

www.vskom.de

User Manual USB-COM Plus

Edition: July 2017



Tel: +49 40 528 401 0

Fax: +49 40 528 401 99

Web: www.visionsystems.de

Support: service@visionsystems.de

The software described in this manual is furnished under a license agreement and may be used only in accordance with the terms of that agreement.

Copyright Notice

Copyright © 2009-2017 Vision Systems. All rights reserved. Reproduction without permission is prohibited.

Trademarks

VScom is a registered trademark of Vision Systems GmbH. All other trademarks and brands are property of their rightful owners.

Disclaimer

Vision Systems reserves the right to make changes and improvements to its product without providing notice.

Vision Systems provides this document “as is”, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Vision Systems reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time.

Information provided in this manual is intended to be accurate and reliable. However, Vision Systems assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use.

This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

Contents

1. Overview	6
2. Introduction	6
2.1. Features	6
2.2. Product Specifications	7
2.2.1. USB-COM Plus Mini and USB-COM Plus Mini ISO	8
2.2.2. USB-COM Plus and USB-COM Plus ISO	8
2.2.3. USB-2COM Plus and USB-2COM Plus ISO	9
2.2.4. USB-4COM Plus and USB-4COM Plus ISO	10
2.2.5. USB-8COM Plus and USB-8COM Plus ISO	11
2.2.6. USB-16COM Plus	12
2.2.7. USB-COM Plus mPCIe	13
2.3. Packing List	13
2.4. Optional Accessories	14
2.4.1. Mounting Options	14
2.4.1.1. Warning:	14
2.4.2. Power Options	14
2.4.3. Connection Options	14
2.5. About this Manual	14
3. The Drivers for Windows	15
3.1. Install Drivers from Windows Update (Automatic)	15
3.2. Install Drivers by Pre-Installation Executable (Offline)	18
3.2.1. Check Installation of the Drivers	21
3.2.2. Uninstall Drivers	21
3.3. Configure the Serial Ports	22
3.3.1. Rename the Serial Port	23
3.3.2. Enable faster Responses	23
3.3.3. Miscellaneous Options	24
3.3.3.1. Serial Enumerator	24
3.3.3.2. Serial Printer	24
3.3.3.3. Cancel if Power Off	24
3.3.3.4. Event On Surprise Removal	24
3.3.3.5. Set RTS On Close	24
3.3.3.6. Disable Modem Ctrl At Startup	24
3.4. USB-COM Plus Configurator	25
3.5. Optimizing Data Throughput	26
4. Hardware Configuration	26
4.1. Configuration via DIP Switches	26
4.1.1. Models with One and Two serial Ports	26
4.1.2. Models with Four and Eight serial Ports	28
4.2. Configuration via Windows Software	29
4.3. Configuration via Terminal Program	31
4.3.1. Exit the Configuration (0)	32
4.3.2. Show Port Configurations (1)	32
4.3.3. Change Port Configuration (2)	33
4.3.4. Default Port Configuration (3)	34

4.3.5. Save Port Configurations (4)	34
4.3.6. Show Configuration File (5)	34
4.3.7. Upload Configuration File (6)	35
5. Technical Background on RS485	36
5.1. Transmission Technique	36
5.2. Termination	36
5.3. Polarization	37
5.4. 2-Wire Scheme	37
5.5. 4-Wire Scheme	38
5.6. On USB-COM Plus Devices	39
6. Connector Definitions	39
6.1. DB9 male	39
6.2. Serial Connector mPCIe	39
6.3. Terminal Block Adapter	40
6.4. Adapter RJ45	40
A. History	40

List of Figures

1. USB-COM Plus Mini	8
2. USB-COM Plus ISO	9
3. USB-2COM Plus ISO	9
4. USB-4COM Plus ISO	10
5. USB-8COM Plus ISO	11
6. USB-16COM Plus	12
7. USB-COM Plus mPCIe	13
8. Confirm Windows Update	16
9. Automatic Installation	17
10. Extract to temporary Space	18
11. Start Installation	19
12. Confirm License conditions	19
13. Copy Files and Install Drivers	20
14. Installation finished	20
15. Uninstall Drivers	21
16. USB Serial Port Properties	22
17. Advanced port options	23
18. Driver Configurator	25
19. No Device	29
20. One USB-8COM Plus detected	30
21. UsbComCfg Multi-Port Selection	31
22. Open Terminal Configuration	32
23. Terminal Main Menu	32
24. Exit Terminal Configuration	32
25. Show all port configurations	32
26. Change port configurations	33
27. Configure a serial port	34

28.	Saving Configuration file	35
29.	2-wire cabling scheme	37
30.	4-wire cabling scheme	38
31.	DB9 male Connector	39
32.	Terminal Block	40
33.	DB9-male to RJ45	40
34.	RJ45 to DB9-male	40

List of Tables

1.	Features of USB-COM Plus Series	7
2.	Serial Port Operation Modes	7
3.	Features of USB-COM Plus (Single Port)	8
4.	Features of USB-COM Plus (Single Port)	8
5.	Features of USB-2COM Plus (Dual Port)	9
6.	Features of USB-4COM Plus (Quad Port)	10
7.	Features of USB-8COM Plus (Octal Port)	11
8.	Features of USB-16COM Plus (Multi Port)	12
9.	Features of USB-COM Plus mPCIe PCI Express Mini card	13
10.	Configuration USB-COM ⁺ Mini and USB-COM ⁺ mPCIe	27
11.	Configuration USB-COM ⁺ and USB-2COM ⁺	27
12.	Configuration USB-4COM ⁺ and USB-8COM ⁺	28
13.	DB9 male Connector	39
14.	mPCIe Connector	39
15.	DB9 to RJ45	40

1. Overview

The USB-COM Plus are serial port adapters, which are connected over USB port. They are industrial strength devices, providing different types of serial ports. The ports may operate in RS232, RS422 or RS485 mode. Ports are either configured by DIP switches or by software for four and eight serial ports adapters. All devices come in an IP30 metal case, providing option for DIN Rail or 19" rack mounting. All connectors including serial ports, power and USB are strongly ESD protected. Each model has a standard and an galvanically isolated version, called "ISO". For reasons of brevity USB-COM Plus are referred to as USB-COM⁺, as used throughout in this manual.

The supplied drivers for Windows operating systems install the ports into the Windows API, so they appear as the standard Com ports. All applications capable to handle a serial port can control the USB-COM⁺ devices in the same way as they control the genuine ports Com1 and Com2 of a PC. On Linux¹ systems the serial ports are installed as `/dev/ttyUSBx`; similar drivers exist for other operating systems.

The driver allows to configure the serial ports for bitrates of up to 3Mbit/s². In RS232 mode the bitrate is limited to 1Mbit/s. Given the very high maximum speed, the USB-COM⁺ allows generating many different bitrates in lower ranges. In fact every single bitrate from 185 bps up to 921,600 bps (1 Mbps) could be chosen as transmission speed. Above 1 Mbps range many dedicated bitrates can be also used.

The interface to the host PC is USB 2.0, Full Speed or High Speed where required.

2. Introduction

This manual describes the hardware of USB-COM Plus serial port adapters. Also the Windows driver for the USB-COM⁺ is described in detail.

2.1. Features

- USB 2.0 High Speed or Full Speed
- Configuration serial ports modes: by DIP switches and/or by software
- Supports RS232, RS422 or RS485
- Termination resistors for RS422 or RS485 automatically
- UART FIFO buffers: 128 Byte
- Max speed 3.0 Mbit/s (or 12 Mbit/s)
- Drivers for WindowsTM operating system
- Supported by Linux, Mac OS X, and Windows CE on several platforms.

¹Generally with Kernel 3.0.0-19 and above

²12Mbps on four and eight port models

2.2. Product Specifications

All USB-COM⁺ models have similar properties. Most important difference is the number of serial ports. The maximum speed is 3 Mbit/s or 12 Mbit/s, which may be used in RS422 or RS485 mode RS232 is limited to 1 Mbit/s.

Host interface	USB 2.0	Full Speed or High Speed
	Connector	Type B
Serial Ports	1, 2, 4, 8	RS232 / RS422 / RS485
Signals	RS232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, GND
	RS232 isolated	RxD, TxD, RTS, CTS, GND
	RS422 / RS485	Tx±, Rx±, GND
	RS485	Data±
Max. Bitrates	3.0 Mbit/s	RS422 and RS485
	921.6 kbit/s	RS232 (1 Mbit/s)
Serial configurations	Data bits	7, 8
	Parity	None, Odd, Even, Mark, Space
	Stop bits	1, 2
FIFO size	128 Bytes	At least, for Transmit and Receive each
LED	Data	Transmit/Green and Receive/Orange for each port
	Ready	Green
	Power	Red
Protection	All Connectors ESD protected	Compliant with IEC 61000-4-2 ESD 8kV contact / 16kV air discharge

Table 1: Features of USB-COM Plus Series

Each of the serial ports may be used in 4 different operation modes as listed below.

Mode	Comment
RS485 “2w”	Half-Duplex, Transmitter activates by automatic
RS485 “4w”	Full-Duplex, Transmitter activates by automatic
RS422	Full-Duplex, Transmitter permanent active
RS232	

Table 2: Serial Port Operation Modes

RS422 is a Full-Duplex mode by definition, RS485 may be used in Half- or Full-Duplex. Half-Duplex is also known as “2 wire” and denoted as “2w” here, while “4w” specifies the “4 wire” Full-Duplex mode.

For RS485 mode it is important to switch off Tx lines of the device immediately after end-of-transmission, enabling other devices to send data (e.g. slave device answering) asap. We call this Automatic Receive Transmit Control (ARTc). ARTc is implemented for USB-COM⁺ devices in hardware, which results in very efficient communication over RS485 “2w” lines.

For the RS422 and RS485 modes an internal termination resistor of 120Ω is provided and automatically controlled by the chosen operating mode. In RS232 mode the termination resistor is not active. The circuits do not require any BIASing of the data lines.

Specifications differing from this general list are given at each product below.

2.2.1. USB-COM Plus Mini and USB-COM Plus Mini ISO

Serial Ports	1	
Connectors	1×DB9 male	RS232, RS422, RS485 (6.1)
Power	Bus powered	via USB cable
FIFO size	128 Bytes	Transmit
	384 Bytes	Receive
Dimensions	W×L×H	50×73×22 mm ³

Table 3: Features of USB-COM Plus (Single Port)

On USB-COM Plus Mini ISO the serial port signals are galvanically isolated (2.5 kV) from the internal logic including USB.



Figure 1: USB-COM Plus Mini

This is the single port USB-COM Plus Mini. The USB-COM Plus Mini ISO looks the same. The DIP switches are accessible on the bottom side.

2.2.2. USB-COM Plus and USB-COM Plus ISO

Serial Ports	1	
Connectors	1×DB9 male	RS232, RS422, RS485 (6.1)
Power	Bus powered	via USB cable
FIFO size	128 Bytes	Transmit
	384 Bytes	Receive
Dimensions	W×L×H	115×73×25 mm ³

Table 4: Features of USB-COM Plus (Single Port)

On USB-COM Plus ISO the serial port signals are galvanically isolated (2.5 kV) from the internal logic including USB.



Figure 2: USB-COM Plus ISO

This is the single port USB-COM Plus, this one and the serial isolated USB-COM Plus ISO look the same.

2.2.3. USB-2COM Plus and USB-2COM Plus ISO

Serial Ports	2	
Connectors	2×DB9 male	RS232, RS422, RS485 (6.1)
Power	Bus powered	via USB cable
FIFO size	128 Bytes	Transmit
	384 Bytes	Receive
Dimensions	W×L×H	115×73×25 mm ³

Table 5: Features of USB-2COM Plus (Dual Port)

On USB-2COM Plus ISO the serial port signals are galvanically isolated (2.5 kV) from the internal logic including USB and from the other serial port.



Figure 3: USB-2COM Plus ISO

This is the dual port USB-2COM Plus, this one and the serial isolated USB-2COM Plus ISO look the same.

2.2.4. USB-4COM Plus and USB-4COM Plus ISO

Serial Ports	4	
Connectors	4×DB9 male	RS232, RS422, RS485 (6.1)
Max. Bitrates	12 Mbit/s	on RS422 and RS485
Power	Dual Mode	Self-powered, bus powered via USB cable
	external	9-54V DC on terminal block at the rear side
FIFO size	2 kBytes	Transmit and Receive
USB-Out	1×USB A	To connect more USB-COM Plus devices
Dimensions	W×L×H	196×147×44 mm ³

Table 6: Features of USB-4COM Plus (Quad Port)

USB-4COM Plus requests 5V@500mA max., so it can run bus powered only. If another USB-xCOM⁺ model is attached/concatenated to the rear USB Type A an external power supply (9-54V DC) must be connected, so it runs self-powered.

USB-4COM Plus ISO requests 5V@600mA max., so it should be operated in self-powered mode with an external power supply. For USB 2.0 ports with higher current capabilities bus-powered mode is also possible. To support all situations an external power supply should be connected on the rear side.

On USB-4COM Plus ISO the serial port signals are galvanically isolated (2.5 kV) from the internal logic including USB and from other serial ports.



(a) Front Side



(b) Rear Side

Figure 4: USB-4COM Plus ISO

This is the quad port USB-4COM Plus, this one and the serial isolated USB-4COM Plus ISO look the same.

2.2.5. USB-8COM Plus and USB-8COM Plus ISO

Serial Ports	8	
Connectors	8×DB9 male	RS232, RS422, RS485 (6.1)
Max. Bitrates	12 Mbit/s	on RS422 and RS485
Power	Dual Mode	Self-powered, bus powered via USB cable
	external	9-54V DC on terminal block at the rear side
FIFO size	2 kBytes	Transmit and Receive
USB-Out	1×USB A	To connect more USB-COM Plus devices
Dimensions	W×L×H	196×147×44 mm ³

Table 7: Features of USB-8COM Plus (Octal Port)

USB-8COM Plus and USB-8COM Plus ISO models usually operate in self-powered mode, so a wide range external power supply (9-54V DC) must be used. However USB 2.0 ports capable supplying 5V@600mA – like USB 3.0 – could operate USB-8COM⁺ model in bus-powered mode.

On USB-8COM Plus ISO the serial port signals are galvanically isolated (2.5 kV) from the internal logic including USB and from other serial ports.



(a) Front Side



(b) Rear Side

Figure 5: USB-8COM Plus ISO

USB-8COM Plus standard and USB-8COM Plus ISO look in the same way.

2.2.6. USB-16COM Plus

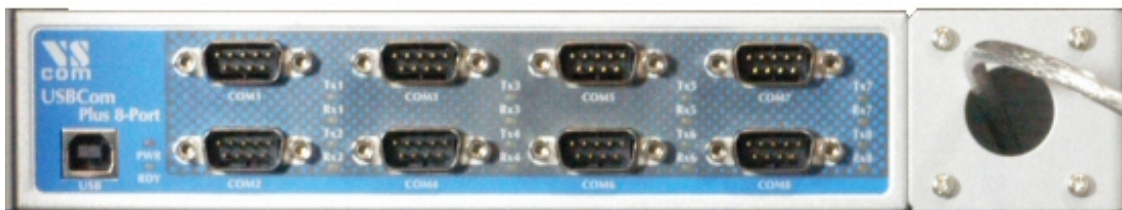
There is a special 19-inch mounting kit for two USB-COM Plus models. It results in a device with 16 serial ports, connected via the USB-Out port. This example uses two USB-8COM Plus.

Serial Ports	16	
Connectors	16×DB9 male	RS232, RS422, RS485 (6.1)
Max. Bitrates	12 Mbit/s	on RS422 and RS485
Power	external	9-54V DC on terminal blocks at the rear side
FIFO size	2 kBytes	Transmit and Receive
USB-Out	1×USB A	To connect more USB-COM Plus devices
Dimensions	W×L×H	437×147×44 mm ³

Table 8: Features of USB-16COM Plus (Multi Port)



(a) 19-Inch mounted complete



(b) Left part in Detail



(c) Details for Right part

Figure 6: USB-16COM Plus

This mounted combination requires only one Height Unit in the 19-inch rack. The sixteen ports are about the maximum possible with DSub-9 in one unit height. To achieve even more serial ports it is possible to connect another 16-port combination to the USB-Out port of the second device. With only one USB cable attached to the PC this results in 32 ports on two units of height.

2.2.7. USB-COM Plus mPCIe

The USB-COM Plus mPCIe is a PCI Express Mini add-on card, for installation into a PC. It is covered in this manual, because the connection on the cards slot is selected as USB 2.0, making the card circuits identical to USB-COM Plus Mini.

Serial Ports	1	
Connectors	1×DB9 male	RS232, RS422, RS485 (6.1)
Power	Bus powered	via Mini PCI Express slot
FIFO size	128 Bytes	Transmit
	384 Bytes	Receive
Dimensions	W×L×H	30×51×3 mm ³

Table 9: Features of USB-COM Plus mPCIe PCI Express Mini card

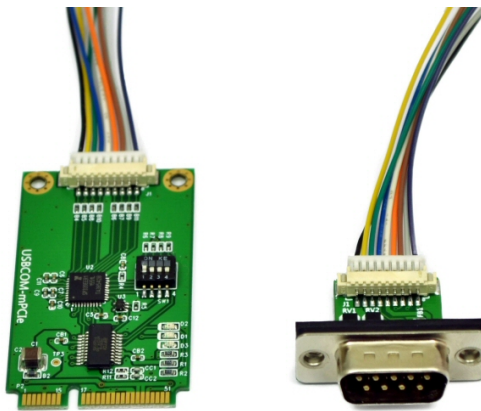


Figure 7: USB-COM Plus mPCIe

On the left this is the add-on card USB-COM Plus mPCIe. The cable connects to a DSub-9 adapter (on the right) to fix on the PC's case. The DIP switches shall be configured before closing the case.

2.3. Packing List

- USB-xCOM⁺ Device
- USB 2.0 High Speed cable
- 4 rubber feet
- USB-4COM⁺ and USB-8COM⁺ only: CD-ROM containing Drivers, Configuration software utility, User manual
- USB-4COM⁺ and USB-8COM⁺ only: 3 pin power connector
- USB-4COM⁺ and USB-8COM⁺ only: 19" Rack mounting brackets
- USB-4COM⁺ and USB-8COM⁺ only: Wall mounting plates

2.4. Optional Accessories

2.4.1. Mounting Options

- Art.No. #6692 DK-NCP Din-Rail Kit (USB-COM⁺ and USB-2COM⁺ only)
- Art.No. #6693 WK-NCP Wall-Mount Kit (2 x side wings)
- Art.No. #6698 DSK-NCP Din-Rail Side Kit (USB-COM⁺ and USB-2COM⁺ only)
Also order Art.No. #6692 to clip to the Rail.
- Art.No. #6699 DPK-NCP Din-Rail Plastic Kit (2 x DK-35A)
Also order Art.No. #6693 to mount on the device USB-COM⁺.

2.4.1.1. Warning: The packages of USB-4COM⁺ and USB-8COM⁺ include a Wall Mount Kit. There are two sets of screws in the package, long screws and short screws. The short screws are designed to fix the Wall Mount Kit. The long screws are intended to fix the 19-inch mounting brackets to the case. These long screws *must not* be used on the Wall Mount Kit, otherwise this may result in damage of the device.

2.4.2. Power Options

- Art.No. #6031 Power Adaptor 12V@1.0A (EU plug) (USB-4COM⁺ and USB-8COM⁺ only)

2.4.3. Connection Options

- Art.No. #663 DB9 male to Terminal Block
- Art.No. #6061 DB9 male to RJ45
- Art.No. #6062 RJ45 to DB9 male

2.5. About this Manual

This manual describes the hardware of USB-COM Plus models, as well as signal assignments of the serial connectors. The screen shots in here are made on an English language version of Windows 10. It is not difficult to find the appropriate part in any other language version of Windows. The current version of the driver is 2.12.26, the version of the firmware in USB-COM⁺ is 1.0.0.5.

Usually any hardware configuration (see [Configuration via DIP Switches](#)) is described before the software, including drivers. However the Configuration software utility of USB-4COM Plus and USB-8COM Plus models requires the serial port drivers installed, so these drivers are covered first.

3. The Drivers for Windows

This chapter of the manual documents the drivers for Windows Operating Systems. The driver is the same for all USB-COM Plus available, independent of the model. The USB-COM⁺ appear in Windows as several components, depending on the specific model. This can be seen in the Device Manager, while or after installation of the drivers.

- Single and Dual port models appear as a “USB Composite Device” first, which installs USB Serial Converter A and B. These then install one or two serial ports.
- Multiport models with 4 and 8 serial ports first appear as a “Generic USB Hub”, plus USB Composite Devices. These install USB Serial Converter A to D each. The converters then install all the serial ports.
- The models with 4 and 8 serial ports also install an additional virtual serial port, this is later used for configuration by software.

Windows already provides a driver for a USB Hub, and it also handles USB Composite Devices without configuration and help by the user. The specific drivers are required for the USB Serial Adapters, and the USB Serial Ports installed by them.

The manual first describes the process of installing the driver software, followed by instructions for removing the drivers in the unlikely case you wish to do this. After this the configuration of the serial ports is described. There is no option to configure the other drivers installed for the USB-COM⁺, so the options are the same for all different models.

Many USB-COM⁺ may be installed simultaneously. In fact the driver does not limit the number of USB-COM⁺ installed, Windows does. In Windows the maximum number of serial ports installed is 256, no matter which driver is installed and controls them. The current version of the driver for USB-COM⁺ is 2.12.26.

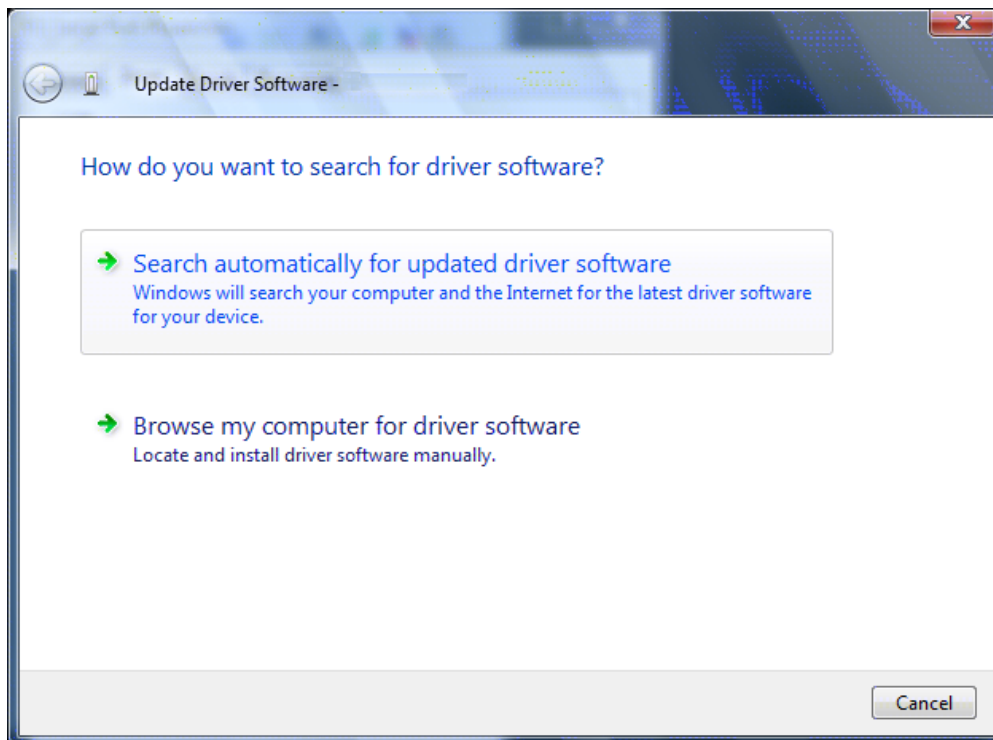
With the current driver version there are two methods of installation. There is the automatic variant, where Windows detects the new hardware as soon as it is connected to the PC. Then Windows contacts to Windows Update and installs the latest version of the drivers. The other way of installation operates offline and uses a Pre-Installation Executable to put the drivers onto the system, *while the hardware is not yet connected to the PC*. Once it is connected later, Windows already knows about the drivers, and installs them without further interaction by the user.

3.1. Install Drivers from Windows Update (Automatic)

This is the easy method of driver installation for USB-COM Plus models. The required steps are very simple. First connect your PC to the network, so it can visit the Windows Update websites provided by Microsoft. Then connect the USB-COM⁺ to a USB port of your PC. Windows will detect the new hardware, and contact Windows Update for available and latest drivers.



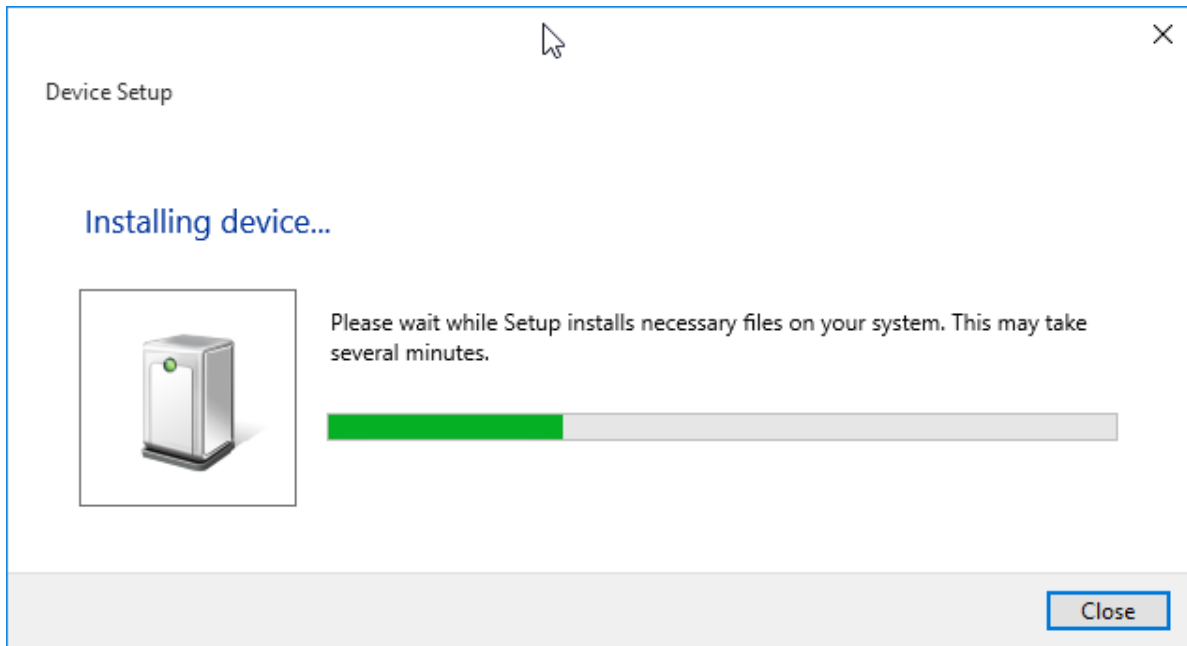
(a) Windows XP



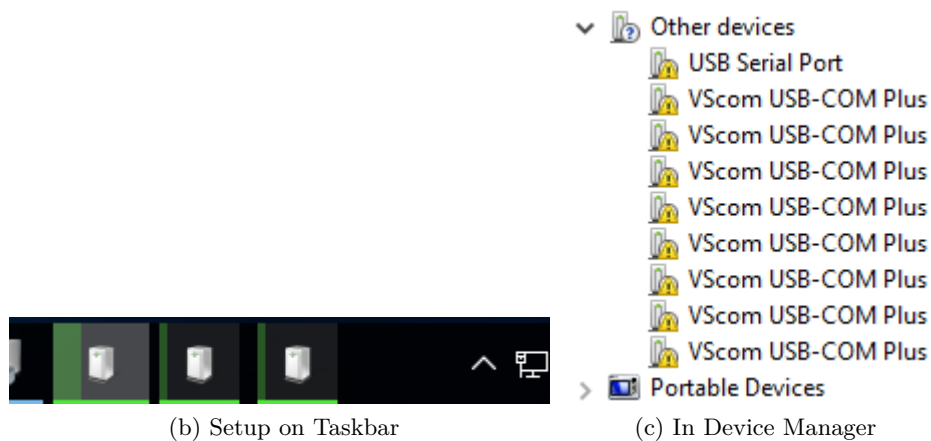
(b) Vista

Figure 8: Confirm Windows Update

On Vista and Windows XP you have to allow the contact to Windows Update. On XP select “Yes, this time only” and proceed, on Vista select “Search automatically...”.



(a) Setup Progress



(b) Setup on Taskbar

(c) In Device Manager

Figure 9: Automatic Installation

The USB Serial Converters and then the USB Serial Ports are installed with these drivers. Depending on the model of USB-COM⁺ there are several installation tasks operating in parallel. If the Device Manager is open at this time the serial ports appear in “Other devices” first, before installation in “Ports (COM&LPT)” is finished. No interaction from the user is required in this process.

Note: This is the recommended practice of driver installation. However recently the answers from the servers on Windows Update have been seen as rather slow, so this process may take much too long time. Then it is better to cancel the installation, and use the driver installation by pre-installation executable. If the installed drivers are not the latest version, Windows will update the drivers automatically later.

3.2. Install Drivers by Pre-Installation Executable (Offline)

The driver installation program is usually named like USBCOM-CDM-21214.exe, depending on the version number. This file is found on the driver CD-ROM. Just log on to Windows with Administrator privileges, right-click on the program and select “Run as administrator...”³. It is a self-extracting archive, which in turn executes the included installation software. Just follow the steps shown in figures 10 to 14.

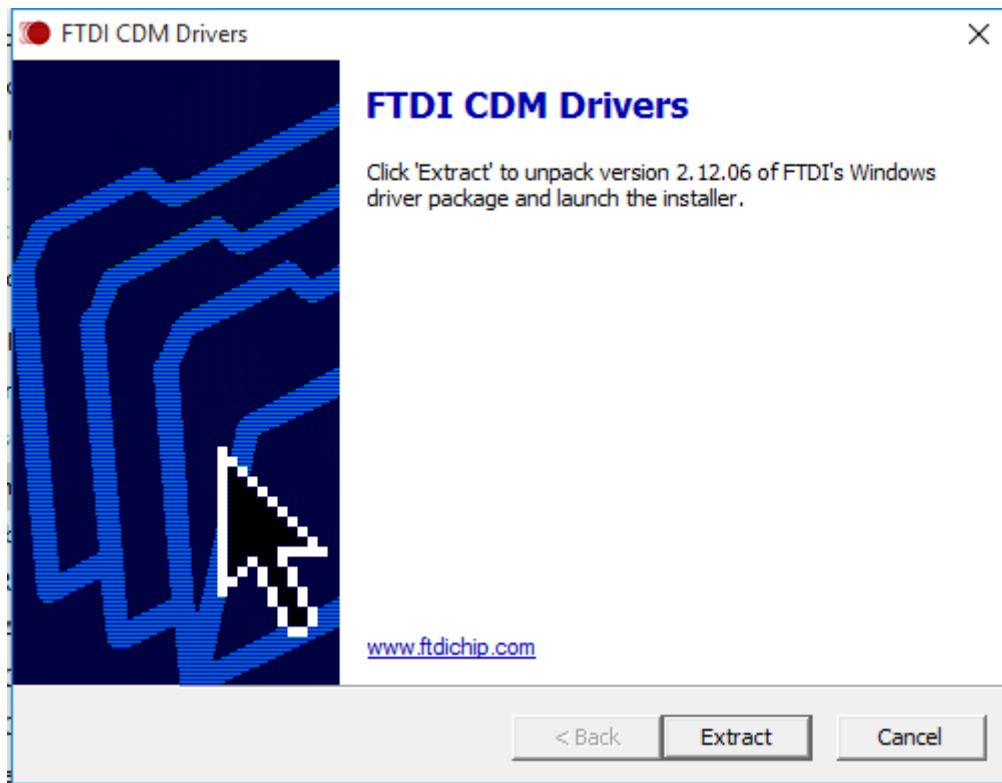


Figure 10: Extract to temporary Space

³On Windows XP just double click

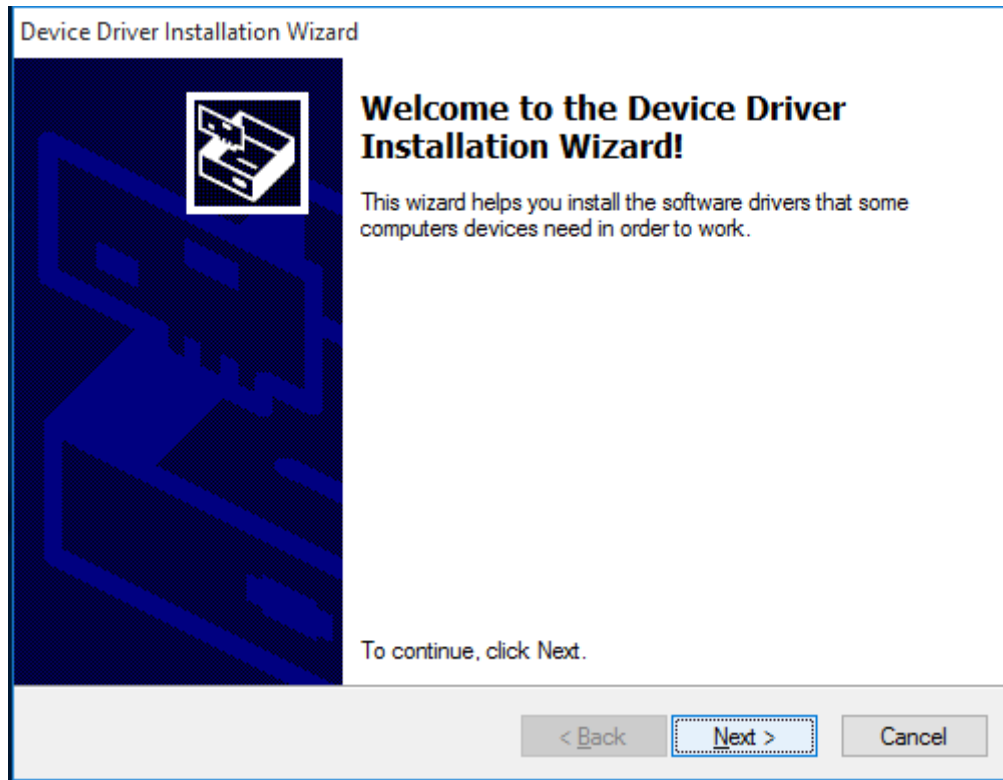


Figure 11: Start Installation

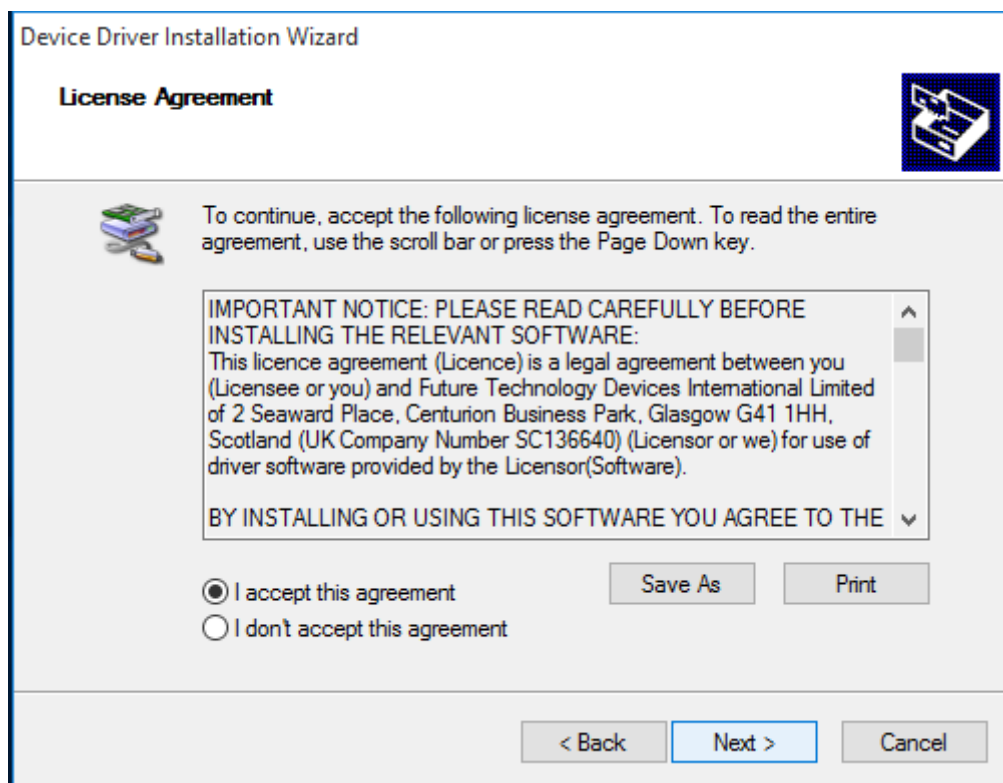


Figure 12: Confirm License conditions

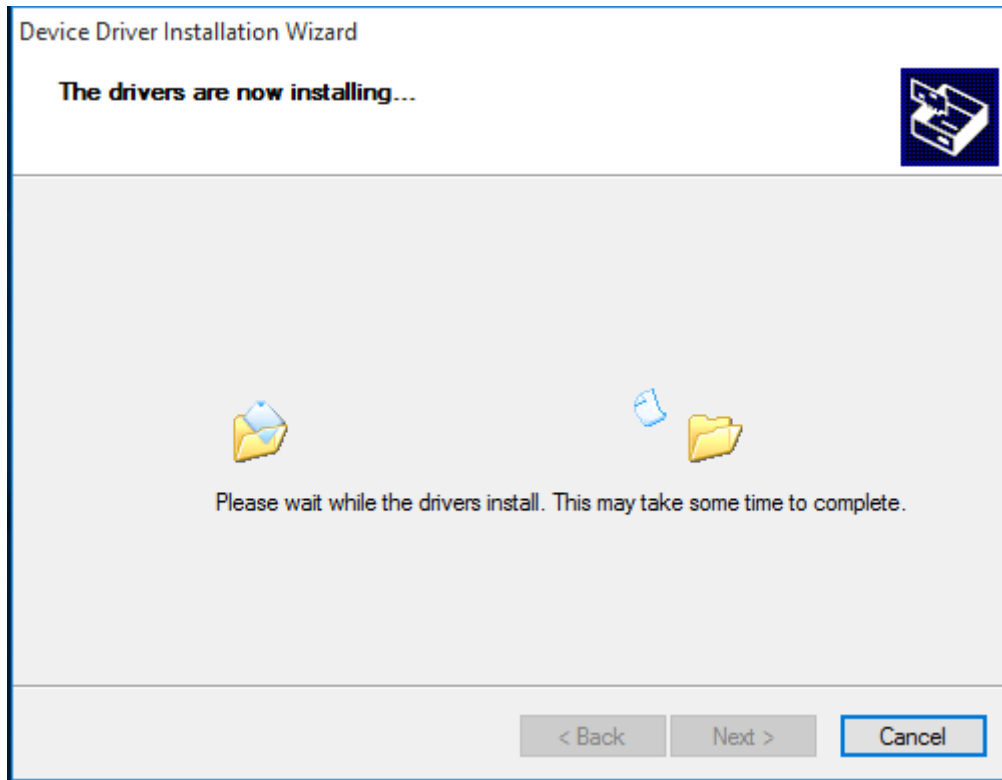


Figure 13: Copy Files and Install Drivers

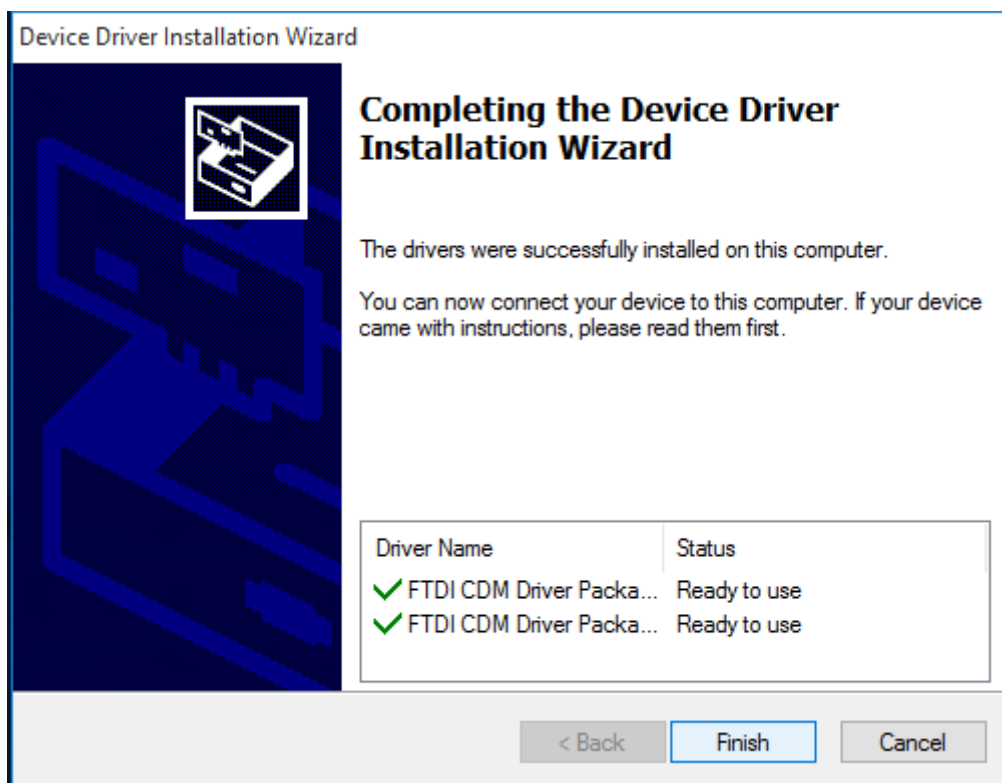



Figure 14: Installation finished

The result of this process⁴ is a Windows system which has the USB-COM Plus added as known hardware. Now connect your USB-COM⁺ to a USB port on your PC. Windows detects the new hardware, and attempts to install the drivers. On Vista or Windows XP it probably offers the option to search the Internet for more recent drivers (fig. 8), just say “No, not this time” to this question. In the next steps select the Automatic option, and click on Next and Finish as requested. Windows knows about the drivers, and installs all components and serial ports without further interaction.

3.2.1. Check Installation of the Drivers

Open the Device Manager, and see if all serial ports are available in the “Ports” Device class. Open the properties of each serial port.

3.2.2. Uninstall Drivers

First disconnect all USB-COM Plus from your system. On the CD-ROM you’ll find the CDMun-installerGUI.exe. Right-Click on this, and select “Run as administrator...”. 

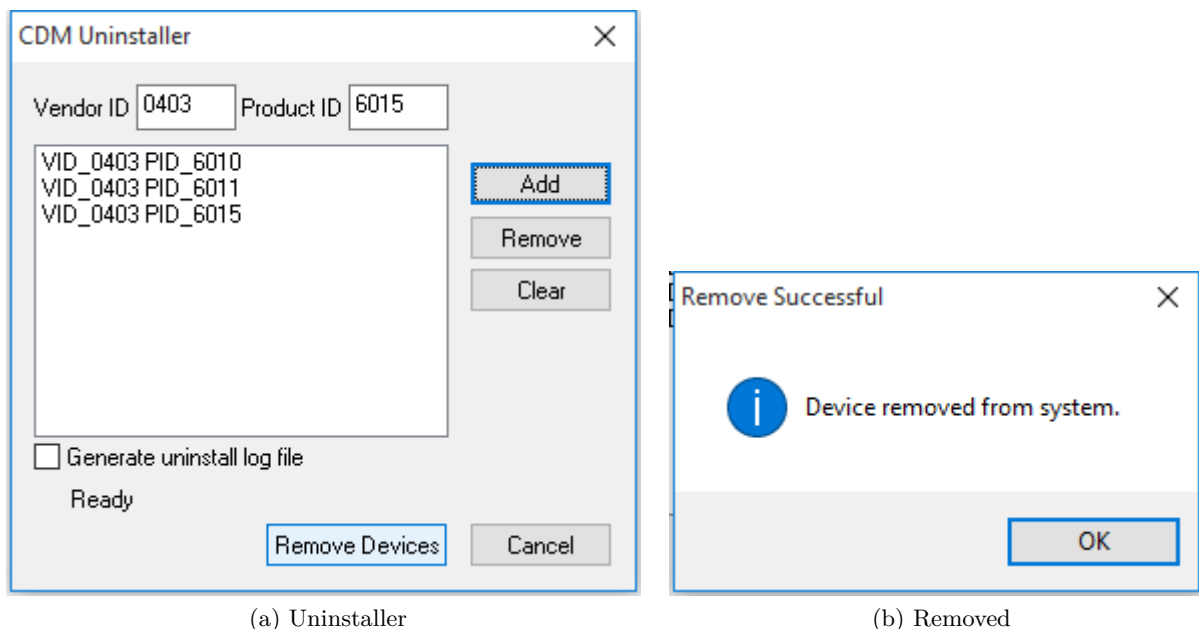


Figure 15: Uninstall Drivers

Change the value in “Product ID” to the values of 6010, 6011 and 6015. For each value click on the “Add” button, so the window looks as above. Then click on “Remove Devices”. The process shows some “removed” messages. Ignore any messages for failures. These may happen when no product with one of the given Product ID is available. This is no problem, just continue. And finally close the program.

This procedure not only removes all entries of USB-COM⁺ from the Device Manager, in the final step also the driver files are removed from the hard disk. As a result reconnecting a USB-COM⁺ causes Windows to start the search for drivers again.

⁴To have this installation perform really quick, disconnect the PC from any network prior to installation and connection of the USB-COM⁺ device.

3.3. Configure the Serial Ports

Open the Device Manager. In the Device Manager open the device class of “Ports (COM & LPT)”⁵. In this class you’ll see all the entries of “USB Serial Port”. Now open the properties of the serial port to configure.

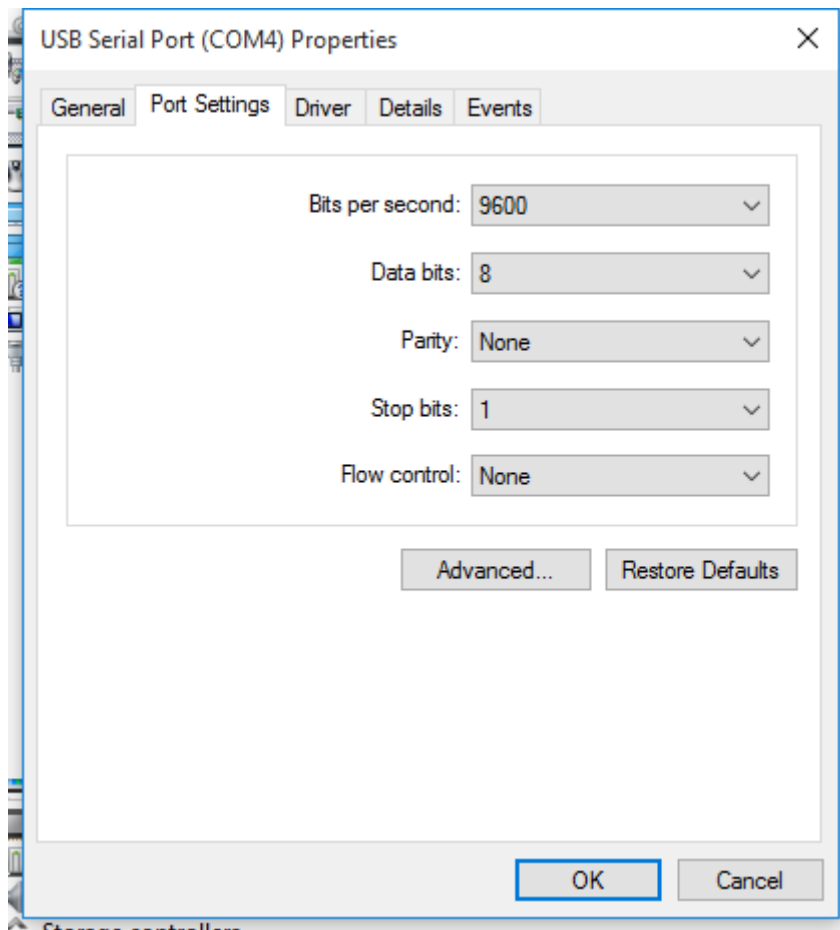


Figure 16: USB Serial Port Properties

Click on the button named “Advanced...” to see the special configuration options. They are available via the panel in figure 17.

⁵“Anschlüsse (COM und LPT)” in einem deutschen Windows

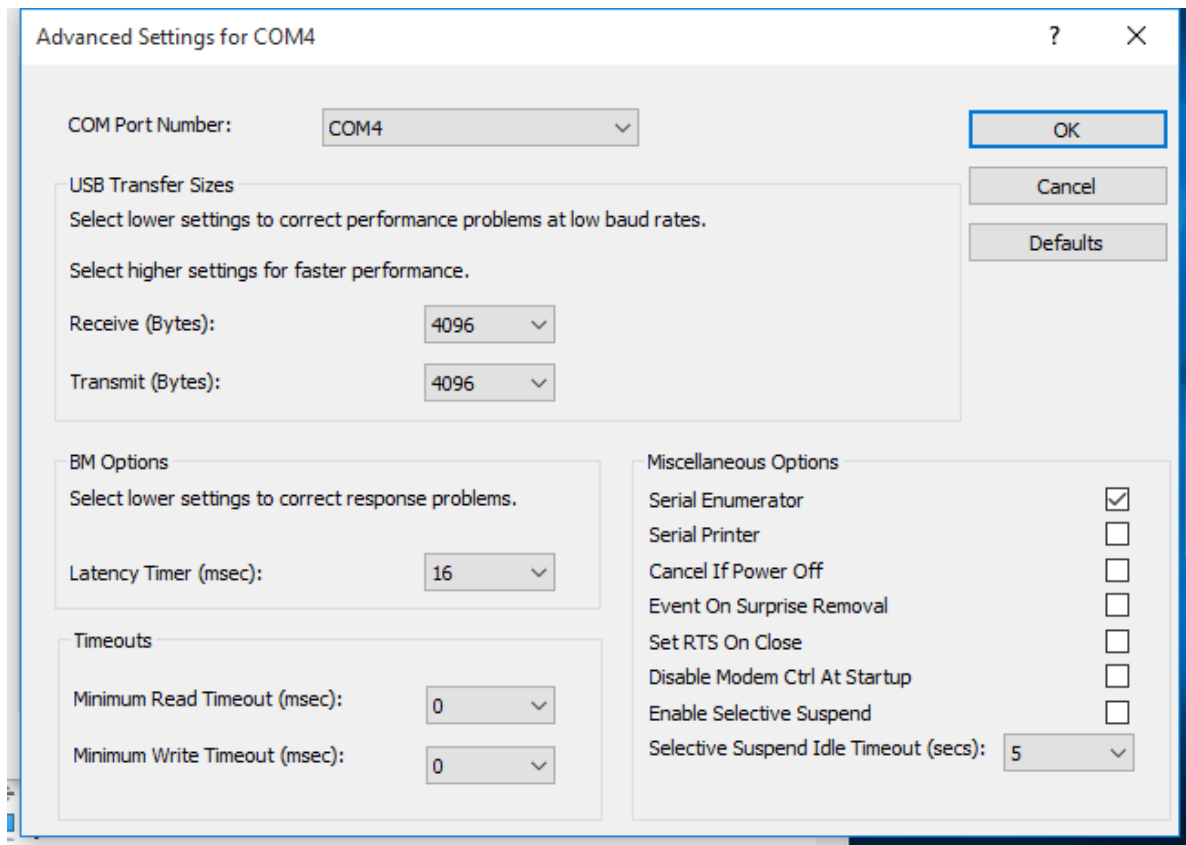


Figure 17: Advanced port options

3.3.1. Rename the Serial Port

The option “COM Port Number” on top of the advanced settings allows to rename the port. You may select any free port number. It is Windows which refuses a change, if the new port name is already occupied.

3.3.2. Enable faster Responses

The section named “BM Options” is in the middle of the left part. The parameter “Latency Timer” can be reduced to 1 millisecond, if there are problems with certain applications. The default value is 16, to reduce the USB protocol overhead and bandwidth usage. Usually the serial port buffers incoming data for up to 16 ms, or until 62 bytes of data are received. When either of the criteria becomes true, the data is sent to the Host PC. Then the application software will recognize the received data. If only a small amount of data is received, or the last part of the transmission does not trigger the 62 byte limit, this causes a short delay in receiving the data. As the consequence the application can not operate on the data very early.

Usually this is not a problem, but with certain serial protocols this can cause strange effects. So a reduction to 1 ms is appropriate then, faster reactions are not possible. This is caused by the protocol structure of USB, not by the serial port.

3.3.3. Miscellaneous Options

In the lower right corner of the Advanced Options (figure 17) are some rarely changed options.

3.3.3.1. Serial Enumerator This option is enabled by default. When the driver is started, Windows tries to detect hardware attached to the serial port. There may be a mouse or a modem. Most often the attached device is neither of that, so it is not a problem to disable this option. Further some devices permanently send data to the Host, without special requests. Windows may falsely recognize such devices as a serial mouse. This causes your pointer to randomly jump over the screen. Disabling the option prevents this.

3.3.3.2. Serial Printer When flow control like RTS/CTS stops data from being sent to the serial port for a sufficient time, this will cause an error event for the application. However if the attached serial device is a printer, sometimes this will halt for paper feed, especially for manual interaction. This will cause the data flow to stop for a significant long time.

Enabling this option causes such errors never to be reported to the application. So printing will never cause erroneous error messages.

3.3.3.3. Cancel if Power Off Instead of a regular shut down the computer can be sent into sleep or hibernate mode. In this situation neither the serial port nor the driver can perform any requests. In rare situations this can cause problems.

Enabling this option dumps any open requests for driver or hardware, if the computer is powered off by any Suspend state.

3.3.3.4. Event On Surprise Removal Usually this option is disabled because software does not use it. When it is enabled, application software can request to be notified if the hardware of the serial port is removed from the system while it is in use. So instead of claiming a malfunction of the attached serial device it can correctly report the removal of the port.

3.3.3.5. Set RTS On Close Usually the signal RTS is disabled when the port gets closed. However some kind of serial hardware may require a permanently active RTS for correct operation. Otherwise there may be error messages when the port is opened again to use the device.

Enabling this option causes the RTS to stay active even when the application closes the serial port.

3.3.3.6. Disable Modem Ctrl At Startup Usually on startup the modem control signals RTS and DTR follow the behavior of standard ports (i.e. Com1). Due to the longer timing compared to built-in serial ports a very short enable or disable pulse on the control signals may become a comparably long pulse on the USB serial port. Such a long pulse can cause external hardware to malfunction. By correct configuration of the serial port application software can avoid that problem. However since it does not appear on Com1, most software does not care about that. This option will help to “heal” such problem.

Note that if the "Serial Enumerator" option is enabled, in the enumeration sequence Windows causes the control signals to change. So if it is necessary to select "Disable Modem Ctrl At Startup", it is likely that "Serial Enumerator" should be disabled too.

3.4. USB-COM Plus Configurator

To configure all parameters (see section 3.3) of the serial ports will be a clumsy work. To do this in a more easy way there is a special tool to perform this for a collection of ports all at once. This is the USB-COM Plus Configurator.

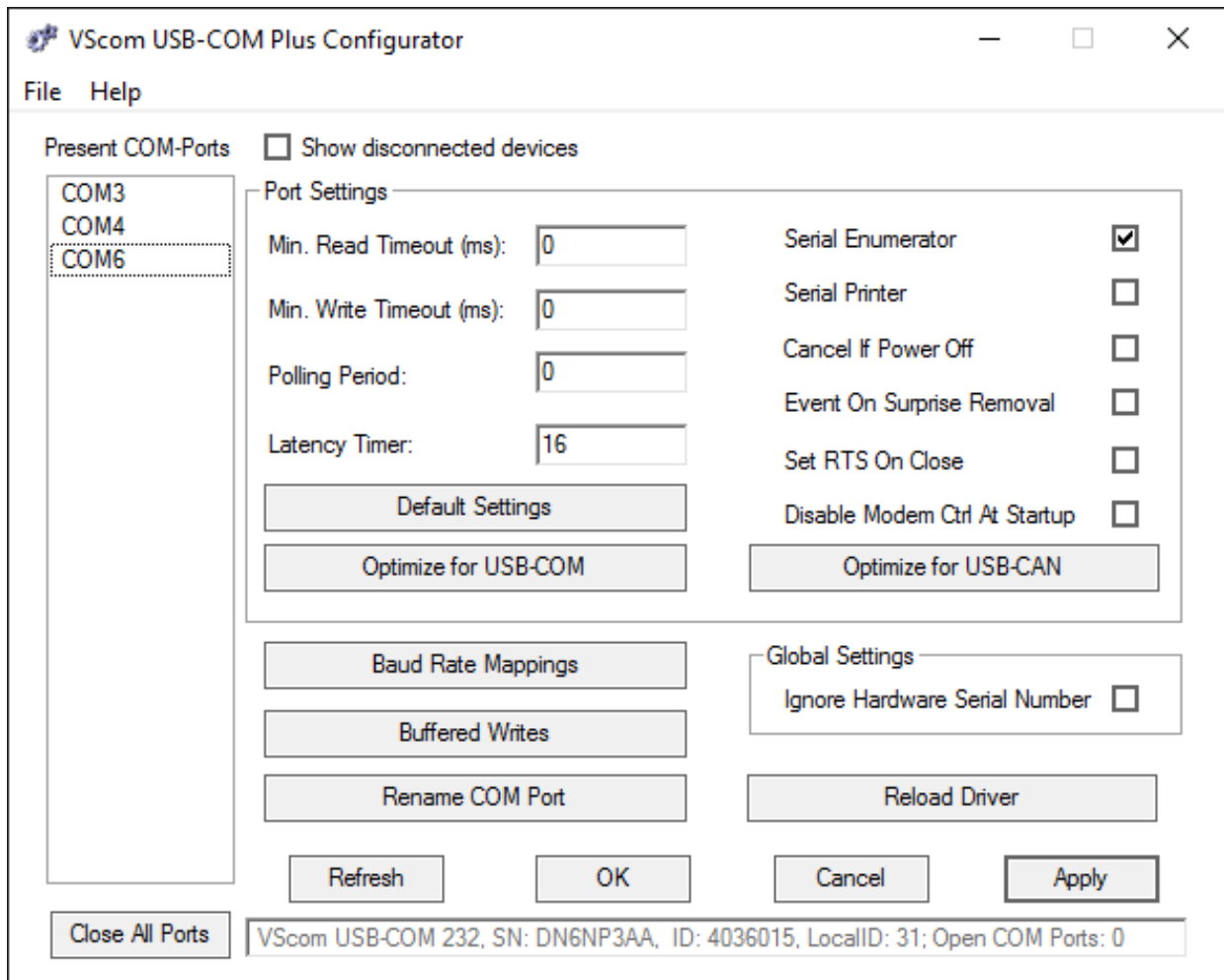


Figure 18: Driver Configurator

In the left list the user selects the COM Ports to configure, multi-select options are available. Aside from the individual parameters in the top area, there is one button for "Default Settings", and a convenient to "Optimize for USB-COM". This configures for best practice parameters, based on experiences.

3.5. Optimizing Data Throughput

Roughly speaking, the USB allows one operation between Host PC and attached USB device per millisecond. This is the basic reason why the Latency (see 3.3.2) can't be shorter than 1 ms. But there is a second effect causing slow transmissions.

Many software sends the data to the driver byte-by-byte. This is not a problem with built-in ports like Com1. Except for extreme high transmission rates the operation time of the driver is much shorter than the serial transmission time. So when the second and following byte are sent to the driver, the previous bytes did not completely leave the PC and its buffers. The serial port does not run out of data.

This is different with USB serial ports. No matter how fast the byte is transmitted over the USB cable (12 Mbit/s or 480Mbit/s), it takes 1 ms to send the next byte. Calculated the other way this is 1000 byte per second, which is equivalent to 9600 bit/s. The user feels the transmission as sluggish. Raising the serial speed does not help, since this is not the problem.

However it requires nearly the same millisecond to send one byte or more. So if the application sends complete buffers instead byte-by-byte, the driver can send more than one byte per millisecond. This way the USB serial port is permanently fed with work to send.

4. Hardware Configuration

The serial ports of USB-COM Plus are configured by DIP switch. The devices with one and two serial ports⁶ provide a block of switches for each serial port. The devices with four and eight serial ports⁷ provide one common block for all serial ports. This either configures all ports to the selected operation mode, or it enables selective per-port configuration by easy to use software. There are no Jumpers inside the USB-COM⁺, so it is never necessary to open the case for configuration.

The configuration by software of course requires to have a driver for USB Virtual Serial Ports on the system. That is not a problem, because without such driver the USB-COM⁺ can not be used anyway. The configuration parameters are saved to a non-volatile memory inside the USB-COM⁺, so configuration and use may happen on different systems.

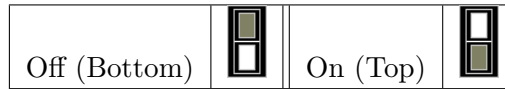
4.1. Configuration via DIP Switches

4.1.1. Models with One and Two serial Ports

The models USB-COM Plus Mini and USB-COM Plus Mini ISO are configured via one block of DIP switches, accessible from the bottom side. Add-on card USB-COM Plus mPCIe is configured on the top/component side. The models USB-COM Plus, USB-COM Plus ISO, USB-2COM Plus and USB-2COM Plus ISO provide one block of DIP switches for each serial port. The switch labeled 4 on each block controls the termination in RS422 and RS485 modes, by attaching 120 Ω to the receiving lines; that switch has no effect in RS232 mode.

⁶USB-COM⁺ Mini, USB-COM⁺ Mini ISO, USB-COM Plus mPCIe, USB-COM⁺, USB-COM⁺ ISO, USB-2COM⁺ and USB-2COM⁺ ISO

⁷USB-4COM⁺, USB-4COM⁺ ISO, USB-8COM⁺ and USB-8COM⁺ ISO



(a) DIP Switch Description

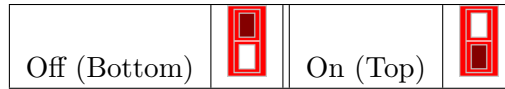
S1	S2	S3	Mode	
On	Off	Off	RS232	
On	On	On	RS422	
On	On	Off	RS485 "4w" ¹	
Off	On	Off	RS485 "2w" ²	

(b) Operation Mode

S4	Termination	
On	Active 120 Ω	
Off	Inactive	

(c) Termination for RS422 / RS485

Table 10: Configuration USB-COM⁺ Mini and USB-COM⁺ mPCIe



(a) DIP Switch Description

S1	S2	S3	Mode	
On	Off	Off	RS232	
On	On	On	RS422	
On	On	Off	RS485 "4w" ¹	
Off	On	Off	RS485 "2w" ²	

(b) Operation Mode

S4	Termination	
On	Active 120 Ω	
Off	Inactive	

(c) Termination for RS422 / RS485

Table 11: Configuration USB-COM⁺ and USB-2COM⁺

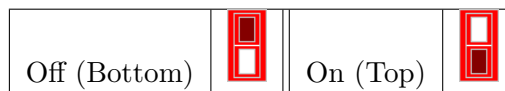
Note 1: This is the Full Duplex Mode

Note 2: This is the Half Duplex Mode. Without further specifications this is usually referred to simply as RS485

4.1.2. Models with Four and Eight serial Ports

The models USB-4COM Plus, USB-4COM Plus ISO, USB-8COM Plus and USB-8COM Plus ISO provide one block of DIP switches, common for all serial ports. The configuration selected by this block is either valid for all serial ports on the device. Or it is the selection to configure by software, in this case extra software will configure each serial port independent from any other.

In RS422 and RS485 modes the switch labeled 4 attaches an internal resistor of 120 Ω to the receiving lines. In other modes the switch has a different function.



(a) DIP Switch Description

	Line Mode, Comment	S1	S2	S3	S4	Switch Positions
	Selected by Software ¹	Off	Off	On	On	
RS232	Operation	On	Off	Off	Off	
RS422	4-wire Operation	On	Off	On	Off	
	4-wire with Rx-Termination	On	Off	On	On	
RS485 ²	4-wire Operation	On	On	On	Off	
	4-wire with Rx-Termination	On	On	On	On	
	2-wire Operation	On	On	Off	Off	
	2-wire with Termination	On	On	Off	On	

(b) Operation Mode

Table 12: Configuration USB-4COM⁺ and USB-8COM⁺

Note 1: The Master DIP switches configure all serial ports to a common operation mode. If diversity in line operation modes is intended, the switch must be set to »Selected by Software«, and the configuration is done via a Windows software or a Terminal Emulation.

Note 2: In RS485 mode the USB-COM⁺ performs the required activation and disabling of the RS485 transmitter by an internal automatic. Application software does not have to perform special operations.

These configurations are selected to be similar with NetCom Plus Serial Device Servers.

4.2. Configuration via Windows Software

This mode of configuration is available on USB-4COM Plus and USB-8COM Plus models.

If the drivers for Windows are not installed, please do that now (chapter 3). Then locate the program UsbComCfg.exe on your CD-ROM. Close any application currently operating via the serial port(s) provided by your USB-COM⁺. Then start the UsbComCfg program, it searches and detects all configurable USB-COM⁺.

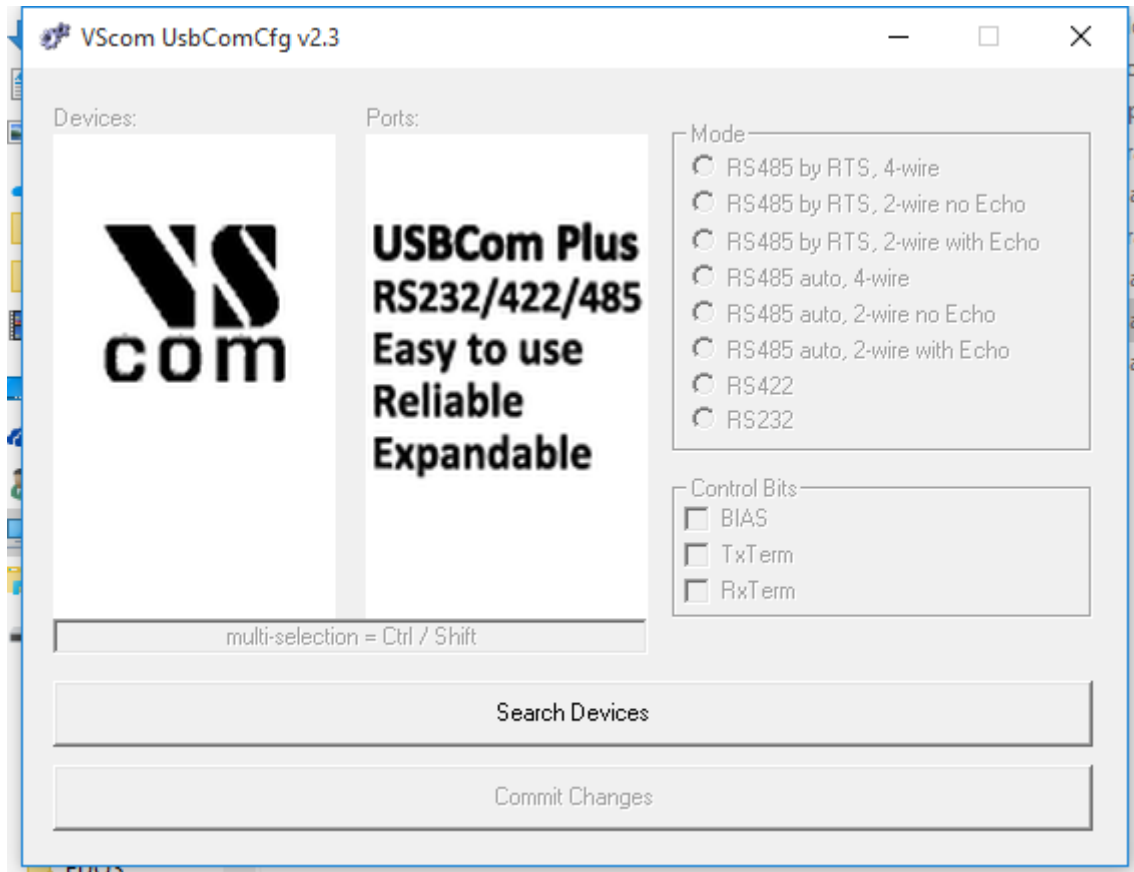


Figure 19: No Device

UsbComCfg will repeat the search when the appropriate button is clicked.

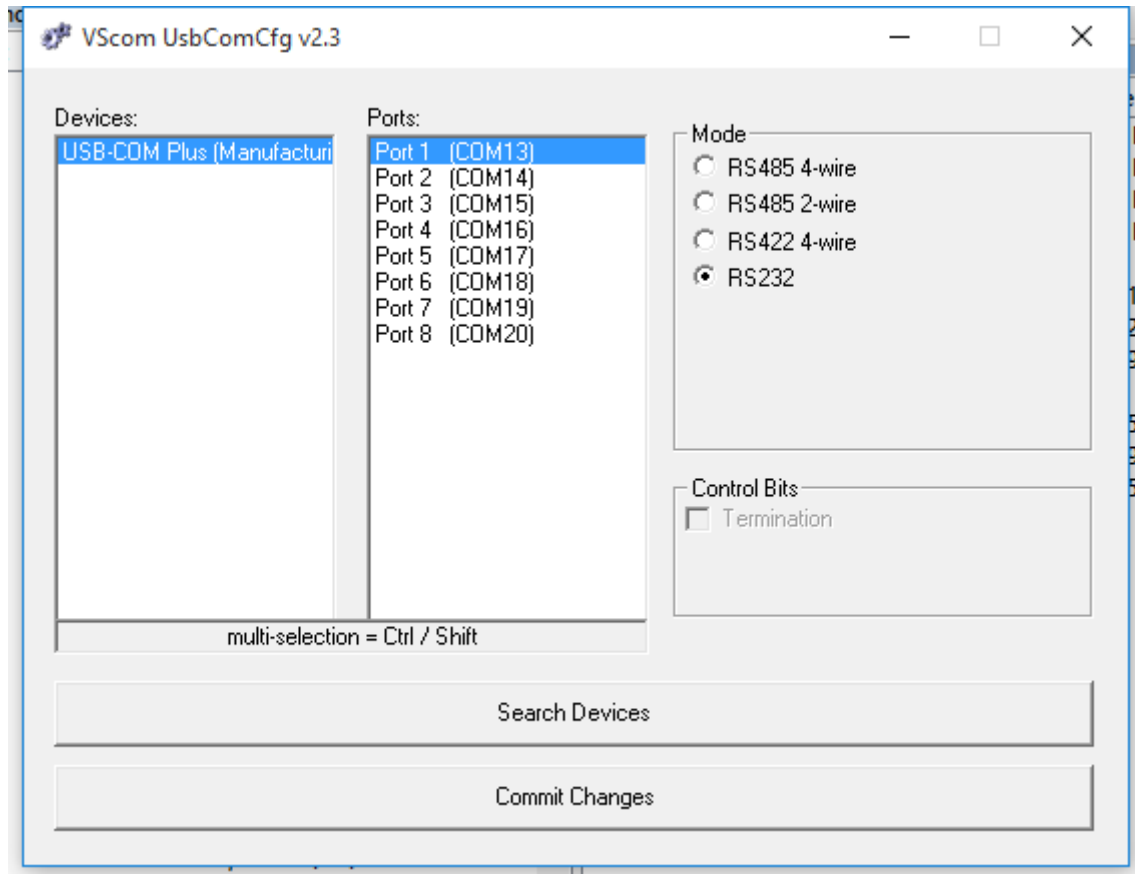


Figure 20: One USB-8COM Plus detected

The left pane displays all devices detected by the search.

By clicking a device in the left pane, a port in the middle pane can be selected. The operation mode is then changed by the radio buttons and the check box in the right pane. The button to “Commit Changes” transfers the required parameters to the selected device. They are stored in non-volatile RAM, and activated after the transfer.

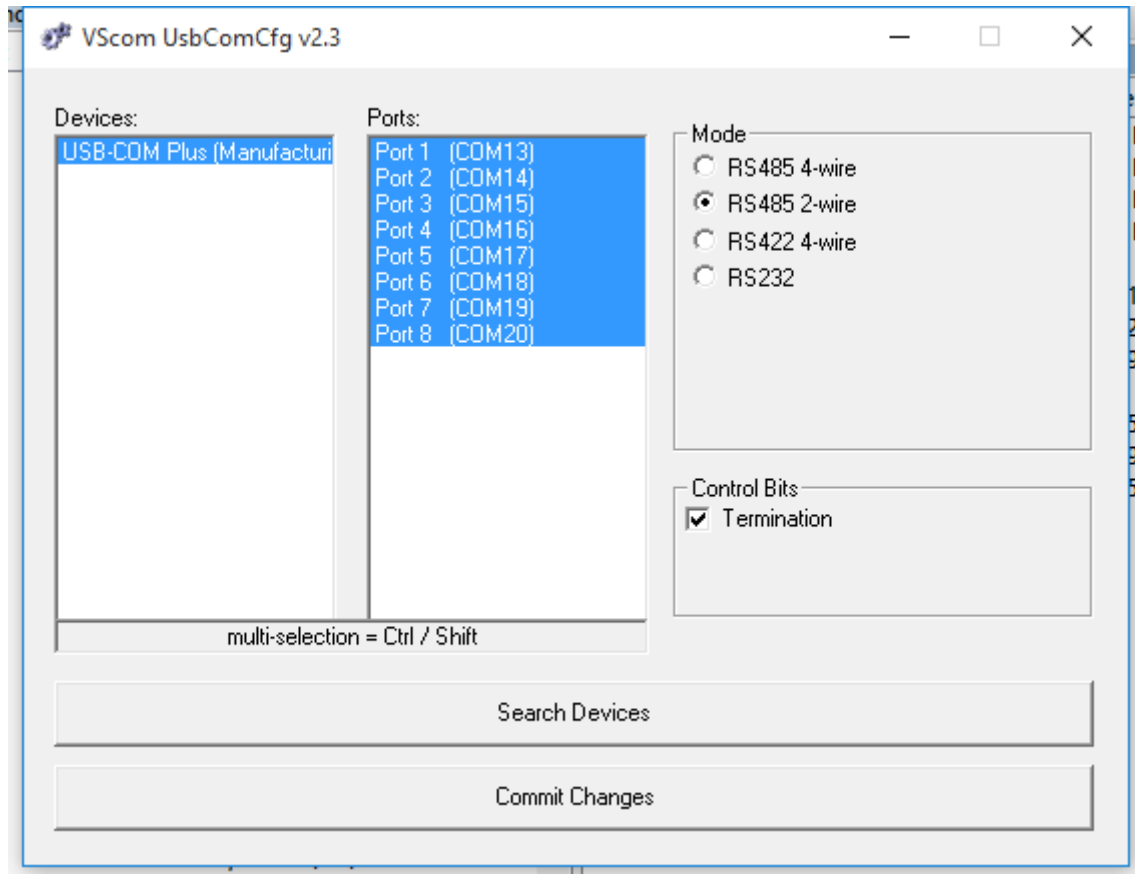


Figure 21: UsbComCfg Multi-Port Selection

You may select several ports of one device at the same time. This is done by the usual Windows method, i.e. clicking with Shift- or Ctrl-Key pressed. The selected configuration then applies to all selected ports, “Commit Changes” configures all of them at the same time.

Finally in the left pane you may select multiple devices at the same time. The configuration then applies to all ports of all selected devices.

4.3. Configuration via Terminal Program

Not all systems operate under Windows, and not all users have the UsbComCfg program available when they want to reconfigure the USB-COM Plus. Even then the configuration is possible via the virtual serial port implemented by the driver. If no driver is available on the system, the USB-COM⁺ will be of little use anyway. If the driver is not installed, please do that now (chapter 3 for Windows).

Then open your terminal emulation software, and open the special serial port provided by the USB-COM⁺. Configure the serial port for 38400 bit per second, eight bit per character without parity, and one stop bit (38400,8N1). Flow control is not used. Select the terminal emulation as a very dumb terminal. Incoming <CR> (Carriage Return characters) shall not result in a line feed, but <BS> (Backspace) should position the cursor one space to the left.

```
[8#1005] USB-COM Plus
```

Figure 22: Open Terminal Configuration

In your terminal screen press Enter, a line similar to the above will appear. The line lists the USB-COM⁺ found using this serial port, here it notes 8 serial ports and version 1.0.0.5

```
0 - Exit
-----
1 - Show all port configurations
2 - Change port configurations
3 - Default port configurations
4 - Save port configurations
-----
5 - Show configuration file
6 - Upload configuration file
>
```

Figure 23: Terminal Main Menu

The Main Menu appears. In this and the other menus just type the character given at the beginning of the line, to get the function described there. Hit the Enter key to activate the command. The optional result as well as the menu will be written on the screen. Just test this by hitting '0' and Enter to exit from the configuration.

In the Main Menu (figure 23) the upper section provides for status information and manual configuration of the serial ports. The lower section provides a method to save the configuration, especially for a later upload of these parameters.

4.3.1. Exit the Configuration (0)

```
> 0
```

Figure 24: Exit Terminal Configuration

Enter a '0' in the Main Menu, this will terminate the Configuration. Unsaved changes to the configuration are discarded.

4.3.2. Show Port Configurations (1)

```
> 1
Port 1: RS232
Port 2: RS232
Port 3: RS232
Port 4: RS232
```

Figure 25: Show all port configurations

The command '1' in the Main Menu lists the current configuration of all serial ports in the USB-COM⁺. This example is for a USB-4COM Plus with all ports in RS232 mode. The list is followed by the Main Menu again.

4.3.3. Change Port Configuration (2)

```
> 2
  0 - Back
-----
  1 - Port 1
  2 - Port 2
  3 - Port 3
  4 - Port 4
-----
  5 - All Ports
>
```

Figure 26: Change port configurations

Command '2' allows to change the configuration of some ports. The ports may be selected individually, or all at once for common configuration.

0 Selects the Main Menu again.

1 Selects the first serial port.

2 Selects the second serial port.

3 Selects the third serial port.

4 Selects the fourth serial port.

5 Selects all serial ports at once. This number changes depending on the number of available serial ports.

For this example port 1 is selected for configuration. To get details about the available modes please read section [2.2](#).

```
> 1
---> Port 1: RS232
-----
0 - Back
-----
1 - RS485 4-wire
2 - RS485 2-wire
3 - RS422 4-wire
4 - RS232
5 - by Software
-----
6 - Termination
>
```

Figure 27: Configure a serial port

The options are selected by entering the number, followed by the Enter key. '0' Returns to the selection of the serial ports, but so far the configuration is not activated. This requires an explicit command, see section [4.3.5](#).

4.3.4. Default Port Configuration (3)

Option '3' in the Main Menu (figure [23](#)) loads a standard configuration for all ports. This is RS232, Termination is disabled.

4.3.5. Save Port Configurations (4)

Each change in Port configuration is temporary, only saved in volatile memory. It needs explicit confirmation for using the new configuration. Option '4' in the Main Menu (figure [23](#)) performs this operation. The configuration data is stored to non-volatile memory inside the USB-COM⁺, and becomes active.

4.3.6. Show Configuration File (5)

Configuring each serial port (and each option of it) in a USB-COM⁺ can become a long process. And this has to be repeated on the next device, and so on. To make the handling much easier the current configuration can be saved to a file, for later use. Option '5' in the Main Menu (figure [23](#)) is prepared for this task.

```
-----  
5 - Show configuration file  
6 - Upload configuration file  
> 5  
  
file construction (decimal values):  
  
1. port number<CR>  
2. port mode<CR> (take a look at "Change port configurations")  
3. port control bits<CR> (bit0: Termination On)  
4. <CR> (empty line)  
  
99<CR> terminates the file  
<-- file start -->  
1  
4  
0  
  
2  
4  
0  
  
3  
4  
0  
  
4  
4  
0  
  
99  
<-- file end -->
```

Figure 28: Saving Configuration file

The configuration is displayed on the terminal screen. It is preceded by a short description of the file format. Using the Terminal Emulation software this output can be captured and saved to a file. It depends on the particular software how this is done, so please check your software documentation.

The configuration of each serial port is written on three distinct lines, followed by an empty line. Port number and operation mode are already described. Port Control is for Termination: a value of '1' activates this option.

4.3.7. Upload Configuration File (6)

When the Terminal Emulation software allows to save the configuration of the device, it will also allow to send a text file to the USB-COM⁺. This way a previously saved configuration can be

loaded to the USB-COM⁺. It can also be loaded to another device, for ease of configuring a series of USB-COM⁺. The last option '6' in the Main Menu (figure 23) provides this function.

5. Technical Background on RS485

This chapter will provide a little bit of theory about RS422 and RS485 data transmission. It is necessary to have this basic knowledge, to avoid or find errors in data transmission. Failures in cabling are responsible for the vast majority of transmission problems.

5.1. Transmission Technique

RS422 and RS485 use the same balanced transmission method. Signals are not transmitted by voltage on a single wire, as RS232 does. Instead two wires are used; when one carries high voltage, the other one carries low voltage. The signal is defined by the difference in voltage between those two wires. This hardens the transmission against noise. Usually twisted pairs are used, which further reduces the sensitivity for noise.

Typical voltages are +4V as high, about +0.5V as low. These voltages are generated and defined against the GND signal of the transmitter. The minimum *differential* voltage is required as $\pm 200\text{mV}$ by the specifications of RS422 and RS485. The receiver detects the polarity of the differential voltage, and thus gets a Zero (negative) or a One (positive) difference.

To do this detection the voltages on the receivers side have to be inside the bounds of the common voltage range defined as -7V through +12V, as measured against the local GND of the *receiver*. The specification of the common voltage range has been given to make the design of receiver circuits more simple and cost effective.

To make sure the signals meet the common voltage range, the GND of sender and receiver must be connected somehow, otherwise the voltages are undefined and may have any value. To assure the correct range RS485 (RS422) usually requires an extra wire for GND⁸, which is often forgotten. The connection may also happen by protective ground, or by other means. If the connection of GND is bad, it may be impossible to receive correct data.

5.2. Termination

When transmitted signals arrive at the end of a cable, they get reflected. They travel on the cable back and forth some more times, which is called ringing. This can cause false reading of transmitted data. When the reflections travel on the cable several times, they are damped and do no longer cause errors. This happens earlier if the cables are short.

For long cables Termination Resistors are required. These increase the damping of reflections. The value of the resistor must match the impedance of the cable, typically 120Ω for twisted pair.

As a rule of thumb⁹, when the cables are longer than $\frac{1000000}{\text{Bitrate}}$ (one million divided by the bit rate) in meters, you should consider Termination Resistors.

⁸with an optional resistor

⁹Assuming group speed of 100.000km/s, 10 travels to damp out, and 10% of bit time

5.3. Polarization

In RS485 the sender must activate the transmitter before sending data, and deactivate it when all data is sent. At times when no devices send data *all* transmitters are inactive. As the result the data lines are floating, and the differential voltage is undefined. It may happen the next data is not correctly recognized, because the change from undefined to data signals is not detected.

To avoid such problems the data lines should be polarized by resistors. These insure the differential voltage to be above +200mV. Typically the positive line is pulled to +5V, while the negative line is pulled to GND. When Termination Resistors are applied, the voltage on these must be +200mV or above. When the termination resistors are of the 120Ω types, the polarization resistors shall have about 750 to 1000Ω.

The Polarization Resistors must not be too small, because they add current to the signals, and a transmitter has to act against that. A typical transmitter provides up to 60 mA of current in high and low level.

5.4. 2-Wire Scheme

In many configurations a very simple cabling is required. RS485 allows for so called 2-wire cabling as shown below. Several devices are connected in parallel to the wires, which is called bus topology. Each device can either send own or receive foreign data at a given time, so it is operating in half-duplex mode.

Shown in figure 29 are three devices, RS485 specifies up to 32. The data lines are named as Data+ and Data-, a positive differential voltage is the state for a transmitted One. The GND is also connected between all devices as required, so the cabling is 3-wire in reality.

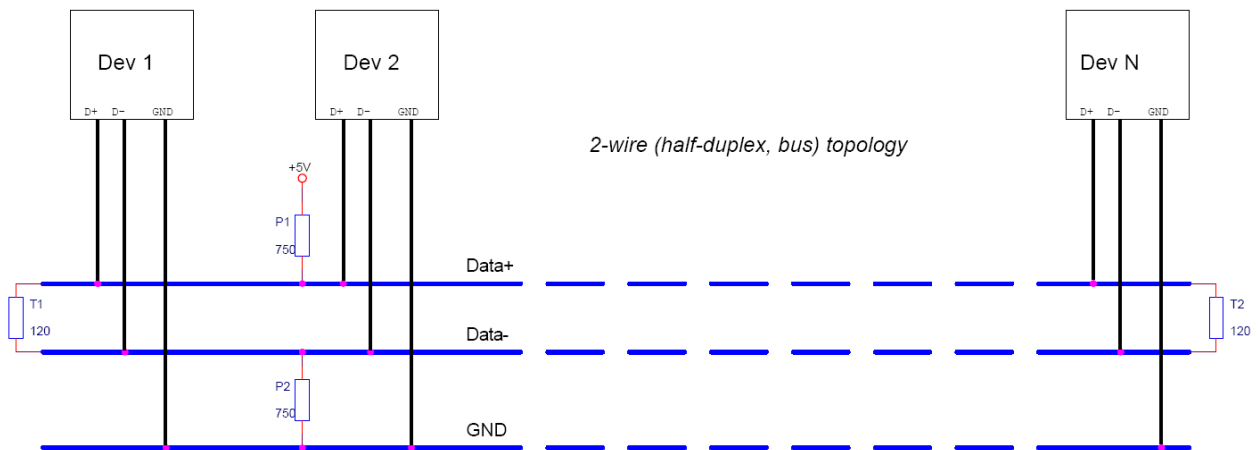


Figure 29: 2-wire cabling scheme

The resistors P1 and P2 are for polarization, T1 and T2 are for the termination function. Polarization of Data+ and Data- appears only once on this net, the termination is at the physical ends of the cable.

All devices appear the same on the cable, they have the same function. There is no Master or Slave defined by the hardware. Such functions are implemented by way of the data transmission protocol. Also RS485 addresses are defined by that protocol, as well as bus access.

5.5. 4-Wire Scheme

RS422 requires dedicated wire pairs for transmit and receive. The transmit wires are used to send data to as many as 10 receivers, as stated in the specifications of RS422. Since the VScom products use RS485 line driver technology, up to 32 receivers are possible.

While one pair is used to transmit, a second pair is available to receive data at the same time. When only two devices are connected, this is a possible replacement of classic RS232 connections (Point-to-Point).

In RS485 4-wire mode the transmit wires may be shared between dedicated stations. As an example a second station can be a backup master for the network. Masters can send data and commands to one station, while they receive information from another device.

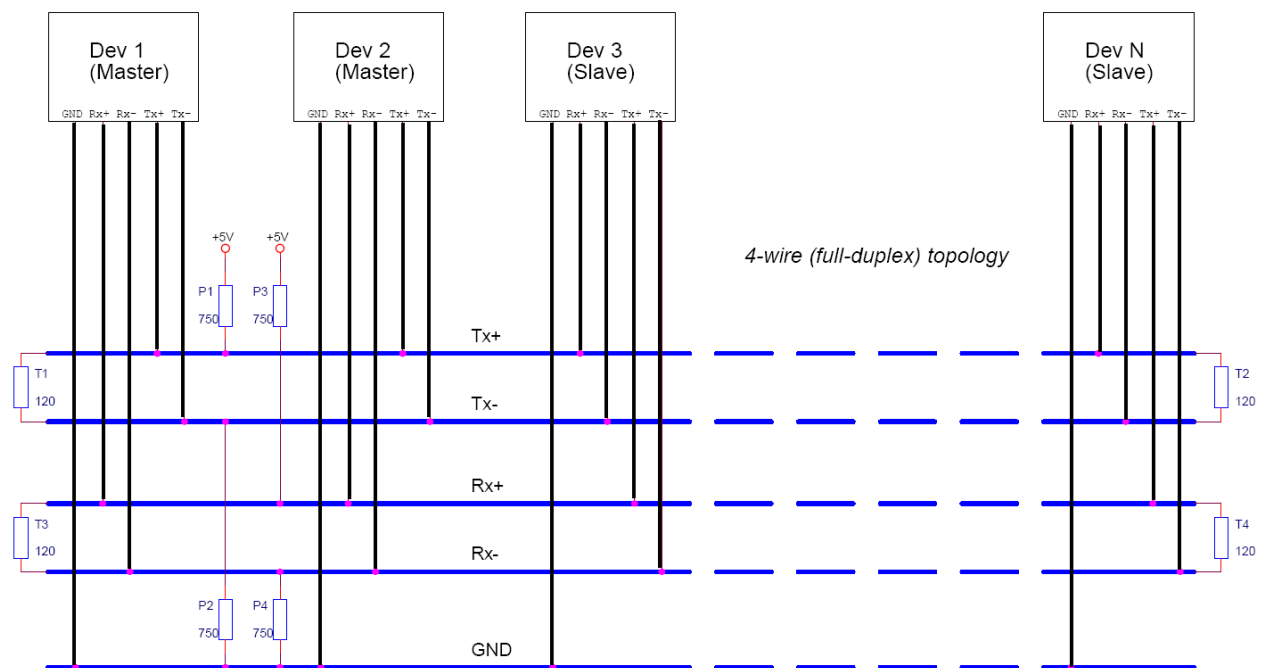


Figure 30: 4-wire cabling scheme

Figure 30 displays the wire pairs Tx_{\pm} and Rx_{\pm} as named for Devices 1 and 2. There are more slave devices, only two of them are shown. The slaves transmit on the Rx-lines, and receive on the Tx-lines. To implement the Common Voltage Range as specified the GND line is also connected.

Masters on the network are identified because they transmit on the Tx-lines. The two Masters in figure 30 have to synchronize their use of the Tx-lines by extra means. E.g. Dev 2 can be a backup master, which is manually activated.

Also indicated in figure 30 are polarization resistors P1/P2 for Tx_{\pm} and P3/P4 for Rx_{\pm} , only once per wire pair. Further Termination resistors T1/T2 for Tx_{\pm} and T3/T4 for Rx_{\pm} are added, one at each end of the cable. To ensure the Common Voltage Range the GND is also connected at each device, making this a 5-wire connection.

5.6. On USB-COM Plus Devices

As described above the USB-COM Plus provide an internal Termination resistor with a value of $120\ \Omega$. This is activated by defining the configuration of the serial port.

The receiver circuits for RS422 and RS485 do not require resistors for BIAS function. Using termination resistors the differential voltage is 0 Volt when all devices are inactive. The circuits recognize this voltage still as positive, so a logical One is detected. Therefore the USB-COM⁺ do not provide resistors for BIAS function.

6. Connector Definitions

All the USB-COM Plus devices provide the today standard DSub9 male connector for the signals of the serial ports. An adapter to Terminal block for RS422 and RS485 signals is available as an add-on (see 6.3).

The signal assignment for RS232 on DB male connectors is defined by the term “RS232” already. Below is the signal definition for the connectors used by the USB-COM Plus.

6.1. DB9 male

Usually this type of connector is used by customers. “4w” denotes the Full-Duplex mode, usually named “4-wire”. Similarly “2w” denotes the Half-Duplex modes.

Pin	RS232	RS422	RS485 “4w”	RS485 “2w”
1	DCD	Tx-	Tx-	Data-
2	RxD	Tx+	Tx+	Data+
3	TxD	Rx+	Rx+	
4	DTR	Rx-	Rx-	
5	GND	GND	GND	GND
6	DSR			
7	RTS			
8	CTS			
9	RI			

Table 13: DB9 male Connector

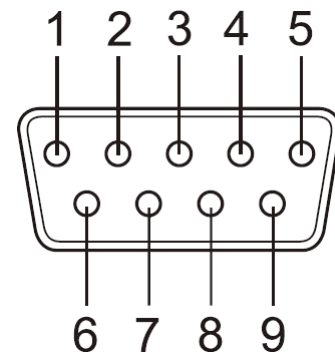


Figure 31: DB9 male Connector

6.2. Serial Connector mPCIe

The add-on card USB-COM Plus mPCIe is shipped with a connection cable and a DB9 male case adapter as accessories. This case adapter is identical to all USB products, see above. To configure your own connection cable find Pin 1 of the card connector near the label J1.

Pin	1	2	3	4	5	6	7	8	9
Signal	RI	CTS	RTS	DSR	GND	DTR	TxD	RxD	DCD

Table 14: mPCIe Connector

6.3. Terminal Block Adapter

When using RS422 and RS485 as transmission mode, sometimes the communication cables do not use a connector. Instead the wires are directly mounted into a clamp. To provide this options an adapter from the DB9 male connector to Terminal Block is available. This connects the pins 1 through 5 to a clamp (Signals Tx- to GND).



Figure 32: Terminal Block

6.4. Adapter RJ45

This adapter enables to use standard Cat5 cables for transmission in RS422 and RS485 modes. The signal assignment is chosen in a way the differential signal pairs for Tx+/- and Rx+/- (Data+/-) use a twisted pair in the cable. Plug the DB9 female side into the serial connector on USB-COM Plus, and use a Cat5 cable of convenient length for the transmission.



Figure 33: DB9-male to RJ45

Pin DB9	RS232	RS422	RS485 "4w"	RS485 "2w"	Pin RJ45
1	DCD	Tx-	Tx-	Data-	8
2	RxD	Tx+	Tx+	Data+	7
3	TxD	Rx+	Rx+		3
4	DTR	Rx-	Rx-		6
5	GND	GND	GND	GND	5
6	DSR				4
7	RTS				2
8	CTS				1
9	RI				

Table 15: DB9 to RJ45

The adapter with reverse function converts signals from standard Cat5 cables back to the usual DB9 format. When used in pairs such transmission over Cat5 cable is like a straight cable with a female connector at the first end and a male connector at the final end.

This is a possible use in RS232 mode, but limitation of transmission length still applies.

However the better use is for transmission in modes of RS422 or RS485. The differential signals for Tx+/- (Data+/-) and Rx+/- use twisted pairs in the Cat5 cable. It is possible to have a good connection with several hundred of meters in length, therefore the isolated serial ports are recommended for transmission over such long distances.



Figure 34: RJ45 to DB9-male

A. History

October 2015 Release of Products and Manual

May 2016 Update with USB-16COM Plus option and Windows Driver
Add USB-COM Plus Configurator Application
Add USB-COM Plus Mini device

February 2017 Add Connection Adapters
Current driver is version 2.12.24
Warning for Wall Mount Kit (long screws not allowed)

March 2017 Add PCI Express Mini Card USB-COM⁺ mPCIe