THE COMPUTER CORNER

No. 167: A Conversation in a Computer

Stan Kaplan, WB9RQR
715 N. Dries Street
Saukville, WI 53080-1664
(262) 268-1949
skaplan@mcw.edu

This article is based on one that I published 148 articles ago, way back in 1995. Many folks have probably never read it, but it is just as pertinent today as it was back then. It deals with memory, and how memory is used when you type a letter. The lesson to be learned is, save your work. And, don't always rely on the computer to do it for you.

There are two major kinds of memory in a computer, *volatile* and *non-volatile*. The platters in a conventional hard drive to which data is written is *non-volatile* memory, because anything written to the platters while you are working remains there after the power is turned off.

But normally, when we talk about computer memory, we are considering the memory chips – those "sticks" that give you 1 or 2 or 4 gigabytes of RAM (Random Access Memory) that is so important to both the proper operation of the computer and its overall speed. The RAM is of the *volatile* type. When you turn off the computer, anything previously written to RAM goes away to never-never land. It is lost forever.

However, RAM is very necessary and very important, because it is really the "scratch pad" for all your work, as well as for the programs that allow you to work. Let me illustrate this with an everyday example.

Last week you started a letter to the ARRL. You started Microsoft Word (which means that the Word program was read from the hard drive and written into RAM). You then began typing the letter (which went into RAM) and finished two paragraphs. Intending to finish it later, you saved it (ARRL.DOC) and turned off the computer. When you saved it, the paragraphs were moved from RAM onto the hard drive platters. When you turned off the computer, everything in RAM (ARRL.DOC and the Microsoft Word program itself) went poof! Gone. Every byte in the RAM chips changed to a zero.

Today, you want to finish the letter. You power up your machine, find the file ARRL.DOC, and click it. That first causes Microsoft Word to be read from the hard drive and written into RAM, as before. Then, your partial letter is read from the hard drive and written into RAM. When all that is done, your letter appears on the screen, ready for editing.

What significant steps have occurred? When you clicked on the ARRL.DOC file, most of the instructions in the Microsoft Word program were read from the hard drive and moved into RAM, where they take over the interpretation of most of your keystrokes. Then, Word opened ARRL.DOC on the hard drive and copied it to RAM. What you see on your screen is an exact copy of your letter as it was found on the hard drive platter, and it now exists also in RAM.

Let us suppose that, after reading what you wrote last week, you decide it is fine, and no changes are needed. All you need to do is to add the closing, so you type the following 16 characters: ***Sincerely yours,*** and you save the letter.

Your keystrokes were sensed and acted on by Microsoft Word (hereafter labeled "W"), acting in concert with your Operating System (hereafter labeled "OS" - XP or Windows 7 or whatever). By the way, the OS also resides in RAM, as does everything that is active when your computer is

turned on. These two entities, the OS and W, are like Siamese twins joined at the brain, so we will nickname the pair "OS-W".

As you typed the character **S** in the word Sincerely, OS-W did two things. First, they were "watching" the keyboard in case you typed something, which you did. When they saw the S typed, they moved the **S** (actually, it's numeric representation) into RAM. Second, they echoed the letter **S** to your screen. They accomplished those two steps pretty darn quickly. Both steps were finished by the time the **S** appeared on your screen. Note carefully, however, that the OS-W did absolutely nothing to the hard drive. You could prove this yourself by watching the hard drive LED on the front of your computer case while typing the **S** – the LED will not light up. The same events were repeated when you type the **i**, then the **n**, and so on. In each case, the character typed was added to the copy of the letter in RAM and also echoed to your screen.

When **Sincerely yours**, was completely typed, you saved the letter. When you did this, the OS-W "talked" to the hard drive for the first time. What did the OS-W and hard drive say to each other? Here is the conversation, with OS-W indicating the word processor working in concert with the operating system, and HD indicating the Hard Drive:

OS-W: Mr. Hard Drive, please find any copy of ARRL.DOC that you might have on your platters and rename it ~$ARRL.DOC (the hard drive complies, and it also lights it's LED while doing so, indicating to you that it is active).

HD: OK, Boss, I did it!

OS-W: Good guy! Now, take this data I am about to send you, write it to your platters, and name it ARRL.DOC.

HD: Right, Boss, send it down. Got it! There, I'm all done.

OS-W: Great, thanks. Now erase that ~$ARRL.DOC file since we no longer need it.

Now you can see that if you turn off your computer (or a power failure occurs) before your work is saved, you are out of luck. Until you close Word (this will cause Word to ask you if you want to save your work) or otherwise give the command to save your work to disk (clicking the little floppy drive button in Word's toolbar), your data resides in only one place – in RAM. And, RAM is terribly volatile. How volatile?

A stick of RAM, so long as it is fed power, acts like a huge bank of capacitors. (It is actually a huge bank of capacitor-transistor pairs, but we can think of them as just many, many capacitors each of which is charged or not charged). A charged capacitor represents the numeral one, and a capacitor that is discharged represents a zero). Sound familiar? Therein resides the zeroes and ones – the binary numbering system – which the computer works with.

However, these RAM chips are really quite forgetful. Even if they are fed a constant source of power, they will loose their memory (zeroes stay zero, but the ones also become zero) within a very short period of time. So the engineers have designed circuitry within the computer to refresh the charges (or lack of them) every 15 milliseconds, so the zeroes and ones that represent your data in the computer don't go "poof". Think about that next time you are editing a critical file and don't have a backup on the hard drive. Only 15 milliseconds before your work is gone!

Happy Computing!