Now, there are times when we want to change the instruction sequence.

If you remember during instruction fetch, we always increment the PC.

We point to the very next instruction in memory.

But there are times when we don't want that next instruction.

Now, if you've done a little bit of programming,

you're familiar with loops, and if-then statements, and function calls.

Now, if these are not familiar to you, that's OK.

We'll kind of cover them later in the course.

But for these type of constructs, we need special control instructions

that directly change the PC to something other than the incremented value.

In order to accomplish this, we need to add control instructions

to our instruction set.

And we're going to talk about two types.

The first is a jump instruction, which is unconditional.

What that means is we always change the PC.

The second type is called a branch instruction.

Here we change the PC only if a specified condition is true.

Now, we have a special register called a condition code register

and bits are set or cleared in that register

depending on the result of an instruction.

And a branch instruction, changes the PC from its incremented value

to something else if the CC, the condition code register,

has a particular value.

And if the condition code is not that value, then we just leave the PC alone.

We just use the incremented PC.

Let's first cover the LC-3 jump instruction

and then later we'll talk about branches.

A jump sets the PC to the value contained in a register.

So we will fetch the next instruction from this address.

Here's the format, the leftmost 4 bits of the opcode,

and we need a base register that we're going to read and load into the PC

and that's bits 8 through 6.

Now, the other bits are zeroes.

This is a case where we have more bits than we need, again,

every LC-3 instruction has 16 bits, and we just

happen to not need all the bits in this case.

Here's a specific example.

The opcode is 1 1 0 0 for a jump and our base register is 0 1 1 or 3.

So what this says is to load the contents of R3 into the PC

and that's the address in memory where we're

going to get our next instruction.