No. 36, Nov 1996

SPEED #1: THE DATA BUS

This is another topic suggested by Tom, WB9LNL. He said, "Write an article on speed." Wow, did he ever open a can of complicated worms! The speed of a computer depends on so many things; it is really difficult to write a single article on the topic, unless it just relates that a Pentium is usually faster than a 486, which is usually faster than a 386, which is usually faster than a 286, which is usually faster than an XT. Usually, but not necessarily always. So, lets explore the topic starting at the beginning, and we'll have to spread it over a couple of articles.

First, we must consider how data moves around in a computer. The central processing unit chip (CPU) in a computer has to "talk" with other parts of the computer. To do this, it sends and receives data using a bundle of wires called the data bus. An 8088 (XT) has 8 wires in the bundle, and is said to have a data bus width of 8 bits. You may recall from previous articles that 8 bits = 1 byte; the letter A can be represented by 8 bits (or 1 byte), as can the letter a. So, think of it this way. An XT CPU can send a single letter or number to some other site in the computer all at once, by sending data down all 8 wires simultaneously. Actually, the CPU does this by raising the voltage on some wires to +5 VDC (which represents a one) and leaving others at 0 volts (which represents a zero). The pattern of ones and zeros in the 8 wires at a given moment might represent the letter a. I can even write it for you. The lowercase letter a is 01100001, where 0 is a wire with no voltage on it and 1 is a wire with a charge of +5 volts.

As you might suspect, a 286 can do better. The CPU in a 286 computer has a data bus width double the size of the XT - 16 bits. The 286 chip can send out two letters simultaneously, so in any task that requires it to send out many letters, it will be faster than an XT.

Both the 386 and 486 CPUs sport 32 bit data busses, making them even faster than the 286. Finally, most Pentium CPUs have a 64-bit bus width. All this is summarized in the table below, along with a few other interesting facts about each of these chip models.

CPU	DATA BUS WIDTH	NUMBER OF TRANSISTORS	YEAR INTRODUCED
8088 (XT)	8	29,000	1979
286	16	134,000	1982
386DX	32	275,000	1985
486DX	32	1,200,000	1989
Pentium 66	64	3,100,000	1993

Let me construct a 'ham style' analogy to bring the point home. Suppose we have a communications center. There are 8 CW operators in the room, sitting at 8 keys, each of which is connected to some remote site by 8 wires (connected by a common return or ground wire, which we won't consider). At the remote site are 8 light bulbs. When the CW ops in the comm center are told to send a letter, they must work together to send it. Each must close his key or not close his key, depending on the letter to be sent. The pattern of on or off light bulbs at remote site tell what letter was sent. If the 8 light show the pattern 01100001, where 0 indicates an unlit bulb and 1 indicates a lit bulb, the lower case letter a was sent. That is an XT.

In the communications center of the 286, there are 16 CW operators, 16 keys and 16 wires. These ops can send not just one, but two letters simultaneously. While the XT ops might send just the letter H, the 286 ops might send the two letters HI, simultaneously, because there are 16 light bulbs on the wall at the receiving end, connected by 16 wires, not just 8. The 386 and 486 ops can send HI OM because there are 32 lights on the wall, each letter requiring the use of 8. And the Pentium ops can send HI OM ES 73, 8 characters all at once, because there are 64 light

bulbs at the receiving end, energized back in the comm center by 64 ops at keys, over a bus of 64 wires.

That is the concept of a data bus. Clearly, the newer a CPU model is, the more complicated (and capable) is its data bus. A wider bus is more efficient because more data can be moved down it at one time. On the other hand, a wider data bus means a more complicated chip, which is part of the reason for the increase in number of transistors shown in the table above. Does that completely explain the differences in speed? Nope. Far from it. More next time. Happy computing!