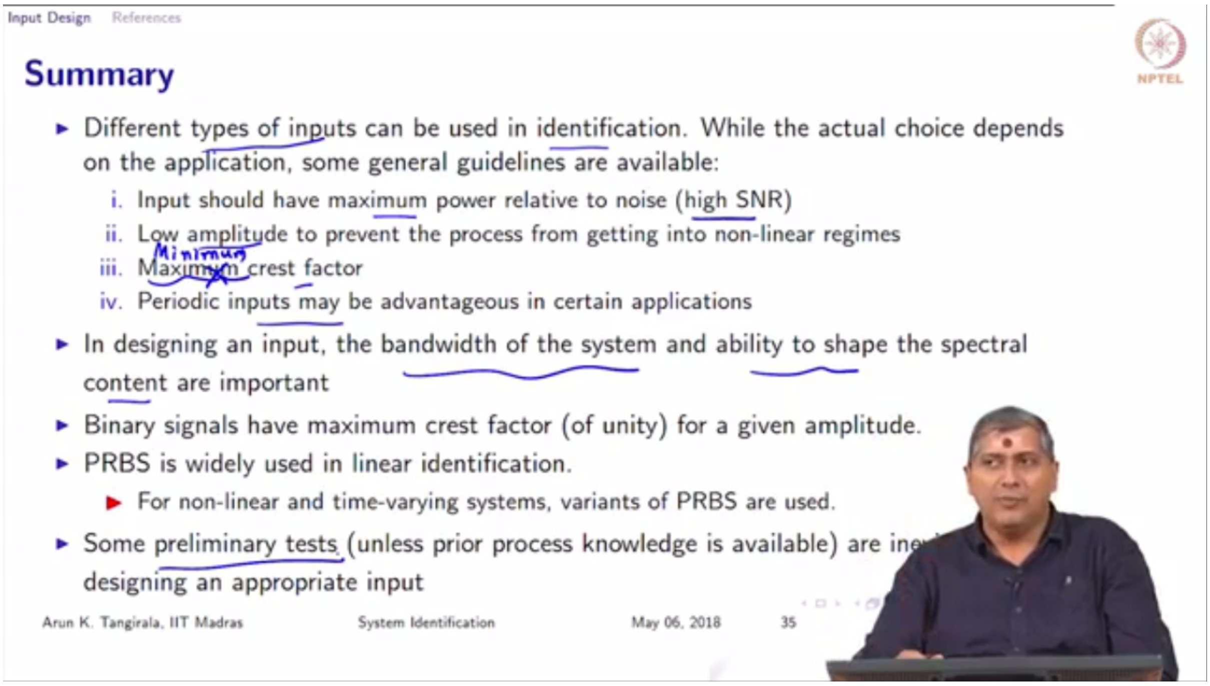
So if they say higher to signal to noise ratio better is the estimate of the FRF at that estimate. N is the number of sampled data points that you have, the small n is the order that you're fitting. And typically sample 10 to 20 times the bandwidth frequency. The bandwidth is a term that you will keep hearing everywhere. In SysID the bandwidth also almost has similar definitions as a signal processing one. But for all practical purposes remove the, remember that the bandwidth is nothing but the range of frequencies over which the system has an active response or a significant response.

(Refer Slide Time: 25:42)



Let's summarize this lecture. Different types of inputs are available for identification and these are some of the guidelines which should have maximum power, high SNR but low amplitude conflicting requirements. And that is why we want. So we want not the maximum but the minimum crest factor. It's wrong. And periodic inputs may be advantageous that [26:10 inaudible] sinusoids, multisines for example maybe advantages. ETFE also turns out to be for example, consistent if you use periodic inputs. And the basic criteria, that you consider in designing input is a bandwidth of the system and the ability to shape the spectral content, these two are important. And the other thing is that we have already talked about remember, input design is iterative. So some preliminary test is inevitable. Don't think that you will have all the information with you and in one shot you will be able to design the input.

(Refer Slide Time: 26:48)



Remember setting a question paper for an exam is not easy. We have to actually struggle a lot as teachers. When I was a student I used to think it was easy. Or you know all the faculty had to do set the question paper and here I am struggling trying to answer. But I find that this is even more difficult, because I have to take into account. All the students in the class what I have thought? What is the bandwidth now of the class? What?

Whether I should put in my effort in. And remember, I should be able to distinguish between our top performing student and a mediocre performing student and so on. I can't ask questions that will not allow me to distinguish and then I'm in a dilemma as to whom what grade to give whom. So same kind of challenges that I face in input design is the one that I face in setting a question paper and that's true for everyone. Okay. So hopefully you have enjoyed this entire course on identification, go back and look at the flow of sequence that we had shown in the early lecture on identification. And ask yourself now do you have, now if you have learnt at least some theory corresponding to each stage of identification starting from input design to non-parametric modelling, parametric modelling, model validation and then the entire parameter estimation itself.

So this course has been fairly rich, simply because of the subject. It is a confluence of four different fields as I've said earlier. And therefore you must have found it to be heavy, but you are not alone there is a lot of company for you. I also found it heavy when I learn and I also find it heavy when I teach it. So it's the same story, but it's an enjoyable subject. It's a very practical subject. It allows you to understand a lot of things that happen around you, because that is what system identification. We observe and we learn. And SysID is also about observing and learning. Learning is a fancy term today, so I am going to use that term and today people are talking about how to marry SysID with machine learning and you can see that happening in, you know, I'm talking about that in my course a few years from now. Okay, then enjoy and happy learning.