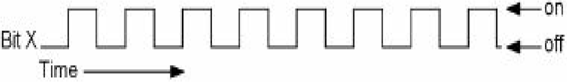
The Clock

We need to turn the appropriate control bits on and off at the appropriate times. We will look at the appropriate bits later, first we will look at the appropriate times.

Here is a new kind of drawing, we will call it a graph. It shows how one bit changes over time. Time starts on the left and marches forward to the right. The line on the graph has two possible positions, up means the bit is on, and down means the bit is off.

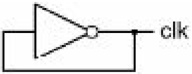


This graph shows bit 'X' going on and off, on and off regularly. There could be a time scale on the bottom to show how fast this is happening. If the whole width of the page represented one second, then bit 'X' would be going on and off about eight times per second. But we won't need a time scale in these graphs, as we will only be concerned with the relative timing between two or more bits. The speed in an actual computer will be very fast, such as the bit going on and off a billion times per second.

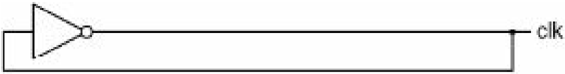
When something repeats some action regularly, one of those actions, individually, is called a cycle. The graph above shows about eight cycles. You can say that from one time the bit turns on to the next time the bit turns on is a cycle, or you can say that from the middle of the off time to the middle of the next off time is the cycle, as long as the cycle starts at one point in time when the bit is at some stage of its activity, and continues until the bit gets back to the same stage of the activity again. The word 'Cycle' comes from the word 'circle,' so when the bit comes full circle, that is one cycle.

There was a scientist who lived in Germany in the 1800's who did some of the early research that led up to radio. His name was Heinrich Hertz, and among other things, he studied electricity that was going on and off very quickly. Some decades after his death, it was decided to use his name to describe how fast electricity was going on and off, or how many cycles occurred per second. Thus, one Hertz (or Hz for short) means that the electricity is going on and off once per second. 500 Hz means 500 times per second. For faster speeds we run into those ancient languages again, and one thousand times per second is called a kilohertz or kHz for short. Going on and off a million times per second is called a megahertz, or mHz for short, and a billion times is called a gigahertz, or gHz for short.

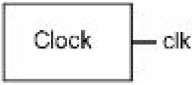
Every computer has one special bit. All other bits in a computer come from somewhere, they are set on and off by other bits or switches. This one special bit turns on and off all by itself. But there is nothing mysterious about it, it just goes on and off very regularly and very quickly. This special bit is built very simply, like this:



This seems a silly thing to do. Just connect a NOT gate's output back to its input? What will this do? Well, if you start with the output on, the electricity travels back to the input, where it enters the gate which turns the output off, which travels back to the input which turns the output on. Yes, this gate will just go on and off as quickly as possible. This will actually be too fast to be used for anything, and so it can be slowed down just by lengthening the wire that makes the loop.



The simplified diagram shows this to be the one special bit in the computer that has an output but does not have any inputs.



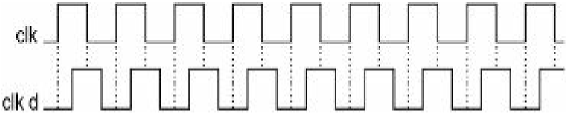
This bit has a name. It is called the clock. Now we usually think of a clock as a thing with a dial and hands, or some numbers on a screen, and we have seen such clocks in the corner of a computer screen. Unfortunately, someone named this type of bit, a clock, and the name stuck with the computer pioneers. It could have been called the drumbeat or the pacesetter or the heart or the rhythm section, but they called it a clock. That is what we will mean when we say clock throughout the rest of this book. I guess it's a clock that ticks, but doesn't have a dial. If we want to talk about the type of clock that tells you what time it is, we will call it a 'time of day clock,' or 'TOD clock' for short. But the word 'clock' will mean this type of bit.

How quickly does this clock go on and off? These days it is well over a billion times per second, or several gigahertz. This is one of the main characteristics that computer companies tell you about to show you how great their computers are. When you see computers for sale, the speed that they advertise is the speed of its clock. The faster a computer is, the more expensive it is, because it can do more things in one second. It is the speed of this single bit going on and off that sets the tempo for the whole computer.

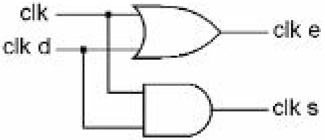
To move data via the bus, we need first to enable the output of one and only one register, so that its electricity can travel through the bus to the inputs of other registers. Then, while the data is on the bus, we want to turn the set bit of the destination register on and off. Since the destination register captures the state of the bus at the instant that the set bit goes off, we want to make sure that it goes off before we turn off the enable bit at the first register to make sure that there are no problems.

Let us first attach a length of wire to the output of the clock. This will delay the electricity slightly. We want it delayed about one quarter of a cycle.

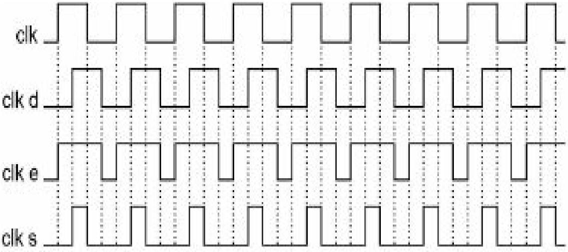
If we show the original clock output (clk) and the delayed clock output (clk d) on a graph, they will look like this:



Now we're going to do something fairly simple. We will take the original clock and the delayed clock, and both AND them and OR them to create two new bits, like so:

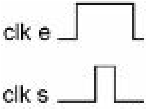


One of the new bits is on when either 'clk' or 'clk d' are on, and the other new bit is on only when both 'clk' and 'clk d' are on. The graph of the inputs and outputs of the AND and OR gates is shown here. They are both still going on and off regularly, but one of them is on for longer than it is off, and the other one is off for longer than it is on. The on time of the second is right in the middle of the on time of the first.



Notice that they have names, 'clk e,' which stands for clock enable, and 'clk s,' which stands for clock set. And what do you know, these two bits have the perfect timing to move a byte of data from one register, across the bus, and into another register. Just connect 'clk e' to the enable bit of the 'from' register, and connect 'clk s' to the set bit of the 'to' register.

Here is a single on/off cycle of these two bits.



If you look at the timing here, this meets our requirements of needing to first enable the output of a register, and then, after the data has a little time to travel down the bus, to turn the set bit of the destination register on and off before turning the enable bit off at the first register.

Of course, these clock bits cannot just be connected directly to every register. There must be other gates in between, that only allow one register to get an enable at any one time, and only the desired register(s) to receive a set. But all enables and sets ultimately come from these two bits because they have the right timing.

Since we will use clk, clk e and clk s throughout the computer, this is the diagram we will use to show the clock: