USB Provides Easy Connectivity with Amateur Devices

Free yourself from interface boxes and tangles of cable with this decades-old solution.

Al Rovner, K7AR

From the late 1980s to the early 1990s, connecting devices to a PC resulted in a cabling mess. There was no commonality among cables or devices. There were Centronics parallel port cables for printing, RS232 breakout boxes and cables for modems, and DIN cables for keyboards and mice. A typical PC user had a drawer full of cables. In the late 1990s, all this changed with the arrival of the Universal Serial Bus (USB).

Starting with Windows 95B and Windows 98, USB connectors became the standard for many devices, including Mac and Linux PCs, resulting in a simpler way to connect peripheral devices with increased speed and performance. Today, many amateur devices include USB connectors, which allow easy interfacing with your computer.

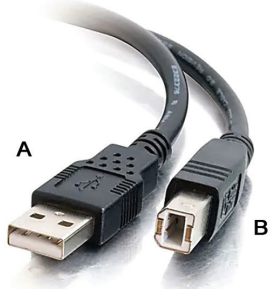

Figure 1 — A USB cable. "A" plugs into your PC, and "B" plugs into your device.

USB Basics

The original USB A-to-B cable is shown in Figure 1. The "A" connector plugs into the computer, while the "B" connector is generally on the rear panel of the device you wish to connect. There are four wires inside of this cable — two for power and two for data. The ground and +5 V wires run straight through while a twisted pair of wires is used to carry data as a differential voltage. The computer USB port typically provides up to 500 mA of current capability. A device that uses more than 500 mA will provide its own power supply. The USB cable is widely used today, and most amateur devices use this cable. Figure 2 shows USB-B ports for a transceiver, amplifier, and rotator control box. Many other amateur devices provide this capability.

USB Hubs and Devices

A typical PC includes two or more USB hubs to expand a single USB port to multiple ports. There is one for front-panel USB connections, one for rear-panel connections, and usually additional USB headers on the PC motherboard. The hub is a smart device that communicates with the operating system. When a device is connected to a hub, the device starts drawing a current that "wakes up" the hub. Let's say I connect an Icom IC-7610 to my PC. A "conversation" takes place that we can imagine goes something like this:

USB Hub Two: Hey, Windows! Someone just connected a device to USB hub two, port three. Go check it out.

Windows: I'm on it.

Windows: Hey, device on hub two, port three, who is your manufacturer?

Figure 2 — The back of an Elecraft KPA1500, an Icom IC-7610, and a Green Heron RT-21 showing the USB-B port.

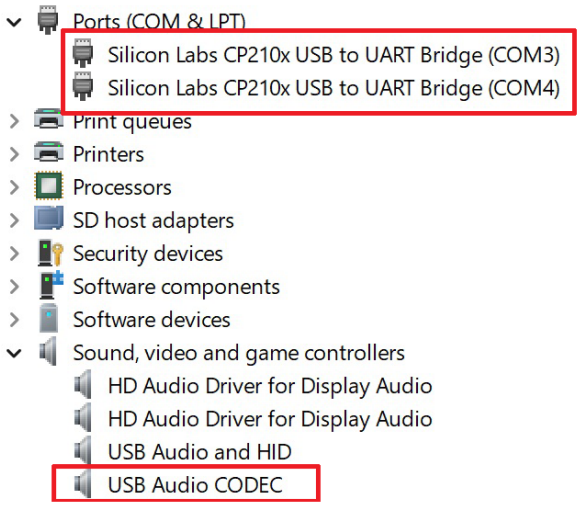

Figure 3 — The PC's Device Manager drop-down menu showing the added devices.

Device: Icom.

Windows: What is your model number?

Device: IC-7610.

What happens next depends on whether the drivers for this device have been installed previously. In this example, Windows will check its list of installed, available drivers for the IC-7610. If a driver is found, Windows will load and activate the driver and include any devices supported by the driver in its Device Manager control panel. Supported devices can be utilized once they appear in the Device Manager control panel.

If no driver is found, Windows will check its online list of available drivers for this device. If no driver is found online, an error message will display, stating the user must manually install drivers for this device. If a driver is found, it will be installed, as discussed earlier.

Continuing with our IC-7610 example, if your computer has never connected to an Icom device using USB, it will need the Icom-specific drivers for the radio to operate properly. Microsoft does not have these on the Windows list of available drivers, so they will not self-install. They are, however, available to download from Icom directly. You'll need to take the additional steps of visiting the Icom website (**https://www.icomamerica.com**), clicking "Support" on the menu bar, and selecting "Firmware Download." Select "Search by Model Name" and enter your Icom model number — in this case, IC-7610. A list of available downloads will be presented, including the USB driver. Select the newest

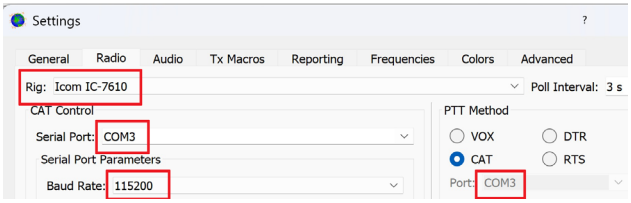

Figure 4 — A sample of what *WSJT-X* should look like when connecting it to the IC-7610.

one (if there is more than one choice), read and agree to the Icom terms and conditions, and press the download button.

The download will appear as a .zip file in your "Downloads" folder. In Windows Explorer, click the file once to highlight it and then press the "Extract All" icon in the top menu bar. It will extract the file into the Downloads directory and create a new folder called "Driver." In that directory, you will find two additional folders that have drivers dependent on your Windows operating system. Click the one you have, and you will see the driver, along with an .exe program to install it. Double-click the correct .exe for your operating system — 32-bit or 64-bit — and it will install the driver.

If you're using a radio by a different manufacturer, you may need to take steps similar to the Icom example to find the specific drivers required.

Verifying Connected Device Status

Now that the IC-7610 is plugged into the PC and an appropriate driver has been installed, Figure 3 shows what new devices are available, highlighted here in red. The IC-7610 uses two Silicon Labs CP210x USB to UART Bridge interface chips. In my case, Windows has assigned these devices to COM3 and COM4. Also, a new sound interface called USB Audio CODEC is provided. Many amateur manufacturers provide similar functionalities and add a sound interface and one or more COM ports similar to the IC-7610 (they may even

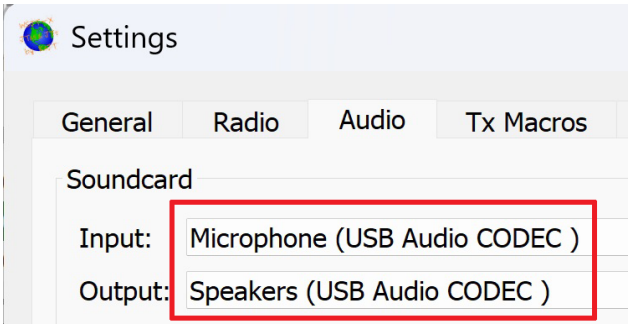

Figure 5 — A sample of what *WSJT-X* should look like when setting up the audio devices.

use the same chips). Having this functionality 20 years ago would have required external interface boxes, audio cables, etc. Now this is all accomplished using one simple USB cable. The two COM ports and the sound interface shown in Figure 3 can be used in any number of amateur software packages. Let's look at *WSJT-X* and *N1MM* as examples.

WSJT-X

Figure 4 shows the **RADIO** tab in the *WSJT-X* Settings page. We can see that the Rig is set to Icom IC-7610, and it's connected to COM3 with a baud rate of 115200. This assumes the IC-7610 already has its COM port set to 115200 baud. We use the computer-aided transceiver (CAT) method to assert the push-to-talk (PTT) line, and the COM3 Serial Port. This setup allows *WSJT-X* to read the frequency from the IC-7610, change the IC-7610 to other bands, and assert the PTT line when *WSJT-X* is ready to transmit.

Figure 5 shows the **AUDIO** tab in the *WSJT-X* Settings page. Here, we can see how the USB Audio CODEC is used. The USB Audio CODEC has two sections, one for Receive Audio and one for pTransmit Audio. While this setup is easy, there are a couple more tasks. The Receive and Transmit Audio levels must be set. To set up the Receive level, consult the *WSJT-X* documentation. To set up the Transmit level, if your rig has an upper sideband-data mode (also referred to as USB-data), use it.

Using the USB-data mode disconnects any front-panel mic audio. Also make sure your mic compressor is turned off. Adjust the Pwr Slider in *WSJT-X*, so your automatic level control meter shows little or no movement when transmitting.

Just a quick note: As you saw, the IC-7610 provides two serial ports (COM3 and COM4). The IC-7610's second serial port is used to run FSK RTTY. COM3 is used as previously described, and COM4 uses MMTTY to run RTTY, mainly in contests. You have frequency control for logging and FSK RTTY capability all through the same USB cable, which makes a very flexible and easy setup!

N1MM

N1MM is set up in a similar manner by using its **HARDWARE CONFIGURE** screen. Figure 6 shows a typical

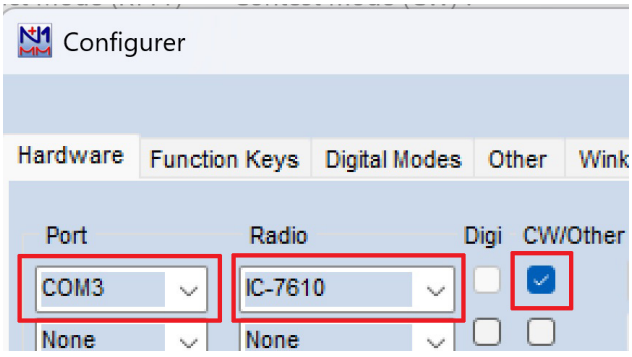

Figure 6 — A sample of what *N1MM* should look like when connecting it to the IC-7610.

setup selecting the COM3 port installed by the driver, the IC-7610 rig, and CW mode. This setup allows *N1MM* to read the frequency from the rig for logging, changing bands, and to perform CW keying. Again, all this is done without needing external keying interfaces or RS232 breakout boxes.

Powering Down

When you power off your device, the USB hub recognizes that a current is no longer drawn from the previously used port. The hub notifies Windows that the device has been physically disconnected or powered off. Then Windows removes the supported devices from its Device Manager control panel and unloads the supporting driver.

As a traveling DXpeditioner, the less I carry, the better it is for weight management, quick setup, and teardown times. USB connectivity has been a boon to amateur radio for 25 years.

All photos provided by the author.

Al Rovner, K7AR, has been a licensed ham radio operator for 53 years, and is an active DXer and contester. He is a retired engineer with a BS in electrical engineering from Drexel University and an MS in computer science from Oregon Health & Science University. Al can be reached at **k7ar@comcast.net**.

For updates to this article, see the *QST* Feedback page at www.arrl.org/feedback.