



# Personal Computers for Engineers: How to Keep Your Computer Safe and Functional

An Online Continuing Education Course for Engineers

**Course Number: PD-2002**

**Credit: 2 Hours / 2 PDH / 2 CPD**

# Personal Computers for Engineers: How to Keep Your Computer Safe and Functional (Revised 2015)

Jack H. Warner Jr, P.E.

## Overview:

The PC's that engineers use to get their daily work done are very complex beasts. It can almost be a full-time job just to keep your hardware and software up-to-date, running efficiently, and free from threats. The purpose of this course is to give you a primer on some of the issues that you need to be aware of, and to give you some basic guidance.

## Hardware:

First and foremost you have to be aware of the current hardware that you are using, and future trends. The heart of the PC is the CPU (Central Processing Unit), and it has a huge impact on how fast your computer will run. It used to be a simple matter of buying a new PC every few years with a faster “clock” speed. It has gotten more complicated now with multi-core systems. Let me tell you about “Moore’s Law”, and the trends in processors.



## Moore’s Law and CPU’s:

Moore’s Law, named for Gordon E. Moore, cofounder of Intel, observes that the number of transistors that can be put on typical chips, such as CPU’s, doubles every 18 to 24 months. For the past 5 decades, this has proven to be pretty accurate. The funny thing is that processor clock speed also seemed to be following the same trend; so many people assumed that Moore’s Law applied there as well, but it turns out that was just a coincidence.

Have you noticed that new machines in recent years have stopped their exponential growth in clock speeds? If the trend of the previous decades had continued, the latest PC’s would be sporting 20 GHz CPUs. But a trip to the Dell web site reveals that the typical CPU I might order today is less than 5 GHz. What’s the deal with that?

Aside from some slightly different design strategies that give a slightly lower “speed” for the same performance, it seems that the CPU designers have hit a bit of a wall. They are not having trouble increasing the number of transistors on a chip (which is consistent with Moore’s law), but they are experiencing problems when trying to increase the clock speed. Heat seems to be a huge problem, along with other details such as current leaking from one circuit into another. This speed issue has caused a major shift from including the clock speed in the name of the CPU, to giving them cryptic names like Intel® Core™ i7-4790.

Of course they are trying to speed up the other parts of the PC (memory, bus, drives, and graphics), but I just want to talk about the CPU for now. So what are they doing? They are

working on other methods to give the computer more power without increasing the clock speed. Since they can add more transistors, there are several possibilities. First, they can increase the amount of memory embedded in the CPU chip. This memory is called “cache,” and is very fast. Typically, the more cache memory a chip has, the faster it can get things done, because it can often use data in fast cache memory instead of the slower system memory. Next, they are working on making the most of each clock cycle. The much advertized “hyperthreading” is an example of this. It creates duplicates of some parts of the CPU, and allows some instructions to be executed in parallel. This can boost performance up to 30%.

Finally, a significant way to increase performance is to actually have multiple CPUs. This is not a brand-new idea. PC servers have incorporated multiple CPUs for many years, but because of the limited alternatives for providing more computing power for desktops and notebooks, multiple CPU’s are becoming common. The new twist is that CPU designers are putting two or more CPU’s on a single chip. This design is called “multicore.” This can bring significant performance increases under the correct circumstances. The first catch is that two processors will not double the performance. Any time you have more complexity, you will have extra overhead in managing it. In this case, there will be extra overhead involved in keeping the two processors in sync with each other and the rest of the machine. So, even under the best of circumstances, you will not get 2x performance with 2 cores. The second catch is that it will not speed up many programs. This is because programs have to be written specifically to take advantage of two or more processors. If you are running more than one program simultaneously, you can get an immediate benefit, but if you have one application that is hogging most of the power, it will not run faster unless it is rewritten specifically for multiple processors.

## Bits

You might hear mention of a CPU being “32-bit” or “64-bit”. Most new CPU’s are capable of running in either mode, though some older ones might only run in 32-bit mode. The truth is it makes absolutely no difference unless you want to run a 64-bit operating system on your computer. Most current versions of Windows and Linux take advantage of 64-bit hardware, and if you run one of these, you get a few benefits. Processes might run a little faster, because more bits can be piped into and out of the processor at one time, and your computer can access HUGE amounts of memory, rather than being limited to about 4GB, which is the limit with a 32-bit operating system. The main reason you might want to run a 64-bit system, though, is if the applications you run to get your job done come in special 64-bit versions. Regular software will not run much faster on a 64-bit system, since regular software is designed for 32-bit systems, but if software is specially designed to run only on 64-bit systems, it could run much faster. So, if you are running a 64-bit operating system, try to use 64-bit versions of the applications that you use.

OK, that covers the major trends in CPU’s. For all other computer components, they just keep getting faster and cheaper. Memory is quite inexpensive, and is one of the most important factors in making your machine perform well. Doubling the recommended memory (up to 4GB in a 32-

bit system, much more in a 64-bit system) is an efficient way to ensure good performance. Hard drives keep getting bigger and faster, and graphics cards do too. For engineers, making sure these are current is a must. You don't always have to have the absolute latest product, but if you let either of these components get over a year or two old, you could see a noticeable improvement by upgrading to a more current product.

### Operating Systems:

The basic choices are some version of Windows, Linux, or Apple OS. Apple OS is only a choice if you buy an Apple computer, but lots of good software is available for Apple computers, and there is also software available that allows you to run Windows software on Apple computers. If you run only Windows applications, an Apple computer is probably not a good choice, but if you can do part of your job using Apple software, you could seriously consider this option.

Linux is inexpensive and efficient, but it can be difficult to learn, difficult to get support and the choice of applications is a bit limited. If the applications you need are available for Linux and you are willing to do your own research and trouble-shooting, Linux can save you some money, but this is not commonly the sort of commitment that an engineer can afford.

The current flavors of Windows are called Windows 10, 8, and 7. These are 64-bit operating systems, though as I mentioned earlier, they run 32-bit applications just fine. There is safety in numbers, and because Windows is so common (over 90% of personal computers run Windows), there are lots of applications available for it, and support is easy to find. One disadvantage to Windows is it is a bit "bloated" (inefficient), and it tends to become more sluggish over time, so let me address that issue.

### Why Does my New Computer Start to Act Old?

Just like the "new car smell" wears off in just a few short weeks, so too does that "new computer smell." Well, perhaps new computers don't really *smell*, but they are just so darn fast and dependable when they are new. But sometimes it seems that within just a few days things are not so good. The computer seems a bit sluggish!

Two basic phenomenons are at work here. First, there is the psychological aspect. When you set the new computer next to your old one, you can see how much faster it is. A few days later, you have likely forgotten how slow your old one was, and are now comparing the new computer's speed to your ideal desired speed (which is "instantaneous" in my case, perhaps yours too).

Second, we start touching our machines. Yep, I hate to tell you this, but they would stay "as fast as new" if we would simply not touch them! But what do we immediately do to our new computers? We rush to start installing software and peripherals. Obviously we have to install some software in order to get our job done, but everything that we install on our computers has an impact on performance.

On the low-end of the impact scale is a simple software application that we seldom use. In the best of cases it simply takes up room on the hard drive, which may slow down other programs just a bit if we leave our hard drive fragmented by such installations, but other than that, not much to worry about.

On the high-end is software that installs drivers, and various bits of software that hook themselves to the operations of the computer. Take an anti-virus program, for example. Every time you start your computer, it is going to take time to start up. It then runs constantly, intercepting every file access that every other program makes. It can cause as much as a 15% slow-down in system performance. Now, in the case of an anti-virus program, I highly recommend sacrificing a bit of speed. But over time you would probably be quite surprised at how many programs you have installed that are all competing for the attention of your CPU.

Let's just take my notebook computer as an example. I have had it over 2 years, so it has had a few various things installed over the years. It runs Windows, so I can simply hit ctrl-alt-del, press the "task manager" button, and see that I currently only have Lotus Notes and MS Outlook running. These may be the only two full-blown applications running, but if I click on the "processes" tab, I can see all the smaller bits of software that are currently running. ***I have 64!*** How many do you have? You can also get a clue that your computer may be running too many things by looking at your "system tray" (the icons on the right side of the taskbar). Your computer came from the factory with maybe a dozen. Has it grown by leaps and bounds?

So what are some of these 64 things that are running on my computer? Every one of these processes is using memory, and competing for my computer's CPU. Some I can guess: DrgToDsk.exe I think installed itself as part of a CD-writing package. Sr\_GUI.exe, SR\_Watchdog.exe, and SR\_Service.exe seem to be parts of the SecureRemote client that I use to access a remote Network. WCESMGR.exe and WCESCOMM.exe appear to be responsible for synchronizing one of my devices with my notebook. RealPlay.exe is RealPlayer for sure, which is taking 9 Meg of memory even though I am not running RealPlayer! Who knows what this 9 Meg program is doing, probably checking every 5 minutes to see if there is a new version of RealPlayer for me to download. Trillian.exe (my IM client) is using 21 Meg of memory just sitting in my system tray. Do I need these things? Most, yes, but after looking things over, I'm thinking I might not need all the bells and whistles of the CD burning package.

There are several sites that can help you identify the various processes that you have running. One good one is [www.answerthatwork.com](http://www.answerthatwork.com). There you can find an alphabetized list of common processes. This can help you track down how they got on your computer, whether you still need them, and how to remove them.

So what's to be done? Some people firmly believe in formatting their hard drive and reinstalling Windows every 6 months and starting over. Though this works, it might be a bit over-the-top. Take a look at your system tray, and look at the applications running on your computer. Are you running a bunch of extra stuff you do not really need? Figure out what they are and uninstall them. Once you've cleaned up the hard-drive a bit, run the defragmenter. Run a registry cleaner to streamline it. Stop streaming music over the Internet while you work and invest in an iPod. Go

to the performance settings and set them for faster performance with less eye-candy (yes, those icon shadows and scrolling menus take CPU cycles!). Replace that giant picture of yourself with some simple wallpaper (yes, it redraws faster!). Move all the extra icons off your desktop (the system has to go get icons for all of them on startup!). There are lots of ways; you simply have to figure out what is really necessary. Of course, buying a new computer helps a lot too! Just don't immediately load it up with everything you "might" ever need.

### Drivers:

While we are talking about operating systems, let me discuss drivers a little bit. Drivers are a concept that engineers can really appreciate... the idea is brilliant. But, they are one of the most common reasons for computer instability, so we have a love/hate relationship with drivers. Pretty much every piece of hardware attached to your computer is linked to the computer with drivers... printers, monitors, hard drives, scanners, mice... you name it. Printers are just one example; so picking on printers rather arbitrarily, I'll describe how drivers came to be and what they do.

Before windows came along, those of us that developed software had to modify our programs for every printer that we wanted to support. Sending plain text to a printer was easy, but if we wanted to support fonts, or draw graphics, we had to get specifications for the type of input that the printer was expecting, and modify our program to write that exact format. For engineers in the early years of PC's, the most important thing to support was HP plotters, and fortunately they all used a similar input format, called HPGL (Hewlett Packard Graphics Language), so that was the main thing that engineering software supported.

Today, the variety of printers being manufactured is quite large, and the types of data software developers are asked to output (text, bitmaps, fonts, and vector graphics) are quite a bit more varied as well. Windows solves this problem for software developers by providing something called a "Hardware Abstraction Layer" (HAL, no relation to the psychotic computer in "2001 – A Space Odyssey"). The hardware abstraction layer allows programs to interface with a "generic" output device (abstract, in other words). It is then up to windows, and a printer driver supplied by the printer manufacturer, to take the generic output commands from the program and convert them into something that each specific printer understands. In fact, Windows gives each program such a generic view of the output device that it might not even be a printer at all. The program code to draw a joist on the screen turns out to be exactly the same code that draws it to a printer. The only difference is if the programmer knows he is drawing to a printer, he might want to draw on more than one page, set it to portrait or landscape, or a few other things that only make sense for real printers.

So, which printers should work fine for your software on your hardware? Any printer that has a proper driver should work. Still, there can be problems, often with the drivers themselves. The driver sits on the other side of the HAL from your programs, and is written by the printer

manufacturer to talk to a specific printer. It is a lot of work to write a driver, and because it has to work with the hardware, it is allowed greater access to the delicate inner workings of your computer than most programs are allowed. That's why a bad driver can not only cause printing problems, but can cause *system* instability... even crashes. To try to keep things under better control, Microsoft has made an effort to reign-in sloppy drivers by creating a certification process. When you buy a new device, look for the certification logo. Also, when installing the driver, Windows will alert you if it does not meet the latest requirements, so think twice before allowing a non-conforming driver to install.

## Safe Computing

Even when you have your hardware, operating system, and drivers all working in perfect harmony, there is the potential for bad things to happen. That's why you must take precautions so that they do not cause a huge drop in your productivity. These are the issues I want to cover next.

## Viruses

A virus is a computer program that can replicate itself, and "infect" a computer without your knowledge or permission. A virus will inject itself into other programs, and then when you share those programs with other people, the virus will then replicate itself into their program files. A virus might do very little to your system, or it might delete your files or cause other mischief. Though viruses were one of the earliest threats to watch for, there are now many more to watch for, such as "worms", "root kits", and "Trojan horses".

When you buy a software package before you check for updates. I recommend you look for a good anti-virus package before you buy. There are many good options out there. I recommend you look for one that is not too expensive. You want one that is not too expensive. There are some free programs that are quite good.

## Spyware

Remember the old "Spyware" perhaps I was the only one who mind the other day when I heard, spyware is the latest that we didn't have enough and blue-screens-of-death list.

So if spyware is just another What makes it unique? Spyware activities without your computer. The most form of spyware is what's called "adware," and it generally is the least odious of the variants. It basically watches the internet sites you browse,

