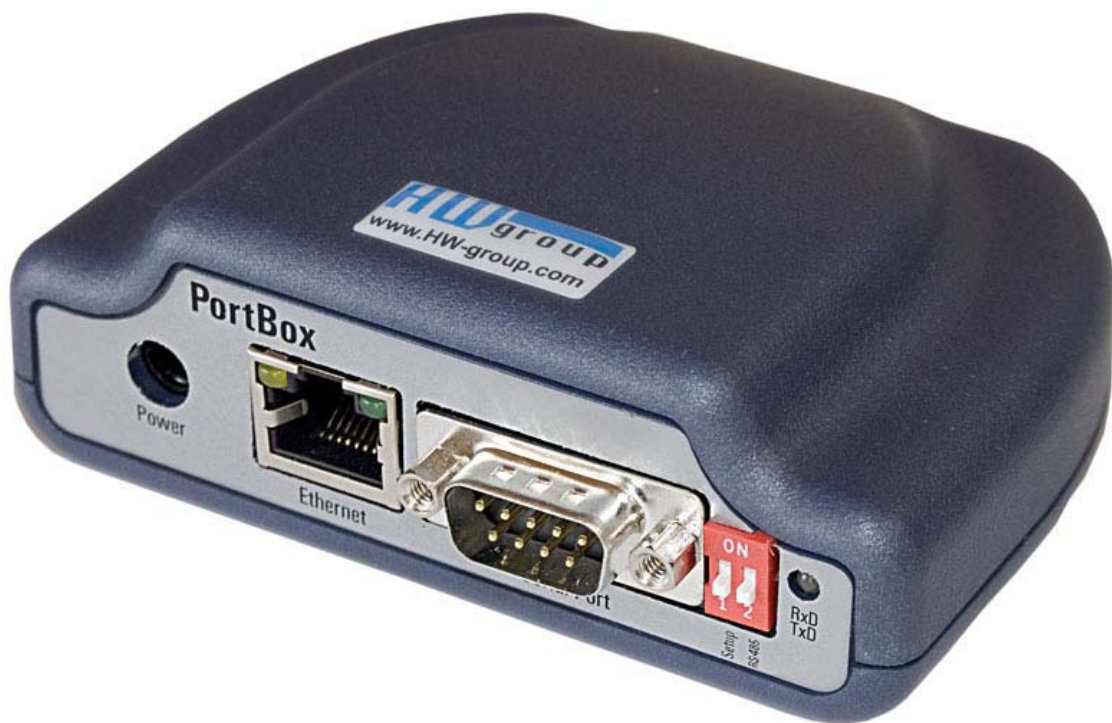


# PortBox PL

## *RS-232/485 - Ethernet Converter*





## Contents

---

The complete shipment includes.....	1
Contents.....	3
Basic features .....	4
Technical parameters.....	5
<b>PORTBox PL – CONNECTOR DESCRIPTION .....</b>	<b>6</b>
Mechanical dimensions.....	7
<b>QUICK SETUP .....</b>	<b>8</b>
Device setup – Frequently asked questions.....	10
<b>HW VSP - VIRTUAL SERIAL PORT .....</b>	<b>11</b>
<b>PORTBox SETUP – COMMANDS DESCRIPTION.....</b>	<b>13</b>
Factory defaults .....	13
Device parameters.....	13
TCP and UDP parameters .....	14
Serial interface parameters .....	16
Other parameters .....	17
UDP/IP mode settings.....	18
Practical setup examples .....	19
Application tips .....	21
Ordering number .....	23
Contact us.....	23

# PortBox PL

## RS-232/485 - Ethernet converter

---

RS-232/485 serial line - Ethernet converter on TCP/IP and UDP/IP protocol. Built-in virtual serial port support, 9-bit support extension. RS-485 question-answer communication support.



### Basic features

---

- 1 serial port **RS-232** or **RS-485** accessible over Ethernet
- The remote port can be controlled with the **Virtual serial port driver for Windows** as for example **COM 5** (the Virtual serial port driver for Windows 2000 / XP free) compatible with RFC2217.
- 10 Mbit/s Ethernet interface - RJ 45 10BASE-T.
- TCP/IP TELNET - NVT (*Network Virtual Terminal*) terminal support.
- **Two devices** can extend („tunnel“) serial port over Ethernet network.

- **The device can be configured in 3 ways:**

Local RS232 (serial terminal) configuration

TCP/IP terminal access on TCP port 99

Simple windows software using UDP Broadcast

- **Data security:**

Predefined IP address and port range – only these devices can access the PortBox.

TCP connection can be authorized using 128 TEA algorithm.

- **Serial interface**

Adjustable communication speed 300..115.200 Bd

Handshake (CTS/RTS, Xon/Xoff, none),

7 to 9-bit parity support (9-bit transfer over Ethernet)

- Wide RS-485 question-answer communication support in the UDP/IP mode.
- Many **software examples and libraries** for MS Visual Basic, Delphi, Borland C++, JAVA, PHP and many others

## Technical parameters

Serial port RS-232	
+ Data bits	7 or 8 or 9
+ Stop bits, Parity	1 or 2, None / Odd / Even / Mark / Space parity
+ Baud rates	50..115.2 kBd – entire range, step = 50 Bd
+ Data flow control	XON/XOFF, CTS/RTS, None
+ Interface	1x DB9M (RxD,TxD,RTS,CTS,GND)
+ Used RS-232 signals	RxD,TxD,RTS,CTS, (DTR output – defined voltage level only)
Serial port RS-485	
+ Termination	No internal termination, external termination 120Ω required
+ Isolation	RS-485 line not optocoupled to the device's power supply
+ RS-485 serial port settings	Half duplex and no HW echo recommended
Ethernet port	
+ Interface	RJ45 (10BASE-T) – 10 Mbit or 10/100 Mbit network compatible only!
+ Compatibility	Ethernet: Version 2.0/IEEE 802.3
+ Supported protocols	IP: ARP, TCP + NVT (Network Virtual Terminal)
+ TCP connection closing	Data - timeout 50s (with NVT - ACK/NOP support)
Environment	
+ Operating temperature	+5 .. 50 °C
+ Storage temperature	-10 to 85 °C
+ Humidity (non-condensing)	5 to 95 %
Physical parameters	
+ Power supply requirements	8-24V / Max. current consumption 200 mA DC - barrel (coaxial) power connector, GND on the shield
+ Dimensions	28 x 105 x 135 [mm] (H x W x D)
+ Weight	395 g
Functional parameters	
Device SETUP configuration options	- over any RS-232 terminal - over TCP/IP using any telnet terminal on the TCP/IP 99 port - basic network parameters via UDP using the Hercules SETUP utility
Diagnostic LEDs	- Power - Link & Activity

## PortBox PL – Connector description

A substantial metal box for wall mounting („L“-shaped metal plates can be ordered separately), DIN moulding, or a standard version (you can order these modifications using a different ordering number).

The RS-232 and RS-485 interface is on a Cannon DB9M connector.

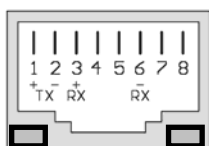


Signalization:

- Power** – green on RJ-45 connector.....External power source
- LINK** – yellow on RJ-45 connector ..... LINK & Ethernet Activity
- RxD** – red ..... Receiving data on RS-232
- TxD** – green ..... Transmitting data on RS-232

### Connectors wiring

Ethernet :



Power LINK

Power source:



DC 8..20V / max. 0.3 A

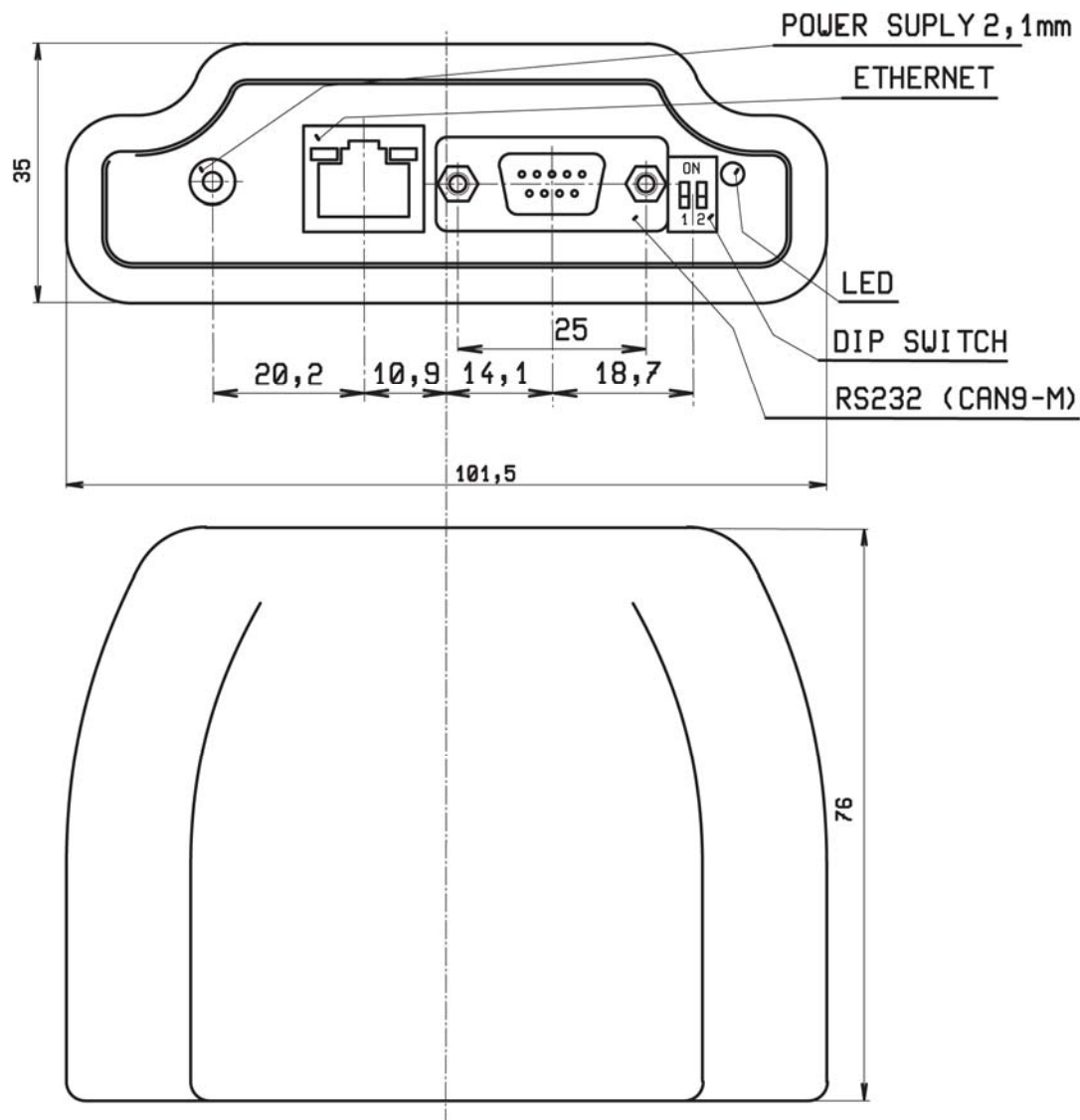
PortBox Port	
Pin	Signal
1 <>	„A“ RS485
2 <-	IN RxD
3 ->	OUT TxD
4 ->	OUT DTR
5 --	GND
6	
7 ->	OUT RTS
8 <-	IN CTS
9 <>	„B“ RS485
Cannon 9 - Male	

Standard IBM PC RS-232 Port	
Pin	Signal
1 <-	CD CD
2 <-	RxD
3 ->	TxD
4 ->	DTR
5 --	GND
6 <-	DSR
7 ->	RTS
8 <-	CTS
9 <-	RI
Cannon 9 - Male	

### DIP switches explanation

DIP 1	DIP 2	DIP function
ON	OFF	<b>RS-232 Setup mode</b> (9600 8N1) <i>Ethernet interface is disabled!</i>
OFF	OFF	Serial port is in <b>RS-232 mode</b>
OFF	ON	Serial port is in <b>RS-485 mode</b> - Check the &R and &H parameters (we recommend &R3 &H1)

## Mechanical dimensions



### Optional accessories for mounting:

- The box is equipped with rubber stands, but the package also includes a quality self-adhesive hook and loop fastener (3M™ Dual Lock™), which is suitable for easy and quick mounting to a flat and non-absorbent surface.
- The box is equipped with rubber stands, but the package also includes a metal holder for wall mounting, allowing...
- The box is equipped with rubber stands, but the package also includes a removable holder for a DIN molding, which is mounted from below into the center of the surface.

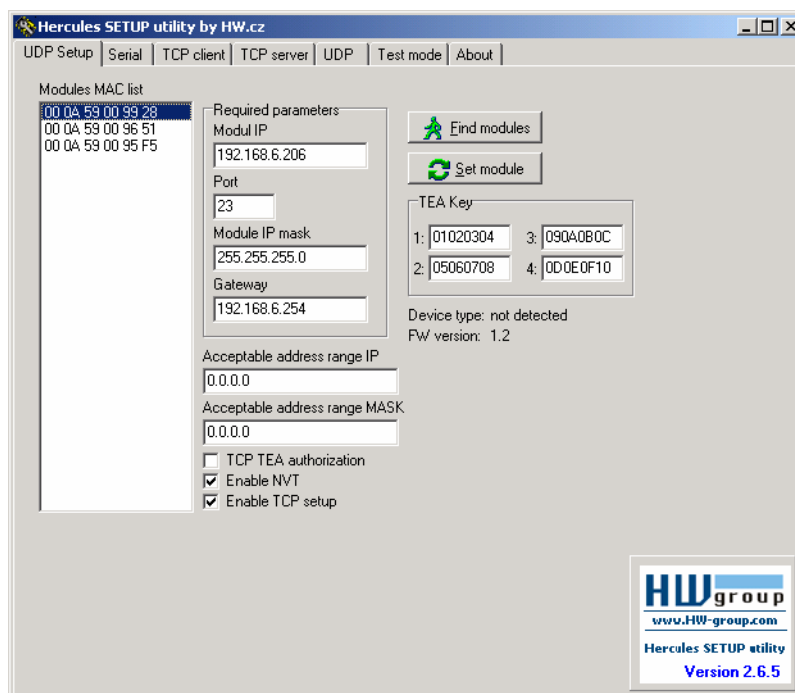
**Note:** These modifications can be ordered using a specific ordering number (see the last page of this manual).

## Quick SETUP

*In this chapter, you will learn, how to configure and ping your device as soon as possible. For more information or in case of any problems please read the Getting started chapter.*

### Connecting cables

- Connect the supplied power adaptor to 230V power source.
- Set **DIP1** and **DIP2** to **OFF** position.
- Connect the PortBox to a **10 Mbit** or 10/100 Mbit Ethernet network.
- Plug the power connector to the PortBox device.
- If the power supply is ok, the green **Power** LED is now lit.
- If the Ethernet connection is ok, the **LINK** LED should light up. This LED will now blink commensurable to data transferred to/from the Ethernet network.



### IP address setup

- Run the „HerculesSetup.exe“ program, that can be found on the supplied CD, or you can download it from our website: [www.HW-group.com](http://www.HW-group.com)
- Click the „Find modules“ button in the „UDP Setup“ tab, you can see the MAC address of the device in the left column. Click on the MAC address and set the desired parameters (you must at least set the IP address, Mask and GateWay).
- Check the „Enable TCP Setup“ box and click on the „Set module“. This will save the network parameters.
- The device’s IP address and network parameters are now set and you can work with the device. You can test the connection using PING to the device’s IP address.

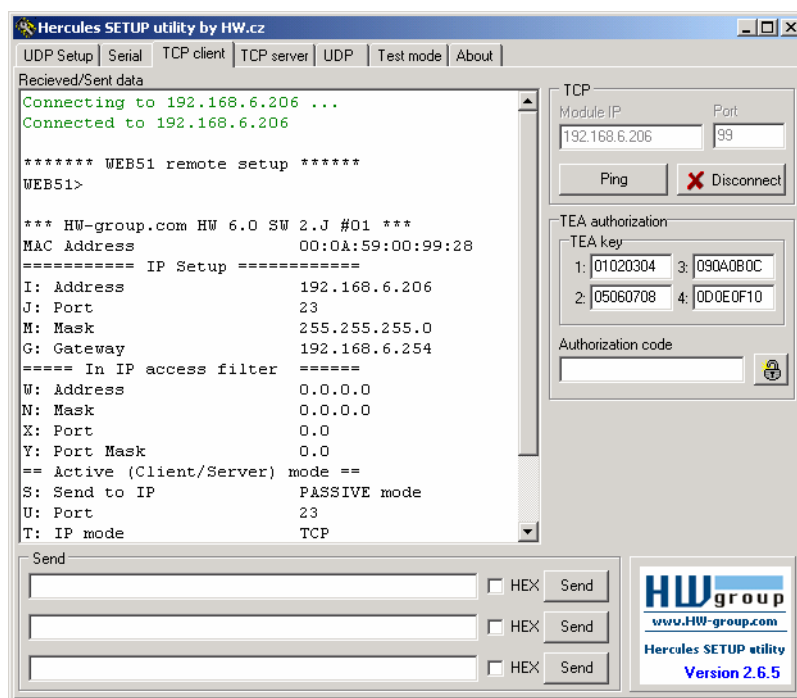
### Are you unable to find the device’s MAC address in the list?

*If the LINK LED is not lit, please check the following:*

- Does your network support **10MBit** communication?
- Are you using the correct TP cable? (TP Patch cable for connecting to switch, TP Crossed cable for connecting directly to a PC)
- Check the DIP switches (they should all be in OFF position).
- Check your power adaptor and the power LED.

## Setting up the device using TCP Setup

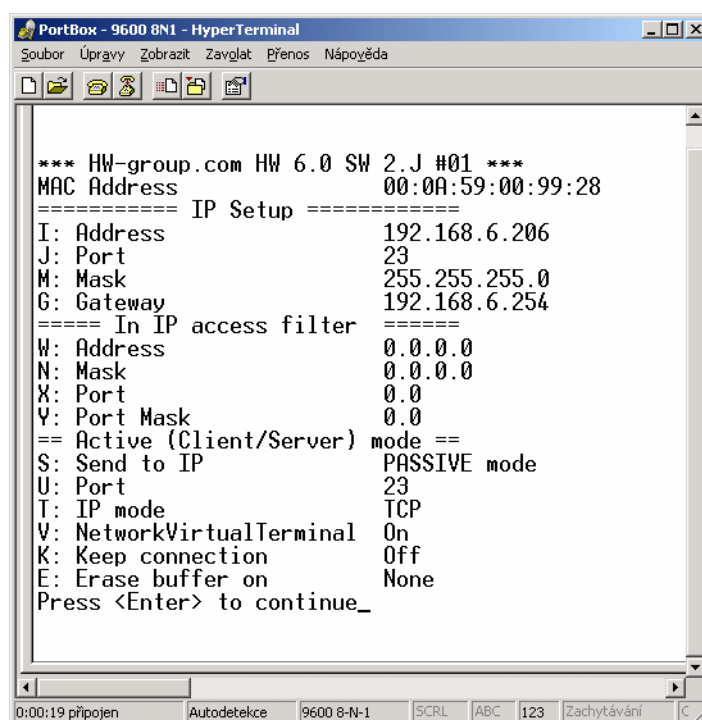
- Click on the „TCP Client“ tab and enter the preset IP address. As a **TCP port number** choose port **99**.
- Click the „Connect“ button, you can see the „**WEB51**“ prompt on the left side. Click in this window and press ENTER. You can now see the parameters listing.
- If you want to change any parameter, you must use the parameter name together with the desired value (for example „**192.168.6.8 <Enter>**“ for setting up the device's IP address). You can see help to any command when you send the commands name together with the question mark (for example „**I? <Enter>**“). You can also find a detailed commands list in another chapter of this manual.
- After you set all the desired parameters, call the „**R**“ function (**Reboot**) and press the „Disconnect“ button to disconnect from the restarting device.



## Setting up the device using RS-232 terminal

If you are not using Windows, or if you don't have the Hercules utility, you can setup the device using any available RS-232 serial terminal.

- Set **DIP1 = ON**, **DIP2 = OFF**,
- Connect the RS-232 cable (Port 1) to the PC using the supplied **LapLink** cable.
- Open your favourite terminal software (for example Hyperterminal), choose the correct serial port and set communication parameters to **9600 8N1**.
- Connect the power supply cable to the PortBox device.
- If the power supply is ok, the green **Power** LED should light up
- If your terminal and connection are working correctly, you should now see the text menu listing.
- From now on, you can continue as in the previous example.



## Device setup – Frequently asked questions

---

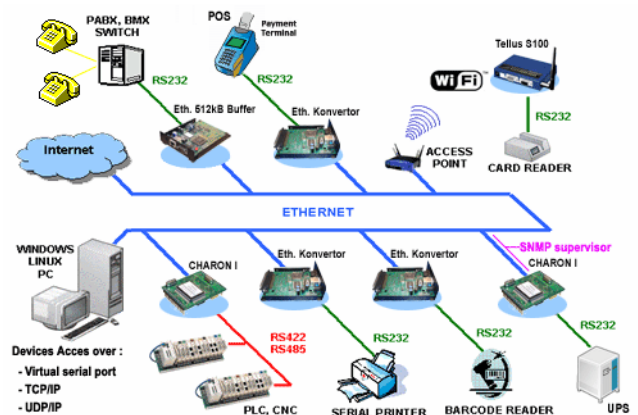
- **Ethernet is not working, but the LINK LED is lit.**  
Didn't you forget to turn off the „RS-232 Setup“ mode? (This mode is enabled when **DIP1** = ON)  
In this mode, the Ethernet part is not working. Set **DIP1** = OFF and restart the device with unplugging it's power source for at least 3 seconds.
- **The RS-485 communication is not working**  
Check if the line is equipped with **termination resistors** (120 – 470 ohm $\dot{u}$ ). If you are using the RS-485 conversion, set the configuration to **&R2** or **&R3** and dont forget to enable HALF DUPLEX using **&H1**.
- **I nee power supply for my RS-232 application**  
If you do not need a flow control (HW handshake) but a power source on your serial port (up to 5 –10 mA), you can power your application from the RTS output (pin 7 on the RS-232 connector). You can enable +8V to +12V power on this pin with setting **&R0** in the setup mode (&R: RS485/RS422 control).

## HW VSP - Virtual Serial Port



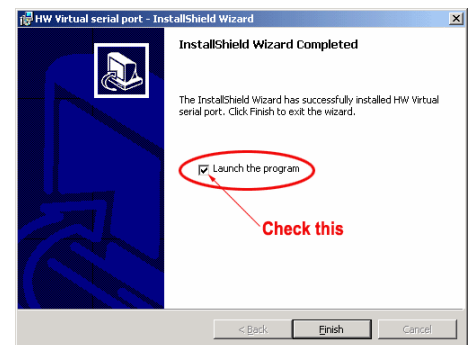
The Virtual Serial Port Driver for Windows is software that creates a virtual serial port (for example COM5) and transfers all data from that port to another hardware device. It is mostly used for connecting RS-232 interface over USB or Ethernet.

- Works with Windows 2000 and Win XP.
- If the device supports RFC 2217 (NVT), you can setup parameters of the remote serial port (speed, parity, stop bits).
- Debugging is possible by logging the communication to a LOG file.
- You can create more Virtual Serial Ports on one PC (COM5, COM6, COM7) with running VSP.EXE with command line parameters



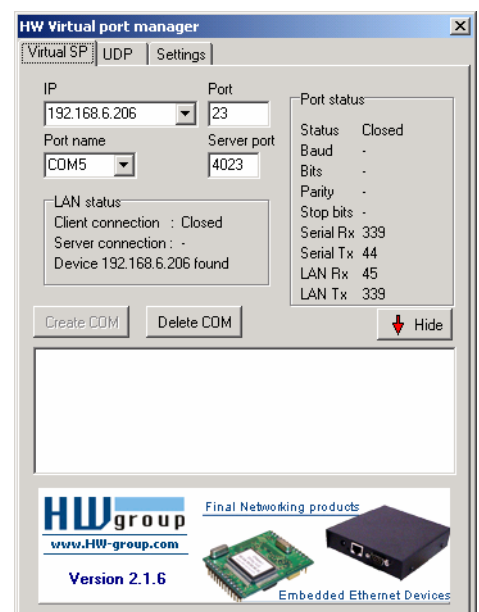
### Using HW VSP with the PortBox

- Install HW VSP (the “**HW VirtualSerialPort**” directory on our CD). Don't forget to check the “Launch the program” checkbox to finish the installation. You need to restart the PC after the installation is finished.
- Check if NVT is enabled (using **UDP Setup** from the **Hercules** utility). If not, enable NVT and save setup.
- Run the HW VSP and find the PortBox device in the „**UDP**“ tab. Select the device's MAC address and click „**Use this IP**“. Switch to the “**Virtual SP**“ tab, the IP and port should be filled in correctly.



This search uses UDP broadcast and works on local network only. If you are using router or any other similar device, you must fill in the IP address and Port manually in the „Virtual SP“ tab.

- Choose which port you want to create (COM1 – COM20) and click the „Create COM“ button. You can see the result in the „LAN status“ window. If the device has been found, the serial port has been created.
- If you run any application that opens the selected port (COM5 in this case), the HW VSP driver establishes a connection with the PortBox, sets the remote port according to local setup and starts a data transfer. Some applications may have problems with port number above COM4, but you can use the original Hyperterminal from Windows, serial tab from the Hercules utility, or a slovenian software „Terminal.exe“, that can be found on our CD.



## HW VSP setup options

- **log. enabled**

The VSP driver creates the "C:\serialport.log" file for VSP activity logging.

- **TCP server mode**

Activates VSP in TCP/IP server mode. The driver then acts as a TCP Client/Server device = the side, which accepts any data, starts acting as a client and opens a connection.

The input port of the TCP server can be set in the „**Virtual SP**“ tab. We recommend using numbers above 1025.

***Note:** For most applications, the TCP server mode is not necessary and you can use the TCP client mode (leave the TCP server mode unchecked).*

- **Create port on start VSP**

If this box is checked, all the virtual port will be created when VSP is started. If you need to create all the ports on windows startup, you also need to check the "Start VSP on boot" checkbox.

- **Hide on startup**

Hides the VSP icon to system tray.

- **Don't create port if ping fails**

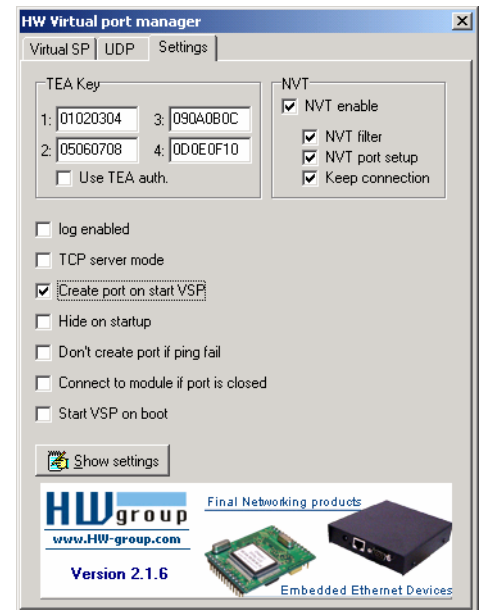
VSP tests the device's IP address on startup, and if the test is unsuccessful, the port will not be created.

- **Connect to module if port is closed**

If the virtual serial port is not used by an application, you can lose data when this box is checked.

- **Start VSP on boot**

inserts the HW VSP path to the Windows registry (HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\run). This makes the VSP start with Windows.



## HW VSP – command line parameters

The HW VSP can be also run using command line parameters. This way you can create several virtual ports on one PC. You can find the detailed explanation on our website: [www.HWgroup.cz](http://www.HWgroup.cz)

**Example:** CharonVirtualCom.exe -R -i192.168.6.21:23 -c5 -S0 -N1 -Nf -Np -H1

## PortBox setup – commands description

Here you can find the detailed description of all the possible setup parameters.

### Factory defaults

This setup can be restored with „D0“ or „D1“ commands using the TCP setup on port 99, or using the RS-232 Setup on 9600 8N1.

```

*** HW-group.com HW 4.7 SW 2.K #01 ***
MAC Address                00:0A:59:XX:XX:XX
===== IP Setup =====
I: Address                  192.168.6.41
J: Port                     23
M: Mask                     255.255.255.0
G: Gateway                  192.168.6.254
===== In IP access filter =====
W: Address                  0.0.0.0
N: Mask                     0.0.0.0
X: Port                     0.0
Y: Port Mask                0.0
== Active (Client/Server) mode ==
S: Send to IP               PASSIVE mode
U: Port                     4023
T: IP mode                  TCP
V: NetworkVirtualTerminal  Off
K: Keep connection          Off
E: Erase buffer on          None
===== Serial Setup =====
&B: Speed                   9600
&D: Data bits                8
&P: Parity                   NONE
&V: Variable Parity          Off
&S: Stop bits                1
&C: Flow Control             NONE
&R: RS485/RS422 control      RTS = On [+8V]
&T: Serial Line Timeout     0 - Off
&G: Char. Transmit Delay    0 - Off
&H: Tx Control               Tx FULL duplex
&M: Serial buffer size       Medium Rx / Medium Tx
&O: Buffer SpaceCompresion  Off
===== Security Setup =====
%A: TCP autorisation         Off
%K: TEA key 0:01:02:03:04 1:05:06:07:08 2:09:0A:0B:0C
3:0D:0E:0F:10
%S: TCP/IP setup             On
===== Other =====
D: Load/Save Settings from/to Flash
R: Reboot

WEB51>

```

### Device parameters

**MAC Address**                      **00:0A:59:00:95:6C**

MAC address is a unique network device address in the Ethernet and it is always factory-preset. You can find it on the label inside the device. Using this address, the devices can be distinguished for example in the UDP mode of the configuration program. The address respects restoring of the default configuration with the „D0“ command.

**I: Address**                      **192.168.6.15**

Configuration of the PortBox's IP address.

**J: Port**                              **23**

Configuration of the PortBox's communication port – range: 1 .. 19.999.

Port 99 is used for TCP configuration, if supported by the version and enabled in the setup.

**M: Mask**                              **255.255.255.0**

Configuration of the IP mask for the local network. All IP addresses outside the area delimited by the PortBox's own IP address and this mask will be accessed via the Gateway.

**G: Gateway**                      **192.168.6.254**

Address of the gateway that provides access to outside networks, as defined by the IP address and the mask.

**IP address, Mask and Gateway description:**

Ethernet device communicates :

- **On local network segment**  
You don't need a Gateway, but IP addresses of both Aires must be in the range allowed by the Mask setting. If you set the Mask to 255.255.255.0 the IP addresses may differ only in the last byte.
- **Outside local network – using Gateway**  
(which must be in the range allowed by Mask parameter).

You can also restrict the IP address range using the „In IP Setup“ commands. We recommend keeping this parameter on 0.0.0.0. for debugging.

**===== In IP Setup =====****W: Address**                      **0.0.0.0**

IP address of a network or computer that is allowed to communicate with the PortBox. This value must result from multiplying the remote IP address and the restriction mask (option N), otherwise the PortBox does not react.

**N: Mask**                              **0.0.0.0**

This mask restricts addresses that can communicate with the PortBox. Security can be greatly enhanced by setting a fixed address or a suitable restrictive mask that disallow communication with unauthorized parties.

**X: Port**                              **0.0****Y: Port Mask**                      **0.0**

This restricts TCP ports that can communicate with the PortBox.

**(IP requesting access AND N ) = W**

If this condition is valid, you can access the device (AND is binary multiplication).

**TCP and UDP parameters****===== Out IP Setup =====****S: Send to IP**                      **192.168.0.252**

Defines the IP address which will the PortBox communicate with (useful on large networks)

**U: Port 23**

Remote IP address and port for establishing a connection (when any data comes from the serial port). Value 0.0.0.0 switches the PortBox into **passive mode**.

**Note:** *If UDP communication is used, a remote address must be specified here. The PortBox does not establish connections, responses are sent to this address only!*

**T: IP mode TCP**

Switches between the TCP and UDP protocols. UDP is faster but packets can be lost or delivered out of order. Hence it is suitable for communication only on a local network segment in request-reply mode, usually for converting a RS-485 communication. UDP communication is difficult to debug since there is no simple PC terminal (like TELNET for TCP/IP). You can use the NVT and TEA encryption. If you are using a question-answer protocol, please note that in some cases the TCP mode can give up to 40% slower responses (the connection must be established).

**IP mode: 0: TCP / 1: UDP**

**V: NetworkVirtualTerminal Off**

Network Virtual Terminal allows the interpreting of Telnet protocol sequences including certain RFC2217 extensions, enabling on-the-fly changes of serial port parameters (speed, parity, ...). NVT description is available in the "**Programming Ethernet Applications**" guide on our website.

When communicating with the serial port using telnet, e.g. with the TeraTerm program, this option should be turned on. Otherwise, telnet control commands (seen as "junk") intended for configuration negotiation at the beginning of the communication are forwarded to the serial port. If you don't want to use this option, set your client to RAW communication mode.

**0: Off** (don't use telnet control code, pass through to serial port)

**1: On** (accept telnet control code)

**K: Keep connection Off**

This option allows keeping the connection alive with sending NOP commands, because TCP automatically closes the connection after 50 seconds of inactivity. NVT must be enabled for this parameter to work.

**0: do not keep connection** (preferred)

**1: keep connection**

**E: Erase buffer on Open connection**

Option to clear the internal PortBox buffer whenever a connection is established or closed. This option is useful e.g. if your device periodically says "I'm alive" and you don't want to waste time retrieving these notifications from the buffer.

**0: none**

**1: Close TCP/IP connection**

**2: Open TCP/IP connection**

**3: Open & Close TCP/IP connection**

## Serial interface parameters

### ==== Serial Setup =====

#### &B: Speed **9600**

Configuration of the communication speed for the serial line, range 50..115.200 Bd, step 50 Bd. To set 9600 Bd enter: „&B9600“.

#### &D: Data bits **8**

Number of data bits for the serial transfer. Options are:

**7: 7 bits / 8: 8 bits** – enter „&D8“

#### &P: Parity **NONE**

Parity of the serial asynchronous communication:

**N: none / O: odd / E: even /**

**M: mark / S: space**

#### &S: Stop bits **2**

Number of stop bits for the RS-232 serial line. You can set 1 or 2 stop bits.

#### &C: Flow Control **NONE**

Serial port data flow control.

**1: none** - no control, RTS (see. &R).

**2: RTS/CTS** – RTS/CTS pin control

**3: Xon/Xoff** - SW flow control.

#### &R: RS485/RS422 control

Defines idle level of the output RTS pin. Important for devices powered from RTS or for add-on RS485 PortBoxes that use RTS to switch direction. Especially for

the internal RS485 module, the "**HW echo**" option should be on. This means that the receiver reads the data back from RS485 and generates hardware echo from the actual RS485 bus.

**0: RTS = Low [+8V] (recommended for non RS485/422 mode)**

**1: RTS = High [-8V]**

**2: TxRTS HW echo ON**

**3: TxRTS HW echo OFF**

**Note:** For most RS-485 devices, set &R=3 TxRTS HW echo OFF.

#### &T: Serial Line Timeout **0 – Off**

Specifies how long the PortBox waits **after receiving the last character** before wrapping up the data in a packet and transmitting them. The timeout is specified as the **number of characters** and displayed as the number of chars as well as the time according to the given serial communication speed. If the speed changes, the time is also changed but the number of characters defining the timeout is not (10 characters at 9600 Bd = about 11 ms, or 5.7ms at 19,200 Bd).

```

Tera Term - COM1 VT
File Edit Setup Control Window Help
WEB51> I192.168.6.15

***** WEB51 v2.3 *****
MAC Address 00:0A:59:00:95:6C
===== IP Setup =====
I: Address 192.168.6.15
J: Port 23
M: Mask 255.255.255.0
G: Gateway 192.168.0.1
===== In IP Setup =====
W: Address 0.0.0.0
N: Mask 0.0.0.0
===== Out IP Setup =====
S: Send to IP 192.168.0.9
U: Port 23
T: IP mode TCP
V: NetworkVirtualTerminal On
K: Keep connection Off
E: Erase buffer on Open connection
Press <Enter> to continue
===== Serial Setup =====
&B: Speed 9600
&D: Data bits 8
&P: Parity NONE
&S: Stop bits 1
&C: Flow Control NONE
&R: RS485/RS422 control RTS = Low [+8V]
&T: Serial Line Timeout 0 - Off
&G: Char. Transmit Delay 0 - Off
&H: Tx Control Tx FULL duplex
&O: Buffer SpaceCompression Off
===== Security Setup =====
%A: TCP autorisation Off
%K: TEA key 0:01:02:03:04 1:05:06:07:08 2:09:0A:0B:0C
%S: TCP/IP setup On
===== Other =====
D: Load/Save Settings from/to Flash
R: Reboot
WEB51> J2323

```

**&G: Char. Transmit Delay 0 – Off**

For controlling units with small RS232 buffer, it is sometimes advantageous to keep relatively high baud rate but insert delays between individual characters. The delay is defined in **milliseconds** and it is defined as the time between the starts of individual characters; so, for 2400 Bd a 2ms delay has no effect since the character starts are 2.4 ms apart.

**&H: Tx Control Tx FULL duplex**

When HALF duplex is activated, the PortBox assumes unidirectional medium connected to the serial line (e.g. RS485) and won't start transmitting data while receiving.

- 0: FULL duplex**
- 1: HALF duplex (RS485)**

**&M: Serial buffer size Medium Rx / Medium Tx**

Sets the size of the ring buffer in the PortBox memory for individual directions. For example, to collect data from the serial line, it is advantageous to use option **"2: High Rx / Low Tx"**.

- 0: Medium Rx / Medium Tx** (approx. 50% / 50% of memory)
- 1: Low Rx / High Tx** (approx. 30% / 70% of memory)
- 2: High Rx / Low Tx** (approx. 70% / 30% of memory)

**Other parameters****Q: Quiet (Batch) mode**

If you need to manage device's configuration from your SW application, you have to work with the list of the device's settings. It can be helpful the quiet mode, you can enable it with **„Q1“**. In the Quiet mode, device reply to pressing Enter not in 2 pages parameters listing but under example:

```
WEB51=2.L=00:0A:59:00:A6:08;I192.168.1.24;J23;M255.255.255.0;G192.168.1.1;W0.0.0.0;N0
.0.0.0;X0.0;Y0.0;S192.168.6.51;U4023;T0;V0;K1;A250;&B9600;&D8;&P1;&S1;&C1;&R0;&T0;&G0
;&H0;%A0;%K01:02:03:04:05:06:07:08:09:0A:0B:0C:0D:0E:0F:10;%S1;#T3;#A3;#B192;#C3;#D0;
#X0;#Y0;#Z3;#W0;*L1;*P0;*S58.0.0.0;*M255.0.0.0;*X10.0.0.0;*Y255.0.0.0;*E999
```

**==== Security Setup =====****%A: TCP autorisation Off**

Activates TEA authorization - requested from the remote side after the connection is established.

- 0: TEA authorisation Off**
- 1: TEA authorisation On**

**%K: TEA key 0:01:02:03:04 1:05:06:07:08 2:09:0A:0B:0C 3:0D:0E:0F:10**

To set the TEA key, use the "%K" option. Set 16 bytes in quadruples using four hexadecimal values separated with colons. The first string defines 0-3rd quadruple of bytes. So, to set the last 4 bytes to the displayed value, use **"%K 3:0D:0E:0F:10"**.

**===== Other =====****D: Load/Save Settings from/to Flash**

**"D0"** loads settings saved by using the **„D2“** command. **"D1"** reloads factory settings.

**R: Reboot**

Soft restart of the PortBox. Necessary after changing the IP address, etc.

## UDP/IP mode settings

If you select „T: IP mode UDP“, the PortBox will communicate with the remote side using unacknowledged UDP packets. Also, the following menu appears in the setup.

### ==== Trigerring Setup =====

**\*L: Trigger Length            1**

Number of bytes of the starting and ending packet trigger condition. Allowed values are from 0 to 4. If the lengths of your start and stop triggers differ, use the trigger mask and don't forget to include the masked characters in the lengths - even though they contain actual frame data.

**\*P: Post Trigger Length    0**

In some protocols, checksum or other info follows the stop trigger. This value defines the number of characters after the stop trigger that should be included in the packet. If the start and stop triggers are equal, this value specifies packet length without the 0..4 bytes of start trigger.

**\*S: Start Trigger Pattern    58.0.0.0**

**(The 4 input bytes of data AND \*M ) = \*S**

Start trigger for packet transmission. Four bytes are set, but only the number of bytes specified in "L: Trigger Length" is considered

**\*M: Start Trigger Mask      255.0.0.0**

Mask of the start trigger. Masking works similarly to the Ethernet netmasks, using a bitwise AND. Value of 255 means that the tested character must be equal to the character specified in "\*S: Start Trigger Pattern". For example, to start the transfer with any control ASCII character (0..31d), use 0.0.0.0 for the trigger pattern, 224.0.0.0 for the mask, and 1 for the length. If you set both the character and the mask to 0, the trigger activates for any character.

**\*X: Stop Trigger Pattern    10.0.0.0**

**(The 4 input bytes of data AND \*Y ) = \*X**

Sets the stop trigger for sending data to the Ethernet.

**\*Y: Stop Trigger Mask      255.0.0.0**

Mask of the stop packet trigger for serial line data. For example, the settings displayed here are intended for transferring data in the IntelHEX format over RS485. The start trigger is a colon and the transfer is terminated after receiving the control character <LF> (0Ah = 10d).

**\*E: Max. Start-Stop Length   999**

Maximum number of characters sent after the START trigger, if the STOP trigger is not found sooner. After transmission, another START trigger is expected. Essentially, this is a "timeout" specified as the number of characters

### Setup example

*If you need to send all data from the serial line to the opposite side, use:*

*L: Trigger Length	0
*P: Post Trigger Length	0
*S: Start Trigger Pattern	0.0.0.0
*M: Start Trigger Mask	0.0.0.0
*X: Stop Trigger Pattern	255.0.0.0
*Y: Stop Trigger Mask	255.0.0.0
*E: Max. Start-Stop Length	200

## Practical setup examples

The following examples show some typical setup settings for the most common applications.

### 2x PortBox devices connected to each other (UDP/IP)

A virtual serial line prolonging over TCP/IP, 19.000 Bd, 8N1, no handshake

#### PortBox 1 (IP = 192.168.1.41)

```
*** HW-group.com HW 4.7 SW 3.0.4 #01 ***
***          PortBox          ***

===== IP Setup =====
I: Address          192.168.1.41
J: Port             4023
M: Mask             255.255.255.0
G: Gateway          192.168.1.1
===== In IP access filter =====
W: Address          0.0.0.0
N: Mask             0.0.0.0
X: Port             0.0
Y: Port Mask        0.0
== Active (Client/Server) mode ==
S: Send to IP      192.168.1.51
U: Port             4023
T: IP mode          UDP
E: Erase buffer on None
A: Max. Packet Length 250
Press <Enter> to continue
===== Serial Setup =====
&B: Speed           19200
&D: Data bits       8
&P: Parity           NONE
&V: Variable Parity Off
&S: Stop bits       1
&C: Flow Control    None
&T: Serial Line Timeout 0 - Off
&G: Char. Transmit Delay 0 - Off
&H: Tx Control      Tx FULL duplex
%S: TCP/IP setup    On
===== Triggerring Setup =====
*L: Trigger Length  0
*P: Post Trigger Length 0
*S: Start Trigger Pattern 0.0.0.0
*M: Start Trigger Mask  0.0.0.0
*X: Stop Trigger Pattern 255.0.0.0
*Y: Stop Trigger Mask   255.0.0.0
*E: Max. Start-Stop Length 200
===== Other =====
D: Load/Save Settings from/to Flash
R: Reboot

WEB51>
```

#### PortBox 2 (IP = 192.168.1.51)

```
*** HW-group.com HW 4.7 SW 3.0.4 #01 ***
***          PortBox          ***

===== IP Setup =====
I: Address          192.168.1.51
J: Port             4023
M: Mask             255.255.255.0
G: Gateway          192.168.1.1
===== In IP access filter =====
W: Address          0.0.0.0
N: Mask             0.0.0.0
X: Port             0.0
Y: Port Mask        0.0
== Active (Client/Server) mode ==
S: Send to IP      192.168.1.41
U: Port             4023
T: IP mode          UDP
E: Erase buffer on None
A: Max. Packet Length 250
Press <Enter> to continue
===== Serial Setup =====
&B: Speed           19200
&D: Data bits       8
&P: Parity           NONE
&V: Variable Parity Off
&S: Stop bits       1
&C: Flow Control    None
&T: Serial Line Timeout 0 - Off
&G: Char. Transmit Delay 0 - Off
&H: Tx Control      Tx FULL duplex
%S: TCP/IP setup    On
===== Triggerring Setup =====
*L: Trigger Length  0
*P: Post Trigger Length 0
*S: Start Trigger Pattern 0.0.0.0
*M: Start Trigger Mask  0.0.0.0
*X: Stop Trigger Pattern 255.0.0.0
*Y: Stop Trigger Mask   255.0.0.0
*E: Max. Start-Stop Length 200
===== Other =====
D: Load/Save Settings from/to Flash
R: Reboot

WEB51>
```

**2x PortBox devices connected to each other (TCP/IP)**

*A virtual serial line prolonging over TCP/IP, 19.000 Bd, 8N1, HW handshake*

**PortBox 1**

```
*** HW-group.com HW 4.7 SW 2.K #01 ***
===== IP Setup =====
I: Address          192.168.1.1
J: Port            4023
M: Mask            255.255.255.0
G: Gateway         192.168.1.254
===== In IP access filter =====
W: Address          0.0.0.0
N: Mask            0.0.0.0
X: Port            0.0
Y: Port Mask       0.0
== Active (Client/Server) mode ==
S: Send to IP      192.168.1.2
U: Port            4023
T: IP mode         TCP
V: NetworkVirtualTerminal On
K: Keep connection Off
E: Erase buffer on None
===== Serial Setup =====
&B: Speed          19200
&D: Data bits      8
&P: Parity         NONE
&V: Variable Parity Off
&S: Stop bits      1
&C: Flow Control   Xon/Xoff
&R: RS485/RS422 control RTS = On [+8V]
&T: Serial Line Timeout 0 - Off
&G: Char. Transmit Delay 0 - Off
&H: Tx Control     Tx FULL duplex
&M: Serial buffer size Medium Rx / Medium Tx
&O: Buffer SpaceCompression Off
===== Security Setup =====
%A: TCP autorisation Off
%K: TEA key 0:01:02:03:04 1:05:06:07:08
2:09:0A:0B:0C 3:0D:0E:0F:10
%S: TCP/IP setup   On
===== Other =====
D: Load/Save Settings from/to Flash
R: Reboot
```

**PortBox 2**

```
*** HW-group.com HW 4.7 SW 2.K #01 ***
===== IP Setup =====
I: Address          192.168.1.2
J: Port            4023
M: Mask            255.255.255.0
G: Gateway         192.168.1.254
===== In IP access filter =====
W: Address          0.0.0.0
N: Mask            0.0.0.0
X: Port            0.0
Y: Port Mask       0.0
== Active (Client/Server) mode ==
S: Send to IP      192.168.1.1
U: Port            4023
T: IP mode         TCP
V: NetworkVirtualTerminal On
K: Keep connection Off
E: Erase buffer on None
===== Serial Setup =====
&B: Speed          19200
&D: Data bits      8
&P: Parity         NONE
&V: Variable Parity Off
&S: Stop bits      1
&C: Flow Control   Xon/Xoff
&R: RS485/RS422 control RTS = On [+8V]
&T: Serial Line Timeout 0 - Off
&G: Char. Transmit Delay 0 - Off
&H: Tx Control     Tx FULL duplex
&M: Serial buffer size Medium Rx / Medium Tx
&O: Buffer SpaceCompression Off
===== Security Setup =====
%A: TCP autorisation Off
%K: TEA key 0:01:02:03:04 1:05:06:07:08
2:09:0A:0B:0C 3:0D:0E:0F:10
%S: TCP/IP setup   On
===== Other =====
D: Load/Save Settings from/to Flash
R: Reboot
```

- If you wish to allow only one IP address for the device to communicate with, set N:255.255.255.255 and W:IP address of the other side.
- If you want a 9-bit communication, set &V1 = &V: Variable Parity On on both devices.
- If you are transferring short messages and you need to reduce response times, set &t10 = &T: Serial Line Timeout 10 on both devices.
- If you want to increase security of the communication, turn off TCP setup on port 99 on both devices: %S0 = %S: TCP/IP setup Off

**IP Filter**

Use the following settings if you want a more secured communication. This will allow PortBox 1 to communicate on local network segment only (0 to 255 at the end of the IP address)

```
I: Address          192.168.1.1
M: Mask            255.255.255.0
===== In IP access filter =====
W: Address          192.168.1.0
N: Mask            255.255.255.0
S: Send to IP      192.168.1.2
```

```
I: Address          192.168.1.2
M: Mask            255.255.255.0
===== In IP access filter =====
W: Address          192.168.1.0
N: Mask            255.255.255.0
S: Send to IP      192.168.1.1
```

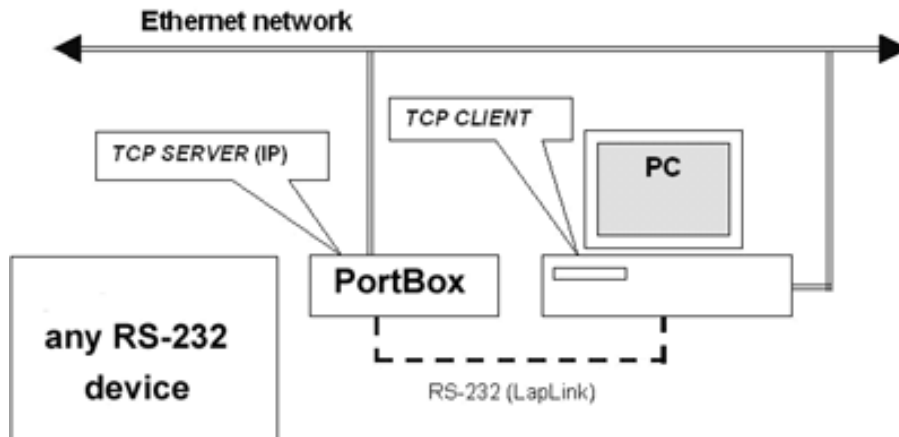
# Application tips

## Communication test

In this example we'll simulate a real application and test the device functionality in connection with the PC and network.

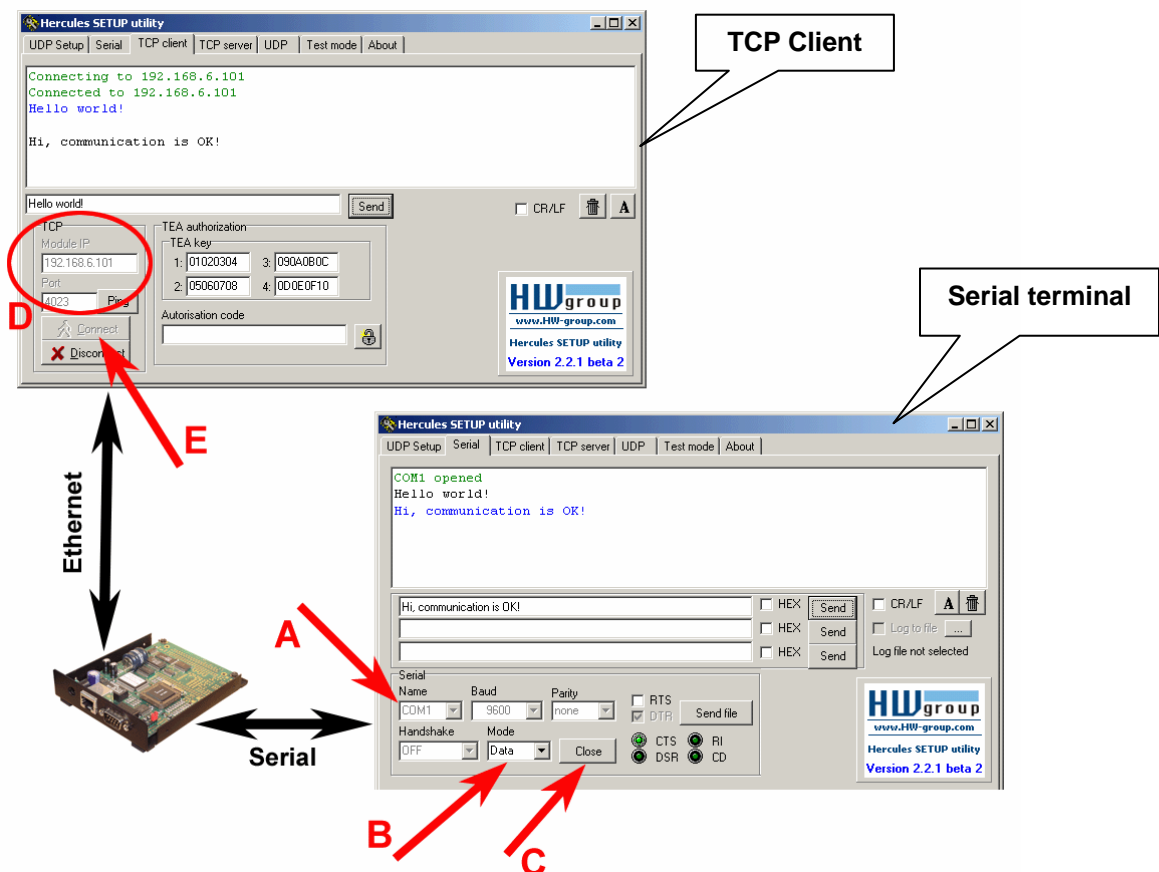
Connect your **PC's serial port to the PortBox** using a LapLink serial cable. Connect your PC and the PortBox to the Ethernet network, similar to scheme in *Figure 6*.

For the serial port communication, open the PC COM1 serial port in any Serial Terminal window (e.g. **Hercules – tab "Serial"**). For the network communication with the PortBox device server, let's use a TCP client application (run 2nd Hercules application, in the **"TCP Client"** tab).



**Figure 6:** Diagram of connecting the devices

Connect both windows and if everything is OK, you should see the data that you are typing in the "Serial terminal" tab appear in the "TCP Client" and vice versa.



**Figure 7:** Communication test setup

### Detailed description of settings:

Serial terminal - the **Serial** tab :

- A - Serial: Name (e.g. COM1), 9600 Baud 8N1 – no handshake (*Figure 7 - detail A*)
- B - Mode: Data (*Figure 7 - detail B*)
- C - Click the Open button (*Figure 7 - detail C*)

TCP Client (**TCP client** tab):

- D - TCP: Modul IP and Port of your device (*Figure 7 - detail D*)
- E - Click the Connect button (*Figure 7 - detail E*)

## Security

For a more secured communication, TEA authorization can be requested upon every TCP/IP connection. This is available only for the TCP/IP connection and the password is not sent through the network. UDP mode is for the local network only. The detailed description of the TEA can be found in the “**Ethernet Applications Programming**” book on our website.

## Timeout prolonging

The PortBox support only one connection at a time, therefore you need to define a timeout for this connection. The standard timeout is 50 seconds. In most cases you can enable the „**K: Keep connection**“, which will send a NVT NOP command every 10 seconds. This will keep the connection open.

## 9-bit communication

If you need to transfer the 9<sup>th</sup> parity bit and you are using 2 PortBoxes in the TCP mode – set „**Virtual Parity**“ **On**. The parity bit will be transferred with the data flow.

## Packet definition in UDP mode

UDP mode supports RS-485 communication protocols. If you setup the correct parameters, you can reduce response times from approx. 0 - 100 ms in the TCP/IP mode to 10-20 ms.

The PortBox can determine a packet end on the RS-485 line and send the data without waiting for the end of the timeout. The UDP mode setting chapter define RS-485 data transfer end, when the packet is sent. If some conditions collide, the packet is sent when any of them is activated. The start and end of the packet are defined by 1-4 byte sequence with mask.

## Network Virtual Terminal

You should learn to use the Network Virtual Terminal in your application. NVT is basically a TELNET protocol system. It's a simple and transparent way to transfer setup information, compatible with RFC standards (for example **RFC2217** for remote serial port control).

## ing number

OID	Device types
600 293	<b>PortBox PL</b> PortBox PL device, standalone unit, no power adaptor, cabling.
600 294	<b>PortBox PL set</b> Recommended starting set, it contains: - <b>PortBox PL</b> [600293] - EU Power adaptor [600080] - DB9 LapLink cable 2m [600063] - Manual, CD
600 025	<b>DIN Rail box set</b> for mounting to a DIN molding
600 295	Metal holder for wall mounting
600 301	„ <b>Dual Lock fastener</b> “ Self-adhesive hook and loop fastener 3M™ Dual Lock™

## Contact us

---

### HW group or HW server

U Pily 103/3  
Praha 4  
143 00

Tel. +420 222 511 918  
Fax. +420 222 513 833  
<http://www.HW-group.com/>