**Keyboard Technology**

The technology that makes up a typical PC keyboard is very interesting. This section focuses on all the aspects of keyboard technology and design, including the key switches, the interface between the keyboard and the system, the scan codes, and the keyboard connectors.

**Key switch Design**

Today's keyboards use any one of several switch types to create the action for each key. Most keyboards use a variation of the mechanical key switch. A mechanical key switch relies on a mechanical momentary contact-type switch to make the electrical contact that forms a circuit. Some high-end keyboards use a more sophisticated design that relies on capacitive switches. This section discusses these switches and the highlights of each design.

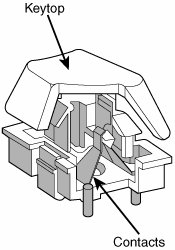
The most common type of key switch is the mechanical type, available in the following variations:

* Pure mechanical
* Foam element
* Rubber dome
* Membrane

**Pure Mechanical Switches**

The pure mechanical type is just that a simple mechanical switch that features metal contacts in a momentary contact arrangement. The switch often includes a tactile feedback mechanism, consisting of a clip and spring arrangement designed to give a "clicky" feel to the keyboard and offer some resistance to the key press (see Figure 16.3).

**Figure 16.3. A typical mechanical switch used in older NMB keyboards. As the key is pressed, the switch pushes down on the contacts to make the connection.**



Mechanical switches are very durable, usually have self-cleaning contacts, and are normally rated for 20 million keystrokes (which is second only to the capacitive switch in longevity). They also offer excellent tactile feedback.

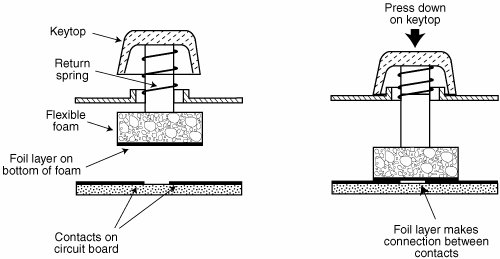
Despite the tactile feedback and durability provided by mechanical key switch keyboards, they have become much less popular than membrane keyboards (discussed later in this chapter). In addition, many companies that produce keyboards that use mechanical key switches either use them for only a few of their high-priced models or have phased out their mechanical key switch models entirely. With the price of keyboards nose-diving along with other traditional devices, such as mice and drives , the pressure on keyboard makers to cut costs has led many of them to abandon or de- emphasize mechanical-key switch designs in favor of the less expensive membrane key switch.

The Alps Electric mechanical key switch is used by many of the vendors who produce mechanical-switch keyboards, including Alps Electric itself. Other vendors who use mechanical keyswitches for some of their keyboard models include Adesso, Inc. (www.adesso.com), Avant Prime and Stellar (revivals of the classic Northgate keyboards and available from Ergonomic Resources; www.ergo-2000.com), Kinesis (www.kinesis-ergo.com), and SIIG (www.siig.com). Many of these [vendors](http://flylib.com/books/en/4.249.1.66/1/) sell through the OEM market, so you must look carefully at the detailed specifications for the keyboard to see whether it is a mechanical keyswitch model.

**Foam Element Switches**

Foam element mechanical switches were a very popular design in some older keyboards. Most of the older PC keyboards, including models made by Key Tronic and many others, used this technology. These switches are characterized by a foam element with an electrical contact on the bottom. This foam element is mounted on the bottom of a plunger that is attached to the key (see Figure 16.4).

**Figure 16.4. Typical foam element mechanical keyswitch.**



When the switch is pressed, a foil conductor on the bottom of the foam element closes a circuit on the printed circuit board below. A return spring pushes the key back up when the pressure is released.

The foam dampens the contact, helping to prevent bounce, but unfortunately it gives these keyboards a "mushy" feel. The big problem with this type of keyswitch design is that little tactile feedback often exists. These types of keyboards send a clicking sound to the system speaker to signify that contact has been made. Preferences in keyboard feel are somewhat [subjective](http://flylib.com/books/en/3.287.1.147/1/) ; I personally do not favor the foam element switch design.

Another problem with this type of design is that it is more subject to corrosion on the foil conductor and the circuit board traces below. When this happens, the key strikes can become intermittent, which can be frustrating. Fortunately, these keyboards are among the easiest to clean. By disassembling the keyboard completely, you usually can remove the circuit board portionwithout removing each foam pad separatelyand expose the bottoms of all the pads. Then, you easily can wipe the corrosion and dirt off the bottoms of the foam pads and the circuit board, thus restoring the keyboard to a "like-new" condition. Unfortunately, over time, the corrosion problem will occur again. I recommend using some Stabilant 22a from D.W. Electrochemicals (www.stabilant.com) to improve the switch contact action and prevent future corrosion. Because of such problems, the foam element design is not used much anymore and has been superseded in popularity by the rubber dome design.

KeyTronicEMS, the most well-known user of this technology, now uses a center- [bearing](http://flylib.com/books/en/2.366.1.39/1/) membrane switch technology in its keyboards, so you are likely to encounter foam-switch keyboards only on very old systems.

**Rubber Dome Switches**

Rubber dome switches are mechanical switches similar to the foam element type but are improved in many ways. Instead of a spring, these switches use a rubber dome that has a carbon button contact on the underside. As you press a key, the key plunger presses on the rubber dome, causing it to resist and then collapse all at once, much like the top of an oil can. As the rubber dome collapses, the user feels the tactile feedback, and the carbon button makes contact between the circuit board traces below. When the key is released, the rubber dome re-forms and pushes the key back up.

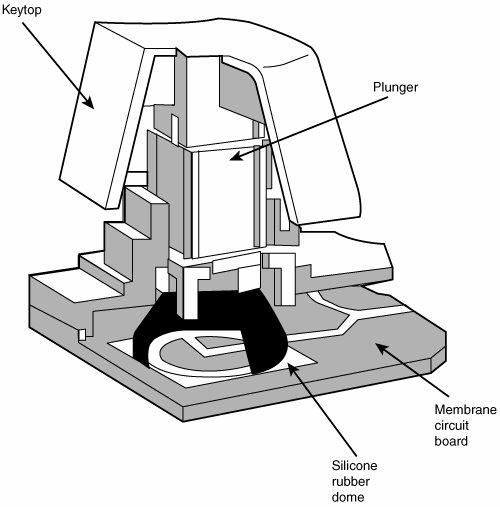
The rubber eliminates the need for a spring and provides a reasonable amount of tactile feedback without any special clips or other parts . Rubber dome switches use a carbon button because it resists corrosion and has a self-cleaning action on the metal contacts below. The rubber domes themselves are formed into a sheet that completely protects the contacts below from dirt, dust, and even minor spills. This type of switch design is the simplest, and it uses the fewest parts. This made the rubber dome keyswitch very reliable for several [years](http://flylib.com/books/en/4.308.1.6/1/) . However, its relatively poor tactile feedback has led most keyboard manufacturers to switch to the membrane switch design covered in the next section.

**Membrane Switches**

The membrane keyswitch is a variation on the rubber dome type, using a flat, flexible circuit board to receive input and transmit it to the keyboard microcontroller. Industrial versions of membrane boards use a single sheet for keys that sits on the rubber dome sheet for protection against harsh environments. This arrangement severely limits key travel. For this reason, flat-surface membrane keyboards are not [considered](http://flylib.com/books/en/4.448.1.26/1/) usable for normal touch typing. However, they are ideal for use in extremely harsh environments. Because the sheets can be bonded together and sealed from the elements, membrane keyboards can be used in situations in which no other type could survive. Many industrial applications use membrane keyboards for terminals that do not require extensive data entry but are used instead to operate equipment, such as cash registers and point-of-sale terminals in restaurants .

Membrane keyswitches are no longer relegated to fast food or industrial uses, though. Over the last few years, the membrane keyswitch used with conventional keyboard keytops has [replaced](http://flylib.com/books/en/2.885.1.10/1/) the rubber dome keyswitch to become the most popular keyswitch used in low-cost to mid-range keyboards. Inexpensive to make, membrane switches have become the overwhelming favorite of low-cost Pacific Rim OEM suppliers and are found in most of the keyboards you'll see at your local computer store or find inside the box of your next complete PC. Although low-end membrane keyswitches have a limited life of only 510 million keystrokes, some of the better models are rated to handle up to 20 million keystrokes, putting them in the range of pure mechanical switches for durability (see Figure 16.5). A few membrane switches are even more durable: Cherry Corporation's G8x-series keyboards use Cherry's own 50-million-keystroke membrane switch design (www.cherrycorp.com).

**Figure 16.5. A typical membrane keyswitch used in NMB keyboards.**



Membrane keyboards provide a firmer touch than rubber dome keyboards or the old foam-element keyboards, but they are still no match for mechanical or capacitive keyswitch models in their feel. One interesting exception is the line of keyboards made by KeyTronicEMS using its center-bearing version of membrane keyswitches. Most of its keyboards feature Ergo Technology, which has five levels of force from 35 grams to 80 grams, depending on the relative strength of the fingers used to type various keys. As little as 35 grams of force is required for keys that are used by the little finger, such as Q, Z, and A, and greater levels of force are required for keys used by the other fingers. The spacebar requires the most force: 80 grams. This compares to the standard force level of 55 grams for all keys on normal keyboards (see Figure 16.6). For more information about keyboards with Ergo Technology, visit the KeyTronicEMS website (www.keytronic.com).

**Figure 16.6. Force levels used on KeyTronicEMS keyboards with Ergo Technology.**

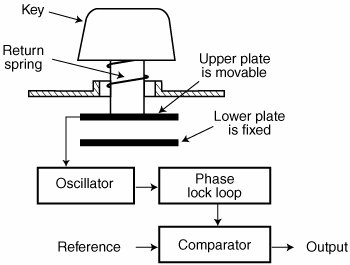


To find the best membrane keyboards from the vast numbers on the market, look at the lifespan rating of the keyswitches. Longer- [lasting](http://flylib.com/books/en/4.492.1.223/1/) keyswitches make the keyboard cost more but will lead to a better experience over the life of the keyboard.

**Capacitive Switches**

Capacitive switches are technically the only truly nonmechanical keyswitches in use today (see Figure 16.7). Although the movement of the key and spring is mechanical in nature, these components do not close a mechanical contact or switch. Capacitive switches are much more expensive than the more common rubber dome mechanical membrane switch, but it is more resistant to dirt and corrosion and offers the highest-quality tactile feedback of any type of switch. Consequently, these are the proverbial Cadillac of keyswitches. This type of keyboard is sometimes referred to as a buckling spring keyboard because of the coiled spring and rocker used to provide feedback.

**Figure 16.7. A capacitive buckling spring keyswitch.**



A capacitive switch does not work by making contact between conductors. Instead, two plates usually made of plastic are connected in a switch matrix designed to detect changes in the capacitance of the circuit.

When the key is pressed, the plunger moves the top plate in relation to the fixed bottom plate. Typically, a buckling spring mechanism provides for a distinct over-center tactile feedback with a resounding "click." As the top plate moves, the capacitance between the two plates changes. The comparator circuitry in the keyboard detects this change.

Because this type of switch does not rely on metal contacts, it is nearly immune to corrosion and dirt. These switches are also very resistant to the key bounce problems that result in multiple characters appearing from a single strike. In addition, they are the most durable in the industryrated for 25 million or more keystrokes, as opposed to 1020 million for other designs. The tactile feedback is unsurpassed because the switch provides a relatively loud click and a strong over-center feel. The only drawback to the design is the cost. Capacitive switch keyboards are among the most expensive designs. The quality of the feel and their durability make them worth the price, however.

Originally, the only vendor of capacitive keyswitch keyboards was IBM. Although some of IBM's older keyboards still feature capacitive keyswitches, most current IBM keyboards use rubber-dome or other lower-cost keyswitches. In 1991, IBM spun off its keyboard/printer division as Lexmark, which then spun off the keyboard division as Unicomp in 1996. Today, Unicomp still manufactures and sells "IBM" keyboards with the classic buckling spring capacitive switch ("clickety" as some would say) technology. As a bonus, it also has models with the IBM [trackpoint](http://flylib.com/books/en/4.464.1.157/1/) built in. You can purchase new Unicomp (IBM) keyboards direct by calling its toll-free number (800-777-4886) or by visiting its online store (http://www.pckeyboard.com).

My personal recommendations are for either the EnduraPro/104 (http://www.pckeyboard.com/ep104.html) or the Customizer 101 or 104 (http://www.pckeyboard.com/customizer.html). These are brand-new , not reconditioned or rebuilt, keyboards.

The EnduraPro/104 is notable for including a built-in TrackPoint pointing device and a pass-through mini-DIN mouse port, being programmable and reconfigurable, requiring no special drivers, and of course having the famous buckling spring keyswitches.

Because of the buckling spring capacitive keyswitches (and the resulting clickety feel), I've always been a huge fan of the IBM, Lexmark, and now Unicomp keyboards. In my opinion, they are the absolute best keyboards in the world and the only ones I willingly use on desktop systems. I especially like the fact that they include the IBM TrackPoint because I use a laptop system as my main machine and therefore use only laptops that include the TrackPoint device ( mainly IBM, Toshiba, and some Dell/HP/others). The feel and durability of the buckling spring capacitive keyswitches is outstanding, and with the integrated TrackPoint, I never have to move my hands off the keyboard, resulting in much greater efficiency when working with my systems.

**The Keyboard Interface**

A keyboard consists of a set of switches mounted in a grid or an array called the key matrix . When a switch is pressed, a processor in the keyboard identifies which key is pressed by determining which grid location in the matrix shows continuity. The keyboard processor, which also interprets how long the key is pressed, can even handle multiple keypresses at the same time. A 16-byte hardware buffer in the keyboard can handle rapid or multiple keypresses, passing each one to the system in succession.

When you press a key, the contact bounces slightly in most cases, meaning that several rapid on/off cycles occur just as the switch makes contact. This is called bounce . The processor in the keyboard is designed to filter this, or debounce the keystroke. The keyboard processor must distinguish bounce from a double key strike the keyboard operator intends to make. This is [fairly](http://flylib.com/books/en/2.480.1.37/1/) easy, though, because the bouncing is much more rapid than a person could simulate by striking a key quickly several times.

The keyboard in a PC is actually a computer itself. It communicates with the main system in one of two ways:

* Through a special serial data link if a standard PS/2 keyboard connector is used
* Through the USB port

The serial data link used by conventional keyboards transmits and receives data in 11-bit packets of information, consisting of 8 data bits, plus framing and control bits. Although it is indeed a serial link (in that the data flows on one wire), the keyboard interface is incompatible with the standard RS-232 serial port commonly used to connect modems.

The processor in the original PC keyboard was an Intel 8048 microcontroller chip. Newer keyboards often use an 8049 version that has built-in ROM or other microcontroller chips compatible with the 8048 or 8049. For example, in its Enhanced keyboards, IBM has always used a custom version of the Motorola 6805 processor, which is compatible with the Intel chips. The keyboard's built-in processor reads the key matrix, debounces the keypress signals, converts the keypress to the appropriate scan code, and transmits the code to the motherboard. The processors built into the keyboard contain their own RAM, possibly some ROM, and a built-in serial interface.

In the original PC/XT design, the keyboard serial interface is connected to an 8255 Programmable Peripheral Interface (PPI) chip on the motherboard of the PC/XT. This chip is connected to the interrupt controller IRQ1 line, which is used to signal to the system that keyboard data is available. The data is then sent from the 8255 to the processor via I/O port address 60h. The IRQ1 signal causes the main system processor to run a subroutine (INT 9h) that interprets the keyboard scan code data and decides what to do.

In an AT-type keyboard design, the keyboard serial interface is connected to a special keyboard controller on the motherboard. This controller was an Intel 8042 Universal Peripheral Interface (UPI) slave microcontroller chip in the original AT design. This microcontroller is essentially another processor that has its own 2KB of ROM and 128 bytes of RAM. An 8742 version that uses erasable programmable read-only memory (EPROM) can be erased and reprogrammed. In the past, when you purchased a motherboard ROM upgrade for an older system from a motherboard manufacturer, the upgrade included a new keyboard controller chip as well because it had somewhat dependent and updated ROM code in it. Some older systems might use the 8041 or 8741 chips, which differ only in the amount of built-in ROM or RAM. However, recent systems incorporate the keyboard controller into the main system chipset.

In an AT system, the (8048-type) microcontroller in the keyboard sends data to the (8042-type) keyboard controller on the motherboard. The motherboard-based controller also can send data back to the keyboard. When the keyboard controller on the motherboard receives data from the keyboard, it signals the motherboard with an IRQ1 and sends the data to the main motherboard processor via I/O port address 60h, just as in the PC/XT. Acting as an agent between the keyboard and the main system processor, the 8042-type keyboard controller can translate scan codes and perform several other functions as well. The system also can send data to the 8042 keyboard controller via port 60h, which then [passes](http://flylib.com/books/en/2.652.1.105/1/) it on to the keyboard. Additionally, when the system needs to send commands to or read the status of the keyboard controller on the motherboard, it reads or writes through I/O port 64h. These commands usually are followed by data sent back and forth via port 60h.

In older systems, the 8042 keyboard controller is also used by the system to control the A20 memory address line, which provides access to system memory greater than 1MB. More modern motherboards typically incorporate this functionality directly into the motherboard chipset. The AT keyboard connector was renamed the "PS/2" port after the IBM PS/2 family of systems debuted in 1987. That was the time when the connector changed in size from the DIN to the min-DIN, and even though the signals were the same, the mini-DIN version became known from that time forward as the PS/2 port.

Keyboards connected to a USB port work in a surprisingly similar fashion to those connected to conventional DIN or mini-DIN (PS/2) ports after the data reaches the system. Inside the keyboard a variety of custom controller chips is used by various keyboard manufacturers to receive and interpret keyboard data before sending it to the system via the USB port. Some of these chips contain USB hub logic to enable the keyboard to act as a USB hub. After the keyboard data reaches the USB port on the system, the USB port routes the data to the 8042-compatible keyboard controller, where the data is treated as any other keyboard information.

This process works very well after a system has [booted](http://flylib.com/books/en/2.717.1.119/1/) into Windows. But what about users who need to use the keyboard at a command prompt or within the BIOS configuration routine? As discussed earlier in this chapter, USB Legacy support must be enabled in the BIOS. A BIOS with USB Legacy support is capable of performing the following tasks :

* Configure the host controller
* Enable a USB keyboard and mouse
* Set up the host controller scheduler
* Route USB keyboard and mouse input to the 8042 Keyboard Controller

Systems with USB Legacy support enabled use the BIOS to control the USB keyboard until a supported operating system is loaded. At that point, the USB host controller driver in the operating system takes control of the keyboard by sending a command called StopBIOS to the BIOS routine that was managing the keyboard. When Windows shuts down to MS-DOS, the USB host controller sends a command called StartBIOS to restart the BIOS routine that manages the keyboard.

When the BIOS controls the keyboard, after the signals reach the 8042 Keyboard Controller, the USB keyboard is treated just like a conventional keyboard if the BIOS is correctly designed to work with USB keyboards. As discussed previously in this chapter, a BIOS upgrade might be necessary in some cases to provide proper support of USB keyboards on some systems. The system chipset also must support USB Legacy features.

**Typematic Functions**

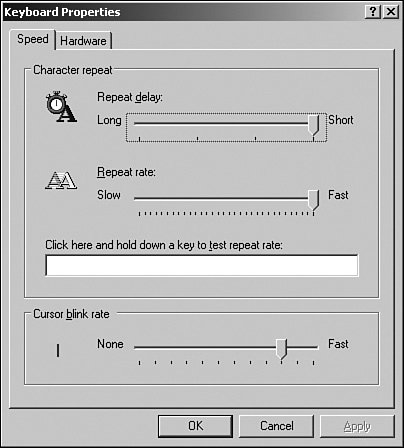
If a key on the keyboard is held down, it becomes typematic , which means the keyboard repeatedly sends the keypress code to the motherboard. In the AT-style keyboards, the typematic rate is adjusted by sending the appropriate commands to the keyboard processor. This is impossible for the earlier PC/XT keyboard types because the keyboard interface for these types is not bidirectional.

AT-style keyboards have programmable typematic repeat rate and delay parameters. You can adjust the typematic repeat rate and delay parameters with settings in your system BIOS (although not all BIOS chips can control all functions) or in your operating system. In Windows you use the Keyboard icon in the Control Panel; in DOS you use the MODE command. The next section describes how to adjust the keyboard parameters in Windows because this is more [convenient](http://flylib.com/books/en/4.288.1.65/1/) than the other methods and enables the user to make further adjustments at any time without restarting the system.

**Adjusting Keyboard Parameters in Windows**

You can modify the default values for the typematic repeat rate and delay parameters in any version of Windows using the Keyboard icon in the Control Panel. The Repeat Delay slider, shown in Figure 16.8, controls the duration for which a key must be pressed before the character begins to repeat, and the Repeat Rate slider controls how fast the character repeats after the delay has elapsed.

**Figure 16.8. Setting the keyboard repeat delay and repeat rate in Windows.**



Note

The increments on the Repeat Delay and Repeat Rate sliders in Keyboard Properties in the Control Panel correspond to the timings given for the MODE command's RATE and DELAY values. Each mark in the Repeat Delay slider adds about 0.25 seconds to the delay, and the marks in the Repeat Rate slider are worth about one character per second each.

The dialog box also contains a text box you can use to test the settings you have chosen before committing them to your system. When you click in the box and press a key, the keyboard reacts using the settings currently specified by the sliders, even if you have not yet applied the changes to the Windows environment.

To learn how to adjust keyboard parameters in DOS, see "Adjusting Keyboard Parameters in DOS" in Chapter 17 of Upgrading and Repairing PCs, 11th Edition , which is available in electronic form on the disc included with this book.

**Keyboard Key Numbers and Scan Codes**

When you press a key on the keyboard, the processor built into the keyboard (8048- or 6805-type) reads the keyswitch location in the keyboard matrix. The processor then sends to the motherboard a serial packet of data containing the scan code for the key that was pressed.

This is called the Make code . When the key is released, a corresponding Break code is sent, indicating to the motherboard that the key has been released. The Break code is equivalent to the Make scan code plus 80h. For example, if the Make scan code for the "A" key is 1Eh, the Break code would be 9Eh. By using both Make and Break scan codes, the system can determine whether a particular key has been held down and determine whether multiple keys are being pressed.

In motherboards that use an 8042-type keyboard controller, the 8042 chip [translates](http://flylib.com/books/en/4.356.1.399/1/) the actual keyboard scan codes into one of up to three sets of system scan codes, which are sent to the main processor. It can be useful in some cases to know what these scan codes are, especially when trouble-shooting keyboard problems or when reading the keyboard or system scan codes directly in software.

When a keyswitch on the keyboard sticks or otherwise fails, the Make scan code of the failed keyswitch usually is reported by diagnostics software, including the power on self test (POST), as well as conventional disk-based diagnostics. This means you must identify the malfunctioning key by its scan code. See the Technical Reference section of the disc included with this book for a comprehensive listing of keyboard key numbers and scan codes for industry-standard 101/102-key (Enhanced) and 104-key Windows keyboards. By looking up the reported scan code on these [charts](http://flylib.com/books/en/4.174.1.44/1/) , you can determine which keyswitch is defective or needs to be cleaned.

Note

The 101-key Enhanced keyboards are capable of three scan code sets. Set 1 is the default. Some systems, including some of the IBM PS/2 machines, use one of the other scan code sets during the POST. For example, my IBM P75 uses Scan Code Set 2 during the POST but switches to Set 1 during normal operation. This is rare, and it really threw me off in diagnosing a stuck key problem one time. It is useful to know whether you are having difficulty interpreting the scan code number, however.

IBM also assigns each key a unique key number to distinguish it from the others. This is important when you are trying to identify keys on foreign keyboards that might use symbols or characters different from what the U.S. models do. In the Enhanced keyboard, most foreign models are missing one of the keys (key 29) found on the U.S. version and have two additional keys (keys 42 and 45). This accounts for the 102-key total instead of the 101 keys found on the U.S. version.

Note

See the Technical Reference section of the disc included with this book for a comprehensive listing of keyboard key numbers and scan codes for both the 101/102-key (Enhanced) keyboard and 104-key Windows keyboard, including HID and hotkey scan codes used on the latest USB and hotkey keyboards.

Knowing these key number figures and scan codes can be useful when you are troubleshooting stuck or failed keys on a keyboard. Diagnostics can report the defective keyswitch by the scan code, which varies from keyboard to keyboard on the character it represents and its location.

Many enhanced and USB keyboards now feature hotkeys that either have fixed usessuch as opening the default web browser, sending the system into standby mode, and adjusting the speaker volumeor are programmable for user-defined functions. Each of these keys also has scan codes. USB keyboards use a special series of codes called Human Interface Device (HID), which are translated into PS/2 scan codes.

**International Keyboard Layouts**

After the keyboard controller in the system receives the scan codes generated by the keyboard and passes them to the main processor, the operating system converts the codes into the appropriate alphanumeric characters. In the United States, these characters are the letters , numbers, and symbols found on the standard American keyboard.

However, no matter which characters you see on the keytops, adjusting the scan code conversion process to map different characters to the keys is relatively simple. Windows (post 3.x) takes advantage of this capability by enabling you to install multiple keyboard layouts to support various languages.

In Windows 9x/Me, [open](http://flylib.com/books/en/2.25.1.168/1/) the Keyboard icon in the Control Panel and select the Language page. The Language box should display the keyboard layout you selected when you installed the operating system. In Windows XP, click the Details button found on the Languages tab in the Regional and Language Options applet (in the Windows Control Panel). By clicking the Add button, you can select any one of several additional keyboard layouts supporting other languages.

These keyboard layouts map various characters to certain keys on the standard keyboard. The standard French layout provides easy access to the accented characters commonly used in that language. For example, pressing the 2 key produces the character. To type the numeral 2, you press the Shift+2 key combination. Other French-speaking countries have different keyboard conventions for the same characters, so Windows includes support for several keyboard layout variations for some languages, based on nationality .

Note

It is important to understand that this feature is not the same as installing the operating system in a different language. These keyboard layouts do not modify the text already displayed onscreen; they only alter the characters generated when you press certain keys.

The alternative keyboard layouts also do not provide support for non-Roman alphabets, such as Russian and Chinese. The accented characters and other symbols used in languages such as French and German are part of the standard ASCII character set. They are always accessible to English-language users through the Windows Character Map utility or through the use of Alt+keypad combinations. An alternative keyboard layout simply gives you an easier way to access the characters used in certain languages.

If you work on documents using more than one language, you can install as many keyboard layouts as necessary and switch between them at will. When you click the Enable Indicator on Taskbar check box on the Language page of the Keyboard control panel, a selector appears in the taskbar's tray area that enables you to switch languages easily. On the same page, you can enable a key combination that switches between the installed keyboard layouts.

**Keyboard/Mouse Interface Connectors**

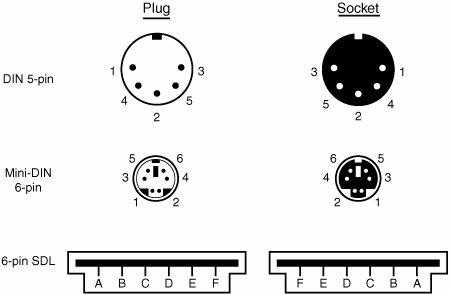
Keyboards typically have a cable with one of three primary types of connectors at the system end. On most aftermarket keyboards, the cable is connected inside the keyboard case on the keyboard end, requiring you to open the keyboard case to disconnect or test it; different vendors use different connections, making cable interchange between brands of keyboards [unlikely](http://flylib.com/books/en/1.124.1.34/1/) . When IBM manufactured its own enhanced keyboards, it used a unique cable assembly that plugged into both the keyboard and the system unit to make cable replacement or interchange easy. Current IBM keyboards, unfortunately, no longer use either the shielded data link (SDL) connector inside the keyboard or the telephone cable-style removable plug-in external keyboard connector used on some more recent models.

Although the method of connecting the keyboard cable to the keyboard can vary, all PC keyboards use one of the following three connectors to attach to the computer:

* 5-pin DIN connector. Used on most obsolete PC systems with Baby-AT form factor motherboards.
* 6-pin mini-DIN connector. Often called a PS/2 connector because it was first used on IBM PS/2 systems, this is the most popular dedicated keyboard and mouse connector.
* USB connector. Many newer systems use USB keyboards and mice.

Figure 16.9 and Table 16.2 show the physical layout and pinouts of the respective keyboard connector plugs and sockets (except USB); although the 6-pin SDL connector is not used in this form by most keyboard vendors, most non-IBM keyboards use a somewhat similar connector to attach the keyboard cable to the inside of the keyboard. You can use the pinouts listed in Table 16.2 to test the continuity of each wire in the keyboard connector.

**Figure 16.9. Keyboard and mouse connectors.**



**Table 16.2. Keyboard Connector Signals and Specifications**

| **Signal Name** | **5-Pin DIN** | **6-Pin Mini-DIN** | **6-Pin SDL** | **Test Voltage** |
| --- | --- | --- | --- | --- |
| Keyboard Data | 2 | 1 | B | +4.8V to +5.5V |
| Ground | 4 | 3 | C |  |
| +5V Power | 5 | 4 | E | +2.0V to +5.5V |
| Keyboard Clock | 1 | 5 | D | +2.0V to +5.5V |
| Not Connected |  | 2 | A |  |
| Not Connected |  | 6 | F |  |
| Not Connected | 3 |  |  |  |
| DIN = Deutsches Institut fr Normung e.V. , a committee that sets German dimensional standards. | | | | |
| SDL = Shielded data link, a type of shielded connector created by AMP and used by IBM and others for keyboard cables. | | | | |
| The 6-pin mini-DIN is usually called a PS/2 connector. | | | | |

PS/2 mouse devices also use the 6-pin mini-DIN connector and have the same pinout and signal descriptions as the keyboard connector; however, the data packets are incompatible. Therefore, you can easily plug a motherboard mouse (PS/2-style) into a mini-DIN keyboard connector or plug the mini-DIN keyboard connector into a motherboard mouse port. Neither one will work properly in this situation, though.

Caution

I have also seen PCs with external power supplies that used the same standard DIN connectors to attach the keyboard and power supply. Although cross-connecting the mini-DIN connectors of a mouse and keyboard is a harmless annoyance, connecting a power supply to a keyboard socket can be disastrous.

USB keyboards use the Series A USB connector to attach to the USB port built into modern computers. For more information on USB, refer to Chapter 15, "I/O Interfaces from Serial and Parallel to IEEE 1394 and USB."

**USB Keyboards**

Most keyboards now on the market can connect to the PC via a USB port instead of the standard PS/2 keyboard port. Because USB is a universal bus that uses a hub to enable multiple devices to connect to a single port, a single USB port in a system can replace the standard serial and parallel ports as well as the keyboard and mouse ports. Most current systems and motherboards still include the standard ports (now called legacy ports) as well as USB, but most so-called legacy-free systems and replacement motherboards have only USB ports for interfacing external devices.

Most keyboard manufacturers now market USB keyboards, but if you want to use your keyboard with both legacy (PS/2) and legacy-free (USB) systems, the most economic way to do so is to specify a keyboard that includes both a USB connector and an adapter to permit the keyboard to work with PS/2 ports. Although Microsoft's Natural Keyboard Elite was the first widely available model to offer USB and PS/2 compatibility, other wired and wireless models from Microsoft, Logitech, Belkin, and others now offer this feature. You can also purchase third-party USB-to-PS/2 adapters, but these can be expensive and might not work with all keyboards.

Not all systems accept USB keyboards, even those with USB ports, because the standard PC BIOS has a keyboard driver that expects a standard keyboard port interface to be present. When a USB keyboard is installed on a system that lacks USB keyboard support, the system can't use it because no driver exists in the BIOS to make it work. In fact, some systems see the lack of a standard keyboard as an error and halt the boot process until one is installed.

To use a keyboard connected via the USB port, you must meet three requirements:

* Have a USB port in the system
* Run Microsoft Windows 98, Windows Me, Windows 2000, or Windows XP (all of which include USB keyboard drivers)
* Have a system chipset and BIOS that feature USB Legacy support

USB Legacy support means your motherboard has a chipset and ROM BIOS drivers that enable a USB keyboard to be used outside the Windows GUI environment. When a system has USB Legacy support enabled, a USB keyboard can be used with MS-DOS (for configuring the system BIOS) when using a command prompt within Windows or when installing Windows on the system for the first time. If USB Legacy support is not enabled on the system, a USB keyboard will function only when Windows is running.

Most recent systems include USB Legacy support, although it might be disabled by default in the system BIOS.

Also, if the Windows installation fails and requires manipulation outside of Windows, the USB keyboard will not function unless it is supported by the chipset and the BIOS. Almost all 1998 and newer systems with USB ports include a chipset and BIOS with USB Legacy (meaning USB Keyboard) support.

Even though USB Legacy support enables you to use a USB keyboard in almost all situations, don't scrap your standard-port keyboards just yet. Some Windows- related [bugs](http://flylib.com/books/en/4.223.1.156/1/) and glitches reported by users include the following:

* Can't log on to Windows the first time after installing a USB keyboard. The solution in some cases is to click Cancel when you are asked to log on and then allow the system to detect the keyboard and install drivers. The logon should work normally thereafter. In other cases, you might have to leave the keyboard unplugged when first booting and then plug it in after the OS desktop is up and running. This allows the keyboard to be detected and drivers loaded.
* Some USB keyboards won't work when the Windows Emergency Boot Disk (EBD) is used to start the system. The solution is to turn off the system, connect a standard keyboard, and restart the system.
* Some users of Windows 98 and Windows 98 SE have reported conflicts between Windows and the BIOS when USB Legacy support is enabled on some systems. This conflict can result in an incapability to detect the USB keyboard if you use the Windows 9x shutdown menu and choose to restart the computer in MS-DOS mode. Check with the system or BIOS vendor for an updated BIOS or a patch to solve this conflict.

If you have problems with Legacy USB support, look at these possible solutions:

* Microsoft's Knowledge Base might address your specific combination of hardware.
* Your keyboard vendor might offer new drivers.
* Your system or motherboard vendor might have a BIOS upgrade you can install.
* Connect the keyboard to the PS/2 port with its adapter (or use a PS/2 keyboard) until you resolve the problem.

|  |  |
| --- | --- |
| http://flylib.com/books/4/57/1/html/2/images/rarr.jpg | See "Universal Serial Bus," p. 980. |

**Keyboards with Special Features**

Several keyboards on the market have special features not found in standard designs. These additional features range from simple things, such as built-in calculators , clocks, and volume control, to more complicated features, such as integrated pointing devices, special character layouts, [shapes](http://flylib.com/books/en/4.145.1.110/1/) , and even programmable keys.

Note

In 1936, August Dvorak patented a simplified character layout called the Dvorak Simplified Keyboard (DSK). The Dvorak keyboard was designed to replace the common QWERTY layout used on nearly all keyboards available today. The Dvorak keyboard was approved as an ANSI standard in 1982 but has seen limited use. For a comparison between the Dvorak keyboard and the common QWERTY keyboard you most likely use, see "The Dvorak Keyboard" in the Technical Reference section of the disc accompanying this book.

**Ergonomic Keyboards**

A trend that began in the late 1990s is to change the shape of the keyboard instead of altering the character layout. This trend has resulted in several so-called ergonomic designs. The goal is to shape the keyboard to better fit the human hand. The most common of these designs splits the keyboard in the center, bending the sides outward. Some designs allow the angle between the sides to be adjusted, such as the now-discontinued Lexmark Select-Ease, the Goldtouch keyboard designed by Mark Goldstein (who also designed the Select-Ease), and the Kinesis Maxim split keyboards. Others, such as the Microsoft Natural keyboard series, PC Concepts Wave, and Cirque Smooth Cat, are fixed. These split or bent designs more easily conform to the hands' natural angles while typing than the standard keyboard. They can improve productivity and typing speed and help prevent repetitive strain injuries (RSI), such as carpal tunnel syndrome (tendon inflammation ). Even more [radical](http://flylib.com/books/en/2.669.1.104/1/) keyboard designs are available from some vendors, including models such as the 3-part Comfort and ErgoMagic keyboards, the Kinesis concave contoured keyboard, and others. A good source for highly ergonomic keyboards, pointing devices, and furniture is Ergonomic Resources (www.ergo-2000.com).

Because of their novelty and trendy appeal , some ergonomic keyboards can be considerably more expensive than traditional designs, but for users with medical problems caused or exacerbated by improper positioning of the wrists at the keyboard, they can be an important remedy to a serious problem. General users, however, are highly resistant to change, and these designs have yet to significantly displace the standard keyboard layout. If you don't want to [spend](http://flylib.com/books/en/3.49.1.124/1/) big bucks on the more radical ergonomic keyboards but want to give yourself at least limited protection from RSI, consider keyboards with a built-in wrist rest or add a gel-based wrist rest to your current keyboard. These provide hand support without making you learn a modified or brand-new keyboard layout.

**USB Keyboards with Hubs**

Some of the latest USB keyboards feature a built-in USB hub designed to add two or more USB ports to your system. Even though this sounds like a good idea, keep in mind that a keyboard-based hub won't provide additional power to the USB connectors. Powered hubs work better with a wider variety of devices than unpowered hubs do. I wouldn't choose a particular model based solely on this feature, although if your keyboard has it and your devices work well when plugged into it, that's great. I'd recommend that you use this type of keyboard with your USB mouse or other devices that don't require much power. Bus- powered devices such as scanners and webcams should be connected to a self-powered hub or directly to the USB ports built in to the computer. USB keyboards and mice correspond to the USB 1.1 standard but can also be connected to the faster USB 2.0 ports on the latest systems.

**Multimedia and Web-Enabled Keyboards**

As I discussed earlier in this chapter, many keyboards sold at retail and bundled with systems today feature fixed-purpose or programmable hotkeys that can launch web browsers, run the Microsoft Media Player, adjust the volume on the speakers , change tracks on the CD player, and so forth. You need Windows 98 or later to use these hotkeys; Windows Me, Windows 2000, and Windows XP add additional support for these keyboards.

For the best results, you should download the latest drivers for your keyboard and version of Windows from the keyboard vendor's website.