PROFESSOR: I want to pretty much finish up today talking about modulation. Part of the reason we don't spend much time on this is, I think most of you have seen some kind of undergraduate class in communication or signal processing or something. Where you do a lot of this because a lot of this is just nice exercises and -- well, nice or not nice, depending on the way you about it -- in multiplying waveforms by cosines and by sines and by complex exponentials. And you keep doing this, and doing this, and doing this and you get these long expressions. And and it all means something. So, most of you have seen a good deal of this. You probably haven't seen the Nyquist criterion before. I'm sure you haven't seen the Nyquist criterion done carefully, because I've never seen it done carefully before. I don't think it is done carefully anywhere else. And I'm sure you don't care about this. But, at some point, if you deal with this stuff, you will care about it. Because at some point you will need it.

But anyway, we were looking at pulse amplitude modulation, you remember. We were looking at the modulated waveform. And this is all done down at baseband, now, remember. And the interesting problems down at baseband are, first, how do you choose a signal set. Namely, how do you choose those quantities u sub k, out of some set of possible values. So they have an appropriate distance between them. And, next, how do you choose this waveform p of t, which is called the pulse, that you're using. And then, after you've chosen those two things, there's nothing more to be done. You simply form the modulated waveform as u of t equals the sum of the signals which are coming in regularly at intervals of t. You multiply each one of them by this waveform, p of t.

Or, alternatively, you think of the waveform as being a sum of impulses. Each impulse weighted by u sub k. And you think of passing this string of impulses through a filter with impulse response p of t. Which is usually the way that it's implemented, except of course you're not using ideal impulses. You're using sharp pulses which have a flat spectrum over the bandwidth of the filter p of t.

But anyway, somehow or other you implement this. And that is this fundamental step we've been talking about for quite a while now, of how do you turn sequences into waveforms. How do you turn waveforms into sequences, it's a fundamental piece of source coding. It's a fundamental piece of channel coding. Here we're doing the channel coding part of it.

Then we did something kind of flaky. We said that when we received this waveform, what we were going to do is to first pass it through another filter, and then sample it. If you've done the homework yet, you will probably have looked at what happens when you take an arbitrary linear operation on this received signal to try to retrieve what these signals u sub k are. And you will have found that this filtering and sampling is in fact not a general linear operation, but it's the only general linear operation that you're interested in as far as retrieving these signals from that waveform. So, in fact, this is, in fact, more general than it looks.

There a confusing thing here. If you receive u of t at the receiver, and your question is, how do we get this sequence of samples u sub k out of it, and suppose that this pulse p of t or the -- I mean, suppose for example that the pulse is a narrow bandwidth pulse, and there's just no way you can perform linear operations and get these signals out from it. Is it possible to do nonlinear operations and get the signals out from it? And you ought to think about this question a little bit because it's kind of an interesting one.

If I don't tell you what the signals u sub k are, if I ask you to build something which extracts these samples u sub k, without any idea of what signal set they're taken from, then there's nothing you can do better than a linear operation. And, in fact, if this pulse p of t has a bandwidth that's too narrow, you're just stuck.

If I tell you that these u sub k are drawn from a particular signal set -- for example, suppose they're binary -- then you have an enormous extra amount of information. And you can, in fact, given this waveform, even though p of t is a very, very narrow band, you can still in principle get these binary signals out.

So what's the game we're playing here? I mean, we're doing kind of a phony thing. We're restricting ourselves to only linear operations. We are restricting ourselves to retrieving these signals without knowing anything about what the signal set is. Which is not really the problem that we're interested in looking at. So what are we really doing?

What we're really doing is trying to look at this question of modulation before we look at the question of noise. And after we start looking at noise, the thing that we're going to find is that this received waveform is a received waveform plus a lot of noise on top of it. And if there's a lot of noise on top of it, these non-linear operations you can think of are not going to work very well. And I guess the problem is, you just have to take that on faith right now. And after we look at random processes, and after we look at how to deal with the noise waveforms, you will in fact see that these operations we're talking about here are exactly the things we want to do.

Now, I don't know whether this is the right way to do it or not. Dealing with all this phony stuff that you see in elementary courses before dealing with the real stuff. I think it probably is, because for most of you this is sort of familiar and we'll come back later and make it all right. But who knows?

Anyway. what we want to do then is to find some composite filter, g, which is what happens when you take the filter p, or the pulse p, pass it through a filter q of t, what you get is the convolution of p and q. And, therefore, what comes out after this filtering is done is a received waveform which is just a sum over k of u sub k times g of t minus k t. In other words, these two filters are not doing anything extra for you. All they are is a way of putting part of the filter at the transmitter, part of the filter at the receiver. When you look at noise you'll find out that there is some real difference between what's done at the transmitter and what's done at the receiver, because the noise comes in between the transmitter and the receiver. But for now, it doesn't make any difference so the only thing we're interested in is the properties of this composite waveform, g of t.

And what we find is that if we receive r of t, which is this waveform sum of u k g of t minus k t, and if we want to retrieve the coefficient u sub k, it becomes duck soup to do so if in fact this waveform is like a sampling waveform. In other words, if g of t is equal to 1 at t equals 0, and it's equal to 0 at each other sample point, then all we have to do is take this waveform, simply sample it each capital T seconds, and we get these coefficients out automatically.

Now, I should warn you at this point that in the notes the scaling business is not done quite right. Sometimes we talk about g of t as a filter whose shifts are orthonormal to each other. And sometimes as a filter whose shifts are orthogonal to each other. I advise you not to worry about that, because you have to make changes about five times in the notes to make it all right. And I will put up a new version of the notes on the web which in fact does this right. It's not important. It's just this old question of, do you use a sinc function when you're dealing with strictly band-limited functions, or do you multiply the sinc function by 1 over the square root of t to make it orthonormal. Or do you multiply it by 1 over t to make it -- I mean, you can scale it in a number of different ways. And, fundamentally, it doesn't make any difference. It's just that if you want to get the right answer you have to scale it the right way. And it's not quite right in the notes. So I'll change it around and send the thing out to you.

Then it says that T-spaced samples of r then reproduce u sub k without intersymbol interference. The Nyquist criterion is different from this business of the pulse being ideal Nyquist. Ideal Nyquist is talking about the time domain. It simply says the trivial thing you, want a pulse which has 0's at every sample point except for the sample point you're interested in. The Nyquist criterion translates that ideal Nyquist property, in time, to a property in frequency. And it says that the frequency, that the Fourier transform of g of t has to satisfy this relationship here. And there's this added condition on g of f that it has to go to 0 fast enough as f goes to infinity. But we won't worry about that today.

So the condition is this: the picture that I showed you last time is this. We defined the nominal band, the Nyquist band, as the base bandwidth w equals 1 over 2t. That's the bandwidth that a sinc pulse would have if you were using a sinc pulse. The actual baseband limit, b, should be close to w. And most of the work that people do trying to build filters and things like this, since what we're trying to do is find a waveform that we're trying to transmit. And we're stuck with the FCC and all these other things that say, you better keep this band-limited. What we're going to do is to make the actual band of the waveform close to the nominal bandwidth. So we're going to assume that it's less than twice the nominal bandwidth. In other words, you can have a little bit of slop, but you can't have too much.

When you try to design a filter that way, and you satisfy the Nyquist criterion, which is talking about all of these bands all the way out to infinity, and the fact they have to add up in a certain way, if you only have this band here and part of the next band -- if this function has to go to 0 before you get out to 2w, then the only thing you have to worry about is what does this waveform look like here. You take this and you pick it up. And you put it over here, and you add it up. You take this, you put it over here and you add it up. And if the pulse, p of t is real, then what you have over here is the complex conjugate of what you have here. And if you make this real in frequency also -- in other words, you have symmetry in time and symmetry in frequency, then this band edge symmetry requirement is that when you add this to this, you get something which is an ideal rectangular pulse.

Now, if you think of taking this waveform here -- now you only have to worry about the positive frequency part of it. And you take that point right there, which is halfway between 0 and t. In other words, this is at t over 2. And it's also at frequency w. And you rotate this thing around by 180 degrees, then this comes up there. And it fills in this little slot up here. That's what the band edge symmetry means. Yes?

AUDIENCE: [UNINTELLIGIBLE]

PROFESSOR: Ah, yes. We could, if we wanted to, put various notches in this filter. But we've defined the bandwidth, b, as the largest frequency, f, such that g hat of f is 0 beyond

b. In other words, everywhere beyond b, g hat of f is equal to 0. Now, if g hat f cuts down to 0, say, back here, there's no way you can meet the Nyquist criterion. Because there's no way you can build this thing up with all these out-of-band components so that you get something which is flat all the way out to w.

So you simply can't have a filter which is band-limited to a frequency less than w. What you need is to use these out-of-band frequencies as a way to help you construct this ideal rectangular pulse. Through aliasing.

In other words, the point here is when we're doing transmission of data, we know what the data is. We know what the filter is, and we can use aliasing to our advantage. When we were talking about data compression, aliasing just hurt us. Because we were trying to represent this waveform that we didn't have any control over. And the out-of-band parts added into the baseband parts and they clobbered us. Because we couldn't get the whole thing back again. Here, we're doing it the other way. In, other words we're starting out with a sequence. We're going to a waveform, and then we're trying to get the sequence back from the waveform. So it's really the opposite kind of problem. And here, the whole game, namely, the thing that Nyquist spotted, back in 1928, was you could use these out-of-band frequencies to, in fact, help you to get rid of this intersymbol interference. Because all you need to do is make these things add up to this so that you have something rectangular. And then when you do the samples, you have to write samples.

Now, the problem that a filter designer comes to when saying this is to say, OK, how do I design a frequency response which has the property that it's going to go to 0 quickly beyond w. Because the FCC, when we translate this up to passband, is going to tell us we can't have much energy outside of minus w to plus w. And if we can't design a good filter, it means we have to make w smaller. So we can keep ourselves within this given bandwidth. And we don't want to do that because that keeps us from transmitting much data. And then we can't sell our product, so suddenly we have to design something which uses all of this bandwidth that we have available.

So what we want to do, then, is to design something where b is just a little bit more than w, but where also we get from t down to 0 very quickly. We could just use the square pulse to start with. And what's the trouble with that? This rectangular pulse, its inverse Fourier transform is the sinc pulse. The sinc pulse, because a discontinuity in the Fourier transform, it can't go to 0 any faster than is 1 over t. And suddenly it goes to 0 as 1 over t goes to 0 very, very slowly. In other words, you have enormous problems over time. And you have enormous delay. And since you have so many pulses adding up together, everything has to be done extraordinarily carefully.

So what you want is a pulse which remains equal to t over a y bandwidth here. Which gets down to 0 very, very fast. So the problem is, how do you design something which gets from here down to here very, very quickly and very smoothly. You want it to go smoothly because if you have any discontinuities in g of f, you're back to the problem where g of t goes to 0 as 1 over t. If you have a slope discontinuity, g of t is going to go to 0 as 1 over t squared. If you have a second derivative discontinuity, it's going to go to 0 as 1 over t cubed. Now, 1 over t cubed is not bad, and filter designers sort of live with that. So they design these filters which are raised cosine filters, which over the band here -- someday I'll get a pen that works -- are flat. Over this band here, from here to here, this is a squared cosine, analytically. And a squared cosine is just the same as a cosine which you take and displace up so it's centered right there. Well, excuse me, it's the same as a sine which is centered there. Which is what you get when you square a cosine pulse as 1/2 plus. Anyway, it looks like this.

So our problem is, how do you design a filter which gets from here to there quickly but where the inverse transform also goes to 0 relatively quickly? Now, if you want to do this and you also face the fact that in a Nyquist criterion, any part of g hat of f which is imaginary, the Nyquist criterion says that what do you have to do in-band and what you have to do out of band have to add up there also. That doesn't help you at all in getting from this real number t here down to 0. So anything you do as a complex part of g hat of f is just wasted. I mean, your problem is getting from t to 0 with a smooth waveform.

You would like to make g hat of f strictly real. You would like to make it symmetric. Why would you like to make it symmetric? Because this thing down here and this thing up here are really part of the same problem. If you find a good way to make a function go to 0 quickly up here, you might as well use the same thing over here. So you might as well wind up with a function which is symmetric and real. That leads us into the next thing we want to look at. That's a slightly flaky argument. We're going to find a better argument as we go.

What we've said is, the real part of g hat of f has to satisfy this band edged symmetry condition. Choosing the imaginary part unequal to 0 simply increases the energy outside of the Nyquist band. You don't get any effect on reducing delay out of that. Thus, we're going to restrict g of f to be real. And we're going to also restrict it to be symmetric. Although that's less important.

Now, when we start to look at noise, we're going to find out something else. We're going to find out that we want to make the magnitude of p of f equal to the magnitude of g of f. Now, magnitude doesn't make any difference. So we want the frequency characteristic of p of f to be the same as the frequency characteristic of q of f. In other words, there's no point descending a p of f which is a perfect sinc function and then using a very sloppy q of f. Because that's kind of silly. There's no point to using it very sloppy p of f and then using a very sharp q of f, because somehow when you start looking at noise, you're going to lose everything. Because the noise gets added to this pulse that you're transmitting. So, what we're going to find when we look at this later, is we really want to choose the magnitudes of these to be equal.

Since g hat of f is equal to this product, and since we've already decided we want to make this real, what this means is that q hat of f is going to be equal to p complex conjugate of f. What that means is the filter q of t should be equal to the complex conjugate of p of minus t. You take a p of t, and you turn it around like this. If p of t is real, this is called a matched filter. And it's a filter which sort of collects everything which is in p of t, and all brings it up to 1p, which is what we would like to do here.

So, anyway, when we do this it means that g of t is going to be this convolution of p of t. And q of t, which we can now write as the integral of p of tau, times p complex conjugate of t minus tau, p tau. And what we're interested in is, is this going to be ideal Nyquist or not. And what does that mean if it is ideal Nyquist?

If g of t is ideal Nyquist, it means that the samples of g of t, times k times this signaling interval, t, have to have the property that these samples are equal to 1 for k equals 0, and 0 for k unequal to 0. What does that mean? If you look at this, that kind of looks like these orthogonality conditions that we've been dealing with, doesn't it?

So that what it says is that this set of functions, p of t minus k t. In other words, the pulse p of t and all of it shifts by t, 2t, 3t, and everything else. This set of pulses all have to be orthogonal to each other. And the thing which is a little screwed up in the notes is whether these are orthogonal or orthonormal, or what. And you need to make a few changes to make all of that right.

These functions are all real L2 functions. But we're going to allow the possibility of complex functions for later. In other words, if we're transmitting a baseband waveform on a channel, how do you transmit an imaginary waveform? Well, I've never seen anything in electromagnetics, or in optics, or anything else, that lets me transmit an imaginary waveform. These are not physical. We will often think of baseband waveforms that are imaginary. Real and imaginary complex. And when we translate them up to baseband, we'll find something real. But the actual waveforms that get transmitted are always real. There's no way you can avoid that. That's real life. Real life is real. That's why they call it real, I guess. That's why they call it real life. I don't know. I mean it's more real than something imaginary, isn't it? So, anyway, what gets transmitted is real. But we'll allow p of t to be complex just for when we start dealing with something called QAM, which is our next topic.

So in vector terms, the integral of u of tau times q of k t minus tau is the projection of u of t onto this waveform. And partly for that reason, q of t it's called the matched filter to p of t. In other words, you use this waveform here as a way of selecting out the parts of this waveform u of tau, u of t, that we're interested in. So that any way you look at it, we're going to use a pulse waveform, p of t, which has this property that its shifts are all orthogonal to each other.

When we start studying noise, you will be very thankful that we did this. Because when you use pulses that are orthogonal to each other, you can break up the noise into an orthogonal expansion. And what goes on at one place is completely independent of what goes on at every other place. And we'll find out about this as we go.

But, anyway, we have the nice property now, that anytime we find a function g of t, that satisfies the Nyquist criterion And any time we choose p of t and g of t so that their Fourier transforms have the same magnitude, then presto, we have freely gotten a set of orthonormal functions. Which just comes out in the wash. Before we worked very hard to get these truncated sinusoid expansions and sinc weighted sinusoid expansions, and all of this stuff to generate different orthonormal sets of waveforms. Suddenly, we just have an orthonormal set of waveforms and a very large set of orthonormal waveforms popping up and staring us in the face here. And in fact these are the waveforms we're going to use for communication, so they're nice things. Nobody uses sinc functions for communication. Nobody uses rectangular functions. You can't use either one of them because rectangular functions have lousy frequency characteristics. Sinc functions have lousy time characteristics. These functions p of t are sort of nice compromises. And they're orthonormal, again.

Let's go on to the rest of modulation. We've been talking about baseband modulation. And when we're thinking about PAM, pulse amplitude modulation, we are thinking in terms of this sequence of symbols coming in. The symbols being turned into signals. The signals been turned into waveforms. And what comes out here, then, is some baseband waveform. That's what the Nyquist criterion is designed for. How do you make a baseband waveform which is very sharply cut off in frequency?

Usually what we want to transmit is something at passband, so we somehow want to take this baseband waveform, convert it up to passband. We're then going to transmit it on a channel. I mean, why do we have to turn it into passband anyway? Well, if you did everything at baseband, you wouldn't have more than one channel available. I mean, wireless, you know you have all these different channels. The way it's done today, you use something called CDMA, where you're not breaking into narrow channels. But you should understand how to break it into narrow channels before understanding how to look at it as co-division multiple access. If you're using optics, you want to send things in different frequency bands. Whether it's optics or electromagnetics simply determines the frequency band you're looking at anyway. You can't propagate things in every frequency band. Things don't propagate very well at baseband.

So for all of these reasons, we want to convert these baseband waveforms to passband. Why don't we generate them originally at passband? Because things are changing too fast there. I mean, you want to do digital signal processing to massage these signals, to do all the filtering. To do most of the other things you want to do. And you can do those very easily at baseband. And it's hard to do them at passband. So the generic way things are done is to first take a signal sequence. Convert it into a baseband waveform. Take the baseband waveform, convert it up to passband. And the passband is appropriate to whatever the channel is. You send it. You take it down from passband back to baseband, and then you filter and sample and get the waveform back again.

You don't have to do this. We could generate the waveform directly at passband. There's a lot of research going on now trying to do this, which is trying to make things a little bit simpler. Well, it's not really trying to make things simpler. It's really trying to trying to pull a fast one on the FCC, I think. But, anyway, this is being done. And it doesn't go through this two-step process on the way. So it saves a little extra work.

So what we're going to do with our PAM waveform, them, we're going to take u of t, which is the PAM waveform. And I'm sure you've all seen this. I hope you've all seen it somewhere or other. Because everybody likes to talk about this. Because you don't have to know anything to talk about this.

So you take u of t. We multiply it by e to the 2 pi i f c t. In other words, you multiply it by a sine wave. A complex sine wave. When you do this, this thing is complex. You can't transmit it. So what do we do about that? Well, this is real. This is complex. If we add the complex conjugate of this, this plus its complex conjugate is real again. So we transmit this times this complex sinusoid, plus this other complex sinusoid, which is the complex conjugate of this. And you get 2 u of t times the cosine of 2 pi f c t. Which is just what you would do if you were implementing this anyway. You take the waveform u of t, you multiply it by cosine wave at the carrier frequency. And bingo, up it goes to carrier frequency. This is real. This was real. And everybody's happy.

And in frequency, what this looks like, since all we're doing here is just, by this shift formula that we have for Fourier transforms, the multiplying a time waveform by an exponential -- by a complex sinusoid, is simply is the same as shifting the frequency response. So the Fourier transform of this is u hat of that minus f c. The Fourier transform of u f t times this is u hat of f the plus f c. And you start out with this waveform, whatever that shape is here. This, I tried to draw to satisfy the Nyquist criteria, which it does it. Satisfies that band edge symmetry condition.

So this gets shifted up. It also gets shifted down. And the transmitted waveform then exists in the band which I'll call b sub u. b sub u is the bandwidth of u of t. Namely, it's this baseband bandwidth that we've been talking about.

But, unfortunately, when we do this thing shifted up and this thing shifted down, the overall bandwidth here is now twice as much as it was before. Now, every communication engineer in the world, I think, measures bandwidth in the same way. When you talk about bandwidth, you're always talking about positive bandwidth. Because, back a long time ago, communication engineers didn't know about complex sinusoids. So everything was done in terms of cosines and sines. Which was very good, because back then communication engineers didn't have much else to do. So they had to learn to write everything twice. And now, since we have so many other things to worry about, we want to use complex sinusoids and only write things once. Well, in fact we have to write it twice here, but we don't write it twice very often.

But, anyway, when this thing, which exists for minus b u up to plus b u gets translated up in frequency, we have something which exists from f c minus b u to f c plus b u. And this negative band is down here.

Now, we're going to assume everywhere, usually without talking about it, that when we modulate this up in frequency, that the bandwidth here, this b u here, is less than the carrier frequency. In other words, when we translate it up in frequency, this and this do not intersect with each other. If this and this intersected with each other, it would be something very much like aliasing. You just couldn't to sort out from this, plus this, what this is. Here, if I drew it on paper at least, if you tell me what this is, I can figure out what that is. Namely, demodulating it independent of how we design the demodulator is, in some sense trivial, too. You just take this and you bring it back down to passband again.

Well, anyway, since communication engineers define bandwidth in terms of positive frequencies, the bandwidth of this baseband waveform is b sub u. The bandwidth of this waveform is 2 b sub u. You can't get away from that. You have doubled the bandwidth, and you wind up with this plus this. And this looks kind of strange. So let's try to sort it out.

The baseband waveform is limited to b. If it's shifted up to passband, the passband waveform becomes limited to 2 b. Might as well put these little u's in here. Because putting in little u's here is a way of getting around the problem of talking about baseband waveforms for a while. And then talking about passband waveforms. And one of them is always twice the other one.

If you filter out this lower band here, now, what's the lower sideband here? Who thinks the lower sideband is this? Who thinks the lower sideband is this little thing here? Well, you all should think that because that's what it is. So when people talk about sidebands, what they're referring to, it's not, this is one sideband and this is another sideband. What they're referring to is this is one sideband and this is one sideband. This is stuff you all know, I'm sure. But haven't thought about for a while.

If you filter out this lower sideband, then this resulting waveform, which now runs only in this upper sideband here, and since it has to be real it has this accompanying lower sideband down here going with it, but you then have the frequency band b sub u like you had before. So, in principle, you can design a communication system by translating things up in frequency by the carrier, and then chopping off that lower sideband. And then you haven't gained anything in frequency. And everything is essentially the same as it was before.

Now, this used to be a very popular thing to do with analog communication. Partly because communication engineers felt the only thing they had to study back then was, how do you change things in frequency and how do you build filters. So they're very good at this. They love to do this. And this was their preferred way of dealing with the problem. They just got rid of that sideband, sent this positive sideband, and then somehow they would get it back down to here.

Single sideband is hardly ever used for digital communication. It's not the usual way of doing things. Partly because these filters become very tricky when you're trying to send data at a high speed. All sorts of noise in here when you try to do this and you don't do it quite right. Affects you enormously, and people have just seen over the years that those systems don't work as well as the systems which do something else that we'll talk about later. Namely, QAM, which is what we want to talk about.

If you don't do this filtering, you call this system a double sideband pulse amplitude modulation system. Which is what happens when you use pulse amplitude modulation. Namely, this baseband choosing a baseband pulse, which is the thing we're interested in. Because that's where the Nyquist criterion and all this neat stuff comes in. And then you translate it up in frequency. And you waste half the available frequency.

If you don't care about frequency management, this is a fine thing to do. Nothing wrong with it. You just waste some frequency. It's the cheapest way to do things. And there are lots of cheap communication systems which do this. But if you're trying to send data, if you're concerned about the frequency efficiency of the system, then you're not going to do this.

So what we're going to do is do something called quadrature amplitude modulation. QAM, which is what quadrature amplitude modulation stands for, solves the frequency waste problem of double sideband amplitude modulation by using a complex baseband waveform u of t. Before, what we were talking about is these signals which were one-dimensional signals. We would use these one-dimensional signals to modulate this waveform p of t. And we wound up with a real waveform. Now what we're going to do is use complex signals, which then have two dimensions. Use them to modulate the same sort of pulse, p of t, usually. And wind up with a complex baseband waveform. And then we're going to take that baseband waveform, translate it up in frequency.

So when we do this, what do we get? We need a waveform to transmit which is real. So we're going to take u of t, which is complex. Translate, shift it up in frequency by the carrier frequency. So we get u of t times e to the 2 pi i f c t.

To make it real, we have to add all this junk down at negative frequencies, which we'd just as soon not think about if we didn't have to. But they have to be there. So our total waveform is x sub t equal this sum of things.

When you look at this and you take the real part of this, the real part of this, the imaginary part of this, and the imaginary part of this, as I'm sure most of you have seen before, x of t becomes 2 times the real part of u of t. Times this complex exponential. Which is equal to 2 times the real part of u of t times this cosine wave minus 2 times the imaginary part of u of t times a sine wave. Which says, you take this real part of this baseband waveform you've generated. You multiply it by cosine wave. You take the imaginary part, and you multiply it by a sine wave.

For implementation, is one thing going to be real and the other thing imaginary? No. You can't make things that are imaginary, so you just deal with two real waveforms.

And you call one of them the real part of u of t. You call the other one the imaginary part of u of t. And the imaginary part of u of t is in fact a real waveform again. So all this imaginary stuff is just in our imagination. And the actual waveforms look like this. You take one waveform which is generated. Multiply it by cosine. Take another waveform. Multiply it by sine.

What about these factors of two here? The factors of two are things that drive everybody crazy. Everyone I talk to, I ask them how they manage to keep all this straight. And they all give me the same answer: they say they can't keep it straight. It's just too hard to keep it straight, and after they're all done, they try to figure out what the answer should be by looking at energy or something else. Or by just fudging things, which is what most people do.

And part of the trouble is, you can do this two ways. You can do it three ways, in fact. You can either I want to view x of t as being some real function times the cosine wave and leave out that 2. And some other function, imaginary part of u t times the sine, and leave out the 2 there. And many people do that. And would that be better? But when you put the 2 in with the cosines and the sines, you have to put a 1/2 in here and a 1/2 in here.

Most people, when they think about these things for a long time, they find it's far more convenient to be able to think of this positive frequency part of x of t as just u of t translated up in frequency. In other words, they like this diagram here. Which says you take this. You translate it up. And after you translate it up, you create something else down here, to make the whole thing real. But what we think of is this going up to this all the time. So that's one way of doing it.

The other way of doing it is thinking in terms of sines and cosines, removing that 2 here. And who can imagine what the third way of doing it is?

Just split the difference. Which means you put a square root of 2 in. And, in fact, that makes a whole lot of sense. Because then when you take the waveform u of t, translate it up in frequency. Make it real, you have the same energy in the baseband waveform as you have in the passband waveform. I'm not going to show that. You can just figure it out relatively easily. I mean, you know that the power in a cosine wave is 1/2, the power in a sine wave is 1/2. So when you're multiplying things by 1/2 in here -- well, this has a power of 1/2. This has a power of 1/2. And when you start looking at power here, you find out that that has to be a square root of 2 rather than 2.

So there are three ways of doing it. People do it any one of three different ways. It doesn't make any difference, because any paper you read will start out doing it one way and then, as they go through various equations, they will start doing it a different way. And these factors of 2 multiply and multiply and multiply. And in big complicated papers, sometimes I've found that these add up to a factor of 8 or 16 or something else. By the time people are all done. And we will explain later why, in fact, you don't really care about that very much. But you can't just totally ignore those factors, so anyway.

This is the way we will do it. We will try to be consistent about this, and usually we will be.

The way we want to think about this conceptually is that quadrature amplitude modulation is going to take this complex waveform u of t. It's going to shift it up in frequency to f sub c, and then we're going to add the complex conjugate. Add it to form the real x sub t.

In other words, we're going to think of it as a two-stage operation. First you take waveform, you translate it up. Then you take the real part or something, or add the negative frequency part. And we're going to think both of this double operation of 1 going from u of t to the positive frequency part of things. And then, of looking at the real waveform that corresponds to that.

What we're going to be doing here in terms of all of this is, we're going to start out with binary data. From the binary data, we're going to go to symbols. And we're going to go to symbols by taking a number of binary data -- a number of binary digits. Framing then into b tuples. Each b tuple will correspond to a set of 2 to the b symbols. We're going to map these symbols into complex signals. We're going to map the complex signals into a baseband waveform u of t. We're going to map the baseband waveform u of t into this positive frequency waveform u of t times this complex sinusoid. And finally we're going to add on the negative frequency part to wind up with x of t.

What do you think we do at the receiver? As always, we do just the opposite. Namely, one of the reasons for wanting to think about this this way, is we want to use this layering idea. And the layering idea says, you start out with the received waveform x of t. And later on we'll have to add the noise to it. You go from there to the positive frequency part. You go from the positive frequency part. You shift it down to u of t. How we got from here to there, I'll explain in a minute. You go from here down to baseband again. You go from the baseband to the complex signals, which we're going to do simply by filtering and sampling. We go from the complex signals to the symbols, which is in fact a trivial operation. It's just a look-up operation. And then from there we un-segment things into binary digits again.

So that's the whole system. And it has all these different pieces to it. I couldn't draw it as our favorite kind of diagram, because it has too many blocks in it. So that has to do.

What we're going to do now is look at each of these pieces one at a time. And the first part is the complex QAM signal set. And, just for some notation here, so we'll be on the same page, but we use r to talk about the bits per second at which data is coming into this whole system. That's the figure you're interested in, when everything is done. How many bits per second can you transmit?

We're going to segment this into b bits at a time. So we're going to have a symbol set with 2 to the b elements in it. We're going to map these m symbols, which are binary b tuples, into elements from the signal set. The signal rate, then, is r sub s, which is r over b. This is the number of signals per second that we're sending. In other words, t, this signal interval that we've always been using in everything we've been doing, is one over r sub s. So t is the signal interval. Every t seconds, you've got to send something. If you didn't send something every t seconds, the way this stuff coming in from the source would start piling up and your buffers would overflow and it wouldn't work.

The signals u sub k are complex numbers. Or real 2-tuples. So we can, when we're trying to decide what signal set we're using, we can just draw our signals on a plane. The signal set is a constellation, then, of m complex numbers or real 2-tuples. So the problem of choosing the signal set is, how do you choose m points on a complex plane. What problem is that similar to? It's similar to the quantization problem where we were trying to choose m representation points. And it's very close to that problem. It's a very similar problems. Has a few small differences, but not many.

But, before getting into that, we want to talk about a standard QAM signal set. In a minute I'll explain why people do that. And, as you might imagine, a standard QAM is just a square array of points. It's the simplest thing to do, and sometimes the simplest thing is the best.

So it's determined by some distance, d, that you want to have between neighboring points. And given that distance, d, you just create a square array here. The square array means that m has to have an integer square root. This is drawn for m equals 16.

If you look at this, you see that the real part of this it's the standard PAM set. The imaginary part is a standard PAM set, which says you can deal with the real part and the imaginary part separately. You take half the bits coming in and you choose your real part signal. Take the other half of the bits coming in. You form your imaginary part signal, and bingo, you're all done.

The energy per 2D signal, we can find the energy for 2D signal by looking at it this way. It's two PAM systems running in parallel to each other. For the PAM system, the energy in one of these dimensions is then d squared times the square root of n squared, minus 1 divided by 12.

But now we want to look at the energy which we have in both the real part and the imaginary part. So we need this extra factor. Well, we need to add together two of these things. So we wind up with d squared times m minus 1 divided by 6. Big deal.

Choosing a good signal set is similar to choosing a 2D set of representation points in quantization. If you like to optimize things, you see this problem and you say, gee, at least there's something I can put my teeth into here. What's the best way to choose a signal set? And we found that for quantization that wasn't a terribly nice problem, although at least we had things like algorithms to try to choose reasonable sets. And we then looked at entropy quantization and things like this, and it was a certain amount of fun.

Here, this problem it's just ugly. There's no other way to express it. I had to be convinced of this. I once spent an inordinate amount of time trying to find the best signal set with eight points in it, in two dimensions. How do you put eight single points in two dimensions in such a way that every point is distance at least d from every other point, and you minimize the energy of the set of points?

The answer is just absolutely ugly. It has no symmetry. Nothing nice about it. You do the same thing for 16 points, and it's just an ugly problem. You do the same thing for any number of points. Except for four points. For four points, it's easy. For four points, you use standard QAM and it's the best thing to do. And that problem is easy. But you know, that's not much fun. Because you say, bleugh. So, partly for that reason, people use standard signal sets. Partly because you don't seem to be able to gain much by doing anything else.

So that's about all we can say about standard -- oh, with eight signals, you can't use a standard signal set. That was one reason we had to worry about it. Back a long time ago, we were trying to design a 7200 bit per second modem. Back in the days when people did 2400 bits per second. And we managed to do 4800 bits per second by using QAM, big deal. And then we said, well, we can pile in an extra bit by using three bits per two dimensions instead of two bits. And spent this enormous amount of time trying to find a sensible signal set. I don't even want to tell you what it was, because it wasn't interesting at all. So, enough for signal sets.

The next thing is, how do you turn the signals into complex waveforms? Namely, how do you go from the signals in two dimensions, complex signals into a baseband waveform u of t? Well, fortunately, Nyquist's theory is exactly the same here as it was when we were dealing with PAM. Everything we said before works here. The only difference is that you don't have to choose the pulse p of t to be real. But if you look back at what we did, we didn't assume that p of t was real before, anyway. We just said, you might as well choose it to be real. But you don't have to choose it to be real.

Bandedge symmetry requires that g of t be real. Anybody know why that is? When you choose g of t to be real, the negative frequency part is the complex conjugate of the positive frequency part. Which is why, when we took this out-of-band stuff at negative frequencies, piled it into the positive frequencies, we got the same thing as if we simply rotate it around on the positive frequency. So that bandedge symmetry condition really requires that g of t be real.

The orthogonality of t a t minus k t, this set of waveforms, requires g of t to be real. Neither of these things require p of t to be real. You can choose p of t to have any old phase characteristic you want to, but if we're choosing p of t -- if we're choosing p hat of f magnitude to be the square root of a Nyquist waveform, then you can choose this phase to be anything you want to make it. But you're just restricted in, aside from the phase, you're somewhat restricted in what p of t can be.

OK. So we're going to make the nominal passband, Nyquist band, with 1 over t. Before we made the passband -- before we made the baseband bandwidth 1 over 2t. When we go up the passband we double the bandwidth so the Nyquist bandwidth is now 1 over t. The passband bandwidth is 1 over t. That's the only thing that's changed. Usually people design these filters, which they design at baseband, to go 5-10% over the Nyquist band. In other words, these filters are very, very sharp, usually.

I mean, once you design a filter, it doesn't cost anything. You put it on a chip and that's the end of it. And the cost of it is zero. So it's just the cost to design it, so you might as well make it small.

So finally, we want to go to base, from baseband to passband. We talked about this a little bit. In terms of these frequencies, the baseband frequency b sub u, we want to assume that that's less than the carrier frequency. This is this condition that we needed to make sure that the positive frequency part -- ah, here it is. That's the condition that makes sure that this is separated from that, and doesn't cause intersymbol interference between the two.

So, everything we do, we'll make this assumption. I mean, a part of this is, if you're going to modulate with such a small carrier frequency, you might as well not do it at all. You might as well just generate the waveform you want directly. Because you don't gain that much by doing it at baseband. Because you don't really have a baseband in that case.

So, u of t times e to the 2 pi i f c t is strictly in the positive frequency band, then. And these two bands don't overlap. As I said before, we're going to view this as two different steps. The first step is, I'm going to take this complex waveform, u of t. Multiply it by a complex sinusoid, which shifts me up in frequency. Just going to call that u passband of t. This is this passband signal that I want to think about. The thing that's up in positive frequencies. I'm going to ignore the thing at negative frequencies. And then I'm going to form the actual waveform x sub t, as this plus its conjugate.

If you think a little bit now, you can see that since these two bands are separated, if you think in terms of complex waveforms, how do you retrieve this band up here by a waveform which has both bands in it? Well, you filter out what's at negative frequencies, OK? So we want to design a filter which filters out the negative frequencies and x of t, and only leaves the positive frequencies. In other words, you want a filter whose frequency response is just 1 for all positive frequencies, 0 for all negative frequencies.

And that filter is called a Hilbert filter. Have any of you ever heard of a Hilbert filter before? I don't know of anybody that's ever built one. And we'll see why they don't built them in a while. But it's a nice idea. I mean, if you try to build one you'll find that it's harder to build -- you'll find that you have to implement four real filters in order to implement this filter. So we'll find out that's not the thing to do. But it's nice conceptually because it lets us study things like energy, power, and linearity, and all of these things.

So the transmitter then becomes this thing you start out with a complex waveform of baseband. You shift it up in frequency. This gives you this high frequency waveform, u sub p of t. You then take 2 times the real part of that to find the real waveform. We won't worry about how to implement this, you just do it. This passband waveform, you then pass it through this Hilbert filter, which just chops off the negative frequency part of it. Gives you a complex waveform again. You multiply by e to the minus 2 pi i f c t, which takes this positive frequency waveform, shifts it back down to baseband.

So this is a nice convenient way of thinking about, how do you go from baseband up to passband and passband down to baseband again.

Now. If you want to view these vectors here as vectors, want to view u of t as a vector in L2, there's an important thing here going on. We'll have to talk about it a good deal later on. This is a complex waveform. You want to deal with it as a vector in complex L2. In complex L2, when we're dealing with vectors, we have scalars, which are complex numbers. When we start dealing with real parts of these things, we want to view the real parts as being elements of real L2. Where the scalars are real numbers.

And what this says is that real L2 is not a subspace of complex L2. It's not a subspace because the scalars are different. This might sound like mathematical nitpicking. But, put it in the back of your mind. Because at some point it's going to come up and clobber you. And at that point, you will want to think that in fact real L2 is not a subspace of complex L2. When we start thinking about orthonormal expansions for u of t and orthonormal expansions for x of t, in fact you have to be quite careful about this. Because you take an orthonormal expansion here, translate it up into frequency. And you wind up with a bunch of complex waveforms. And they aren't real waveforms. And funny things start happening. So we'll deal with all of that later. This is just to warn you that we have to be careful about that.

This is not the way people implement these things. Because these Hilbert filters are in fact for real filters. So the implementation is what you've seen. I mean, the implementation is old. And it's the way you want to build these things. You start out with two real baseband waveforms. One which we call the real part of u of t. One which we call the imaginary part of u of t. In this single diagram, one of them is the stuff that goes this way. And the other one is the stuff that goes this way. And if, in fact, you're using a standard QAM signal set, the two are completely independent of each other.

So the real part of u of t is just the sum of these shifted pulses times the real parts. This is the sum of the shifted pulses times the complex parts. The defined u sub k prime is a real part, and u sub k double prime is the imaginary part. In the notes, this and this are called a sub k and a sub k double prime. Which doesn't correspond to anything else. So, this is the correct way of doing it. But, anyway, when you get all done, x sub t is 2 times the cosine of this low pass modulated PAM waveform. Minus 2 times the sine of this low pass PAM modulated waveform.

So, QAM, when you look at it this way, is simply two different PAM systems. One of them modulated on a cosine carrier, one of them modulated on a sine carrier.

And the picture of that is this. Aargh. Can't keep my notation straight. I'm sure it doesn't bother most of you that much, but it bothers me. All of those a's should be u's. They were a's last year, but they don't make any sense as a's.

So the thing we're going to do now is we start out with the sequence of signals. The real part of the signals and the imaginary part of the signals. This is why it's called double side band quadrature carrier, because in fact we're doing two different things in parallel.

We generate this as a pulse waveform. We filter it by p of t. We're thinking of p of t as a real waveform now. If you want p of t to be complex you have to modify this all a little bit. But there's no real reason to make p of t complex anyway.

So when you get out of here, what you have is just this low pass real waveform, real PAM waveform. Here's another low pass real PAM waveform. You module this up by multiplying by cosine of 2 pi f c t, in fact, by 2 cosine of 2 pi f ct . You modulate this up by multiplying by minus sine. And you get the actual waveform that you're going to transmit.

How do you demodulate this? Well, again, I'm sure you've seen it in one of the undergraduate courses you've taken. Because if you take this waveform, which is the sum of this and this, and you multiply this by cosine of 2 pi f c t, what's going to happen? Taking this waveform and multiplying it by cosine is going to take this cosine waveform. Half of it goes up in frequency by f sub c. The other half goes down in frequency by f sub c of t. When you multiply by sine, the same thing happens. And all of the stuff at this double frequency term all gets filtered out. I mean, you have enough filtering to just wash that away. And you wind up, just with this one waveform which is the result of this. Another waveform which is the result of this.

You can show that the two don't interfere at all, and you just have to do the multiplication to find this out. It looks a little bit like black magic when you look at it like this. Because when you're multiplying by a cosine wave, I mean it's easy to see what cosine squared does here. But it's a little harder to see what happens to all of this. And when we look at it the other way, which was this Hilbert filter kind of thing, when you look at in terms of the Hilbert filter it's quite clear that you can filter out the lower sideband and then you can just go back down to baseband again. So it's very clear that the whole thing works. Except you wouldn't implement it this way. Here you have to be more careful to see that works. But in fact you would. Well, after you get all done, then you get these baseband PAM waveforms back again. You sample them after filtering. And you're all done.

With that, we are almost done with what we want to do with modulating up to passband and down to baseband. We'll spend a little bit of time reviewing a couple of minor points on this next time. Like, I guess, the main thing we have to talk about is how do you do frequency recovery, which is kind of a neat thing. And then we'll go on to talking about random processes and how you deal with noise. So.

If you want to read ahead, we will probably have the notes on random processes on the web sometime tomorrow afternoon or Sunday. Thanks.