MITOCW | 9. Transmitting on a physical channel

PROFESSOR: I wanted to begin by just picking up a couple of things from last time. This is part of our sneaky agenda of trying to teach you some probability as we go along. This is maybe a little less crucial than other things we've been doing, but to make sense of the last couple of slides from last time, there was actually stuff I swept under the rug that it won't hurt you to know.

Just a reminder, we talked last time about the PDF of a random variable. I neglected to explicitly say that about something that's got to be non-negative for all values of x. And the reason is that if the area under this thing is going to be the probability for any interval, no matter how small, if the area under it is going to be the probability that the random variable falls in that interval, since probabilities have to be non-negative, well, this function itself had better be non-negative. OK?

So that's an explicit condition, and then there's the normalization. The question now is, if you're dealing with multiple random variables, how does the story change? So if you've got two random variables, let's say x and y, so two things that can take numerical values, say height and weight of a randomly picked person. We use a very analogous object. It's the joint PDF of the two random variables. So it's some function of two variables, non-negative, so you can imagine it like some probability mass that sits on the plane normalized to unit area.

And the amount of mass over any particular piece of area tells you what the probability is that you fall in that region. So the expected value, we talked about how to get expected value in the single dimensional case. In the 2-D case, it's a natural generalization. So the expected value of a function of x and y, just take that function under the integral signs, so you're taking an average with respect to the PDF. OK?

So it's a very natural extension. This is for the case of two variables, and in the same way for m variables. Now last time, I talked about a very special case that involved multiple random variables. These were the random variables corresponding to the noise samples. So we sent out a nice clean looking signal. This was our x of n. And then what we received was something that was perturbed. And so we would have liked to get in the noise-free case the same thing, but what we got was this perturbed by a certain random amount.

So the wi's this was w1, for instance. It's the amount that you add on or subtract from a given value to get the actually received values. OK? OK. So these are the wi's or wn's. So we had multiple random variables because we were taking many samples in the bit slot. And in particular, we looked at taking the average of a bunch of measurements. But now, what you've got is a function of many random variables. So what does it mean to take the expected value of a function of many variables? What does it mean to find the variance of that? Well, we didn't actually get into the details of it.

But it turns out, there's special structure here that made those computations very simple. So one key thing was we said that these noise samples were independent from one time incident to another. And that's really the crucial thing. The other piece was just for-- well, it made sense for our application because of the central limit theorem, but we assumed that these noise samples were Gaussian with varying sigma squared.

There's a term I didn't use, by the way, for this kind of noise, but you'll see it in the notes. It's what's referred to as additive white Gaussian noise. The additive part is clear. The white means that it's IID noise. The reasons for that name become clear when you think about frequency domain, so we won't get into the origins of the name. But the key thing here is that these are independent random variables.

So we should have actually been talking about the joint density of all these random variables when it came to computing expected values and variances. But it turns out that there are actually some simplification, so I'm just going to give two statements which are the things I want you to carry away. One is that expectation is always additive. So when you take the expected value of-- I'm doing this for the case of two random variables by the way, but the same thing goes for m.

If you've got the expectation of a sum of two functions of these random variables that you're interested in, the result is the sum of the individual expectations. And that's just the consequence of the fact that expectation is defined through integrals and integrals are additive in their arguments, in the integrands. Right? So that's all that's involved there.

Now, the particular use we make of it is for actually for a sum of this kind where one of the functions is a function of just one of the random variables, the other function is a function of just the other random variable. And so if you apply the result up there, you get sum of the expectations. And the nice thing now is that each of these is just a 1-D expectation. So we never have to deal with joint PDFs, joint distributions, multiple integrals, and so on. It all stays as simple as the 1-D case.

And the reason is that in every instance we talk about, when we have the expectation of a sum of functions of multiple random variables, well, the sum actually involves functions, each of which is a function of only one of those random variables. So it actually becomes very easy to compute.

Here's another thing that's interesting with expectations, which is that under independence, you can actually have expectations be multiplicative. In fact, let's see, so the expected value of the product of a function of just one of the random variables and a function of just the other factors into the product of the individual expectations if these two random variables are independent. So that, again, ends up being used.

For instance, when you're computing the variance of, let's say, w1 plus w2, so the variance of this is going to involve computing-- this is the 0 mean case. So you'll find that you're computing the expected value of w1 squared and then the expected value of w2 squared. And then you've got twice the expected value of w1 w2. OK?

So while this is easy, this is sigma squared, this is sigma squared, what do we do with this term? Well, if it was a general function of two random variables, then you've got to pull in the joint density, and it becomes a big operation. But if these are independent, then this expectation factors into the product of the individual ones. These are 0 mean random variables. And so this goes to 0. OK?

So this kind of computation was going on in the results I quoted or claimed in the last lecture, and I just wanted you to have that in mind. OK. But what I really want to talk about for the rest of the lecture is going back to understanding and modeling the single link. So we'll leave the probabilistic stuff and that for now. OK.

So we're back to this picture of bits coming in, being converted to signals. The signals are in discrete time here. They're being adapted then to transmit on an analog channel, which has the noise in it, and then at the other end back to discrete time through an inverse kind of operation and out to bits again. So we're going to look in more detail at what goes on in there.

What we did with all our noise analysis last time was really focused on this box where the decision is. Right? You get a noisy sample and you're trying to decide what you have. So the lecture last time was focused on that box, and we're going to look at the other parts of the picture this time. OK.

So the digitized symbols that we talk about, here's an example. We saw this last time. So what you're doing is taking the bitstream and deciding that you're going to represent it, for instance, as a voltage one held for a number of samples to indicate a 1, a voltage of 0 held for a number of samples to indicate a 0, and so on. OK? So this is the signal that you want to get across the channel. And then at the other end, that signal will get interpreted as a string of bits. OK? When you do the optimal detection that we talked about last time.

So this link actually has-- it's a very hybrid kind of thing because you've got clock discrete time stuff happening here. This is a digital-to-analog converter, so you're going from clock discrete time to continuous time. You typically have a continuous analog channel. And then at this end, again, back to discrete time, so analog to digital. Digital, by the way, we use that word a lot. What we typically mean is something gets sampled, it's a discrete time signal, and there's often the implication that it's quantized to one of a set of levels. It's basically the signal you'll deal with in a processor whereas all of this is the signal that you deal with in the physics, in the analog part. OK.

Then you're back to the screen time here. And actually, there are two clocks even there because there's a particular clock that drives all of this signal processing. But then when you come to spitting out the bits, you're only going to spit on one bit per bit slot. So you've got many samples per bit slot, and then when you come all the way out, you're only going to report one number, a 1 or a 0 per bit slot. OK? So there's all of this mixed together in the system. All right.

So let's look at the particular case you're going to be seeing in lab. I put this up on the slides last time as well. So we're going to talk about the specific case of a channel that's just an acoustic channel. So we're going to have sound coming out of a loudspeaker, that's your transmitter, sound getting picked up in a microphone, and that's your receiver, and then all of the signal processing. OK? So labs four through six are going to be centered around this.

So the challenge then is taking the digitized symbols there and putting them to a physical channel. So what is it that happens in between? Let's see. The D to A converter, we don't say much about that, but a typical D to A converter is taking a sequence of samples which are just numbered and then converting the samples to a continuous waveform, which is on a time axis. So a discrete time sequence is typically on an integer axis, but we convert in a D to A converter, what typically happens is that you're converting this to a continuous time waveform.

And the simplest way to do that is through what's called a zero order hold. So you take the value here, hold it for some interval of time, t seconds, and you take the next value, you hold it for t seconds and so on. And then when you get a change in value, you change to the new value. So you would hold this here, and then come down here, hold it. So at the end of it, what you've done is convert a discrete time sequence into a continuous time waveform that can then be applied to something like the loudspeaker. Right?

So you'll have to specify in the D to A converter what your reconstruction interval is. This kind of a D to A converter would be called a zero order hold. Zero order because it just looks at the most recent sample and holds it. A first order hold would look at the last two samples and do a linear projection. So you can imagine more elaborate ways of doing the digital-to-analog conversion.

So what I want you to imagine is that when we get to the DAC finally, it's going to be something like this. So all my pictures will be discrete time sequences, and I won't say much about what goes on here. So I want you to imagine that whenever I have a sequence like this, and then I end up putting it on the physical channel, there's been a conversion of the state. OK? That's what you're D do A card will do in the computer at this point. So you feed it a bunch of numbers. You give it a sampling rate or a reconstruction rate. And then it does this kind of interpolation. OK.

So let's see. Is this a good voltage to put on a loudspeaker? If I wanted to signal a one or a whole series of ones, do I want to put a constant voltage on a loudspeaker? A loud speaker is not very happy getting a DC voltage on it. Right? So the point is here that you have to think about your transmission medium and what its happiest responding to. So you've got to adapt your signal to the capabilities of the physical medium. And that's what modulation is all about, or at least that's a key part of it. Another part of modulation, we'll see later, is to allow you to have multiple users share the same channel. But a big part of it is just adapting your signal to a form that is comfortable for the channel.

So here's what you might try in the case of the loudspeaker. So what you've got is the acoustic channel. You would like to transmit these two levels, v0 and v1, to represent the 0 and the 1. I'm just generalizing here. I'm just saying that there's some level v0 that represents the 0 and there's some level of v1, and I will allow you to pick different possibilities here. So what's typically done is instead of trying to transmit the DC, you transmit a burst of a sinusoid because loud speakers like sinusoids provided they're at the right frequency.

So again, you've got to think about what frequency makes sense. It's this cone trying to move a massive air to try to make a massive air oscillate. There are particular frequencies that are good for that, so you have to think about that. So you might, for instance, say that you want sinusoidal bursts at two kilohertz, and you're going to modulate the amplitude. So you'll send a burst v0 cosine 2 pi of ct to represent the 0, and you'll send a burst of v1 cosine 2 pi of ct to represent the 1.

So if it's simple on-off keying that we've been talking about, you'll have v0 equals 0. In other words, you'll signal a 0 by sending nothing out on the loudspeaker. And then when you want to signal a 1, you'll have a cosine of amplitude capital V. In the other case here, what you do is you have minus v cosine going out to signal a 0 and plus v cosine going out to signal a 1. So that's basically just a 180 degree change of phase. Every time you want to shift from a 1 to a 0 or a 0 to a 1, you're going to change the phase by 180 degrees.

That changes the sign. Right? Because it's fixed amplitude, and you step the phase each time you want to change. So this would be a natural way to do it. Why two kilohertz? Well we know that this is the kind of frequency that a loud speaker likes. And the more general principle is whenever you're trying to radiate energy, the size of the antenna that you use, the antenna element that you use, for efficient transfer of power, has to be comparable with the wavelength of the signal that you have.

So for instance, if you're talking about sound at two kilohertz, the speed of sound and air at room temperature is something on that order 340 meters per second. If you do the computation of the wavelength, and I always do it with the units, so I may get it wrong here. But 340 meters per second, and for wavelength, I know that I want the answer to come out in meters. I've got two kilohertz, so I've got to divide by 2000 per second. Right? That's the units of frequency, and that's going to give me something in meters.

And if I wanted to come out in centimeters, then it's 340 divided by 20, so that's 17. And, well, 17 is in the ballpark for the dimensions of a speaker. You actually-- it depends on the details of how this is done, but you might be satisfied with a quarter wavelength of the transmission. Quarter wavelength of 17 is really very well within the range of what a speaker might be on a laptop, depending on the size of your laptop, I guess. OK. So all of this goes on in trying to figure out how to modulate the signal onto a channel.

So these are instances, actually the very simple instances, of what's called amplitude modulation. Actually, in the very first lecture, I mentioned when we were trying to distinguish analog communication from digital communication, I said that the typical analog communication scheme might be AM, amplitude modulation, where you take a carrier and you modulate its amplitude. And the amplitude is what carries the information. So this would be something of the type x of t cosine 2 pi of ct. All right.

So it's a carrier, which is a pure cosine, and you have an amplitude that slowly varying. And it's the amplitude that carries the information for the analog communication. So the receiver what would be done is figure out some way of extracting the envelope here. All right? That's classic AM. Ours is a very simple case where we're saying rx of t is either 0 or v in the case of on-off keying, or it's a minus v or plus v in the case of bipolar keying. But it's still on the AM kind of modulation. OK.

So what I have up here is actually just to remind you that this also happened in the neighborhood here. This is Fessenden in 1906 is credited on Christmas Eve with making the first voice transmission, wireless voice transmission, as opposed to Morse code transmissions which had been around for a while. And what was his oscillator? Well, what was his antenna? It was this 420 foot thing, that's about, I think, about 120 meters.

So if you actually put in the speed of light, you'll see what kind of antenna size you need, sorry, what kind of frequency you need to excite this with for the wavelength to be the comparable to the size of the antenna. So let's see, 3 times 10 to the 8 is speed of light, and I've got an antenna that's 120 meters. So let's say 10 to the 2, so it's about 3 times 10 to the 6 Hertz that I need to be exciting this at. Well, he wasn't able to get anywhere near that. He actually had this big electrical machine that could generate a sinusoid of about 50 kilohertz for which he'd have needed a much taller antenna to have efficient transmission, but it was enough for the signal to be picked up a few kilometers away.

And he claimed that it was heard all the way down the coast of Virginia and so on, but there's some controversy about that. Anyway, he's credited with developing a lot of the basic technology for AM and for developing these machines and all of that. I like the name of the cocktail named in his honor by the city of Marshfield. Here's a picture of the antenna from an old postcard. Looks very Cape Cod and Marshfield-y. He had a companion, a system built in Scotland, but a careless workman, at one point, disconnected a particular cable that was tying it to the ground, and the whole tower collapsed. So his transatlantic experiments were set back for a while. OK.

So how is this done? Well, for our setting, it's actually quite easy. We've got our digitized symbols coming out here. This is the x of n, something like this if we're doing bipolar signaling. And we're going to multiply it by a cosine. So here's our cosine carrier. We'll use capital omega to denote frequency in these discrete time signals. And I'm using angular frequency, so this is 2 pi times whatever other frequency you're used to thinking in terms of, but this is typical for discrete time signals. And so this is what it looks like.

This is for the specific case of the x event I showed earlier. So let me flip back to show you that. This one. OK? So we're taking this waveform and multiplying it by cosine. So what you're going to have is a burst of cosine and then 0, then a longer burst of cosine and then 0, and then a burst of cosine and then 0, and then a burst of cosine again. So that's what we're seeing here. OK.

A short burst of cosine. This is 16 samples long and then 32 samples of 0 and then 48 samples of cosine and so on. All right? So your loudspeaker is emitting power and then turning off and emitting power and turning off, and this is what gets picked up by the microphone. So the microphone has to figure out-- this is the particular case of an on-off signaling scheme. So the microphone has to figure out what's being sent. OK.

So any particular thoughts on how you might recover things at the receiving end? So what I'm not showing you is the D to A converter, which is going to take this thing and interpolate these points and put it out on real time access and put it out on the channel. OK? I'm imagining all of this kind of stuff going on, but we're just going to look from discrete signal to discrete signals, so I'm suppressing all the stuff in between.

So at the other end, you somehow-- let me assume there was no distortion on the channel, and you figured out a way to get exactly this after sampling at the receiving end. If you got the signal, what might you do to it to recover those 0 and 1? Any particular ideas? If you had to write an algorithm to do that? So I'm saying imagine no distortion on your channel. You hear a sound on your microphone, you take samples of it, you get a signal like this, and now you've got to decode. Anything? It ought to be pretty simple. Right? Yeah.

STUDENT: Take the absolute value [INAUDIBLE].

PROFESSOR: OK, yeah, I like that. So let's take the absolute value. So you might get something like this. And then there's a gap, and then you get something like this, and so on. Right? I'm drawing these as though they're continuous, but actually, what you're getting is a bunch of samples. And these are supposed to be sections of sinusoid. This is what we call a rectified sinusoid. The electrical engineers say rectified, it's just taking the absolute value. Right?

Once you have this, you're probably in better shape to try and figure out where there is signal and where there is not. So what could you do? What might you try doing? Felix? You want to continue? What's the next step?

STUDENT: So you can take the moving maximum [INAUDIBLE]?

PROFESSOR: OK. So you're saying, let's have a sliding window of some kind that looks to see how much of the signal is in there. Is there any window size that will actually give you a constant signal if you're in the body of this? I mean, if you took a window that was equal to a period here, this is periodic. Right? While the sinusoid is ringing, its periodic. So if you took a window of this size and then slid it along, at least in the body of this, you're going to get a constant because you're picking up the average value of the rectified sinusoid. And then near the ends, you're going to get some effects whatever they are.

But that should be enough to give you a good stretch of signal. And again, you're not getting a continuous green thing, you're actually getting samples. Right? But that should be enough to help you figure out where you have zeros and where you have ones. So very simple.

What if we have a bipolar scheme? Suppose we have a signal that can be plus or minus. And so what we have is not this, but we have a phase change every time we go from a 1 to 0. Then, if you take the absolute value, you've lost all the information. Right? So we've got to figure out something else. OK. So the more general way to do this, for instance, for the case where you have the bipolar transmission, is-- well, this is actually just to-- it turns out that I had the same idea as you did, which was take absolute value and then a local average over a half period. It's a very natural thing to think of as a way to extract that.

That works for the on-off signaling, but for the bipolar, it's a little trickier. So here's what a typical general demodulation scheme is for amplitude modulated signals. Here's the transmitted signal. It's been received. I've converted it from analog to digital and so on. I'm going to do the same thing again. I'm going to multiply it locally by a cosine at the same frequency. So I have a local oscillator at my receiver that's got the same frequency as the carrier frequency that was used for transmission. OK?

When you tune your radio on AM, what you're doing is actually adjusting the local oscillator frequency to match that of the station frequency. OK. And the station frequencies, by the way, should be obvious by now when you have, what is it, 820 AM or whatever, what they're announcing is the frequency of their carrier. That's how they're known. They're known by the frequency of the carrier. OK.

So here's what the result is. You've got the signal that you transmitted multiplied by the cosine. What's the signal that you transmitted? Well it was the signal you wanted to get across but multiplied by the carrier at the transmitting end. OK. You've got the cosine squared there. We have a standard identity for cosine squared. That's half of 1 plus cosine twice the angle. Right? Yeah?

STUDENT: What if you're given a phase shift?

PROFESSOR: OK, yeah. Good question. The question is what if you have a phase shift because this assumes not just that you have the right frequency but that you have exactly the right phase. And it turns out that that's something that has to be dealt with. So there are mechanisms for doing that. So basically, what you end up doing in one way of tackling that is you multiply by cosine, you multiply by a sine. And by looking at the outputs of both of those, you can actually figure out what the right phase shift is. So that's a good question.

It's not just phase shift, there's also time delays and propagation and so on, so it's a real issue. But let's just deal with the simplest case for now. OK. So what you have after the multiplication is something that actually has the signal you're interested in, that first part, just scaled by a factor of a half, and then it's got some stuff that you don't want. It's got a double frequency component which you have to try and get rid of. But the nice thing here is if you were able to get rid of the double frequency component, then you have x of n there whether or not it's plus or minus. So the sign is not lost. It's not the absolute value of x anymore that we're recovering, it's the actual x.

So your x can go positive or negative, and you'll pull it out. OK? So this is better than just taking the absolute value and doing a local filtering. OK. So here's what that looks like for the particular example we have. So you can see the double frequency cosine over here, and then the average value that that cosine is riding on is going between 1 and 0 in this particular case. OK?

So this was a 0-1 waveform that we transmitted, but we're demodulating it using a scheme that could have actually handled a signal that went negative as well. But this example doesn't have a signal that goes negative. It was the one that I showed you earlier. But you can see that the average value here is picking out exactly what you want. So your challenge now is to get rid of the double frequency piece.

So does someone want to suggest to me how you might do that computationally? Yeah.

STUDENT: Question.

PROFESSOR: Yeah.

STUDENT: If you wanted to extend [INAUDIBLE] by extra sign, why don't we want to multiply by it again?

PROFESSOR: Oh, I see what you're saying. What you're saying is, we could have done this more simply if what came in-- that's a good question. We could just divide by the cosine here and get what we want because what went down was the transmitted times the cosine. I never thought of that. Could there be a problem with it?

STUDENT: Maybe if cosine has a value of 0.

PROFESSOR: Yeah. So you see the point is that the cosine has multiple zero crossings. And in the discrete time case, of course, it depends on what that frequency is. You might not go exactly through 0. But then you're going to be horrendously sensitive to noise and other things in the system. So that's a good thought. It's like in the Viterbi case, by the way, if you're not thinking of noise, then the very simple ways to combine the parity stream to recover the input. But as soon as there's some noise, all of this falls apart. So the scheme is robust up to a point to noise. I mean, only up to a point, you know if you've listened to AM radio that it can get annoyingly noisy, but up to a point, it's fine. OK.

So we were here, and I was asking, how might you get rid of that double frequency component? Any ideas? Someone who hasn't spoken today maybe? I want to do some filtering operation on this. I want to run some algorithm on this that's going to eliminate the double frequency piece and just get me the nice waveform back. Yeah.

STUDENT: You can do the same thing.

PROFESSOR: I can do the same thing. OK. So what interval would I pick then?

STUDENT: It would be not the double frequency, the single frequency.

PROFESSOR: OK. I could get away with, let's see, I could get away with the period of the double frequency. Right. If it was equal to the period of the double frequency component, then as if I take the average with that, then I've got a full cycle of the double frequency component, and it will go off to 0. Right? So that's what I need to do. And that's the simplest way to do it. So the filter here, the simplest one, just puts out a signal that sums L plus 1 of these values where L plus 1 is 8. That's exactly the period of the double frequency component. Remember, we had 16 samples for the original carrier, so double frequency component has a period of 8.

And so the 2 omega piece gets eliminated, and here's what we get. OK. So there's some transition at the ends because when you get to the ends of something like this, and you're doing the averaging, well, now you've got a little bit of the previous bits worth and a little bit of the current bits worth, so there's going to be a transition. But it'll still leave you with plenty of room in which to make your decision as to what you have. OK.

So now what I want to do for the rest of the lecture, and we're going to continue is well into the next few lectures, is say we've understood, at least at some level, how you might get across the analog channel. So let's now just focus on input to output here. So we've got a discrete time signal here, and I get a discrete time signal coming out there. And I can think of that as my channel input and output. I can suppress all this other stuff. Knowing that all of that goes on, but in terms of my designing what I want to do with the signals, I can just look at it end-to-end.

So we're going to talk about models for end-to-end behavior from the discrete time sequence that goes in to the discrete time sequence that you reconstruct at the other end. OK? So abstractly, what we have is some system, S. We've got an input sequence, an output sequence. And this is typically how these things are drawn. I just want to caution you on this. When you see a diagram like this, you want to think of it as a snapshot of the system at a time n. You don't want to think of it as saying that the value of y at the output at some particular time n is determined by the value of x at that same time n. OK?

So the real story is that, in general for such systems, the value of the output at any particular time n is determined by all the input values of the input. If it's a causal system, then it only depends on the present and past inputs. But in some settings, you can think about non-causal kind of processing. So in general, when you see a picture like this, think of it as a snapshot of the system at time n. But it's not telling you that x and that one value gets mapped to yn that one value. That's not what's happening.

I'm a little fuzzy about this because, often, that's glossed over, and then it leads to confusion. So more, generally, if I want you to be thinking of the signal as a whole, I'll use this notation. I'll just put a dot there to say it's the entire waveform that I'm referring to because I don't want you to fixate on a particular time instant. OK. So that's our system. Here's a little more fussing about notation, but I don't think I want to bother with that now. Please look at it when you review the slides.

Let's go to talking about some particular signals, and these may be ones you've seen in 601 is convenient signals to talk about. We'll do a lot with unit step functions. So these are signals. This is a signal that is 0 for all negative time. And then at time 0, it goes up to the value 1 and then stays at 1. So that's the unit step. And our standard notation for it is u of n. So when you see u of n, that's what you want to imagine.

And u of n minus 3 then, well, this will have the same value at n equals 3 that this had at n equals 0. So the point of transition for this waveform must be at n equals 3. OK? So it's the same signal but just delayed by 3. So u of n minus 3 just has it step three instance later.

Here's another very special signal that we use a lot. It's what's called the unit sample function or unit sample signal. So it's an entire signal. It's not just one value, it's signal denoted by the symbol delta of n. It's 0 for all positive and negative time. But at time 0, it has the value 1. So that's the unit sample. And so if you had, for instance, delta of n plus 5, it's the shifted version of that function. And what happened that n equals 0 here will now happen at n equals minus 5. In other words, the spike up there is at n equals minus 5.

So we're going to need to get comfortable with unit step functions and their shifted versions and the unit sample functions and their shifted versions. OK. And there's a relation between the two as well. So you should see fairly clearly that you can actually write the unit sample function as a difference of a unit step and a delayed unit step. OK?

Now, we can do standard algebraic operations on signals. So I can take a unit step function. I can do things like u of n plus 3 times u of n minus 7. And what this means is draw the signal u of n. Draw the signal 3 times u of n minus 7, which is just u of n minus 7, each value scaled by 3, and then just add them instant by instant. So it's the most natural way of adding signals. It's exactly what you would do if you're adding or multiplying the functions of continuous time from your calculus course. It's the same idea. So we can do all of these algebraic operations.

Now, the response of a system to the unit step or the unit sample is interesting. It's not interesting for all sorts of systems, but it's interesting for the class of systems that we'll be focusing on. So let's talk about that. So the unit sample response of a system is just the output signal that you get when the input signal is the unit sample. OK? This is the traditional symbol for it, h of n. So when you see h of n, you're typically thinking of unit sample response. Similarly, the unit step response, put in a unit step. The signal that you have the output is the unit step response, and we'll use the symbol s of n for that.

So the reason that these are useful in many contexts is that you can take a general signal, for instance, the one at the top there and represent it as a weighted sum of scaled and shifted unit sample functions. OK? So this is just saying I can think of this function as being advanced unit samples scaled by something and another one scaled by something else and another one scaled by something else and so on. Here is the analytical expression that goes with that.

So any signal can be thought of as being made up of a bunch of unit samples appropriately scaled and appropriately delayed. The same thing with the unit steps. And these are waveforms, as you've seen. This is the kind of waveform we work with a lot in the context of communication. So we've got these sorts of rectangular waveforms. And it's very useful to think of them as being combinations of unit steps.

So the waveform at the top can be generated by having a unit step at this time climbing up. And then I've got to calculate its effect on this next transition, so I put a negative going unit step. And I want to bring it back up again at that time, so I put another unit step and so on. So you can synthesize a signal of this type as a linear combination of unit steps scaled and delayed.

Now, this is actually important when you get to particular classes of systems for which you can exploit these properties. So let me tell you what linearity is and what time invariance is, because the rest of our talk about systems is going to be focused on linear and time invariant systems. So let's start with what a time invariant system is.

It's just a system whose response to a given input doesn't depend on the day of the week that you do the experiment. If you come back tomorrow and do the same experiment, you'll get the same result except it's happening tomorrow instead of today. Right? So it's a system where if you delayed the input by some capital N, the response is the response you had previously but just delayed by the same amount. So time invariant system is one where the response doesn't depend on absolute position on the time axis. If you shift the input by some amount, you get the same response but shifted by that same amount. So it's a very easy idea. OK.

So for instance, if you had a time invariant system and you put in a delayed unit sample, this is a unit sample that has the value 1 at the point capital N. Your response will be the unit sample response but correspondingly delayed. Right? So time invariance of a system allows you to do that, and that's very convenient. Here's the other property that's crucial, which is linearity. And we've talked about linearity a lot along the way. Here is a definition in this context. OK.

So we say no the system S is linear if you can do the following. Take the response y1 to an experiment with an arbitrary input x1. OK? So you put in an arbitrary input x1, you get the response y1. Put in an arbitrary input x2 and another experiment get out the response y2. If it's true that any weighted linear combination of those inputs from the previous two experiments will give rise to a response that's the same weighted combination of the original responses, then you have a linear system. OK?

So if superposition of inputs, according to this formula so a weighted sum, leads to a response and output, that's the same weighted combination, then what you have is a linear system. OK? So if this is true for arbitrary inputs and arbitrary scale factors. So one important conclusion from that, by the way, is if you have a linear system, and you put in the all-zero input, the response must also be an all-zero response. And I'll leave you to think about why that might be true. OK.

So the systems that we are going to be focusing on will be linear and time invariant. OK? LTI. And so we're going to be thinking of end-to-end models of the channel, our digitized sequence xn, all the way through the processing that happened in between. OK, so what did we have? We had modulate. We had D to A conversion. We had the channel. We had A to D. We had DMod. And there was a filtering operation, as well, there as part of the demodulation. So all of that, end-to-end, we're going to try representing that as a linear time invariant model.

So it'll be an approximation. Real channels are more complicated. But it turns out that LTI is a very good place to start, and not just for the communication setting, for a whole variety of settings. So it turns out that if you're just talking about small deviations from some nominal operating point, a linear model is not bad. And the reason is, well, it goes back to Taylor series kinds of thinking. You can have a very non-linear function, but if you're looking in the neighborhood of some operating point-- stick it up there. So here's some operating point, and you're only looking at small perturbations around there. The linear approximation is not a bad one. OK?

So it's really essentially that idea that first order Taylor series kinds of approximations are good. And so linearity works. Time invariance works because many systems are inherently time invariant. Now, that's not always true for communication channels. If you've got a mobile device, for instance, the channel is changing all the time. Well, if you're mobile with your mobile device, the channel is changing all the time. So you need to reckon with time varying channels or what are called fading channels. But for many situations, a time invariant channel is reasonable.

So time invariant and linear is a good approximation. Now, as soon as you invoke linearity and time invariance, you've got such a rich structure that opens up that there's a lot you can do by way of analysis and developing computational tools and so on. And that gives you a very good handle on doing design. So if you're trying to design something that's, ideally, linear time invariant, you have a large array of tools at your disposal.

So you'll find that in engineered systems, people are trying to design modules that are thought about as LTI systems and then interconnecting them, maybe, in non-linear ways or time varying ways. OK. We'll pick up on this in recitation tomorrow and next lecture which will be Wednesday of next week. And make sure you're aware of what the portions are for the quiz and where you have to go for the quiz rooms and all of that on Thursday evening next week.