CHAPTER 18

Input/Output

In this chapter, you will learn how to

- Explain how to support common input/output ports
- Identify certain common input/output devices on a PC
- Describe how certain specialty input/output devices work on a PC

In Chapter 3, "The Visible PC," you learned how to recognize and connect a number of common devices and the ports they use. Because these devices and their ports sometimes fail, it is important that you learn how they work and how to troubleshoot them when problems arise. This chapter reviews some of the major types of input ports, discusses a number of common and not-so-common input/output (I/O) devices, and deals with some of the troubleshooting issues you may encounter with I/O devices and their ports.

The CompTIA A+ certification exam domains split computer I/O devices into three groups: common, multimedia, and specialty. Common I/O devices, such as keyboards and mice, are found on virtually every PC. Multimedia I/O devices support video and sound functions. Specialty I/O devices run the gamut from common (touch screens) to rare (biometric devices). In fact, the exams deal with an entire set of I/O devices—networking devices—as completely distinct technologies. This book dedicates entire chapters to sound, printing, video, and networking, providing details about dealing with these types of devices and the ports they use. This chapter concentrates on two of the I/O device groups: the common devices and the specialty devices. You'll learn how to identify and support both the most common and some of the most unusual I/O devices used in today's PCs.

Essentials

Supporting Common I/O Ports

Whenever you're dealing with an I/O device that isn't playing nice, you need to remember that you're never dealing with just a device—you're dealing with a device and the port to which it is connected. Before you start looking at I/O devices, you need to take a look into the issues and technologies of some of the more common I/O ports and see what needs to be done to keep them running well.

EXAM TIP This entire chapter shows up in both of the CompTIA A+ certification exams, so it has no Practical Application section. Both of the CompTIA A+ exams test you on certain aspects of I/O devices, ports, configuration, and so on, so don't skip this chapter.

Serial Ports

Finding a new PC with a real serial port is difficult, because devices that traditionally used serial ports have for the most part moved on to better interfaces, in particular, USB. Physical serial ports may be hard to find on new PC cases, but many devices—in particular, the modems many people still use to access the Internet—continue to use built-in serial ports.

In Chapter 8, "Expansion Bus," you learned that COM ports are nothing more than preset I/O addresses and interrupt request lines (IRQs) for serial ports. Want to see a built-in serial port? Open Device Manager on a system and see if you have an icon for Ports (COM and LPT). If you do, click the plus (+) sign to the left of the icon to open it and see the ports on your system—don't be surprised if you have COM ports on your PC. Even if you don't see any physical serial ports on your PC, the serial ports are there; they're simply built into some other device, probably a modem.

NOTE Having trouble finding a PC with serial ports? Try a laptop—almost all laptops come with built-in modems.

Your PC's expansion bus uses parallel communication: multiple data wires, each one sending one bit of data at a time between your devices. Many I/O devices use serial communication: one wire to send data and another wire to receive data. The job of a *serial port* is to convert data moving between parallel and serial devices. A traditional serial port consists of two pieces: the physical, 9-pin DB connector (Figure 18-1) and a chip that actually does the conversion between the serial data and parallel data, called the *universal asynchronous receiver/transmitter* (*UART*) chip. If you want to be completely accurate, the UART *is* the serial port. The port on the back of your PC is nothing more than a standardized connector that enables different serial devices to use the serial port. The UART holds all of the smarts that make the true serial port.

NOTE Every UART in a system is assigned a COM port value. An internal modem snaps right into your expansion bus, so every internal modem has a built-in UART. Therefore, even though a modem doesn't have a physical serial connection, it most certainly has a serial port—a built-in one.

RS-232 is a very old standard that defines everything about serial ports: how fast they communicate, the language they use, even how the connectors should look. The RS-232 standard specifies that two serial devices must talk to each other in 8-bit chunks

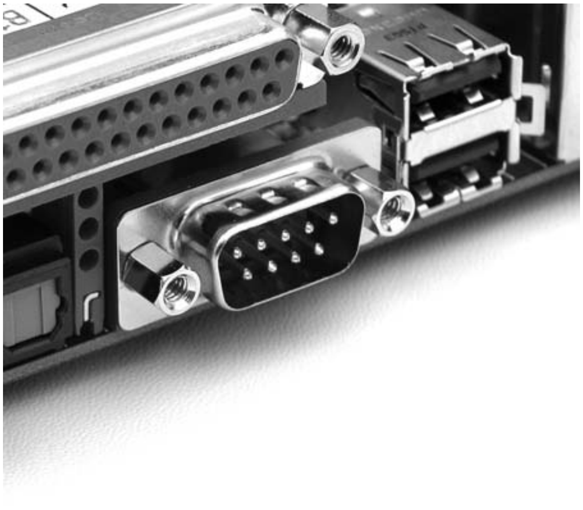


Figure 18-1
Serial port

of data, but it also allows flexibility in other areas, such as speed and error-checking. Serial came out back in the days when devices were configured manually, and the RS-232 standard has never been updated for automatic configuration. Serial ports are a throwback to the old days of computer maintenance (though they're still very prevalent in some hardware, such as high-end routers) and are the last manually configured port you'll find on a PC.

So what type of settings do you need to configure on a serial port? Find a PC with a real serial port (a real 9-pin connector on the back of the PC). Right-click the COM port and choose Properties to see the properties of that port in Device Manager. Open the Port Settings tab and click the Advanced button to see a dialog box that looks like Figure 18-2.

Devices such as modems that have built-in serial ports don't have COM port icons in Device Manager, because there's nothing to change. Can you see why? Even though these devices are using a COM port, that port is never going to connect to anything other than the device it's soldered onto, so all of the settings are fixed and unchangeable—thank goodness!

When you are configuring a serial port, you will have a lot of different settings to configure, many which may or may not make sense. The convenient part about all this is that when you get a new serial device to plug into your serial port, the instructions will tell you what settings to use. Figure 18-3 shows an instruction sheet for a Cisco switch.

NOTE If you need a serial port to support some older device but have a motherboard that doesn't have one, don't fret. You can always get a PCI expansion card with classic, 9-pin serial ports.

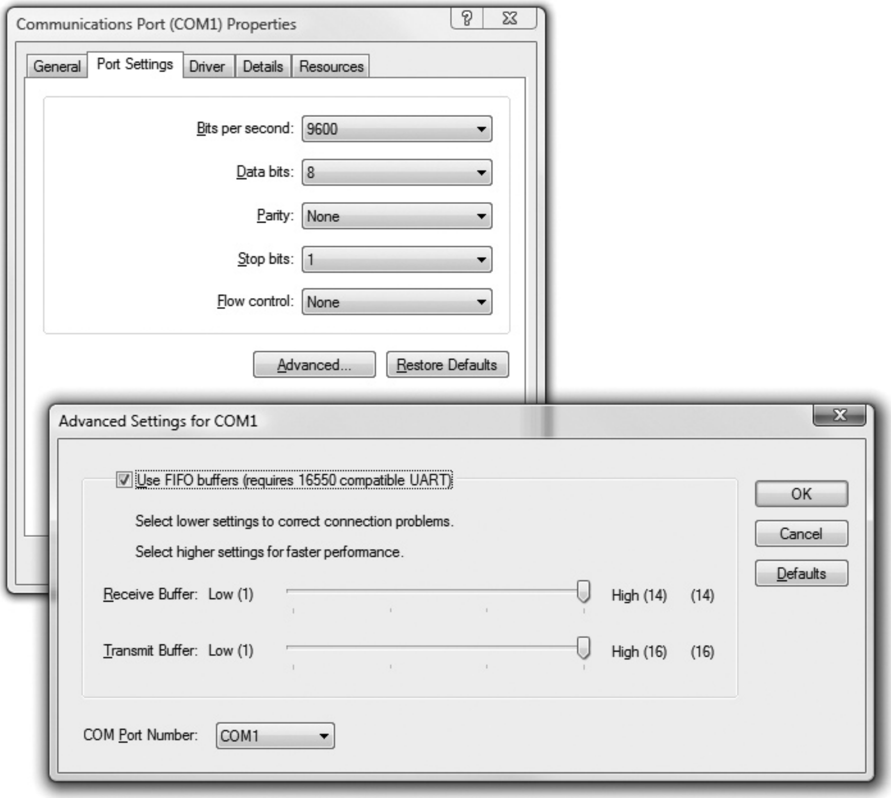
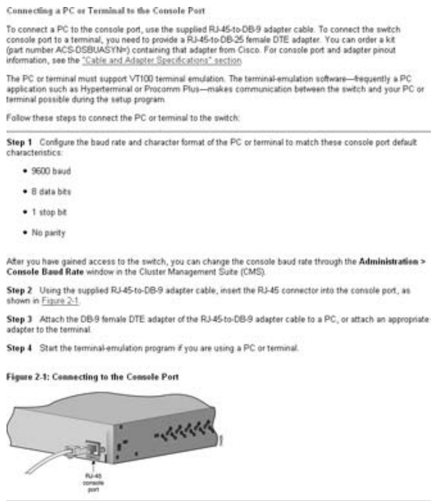


Figure 18-2 Serial port settings

Figure 18-3 Serial port instructions



Connecting a PC or Terminal to the Console Port

To connect a PC to the console port, use the supplied RJ-45-to-DB-9 adapter cable. To connect the switch console port to a terminal, you need to provide a RJ-45-to-DB-25 female DTE adapter. You can order a kit (part number ACS-DSBUASYN=) containing that adapter from Cisco. For console port and adapter pinout information, see the "Cable and Adapter Specifications" section.

The PC or terminal must support VT100 terminal emulation. The terminal-emulation software—frequently a PC application such as Hyperterminal or Procomm Plus—makes communication between the switch and your PC or terminal possible during the setup program.

Follow these steps to connect the PC or terminal to the switch:

Step 1 Configure the baud rate and character format of the PC or terminal to match these console port default characteristics:

- 9600 baud
- 8 data bits
- 1 stop bit
- No parity

After you have gained access to the switch, you can change the console baud rate through the **Administration > Console Baud Rate** window in the Cluster Management Suite (CMS).

Step 2 Using the supplied RJ-45-to-DB-9 adapter cable, insert the RJ-45 connector into the console port, as shown in Figure 2-1.

Step 3 Attach the DB-9 female DTE adapter of the RJ-45-to-DB-9 adapter cable to a PC, or attach an appropriate adapter to the terminal.

Step 4 Start the terminal-emulation program if you are using a PC or terminal.

Figure 2-1: Connecting to the Console Port

USB Ports

You should be familiar with the concept of USB, USB connectors, and USB hubs from the discussion of those concepts in Chapter 3, "The Visible PC." Here's a more in-depth look at USB and some of the issues involved with using USB devices.

Understanding USB

The cornerstone of a USB connection is the *USB host controller*, an integrated circuit that is usually built into the chipset and controls every USB device that connects to it. Inside the host controller is a *USB root hub:* the part of the host controller that makes the physical connection to the USB ports. Every USB root hub is really just a bus—similar in many ways to an expansion bus. Figure 18-4 shows a diagram of the relationship between the host controller, root hub, and USB ports.

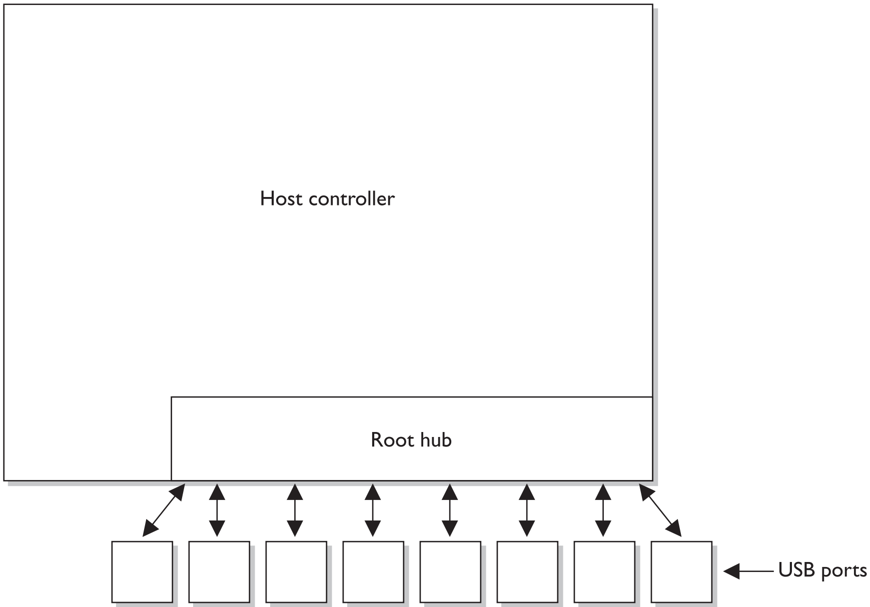


Figure 18-4 Host controller, root hub, and USB ports

No rule says how many USB ports a single host adapter may use. Early USB host adapters had two USB ports. The most recent ones support up to ten. Even if a host adapter supports a certain number of ports, there's no guarantee that the motherboard maker will supply that many ports. To give a common example, a host adapter might support eight ports while the motherboard maker only supplies four adapters.

The most important point to remember about this is that every USB device connected to a single host adapter/root hub *shares* that USB bus with every other device connected to it. The more devices you place on a single host adapter, the more the total

USB bus slows down and the more power they use. These issues are two of the biggest headaches that take place with USB devices in the real world.

USB devices, like any electrical device, need power to run, but not all take care of their own power needs. A powered USB device comes with its own electrical cord that is usually connected in turn to an AC adapter. *Bus-powered* USB devices take power from the USB bus itself; they don't bring any AC or DC power with them. When too many bus-powered devices take too much power from the USB bus, bad things happen—devices that work only some of the time and devices that lock up. You'll also often get a simple message from Windows saying that the hub power has been exceeded and it just won't work.

Every USB device is designed to run at one of three speeds. The first USB standard, version 1.1, defined two speeds: *Low-Speed USB*, running at a maximum of 1.5 Mbps (plenty for keyboards and mice), and *Full-Speed USB*, running up to 12 Mbps. Later, the USB 2.0 standard introduced *Hi-Speed USB* running at a whopping 480 Mbps. The industry sometimes refers to low-speed and Full-Speed USB as USB 1.1 and hi-speed as USB 2.0.

NOTE USB 2.0 defined more than just a new speed. Many low-speed and Full-Speed USB devices are also under the USB 2.0 standard.

In addition to a much faster transfer rate, Hi-Speed USB is fully backward compatible with devices that operate under the slower USB standards. Those old devices won't run any faster than they used to, however. To take advantage of the fastest USB speed, you must connect Hi-Speed USB devices to Hi-Speed USB ports by using Hi-Speed USB cables. Hi-speed USB devices function when plugged into Full-Speed USB ports, but at only 12 Mbps. Although backward compatibility at least allows you to use the newer USB device with an older port, a quick bit of math tells you how much time you're sacrificing when you're transferring a 240-MB file at 12 Mbps instead of 480 Mbps!

EXAM TIP The USB Implementers Forum (USB-IF) does not officially use "low-speed" and "full-speed" to describe 1.5 Mbps and 12 Mbps devices, calling both of them simply "USB." On the CompTIA A+ certification exams, though, you'll see the marketplace-standard nomenclature used here.

When USB 2.0 came out in 2001, folks scrambled to buy USB 2.0 controllers so their new hi-speed devices would work at their designed speeds. Of the variety of solutions people came up with, the most popular early on was to add a USB 2.0 adapter card like the one shown in Figure 18-5.

Motherboard makers quickly added a second USB 2.0 host controller—and they did it in a clever way. Instead of making the USB 2.0 host controller separate from the USB 1.1 host controller, they designed things so that both controllers share all of the connected USB ports (Figure 18-6). That way, no matter which USB port you choose, if you plug in a low-speed or full-speed device, the 1.1 host controller takes over, and if you plug in a hi-speed device, the USB 2.0 host controller takes over. Clever, and convenient!

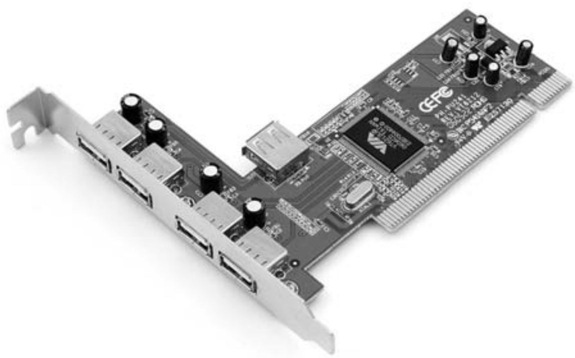


Figure 18-5 USB adapter card

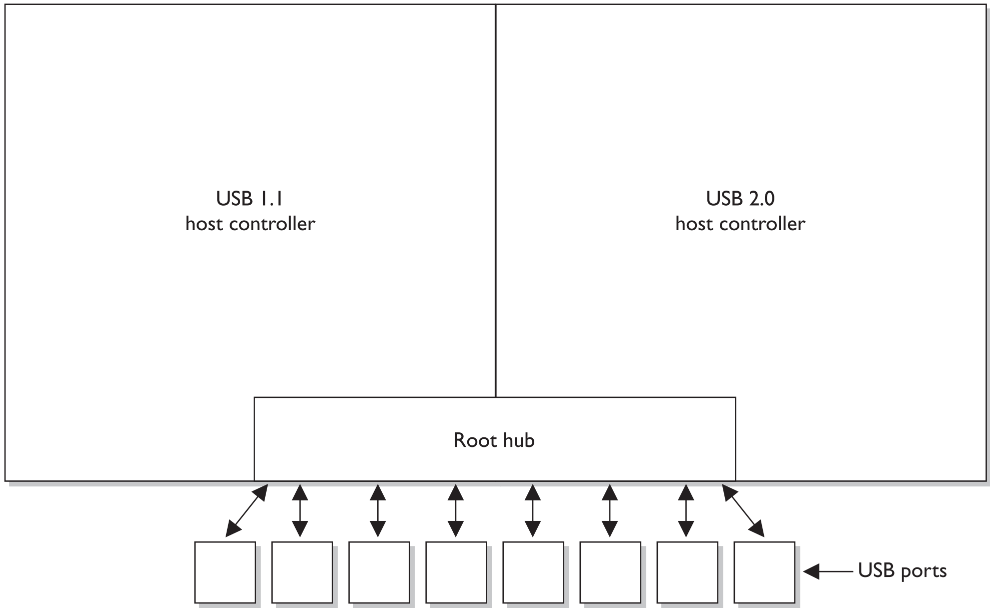


Figure 18-6 Shared USB ports

USB 2.0 has remained the standard for quite a while, but, as of this writing, the future of USB is nigh! USB 3.0 (also called SuperSpeed) devices are set to appear on the market sometime in 2010, with massively increased speed (up to 4.8 Gbps), increased power to peripherals, and full backward compatibility with older devices. USB 3.0 probably won't show up on the CompTIA A+ exams until it becomes widely adopted, but you should definitely be aware of it, because if it retains the popularity USB has enjoyed up to this point, it's going to be huge.

NOTE Using a PC running Windows 2000 or later, open the Device Manger and locate two controllers under the Universal Serial Bus icon. The one named Standard Enhanced Host Controller is the hi-speed controller. The Standard OpenHCD Host Controller is the low- and full-speed controller.

USB Hubs and Cables

Each USB host controller supports up to 127 USB devices, but as mentioned earlier, most motherboard makers provide only six to eight real USB ports. So what do you do when you need to add more USB devices than the motherboard provides ports? You can add more host controllers (in the form of internal cards), or you can use a USB hub. A *USB hub* is a device that extends a single USB connection to two or more USB ports, almost always directly from one of the USB ports connected to the root hub. Figure 18-7 shows a typical USB hub. USB hubs are sometime embedded into peripherals. The keyboard in Figure 18-8 comes with a built-in USB hub—very handy!

Figure 18-7
USB hub

Figure 18-8
USB keyboard with built-in hub

USB hubs are one of those parts of a PC that tend not to work nearly as well in the real world as they do on paper. (Sorry, USB folks, but it's true!) USB hubs have a speed just like any other USB device; for example, the hub in the keyboard in Figure 18-8 runs at full-speed. This becomes a problem when someone decides to insert a Hi-Speed USB device into one of those ports, as it forces the hi-speed device to crawl along at only 12 Mbps. Windows XP and Windows Vista are nice enough to warn you of this problem with a bubble over the system tray like the one shown in Figure 18-9.

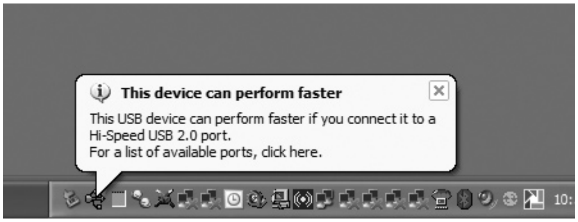


Figure 18-9 Windows XP speed warning

Hubs also come in powered and bus-powered versions. If you choose to use a general purpose USB hub like the one shown in Figure 18-7, try to find a powered one, as too many devices on a single USB root hub will draw too much power and create problems.

Cable length is an important limitation to keep in mind with USB. USB specifications allow for a maximum cable length of 5 meters, although you may add a powered USB hub every 5 meters to extend this distance. Although most USB devices never get near this maximum, some devices, such as digital cameras, can come with cables at or near the maximum 5-meter cable length. Because USB is a two-way (bi-directional) connection, as the cable grows longer, even a standard, well-shielded, 20-gauge, twisted-pair USB cable begins to suffer from electrical interference. To avoid these problems, I stick to cables that are no more than about 2 meters long.

If you really want to play it safe, spend a few extra dollars and get a high-quality USB 2.0 cable like the one shown in Figure 18-10. These cables come with extra shielding and improved electrical performance to make sure your USB data gets from the device to your computer safely.

USB Configuration

The biggest troubleshooting challenge you encounter with USB is a direct result of its widespread adoption and ease of use. Pretty much every modern PC comes with multiple USB ports, and anyone can easily pick up a cool new USB device at the local computer store. The problems arise when all of this USB installation activity gets out of control, with too many devices using the wrong types of ports or pulling too much power. Happily, by following a few easy steps, you can avoid or eliminate these issues.

The first and often-ignored rule of USB installation is this: Always install the device driver for a new USB device *before* you plug it into the USB port. Once you've installed the device and you know the ports are active (running properly in Device Manager), feel free to plug in the new device and hot-swap to your heart's content. USB device installation really is a breeze as long as you follow this rule!

NOTE There are exceptions to the "install the driver first" rule. USB thumb drives, for example, as you will recall from Chapter 13, "Removable Media," don't need extra drivers at all. Just plug them in and Windows picks them up. (Technically speaking, though, that means the drivers came *preinstalled* with the operating system!)

Figure 18-10
USB 2.0 cable

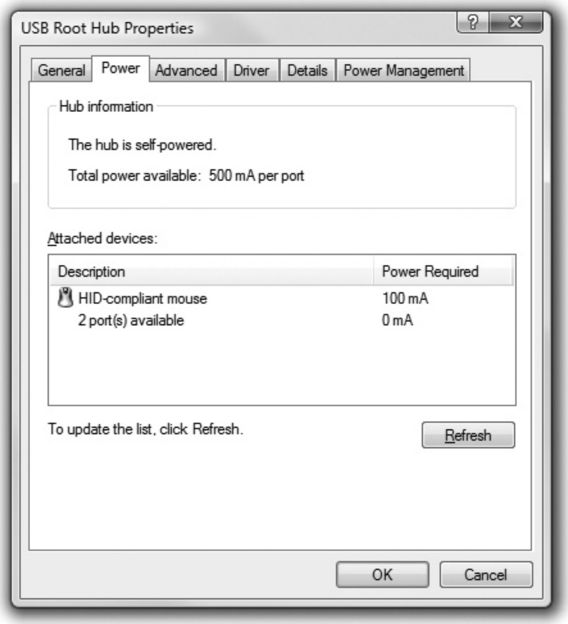
Windows 2000, XP, and Vista have a large number of built-in drivers for USB devices. You can count on Windows 2000, Windows XP, and Windows Vista to recognize keyboards, mice, and other basic devices with their built-in drivers. Just be aware that if your new mouse or keyboard has some extras, the default USB drivers will probably not support them. To be sure I'm not missing any added functionality, I always install the driver that comes with the device or an updated one downloaded from the manufacturer's Web site.

When looking to add a new USB device to a system, first make sure your machine has a USB port that supports the speed you need for the USB device. On more modern PCs, this is likely to be a nonissue. Even then, if you start adding hubs and such, you can end up with devices that either won't run at all or, worse yet, exhibit strange behaviors.

The last and toughest issue is power. A mismatch between available and required power for USB devices can result in nonfunctioning or malfunctioning USB devices. If you're pulling too much power, you must take devices off that root hub until the error goes away. Buy an add-in USB hub card if you need to use more devices than your current USB hub supports.

To check the USB power usage in Windows, open Device Manager and locate any USB hub under the Universal Serial Bus Controller icon. Right-click the hub and select Properties, and then select the Power tab. This shows you the current use for each of the devices connected to that root hub (Figure 18-11).

Figure 18-11 USB hub Power tab



NOTE The USB Hub Power Properties tab shows you the power usage only for a given moment, so to ensure you keep getting an accurate readings, you must click the Refresh button to update its display. Make sure your USB device works, and then refresh to see the maximum power used.

Most root hubs provide 500 mA per port—more than enough for any USB device. Most power problems take place when you start adding hubs, especially bus-powered hubs, and then you add too many devices to them. Figure 18-12 shows the Power tab for a bus-powered hub; note that it provides a maximum of 100 mA per port.

There's one more problem with USB power: sometimes USB devices go to sleep and don't wake up. Actually, the system is telling them to sleep, to save power. You can suspect this problem if you try to access a USB device that was working earlier but that suddenly no longer appears in Device Manager. To fix this, head back in to Device Manager to inspect the hub's Properties, but this time open the Power Management tab and uncheck the *Allow the computer to turn off this device to save power* checkbox, as shown in Figure 18-13.

Figure 18-12 General purpose bus-powered hub

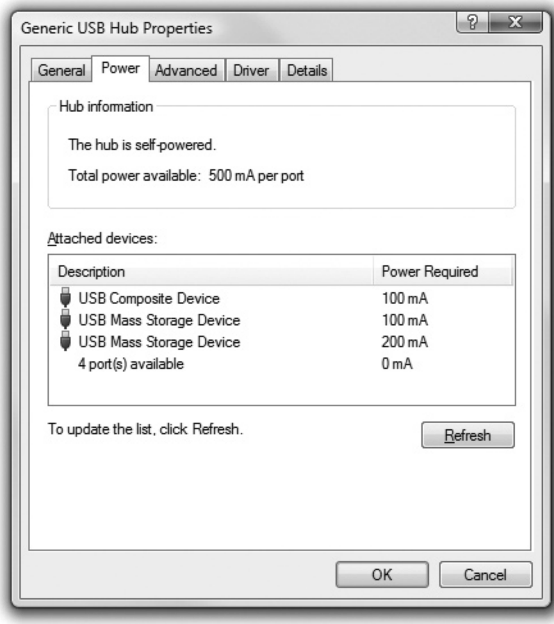
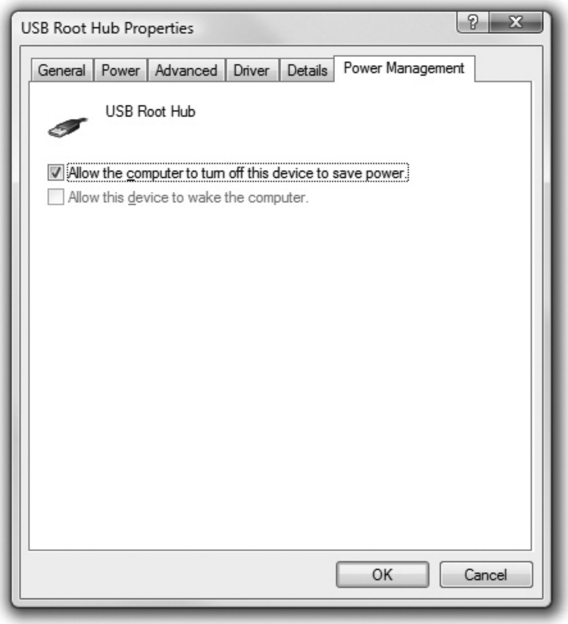


Figure 18-13 Power Management tab



FireWire Ports

At first glance, *FireWire*, also known as IEEE 1394, looks and acts much like USB. FireWire has all of the features of USB, but it uses different connectors and is actually the older of the two technologies. For years, FireWire had the upper hand when it came to moving data quickly to and from external devices. The onset of Hi-Speed USB changed that, and FireWire has lost ground to USB in many areas. One area where FireWire still dominates is editing digital video. Most modern digital video cameras use the IEEE 1394 interface for transferring video from camera to PC for editing. The high transfer speed of FireWire makes transferring large video files quick and easy.

NOTE Even Apple, the inventors of FireWire, dropped FireWire for USB in its iPod.

Understanding FireWire

FireWire has two distinct types of connectors, both of which are commonly found on PCs. The first is a 6-pin *powered* connector, the type you see on many desktop PCs. Like USB, a FireWire port is capable of providing power to a device, and it carries the same cautions about powering high-power devices through the port. The other type of connector is a 4-pin *bus-powered* connector, which you see on portable computers and such FireWire devices as cameras. This type of connector does not provide power to a device, so you need to find another method of powering the external device.

FireWire comes in two speeds: *IEEE 1394a*, which runs at 400 Mbps, and *IEEE 1394b*, which runs at 800 Mbps. FireWire devices can also take advantage of bus mastering, enabling two FireWire devices—such as a digital video camera and an external FireWire hard drive—to communicate directly with each other. When it comes to raw speed, FireWire 800—that would be 1394b, naturally—is much faster than Hi-Speed USB.

FireWire does have differences from USB other than just speed and a different-looking connector. First, a USB device must connect directly to a hub, but a FireWire device may use either a hub or daisy chaining. Figure 18-14 shows the difference between hubbed connections and daisy chaining. Second, FireWire supports a maximum of 63 devices, compared to USB's 127. Third, each cable in a FireWire daisy chain has a maximum length of 4.5 meters, as opposed to USB's 5 meters.

Configuring FireWire

FireWire was invented by and still controlled to a degree by Apple Computer. This single source of control makes FireWire more stable and more interchangeable than USB—in plain language, FireWire is ridiculously easy to use. In a Windows environment, FireWire is subject to many of the same issues as USB, such as the need to pre-install drivers, verify that onboard devices are active, and so on. But none of these issues is nearly as crucial with a FireWire connection. For example, as with USB, you really should install a FireWire device driver before attaching the device, but given that 95 percent of the FireWire devices used in PCs are either external hard drives or digital video connections, the pre-installed Windows drivers almost always work perfectly.

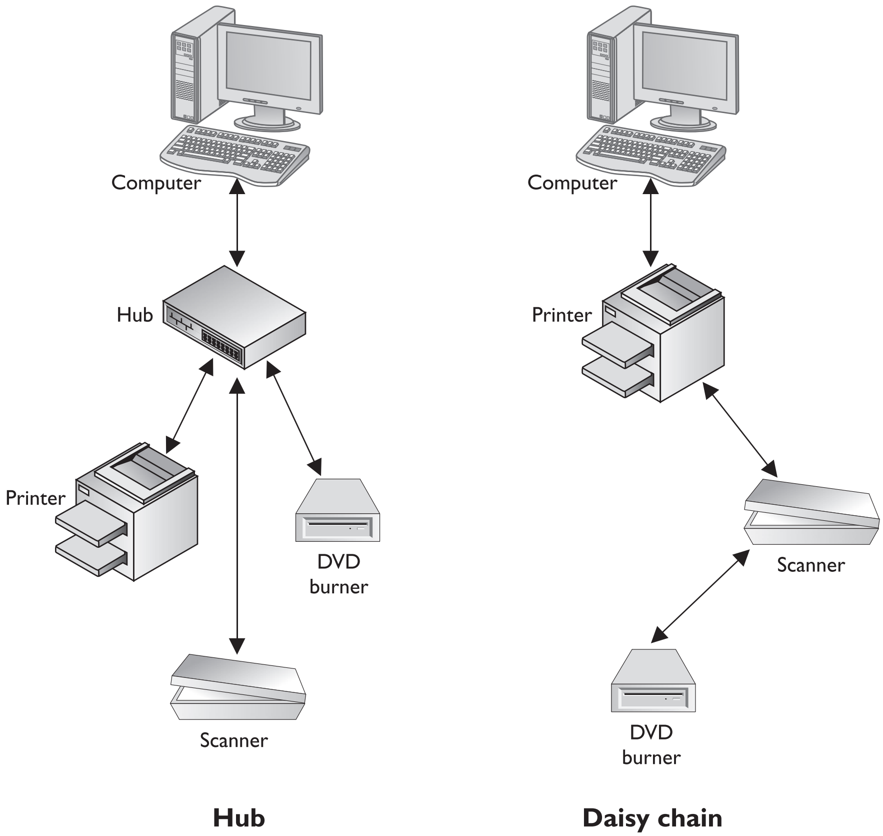


Figure 18-14 Hubbed versus daisy chain connections

FireWire devices do use much more power than USB devices, but the FireWire controllers are designed to handle higher voltages, and they'll warn you on the rare chance that your FireWire devices pull too much power.

General Port Issues

No matter what type of port you use, if it's not working, you should always check out a few issues. First of all, make sure you can tell a port problem from a device problem. Your best bet here is to try a second "known good" device in the same port to see if that device works. If it does *not*, you can assume the port is the problem. It's not a bad idea to reverse this and plug the device into a known good port.

NOTE A "known good" device is simply a device that you know is in good working order. All techs count heavily on the use of known good devices to check other devices. For example, if you think a PC has a bad keyboard, borrow one from the PC next door and see if that keyboard works on the broken machine.

If you're pretty sure the port's not working, you can check three things: First, make sure the port is turned on. Almost any I/O port on a motherboard can be turned off in CMOS. Reboot the system and find the device and see if the port's been turned off. You can also use Windows Device Manager to disable most ports. Figure 18-15 shows a disabled parallel port in Device Manager—you'll see a small down-pointing arrow in Windows Vista/7 or a red *X* over the device icon if you are using Windows 2000/XP or. To turn the port back on, right-click the device's icon and choose Enable.

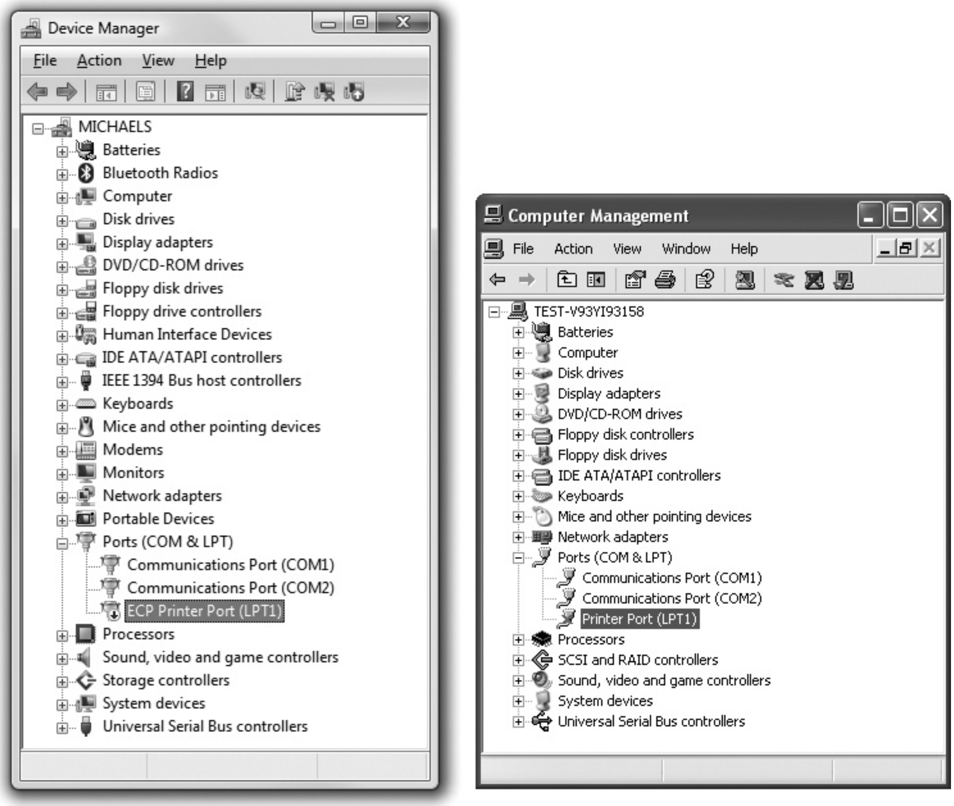
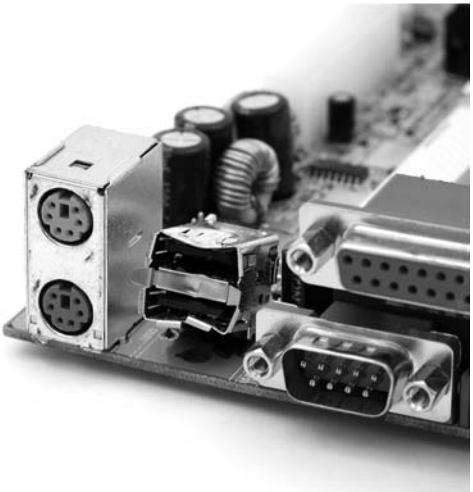


Figure 18-15 Disabled parallel port in Device Manager in both Vista and XP

Being able to turn off a port in Device Manager points to another not-so-obvious fact: ports need drivers just as devices need drivers. Windows has excellent built-in drivers for all common ports, so if you fail to see a port in Device Manager (and you know the port is turned on in CMOS), you can bet the port itself has a physical problem.

Because ports have connectors inserted and removed from them repeatedly, eventually they can physically break. Figure 18-16 shows the back of a USB port that's been pushed on too hard for too long and has physically separated from the motherboard.

Figure 18-16
Broken USB port



Unless you're an expert solderer, you either must stop using those ports or replace the entire motherboard.

Figure 18-17
Badly bent PS/2 plug

Many ports (or the plugs that fit into those ports) use tiny pins or relatively delicate metal casings that are susceptible to damage. PS/2 plugs are some of the worst for bent pins or misshaped casings. Figure 18-17 shows what happened to a PS/2 plug when I was in a hurry and thought that force was an alternative to lining up the plug properly. Replacement plugs are available—but again, unless you're excellent at soldering, they're not a viable alternative. Still, if you're patient, you might be able to save the plug. Using needle-nose pliers and a pair of scissors, I was able to reshape the plug so that it once again fit in the PS/2 port.

Common I/O Devices

So what is a "common" I/O device? I'm hoping you immediately thought of the mouse and the keyboard, two of the most basic, necessary, and abused I/O devices on a computer. Another fairly common input device that's been around a long time is the scanner. To these oldsters, you can add relative newcomers to the world of common devices: digital cameras and Web cameras.

NOTE If you want to get picky, these five common I/O devices enable a user only to *input* data; they don't provide any output at all.

Keyboards

Keyboards are both the oldest and still the primary way you input data into a PC. Windows comes with perfectly good drivers for any keyboard, although some fancier keyboards may come with specialized keys that require a special driver be installed to operate properly. About the only issue that might affect keyboard installation is if you're using a USB keyboard: make sure that the USB Keyboard Support option is enabled in your CMOS (Figure 18-18). Other than that, any keyboard installation issue you're likely to encounter is covered in the general port issues sections at the beginning of this chapter.

Figure 18-18 CMOS USB Keyboard Support option

TIP Wireless keyboards are a wonderful convenience because they remove the cable between you and the PC, but make sure you keep a complete set of spare batteries around.

There's not much to do to configure a standard keyboard. The only configuration tool you might need is the Keyboard Control Panel applet. This tool enables you to change the repeat delay (the amount of time you must hold down a key before the keyboard starts repeating the character), the repeat rate (how quickly the character is repeated after the repeat delay), and the default cursor blink rate. Figure 18-19 shows

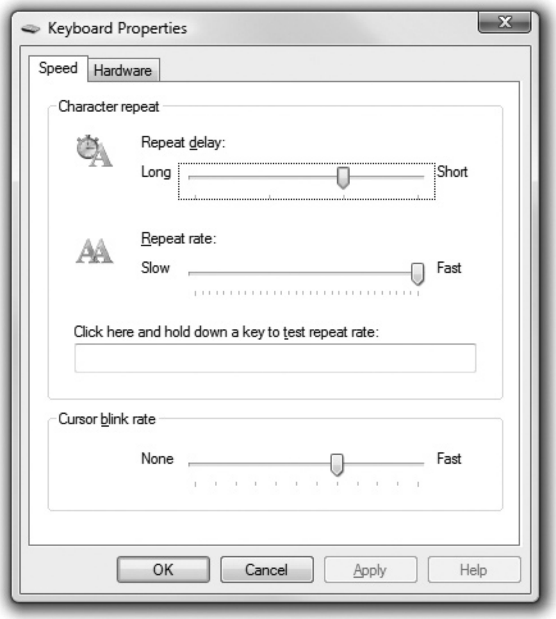


Figure 18-19 Keyboard Control Panel applet

the default Windows Keyboard Properties window—some keyboard makers provide drivers that add extra tabs.

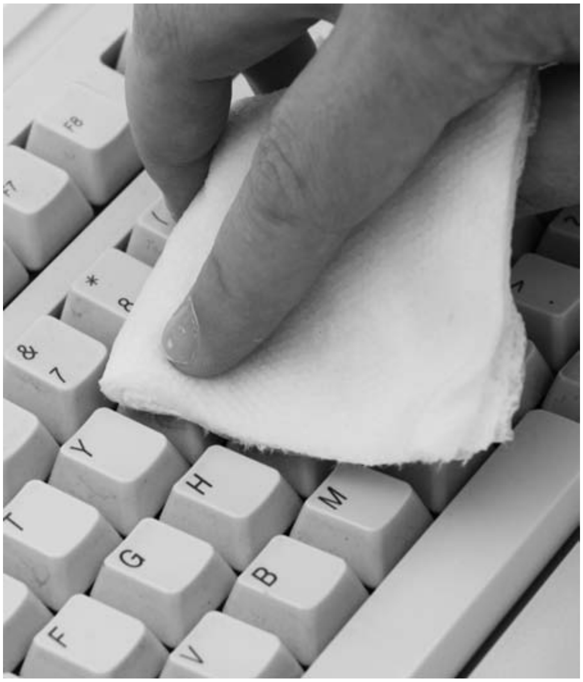
Keyboards might be easy to install, but they do fail occasionally. Given their location—right in front of you—the three issues that cause the most keyboard problems stem from spills, physical damage, and dirt.

Spilling a soda onto your keyboard can make for a really bad day. If you're quick and unplug the keyboard from the PC before the liquid hits the electrical components, you might be able to save the keyboard. It'll take some cleaning, though (keep reading for cleaning tips). More often than not, you'll get a sticky, ill-performing keyboard that is not worth the hassle—just replace it!

Other common physical damage comes from dropping objects onto the keyboard, such as a heavy book (like the one in your hands). This can have bad results! Most keyboards are pretty resilient, though, and can bounce back from the hit.

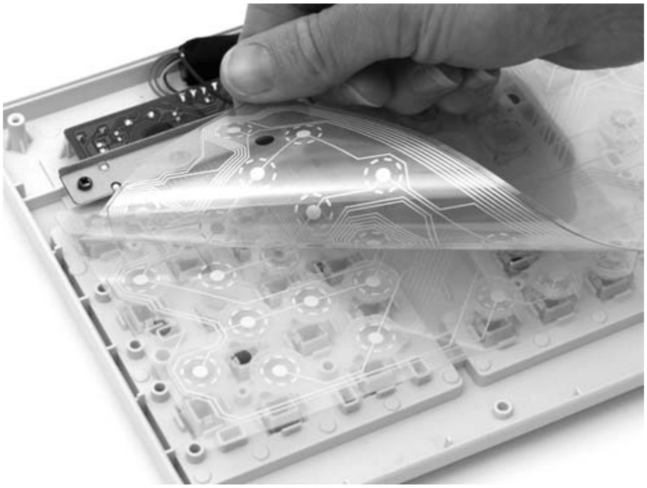
Clean dirt and grime off the keys by using a cloth dampened with a little water, or if the water alone doesn't do the job, use a bit of isopropyl alcohol on a cloth (Figure 18-20).

Figure 18-20
Cleaning keys



Dirty keys might be unsightly, but dirt under the keys might cause the keyboard to stop working completely. When your keys start to stick, grab a bottle of compressed air and shoot some air under the keys. Do this outside or over a trash can—you'll be amazed how much junk gets caught under the keys! If you really mess up a keyboard by dumping a chocolate milkshake on the keys, you're probably going to need to dismantle the keyboard to clean it. This is pretty easy as long as you keep track of where all of the parts go. Keyboards are made of layers of plastic that create the electrical connections when you press a key. Unscrew the keyboard (keep track of the screws!) and gently peel away the plastic layers, using a damp cloth to clean each layer (Figure 18-21). Allow the sheets to dry and then reassemble the keyboard.

Figure 18-21
Serious keyboard surgery



Sometimes dirt or foreign objects get under individual keys, requiring you to remove the key to get to the dirt or object. Removing individual keys from a keyboard is risky business, because keyboards are set up in many different ways. Most manufacturers use a process in which keys are placed on a single plastic post. In that case, you may use a screwdriver or other flat tool to safely pop off the key (Figure 18-22). Be careful! You'll need to use a good amount of force and the key will fly across the room. Other keyboard

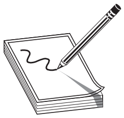
Figure 18-22
Prying off a key

makers (mainly on laptops) use tiny plastic pins shaped like scissors. In that case, beware—if you try prying one of these off, you'll permanently break the key!

The bottom line when it comes to stuck keys is that the keyboard's probably useless with the stuck key, so you might as well try to clean it. Worse comes to worst, you can always buy another keyboard.

Mice

Have you ever tried to use Windows without a mouse? It's not fun, but it can be done. All techs eventually learn the Windows navigation hot keys for those times when mice fail, but all in all we do love our mice. Like keyboards, Windows comes with excellent drivers for all standard mice; the exception you're likely to encounter is the more advanced mice that come with extra buttons. Conveniently, the built-in Windows drivers consider a mouse's scroll wheel to be standard equipment and will support it.



NOTE Everything in this section works equally well for trackballs.

You can adjust your mouse settings through the Mouse Control Panel applet. Figure 18-23 shows the Windows 2000 version. Be aware that the Mouse Properties window in Windows 2000 uses a different layout than that of Windows Vista (Figure 18-24) or Windows XP (which are almost identical).

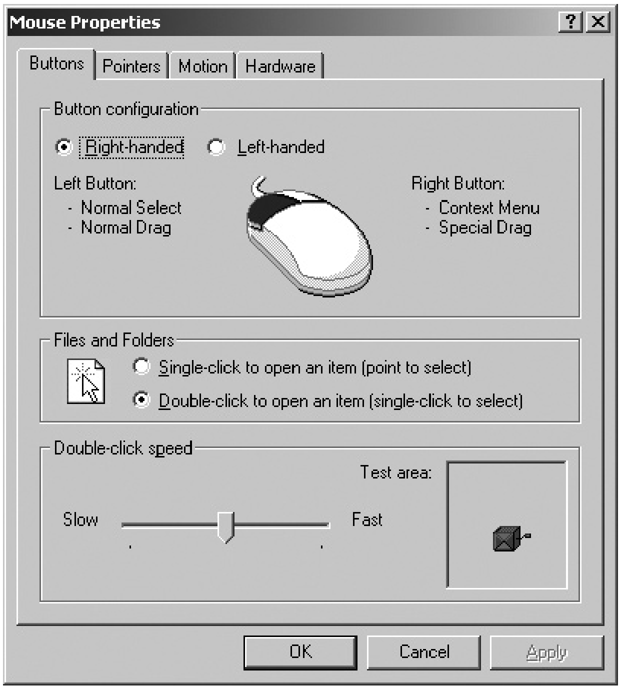


Figure 18-23 Windows 2000 Mouse Control Panel applet

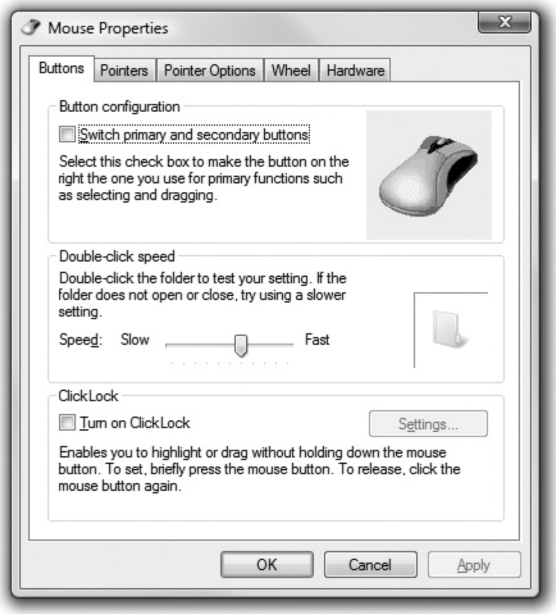


Figure 18-24 Windows Vista Mouse Control Panel applet

All of the settings you need for adjusting your mouse can be found in the Mouse Properties window. In particular, make sure to adjust the mouse speed, double-click speed, and acceleration to fit your preferences. Mouse speed and double-click speed are obvious, but mouse acceleration needs a bit of explaining as it has changed from Windows 2000 to Windows XP and Windows Vista. Originally, mouse *acceleration* referred to a feature that caused the mouse speed to increase when the mouse moved a relatively large distance across the screen. The Windows 2000 Mouse Properties window included a Motion tab where you could set the mouse speed and acceleration. Windows XP and Vista dropped the Motion tab in favor of an *Enhance pointer precision* checkbox on the Pointer Options tab (Figure 18-25). Enhance pointer precision is a much more advanced form of automatic acceleration. Although it works well, it can cause erratic mouse movements in some applications.

Currently, two types of mouse technologies dominate the market: ball mice and optical mice. *Ball mice* use a small round ball, while *optical mice* use LED or lasers and a camera to track their movements and thus move the mouse pointer across the screen. The problem with ball mice is that the ball inside the mouse picks up dirt over time and deposits the dirt on internal rollers that contact the ball. Dirt builds up to the point that the mouse stops responding smoothly. If you are struggling with your mouse to point at objects on your screen, you need to clean the mouse. Few mice manufacturers still make ball mice, as they tend to require far more maintenance than optical mice.

To access the internals of a ball mouse, turn it over and remove the protective cover over the mouse ball. The process of removing the cover varies, but it usually involves rotating the collar that surrounds the ball until the collar pops out (Figure 18-26). Be careful—without the collar, the mouse ball will drop out the instant you turn the mouse upright.

Figure 18-25 Enhance pointer precision checkbox on the Pointer Options tab

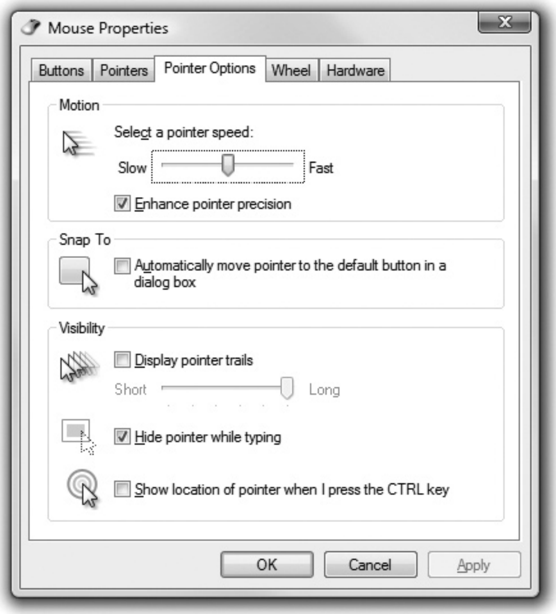


Figure 18-26 Removing the collar on a ball mouse

Use any nonmetallic tool to scrape the dirt from the roller without scratching or gouging the device. Although you could use a commercial "mouse cleaning kit," I find that a fingernail or a pencil eraser cleans the rollers quite nicely and at much less expense (Figure 18-27). Clean a ball mouse in this way at least every two or three months.

Figure 18-27
Cleaning the rollers on a ball mouse

Optical mice require little maintenance and almost never need cleaning, as the optics that make them work are never in contact with the grimy outside world. On the rare occasion where an optical mouse begins to act erratically, try using a cloth or damp cotton swab to clean out any bits of dirt that may be blocking the optics (Figure 18-28).

Figure 18-28
Cleaning an optical mouse

Scanners

You can use a scanner to make digital copies of existing paper photos, documents, drawings, and more. Better scanners give you the option of copying directly from a photographic negative or slide, providing images of stunning visual quality—assuming the original photo was halfway decent, of course! In this section, you'll look at how scanners work and then turn to what you need to know to select the correct scanner for you or your clients.

How Scanners Work

All consumer-level scanners—called *flatbed scanners*—work the same way. You place a photo or other object facedown on the glass, close the lid, and then use software to initiate the scan. The scanner runs a bright light along the length of the glass tray once or more to capture the image. Figure 18-29 shows an open scanner.

Figure 18-29 Scanner open with photograph face down

The scanning software that controls the hardware can be manifested in a variety of ways. Nearly every manufacturer has some sort of drivers and other software to create an interface between your computer and the scanner. When you push the front button on the Epson Perfection scanner in Figure 18-30, for example, the Epson software opens the Photoshop program as well as its own interface.

You can also open your favorite image-editing software first and choose to acquire a file from a scanner. Figure 18-31 shows the process of acquiring an image from a scanner in the popular shareware image-editing software, Paint Shop Pro. As in most such

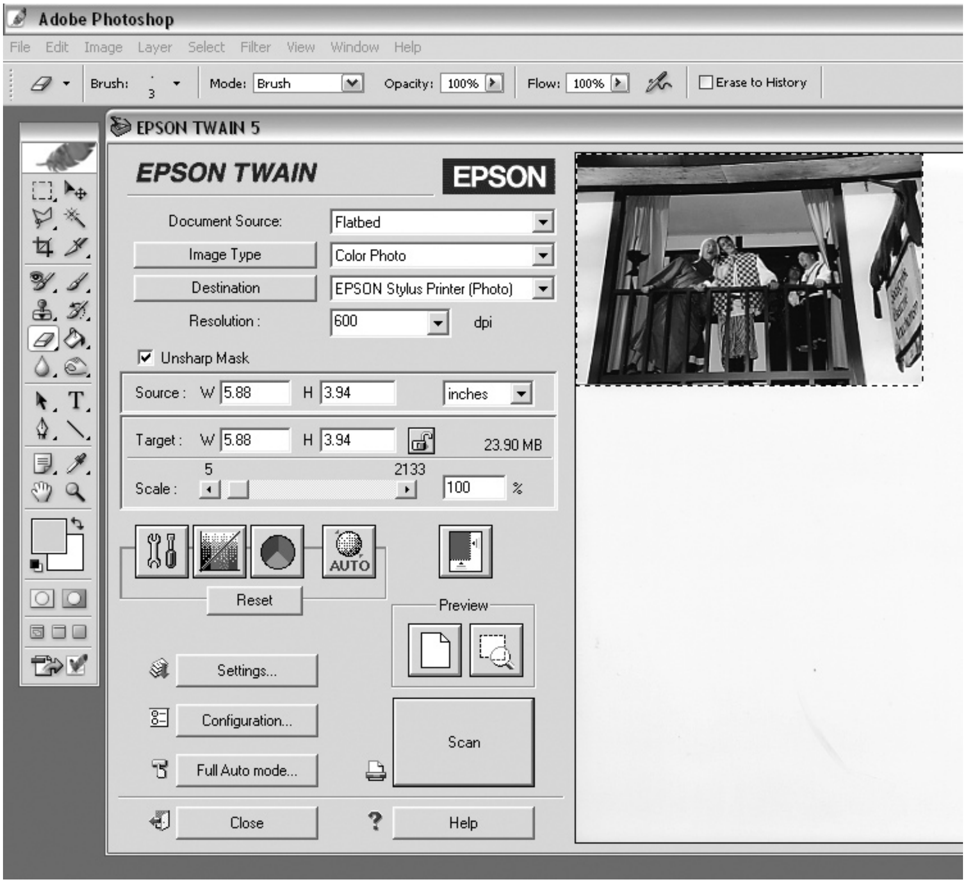


Figure 18-30 Epson software with Photoshop open in the background

software, you choose File | Import and then select a source. In this case, the scanner uses the traditional TWAIN drivers. *TWAIN* stands for *Technology Without an Interesting Name*—I'm not making this up!—and has been the default driver type for scanners for a long time.

At this point, the drivers and other software controlling the scanner pop up, providing an interface with the scanner (as shown in Figure 18-31). Here you can set the resolution of the image as well as many other options.

NOTE In addition to loading pictures into your computer, many scanners offer a feature called *optical character recognition (OCR)*, a way to scan a document and have the computer turn the picture into text that you can manipulate by using a word processing program. Many scanners come with OCR software, such as ABBYY FineReader.

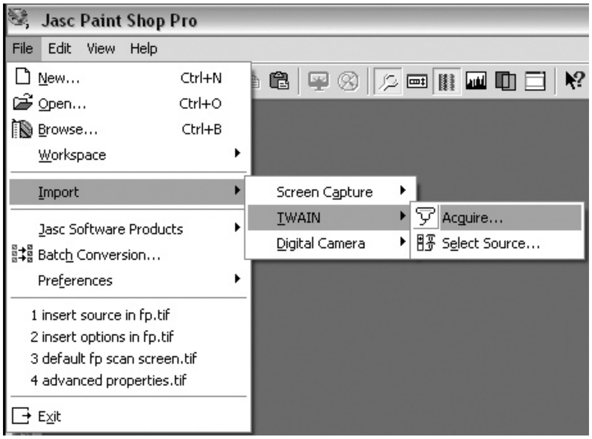


Figure 18-31 Acquiring an image in Paint Shop Pro

How to Choose a Scanner

You must consider five primary variables when choosing a scanner: resolution, color depth, grayscale depth, connection, and scan speed. You can and will adjust the first three during the scanning process, although probably only down from their maximum. You need to decide on the connection before you buy. The scan speed relates to all four of the other variables, and the maximum speed is hard-coded into the scanner.

Configurable Variables Scanners convert the scanned image into a grid of dots. The maximum number of dots determines how well you can capture an image and how the image will look when scaled up in size. Most folks use the term *resolution* to define the grid size. As you might imagine, the higher the resolution, the better the scanned image will look and scale.

Older scanners can create images of only 600 × 600 dots per inch (dpi), while newer models commonly achieve four times that density and high-end machines do much more. Manufacturers cite *two* sets of numbers for a scanner's resolution: the resolution it achieves mechanically—called the *optical resolution*—and the enhanced resolution it can achieve with assistance from some onboard software.

The enhanced resolution numbers are useless. I recommend at least 2400 × 2400 dpi optical resolution or better, although you can get by with a lower resolution for purely Web-destined images.

The *color depth* of a scan defines the number of bits of information the scanner can use to describe each individual dot. This number determines color, shade, hue, and so forth, so a higher number makes a dramatic difference in your picture quality. With binary numbers, each extra bit of information *doubles* the quality. An 8-bit scan, for example, can save up to 256 color variations per dot. A 16-bit scan, in contrast, can save up to 65,536 variations, not the 512 that you might expect!

Modern scanners come in 24-bit, 36-bit, and 48-bit variations. These days, 48-bit scanners are common enough that you shouldn't have to settle for less, even on a budget. Figures 18-32, 18-33, and 18-34 show pretty clearly the difference resolution makes when scanning.

Figure 18-32 Earring scanned at 72 dpi and 24-bit color

Figure 18-33 Same earring, scanned at 300 dpi and 24-bit color

Figure 18-34 Same earring, scanned at 1200 dpi and 24-bit color

Scanners differ a lot in *grayscale depth*, a number that defines how many shades of gray the scanner can save per dot. This matters if you work with black-and-white images in any significant way, because grayscale depth is usually a much lower number than color depth. Current consumer-level scanners come in 8-bit, 12-bit, and 16-bit grayscale varieties. I recommend 16-bit or better.

Connection Almost all modern scanners plug into the USB port on your PC, although some high-end models offer FireWire as well. Older scanners come in SCSI and parallel varieties.

Scanning Speed Scanners have a maximum scanning speed defined by the manufacturer. The time required to complete a scan is also affected by the parameters you set; the time increases as you increase the amount of detail captured. A typical low-end scanner, for example, takes upwards of 30 seconds to scan a 4 × 6 photo at 300 dpi. A faster scanner, in contrast, can crank out the same scan in 10 seconds.

Raise the resolution of the scan to 600 dpi at 48-bit resolution, and that faster scanner can take a full minute to complete the scan. Adjust your scanning settings to optimize for your project. Don't always go for the highest possible scan if you don't need the resolution.

Connections matter as well. A good Hi-Speed USB scanner can scan an 8 × 10 image in about 12 seconds at 300 dpi. I made the mistake of taking the scanner to a friend's house to scan some of her jewelry, but she had only a Full-Speed USB port. I plugged the scanner into her PC and it took about 45 seconds to scan each 8 × 10 image. We were up all night finishing the project!

Installation and Scanning Tips

Most USB and FireWire devices require you to install the software drivers before you plug in the device for the first time. I have run into exceptions, though, so I strongly suggest you read the scanner's documentation before you install.

As a general rule, you should obtain the highest quality scan you can manage, and then play with the size and image quality when it's time to include it in a Web site or an e-mail. The amount of RAM in your system—and to a lesser extent, the processor speed—dictates how big a file you can handle.

For example, don't do 8 × 10 scans at 600 dpi if you have only 128 MB of RAM, because the image file alone weighs in at over 93 MB. Because your operating system, scanner software, image-editing program, and a lot of other things are taking up plenty of that RAM already, your system will likely crash.

If you travel a lot, you'll want to make sure to use the locking mechanism for the scanner light assembly. Just be sure to unlock before you try to use it or you'll get a light that's stuck in one position. That won't make for very good scans!

EXAM TIP The CompTIA A+ certification Practical Application exam tests you more thoroughly on troubleshooting scanner problems and preventive maintenance issues than does the Essentials exam. Look for questions on using the locking mechanism, keeping the scanner surface clean, and avoiding scanning sharp objects that could damage the scanner.

Digital Cameras

Another option available for those not-yet-taken pictures is to put away your point-and-shoot film camera and use a digital camera. *Digital cameras* electronically simulate older film-technology and provide a wonderful tool for capturing a moment and then sending it to friends and relatives.

In a short period of time, digital camera prices have gone from levels that made them the province of a few wealthy technogeeks to being competitive with a wide range of electronic consumer goods. Because digital cameras interface with computers, CompTIA A+ certified techs need to know the basics.

Storage Media—Digital Film for Your Camera

Every consumer-grade camera saves the pictures it takes onto some type of *removable storage media*. Think of it as your digital film. Probably the most common removable storage media used in modern digital cameras (and probably your best choice) is the Secure Digital (SD) card (Figure 18-35). About the size of a Wheat Thin (roughly an inch square), you can find these tiny cards with capacities ranging from 64 MB to more than 1 GB. They are among the fastest of the various media types at transferring data to and from a PC, and they're quite sturdy.

Figure 18-35 Secure Digital card

Connection

These days, almost all digital cameras plug directly into a USB port (Figure 18-36). Another common option, though, is to connect only the camera's storage media to the computer, using one of the many digital media readers available.

You can find readers designed specifically for SD cards, as well as other types. Plenty of readers can handle multiple media formats. Many computers come with a decent built-in digital media reader (Figure 18-37).

Quality

As with scanners, you should consider the amount of information a particular model of camera can capture, which in the digital camera world is expressed as some number of *megapixels*. Instead of light-sensitive film, digital cameras have one CCD (charged

Figure 18-36
Camera connecting to USB port

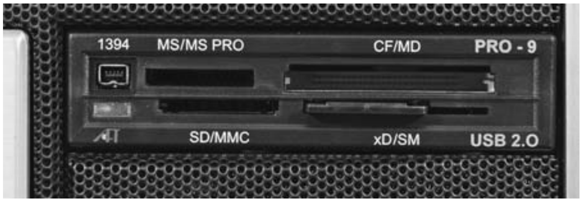


Figure 18-37
Digital media reader built into computer

coupled device) or CMOS (complementary metal-oxide semiconductor) sensor covered with photosensitive pixels (called *photosites*) to capture the image; the more pixels on the sensor, the higher the resolution of the images it captures.

Not so long ago, a 1-megapixel digital camera was the bleeding edge of digital photographic technology, but now you can find cameras with 10 times that resolution for a few hundred dollars. As a basis of reference, a 2-megapixel camera produces snapshot-sized (4 × 6 inch) pictures with print photograph quality, whereas a 5-megapixel unit can produce a high-quality 8 × 10 inch print.

Another feature of most digital cameras is the capability to zoom in on your subject. The way you ideally want to do this is the way film cameras do it, by using the camera's optics—that's the lens. Most cameras above the basic level have some *optical zoom*—meaning the zoom is built into the lens of the camera—but almost all models include multiple levels of *digital zoom*, accomplished by some very clever software in the camera. Choose your camera based on optical zoom: 3× at a minimum or better if you can afford it. Digital zoom is useless.

Form Factor

As was the case with film cameras, size matters on digital cameras. Digital cameras come in several form factors. They range from tiny, ultra-compact models that readily fit in a shirt pocket to monster cameras with huge lenses. Although it's not universally true, the bigger the camera, the more features and sensors it can have. Thus bigger is usually better in terms of quality. In shape, they come in a rectangular package, in which the lens retracts into the body, or as an SLR-type, with a lens that sticks out of the body. Figure 18-38 shows both styles.

Figure 18-38 Typical digital cameras

Web Cameras

PC cameras, often called *webcams* because their most common use is for Internet video communication, are fairly new to the world of common I/O devices. Too many people run out and buy the cheapest one, not appreciating the vast difference between a discount webcam and more expensive models; nor do they take the time to configure the webcam properly. Let's consider some of the features you should look for when buying webcams and some of the problems you can run into when using them.

The biggest issue with webcams is the image quality. Webcams measure their resolution in pixels. You can find webcams with resolutions of as few as 100,000 pixels and webcams with millions of pixels. Most people who use webcams agree that 1.3 million pixels (megapixels) is pretty much the highest resolution quality you can use before your video becomes so large it will bog down even a broadband connection.

The next issue with webcams is the frame rate, that is, the number of times the camera "takes your picture" each second. Higher frame rates make for smoother video; 30 frames per second is considered the best. A good camera with a high megapixel resolution and fast frame rate will provide you with excellent video conferencing capabilities. Figure 18-39 shows the author using his headset to chat via webcam using Skype software.

NOTE Read more about pixels and frame rates in Chapter 19, "Video."

Most people who use online video also want a microphone. Many cameras come with microphones, or you can use your own. Those who do a lot of video chatting may prefer to get a camera without a microphone and then buy a good quality headset with which to speak and listen.

Many cameras now can track you when you move, to keep your face in the picture—a handy feature for fidgety folks using video conferencing! This interesting technology recognizes a human face with little or no "training" and rotates its position to keep your face in the picture. Some companies even add funny extras, which, although not very productive, are good for a laugh (Figure 18-40).

Almost all webcams use USB connections. Windows comes with a limited set of webcam drivers, so always make sure to install the drivers supplied with the camera

Figure 18-39 Video chatting by webcam with Skype

Figure 18-40 This webcam program's animated character mirrors your movements as you conference with friends or coworkers.

before you plug it in. Most webcams use Hi-Speed USB, so make sure you're plugging your webcam into a Hi-Speed USB port.

Once the camera's plugged in, you'll need to test it. All cameras come with some type of program, but finding the program can be a challenge. Some brands put the program in the system tray, some place it in My Computer, others put it in the Control Panel—and some do all three! Figure 18-41 shows the Control Panel applet that appeared when I installed the webcam driver.

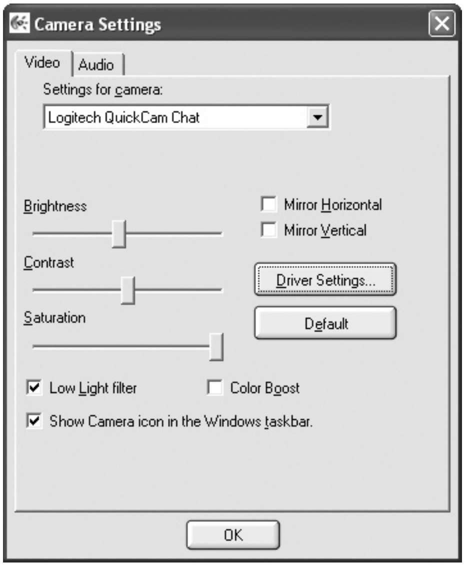


Figure 18-41 Camera Settings applet

The biggest challenge to using webcams is getting your webcam applications to recognize that your webcam is available and configured for use. Every program does this differently, but conceptually the steps are basically the same (with plenty of exceptions):

1. Tell the program you want to use a camera.
2. Tell the program whether you want the camera to turn on automatically when you chat.
3. Configure the image quality.
4. Test the camera.

If you're having problems with a camera, always go through the general I/O problems first, as this will clear up most problems. If you're still having trouble getting the camera to work in a program, be sure to turn off all other programs that may be using the camera. Windows allows only one program at a time to use a webcam.

Specialty I/O Devices

The CompTIA A+ certification exams want to make sure you're aware of four other types of I/O devices: biometric scanners, bar code readers, touch screens, and KVM switches. Let's look at these fairly specialized devices.

Biometric Devices

If you look up *biometrics* on the popular Wikipedia Web site, you'll get the following definition: "Biometrics (ancient Greek: *bios* ='life,' *metron* ='measure') is the study of automated methods for uniquely recognizing humans based upon one or more intrinsic physical or behavioral traits." (The quote might differ from what you find because Wikipedia changes pretty frequently, but the gist should be the same.)

The field of biometrics also encompasses a number of security devices, such as door locks and security cameras, that don't really fit into the world of PCs. This section concentrates on the types of biometrics that you can actually buy and use on your PC. Within the realm of computers, biometrics includes a huge number of technologies, from thumb drives that read fingerprints to software that does voice recognition.

PCs use biometrics for security. *Biometric devices* scan and remember unique aspects of various body parts such as your retina, iris, head image, or fingerprint, using some form of sensing device such as a retinal scanner. This information is used as a key to prevent unauthorized people from accessing whatever the biometric device is securing. Most biometric devices currently used in PCs secure only themselves. The USB thumb drive in Figure 18-42 has a tiny fingerprint scanner. You slide your finger (any finger you choose) over the drive to unlock the contents of the thumb drive.

Figure 18-42 USB thumb drive with fingerprint scanner (*photo courtesy of Lexar Media, Inc*)

Less common are biometric security devices that secure entire computers. The Microsoft fingerprint scanner is a USB device that replaces standard user name and password security. Figure 18-43 shows the scanner built into a keyboard. When a program or Web site asks for a user name and password, you simply press your finger against the fingerprint scanner. It confirms your identity (assuming your fingerprint matches), and then special software that comes with the scanner supplies the program or Web site with your stored user name and password.

Biometric devices are also used for recognition. Recognition is different from security in that the biometric device doesn't care who you are, it just wants to know what you're doing. The best example of this is voice recognition. Voice recognition programs convert human voice input into commands or text. Voice recognition for PCs has been around for some time. Although it has never achieved enough accuracy to replace a keyboard completely, voice recognition is common in devices that have a limited number of commands to interpret, such as cell phones and PDAs. If you speak the words

Figure 18-43 Microsoft fingerprint scanner on a keyboard

"Call Mike Meyers" into your smartphone, your phone knows what to do—at least, *my* phone does!

No matter what biometric device you use, you use the same steps to make it work:

1. Install the device.
2. Register your identity with the device by sticking your eye, finger, or other unique body part (Why are you snickering?) into the device so it can scan you.
3. Configure its software to tell the device what to do when it recognizes your scanned identity.

Bar Code Readers

Bar code readers are designed to read standard *Universal Product Code* (*UPC*) bar codes (Figure 18-44). We read bar codes for only one reason—to track inventory. Bar code readers enable easy updating of inventory databases stored on PCs. Bar code readers are just about the oldest "specialty" I/O device used with PCs.

Figure 18-44 Typical UPC code

Two types of bar code readers are commonly found with PCs: pen scanners and hand scanners. Pen scanners look like an ink pen and must be swiped across the bar code (Figure 18-45). Hand scanners are held in front of the UPC code while a button is pressed to scan. All bar code readers emit a tone to let you know the scan was successful.

Older bar code readers used serial ports, but all of the newer readers use either PS/2 or USB ports. No configuration is usually necessary, other than making sure that the particular bar code reader works with whatever database/point of sale software you use. When in doubt, most people find the PS/2-style bar code readers work best, as they

Figure 18-45
Pen scanner (*photo courtesy of Wasp® Barcode Technologies*)

simply act like a keyboard. You plug the reader into your keyboard port and then plug your keyboard into the reader. Then all you need is software that accepts keyboard input (and what one doesn't!), and it will work.

Touch Screens

A *touch screen* is a monitor with some type of sensing device across its face that detects the location and duration of contact, usually by a finger or stylus. All touch screens then supply this contact information to the PC as though it were a click event from a mouse. Touch screens are used in situations for which conventional mouse/keyboard input is either impossible or impractical. Here are a few places you'll see touch screens at work:

- Information kiosks
- PDAs
- Point of sale systems
- Tablet PCs

Touch screens can be separated into two groups: built-in screens like the ones in PDAs, and standalone touch screen monitors like those used in many point of sale systems. From a technician's standpoint, you can think of a standalone touch screen as a monitor with a built-in mouse. All touch screens have a separate USB or PS/2 port for the "mouse" part of the device, along with drivers you install just as you would for any USB mouse.

KVM switches

A *keyboard, video, mouse* (*KVM*) *switch* is a hardware device that most commonly enables multiple computers to be viewed and controlled by a single mouse, keyboard, and screen. Some KVMs reverse that capability, enabling a single computer to be controlled by multiple keyboards, mice, or other devices. KVMs are especially useful in data centers where multiple servers are rack mounted, space is limited, and power is a concern.

An administrator can use a single KVM to control multiple server systems from a single keyboard, mouse, and monitor.

There are many brands and types of KVM switches. Some enable you to connect to only two systems, and some support hundreds. Some even come with audio output jacks to support speakers. Typical KVMs come with two or more sets of wires that are used for input devices such as PS/2 and/or USB mice and video output (Figure 18-46).

Figure 18-46
A typical KVM switch

To use a KVM, you simply connect a keyboard, mouse, and monitor to the KVM and then connect the KVM to the desired computers. Once connected and properly configured, assigned keyboard hotkeys—a combination of keys typically assigned by the KVM manufacturer—enable you to toggle between the computers connected to the KVM. In most cases, you simply tap the Scroll Lock key twice to switch between sessions.

Installing a KVM is not difficult; the most important point to remember is to connect the individual sets of cables between the KVM ports and each computer one at a time, keeping track of which keyboard, mouse, and video cable go to which computers. (I highly recommend labeling and using twist or zip-ties.)

If you get the connections wrong, the KVM won't function as desired. If you connect a mouse and keyboard wires to the correct KVM port, for example, but attach the same computer's video cable to a different port on the KVM, you won't get the correct video when you try to switch to that computer. The same holds true for the mouse and keyboard cables. Don't cross the cables!

NOTE Older KVMs are said to be passive, meaning they don't continuously communicate with all connected systems. This can cause problems if the connected systems automatically reboot after a power surge or loss. Modern day active KVMs resolve this issue through *peripheral emulation*, meaning they communicate with and monitor all systems connected to the KVM.

Chapter Review Questions

1. A serial port receives and sends serial data. What device translates that serial data into parallel data for the computer to use?
   - A. Parallel translator chip
   - B. Serial translator chip
   - C. COM chip
   - D. UART chip
2. What integrated circuit device controls USB devices connected to a USB port?
   - A. Host controller
   - B. IC-USB
   - C. Serial port
   - D. UART
3. What happens to bus speed and power usage when you plug multiple devices into a USB hub?
   - A. The bus speed stays constant, but power usage increases.
   - B. The bus speed increases because each device brings a little burst; power usage increases.
   - C. The bus speed decreases because all devices share the same total bandwidth; power usage increases.
   - D. The bus speed decreases because all devices share the same total bandwidth; power usage decreases.
4. Which port type offers the fastest transfer speed?
   - A. IEEE 1394a
   - B. IEEE 1394b
   - C. Full-Speed USB
   - D. Hi-Speed USB
5. You take a tech call from a user who complains that she gets an error message, "Hub power exceeded," when she plugs her new thumb drive into her USB keyboard's external USB port. Worse, the device won't work. What's most likely the problem?
   - A. Her USB port is defective.
   - B. She has a defective thumb drive.
   - C. She plugged a hi-speed device into a full-speed port.
   - D. She plugged one too many devices into the USB hub.
6. What is the fastest speed that Hi-Speed USB 2.0 can go?
   - A. 12 Mbps
   - B. 120 Mbps

C. 400 Mbps

D. 480 Mbps

7. USB 1.1 devices can run at two speeds. What are the speeds?

A. 1 and 2 Mbps

B. 1.5 and 12 Mbps

C. 1.5 and 15 Mbps

D. 12 and 48 Mbps

8. What's the maximum cable length for USB?

A. 1.2 meters

B. 1.2 yards

C. 5 meters

D. 5 feet

9. Which of the following mice technologies most needs to be cleaned?

A. Ball

B. Optical

C. Parallel

D. Serial

10. If you attempt to scan an item and the scanner light assembly does not move, what is most likely the problem?

A. The scanner is frozen.

B. The scanner is broken.

C. The scanner light assembly is locked.

D. The scanner light assembly is resetting.

Answers

1. **D.** The UART handles the serial to parallel and parallel to serial translation.
2. **A.** The host controller controls USB devices plugged into the USB bus via a USB port.
3. **C.** The bus speed decreases because all devices share the same total bandwidth; power usage increases.
4. **B.** FireWire 800 easily spanks the competition here.
5. **D.** Just like the error message said, the thumb drive drew too much power for the hub to handle.
6. **D.** Hi-speed USB 2.0 has a theoretical maximum of 480 Mbps.
7. **B.** USB 1.1 devices can run at either 1.5 Mbps or 12 Mbps.
8. **C.** USB has a maximum cable length of 5 meters.
9. **A.** Ball mice get the dirtiest.
10. **C.** The scanner light assembly is most likely locked.