All right. Good morning.

Let's get started. So the last lecture we showed you how to go digital. The fact that going digital had some key benefits for us. And what we'll do today is go inside the digital gate.

Let's do a quick review. We began life by observing nature. We said those Maxwell's equations are tough. Let's simplify our lives by discretizing or lumping matter. So we got the lumped circuit abstraction. Then we had this noise problem here. In order to be able to handle that let's do some more discretization, some more lumping. So we said let's discretize values and deal with two levels, a high and a low.

That's where the binary voltage levels come up, a high level and a low level. And then we said that in discretizing it we have to make some assumptions.

We have to impose some constraints on ourselves.

Just as with the lumped matter discipline, we imposed a couple of constraints in going from the continuous matter world to a lumped matter world. Similarly, we have to impose some discipline on ourselves, some constraints on ourselves in going from the continuous value regime to the digital value regime. And that discipline is called the static discipline.

And what the static discipline says is that if you have senders and receivers in a digital system then they all need to adhere to some standard. If I was a sender I had to adhere to some tough output standards.

I had to be sure to shift values that exceeded some high voltage threshold. And if I was sending a low value I had to make sure my values were lower than some output low voltage threshold. Similarly, if I was the receiver then I had to guarantee to recognize as a one all voltages that where above some input high voltage threshold.

And similarly I had to guarantee to recognize as a zero voltages that were below some input low voltage threshold.

So provided senders and receivers in a system adhere to these voltage levels, to this discipline then they would all very comfortably work correctly in a digital system.

Then we also said that once you deal with such values, one you deal with digital values we can now postulate a bunch of digital elements that process these values in a manner very reminiscent of our analog circuits where we get analog signals. And you've already learned how to process analog signals. You've learned about resistor dividers and so on and so forth. You feed in an analog signal and you get an output analog signal as well.

Now, here the resistor in the analog domain, elements like resistors and voltage sources were the symbols that you dealt with. Here, in the digital domain, the primitive elements that we will be using are called gates.

As one example, this is called the NAND gate.

So we looked at the AND gate in the previous lecture.

This is an example of another gate called the NAND gate.

The NAND gate has the following truth table.

Our two inputs A and B and this output C.

And the NAND gate works as follows.

The output -- In English I can describe its properties as the output is a high at all times when at least one of these inputs is a low value.

So it's high whenever at least one input is a low.

So it's high here. It's high here.

Oops, it's high here, high here.

And when, oops. And when both inputs are a high the output is a low. This is a NAND gate.

Notice that these are exactly complimentary to the AND gate.

The AND gate outputs were 0-0-0-1.

And the AND gate symbol looked like this.

In general, notice that this little bubble here, it's called a bubble. That bubble implies a negation, an inversion. So we take the AND gate, invert the output and negate the output and you get the NAND gate. So these elements are combinational gates. And in combinational gates they adhere to two properties. One is that they must satisfy the static discipline.

All the systems, all the elements in our repertoire in the digital domain need to satisfy the static discipline. And the properties of a combinational gate are that its outputs are a function of inputs alone.

In other words, it doesn't store any state or doesn't store any history inside it.

You can figure out its output just by looking at the inputs at that instant. Think of it as a completely transparent entity where its output reflects some function of the inputs at every instant of time.

So I'll show you an example of a digital circuit.

So much as I could interconnect resistors and voltage sources and current sources to build analog circuits, I can now build digital circuits using primitive elements such as these. So, for example, I could build a simple circuit that looked like this, two inputs A and B here, I get an output.

And I feed that to another NAND gate with another input C.

This device is called an inverter.

The inverter simply flips the sense of the input.

So if C is a 1 the output is a 0, if C is a 0 the output becomes a 1. It's an inverter.

It simply inverts its input. Yet another primitive device.

And this is my output D. So there are three gates in this design. And I can quickly write down what the output looks like using some very simple Boolean algebra or dealing with Boolean values here.

So for AND gate the output is A and B.

Remember dot is a short form for and.

But there's a negation, inversion, so represent inversions with a bar. So my output is A dot B bar.

There is a C here. So this is my output C bar.

And this is a NAND gate. So it takes one input A dot B.

It takes the second input C bar and ANDs those and inverts them.

So that's the output. So there are three gates in this example. So you can think of building very complicated circuits containing large numbers of gates. In fact, the microprocessors that you use in your laptop contain a large number of gates.

Can someone guess how many gates are in the Pentium IV, roughly? Approximate, how many? How many gates in a Pentium IV?

40 million. 100 million.

In the Pentium IV you have on the order of 20 million gates.

20 million gates in the Pentium IV.

And life begins in 002. Here you learn about onsies and twosies, and in the real world you will be dealing with tens of millions of gates. But this is for the Pentium IV.

My research group at Laboratory for Computer Science built a chip called the Raw chip. And this chip has 3 million gates. And so there are several undergraduate students involved in this project in their third year, and they're beginning to deal with millions of gates.

So the key thing to remember is that 002 provides the foundations where you make the switch from the analog signal to the digital signal or from continuous matter to lumped matter. And learn about the foundations of these primitive elements. And by the end of this course you will begin dealing with small systems, analog systems that contain on the order of 10 to 20 primitive elements. You will also begin dealing with small digital systems that contain tens of gates.

In your final project you will build a mixed signal circuit involving an audio playback system.

You will have digital data stored in a memory chip and you will build a circuit to extract that data, filter it and then convert it to the analog domain and then play it on a set of speakers. And that has on the order of about 50 to 100 primitive elements.

So by the end of 002 you will have learned to deal with hundreds of elements. And then you will take other courses like 004 and so on where you will then make the leap to learn further abstractions that will take you from subsystems to systems with millions of gates. So the key is to manage the complexity of dealing with millions of gates it's all about abstractions. You have to build abstractions and double abstractions so you can deal with complexity.

So the rest of EECS will take you from three gates to 20 million gates and software systems that operate on 20 million gates or whatever. So there is still a ways to go.

Lorenzo, our friend has gone to bring a demonstration that we forgot to bring today. That will show you that little digital circuit in a mock up form.

So what's today's lecture about?

Today's lecture is going to be about what's inside a gate?

How to build a gate. Once you build a gate you can then put millions of them into computer systems or analog systems or other sorts of systems.

And what we'll do here is understand what's inside this abstraction. This is an abstract element that looks like a little circle and a line with some stuff inside it, with some properties. But someone's got to build that. It doesn't come from nature.

You don't go and harvest gates from trees, you got to go build that, and someone has got to do that.

So what to learn here is how do we go about building a gate?

And here you will see practically how do you deal with voltage thresholds that satisfy a given static discipline?

So before I jump into building a gate, let me try to build up some intuition. As is my usual practice, I'd love to get you to build some intuition as to how to build a gate. And then we'll go through the mechanics of doing it. So to build intuition, let me show you an analogous situation in fluids.

So let's say I have a cauldron of water.

This is like a power supply. And I need to feed this fluid down at some output source. And what I do in the middle is put in a couple of taps, faucets, all right?

And so what do these guys do? Under what condition do you have fluid flow out of the tube at the other end?

You will have fluid flow if -- So let me call this A and B.

If A is on and B is on then C has water.

Otherwise, if both A and B are not on then C has no water.

So this is already beginning to sound like a AND gate, correct, where you get water only if A and B are both turned on. So we're going to use this insight, a stream of some flow and I put things to obstruct the flow. And when both the obstructions are lifted I get the output. I want to use that intuition to build an AND gate. Similarly, I could build a system that allows me to build the following structure -- So in this scenario let me call this -- -- the signal of A and B here. And in this situation under what conditions, provided the power supply has water, under what conditions do I get water out?

In this situation, it is I get water if A or B are turned on. So I don't need to turn both A and B on. If either one of them is on, I'm going to get fluid flow here.

So this will help us build the inside to build the OR gate.

So that's an analogy involving items we see in everyday life.

Let me now move into the electrical domain.

In the electrical domain my analogy would be something like this. Let's say I have a power supply and I have two switches A and B. And I build a little circuit that connects this voltage source across the bulb using a couple of switches. In this case, the bulb is on if both switches A and B are on.

My bulb turns on. If I switch either one of them off my bulb turns off. So notice that I can begin implementing things like this if I had this element.

I had sources already. I know how to deal with bulbs.

I model them as resistors. So I need to do something about this new element called a "switch".

So let me build an abstract device.

I'll tell you how to do that in real life in a second.

So if I had the switch I could build things like this.

I could put switches in series in a circuit and get myself something that looks like a AND function.

So let me go ahead and build an equivalent circuit for a switch.

So the switch has a couple of terminals here and I have a control. Switches have a control and they have a pair of terminals. And the equivalent circuit for this looks like this. This is for my switch.

So when control is a 0. Then my switch is open to give me an open circuit in the circuit that I've shown you here. And, by the same token, if my control is a 1 then -- -- I have a connection between in and out.

And this is a short circuit. So, in other words, if my switch has 0 at its control, I'll talk about how to get that, I have an open circuit, and if it's a 1 then I have a short circuit. This is a switch going on and off. Now, in traditional switches mechanical pressure is my control signal.

If I apply mechanical pressure my switch could turn on.

And if I take away the mechanical pressure then I could get an off situation. So let's for now imagine that we have a switch. I still haven't told you how I am going to get a switch in real life.

Let's imagine you have a switch.

It's a three terminal device. There's a control thingamajig coming in. Input and an output.

So let's build the following little circuit containing a switch. So what I'm going to do, I will take a resistance RL and plug it in here.

And connect my power supply like so.

So the little circuit that I build has a resistor.

And I connect the switch in this pattern and I get a VS.

Lorenzo, you can set that up there if you'd like.

No problem. So I get a VS here.

Now, a couple of lectures ago I told you that 6.002, and for that matter, 004 and many of our other courses deal with combinations of elements.

And we often deal with the same kinds of combinations again and again and again. We see the same sorts of patterns happening, and we need to begin to learn to identify these patterns. This is an incredibly common pattern. You'll see this pattern more times in 6.002 than any other pattern, I promise you.

A power supply connected to a resistor and connected to a couple of terminals of some interesting device.

I promise there will be at least one such pattern on the quiz, for example. These patterns are incredibly common. So let's take a look at the interesting properties of this pattern.

Since this pattern occurs so commonly, I am going to create a short form. I have already created a short form which is this ground node here.

By putting ground 0 all I'm really saying is that there is a wire connecting these two and that's my ground.

So I already have a short form here.

My second short form is when I connect a power supply to a node. Then what I'm going to do is come up with yet another short form that looks like this, an up arrow with the voltage written there.

This symbol simply says that this node is connected to a power supply with voltage, or a voltage source voltage VS.

So I just have come up with a slightly simpler representation for the little pattern that I have.

Now let's take a look at the properties of this little system. Let's first look at what happens when C is 0. When C is 0, let me draw the equivalent circuit for this using the open circuit out there.

That's what I get, OK?

So when C is 0, if VS is a high voltage, let's say 5 volts, what do you expect at the output if C is a 0? This voltage VS appears at V out because this is an open circuit here.

Remember, RL and this little device form a voltage divider.

But since it's an open circuit its resistance is infinity.

And so therefore in this resistor divider all the voltage falls across this open circuit. So, in this case, v out is a 1 or a high voltage. But let's take a look at what happens when C is a 1. In this situation, I have my RL, that's what I have.

It's a short circuit at the switch and C is a 1.

So what's the voltage v out in this case?

Not surprisingly, since I've shorted this node to ground the voltage at this point is 0.

So if I have low voltage that's corresponding to logical 0s that corresponds to a 0. So I can build a simple truth table for C and use logical symbols here.

So when C is a 0 I get a high at the output and when C is 1 I get a low at the output. Have you seen a device that behaves like this so far? That's a little inverter.

That's the exact behavior of an inverter.

So this thing I've written here is a truth table for an inverter. So notice with just a simple little switch and a resistor, I have managed to build an inverter. Before I go on, I guess we have some things to show you.

And let me pause for a couple of seconds and do that.

First of all, what I want to show you is the following idea. So as I was preparing for this lecture last night I said, now here I am telling the 6.002 gang that you need to learn about analog circuits and resistors and all of that stuff, and you also need to learn about digital systems and all of that stuff.

And I said, because these two are very commonplace and often times they occur together. So I said well, if I really believe in my own BS then there should be something around me where I can find both of them instantaneously. So I said let me do the following experiment. Let me close my eyes and reach out and see what I touch. So I closed my eyes, reached out, and guess what?

I touched the lonely mouse. The mouse.

So I said let me see what is in side the mouse.

And if I believe in my BS we should find analog, little components and digital components in there, right? So let's see what is inside the mouse. All right.

There we go. Don't try this at home, as with many other things we do in lecture.

Come on. Show me what I want to see.

OK, here we go. Not bad.

Let me show you what we have here in this poor shattered mouse. That's my finger, silly. You should recognize this little resistor here. That thing with the little bands, oh, here we go. We'll use this.

That's a resistor. And you'll see capacitors in about four weeks. That's a capacitor.

And there is a digital IC here. That's a digital IC.

That contains a bunch of gates inside it.

So this mouse has not made a liar out of me.

So what I just showed you was a little device that we use in everyday life that has both analog components and digital components. A large number of devices that we use in daily life are this way.

You can do the same thing to your laptop.

You could go try it out. And you will find a bunch of analog components and a bunch of digital components.

And you really, really need to understand the whole caboodle here. Let me show you a fun little demo involving gates. Now, I want you to be very careful here. Lots of caveats here.

If your grandmother asks you how big is a gate don't say this big. This is how big gates used to be, I would say, when they were first invented.

When they built gates out of discrete vacuum tubes and so on, this is how big a gate used to be.

This is roughly that big. Today in a chip, in a small VLSI, very large scaled integrated circuit in a chip, which is about 1 cm on the side, how many gates do you think I can fit in a thumbnail sized chip? Any guesses?

With today's technology, how many gates can I fit on a chip? It has to be more than a million because I just told you that Pentium IV was 20 million and that was a year ago. How many?

40 million is a good guess. So on the order of 40 to 80 million gates in a 1 square centimeter.

Intel just announced that they will be shipping a chip containing 1 billion switches. Remember, this whole thing is a gate, right? Inverter, a resistor and a switch. This thing is a switch.

So Intel is going to be shipping something containing a billion of those little elements.

Just keep those large numbers in mind.

So here is a little circuit that I showed you here, A, B, the NAND gate, the NAND gate at the output and the inverter. So this output A is going to be 1 whenever either A or B is off. So the output is a 1 in this case when both A and B are off. I turn A to 1, output is still a 1. So the moment I turn both of these inputs into a 1, these are 1s, the output goes to 0. That's behavior for NAND gate.

If I switch any one of the inputs to a 0 the output should go to a 1. Similarly, for the inverter here, when the input is a 0 the output is a 1.

And when I switch it so should the output.

Now imagine a circuit, a little chip containing billions of these devices. And just imagine all of these 1s and 0s flying around. So one simple switch in the input, like a click of a keystroke could actually cause a billion signals in your circuit to be flipping around.

And that causes some fun stuff to happen, which we will learn about a few months from now. But for now that's a quick show of a little circuit that looks like that.

Let me go back to talking about building other types of gates.

So that was an inverter. So now you know.

You're almost halfway to being able to build a Pentium IV.

You've come all the way from nature to gates.

And Pentium IV contains 20 million of them so you now know how gates are built. So that's an inverter.

Let's look at how we can build other forms of gates.

To build another gate let me do this.

How about this pattern? If I build a pattern like this with A and B coming in here and I put two switches with their inputs in and out, so two switches in series.

Let's write down the truth table for what this looks like.

Let's see. When A and B are both 0, what should the output be? These are both off so the output is directly VS which is a high.

When either of these switches is off 0-1 or 1-0.

If either switch is off then this node is cut off from ground. There is no current flowing here. So this entire voltage drops across this infinite resistance here, and so I get 1s at the output as well. If both switches are on what happens? If both A and B are on then I get a short circuit to ground and my output is a 0.

So can someone tell me what gate this is?

Awesome. We just build a NAND gate.

This is unbelievable. Five lectures and you've already come all the way from nature to the primitive building blocks of microprocessors. It's pretty amazing.

So what about this one here?

What's this? I haven't told you this before but if an AND gate becomes a NAND gate, this is kind of an OR arrangement, what should an OR become?

NOR. It's all completely logical.

So you can go home and practice a truth table for this.

A, B and C. I'll just fill in one of the rows. So in this particular situation, if both A and B are 0, if A is 0 and B is 0, both the switches are off, so it's as if this little sucker here is cut off from ground and VS falls across from C to ground here and the output is a 1, so on and so forth.

So I can build other interesting forms of gates.

So let's say I build something that looks like this.

I build something like this.

You can write the truth table for this or you can look at this and write down the function that this one supports.

Notice that this output here is going to be a high only when both of these are not connected to ground.

And if you stare at it some more the function this one presents, this is my AND function.

Suppose this one didn't exist, that would be my AND function.

But because this one exists that's in an OR configuration and so I get a C. And so because of that I get something that looks like this. So this is my A dot B, this is my plus because of a parallel here, and ultimately this caused an inversion in this gate.

So the primitive pattern has a generic inversion built into the output. That is why they commonly end up building NAND gates and NOR gates and so on as the simplest gates. We don't build AND gates and OR gates. How can I convert this one to an AND gate? Anybody?

Put an inverter on the output. So what I can do is take this little sucker here, put an inverter here and I get an AND gate. So the real primitives in circuits tend to be NANDs and NORs.

OK. So the real practical among you should be saying at this point all right, all right, I buy this, if there existed a switch.

I know exactly how to go from nature to building Pentium IVs if there exists a switch. So that the obvious next step for me is to show you a switch, a physical switch device.

And to introduce a switch device, let me show you a three terminal element. Remember, the switch has three terminals, an input, output and something called the control, C. So I'm going to introduce a new primitive element called "The MOSFET Device".

MOSFET stands for metal-oxide semiconductor field-effect transistor. This is shortened to FET or transistor. Now

I'm going to show you that this works like a switch. And before I do that, in fact, let me do that first. Then I'll show you something else. So this device has the following symbol. It has a terminal called a gate, the drain and the source. Gate, drain and source.

Three terminals. This is the primitive element that forms virtually every electronic component built today. This is the foundation of the universe. So this little MOSFET device, we can look at how it behaves. I'll show you this thing on the screen in a second, but this guy behaves very much like this device I was postulating earlier.

Let's take a look at this device on the scope.

To do so let me label some voltages and currents.

So let me label this voltage as vDS.

Let me label this voltage as vGS between the gate and the source. And let me label the current coming into this node iG. In this device, the physical device that I'm going to show you, the current going into the gate is always 0.

So iG is always going to be 0 for 6.002.

In real life there is some leakage and so on.

But in 6.002 for now we deal with a very simple abstract model, iG is 0. And let me label the current here as iDS. To be correct with the nomenclation, the current into node D should be labeled iD, but because iG is 0 iD flows out through the source as well, so I would simply call it iDS just so that I can show that vDS and iDS are the two voltages and currents that I am going to deal with.

So that's my little device here.

And notice that the source terminal is common.

I use the source both for the control GS and I use the source for the drain as well. So you can view this as input, view this as out, and you can view this, if you like, as the control abstractly.

So let me show you a plot of how this behaves.

To understand how it behaves, I can draw an equivalent circuit for it. So in this particular situation, if its behavior is characterized by the voltage applied to vGS. Much like the control on the switch, vGS is my control. So if vGS is 0, oh, I'm sorry. If vGS is greater than or equal to some threshold voltage VT -- So vGS, the voltage applied here is greater than some voltage, VT, a threshold voltage, or the pressure of the switch is greater than some threshold pressure then this guy behaves like a short circuit.

This is iDS, this is my drain and this is my source. So if the voltage applied between the gate and the source is higher than some threshold then this behaves like a short circuit.

Similarly, if the voltage vGS is less than some threshold VT then in that situation -- -- I get an open circuit. And when I have an open circuit between D and S then the current iDS is going to be 0.

So this is the idealized model. And this idealized model is called "the switch model of the MOSFET".

The switch model or the S model of the MOSFET.

Well, if you want to see the internals of the MOSFET, I won't cover that in lecture or recitation.

You can look at the section, I believe Section 6.7 of the course notes. That has the internal structure of the MOSFET and how you physically construct such a device. So what I can do here is step back and stare at the device for a second or two.

And what it says is that if I apply a lot of pressure, if vGS is greater than a threshold VT then I get a short circuit here just like my switch.

When in doubt think faucet. If you put pressure on the faucet, think of this as closing, and when I open it, when vGS goes less than VD, less than a threshold, I take off the pressure and then it becomes an open circuit.

So I can plot the following.

Much like I plotted the iV characteristics of two terminal elements, I can plot the iV characteristics of this three terminal element in the following way.

I can focus on two terminals and look at vDS and iDS for that terminal pair and draw the curves for how it will behave as I change vGS that I applied. So what I'm going to show you is that if vGS is less than a threshold then this behaves like a open circuit. So no matter what the voltage is the current is 0. Similarly, if vGS greater than equal to some threshold voltage then I get the behavior iV curve of a short circuit where the current can be anything and controlled by external forces like in any short circuit.

So let me show you on the screen.

Lorenzo has kindly put the graph up already.

So I'm showing the iV curve of a switch.

Notice that when vGS is greater than VT, greater than a threshold I get the vertical line corresponding to a short circuit. Is it this one?

This one. There we go.

So what I'm going to do here is I'm going to reduce vGS to below VT. What should you see happening?

The curve, from being a short circuit, should hammer down to becoming an open circuit. That's the curve for an open circuit as I drew out there for you.

VGS pressure ain't enough. Lots of pressure, boom, it's a short circuit. I really like to think of this pressure analogy if I get confused whenever I look at a MOS transistor and I need to look at vGS and so on I always think vGS is greater than VT. Lots of pressure on the switch it turns on. Just remember that, and then you won't forget this vGS thing here.

So that's the behavior of a switch.

And so viola, there's our switch.

So I've given you a three terminal element that is a switch that is controlled like a mechanical switch.

So I can build a, if I replace -- This was my switch earlier. And what I can do is replace this with my MOSFET and that's what I get.

And I won't bother showing you this is your inverter.

All of that has replaced the abstract switch with a physical switch which behaves as shown in the graph up there.

And so I apply an input here and I take the output here.

So as 6.002 you could look at this and say ah-ha, that is an inverter. When you go to 004 what you will do is build this triangle and a circle around it and you will ignore what's inside and just look at that.

So in 002 we showed you that the internals look like a pattern with a MOSFET and a resistor, but it's really the abstract inverter looking in from the outside.

I'm just going to close the loop inside the digital gate, and this was inside your little inverter with a resistor and a switch. Let me continue with this for a little longer here -- -- and do something that we like to do a lot, which is plot what are called input / output curves.

So let's say the voltage applied here is v in and let's call this v out. For fun let's plot a v in versus v out for this inverter. So when input is a 0, let's say VT is 1 volt for the inverter.

The threshold voltage is 1 volt.

The threshold pressure is 1 volt.

So when input is a 0, and let's say VS is 5 volts.

So when the input is a 0, this guy is turned off.

So what's the output? What's the output voltage?

If this is turned off, what's the output voltage?

It's the supply. The supply directly shows up here. And so as long as the input is 0 the output is at 5 volts. And this is true until the input reaches 1 volt. As long as the input is less than 1 volt my output stays high.

And then when my input exceeds or hits 1 volt then at that point the switch turns on and the MOSFET turns on and shorts the output to ground in which case boom, this is what I get.

And then, no matter how much I increase the input, my switch stays on and the output follows a zero volts at the output. So this is my v in versus v out curve for the inverter. One of the interesting things that we do a lot is see whether this satisfies some voltage threshold. So let's say I have a VOL of 0.5 volts, VOH of 4.5, VIL of 0.9 and VIH of 4.1 volts. So VOL says in its low value is the output less than 0.5? Yup, output less than 0.5.

In its high is it more than 4.5?

Yup, it's more than 4.5. Does it recognize all values below VIL as a low input? Yup.

So anything below 0.9 or 1 for that matter is viewed as a low.

That's good. So these pass.

And high, anything above 4.1, is that treated as a high?

Yes. So anything above 4.1 is treated as a high and the output goes low.

So therefore this inverter that I've designed for you here satisfies the static discipline and this inverter can be used in circuits or other devices that conform to this value here.

In your recitation, you will look at a slightly more detailed model of the switch where the switch behaves like a resistor.