

Radio Interface/Programming Cable Hints and Tips

Dispelling, Correcting Myths...



Introduction

This document is intended to provide examples and clarification about many common issues when deploying and challenged by a variety of radio interface cables – whether used for radio programming or controlling radios for new modes/features (WinLink, FT8, etc.)

Unless your radio manufacturer and PC still provides the basic reference RS-232C standard I/O connections and chips you're likely to encounter at least two types of interface cables and schemes to get data into and out of your radio equipment.

These would include:

- USB (PC-side) -to- serial I/O (radio-side) (usually there is an adapter chip blob at the USB end of the cable)
- USB (PC-side) -to- serial I/O (device-side) (usually an adapter chip at the device, DB-9,-25 end of the cable)
- USB (PC-side) - to – hybrid/embedded USB (radio-side) cabling configuration (typically “no blob”)

Amid the USB-to-serial/radio interfaces and cables you will likely find an adapter chip from makers such as FTDI, Gigadevice, Prolific or STM. FTDI and Prolific are the most prevalent for USB-to-serial interfaces, with frequently problematic knockoff ‘Prolific’ chips. Gigadevice and STM devices are typically found as in-radio USB terminations for “direct USB cables” (adapter chip blob at either connector.)

Common Problems

Depending on the interface chip and/or ‘driver’ installation procedures at time of software installation, or connecting the USB end of the cable to the PC, or USB-to-PC and connector to radio, users may encounter various “device not found”, “device not recognized”, “no driver” or similar error messages that may or may not ‘suggest’/be perceived as a problem with Windows, a “bad driver”, “bad software”, “bad cable”, unsupported cable/interface – whatever.

In reaction to such errors the user is compelled to reach out to any number of vendor, PC, or hobby venues (social media, groups.io, etc.) seeking a solution, and getting various ‘solutions’, some few correct, many not. With not too much investigation it can be easy, or not, to identify the culprits and perhaps resolve them – through proper procedures or getting a more reliable product.

‘Hive’ Support

Since Microsoft Windows “day one” a lot of actual and sadly made-up, imagined, mis-guided, and simply wrong information has grown about Windows device detection, drivers, management and related software, the Windows Registry, et al.

On-line forums are full of less-than-qualified responses such as:

“just uninstall and re-install driver(s)”

“Windows version ‘x’ changed and won’t allow ‘y’”

“...not Windows 10...11 compatible...”

“AnyTone/Baofeng/Bridgecom/Connect/TYT/Wouxon... released garbage software/provided bogus drivers...”

All ‘challenged’ about the original legacy device functions and settings, progression through the Desktop Management Institute which largely became Plug-and-Play, the introduction of USB, etc. to maintain legacy serial port functions.

Rather than taking the time to walk through simple steps with Windows Device Manager to identify what we can know about the product involved, determine and lookup what Windows (PnP) knows about the device in question, a lot of time can be wasted, perhaps money as well, trying things that can do little to know and correct the real issue(s).

Common, Knowable Issues

#1 – the Windows operating system(s) have not fundamentally, substantially changed how drivers, or driver configuration is implemented – be it Windows 7, 8, 10 or 11 – especially at most common/legacy I/O device levels. Microsoft and hardware vendors have better qualified legit, known device drivers especially if they involve executable code beyond applying and linking to device description and parameters data in the Registry.

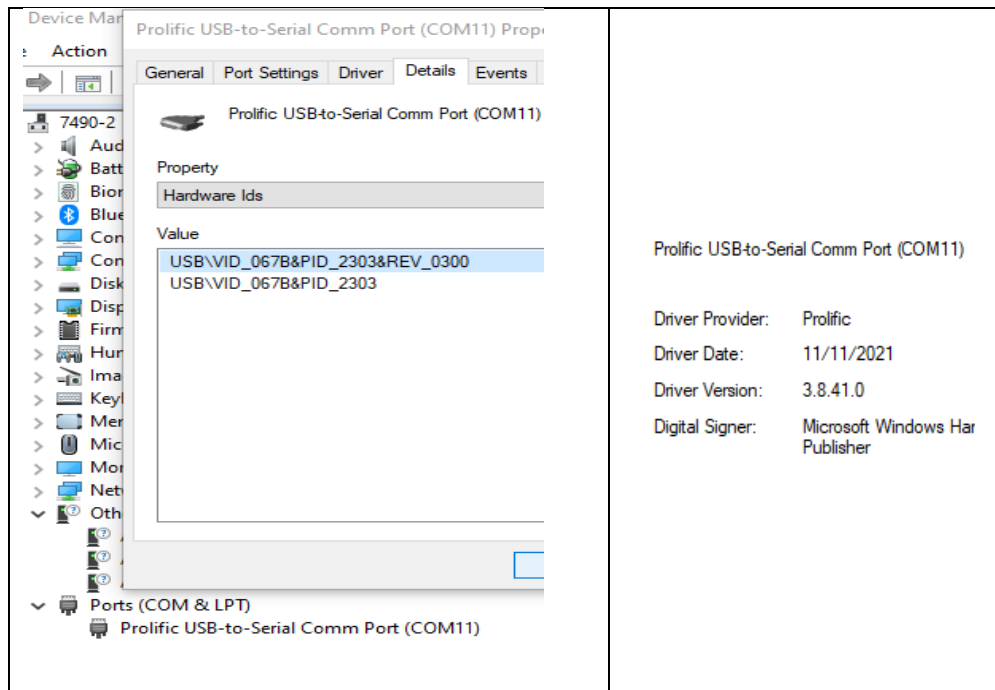
#2 – It’s been known for a ‘few’ years that someone cloned/ripped-off Prolific’s USB-to-serial adapter chips. Prolific reacted by coding their “driver installation” elements to detect the cloned/counterfeit devices and got creative about preventing Windows’ access to the devices, providing instead a clear message presented in Device Manager.

#3 – Even a “shallow dive” into Device Manager reveals that Prolific does NOT provide/install a ‘binary’ or functional driver but merely adds the device ID and specific “how to use me” parameters in the Windows Registry database as instructions for Windows’ native serpl.sys driver. No Prolific ‘driver’ is installed, thus there is no ‘driver’ to remove. In fact Prolific’s “uninstall driver” process does not even clear the Registry of the data it put in there. “Uninstall driver” does nothing, which is why ‘re-installing’ a Prolific ‘driver’ leaves you/Windows with the same bad data and failed results in some cases. To correct this requires looking for the Device ID and the lengthy GUID identifier in the Registry and deleting all such references. A real mess.

#4 – FTDI, Gigadevices and STM do provide their own specific driver ‘code’, the former to be able to provide an uncompromised legitimate RS-232 standard serial I/O port. Giga/STM defer to USB protocols using their specific device capabilities and parameters.

Good Cable with Legit Prolific Chip and 'Driver'

In this example a valid Prolific chip has been detected and cross-referenced in the Prolific-provided driver reference/parameter data, assigned a good Hardware ID and registered the 'driver' parameter information properly.

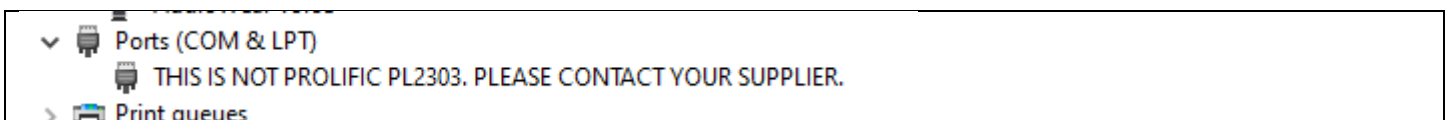


Note Device Details, Hardware ID versus the lack of data, terse warning of a 'bad' cable.

BAD - Prolific clone (generic no-brand, knock-off, counterfeit cables/interfaces)


An example un-known 'Baofeng'-labelled cable (from Amazon), uselessly marketed as Windows 10 compatible, has nothing to do with Windows, it's not a Windows, it's Prolific's device references being aggressive denying use of cloned devices. Prolific tells Windows to do nothing with such devices but display a terse message. Such knock-off chip implementations can appear under any other radio names from myriad vendors. Amazon returns is your friend!!!

There is NO solution for using these clone/counterfeit cables under Windows. Success only with Chirp on MacOS.








FTDI Device

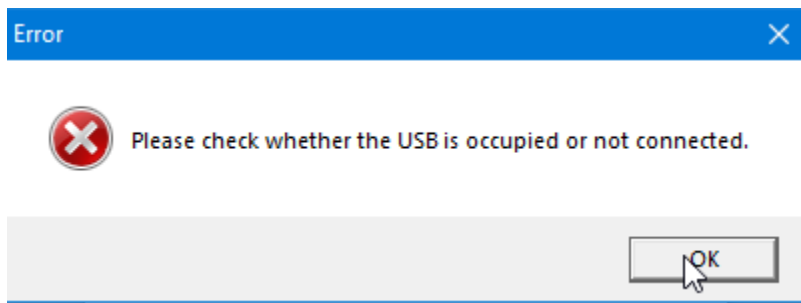
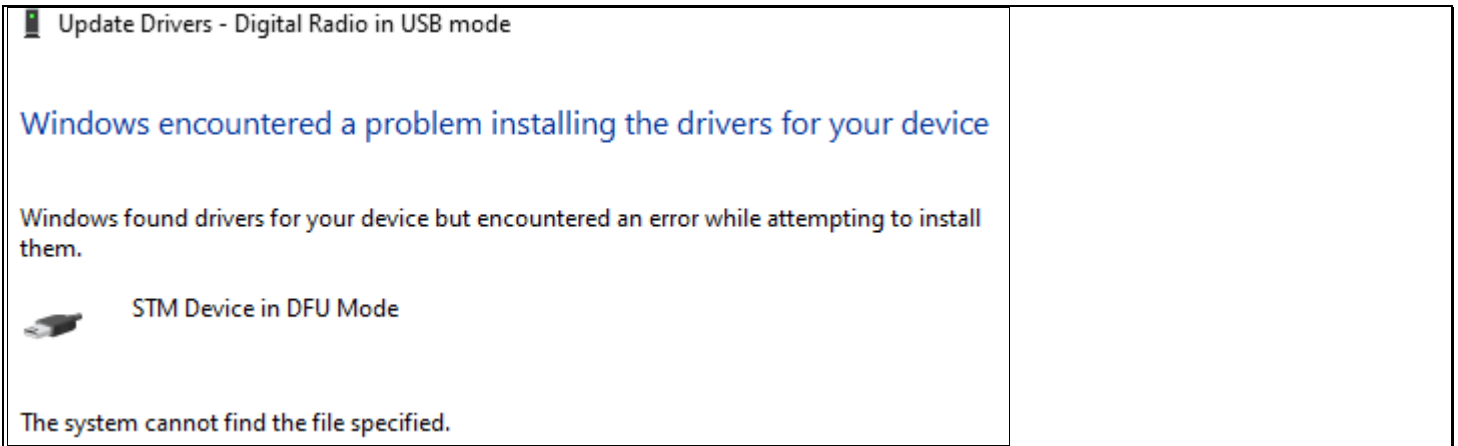
With the exception of the legacy 8250/16450 UAR/T serial port interface chips of early PCs and other devices, and subsequent integrated chipset implementations for genuine RS-232 serial I/O, FTDI chips are probably one of the best at emulating the RS-232 in new chip forms. FTDI provides its own 'binary' drivers rather than just feeding parameters to MS Windows serial driver stack.

 USB Serial Port (COM10)	Driver Provider: FTDI
Device type: Ports (COM & LPT)	Driver Date: 7/5/2021
Manufacturer: FTDI	Driver Version: 2.12.36.4
Location: on USB Serial Converter	Digital Signer: Microsoft Windows Hardware Compatibility Publisher

TYT

TYT employs an STM hybrid USB device in the radio – no in-cable USB-to-serial adapter. In some cases the provided or downloaded drivers may not install properly, but with the radio connected and powered on, and going through to process to manually install the driver (pointing the install driver process to the folder with the extracted .INF, .DAT and/or .SYS files is usually successful.










- ▼  Other devices
 -  ARfcomm
 -  AudioWear
 -  AudioWear Voice
 -  Digital Radio in USB mode



Apparently Windows has a problem with STM's driver info, so follow this with the indicated or later STM drivers:

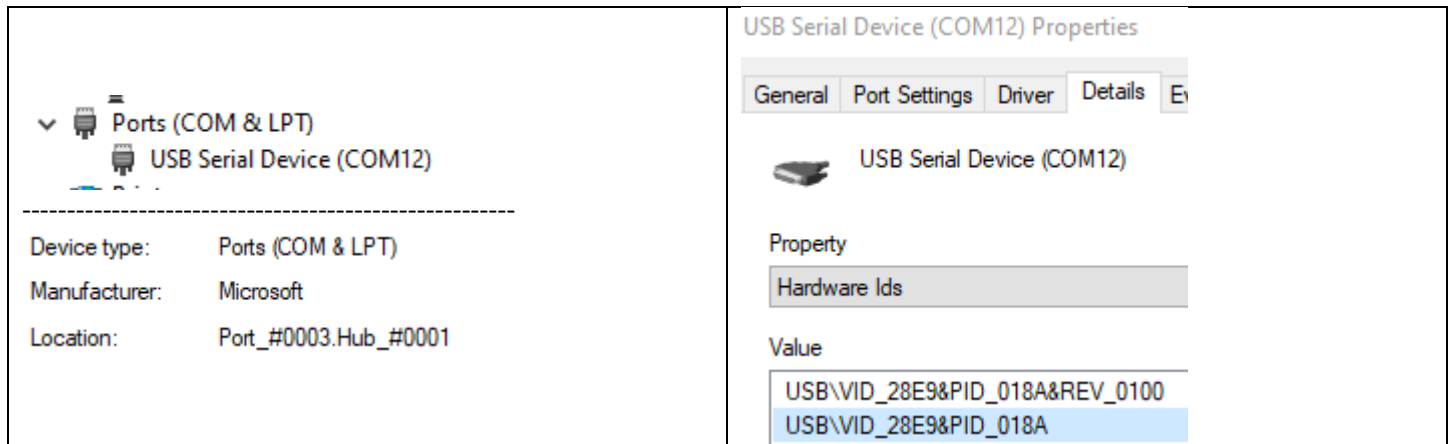
https://www.reddit.com/r/HamRadio/comments/pb89b9/tyt_md380_issues_with_windows_10_drivers/ or STM's website, which requires an email address to gain download access.

When good you should see something like this:

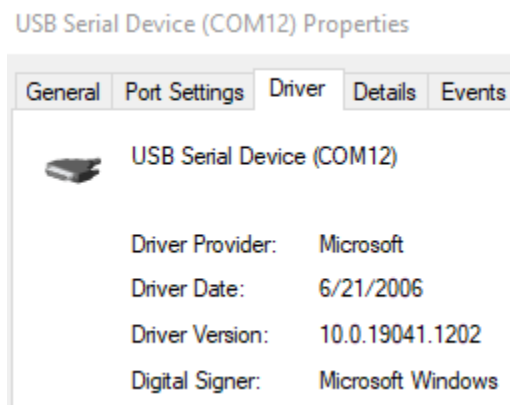
- ▼  Universal Serial Bus controllers
 -  Generic SuperSpeed USB Hc
 -  Generic SuperSpeed USB Hc
 -  Generic USB Hub
 -  Generic USB Hub
 -  Generic USB Hub
 -  Generic USB Hub
 -  Intel(R) USB 3.0 eXtensible I
 -  STM Device in DFU Mode

AnyTone

USB Virtual COM Port built-into radio:



Uses “USB Virtual Com Port Driver_v2.0.2.2673” or similar, by Gigadevice, which may actually be a new/different FTDI chip. The driver should come with the AnyTone package/software. Or try here: <https://ftdichip.com/drivers/vcp-drivers/>. The hardware is not Microsoft, but appears so under the device Manufacturer. It is a Gigadevice/FTDI chip uses the native Microsoft drivers and FTDI/Giga device parameters and ID.



Connect Systems CS700

Appears to setup much the same way the TYT does – STM driver that you have to “point to” in-folder to install from Device Manager. The device/driver appears only when the cable is connected at both ends and the radio is powered on.

‘Generic’ USB to Serial Adapters

(DB-9/DB-25 connectors)

Cables to interface with ‘digital’ mode (SignalLink, etc.) or other devices for computer-based rig control can present either of the conditions above for genuine FTDI or Prolific chips, or the cloned/counterfeit Prolific chips.

Mirkit Multi-Radio USB to Serial Adapter

A relatively 'new' offering is the multi-adapter Mirkit package that offers 6 cable adapters to interface between various radios and the single FTDI-based USB-serial I/O adapter.



<https://www.amazon.com/MIRKIT-Baofeng-Programming-Compatible-Radios/dp/B09LD2HCZV>

